Artificial intelligence for real-time decoding of motor commands from ECoG of disabled subjects for chronic brain computer interfacing

Maciej Sliwowski

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

École doctorale : EEATS - Electronique, Electrotechnique, Automatique, Traitement du Signal (EEATS)
Spécialité : Signal Image Parole Télécoms
Unité de recherche : Laboratoire d'Electronique et de Technologie de l'Information (LETI - CEA)

Intelligence artificielle pour le décodage de commandes motrices de sujets handicapés, grâce à des interfaces cerveau-machine à usage chronique

Artificial intelligence for real-time decoding of motor commands from ECoG of disabled subjects for chronic brain computer interfacing

Présentée par :

Maciej SLIWOWSKI

Direction de thèse :

Antoine SOULOUMIAC INGENIEUR HDR, Université Paris-Saclay, CEA, List,	Directeur de thèse
Tetiana AKSENOVA INGENIEUR HDR, CEA CENTRE DE GRENOBLE	Co-directrice de thèse

Rapporteurs :

Martin BOGDAN
PROFESSEUR, Universität Leipzig
Sebastian STOBER
PROFESSEUR, Otto-Von-Guericke Universität Magdeburg

Thèse soutenue publiquement le **2 décembre 2022**, devant le jury composé de :

Martin BOGDAN PROFESSEUR, Universität Leipzig	Rapporteur
Sebastian STOBER PROFESSEUR, Otto-Von-Guericke Universität Magdeburg	Rapporteur
Blaise YVERT DIRECTEUR DE RECHERCHE, INSERM DELEGATION AUVERGNE-RHONE-ALPES	Président
Tetiana AKSENOVA INGENIEUR HDR, CEA CENTRE DE GRENOBLE	Co-directrice de thèse
Bertrand RIVET MAITRE DE CONFERENCES HDR, GRENOBLE INP	Examinateur
François CABESTAING PROFESSEUR DES UNIVERSITES, UNIVERSITE DE LILLE	Examinateur

Invités :

Antoine SOULOUMIAC
INGENIEUR HDR, Université Paris-Saclay, CEA, List,
Pierre BLANCHART
INGENIEUR DOCTEUR, Université Paris-Saclay, CEA, List,

Artificial intelligence for real-time decoding of motor commands from ECoG of disabled subjects for chronic brain-computer interfacing

A DISSERTATION PRESENTED
BY
MACIEJ ŚLIWOWSKI

FOR THE DEGREE OF
DOCTEUR DE L'UINVERISTÉ GRENOBLE ALPES
IN THE SUBJECT OF
SIGNAL IMAGE PAROLE TELECOMS

UNIVERSITÉ GRENOBLE ALPES
GRENOBLE, FRANCE
DECEMBER 2022

Artificial intelligence for real-time decoding of motor commands from ECoG of disabled subjects for chronic brain-computer interfacing

Abstract

Brain-computer interfaces (BCIs) may significantly improve tetraplegic patients' quality of life by creating an alternative communication path between humans and the environment, potentially compensating for motor function loss. This thesis focuses on ECoG-based BCI systems that showed a high potential to provide efficient communication while being less invasive than intracortical recordings. In particular, we explored problem of continuous 3D hand translation decoding in a tetraplegic patient. In this case, most studies use linear models that may be too simple to analyze brain processes and may suffer from low decoding accuracy. Models based on deep learning (DL) have been proven effective in various tasks and thus emerge as a potential solution to create a robust brain signals representation. In this thesis, we studied the potential of DL-based methods for hand translation decoding from ECoG signals.

First, we evaluated several DL models to predict 3D hand translation from ECoG time-frequency features. The analysis was performed on a dataset recorded with a tetraplegic subject in the BCI and Tetraplegia clinical trial (NCT02550522). We started the investigation with a multilayer perceptron taking vectorized features as input. Then, we proposed convolutional neural networks (CNN), which take matrix-organized inputs approximating the spatial arrangement of the electrodes. In addition, we investigated the usefulness of long short-term memory (LSTM) to analyze temporal information. Results showed that CNN-based architectures performed better than the current state-of-the-art multilinear model on the analyzed ECoG dataset. The best architecture used a CNN-based model to analyze the spatial representation of time-frequency features followed by LSTM exploiting the sequential character of the desired hand trajectory. Compared to the multilinear model, DL-based solutions increased average cosine similarity by up to 60%.

In the case of BCI, access to large datasets is limited because recordings are time-consuming and tiring. To investigate the influence of the dataset size on the decoding performance, we compared the learning curve characteristics of DL and multilinear models evaluated in the previous step. The training dataset size was gradually increased from 5 to 140 minutes of signal in different computational experiments, focusing on dataset size requirements and patient adaptation effects. Our results revealed that DL models have a learning curve profile similar to the multilinear model, increasing performance for almost all training dataset sizes. This result validates the DL-based models as a good candidate for real-life applications. We also observed increased data quality for recordings performed later, indicating improved patient ability to generate meaningful patterns.

DL proved its usefulness for computer vision, primarily in the case of end-to-end learning. It enabled extracting more powerful representations trained for a specific task and removed the step of hand-crafted feature extraction. We evaluated methods using raw ECoG signals as a natural extension of hand-crafted feature analysis. In the data processing pipeline evaluated so far, continuous wavelet transform was used to extract

time-frequency representation, which can be seen as a convolution between a set of wavelet filters and the ECoG signal. In this setup, the gradient w.r.t. filters coefficients can be computed, and thus the whole network can be trained within an end-to-end scheme. The parameters of wavelet filters were optimized end-to-end to see potential profit from adjusting the parameters to this specific problem. The results showed only minor or no benefit from training the wavelets in terms of cosine similarity, while end-to-end models require more computational power. This may suggest that training first layer parameters may be less beneficial and more challenging in the case of ECoG-based BCI.

Keywords: brain-computer interface, BCI, ECoG, tetraplegia, deep learning, convolutional neural networks, multilayer perceptron, LSTM, dataset size, learning curve, end-to-end, hand-crafted features, time-frequency representation, brain signals

Contents

Acknowledgments

I would like to thank all the people involved in this project, particularly Tetiana Aksenova, Antoine Souloumiac, and Pierre Blanchart, for fruitful discussions and indicating potential research directions in this quickly changing environment and circumstances. Many thanks go to Matthieu Martin for being an understanding contributor, always trying to decrease the chaos in our work, making the analysis less messy. Additionally, I want to thank Vincent Rouanne, Felix Martel, and the Clinatec team for keeping me company and having long discussions about machine learning but not only which were fun and constructive.

I would also thank my parents, my sister and her family, and my grandparents for supporting me in solving every problem they could help, motivating me in the phone calls, and making coming home unforgettable every time. Finally and most importantly, this PhD would not be possible without my fiancée Paula, her daily involvement, her willingness to move to France during this hard time, her constant support and motivation, and the ability to patiently listen to my thoughts and cheer me up. Undoubtedly, I would not have been able to finish this thesis without your everyday presence, considering how much time we have spent together being trapped at home by the pandemic.

This PhD project was funded by the CEA NUMERICS program, which has received funding from European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant agreement No 800945 — NUMERICS — H2020-MSCA-COFUND-2017.

Publications

Articles used in the manuscript

Śliwowski, M., Martin, M., Souloumiac, A., Blanchart, P., and Aksenova, T., (2022). Decoding ECoG signal into 3D hand translation using deep learning. Journal of Neural Engineering, 19.
URL https://doi.org/10.1088/1741-2552/ac5d69

Śliwowski, M., Martin, M., Souloumiac, A., Blanchart, P., and Aksenova, T., (2022). Impact of dataset size and long-term ECoG-based BCI usage on deep learning decoders performance. Frontiers in Human Neuroscience. **(accepted)**
URL https://arxiv.org/abs/2209.03789

Śliwowski, M., Martin, M., Souloumiac, A., Blanchart, P., and Aksenova, T., (2022). Deep learning for ECoG brain-computer interface: end-to-end vs. hand-crafted features. AIxIA 2022 – Advances in Artificial Intelligence. Lecture Notes in Computer Science. **(accepted)**
URL https://arxiv.org/abs/2210.02544

Other articles

Wei, X., Faisal, A.A., Grosse-Wentrup, M., Gramfort, A., Chevallier, S., Jayaram, V., Jeunet, C., Bakas, S., Ludwig, S., Barmpas, K., Bahri, M., Panagakis, Y., Laskaris, N., Adamos, D.A., Zafeiriou, S., Duong, W.C., Gordon, S.M., Lawhern, V.J., Śliwowski, M., Rouanne, V. & Tempczyk, P.. (2022). 2021 BEETL Competition: Advancing Transfer Learning for Subject Independence & Heterogenous EEG Data Sets. Proceedings of the NeurIPS 2021 Competitions and Demonstrations Track, in Proceedings of Machine Learning Research 176:205-219
URL https://proceedings.mlr.press/v176/wei22a.html.

Rouanne V., Śliwowski M., Costecalde T., Benabid A. and Aksenova T. (2021). Detection of Error Correlates in the Motor Cortex in a Long Term Clinical Trial of ECoG based Brain Computer Interface.In Proceedings of the 14th International Joint Conference on Biomedical Engineering Systems and Technologies - Volume 2: BIOSIGNALS, pages 26-34.
URL https://www.scitepress.org/Link.aspx?doi=10.5220/0010227800260034

Oral communications

Śliwowski, M., Martin, M., Aksenova, T. (2022). End-to-end vs. hand-crafted features: how much ECoG data do we need to train deep learning decoders? Talk at CORTICO 2022: Invasive and non invasive Brain-Computer Interfaces - A handshake over the cliff.

Śliwowski, M., Rouanne, V., & Tempczyk, P. (2021) EEG transfer learning with SPDNet - BEETL NeurIPS 2021 competition solution. Talk and poster at NeurIPS 2021 Competition Workshop.

Śliwowski, M., Matrin, M., Souloumiac, A., Blanchart, P., & Aksenova, T. (2021). Deep learning for decoding 3D hand translation based on ECoG signal. Talk at ML in PL Conference 2021.

Achievements

3rd place award in BEETL: Benchmark for EEG Transfer Learning NeurIPS 2021 Competition as `ms_01` team.

The Contributed Talk Award Runner-up at ML in PL Conference 2021.

0

Thesis structure

The main part of this thesis is composed of three scientific articles created during the 3-years-long PhD project. Each article presents a piece of methodological development together with computed results on the topic of artificial intelligence used to decode motor commands from electrocorticography signals for brain-computer interfaces. All three major contributions were prepared gradually, step-by-step, aiming to understand and improve the current state-of-the-art system as well as previously proposed methods. Before presenting the main contributions, we start with a general introduction to the topic of brain-computer interfacing and provide detailed context regarding the clinical trial. Next, we present artificial intelligence and deep learning methods that, together with brain-computer interfaces, constitute the thesis's core. As a final part of the introduction, we precise our research goals and describe the process of research project development, creating a link between each scientific contribution and explaining their importance in the context of current deep learning and brain-computer interfaces research. Finally, after presenting the main contributions, a conclusion chapter provides a global view of the presented research, describes the thesis's known limitations, and presents potential further developments.

1 Introduction

1.1 Brain-computer interfaces

Brain-computer interface (BCI) is a device that enables a direct connection between brain and computer without requiring any muscle activation. In recent years, BCIs have become more popular in the mainstream thanks to ambitious projects on the border of science and fiction. In real-life applications, BCIs are still far away from the vision of capable and widespread mind-reading technology presented by startups and entrepreneurs. However, significant progress in BCI-related research was made with several aspiring, cutting-the-edge projects. While BCI research is driven by various motivations, e.g., entertainment [Cattan, 2021], enhancing human capabilities [Brunner et al., 2015], it is medical applications that lead the progress in the field. In particular, BCI can be used in several medical applications, for example, consciousness assessment [Dovgialo et al., 2019], rehabilitation [Mane et al., 2020; Sebastián-Romagosa et al., 2020], motor deficits compensation [Ajiboye et al., 2017; Benabid et al., 2019; Hochberg et al., 2012]. In this thesis, we focus on motor BCI used to restore/compensate lost functionalities by providing control over various effectors, e.g., computer cursor [Wolpaw and McFarland, 2004], robotic arm [Hochberg et al., 2012], exoskeleton [Benabid et al., 2019], or electrical stimulation [Ajiboye et al., 2017; Lorach et al., 2022]. In the case of motor BCI, many patients, for example, with locked-in syndrome or after spinal cord injury, preserve a fully-functional cortex so they can

generate distinct motor imagery patterns. However, the connection between the brain and the muscles is interrupted due to injury or disease, so the neuronal signals are not properly transmitted. Another group of patients consists of those whose cortex is damaged, for example, after a stroke. In this case, brain activity is affected due to the loss of neurons. In both cases, as a result, patients lose motor functions. Trying to respond to the difficulties of motor disabled individuals, motor BCI research develops quickly because of the idea that motor BCI could treat paralysis by enabling voluntary control of prosthetic limbs [Volkova et al., 2019]. Additionally, a large number of individuals are impacted by motor disabilities. Just in the United States, 12,000 people sustain motor deficits each year due to spinal cord injury only [Hachem et al., 2017]. This creates huge demand all over the world for assistive technologies. At the same time, many best-performing BCI systems are used only in the laboratory settings [Shih et al., 2012] due to several factors, e.g., the complexity of the system, relatively low performance and usefulness, high cost, or in many cases non-negligible health risk.

1.1.1 System outline

Typically, a motor BCI system is built up of multiple pieces that communicate with each other and constitute a data processing pipeline. The zeroth and most crucial step in the motor BCI system is the human, who performs a mental task that induces changes in brain activity. Next, these changes caused by neuronal activity modulation in the cortex are captured with a recording device. Signals are then transmitted to the signal processing block that translates the recorded signal into commands understandable by effectors. Finally, decoded commands representing the participant's intent are executed by an effector. Performed actions are observed by the subject, which provides visual feedback to the participant. In some systems, feedback can also be presented in different forms, for example, proprioceptive feedback [Flesher et al., 2016]. In the following subsections, BCI components are described in detail.

1.1.2 Recording devices

BCIs rely on recording devices to obtain information about a user's intention through brain activity changes. It can be recorded with two main approaches [Nicolas-Alonso and Gomez-Gil, 2012], i.e., indirect, measuring blood hemody-

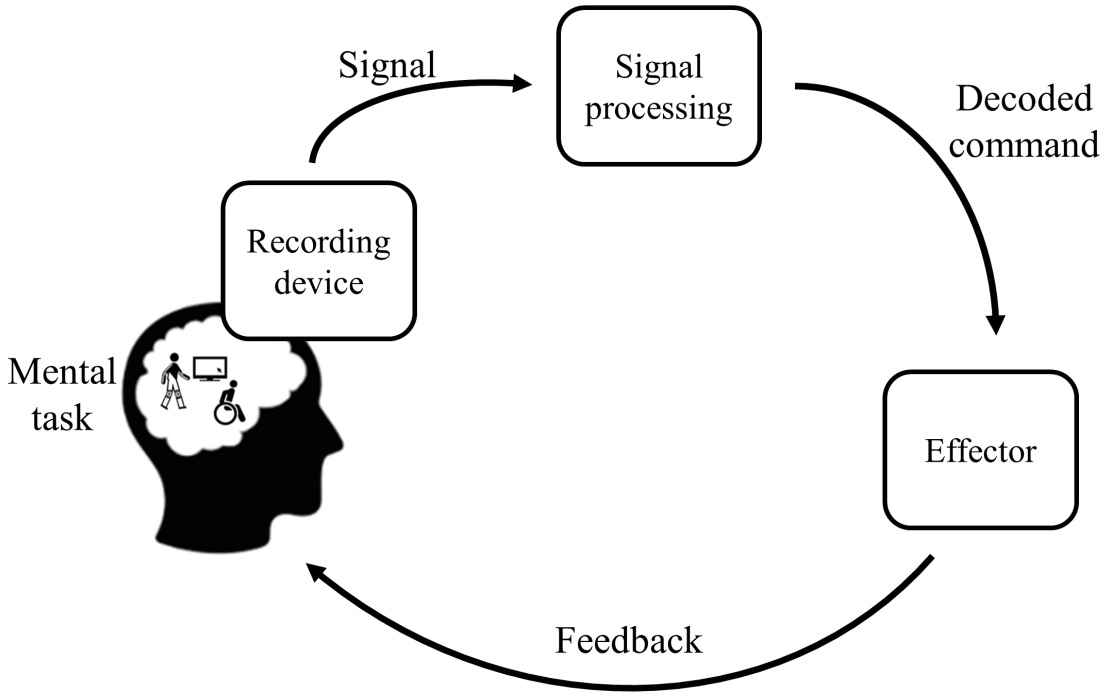


Figure 1.1: Visualization of typical BCI system.

namic response to neuronal activity with functional near-infrared spectroscopy (fNIRS) or functional magnetic resonance imaging (fMRI) and direct, recording electrical/magnetic traces of neuronal activity with electroencephalography (EEG)—electrical potential from the skull surface, magnetoencephalography (MEG)—magnetic fields in the brain tissue, electrocorticography (ECoG)—electrical potential from the brain surface, or intracortical micro-electrode arrays (MEAs)—electrical potential recorded with electrodes penetrating cortex. In this thesis, we focus on BCIs utilizing direct recording devices that are more often used and have higher temporal resolution [Nicolas-Alonso and Gomez-Gil, 2012]. Recording devices can also be divided into invasive and non-invasive. Invasive methods require implantation, which carries the health risk and costs associated with brain surgery. On the other hand, non-invasive methods are cheaper, safer, and easier to use while usually providing less information about brain activity changes [Leuthardt et al., 2021].

EEG

EEG is a non-invasive method of recording electrical brain activity, i.e., voltage fluctuations, from electrodes placed over the scalp [Buzsáki et al., 2012]. This is by far the most popular recording modality, thanks to its low cost, portability, and

ease of use. However, the skull and skin separating electrodes from the cortex increase the distance signals travel. This poses as a low-pass filter and decreases the spatial resolution and overall recording quality [Burle et al., 2015]. The signal is also more contaminated with artifacts not originating from brain activity, e.g., blinks and eye movements [Ball et al., 2009], compared to invasive recording methods. Due to the distance between electrode and signal sources, EEG sensors, similarly to ECoG and MEG, can measure the activity of synchronized neuron populations without access to single unit activation. Usually, EEG systems record signals from multiple electrodes.

MEG

Magnetoencephalography records magnetic fields induced by electrical currents flowing in neurons in the brain. MEG provides higher spatial resolution than EEG as the recorded magnetic field is less influenced by the variability of tissue and bone impedance [Hämäläinen et al., 1993]. However, MEG devices are impractical for BCI due to the high cost of the recording device and the size of the system requiring a user to stay still during the recording.

ECoG

Electrocorticography is an invasive (intracranial) recording method measuring electrical potential on the brain surface, which does not require cortex penetration. During surgery, a piece of the skull (craniotomy) is removed through which implants/grids of electrodes are placed on the brain surface, covering only a small part of the cortex. Compared to EEG, the ECoG signal contains higher frequencies because the low-pass filter of the skull and skin is removed. Recorded signal also has a higher spatial resolution, signal-to-noise ratio, and overall better data quality [Schalk and Leuthardt, 2011]. However, EEG and ECoG can only record activity of neurons' population without access to single neurons. Compared to MEAs, ECoG is less invasive, with a lower risk of medical complications [Volkova et al., 2019].

ECoG recordings for motor BCI were typically performed as a secondary goal during medical assessment aiming primarily to, for example, localize epileptic foci. In this case, devices were implanted for a few days to minimize patient risk and removed when the primary goal was achieved. Recently, ECoG implants have become viable for chronic recording with the long-term stability of the

signal [Larzabal et al., 2021], which is especially important in the case of BCI in clinical application. In the case of ECoG recordings performed primarily for motor BCI, signals can be recorded from localization optimized for motor imagery patterns [Benabid et al., 2019; Vansteensel et al., 2016], in contrast to epileptic foci detection, where ECoG grids position was selected to identify epileptic foci. ECoG arises as a promising recording modality, being a compromise between invasiveness and signal quality [Schalk and Leuthardt, 2011].

Intracortical

The most invasive method of recording brain signals for BCI involves arrays of electrodes (MEAs) that penetrate the cortex. This is a highly invasive procedure requiring craniotomy and placing electrodes inside the cortex. Cortex penetration may influence the signal stability in time as it may cause local neurodegeneration [McConnell et al., 2009] and astroglial scar formation covering electrodes [Gunasekera et al., 2015] which significantly decreases signal quality. MEAs can record single-unit spiking activity, providing a much higher information level than other recording methods. However, intracortical recordings are less stable with day-to-day variability due to physiological mechanisms in the cortex [Perge et al., 2013]. Thus, BCI systems based on MEAs can provide high performance but may require more often recalibration [Jarosiewicz et al., 2013]. To this day, intracortical-based BCIs are most powerful systems with control of neuroprosthetic arm with up to ten degrees of freedom [Collinger et al., 2013; Wodlinger et al., 2014] or high performance brain-to-text communication [Willett et al., 2021]. Unfortunately, those systems are still far from getting out of the lab due to high invasiveness, mandatory wired connection decreasing portability and the need for often system recalibration.

1.1.3 Signal processing

While recording devices provide high-quality signals, data cannot be directly deciphered into understandable commands. Therefore, typical BCI systems require a signal processing step that creates a mapping between brain signals and instructions for effectors. Due to the complexity of the problem, machine learning (ML) pipelines are usually utilized. ML describes a family of data-driven approaches in which algorithm behavior is learned from the data enabling the extraction of desired information.

Pre-processing may be applied as a first step in the data processing pipeline, aiming to improve data quality and adapt data to match the decoder's requirements. Data transformations commonly include artifact cleaning, data standardization, signal resampling and referencing, and temporal filtering. [Bashashati et al., 2007]

Secondly, an optional step of feature extraction may be performed. It aims to transform data into a well-suited representation for the decoder and contain pertinent information for decoding the user's intention. The choice of the representation has an enormous effect on the performance of ML models [Goodfellow et al., 2016]. Thus, many BCI studies used feature extraction methods to transform the temporal signal into time-frequency representation [Bashashati et al., 2007] by decomposing the signal into modulation of activity in multiple frequency bands, e.g., wavelet transform, filter-bank methods, bandpass filtering. Designing feature extraction pipelines is typically time-consuming and requires data-specific knowledge, usually with a neuroscientific background. To avoid that and benefit from end-to-end optimization, end-to-end models that use deep learning (DL) were proposed. In this case, there is no feature extraction step, and models analyze raw brain signals data (usually only pre-processed) [Schirrmeister et al., 2017].

After feature extraction, the data is further analyzed with ML models that aim to learn a mapping between brain signals and the user's intentions to decode BCI commands. To operate properly, ML models (also called decoders) require datasets to estimate parameters defining model behavior. In the case of BCI, supervised learning is the most popular. Thus, besides recorded brain signals, ML models also require labels indicating the patient's intention. To collect both brain signals and ground-truth labels, calibration sessions are required in which the user is asked to perform tasks defined by the experimental protocol. This means that the participant cannot use the BCI system for any practical task but has to follow experimental instructions. Furthermore, during calibration, experimenters record multiple realizations of a user performing task to collect a dataset containing numerous pairs of brain signals and ground-truth label $(\mathbf{X}_i, \mathbf{y}_i)$. This is usually tiring and time-consuming, reducing BCI's real-life operation capabilities.

Formally, a decoder represents a function $f(\mathbf{X}_i; \theta) \rightarrow \hat{\mathbf{y}}_i$, where f is a function that maps i-th observation $\mathbf{X}_i$ into user's intention prediction $\hat{\mathbf{y}}_i$ that in the case

of accurate decoding should be equal to ground-truth label $\mathbf{y}_i$. A model function f typically depends on a set of parameters θ estimated using the dataset to maximize decoding performance.

1.1.4 Effector

BCI systems can have various effectors that execute commands decoded by the data processing pipeline. The most basic effectors in the case of motor BCI can be as simple as a computer cursor on a monitor screen. However, more advanced tools were used for motor BCI that are closer to functions that are most needed in daily life of disabled people, e.g. robotic arm [Collinger et al., 2013; Wodlinger et al., 2014] or exoskeleton [Benabid et al., 2019]. What is more, recently, implants stimulating the spinal cord in paraplegic patients were used to give users back the ability to move their legs and even walk outside the lab. This opens a bright perspective for new and more advanced BCI applications for supporting everyday mobility and providing neurorehabilitation methods to people with spinal cord injury [Rowald et al., 2022].

1.1.5 Human in the loop and feedback

The presence of a human in the BCI loop is an essential factor influencing almost all aspects of the system. The user and the BCI work together [Shih et al., 2012]. Achieving BCI control is impossible without the ability to produce significant changes in brain activity for different motor commands. Motor imagery patterns can be observed in the form of frequency band changes originating from event-related desynchronization (ERD) and event-related synchronization (ERS) [Yi et al., 2014]. Several factors originating from the presence of a human in the loop affect the quality of MI patterns and brain signals datasets, which is essential for a BCI to operate properly.

- **Brain signals datasets are typically small in the number of observations.** To record data, a human subject has to participate in experiments. Due to the nature of the exercise, i.e., tiring and repetitive experiments requiring focus and strong attention through numerous realizations of the same task, it is mentally exhaustive for the subject, and it limits the amount of available data for calibrating the BCI systems.
- Recording sessions are usually long and exhaustive, causing variation in the attention and engagement level of the participant. In the case of disabled subjects, ground-truth labels for decoder training are estimated

based on the instruction for the patient under the assumption that the user is thoroughly executing the task (imaging movements without any actual motion). Thus, **mislabelled observations** appear in the dataset, decreasing data quality.

- It is not only the signal processing pipeline that learns how to decode brain signals into commands but also the BCI user that can adjust motor imagery patterns to operate the BCI more efficiently in the case of closed-loop experiments [Jarosiewicz et al., 2013]. In closed-loop experiments, a patient has actual control over effectors, typically with visual feedback (systems with proprioceptive feedback were also proposed [Flesher et al., 2016]), in contrast to open-loop experiments, in which the patient cannot see the results of the imagined commands. Altogether, **patient-model co-adaptation** makes the problem more complicated due to the learning of two actors, namely the BCI data processing system and the subject.

- Additional challenge in the case of BCI datasets is the **non-stationarity of the signal** [Gramfort et al., 2013; Perge et al., 2013]. Due to physiological changes in the brain as well as day-to-day variability of brain activity, the recorded brain signal is unstable with varying statistics across time. Thus, models may not generalize well to data recorded later, resulting in decreased decoding performance with time. Decoders may require recalibration, which limits real-life application suitability.

- In the case of invasive BCI, the **high cost and health risk** associated with the surgery reduces the number of studies conducted worldwide, with only a small number of patients willing to test those systems. This creates additional legal/ethical concerns regarding experimental design and capabilities to perform experiments evaluating new methods within close to real-life scenarios. Finally, it is one of the limiting factors of the current BCI research, including the topic analyzed in this thesis.

All these factors make brain signals datasets challenging to analyze and unique compared to popular ML datasets, for example, in computer vision. The datasets are small and noisy with distribution shifts, which may significantly deteriorate decoding performance and require methods adapted to brain signal characteristics.

1.2 Clinical trial

This thesis was performed in the scope of “BCI and tetraplegia" (ClinicalTrials.gov identifier: NCT02550522) clinical trial conducted at Clinatec, which was approved by the Agency for the Safety of Medicines and Health Products (Agence nationale de sécurité du médicament et des produits de santé—ANSM) with the registration number: 2015-A00650-49 and the ethical Committee for the

Protection of Individuals (Comité de Protection des Personnes—CPP) with the registration number: 15-CHUG-19.

The main goal of the clinical trial is a proof of concept enabling tetraplegic subjects to interact with their environment by controlling complex BCI effectors such as a 4-limb exoskeleton [Eliseyev et al., 2014]. This is done by decoding tetraplegic subjects' intentions from the ECoG signal recorded from the brain surface. The proposed BCI system is designed for long-term application to provide control of multiple effectors to disabled subjects by using brain activity modulation. In the project, substantial developments of all the BCI components were desired, i.e., creating ECoG implants suitable for chronic recordings [Mestais et al., 2015], building a signal processing platform for real-time signals decoding [Eliseyev et al., 2017b], and finally designing a set of effectors including both robotic exoskeleton [Morinière et al., 2015] and purely computer command executors. As a result, the project may be a step toward improving the quality of tetraplegic subjects' life by providing them with tools that can compensate for part of the lost motor function [Benabid et al., 2019].

1.2.1 Recording device

In the Clinatec BCI project, a WIMAGINE (wireless implantable multichannel acquisition system for generic interface with neurons) [Mestais et al., 2015] ECoG implant was created (see figure 1.2). The aim was to design a device that can be implanted in humans for various clinical applications requiring long-term chronic recordings. WIMAGINE implant is a wireless device (both signal transmission and power supply) placed inside a 50 mm craniotomy. It records epidural ECoG signal, which can be a reasonable trade-off between health risk associated with the surgery and quality of recorded signal [Martens et al., 2014]. The implant can record signals from 64 biocompatible electrodes arranged on a grid and additional three reference electrodes. The signal is bandpass filtered on-chip between 0.5 Hz and 300 Hz. Power is supplied to the implant inductively from a helmet that is also used to receive the data. The helmet is connected to the base station and finally to a computer. More details about the recording system can be found in [Mestais et al., 2015].

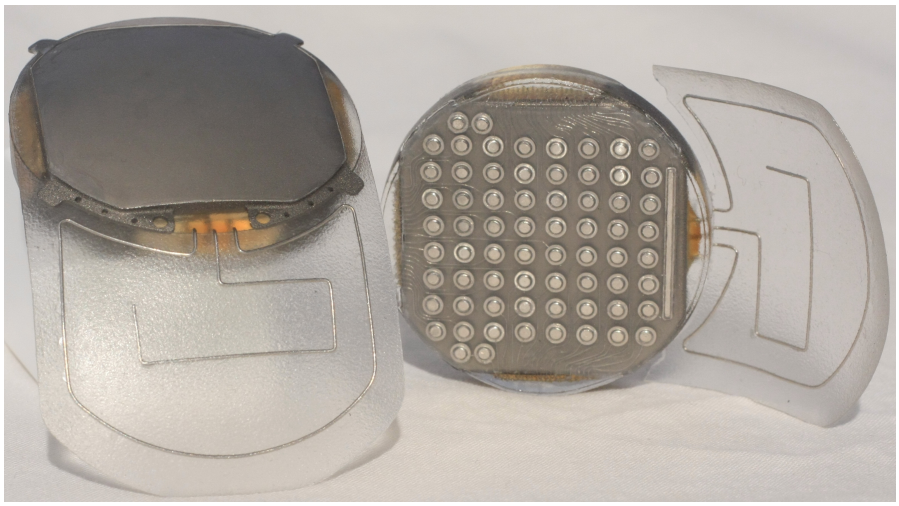


Figure 1.2: WIMAGINE implants.

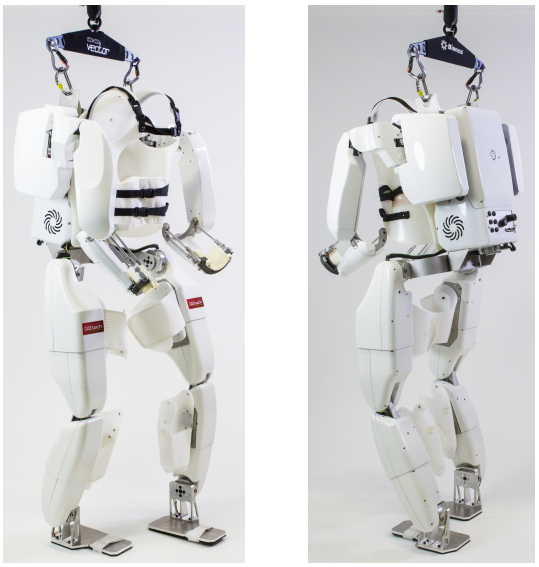


Figure 1.3: EMY exoskeleton designed to be used as a BCI effector for tetraplegic patients.

1.2.2 Effectors

Various experimental paradigms and effectors were designed for the clinical trial. The most advanced assistive technology designed in the BCI project is an EMY (Enhancing MobilitY) 4-limb robotic exoskeleton [Morinière et al., 2015] (see figure 1.3 and 1.4.c) in which patient can stand, walk, grasp, translate both hands and rotate wrists. This provides the ability to perform a fair number of real-life tasks, such as reaching intended localization, moving objects, or even pouring water.

Apart from the exoskeleton, more effectors were created. For example, a virtual avatar environment which reflects real exoskeleton movements (figure 1.4.d).

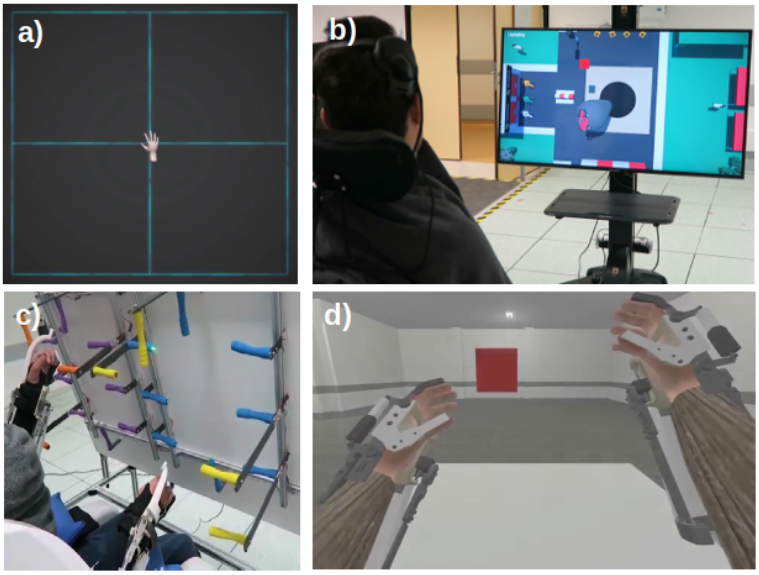


Figure 1.4: Example of BCI games/simulators implemented in the clinical trial virtual environment, i.e., a)—2D cursor movement, b)—2D joystick control to drive formula one car, d)—3D virtual avatar hand movement environment, and an image of the patient performing hand translation and rotation experiment with a real exoskeleton (c).

This virtual environment is useful in training patients to operate such a complex effector in a safer and more comfortable position. This is needed in the case of long recording sessions. Virtual effectors provide an opportunity to perform experiments outside the lab, remotely, at the patient's home. This is especially important considering the current worldwide situation and the patient's mobility limitations. Simulators can also be used at the beginning of patient training to let the user learn how to use the system in more standard conditions and gradually increase the difficulty. As the exoskeleton's limbs move together with the patient's limbs, the perception of the movement can be modified, influencing the brain activity, so the transfer between virtual and real environments may not be straightforward. Figure 1.4 shows an example of virtual environment games.

The exoskeleton consists of multiple effectors controlled with two main types of commands, i.e., discrete and continuous. In the case of discrete commands, patient intention can be expressed as a binary instruction for an effector, e.g., in the case of walking, the command represents whether the exoskeleton should move or not; for grasping, whether to open/close the exoskeleton hand. Discrete tasks are usually easier to use as they require less information and precision to be decoded from the signals and are the most popular type of task for non-invasive recordings. Continuous control effectors require a command describing the de-

sired movement parameters, e.g., position, direction, and velocity. Examples of effectors that use continuous commands can be hand translation—requires a 3D vector defining the desired direction of movement or wrist rotation—requires the angle to which the wrist should be rotated. Continuous decoding is more challenging due to the higher level of information needed to perform correct movements.

1.2.3 Signal processing

The overall signal processing pipeline used in the clinical trial is presented in figure 1.5. The first step in the current signal processing pipeline is feature extraction. The recorded ECoG signal is transformed into a time-frequency representation. Every 100 ms, one-second-long windows of the signal are analyzed using complex continuous wavelet transform (CWT). 15 Morlet wavelets with central frequencies between 10 and 150 Hz (interval 10 Hz) were used to decompose the signal into time-frequency representation. The modulus of the convolved complex signal was averaged over 0.1 s fragments. Every observation of the ECoG signal of shape 64×590 was transformed into a feature tensor of shape $64 \times 15 \times 10$ with axes corresponding to electrodes, frequency bands, and time steps. Obtained features were passed to the decoder. [Eliseyev et al., 2017a]

A signal processing system was designed to decode ECoG signals into actions of multiple effectors using a mixture of experts. In the recursive exponentially weighted Markov-switching linear model (REW-MSLM) [Moly et al., 2022], the user can control several states, e.g., left or right hand translation, left or right wrist rotation. The BCI works asynchronously, so the system can also be in an idle state, meaning no effectors are activated. For every state, a multilinear regression model (expert) predicts the desired hand, or more generally effector, movement. Expert outputs are then weighted using the gate model (hidden Markov model) output, i.e., the probability of every expert being activated. This approach assumes that neuronal signals differ significantly for every state, so the mapping between ECoG signal and desired movement can be modeled using state-specific experts.

Described decoding model works in real-time, providing predictions to the effector every 100 ms. The system's main component is a multilinear expert optimized using recursive exponentially weighted n-way partial least squares (REW-NPLS) [Eliseyev et al., 2017a]. REW-NPLS allows for estimating parame-

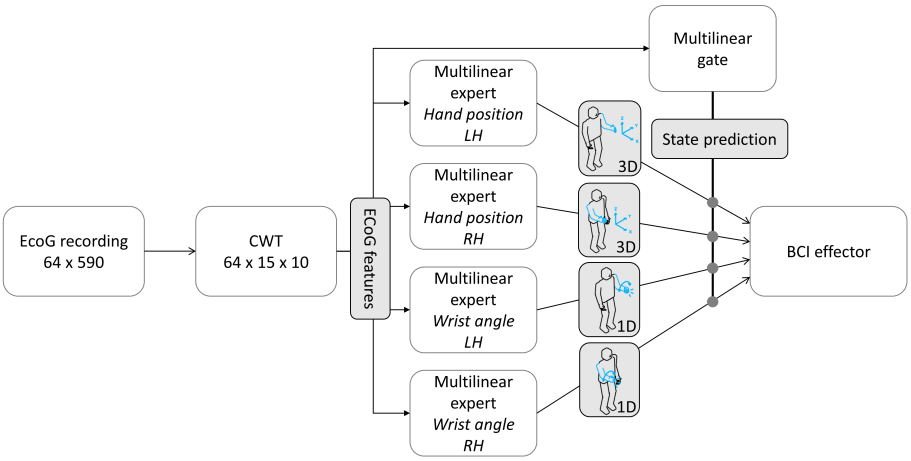


Figure 1.5: Visualization of a mixture of models system used in the clinical trial in the case of upper-limb 8D alternative control [Moly et al., 2022]. Numbers below the component name indicate the shape of a signal or feature tensor returned by the component. In italic, the state's name analyzed by an expert is described. This visualization presents the experiment in which the patient can control 3D left hand position, 3D right hand position, 1D left wrist rotation and 1D right wrist rotation. LH and RH are acronyms for left hand and right hand.

ters online, i.e., every 15 seconds expert is fitted to the new batch of data. Thanks to that, models can adjust to the new incoming data in real-time and potentially improve decoder performance [Moly et al., 2022].

1.2.4 Thesis goals in the clinical trial context

A modern BCI decoding system should fulfill several requirements influencing control efficiency and communication performance. Decoders operate in a complex environment consisting of several components and interact with a tetraplegic patient. Skomrock et al. [2018] identified the three most desired features of a BCI that may facilitate the adoption of the system in everyday clinical use. We are going to present them briefly here with additional points that we consider important from the clinical trial and ML models development perspective:

- A decoder has to predict motor commands with **high accuracy**, so the system is able to follow user's intentions. In the case of low performance, the BCI system is hard to use as it does not execute properly tetraplegic patient commands and requires more time to accomplish tasks Marathe and Taylor [2015]. For example, upper-limb movements usually require high precision, especially in the case of object manipulation. Thus, BCI

decoding performance is crucial to achieving usability in everyday life applications. Finally, persistent low decoding performance may also decrease the patient's motivation and attention.

- As BCI interacts with a user in real-time, decoding has to be done with **low latency**. Marathe and Taylor [2015] showed in a BCI simulator that increasing processing delay between brain signals and effector actions strongly affects the overall system performance. Moreover, increased latency decreases the sense of agency [Evans et al., 2015] which is sensitive to neuro-visual delay.

- BCI system should be **multi-functional**, i.e., enable performing a wide variety of tasks. This is usually related to the number of effectors and dimensions the user can control. For example, most daily life actions require bimanual movements and grasping that allow for object manipulation. Typically, extending the capabilities of a BCI system to new functions can decrease decoding accuracy and slow down the data processing pipeline [Skomrock et al., 2018].

- Ideal decoder should **require a small amount of data** to achieve high performance. Different ML algorithms may vary due to the necessary training dataset size [Perlich et al., 2003]. In BCI, access to big datasets is limited as recordings are performed in a specific experimental setup with a human in the loop. Models requiring less data could reduce the time needed for model calibration and make a BCI system more convenient.

Multilinear models used in the Clinatec signal processing pipeline are simple yet capable models designed to match BCI decoder requirements. However, their predictive power is limited due to their ability to express only linear relationships. In the case of BCI systems, decoding performance is an essential factor influencing overall BCI capabilities, especially when considering upper-limb movements requiring high-performance decoding. Deep learning is a family of machine learning models that model non-linear relationships in the data. This allows for recognizing more complex patterns, which may not be possible in the case of linear methods. In the last decade, the number of applications in which deep learning showed effectiveness increased drastically. The first objective of the thesis was to seek performance improvement by investigating whether DL-based models are suitable for 3D hand translation decoding in the ECoG BCI context. We focused on upper-limb translation, which is highly important for increasing tetraplegic patients' autonomy and requires particularly accurate decoding. Additionally, the increase in performance was the most needed for 3D hand translation, considering multilinear model accuracy. Thanks to the modular structure of the signal processing pipeline consisting of a mixture of experts, it

is possible to replace some experts with another type of model so that DL-based methods can be easily integrated into the experimental setup. We focused on improving 3D hand translation decoding performance in the article presented in chapter 2, where we applied DL-based models to features extracted with CWT, and in chapter 4, where we investigated potential further improvements from using end-to-end optimization on raw ECoG signals.

The second goal of the thesis was to study the optimal training dataset size for multilinear and DL-based models. The relationship between the size of the dataset and decoding performance is especially important in the case of BCI because it defines how long the calibration experiments have to be. Therefore, reducing the training dataset size decreases the time needed to calibrate the BCI system and reduces the subject's inconvenience. Thus, models that require smaller datasets are much preferable. Additionally, our analysis focusing on signal non-stationarity showed data quality changes throughout the experiment. Finally, we investigated whether long-term recordings and patient adaptation play an important role from the perspective of decoding performance. The results of our analysis were presented in chapter 3.

1.3 Deep learning

Artificial intelligence (AI), the science and engineering of making intelligent machines [McCarthy, 2004], is a popular research topic with many studies published every year. Sometimes, AI tends to be interpreted as systems of human-like general intelligence[1], especially in business and popular science publications. Here, we focus on algorithms capable of solving complex tasks, effective only in a narrow specialization. AI describes a wide family of algorithms, while this thesis covers only a small part, namely machine learning (ML), with a particular focus on the deep learning (DL) subfield, sometimes called also artificial neural networks (ANN), due to the inspiration from the brain. ML methods are meant to learn patterns from the data without being explicitly programmed. Based on the data, models adjust their behavior to minimize a loss function, expressing desired model behavior.

DL is a group of algorithms loosely inspired by, in a great simplification, how the human brain works. However, they are not realistic models of brain bio-

[1]artificial general intelligence (AGI) is also extensively researched

logical mechanisms. Generally, deep learning represents systems consisting of multiple units called neurons. Neurons interact with each other and perform a simple operation, i.e., linear combination, with a non-linear activation function applied after. Most of the computational units in DL are based on a rectified linear unit (with activation function $g(z) = \max(0, z)$), a simplified model of a neuron [Goodfellow et al., 2016]. Computational neurons in ANNs correspond to biological neurons that are connected with other neurons with synapses. Arriving impulses (synaptic inputs) affect the state of a neuron, i.e., membrane potential, to a different extent and ways (excitatory or inhibitory connections). When a neuron has enough excitation to reach a threshold value from below, it fires an all-or-nothing spike (action potential) towards other connected neurons. [Burkitt, 2006] In this system, a neuron is a basic computational unit that processes input data and affects other units. Neurons activate according to a non-linear function, i.e., reaching a threshold from below causes an action potential. This model of neuron functioning is generally called an integrate-and-fire model.

The deep learning name originates from a deep structure created by the system components. DL models are built up from a series of non-linear transformations, called layers. Layers are stacked one after another so that the output of one layer is the input to the next one. This allows for creating a complex representation of the input data. The core idea behind deep learning is hierarchical representation learning, i.e., creating complex representations out of a set of simpler representations optimized to solve a specific problem. Every layer can build a more complex representation based on the output of the previous one, reflecting a higher level of abstraction. For example, in images, the first layers detect simple structures such as edges that, in the next layers, form motifs, parts, and finally, objects [Goodfellow et al., 2015].

In recent years, deep learning has grown rapidly with numerous highly-influential studies. Several architectures and layers were utilized to perform specific operations to extract certain information from the data. In the following subsections, we are going to present DL concepts that were used throughout the thesis. However, we limit ourselves to brief descriptions of the ideas required to understand the main contributions. For a more detailed explanation, we refer the reader to, for example, the Deep Learning book [Goodfellow et al., 2016].

1.3.1 Multilayer perceptron

Multilayer perceptron (MLP) is a basic feedforward neural network yet powerful architecture, implementing a series of fully-connected layers (FC). Every FC is composed of several neurons that, in the case of the fully connected layer, are linked to all the units in the previous and next layer. The strength of the connection is defined with connection weights matrix $\mathbf{W} \in \mathbb{R}^{l \times d}$—trainable parameters of the layer. The number of trainable parameters depends on the number of neurons in a layer l and the number of features d in the input vector $\mathbf{x} \in \mathbb{R}^d$. FC layer function f_{FC} is defined to be:

$$f_{\text{FC}}(\mathbf{x}; \mathbf{W}, \mathbf{b}) = g(\mathbf{W}\mathbf{x} + \mathbf{b}) \tag{1.1}$$

where $\mathbf{b} \in \mathbb{R}^l$ is the bias vector. The matrix W and bias vector b comprise trainable parameters θ of an FC layer. After an affine transformation, a non-linear activation function g is applied element-wise. A typical recommended activation function for most feedforward neural networks is rectified linear activation function $g(z) = \max(0, z)$ [Goodfellow et al., 2016].

The last MLP layer is called the output layer, and it provides the model prediction $\hat{\mathbf{y}}_i$ for input data $\mathbf{X}_i$ that corresponds to the ground-truth label/target $\mathbf{y}_i$. The size of the last layer reflects the number of categories for the classification or number of targets to predict in the case of regression problems. In general, every k-th layer can be expressed as a function $f^{(k)}(\mathbf{x}; \theta^{(k)})$ whose behavior for every input $\mathbf{x}$ is defined by the set of parameters $\theta^{(k)}$. For example, output of 3-layers-deep model is computed as $f(\mathbf{X}_i; \theta) = f^{(3)}(f^{(2)}(f^{(1)}(\mathbf{X}_i; \theta^{(1)}); \theta^{(2)}); \theta^{(3)})$, where θ contains all network parameters $\theta^{(k)}$ for $k \in \{1, 2, 3\}$.

1.3.2 Convolutional neural networks

A special type of feedforward neural network is a convolutional neural network (CNN). The core idea behind convolution is exploiting structures in the input data. For example, images are organized on a grid with a correlation between neighboring pixels and occurring space invariant patterns; time-series data points are related in time between consecutive samples with possibly occurring time-invariant patterns. CNNs can be efficient for brain signals analysis as they are designed to catch these patterns and make use of this kind of relationship when creating a complex representation. To do that, CNNs use convolution

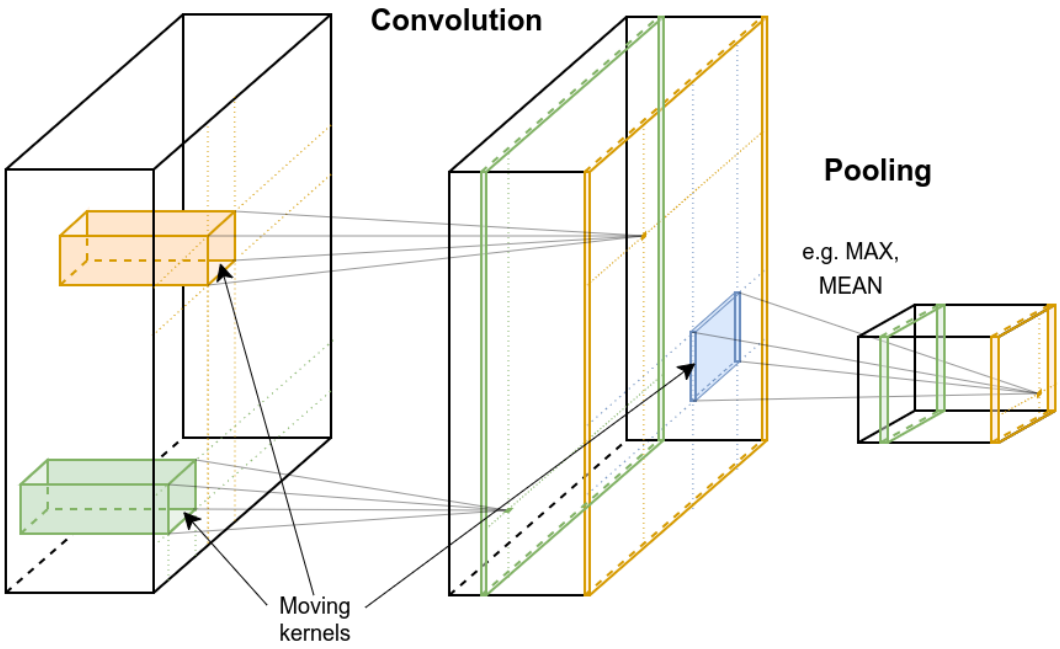


Figure 1.6: Visualization of convolution and pooling operations in a 2D case with multiple channels in the input. Convolution was visualized for two different kernels (orange and green). The blue pooling kernel performs, for example, max or mean operation.

operation instead of general matrix multiplication used in fully-connected layers [Goodfellow et al., 2016]. The convolution operation in the discrete case (which is the case for DL) and 1D can be defined as:

$$s(t) = (x * w)(t) = \sum_{a=-\infty}^{\infty} x(a)w(t-a), \tag{1.2}$$

where t is a discrete sample index, x is the input, and w is the weighting function, also called kernel or filter. In the case of multidimensional input (see figure 1.6), as images or multichannel brain signals input, it can be generalized to:

$$\mathbf{S}(i,j) = \sum_{m=1}^{M}\sum_{n=1}^{N} \mathbf{X}(i-m, j-n)\mathbf{W}(m,n), \tag{1.3}$$

where i and j are integer indices of, for example, pixels, M and N are the height and width of the kernel $\mathbf{W} \in \mathbb{R}^{M\times N}$, $\mathbf{X}$ is the input data. After convolution, an activation function is applied as in a fully-connected layer. Note that current DL frameworks implement cross-correlation instead of convolution, which does not flip the kernel. However, it is adopted in the community to use the name convolution in this context.

Compared to fully-connected layers having all neurons connected to all input features, convolutional layers are sparsely connected. Kernels are typically smaller than the input matrix/signal so that a kernel at every evaluation analyzes only a segment of the data, e.g., a group of neighboring pixels (not necessarily neighboring, for example, in the case of dilated convolution [Yu and Koltun, 2016]). It speeds up the computations and makes estimating kernel parameters more efficient. Another convolutional layer characteristic is weight sharing, meaning that the same kernel is applied many times to analyze the input. This reduces the number of model parameters but also forces equivariance to translation. This is an important property that allows for detecting the same patterns/objects occurring in different locations with the same set of filters. For time-series data, convolutional layer output shows the location of detected patterns in time. This can be particularly useful in the case of structures in the data that represent the same pattern but may appear at any moment/localization, such as motor imagery patterns in brain signals. The temporal and spatial structure of brain data makes CNNs the most common DL architecture used in BCI research [Roy et al., 2019].

In computer vision, first convolutional layers may work similarly to traditional feature extraction pipelines, for example, Gabor filters (see visualization of learned filters in [Krizhevsky et al., 2012]). Thus, they extract low-level features, detecting basic structures in images like edges. The deep structure of multiple layers allow for creating high-level features, revealing more complex and task-specific objects. Similarly to computer vision, in EEG analysis, learned filters can be strongly correlated with traditional feature extraction pipelines [Lawhern et al., 2018]. On the other hand, shallower approaches are more popular for brain signals analysis with majority of architectures having at most 10 layers. However, there is no consensus that deeper models are not suitable for EEG analysis and the optimal depth of the model depends on a problem [Roy et al., 2019].

Pooling

Usually, convolutional layers are followed by a pooling layer that computes a summary statistic of a group of points in a feature map (output for a given layer). The most popular pooling method is max pooling, which replaces the value with the maximum in its neighborhood. Alternatively, average pooling

can be applied that computes an average of the neighborhood. Pooling is a way to make the representation less sensitive to small translations. However, at the same time, it spreads the localization of a pattern, making it less precise. A detailed theoretical analysis of feature pooling was presented in [Boureau et al., 2010].

1.3.3 Recurrent neural networks

Recurrent neural networks (RNNs) are a type of deep learning approach specialized in sequential data analysis. In contrast to feedforward networks, the forward pass in RNNs contains recurrent connections, so the prediction at the current time step also depends on a history of previous inputs. Both CNNs and RNNs are based on parameters sharing paradigm. However, RNNs preserve information from previous time steps, i.e., output (or hidden state, depending on the connection type) from the previous time step $\mathbf{h}^{(t-1)}$ is also input to the next processing step together with input data $\mathbf{x}^{(t)}$ for time step t (see equation 1.4). This makes RNN a stateful model that has access to a broader context accumulated from previous time steps, in contrast to CNN, which can analyze only local neighborhood at every step. Hence, RNNs could effectively solve problems like natural language processing, where long-range context affects the perception of words and sentences. Temporal structure of brain data makes RNNs a good candidate to analyze BCI datasets and a popular method for EEG data processing [Roy et al., 2019].

$$\mathbf{h}^{(t)} = f(\mathbf{h}^{(t-1)}, \mathbf{x}^{(t)}; \theta) \tag{1.4}$$

In theory, RNNs are capable of modeling long-range dependencies in the case of sequential data. RNNs create a deep computational graph with a chain-like application of the same function (see more details about gradient propagation in Section 1.3.4). When the gradient is propagated through multiple timesteps, it tends to vanish or explode due to multiplication by the same weight several times when processing data. Moreover, long-term dependencies have decreased importance compared to short-term relationships due to the depth of the computational graph resulting in a vanishing gradient [Bengio et al., 1994]. Responding to the challenge of modeling long-range relationships, several types of RNNs were proposed, e.g., long short-term memory (LSTM) [Hochreiter and Schmidhuber, 1997], gated recurrent unit (GRU) [Chung et al., 2014, 2015].

Long short-term memory

Long short-term memory networks (LSTM) are a type of recurrent neural network (RNN) that were introduced to solve the particular problem of learning long-term dependencies in the case of sequential data [Hochreiter and Schmidhuber, 1997]. LSTM cell has a gating mechanism that controls the unit's state, i.e., whether to forget the previous state or add some information. The cell state is used to compute the current time step output and is passed to the next time step prediction. Transferring cell state through time with a gating mechanism allows for efficient information propagation through long data sequences by removing several multiplications by the same weight. For a detailed and clear explanation of LSTM mechanisms, we refer the reader to the popular blog post by Olah [2015].

LSTMs were successfully used in a variety of applications, including brain signals analysis. In many cases, LSTMs were preceded with hand-crafted features extraction [Wang et al., 2018; Zhou et al., 2018] or CNNs [Garcia-Moreno et al., 2020; Jeong et al., 2020; Xie et al., 2018; Zhang et al., 2021].

Attention

Attention is a mechanism that enables networks to focus on relevant information. Often, not all the input data has to be analyzed when solving a particular task. Similarly to human perception, the attention module highlight and concentrate on the part of input useful for prediction. Several attention mechanisms were proposed to tackle the problem of learning long-range dependencies, primarily for natural language processing (neural machine translation) for sequence to sequence models [Bahdanau et al., 2015; Britz et al., 2017]. As brain signals are sequential data, various types of attention were used in neuroscience [Cisotto et al., 2020; Lan et al., 2021; Zhang et al., 2019], however without much applications [Livezey and Glaser, 2020], especially in the case of invasive recordings.

In 2017, transformer architecture [Vaswani et al., 2017] with a multi-head self-attention module revolutionized natural language processing, providing superior performance compared to previous methods. Additionally, as it is not a recurrent model, i.e., it analyzes the whole input sequence at once, it allowed to greatly speed up computations. Transformer-based architectures were evaluated in neuroscience to a limited extent, mainly for EEG so far, with promising results [Lee and Lee, 2022; Song et al., 2021; Sun et al., 2021]

1.3.4 Training

Model parameters θ are optimized to solve a supervised learning problem (this is valid for all the DL-based models analyzed in this thesis). The function $f(\mathbf{X}_i; \theta)$ can be built up from any architecture/layer presented here. Typically, model parameters θ are estimated with the maximum likelihood estimation (MLE) method. The goal is to find a set of model parameters θ_{ML} so the likelihood of observed data $\mathbf{X}_i$ is the highest. For the supervised learning problem and independent and identically distributed random variables, MLE can be expressed as searching for θ that maximizes the conditional probability of $\hat{\mathbf{y}}_i$ given $\mathbf{X}_i$ (equation 1.5).

$$\theta_{\text{ML}} = \arg\max_{\theta} \sum_{i=1}^{n} \log P(\hat{\mathbf{y}}_i | \mathbf{X}_i; \theta) \tag{1.5}$$

Based on the MLE, various loss functions can be defined. Loss functions are used to specify model performance criteria and serve as optimization objectives for DL model training. For regression models, a popular approach is mean squared error (MSE) loss:

$$\mathcal{L}_{\text{MSE}}(\theta) = \frac{1}{n} \sum_{i=1}^{n} (\mathbf{y} - f(\mathbf{X}_i; \theta))^2 \tag{1.6}$$

The goal of optimization is to minimize the loss function. We have described forward pass in the neural network's framework used to obtain model prediction $\hat{\mathbf{y}}$, which is a series of non-linear transformations. All the layers may have trainable parameters. Usually, to update the parameters, gradient-based methods are used. Gradient of loss function w.r.t. parameters $\nabla_\theta \mathcal{L}(\theta)$ in the model with several layers can be obtained with backpropagation method [Rumelhart et al., 1986] of computing derivatives of function composition. Using a chain rule of calculus, it computes the gradient of the loss function for any parameter in the model, starting from the output layer and propagating gradients up to the input.

After obtaining gradient values for all trainable parameters in a model, weights are updated with a gradient descent procedure to find values of parameters θ that minimize the loss function $\mathcal{L}(\theta)$. This is done by performing steps, i.e., updating parameters with $-\lambda \nabla_\theta \mathcal{L}(\theta)$ following the direction of the steepest descent of the loss function. The learning rate parameter λ (a training hyperparameter) controls the step size. High values of λ may cause jumping over

minima while setting small values increases the time needed to reach minima and the probability of getting stuck in a local minimum. However, for DL models with numerous parameters, reaching the global minimum (which often leads to overfitting) is not required as typically, many local minima provide low enough generalization error [Choromańska et al., 2015]. In the case of DL optimization, a variant of gradient descent, namely minibatch stochastic gradient descent (SGD), is often used. Updates of parameters are based only on a subset of a dataset, i.e., minibatch, instead of the whole dataset as in the standard gradient descent. Using the expected gradient value estimated from a random subset speeds up the computation and has a regularizing effect [Wilson and Martinez, 2003]. Minibatch size (more often called batch size) is another training hyperparameter that affects the optimization process. Smaller batches strengthen regularization (by introducing additional noise to the optimization process), while using bigger batches improves the precision of gradient computation.

Besides SGD, several first-order and second-order gradient-based optimization methods are commonly used for DL training. The most popular method applied to brain signals is ADAM [Kingma and Ba, 2015] which adaptively tunes learning rate for every parameter based on estimates of first and second order moments of the gradient. Other optimizers with adaptive learning rates (e.g., AdaGrad [Duchi et al., 2011] and RMSProp [Tieleman et al., 2012]) are less often used in neuroscience [Roy et al., 2019].

Batch normalization

Batch normalization layer [Ioffe and Szegedy, 2015] allow for faster convergence and better generalization in neural networks [Bjorck et al., 2018]. It shifts network activation to be zero-mean by subtracting batch mean μ_b and to have unit standard deviation by dividing by batch standard deviation $\sqrt{\sigma_b}$ for every batch. Then, the value is scaled by trainable parameter γ and moved by adding another parameter β.

$$\mathrm{BN}(x_i)_{(\gamma,\beta)} = \gamma \frac{x_i - \mu_b}{\sqrt{\sigma_b + \epsilon}} + \beta, \tag{1.7}$$

The reason why batch normalization improves training was primarily attributed to reducing internal covariate shift [Ioffe and Szegedy, 2015] in deep models. However, Bjorck et al. [2018] showed that it is mainly caused by larger learning rates that increase the regularization of stochastic gradient descent and

improves generalization.

1.3.5 Regularization

Overfitting is one of the major challenges in machine learning. Ideally, one would like a model that performs well on new unseen data (test set). However, the optimization objective focuses on the training dataset performance only. This usually results in models with high generalization error, i.e., worse performance on the test set. To address this issue, regularization methods are applied with the main goal of decreasing the generalization error. This is especially important in the case of small training datasets such as brain signals datasets. Hence, we applied several regularization strategies in the presented experiments. In the following subsections, we describe these regularization methods.

In the simplest case, models are evaluated using two datasets, i.e., training and test datasets. First, the training dataset is used to estimate model parameter values that define model behavior. Then, the model is fixed and evaluated on the test dataset, i.e., the model provides predictions for the test observations that are used to compute scores expressing decoding performance. The test set must not be used for hyperparameters (e.g., learning rate, batch size, regularization strength) or architecture (e.g., number of layers, number of neurons) tuning to not cause data leakage, which affects the credibility of generalization error evaluation. To avoid overfitting to the test dataset, model hyperparameters are selected based on a validation dataset score, i.e., datasets separated from the training dataset and not used to optimize model parameters (weights).

Early stopping

It is often observed that training loss decreases, but generalization error may start to increase due to overfitting. To reduce overfitting, the validation dataset can be also used for early stopping, i.e., stopping the training when validation loss is not decreasing. This prevents overfitting by measuring generalization error on the unseen data and selecting models that achieved high performance on the validation dataset.

Dropout

One of the most popular regularization methods is dropout [Srivastava et al., 2014]. Every neuron activation is randomly dropped during training, i.e., set to

zero, with a specified dropout probability (often 0.5). This forces the network to create sparse and redundant representations by destroying co-adaptation between units. Dropout can also be interpreted as an easy and cheap, in terms of computations, way to create an ensemble of models. When using dropout, multiple subnetworks are created thanks to the random neuron zeroing. For the prediction, the activations of all subnetworks are averaged to obtain the final estimate.

L2 regularization

L2 regularization (also called weight decay) is a regularization method that, by imposing an additional penalty on the weights (model parameters), pushes their value towards zero. Effectively, it decreases the strength of interaction, making the model less complex and keeping high values of parameters only for crucial connections. L2 regularization is implemented by adding a term to the loss function penalizing model for having weights other than zero, i.e., $||\mathbf{W}||^2$.

1.4 Objectives and contributions

Applying deep learning to a new dataset usually requires several choices, e.g., architecture type and depth, loss function, regularization methods, and, more generally speaking, all different hyperparameters. This complex task has to take into account numerous factors and, in the end, is evaluated with an experimental assessment that must be adapted to the problem.

The first objective of the thesis was to investigate potential performance gains from using DL-based methods to decode hand translation signals from ECoG signals. In chapter 2, we proposed several architectures, based on MLP, CNNs, and LSTMs, for analyzing the ECoG data recorded in the clinical trial. As a first step, the problem of hand translation decoding was formulated in terms of the loss function and evaluation setup, i.e., session-wise cross-validation, hyperparameters search, and train-test split. Next, DL-based methods were compared to a multilinear model already proven effective in this task. In this study, we decided to analyze hand-crafted time-frequency features extracted using a standard pipeline designed for the clinical trial. This way, we tried to avoid changing multiple factors at once and kept similar conditions for both DL and multilinear models for a fair comparison. Furthermore, as the choice of architecture

and hyperparameters is complicated, we performed an ablation study where we removed/replaced particular mechanisms in the DL models to assess the importance of each choice. Finally, model performance was evaluated from the perspective of model usefulness in the long-term recordings with a human in the loop, i.e., evaluating decoding stability and screening for potential model biases. The best DL architecture was checked for the most important input features (spatial, temporal, frequential) contributing to the prediction to validate the results from the neuroscientific point of view. To sum up, chapter 2 presents the first step toward implementing DL-based models in the real-life clinical trial evaluation, i.e., the proposition of DL architectures together with their offline evaluation showing potential improvement and most important components. The main questions considered were:

- Is DL suitable for 3D hand translation prediction from ECoG signal? How does it compare to the baseline multilinear model?
- Which architecture can be efficient for ECoG 3D hand translation decoding?
- What makes DL model effective?
- Can DL fulfill the requirements for BCI decoders, e.g., decoding stability in time?

The second main objective of the thesis was to identify dataset size requirements of different models, in particular DL-based and linear methods. One of the reasons for DL's success is easier access to big datasets in computer vision or natural language processing that started in the 2010s. However, in brain signals analysis size of the dataset is limited by experimental constraints. As a result, decoders should require a small amount of data to decrease the time needed for system calibration. In chapter 3, we further investigated characteristics of the models evaluated in chapter 2 with particular focus on dataset size requirements. This was possible thanks to a unique dataset of long-term recordings. Several factors influence data recorded in real-life experiments with humans, so we decided to increase the training dataset size in different manners. It allowed us to isolate the influence of dataset increase, signal non-stationarity, and patient adaptation to analyze how those factors affect ECoG BCI decoders trained on this dataset. Additionally, we explored distribution changes that may indicate data quality increase, confirming patient adaptation during the experiments. The main questions considered were:

- How does adding more data to the training dataset affect 3D hand translation decoders?
- How do DL-based approaches compare to multilinear models in terms of required dataset size?
- Do we observe data quality changes across long-term recordings suggesting patient adaptation?

The next problem explored was the effectiveness of end-to-end optimization of manually-designed feature extraction step. Models evaluated in chapter 2 and chapter 3 used features extracted with continuous wavelet transform as input, so the data is in a form of time-frequency representation. However, many DL-based approaches are end-to-end models utilizing raw data as input. In end-to-end models, the whole data analysis process is optimized together. Thanks to that, the feature extraction step can be adjusted to the needs of the problem and model architecture. Thus, potentially, end-to-end optimization can be a solution providing higher performance and allowing for extracting features in a data-driven way that humans could not design. Generally, feature extraction design is time-consuming and sensitive to researcher's bias, so end-to-end optimization could be a solution to remove the process of manual tweaking. In chapter 4, DL-models analyzing raw ECoG signals were evaluated. Particularly, we trained models proposed in chapter 2 from scratch and starting from reproducing continuous wavelet transform (but also ShallowFBCSPNet and Deep4Net [Schirrmeister et al., 2017] as a baseline only from scratch). The main goal was to investigate whether end-to-end training can improve decoding performance. Further, the difference between hand-crafted features-based and end-to-end models was studied when adding more data to the training dataset to verify the hypothesis that end-to-end models may require more data to estimate robust parameters. Obtained filters were analyzed in order to identify changes in the feature extraction step that could potentially lead to modification of the data processing pipeline. The main questions considered were:

- Can end-to-end optimization improve the performance of 3D hand translation decoding from ECoG signal?
- Does end-to-end optimization require more data to achieve high performance?
- What kind of features do end-to-end models learn to detect?

All three contributions focus on applying DL models to a unique ECoG dataset recorded in the clinical trial. The main goal of the analysis from the application point of view was to provide higher decoding performance needed in the clinical trial to make the BCI system more powerful. We evaluated several approaches and tested hypotheses in order to explore the problem of 3D hand translation prediction from ECoG signals with DL and to better understand data and model characteristics.

Bibliography

A Bolu Ajiboye, Francis R Willett, Daniel R Young, William D Memberg, Brian A Murphy, Jonathan P Miller, Benjamin L Walter, Jennifer A Sweet, Harry A Hoyen, Michael W Keith, P Hunter Peckham, John D Simeral, John P Donoghue, Leigh R Hochberg, and Robert F Kirsch. Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration. *The Lancet*, 389(10081):1821–1830, May 2017. ISSN 01406736. doi: 10.1016/S0140-6736(17)30601-3. URL https://linkinghub.elsevier.com/retrieve/pii/S0140673617306013. Number: 10081.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2015.

Tonio Ball, Markus Kern, Isabella Mutschler, Ad Aertsen, and Andreas Schulze-Bonhage. Signal quality of simultaneously recorded invasive and non-invasive eeg. *NeuroImage*, 46(3):708–716, 2009. ISSN 1053-8119. doi: https://doi.org/10.1016/j.neuroimage.2009.02.028. URL https://www.sciencedirect.com/science/article/pii/S1053811909001827.

Ali Bashashati, Mehrdad Fatourechi, Rabab K Ward, and Gary E Birch. A survey of signal processing algorithms in brain–computer interfaces based on electrical brain signals. *Journal of Neural Engineering*, 4(2):R32–R57, mar 2007. doi: 10.1088/1741-2560/4/2/r03. URL https://doi.org/10.1088/1741-2560/4/2/r03.

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. *The Lancet Neurology*, 18(12):1112–1122, December 2019. ISSN 14744422. doi: 10.1016/S1474-4422(19)30321-7. URL https://linkinghub.elsevier.com/retrieve/pii/S1474442219303217. Number: 12.

Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994. doi: 10.1109/72.279181.

Nils Bjorck, Carla P Gomes, Bart Selman, and Kilian Q Weinberger. Understanding batch normalization. In S. Bengio, H. Wallach, H. Larochelle,

K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/36072923bfc3cf47745d704feb489480-Paper.pdf.

Y-Lan Boureau, Jean Ponce, and Yann LeCun. A theoretical analysis of feature pooling in visual recognition. In *ICML*, 2010.

Denny Britz, Anna Goldie, Minh-Thang Luong, and Quoc V. Le. Massive exploration of neural machine translation architectures. *ArXiv*, abs/1703.03906, 2017.

Clemens Brunner, Niels Birbaumer, Benjamin Blankertz, Christoph Guger, Andrea Kübler, Donatella Mattia, José del R. Millán, Felip Miralles, Anton Nijholt, Eloy Opisso, Nick Ramsey, Patric Salomon, and Gernot R. Müller-Putz. Bnci horizon 2020: towards a roadmap for the bci community. *Brain-Computer Interfaces*, 2(1):1–10, 2015. doi: 10.1080/2326263X.2015.1008956. URL https://doi.org/10.1080/2326263X.2015.1008956.

Anthony N. Burkitt. A review of the integrate-and-fire neuron model: I. homogeneous synaptic input. *Biological Cybernetics*, 95:1–19, 2006.

Borís Burle, Laure Spieser, Clémence Roger, Laurence Casini, Thierry Hasbroucq, and Franck Vidal. Spatial and temporal resolutions of EEG: Is it really black and white? A scalp current density view. *International Journal of Psychophysiology*, 97(3):210–220, September 2015. ISSN 0167-8760. doi: 10.1016/j.ijpsycho.2015.05.004. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4548479/.

György Buzsáki, Costas A. Anastassiou, and Christof Koch. The origin of extracellular fields and currents — EEG, ECoG, LFP and spikes. *Nature Reviews Neuroscience*, 13(6):407–420, June 2012. ISSN 1471-0048. doi: 10.1038/nrn3241. URL https://www.nature.com/articles/nrn3241. Number: 6 Publisher: Nature Publishing Group.

Grégoire Cattan. The use of brain–computer interfaces in games is not ready for the general public. *Frontiers in Computer Science*, 3, 2021. ISSN 2624-9898. doi: 10.3389/fcomp.2021.628773. URL https://www.frontiersin.org/articles/10.3389/fcomp.2021.628773.

Anna Choromańska, Mikael Henaff, Michaël Mathieu, Gérard Ben Arous, and Yann LeCun. The loss surfaces of multilayer networks. In *AISTATS*, 2015.

Junyoung Chung, Çaglar Gülçehre, Kyunghyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *ArXiv*, abs/1412.3555, 2014.

Junyoung Chung, Çaglar Gülçehre, Kyunghyun Cho, and Yoshua Bengio. Gated feedback recurrent neural networks. In *ICML*, 2015.

Giulia Cisotto, Alessio Zanga, Joanna Chlebus, Italo Zoppis, Sara Manzoni, and Urszula Markowska-Kaczmar. Comparison of attention-based deep learning models for eeg classification. *ArXiv*, abs/2012.01074, 2020.

Jennifer L. Collinger, Brian Wodlinger, John E. Downey, Wei Wang, Elizabeth C. Tyler-Kabara, Douglas J. Weber, Angus JC McMorland, Meel Velliste, Michael L. Boninger, and Andrew B. Schwartz. High-performance neuroprosthetic control by an individual with tetraplegia. *The Lancet*, 381:557–564, 2013.

Marian Dovgialo, Anna Chabuda, Anna Duszyk, Magdalena Zieleniewska, Marcin Pietrzak, Piotr Różański, and Piotr Durka. Assessment of statistically significant command-following in pediatric patients with disorders of consciousness, based on visual, auditory and tactile event-related potentials. *International Journal of Neural Systems*, 29(03):1850048, 2019. doi: 10.1142/S012906571850048X. URL https://doi.org/10.1142/S012906571850048X. PMID: 30606086.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(61):2121–2159, 2011. URL http://jmlr.org/papers/v12/duchi11a.html.

Andrey Eliseyev, Corinne Mestais, Guillaume Charvet, Fabien Sauter, Neil Abroug, Nana Arizumi, Serpil Cokgungor, Thomas Costecalde, Michael Foerster, Louis Korczowski, Boris Morinière, Jean Porcherot, Jérémy Pradal, David Ratel, Nicolas Tarrin, Napoleon Torres-Martinez, Alexandre Verney, Tetiana Aksenova, and Alim-Louis Benabid. Clinatec® bci platform based on the ecog-recording implant wimagine® and the innovative signal-processing: Preclinical results. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 1222–1225, 2014. doi: 10.1109/EMBC.2014.6943817.

Andrey Eliseyev, Vincent Auboiroux, Thomas Costecalde, Lilia Langar, Guillaume Charvet, Corinne Mestais, Tetiana Aksenova, and Alim-Louis Benabid. Recursive Exponentially Weighted N-way Partial Least Squares Regression with Recursive-Validation of Hyper-Parameters in Brain-Computer Interface Applications. *Scientific Reports*, 7(1):16281, December 2017a. ISSN 2045-2322. doi: 10.1038/s41598-017-16579-9. URL http://www.nature.com/articles/s41598-017-16579-9. Number: 1.

Andrey Eliseyev, Vincent Auboiroux, Thomas Costecalde, Lilia Langar, Guillaume Charvet, Corinne Mestais, Tetiana Aksenova, and Alim-Louis Benabid. Recursive Exponentially Weighted N-way Partial Least Squares Regression with Recursive-Validation of Hyper-Parameters in Brain-Computer Interface Applications. *Scientific Reports*, 7:16281, 2017b. doi: 10.1038/s41598-017-16579-9.

Nathan Evans, Steven Gale, Aaron Schurger, and Olaf Blanke. Visual feedback dominates the sense of agency for brain-machine actions. *PLOS ONE*, 10(6):1–17, 06 2015. doi: 10.1371/journal.pone.0130019. URL https://doi.org/10.1371/journal.pone.0130019.

Sharlene N. Flesher, Jennifer L. Collinger, Stephen T. Foldes, Jeffrey M. Weiss, John E. Downey, Elizabeth C. Tyler-Kabara, Sliman J. Bensmaia, Andrew B. Schwartz, Michael L. Boninger, and Robert A. Gaunt. Intracortical microstimulation of human somatosensory cortex. *Science Translational Medicine*, 8 (361):361ra141–361ra141, 2016. doi: 10.1126/scitranslmed.aaf8083. URL https://www.science.org/doi/abs/10.1126/scitranslmed.aaf8083.

Francisco M. Garcia-Moreno, Maria Bermudez-Edo, María José Rodríguez-Fórtiz, and José Luis Garrido. A cnn-lstm deep learning classifier for motor imagery eeg detection using a low-invasive and low-cost bci headband. In *2020 16th International Conference on Intelligent Environments (IE)*, pages 84–91, 2020. doi: 10.1109/IE49459.2020.9155016.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

Ian J. Goodfellow, Yoshua Bengio, and Aaron C. Courville. Deep learning. *Nature*, 521:436–444, 2015.

A. Gramfort, D. Strohmeier, J. Haueisen, M.S. Hämäläinen, and M. Kowalski. Time-frequency mixed-norm estimates: Sparse m/eeg imaging with non-stationary source activations. *NeuroImage*, 70:410–422, 2013. ISSN 1053-8119. doi: https://doi.org/10.1016/j.neuroimage.2012.12.051. URL https://www.sciencedirect.com/science/article/pii/S1053811912012372.

Bhagya Gunasekera, Tarun Saxena, Ravi Bellamkonda, and Lohitash Karumbaiah. Intracortical recording interfaces: Current challenges to chronic recording function. *ACS Chemical Neuroscience*, 6(1):68–83, 2015. doi: 10.1021/cn5002864. URL https://doi.org/10.1021/cn5002864. PMID: 25587704.

Laureen D. Hachem, Christopher S. Ahuja, and Michael G. Fehlings. Assessment and management of acute spinal cord injury: From point of injury to rehabilitation. *The Journal of Spinal Cord Medicine*, 40(6):665–675, November 2017. ISSN 1079-0268, 2045-7723. doi: 10.1080/10790268.2017.1329076. URL https://www.tandfonline.com/doi/full/10.1080/10790268.2017.1329076.

Matti Hämäläinen, Riitta Hari, Risto J. Ilmoniemi, Jukka Knuutila, and Olli V. Lounasmaa. Magnetoencephalography—theory, instrumentation, and applications to noninvasive studies of the working human brain. *Rev. Mod. Phys.*, 65:413–497, Apr 1993. doi: 10.1103/RevModPhys.65.413. URL https://link.aps.org/doi/10.1103/RevModPhys.65.413.

Leigh R. Hochberg, Daniel Bacher, Beata Jarosiewicz, Nicolas Y. Masse, John D. Simeral, Joern Vogel, Sami Haddadin, Jie Liu, Sydney S. Cash, Patrick van der Smagt, and John P. Donoghue. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, 485:372 – 375, 2012. doi: 10.1038/nature11076.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 11 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015. URL http://arxiv.org/abs/1502.03167.

Beata Jarosiewicz, Nicolas Y Masse, Daniel Bacher, Sydney S Cash, Emad Eskandar, Gerhard Friehs, John P Donoghue, and Leigh R Hochberg. Advantages of closed-loop calibration in intracortical brain–computer interfaces for people with tetraplegia. *Journal of Neural Engineering*, 10(4):046012, jul 2013. doi: 10.1088/1741-2560/10/4/046012. URL https://doi.org/10.1088/1741-2560/10/4/046012.

Ji-Hoon Jeong, Kyung-Hwan Shim, Dong-Joo Kim, and Seong-Whan Lee. Brain-controlled robotic arm system based on multi-directional cnn-bilstm network using eeg signals. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28(5):1226–1238, 2020. doi: 10.1109/TNSRE.2020.2981659.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

Zhen Lan, Chao Yan, Zixing Li, Dengqing Tang, and Xiaojia Xiang. Macro: Multi-attention convolutional recurrent model for subject-independent erp detection. *IEEE Signal Processing Letters*, 28:1505–1509, 2021. doi: 10.1109/LSP.2021.3095761.

Christelle Larzabal, Stéphane Bonnet, Thomas Costecalde, Vincent Auboiroux, Guillaume Charvet, Stéphan Chabardes, Tetiana Aksenova, and Fabien Sauter-Starace. Long-term stability of the chronic epidural wireless recorder WIMAGINE in tetraplegic patients. *Journal of Neural Engineering*, 18(5):056026, sep 2021. doi: 10.1088/1741-2552/ac2003. URL https://doi.org/10.1088/1741-2552/ac2003.

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. *Journal of Neural Engineering*, 15(5):056013, jul 2018. doi: 10.1088/1741-2552/aace8c. URL https://doi.org/10.1088/1741-2552/aace8c.

Young-Eun Lee and Seo-Hyun Lee. Eeg-transformer: Self-attention from transformer architecture for decoding eeg of imagined speech. In *2022 10th International Winter Conference on Brain-Computer Interface (BCI)*, pages 1–4, 2022. doi: 10.1109/BCI53720.2022.9735124.

Eric C. Leuthardt, Daniel W. Moran, and Tim R. Mullen. Defining surgical terminology and risk for brain computer interface technologies. *Frontiers in*

Neuroscience, 15, 2021. ISSN 1662-453X. doi: 10.3389/fnins.2021.599549. URL https://www.frontiersin.org/articles/10.3389/fnins.2021.599549.

Jesse A Livezey and Joshua I Glaser. Deep learning approaches for neural decoding across architectures and recording modalities. *Briefings in Bioinformatics*, 22(2):1577–1591, 12 2020. ISSN 1477-4054. doi: 10.1093/bib/bbaa355. URL https://doi.org/10.1093/bib/bbaa355.

Henri Lorach, Guillaume Charvet, Jocelyne Bloch, and Grégoire Courtine. Brain–spine interfaces to reverse paralysis. *National Science Review*, 01 2022. ISSN 2095-5138. doi: 10.1093/nsr/nwac009. URL https://doi.org/10.1093/nsr/nwac009. nwac009.

Ravikiran Mane, Tushar Chouhan, and Cuntai Guan. BCI for stroke rehabilitation: motor and beyond. *Journal of Neural Engineering*, 17(4):041001, aug 2020. doi: 10.1088/1741-2552/aba162. URL https://doi.org/10.1088/1741-2552/aba162.

A R Marathe and D M Taylor. The impact of command signal power distribution, processing delays, and speed scaling on neurally-controlled devices. *Journal of Neural Engineering*, 12(4):046031, jul 2015. doi: 10.1088/1741-2560/12/4/046031. URL https://doi.org/10.1088/1741-2560/12/4/046031.

Suzanne Martens, Michael Bensch, Sebastian Halder, Jeremy Hill, Femke Nijboer, Ander Ramos-Murguialday, Bernhard Schoelkopf, Niels Birbaumer, and Alireza Gharabaghi. Epidural electrocorticography for monitoring of arousal in locked-in state. *Frontiers in Human Neuroscience*, 8, 2014. ISSN 1662-5161. doi: 10.3389/fnhum.2014.00861. URL https://www.frontiersin.org/articles/10.3389/fnhum.2014.00861.

John McCarthy. WHAT IS ARTIFICIAL INTELLIGENCE? page 14, 2004.

George C McConnell, Howard D Rees, Allan I Levey, Claire-Anne Gutekunst, Robert E Gross, and Ravi V Bellamkonda. Implanted neural electrodes cause chronic, local inflammation that is correlated with local neurodegeneration. *Journal of Neural Engineering*, 6(5):056003, aug 2009. doi: 10.1088/1741-2560/6/5/056003. URL https://doi.org/10.1088/1741-2560/6/5/056003.

Corinne S. Mestais, Guillaume Charvet, Fabien Sauter-Starace, Michael Foerster, David Ratel, and Alim Louis Benabid. Wimagine: Wireless 64-channel ecog recording implant for long term clinical applications. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(1):10–21, 2015. doi: 10.1109/TNSRE.2014.2333541.

Alexandre Moly, Thomas Costecalde, Félix Martel, Matthieu Martin, Christelle Larzabal, Serpil Karakas, Alexandre Verney, Guillaume Charvet, Stephan Chabardes, Alim Louis Benabid, and Tetiana Aksenova. An adaptive closed-loop ECoG decoder for long-term and stable bimanual control of an exoskeleton by a tetraplegic. *Journal of Neural Engineering*, 19(2):026021, mar 2022. doi: 10.1088/1741-2552/ac59a0. URL https://doi.org/10.1088/1741-2552/ac59a0.

B. Morinière, A. Verney, N. Abroug, P. Garrec, and Y. Perrot. Emy: a dual arm exoskeleton dedicated to the evaluation of brain machine interface in clinical trials. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5333–5338, 2015. doi: 10.1109/IROS.2015.7354130.

Luis Fernando Nicolas-Alonso and Jaime Gomez-Gil. Brain computer interfaces, a review. *Sensors*, 12(2):1211–1279, 2012. ISSN 1424-8220. doi: 10.3390/s120201211. URL https://www.mdpi.com/1424-8220/12/2/1211.

Christopher Olah. Understanding lstm networks, 2015. URL http://colah.github.io/posts/2015-08-Understanding-LSTMs/.

János A Perge, Mark L Homer, Wasim Q Malik, Sydney Cash, Emad Eskandar, Gerhard Friehs, John P Donoghue, and Leigh R Hochberg. Intra-day signal instabilities affect decoding performance in an intracortical neural interface system. *Journal of Neural Engineering*, 10(3):036004, apr 2013. doi: 10.1088/1741-2560/10/3/036004. URL https://doi.org/10.1088/1741-2560/10/3/036004.

Claudia Perlich, Foster Provost, and Jeffrey S. Simonoff. Tree induction vs. logistic regression: a learning-curve analysis. *The Journal of Machine Learning Research*, 4(null):211–255, 2003. ISSN 1532-4435. doi: 10.1162/153244304322972694. URL https://doi.org/10.1162/153244304322972694.

Andreas Rowald, Salif Komi, Robin Demesmaeker, Edeny Baaklini, Sergio Daniel Hernandez-Charpak, Edoardo Paoles, Hazael Montanaro, Antonino Cassara, Fabio Becce, Bryn Lloyd, Taylor Newton, Jimmy Ravier, Nawal Kinany, Marina D'Ercole, Aurélie Paley, Nicolas Hankov, Camille Varescon, Laura McCracken, Molywan Vat, Miroslav Caban, Anne Watrin, Charlotte Jacquet, Léa Bole-Feysot, Cathal Harte, Henri Lorach, Andrea Galvez, Manon Tschopp, Natacha Herrmann, Moïra Wacker, Lionel Geernaert, Isabelle Fodor, Valentin Radevich, Katrien Van Den Keybus, Grégoire Eberle, Etienne Pralong, Maxime Roulet, Jean-Baptiste Ledoux, Eleonora Fornari, Stefano Mandija, Loan Mattera, Roberto Martuzzi, Bruno Nazarian, Stefan Benkler, Simone Callegari, Nathan Greiner, Benjamin Fuhrer, Martijn Froeling, Nik Buse, Tim Denison, Rik Buschman, Christian Wende, Damien Ganty, Jurriaan Bakker, Vincent Delattre, Hendrik Lambert, Karen Minassian, Cornelis A. T. van den Berg, Anne Kavounoudias, Silvestro Micera, Dimitri Van De Ville, Quentin Barraud, Erkan Kurt, Niels Kuster, Esra Neufeld, Marco Capogrosso, Leonie Asboth, Fabien B. Wagner, Jocelyne Bloch, and Grégoire Courtine. Activity-dependent spinal cord neuromodulation rapidly restores trunk and leg motor functions after complete paralysis. *Nature Medicine*, 28(2):260–271, February 2022. ISSN 1546-170X. doi: 10.1038/s41591-021-01663-5. URL https://www.nature.com/articles/s41591-021-01663-5. Number: 2 Publisher: Nature Publishing Group.

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

aug 2019. doi: 10.1088/1741-2552/ab260c. URL https://doi.org/10.1088/1741-2552/ab260c.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

Gerwin Schalk and Eric C. Leuthardt. Brain-computer interfaces using electrocorticographic signals. *IEEE Reviews in Biomedical Engineering*, 4:140–154, 2011. doi: 10.1109/RBME.2011.2172408.

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. *Human brain mapping*, 38(11):5391–5420, 2017.

Marc Sebastián-Romagosa, Woosang Cho, Rupert Ortner, Nensi Murovec, Tim Von Oertzen, Kyousuke Kamada, Brendan Z. Allison, and Christoph Guger. Brain computer interface treatment for motor rehabilitation of upper extremity of stroke patients—a feasibility study. *Frontiers in Neuroscience*, 14, 2020. ISSN 1662-453X. doi: 10.3389/fnins.2020.591435. URL https://www.frontiersin.org/articles/10.3389/fnins.2020.591435.

Jerry J. Shih, Dean J. Krusienski, and Jonathan R. Wolpaw. Brain-computer interfaces in medicine. *Mayo Clinic Proceedings*, 87(3):268–279, 2012. ISSN 0025-6196. doi: https://doi.org/10.1016/j.mayocp.2011.12.008. URL https://www.sciencedirect.com/science/article/pii/S0025619612001231.

Nicholas D. Skomrock, Michael A. Schwemmer, Jordyn E. Ting, Hemang R. Trivedi, Gaurav Sharma, Marcia A. Bockbrader, and David A. Friedenberg. A characterization of brain-computer interface performance trade-offs using support vector machines and deep neural networks to decode movement intent. *Frontiers in Neuroscience*, 12, 2018. ISSN 1662-453X. doi: 10.3389/fnins.2018.00763. URL https://www.frontiersin.org/articles/10.3389/fnins.2018.00763.

Yonghao Song, Xueyu Jia, Lie Yang, and Longhan Xie. Transformer-based spatial-temporal feature learning for eeg decoding. *ArXiv*, abs/2106.11170, 2021.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014. URL http://jmlr.org/papers/v15/srivastava14a.html.

Jiayao Sun, Jin Xie, and Huihui Zhou. Eeg classification with transformer-based models. In *2021 IEEE 3rd Global Conference on Life Sciences and Technologies (LifeTech)*, pages 92–93, 2021. doi: 10.1109/LifeTech52111.2021.9391844.

Tijmen Tieleman, Geoffrey Hinton, et al. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

Mariska J. Vansteensel, Elmar G.M. Pels, Martin G. Bleichner, Mariana P. Branco, Timothy Denison, Zachary V. Freudenburg, Peter Gosselaar, Sacha Leinders, Thomas H. Ottens, Max A. Van Den Boom, Peter C. Van Rijen, Erik J. Aarnoutse, and Nick F. Ramsey. Fully implanted brain–computer interface in a locked-in patient with als. *New England Journal of Medicine*, 375(21):2060–2066, 2016. doi: 10.1056/NEJMoa1608085. URL https://doi.org/10.1056/NEJMoa1608085. PMID: 27959736.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Ksenia Volkova, Mikhail A. Lebedev, Alexander Kaplan, and Alexei Ossadtchi. Decoding movement from electrocorticographic activity: A review. *Frontiers in Neuroinformatics*, 13, 2019. ISSN 1662-5196. doi: 10.3389/fninf.2019.00074. URL https://www.frontiersin.org/articles/10.3389/fninf.2019.00074.

Ping Wang, Aimin Jiang, Xiaofeng Liu, Jing Shang, and Li Zhang. Lstm-based eeg classification in motor imagery tasks. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(11):2086–2095, 2018. doi: 10.1109/TNSRE.2018.2876129.

Francis R. Willett, Donald T. Avansino, Leigh R. Hochberg, Jaimie M. Henderson, and Krishna V. Shenoy. High-performance brain-to-text communication via handwriting. *Nature*, 593 7858:249–254, 2021.

D. Randall Wilson and Tony R. Martinez. The general inefficiency of batch training for gradient descent learning. *Neural networks : the official journal of the International Neural Network Society*, 16 10:1429–51, 2003.

B Wodlinger, J E Downey, E C Tyler-Kabara, A B Schwartz, M L Boninger, and J L Collinger. Ten-dimensional anthropomorphic arm control in a human brain-machine interface: difficulties, solutions, and limitations. *Journal of Neural Engineering*, 12(1):016011, dec 2014. doi: 10.1088/1741-2560/12/1/016011. URL https://doi.org/10.1088/1741-2560/12/1/016011.

Jonathan R. Wolpaw and Dennis J. McFarland. Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *Proceedings of the National Academy of Sciences*, 101(51):17849–17854, 2004. doi: 10.1073/pnas.0403504101. URL https://www.pnas.org/doi/abs/10.1073/pnas.0403504101.

Ziqian Xie, Odelia Schwartz, and Abhishek Prasad. Decoding of finger trajectory from ECoG using deep learning. *Journal of Neural Engineering*, 15(3):036009, feb 2018. doi: 10.1088/1741-2552/aa9dbe. URL https://doi.org/10.1088/1741-2552/aa9dbe.

Weibo Yi, Shuang Qiu, Kun Wang, Hongzhi Qi, Lixin Zhang, Peng Zhou, Feng He, and Dong Ming. Evaluation of eeg oscillatory patterns and cognitive process during simple and compound limb motor imagery. *PLOS ONE*, 9(12): 1–19, 12 2014. doi: 10.1371/journal.pone.0114853. URL https://doi.org/10.1371/journal.pone.0114853.

Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *CoRR*, abs/1511.07122, 2016.

Dalin Zhang, Lina Yao, Kaixuan Chen, and Jessica Monaghan. A convolutional recurrent attention model for subject-independent eeg signal analysis. *IEEE Signal Processing Letters*, 26(5):715–719, 2019. doi: 10.1109/LSP.2019.2906824.

Ruilong Zhang, Qun Zong, Liqian Dou, Xinyi Zhao, Yifan Tang, and Zhiyu Li. Hybrid deep neural network using transfer learning for eeg motor imagery decoding. *Biomedical Signal Processing and Control*, 63:102144, 2021. ISSN 1746-8094. doi: https://doi.org/10.1016/j.bspc.2020.102144. URL https://www.sciencedirect.com/science/article/pii/S1746809420302901.

Jie Zhou, Ming Meng, Yunyuan Gao, Yuliang Ma, and Qizhong Zhang. Classification of motor imagery eeg using wavelet envelope analysis and lstm networks. In *2018 Chinese Control And Decision Conference (CCDC)*, pages 5600–5605, 2018. doi: 10.1109/CCDC.2018.8408108.

2

Decoding ECoG signal into 3D hand translation using deep learning

This chapter contains article published in Journal of Neural Engineering: Śliwowski, M., Martin, M., Souloumiac, A., Blanchart, P., and Aksenova, T., (2022). Decoding ECoG signal into 3D hand translation using deep learning. Journal of Neural Engineering, 19. URL https://doi.org/10.1088/1741-2552/ac5d69

Journal of Neural Engineering

PAPER

Decoding ECoG signal into 3D hand translation using deep learning

RECEIVED
8 October 2021

REVISED
2 March 2022

ACCEPTED FOR PUBLICATION
14 March 2022

PUBLISHED
31 March 2022

Maciej Śliwowski[1,2,*], **Matthieu Martin**[1], **Antoine Souloumiac**[2], **Pierre Blanchart**[2] **and Tetiana Aksenova**[1,*]

[1] Université Grenoble Alpes, CEA, LETI, Clinatec, F-38000 Grenoble, France
[2] Université Paris-Saclay, CEA, List, F-91120 Palaiseau, France
[*] Authors to whom any correspondence should be addressed.

E-mail: maciej.sliwowski@cea.fr **and** tetiana.aksenova@cea.fr

Keywords: brain-computer interface, ECoG, motor imagery, hand movements, deep learning, convolutional neural networks, tetraplegia

Abstract

Objective. Motor brain-computer interfaces (BCIs) are a promising technology that may enable motor-impaired people to interact with their environment. BCIs would potentially compensate for arm and hand function loss, which is the top priority for individuals with tetraplegia. Designing real-time and accurate BCI is crucial to make such devices useful, safe, and easy to use by patients in a real-life environment. Electrocorticography (ECoG)-based BCIs emerge as a good compromise between invasiveness of the recording device and good spatial and temporal resolution of the recorded signal. However, most ECoG signal decoders used to predict continuous hand movements are linear models. These models have a limited representational capacity and may fail to capture the relationship between ECoG signal features and continuous hand movements. Deep learning (DL) models, which are state-of-the-art in many problems, could be a solution to better capture this relationship. *Approach.* In this study, we tested several DL-based architectures to predict imagined 3D continuous hand translation using time-frequency features extracted from ECoG signals. The dataset used in the analysis is a part of a long-term clinical trial (ClinicalTrials.gov identifier: NCT02550522) and was acquired during a closed-loop experiment with a tetraplegic subject. The proposed architectures include multilayer perceptron, convolutional neural networks (CNNs), and long short-term memory networks (LSTM). The accuracy of the DL-based and multilinear models was compared offline using cosine similarity. *Main results.* Our results show that CNN-based architectures outperform the current state-of-the-art multilinear model. The best architecture exploited the spatial correlation between neighboring electrodes with CNN and benefited from the sequential character of the desired hand trajectory by using LSTMs. Overall, DL increased the average cosine similarity, compared to the multilinear model, by up to 60%, from 0.189 to 0.302 and from 0.157 to 0.249 for the left and right hand, respectively. *Significance.* This study shows that DL-based models could increase the accuracy of BCI systems in the case of 3D hand translation prediction in a tetraplegic subject.

1. Introduction

Brain-computer interfaces (BCIs) enable people to interact with their environment using a direct connection between their brain and external effectors. Such devices could improve a paralyzed person's quality of life by offering them a certain autonomy. Several communication paradigms have been designed for a broad range of tasks, like keyboard typing or binary decision making. This study focus on motor imagery (MI) based BCI, which can be used by a tetraplegic for continuous and asynchronous [1] control of complex effectors, for example, exoskeleton [2]. More precisely, we studied upper-limb movements imagination because regaining arm and hand motor functions are at the

top of the priority list in individuals with tetraplegia [3, 4].

In the MI BCI paradigm, subjects imagine or attempt movements that cause changes in brain activity. These changes are caught by a recording device that is the first part of a BCI system. Acquired signal characteristics strongly depend on recording devices. From the most invasive to less invasive, the most common types of recordings are: intracortical microelectrode arrays (MEAs) placed inside the brain tissue [5–7], intracranial electrocorticography (ECoG) recording signal from the brain surface [2, 8, 9] and electroencephalography (EEG) monitoring the brain activity with a set of electrodes placed over the scalp [10–13]. This study focuses on ECoG-based BCI because it has better signal characteristics than noninvasive methods while decreasing the risk connected to implantation and biocompatibility issues compared to MEA. ECoG signal is also more stable in time [14].

The next step after signal recording is to extract features from the signal. Those features represent brain activity in a form that a decoder can exploit. The most common and effective representation of ECoG and EEG signals for MI BCI are time-frequency features [2, 15–19] or low-frequency component (LFC)/local motor potential [15, 20–22].

Another important step in a typical BCI system is translating brain signal features into BCI commands with a decoder. Decoding performance is crucial for the quality of interaction between humans and computers. Higher accuracy of decoding improves correctness and speed of interaction. Most current decoders use supervised machine learning (ML) algorithms to predict BCI commands.

Predicting BCI commands based on brain signals is a challenging task due to several limitations originating from the nature of the application. Recorded brain signals have a strong component of noise generated by spontaneous brain activity unrelated to the task. The recorded signal is nonstationary in time (intra-subject variability) which often makes models valid only for a limited time and requires decoder retraining. BCI experiments are monotonous and time-consuming, so ML models have to be trained with small datasets to reduce the time needed for calibration. Another important constraint is the need for real-time decoding—the whole system should produce several movement predictions per second. In the case of tetraplegic subjects, the real intention of the patient is not known to the experimenters. Even if a subject is provided with explicit instruction, actual imagination patterns can be affected by several factors, e.g. attention and fatigue levels.

Several types of algorithms were used to decode brain signals. Majority of studies use 'conventional' ML techniques [23] for decoding of hand movement. In the case of intracortical recordings, linear models, e.g. Ridge linear regression [6, 7] and Kalman filtering [5, 24] were applied to decode continuous imagined hand translation. For ECoG signal decoding, many studies focus on discrete decoding of hand gestures/movements or fingers flexion, typically using standard classifiers. This includes linear discriminant analysis classifier [25–28] or support vector machine [29–34], as well as other methods like naive Bayes classifier [35] or spatiotemporal template matching [36, 37]. Some of them were combined with additional data dimensionality reduction methods like principal component analysis (PCA) [31, 32] or common spatial pattern [28]. Another group of studies demonstrated prediction of continuous outputs—2D and 3D hand movements and fingers flexion—mostly using linear models including linear regression and its variants [15, 18, 38–42] partial least squares (PLS) [43], recursive exponentially weighted n-way partial least squares (REW-NPLS) [2], and Kalman filtering [20, 44, 45].

Recently, deep learning (DL) based methods got more attention and demonstrated their usefulness for a variety of tasks, e.g. in computer vision [46] or natural language processing [47], but 'conventional' ML models (mainly linear) are still predominantly used in ECoG-based studies. However, some studies investigated the use of DL to decode MI patterns in humans. For example, studies analyzing intracortical recordings used recurrent neural networks (RNNs) [48] (classification of 31 letters and continuous decoding), RNN-based autoencoders [49] (continuous decoding) as well as CNN [50], and a combination of RNN and CNN [51] to recognize several MI classes using neuronal firing rates or time-frequency representation of single units activation.

In the EEG domain, DL was used for signal classification from raw signals or hand-crafted features. Most common architectures employed CNN [52–55], RNN [56] (for lower limb continuous kinematics decoding), long short-term memory networks (LSTM) [57, 58], and a mix of CNN and LSTM analyzing time-frequency features [59–61].

A few articles analyzed ECoG signals with DL to predict overt hand/fingers movements. LSTMs were used to classify fingers activation [19, 62] and three different hand gestures [63] from time-frequency features. Rashid *et al* [64] used LSTM to discriminate tongue and hand movements from the raw signal. Xie *et al* [65] predicted continuous flexion and extension of five fingers using end-to-end DL with four spatial/temporal convolutional layers as feature extractors and one LSTM layer to predict fingers activation. Contrary to our study, only non-disabled patients participated in the experiments, which enables the creation of an explicit mapping between brain activity and trajectory and removes the uncertainty introduced in the case of only imagined movements.

For neural population activity recordings, the majority of studies only classified brain signals with DL. However, continuous decoding is the only way to provide paralyzed patients with normal-like control of complex neuroprosthetics. From ECoG signal, multilinear models have been used to predict up to 3D continuous movements [2, 38, 39, 43] whereas DL was only tried to predict 1D finger flexion [65].

As DL achieved state-of-the-art performance in solving several complex problems from different domains, in this work, we propose and evaluate DL architectures to predict 3D continuous hand movements. For the first time, we show that DL can efficiently predict complex, high-dimensional, upper-limb translation from ECoG signals in a tetraplegic patient. We compare several architectures that exploit particular ECoG and MI task characteristics, e.g. CNNs analyze the relationship between electrodes organized on two grids, LSTMs analyze temporal relationships in the ECoG signal. Apart from architecture type, many design choices can modify model complexity and its capacity to recognize patterns. Therefore, an in-depth study of this topic for ECoG signals is crucial, especially considering that the choice of the methods is data specific (depends, for example, on the level of noise, dataset size, the dimensionality of the data) and can drastically influence the performance. Thus, in an ablation study, we investigate the influence of the most important and most data-specific design choices (dropout, batch normalization, activation function, number of layers) on the model's performance.

All methods were compared offline on a dataset recorded in a closed-loop experiment in which adaptive multilinear REW-NPLS [66] models were used to predict 3D hand translation. Recordings were performed with a tetraplegic patient within more than 200 days. DL-based models obtained better performance than multilinear models optimized by REW-NPLS.

2. Methods

2.1. Clinical trial and patient

This study was a part of a clinical trial 'BCI and Tetraplegia' (ClinicalTrials.gov identifier: NCT02550522) approved by French authorities: Agency for the Safety of Medicines and Health Products (Agence nationale de sécurité du médicament et des produits de santé—ANSM) with the registration number: 2015-A00650-49 and the ethical Committee for the Protection of Individuals (Comité de Protection des Personnes—CPP) with the registration number: 15-CHUG-19. Clinical trial details are described by Benabid *et al* [2].

The subject involved in this study was a 28-year-old right-handed man with tetraplegia caused by a C4–C5 spinal cord injury. The patient had little motor control of the upper limbs with American Spinal Injury Association Impairment (ASIA) scores at the elbow: 4 right, 5 left and at the extensors of the wrist: 0 right, 3 left. Other muscles below were all scored 0 on the ASIA scale with a complete sensory-motor deficit. The patient used his left hand to control a wheelchair [2].

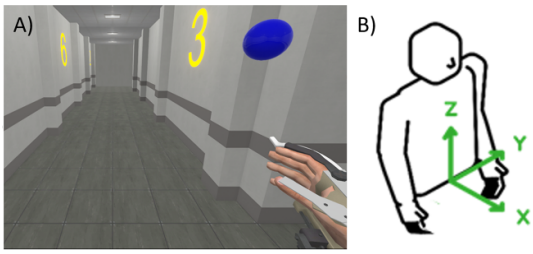


Figure 1. (A) Screenshot from the virtual environment displaying the hand of the avatar and the target. (B) Visualization of the axes of the coordinate system of the virtual avatar.

The first experiment analyzed in this study was recorded 463 days after the implantation. Shortly after the surgery, BCI experimental sessions began. The difficulty of the BCI task was gradually increased starting from discrete control and 1D movements tasks up to 8D movements in one experiment.

2.2. Experiment and dataset

During the experiment, the patient had control over an avatar with a first-person view in a virtual environment. He could control one state out of 5: idle, left or right hand 3D translation, and left or right wrist rotation. A REW-MSLM state classifier (further referred to as gate model) was used to detect the patient's intention and select the controlled effectors [67]. This article focuses only on the left and right hand translation. In that case, spherical targets (10 cm diameter) were displayed in space and the patient task was to reach them successively (figure 1(A)). To control the virtual avatar, the patient used an MI strategy developed in the previous experiments of the clinical trial. This strategy consisted of imagined/attempted, repeated arm, wrist, and finger movements. The patient was instructed to imagine/attempt upper-limb movements without any actual muscle activation and maintain a constant imagination strategy through the experiments.

During the recordings, the hand movement predictions were performed by multilinear models obtained by REW-NPLS [66]. Separate models for each hand and a gate model were incrementally calibrated during the six first sessions and used without re-calibration for the following 37 sessions. The data acquired during these sessions was used to perform simulations of models offline training to evaluate their potential benefits for this application. To ensure equal conditions for further comparisons, we decided to retrain all models on the same datasets.

Targets were placed in 28 (LH) and 23 (RH) positions during the experiments (see appendix A for

Table 1. Datasets size in the number of trials and length of the recordings.

	Left hand		Right hand	
	Trials	Duration (min)	Trials	Duration (min)
Calibration	174	42	164	39
Test	691	314	649	327

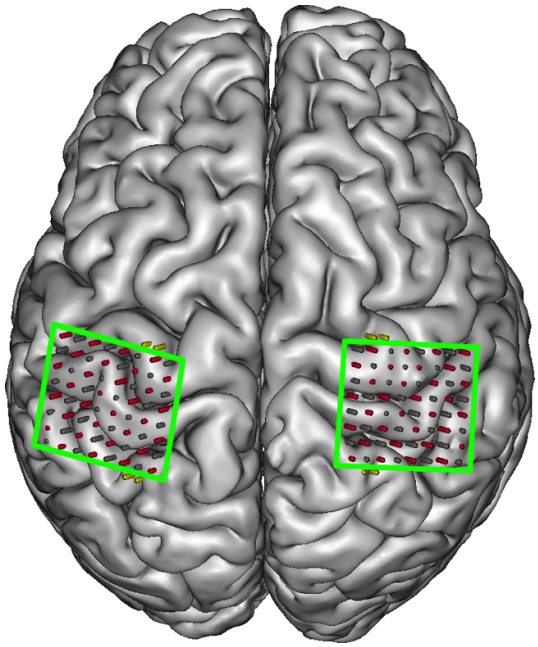


Figure 2. Position of the implants (green) and electrodes (red) on an MRI reconstruction of the patient's brain.

targets positions visualization). The number of trials and minutes of the recorded signal are given for both hands and the calibration and test datasets in table 1.

2.3. Signal recording

Brain signals were recorded by two WIMAGINE® ECoG implants [68] more than 15 months after the surgery. Implants were placed over the left and right primary motor and sensory cortex (figure 2) controlling upper limbs. This position aims to provide the best signal for MI decoding [2]. Each implant was composed of an 8×8 grid of electrodes, but only half of the electrodes, arranged in a chessboard-like manner, were used during the experiment due to the data transfer limit. The signal sampling frequency was 586 Hz, while cursor and target positions in the virtual environment were recorded at 10 Hz.

2.4. Preprocessing and feature extraction

We extracted time-frequency features from each ECoG channel using continuous complex wavelet transform with 15 Morlet wavelets whose central frequencies were regularly chosen between 10 and 150 Hz. The procedure was performed for each one-second-long window i with 90% overlap. Absolute values of the wavelet coefficients were finally averaged over 0.1 s long windows. As a result, we obtained a feature tensor $\underline{\mathbf{X}}_i \in \mathbb{R}^{64\times15\times10}$ whose dimensions correspond to ECoG channels, frequency bands, and time steps.

2.5. Loss function

The main problem considered in this study was to predict 3D hand translation from ECoG time-frequency features. At each time step i, the desired hand movement $\mathbf{y}_i$ was defined as $\mathbf{t}_i - \mathbf{c}_i$ where $\mathbf{t}_i$ and $\mathbf{c_i}$ respectively correspond to the target and the cursor (point of the avatar hand) position (figure 3). The coordinate system origin was placed in the pelvis of the virtual avatar (figure 1(B)). Our problem was then to predict $\mathbf{y}_i$ from the feature tensor $\underline{\mathbf{X}}_i$. Hand movement predictions $\hat{\mathbf{y}}_i$ were performed every $\tau = 100$ ms and the cursor moved according to this direction by the vector $\mathbf{m}_i$ until the next prediction. Since predicted trajectories were followed only for 100 ms (maximum hand speed equal 1 cm/100 ms), we compared the predicted and desired vectors with respect to their direction using cosine similarity defined as:

$$\mathrm{CS}(\mathbf{y}_i, \hat{\mathbf{y}}_i) = \frac{\mathbf{y}_i \cdot \hat{\mathbf{y}}_i}{\|\mathbf{y}_i\| \cdot \|\hat{\mathbf{y}}_i\|} = \cos\alpha_i, \tag{1}$$

calculating the cosine of the α_i angle between $\mathbf{y}_i$ and $\hat{\mathbf{y}}_i$ vectors. $\mathrm{CS} \in [-1, 1]$ is equal to 1 when vectors have exactly the same direction, 0 for orthogonal vectors, and -1 for opposite vectors. We used cosine loss defined as:

$$\mathrm{CL}(\mathbf{y}_i, \hat{\mathbf{y}}_i) = 1 - \mathrm{CS}(\mathbf{y}_i, \hat{\mathbf{y}}_i) \tag{2}$$

as an objective function to train DL models.

2.6. REW-NPLS

As a baseline model, we used multilinear models obtained by REW-NPLS regression algorithm [66]. PLS regression projects both input and response variables to a low-dimensional latent space whose components are designed to provide the highest correlation between input and output variables. This regression method is particularly well-suited in the case of high-dimensional and nonindependent input data. During the experiments, these multilinear models were trained and used to provide real-time control of the avatar to the patient. The latent space dimension was limited to 100 and its optimal dimension was determined every 15 s using recursive validation [66]. To obtain the results reported in this study, we retrained the REW-NPLS model in a pseudo-online manner with an update after each 15 s, which simulates the online training. Next, we used this pseudo-online model to compute predictions on the test set.

2.7. Multilayer perceptron

Multilayer perceptron (MLP) is a classic method used in ML to model complex functions. MLP treats each

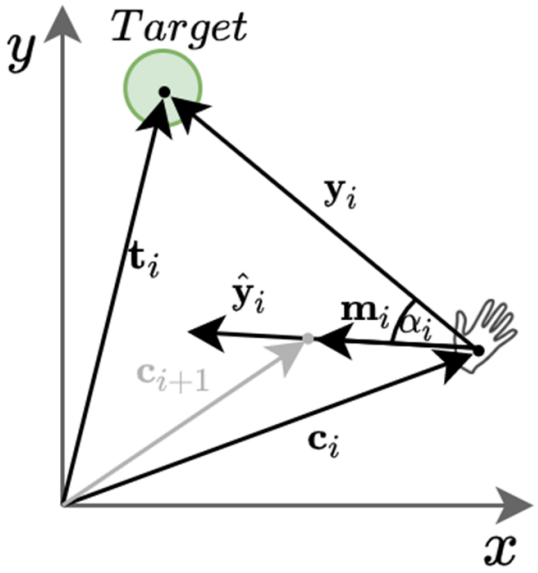


Figure 3. Visualization of movement scheme in 2D. Green circle indicates target position, vectors $\mathbf{c}_i$ and $\mathbf{c}_{i+1}$ are the cursor position vectors for respectively samples i and $i+1$, $\mathbf{t}_i$ is the target position vector, $\mathbf{y}_i$ is the desired trajectory vector for sample i, $\mathbf{m}_i$ is the actual move performed by hand based on the prediction $\hat{\mathbf{y}}_i$. α_i is the angle between $\mathbf{y}_i$ and $\hat{\mathbf{y}}_i$.

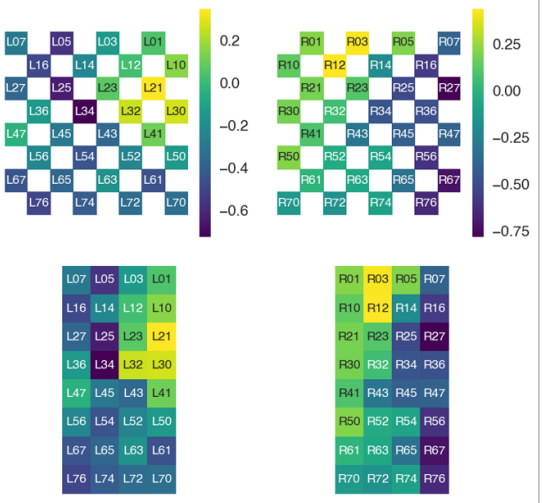


Figure 4. Features from one randomly chosen sample in a frequency band centered at 10 Hz. Values are arranged on a grid approximating actual electrodes' positions for left and right implants, respectively. Top row grids also include electrodes that were not recording signals (white squares). Bottom row plots show electrodes' arrangement after removing missing values. The distance between implants is not preserved.

feature with a set of independent weights to create a representation consisting of neurons excitations that are next processed with nonlinear activation function and finally passed to the next layer. In our experiments, the number of fully connected (FC) layers varied between 1 and 5. Each FC was composed of 50 neurons with ReLu activation. In addition, batch normalization [69] and dropout [70] with probability of neuron being zeroed equal 0.5 were applied between hidden layers (except last). Both methods have a strong regularization effect and are commonly used to increase the generalization of models. This is especially important in the case of small and noisy datasets. The input to MLP was a flattened tensor $\underline{\mathbf{X}}_i$, consisting of 9600 features standardized using Z-score.

2.8. 2D CNN

One reason why convolutional neural networks are widely used in image processing is that they effectively recognize similar patterns in different parts of images. In each layer, several convolutional filters with trainable parameters are shifted over an image to detect patterns and obtain feature maps. This enables CNNs to limit the number of parameters needed to solve complex problems, generalize better, and effectively use information encoded in the local structure of the data and spatial relationships between pixels (for a detailed description of CNN, see explanation by Goodfellow *et al* [71]). This kind of architecture can be well-suited for brain signals analysis if its inputs are shaped in a way that enables spatial or temporal pattern detection. Methods proposed further in this section are inspired by architectures used in computer vision [72] and EEG signal processing (mainly by Bashivan *et al* [59] but also by other methods [10, 73]).

2.8.1. Input representation and processing

We propose to preserve the spatial relationship between signals from different electrodes by using an analogous transformation to the one proposed for EEG [59]. Contrary to EEG, ECoG electrodes are distributed on two dense square grids. Hence, to obtain a representation of the actual electrode array, we projected recorded signals onto a grid according to the approximate physical electrodes position presented in figure 2. We created two arrays of 64 electrodes placed on a grid of shape 8×8 with only half of the electrodes recording signal. We removed unused electrodes and represented each implant with a rectangle of shape 8×4 merging neighboring columns of electrodes (see figure 4). The introduced distortion of the image does not affect convolution as we preserved the spatial neighborhood of each electrode.

Then, the input to all CNN models was a tensor of shape $2 \times 8 \times 4 \times 15 \times 10$, where dimensions correspond to the number of implants, height and width of implants, frequency bands, and time steps, respectively. Features were Z-score standardized using mean and standard deviation per frequency band. Each observation may be interpreted as a time series of consecutive spatio-frequential representations that form two 'images' with 15 frequency channels, an analog of the three RGB channels used in computer vision.

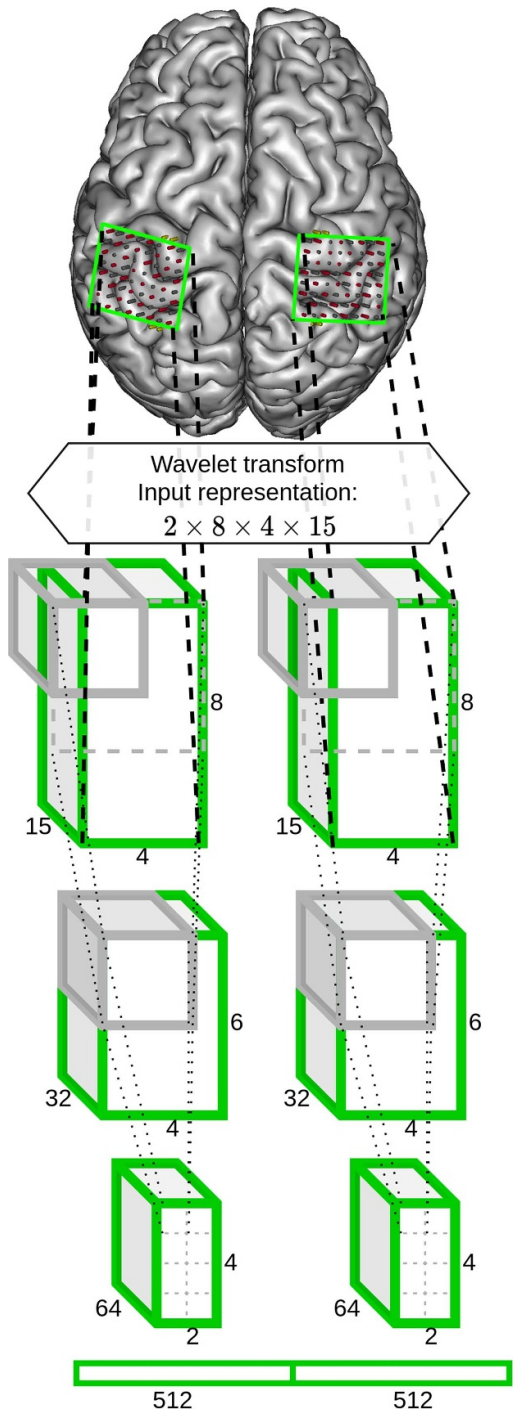


Figure 5. 2D CNN processing visualization. Two implants record signals, then time-frequency features are extracted using wavelet transform. Features from each implant (green cuboids) are analyzed using filters (gray cuboid) by consecutive convolutional layers to obtain representation in the form of a vector. We also marked the receptive field (gray dashed rectangle) of the one particular feature from the last layer. Note that this is only a diagram. For simplicity, we visualized a model with two convolutional blocks and we skipped activation, batch normalization, dropout. See detailed specification in tables 2 and 3.

Proposed two-dimensional CNN (2D CNN) analyzes each time step independently by performing convolution only in space (two dimensions). The same convolutional kernels were applied separately for each implant (see figure 5).

Table 2. Architecture specification of 2D CNN model with two convolutional blocks.

Block	Layer	Filters	Padding	Output shape
	Input			$2 \times 8 \times 4 \times 15$
1	Convolution	32	(0, 1)	$2 \times 6 \times 4 \times 32$
	ReLU			$2 \times 6 \times 4 \times 32$
	Batch normalization			$2 \times 6 \times 4 \times 32$
	Dropout			$2 \times 6 \times 4 \times 32$
2	Convolution	64	(0, 1)	$2 \times 4 \times 2 \times 64$
	ReLU			$2 \times 4 \times 2 \times 64$
	Dropout			$2 \times 4 \times 2 \times 64$

After the final convolutional layer, the features extracted from one second of the signal were flattened and aggregated. We present details of temporal information analysis in sections 2.9–2.12.

2.8.2. Convolutional block design

CNNs consist of multiple layers organized in convolutional blocks, usually composed of a convolutional layer and a nonlinear activation function followed by max pooling layer [71]. Optionally, one can add batch normalization [69] and dropout [70]. The optimal structure of convolutional block depends on the characteristic of the problem, e.g. size and type of data, signal, and problem characteristic. Further in this section, we describe the convolutional block design choices that we made in the study.

2.8.2.1. Batch normalization and dropout

We decided to use both dropout and batch normalization to achieve a strong regularization effect, as our dataset suffers from a small number of samples, low signal to noise ratio, and uncertain labels. We decided to include batch normalization between the activation and the dropout layer, which might be more effective in the case of ReLU activation [74]. Merged batch normalization and dropout can be interpreted as an independent-component layer, reducing mutual information between neurons in the input [74]. The batch normalization layer was included in all convolutional blocks except last.

2.8.2.2. No-padding

Another design choice that we propose to use is to remove the max pooling layer. Instead, we used no-padding option (removing padding around grid edges) to reduce the size of the 'images' (see padding sizes in table 3). We considered this variant as a reasonable choice because our 'images' (size 8×4) are much smaller than typical computer vision images (e.g. $224 \times 224 \times 3$ for ResNet [75]) and spatio-frequential EEG 'images' ($32 \times 32 \times 3$ analyzed by Bashivan *et al* [59]). Reducing the size of our images too much could have resulted in the loss of useful information. A convolution operation (kernel size 3×3 and stride equal to 1) without padding

Table 3. Description of the 2D CNN architectures with different number of layers. The parameters of convolutional blocks are indicated in the form 'conv(padding height, padding width)-<number of channels>'. The last row presents the number of extracted features.

2D CNN				
1 layer	2 layers	3 layers	4 layers	5 layers
Input size: $2 \times 8 \times 4 \times 15$				
conv(0, 0)-32	conv(0, 1)-32	conv(0, 1)-32	conv(0, 1)-32	conv(0, 1)-32
			conv(1, 1)-32	conv(1, 1)-32
	conv(0, 0)-64	conv(0, 1)-64	conv(0, 1)-64	conv(0, 1)-64
				conv(1, 1)-64
		conv(0, 0)-128	conv(0, 0)-128	conv(0, 0)-128
Flatten				
768	1024	1024	1024	1024

reduces an image size of 2 pixels along each dimension, whereas a max pooling operation (kernel size 2×2, stride 2) halves its dimension.

2.8.2.3. Activation function

Following the results presented by Schirrmeister *et al* [73] and Lawhern *et al* [10] regarding activation function choice, we considered two potential candidates—ReLU, used by Bashivan *et al* [59], and exponential linear unit (ELU) [76] that was proven to be more effective in the case of EEG processing [10, 73].

Finally, we used convolutional blocks consisting of: convolutional layer $\rightarrow$ ReLU $\rightarrow$ batch normalization $\rightarrow$ dropout. All convolutional layers use typical parameters: kernel size equal to 3×3, stride equal to 1, and variable size of zero padding (denoted in table 3). For dropout we used probability of zeroing whole channel equal 0.5.

2.8.3. Number of blocks

Another important architecture choice is the number of convolutional blocks. In computer vision, typical models consist of dozens of convolutional layers (e.g. ResNet [75] with more than 1000 convolutional layers, VGG [72] with up to 19 layers, Inception-v3 [77] with 48 layers). However, methods proposed for brain signals analysis used a significantly lower number of layers (e.g. ShallowConvNet and DeepConvNet [73] with two and five convolutional layers respectively, EEGNet [10] with three layers, Bashivan *et al* [59] proposed architectures with up to seven convolutional layers). To investigate this problem, the number of convolutional blocks in the 2D CNN was set between 1 and 5 (table 3). The optimal model depth was selected based on the session-wise cross-validated CS obtained with the simplest proposed architecture: CNN2D + FC (see section 2.9). This depth value was then used for more complex approaches.

2.9. CNN2D + FC

We proposed several architectures to aggregate temporal information from representation extracted by the 2D CNN model. In the most straightforward approach, the features extracted by 2D CNN at each time step were concatenated and given as an input to an FC layer composed of three neurons (figure 6(A)). It provided three outputs that correspond to $\hat{\mathbf{y}}_i$ components. Further, we will refer to this approach as CNN2D + FC.

2.10. CNN2D + LSTM

LSTM [78] is a type of RNN that can efficiently analyze long temporal relationships in the data. As in typical RNNs, a module called cell is applied to each time step. LSTM cell carries relevant information through time and decides what to forget and what to store based on the current input [79]. In order to analyze temporal information, we stacked two LSTM layers at the top of 2D CNN (see figure 6(B)). At each time step, the first LSTM cell with a hidden state of size 50 was provided with the flattened output features of the 2D CNN. The second LSTM layer had three neurons that were used to provide three output coordinates for $\hat{\mathbf{y}}_i$ components. Stacking two LSTM layers enabled the network to model complex temporal relationships and efficiently incorporate information from the last second of the signal. It was already proven effective in a variety of tasks including ECoG decoding [19, 62, 65].

2.10.1. CNN2D + LSTM + MT

As an extension of CNN2D + LSTM, we also tested a modification, referred to readers as multiple trajectory prediction (MT), in which each LSTM output from the last layer is used for network training (see figure 6(C)). It means that $\hat{\mathbf{y}}_i$ and $\mathbf{y}_i$ were compared for ten consecutive time steps to compute the model error. This was possible because desired trajectory vector $\mathbf{y}_i$ was registered every 100 ms. Thus, LSTM may utilize the relation between close desired trajectory to create a more general representation of the system state. As a consequence, the loss function was modified as follows:

$$\mathrm{CL}_{\mathrm{MT}}(\mathbf{y}_i, \hat{\mathbf{y}}_i) = \sum_{j=i-N-1}^{i} \mathrm{CL}(\mathbf{y}_j, \hat{\mathbf{y}}_j), \qquad (3)$$

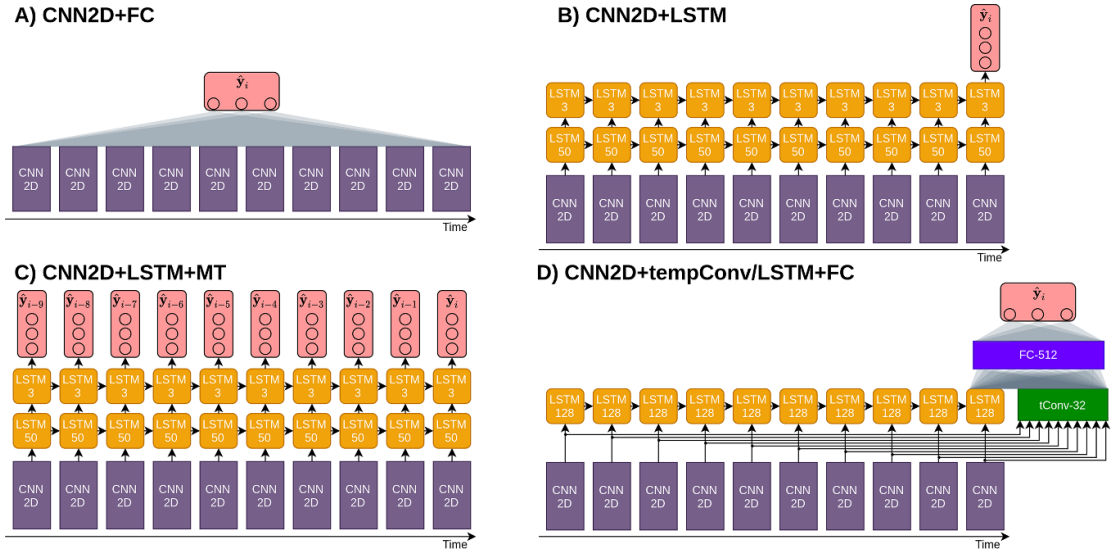


Figure 6. Four ways of temporal aggregation are presented for architectures based on the 2D CNN model. Numbers after the name of the layer denote hidden state size in the case of LSTM, filter number in the case of tConv, and the number of neurons in the case of FC. Visualization inspired by figures presented by Bashivan *et al* [59].

where N is the number of $\mathbf{y_i}$ recorded during a one-second-long window. In our case, N was equal to 10 because the desired trajectory was recorded at 10 Hz.

2.11. CNN2D + tConv/LSTM + FC

To compare our architectures with the state-of-the-art, the best temporal aggregation variant proposed by [59] (variant D) was included in our analysis. For consistency, we will refer to this approach as CNN2D + tConv/LSTM + FC. Bashivan *et al* [59] used one LSTM layer with a hidden state of size 128 to analyze temporal information in parallel with temporal convolution. The temporal convolution (tConv) consisted of 32 filters of size 3 that were shifted over features extracted by 2D CNN in the time domain. It enables recognizing time-invariant patterns in the data (similar patterns that occur at different moments). Finally, the output of tConv and the LSTM cell from the last time step were concatenated and fed into an FC layer with 512 neurons followed by an FC with 3 neurons predicting $\hat{\mathbf{y}}_i$ (see figure 6(D)). A dropout layer was added before each FC layer.

2.12. CNN3D + FC

We tested one more approach that can analyze spatial and temporal patterns at the same time at all levels of the data processing. Inspired by three-dimensional CNNs (3D CNNs) [80] used to recognize human actions on videos, we propose to extend the 2D CNN model and to perform convolution not only in space but also in time (see table 4). These architectures have the advantage of propagating temporal information from the first to the last layer. Similar to 2D CNN, an FC was used as the output layer and the same convolutional parameters were chosen (except for convolution performed in time).

Table 4. Architecture specification of 3D CNN model with two convolutional blocks.

Block	Layer	Filters	Padding	Output shape
	Input			$2 \times 8 \times 4 \times 15 \times 10$
1	Convolution	32	(0, 1, 0)	$2 \times 6 \times 4 \times 32 \times 8$
	ReLU			$2 \times 6 \times 4 \times 32 \times 8$
	Batch normalization			$2 \times 6 \times 4 \times 32 \times 8$
	Dropout			$2 \times 6 \times 4 \times 32 \times 8$
2	Convolution	64	(0, 0, 0)	$2 \times 4 \times 2 \times 64 \times 6$
	ReLU			$2 \times 4 \times 2 \times 64 \times 6$
	Dropout			$2 \times 4 \times 2 \times 64 \times 6$

2.13. Sensitivity analysis

To analyze model behavior and identify important features, we computed the gradient of the model's outputs with respect to the inputs on the test datasets—network jacobian $\underline{\mathbf{J}} \in \mathbb{R}^{N \times 2 \times 8 \times 4 \times 15 \times 10 \times 3}$ representing the sensitivity of each network output to the inputs. For each input feature, the higher the absolute gradient value, the stronger the influence on the prediction. For visualization, the sensitivity of the three outputs was averaged. We analyzed feature importance in three domains: spatial with projection on two implants, frequential within 15 frequency bands from 10 to 150 Hz, and temporal within 10 time steps containing 1 s of the signal.

2.14. Optimization, hyperparameters, and evaluation

We selected optimal hyperparameters values (initial learning rate, weight decay, and batch size) with the tree of Parzen estimators (TPE) [81] algorithm from hyperopt package. The selection was performed

Table 5. Cosine similarity for each proposed method for left and right hand datasets. Asterisks denote models which had significantly different cosine similarity in comparison to CNN2D + LSTM + MT. Bold text indicates the highest cosine similarity for each hand.

	Cosine similarity	
	Left hand	Right hand
CNN2D + LSTM + MT	0.302 ± 0.017	$\mathbf{0.249 \pm 0.008}$
CNN2D + FC	0.296 ± 0.015	0.237 ± 0.011
CNN2D + tConv/LSTM + FC	$\mathbf{0.306 \pm 0.017}$	$0.223 \pm 0.012^*$
CNN3D + FC	0.294 ± 0.016	$0.226 \pm 0.011^*$
CNN2D + LSTM	$0.264 \pm 0.015^*$	$0.222 \pm 0.011^*$
MLP	$0.229 \pm 0.01^*$	$0.205 \pm 0.011^*$
Multilinear model	0.189^*	0.157^*

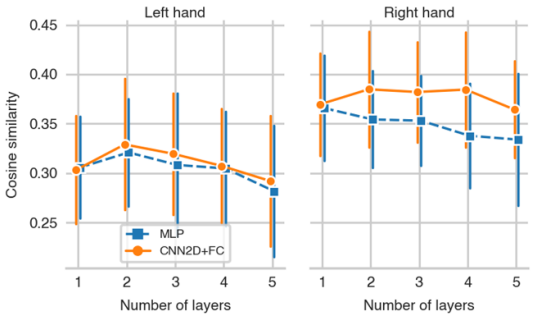


Figure 7. Influence of the number of layers on model performance. Error bars denote the standard deviation of six folds.

on the left hand calibration dataset using session-wise 6-fold cross-validation. Hyperparameters values selected after 200 iterations can be found in the appendix in table C1. The optimal number of layers was selected using hyperparameters presented in table C2.

Models were trained for 60 epochs with early stopping (to limit overfitting) with the patience of 20 epochs without any loss function improvement on the validation dataset. The learning rate was changed using cosine annealing [82]. The last 10% of the calibration dataset was used for validation and the rest was used for training. We discarded the random cross-validation scheme as the high correlation between neighboring samples in time would have biased performance on the validation datasets. The model which achieved the best score on the validation dataset was retained. It was used to compute average CS on the test dataset (table 5).

To limit the influence of network weights initialization and the optimization process on the results, each model was trained five times. The mean and standard deviation of the performance indicator was computed. We used a T-test for independent samples to assess the statistical significance of the difference in performance between architectures.

We used PyTorch [83] and skorch [84] for DL models training and evaluation, MATLAB [85] for multilinear models training and evaluation, Seaborn [86] and Matplotlib [87] for data visualization, and Pandas [88] for results analysis.

3. Results

Our analysis started with determining the optimal number of layers in the 2D CNN model and the MLPs. Next, we compared all proposed methods in terms of CS on the left and right hand datasets. Then, a detailed comparison of the best DL-based architecture and the state-of-the-art multilinear model is presented. Finally, we analyzed the influence of particular design choices on DL models' performance.

3.1. Number of layers

The influence of the number of layers on the calibration cross-validated accuracy of MLP and CNN2D + FC is presented in figure 7. In the LH case, the best CS was obtained with two layers for both architectures. For the RH, the best CS was obtained with one layer for MLP and again with two layers for CNN. We can observe a decrease in performance when adding more layers starting from 2 (LH) or even 1 (MLP and RH). CNN architecture performed more stable on the RH dataset and decreased accuracy was observed for more than four layers. Based on these results, the number of blocks in 2D CNN and the number of hidden FC layers in MLP were chosen to be two as this choice maximized the average CS over LH and RH results. It simplified further analysis and limited computation times without decreasing the performance.

3.2. Overall model performance

The CS of all evaluated approaches on both test datasets is given in table 5. The CNN2D + LSTM + MT model achieved the best average performance across left and right hand datasets with a CS of 0.302 for the left hand and 0.249 for the right hand, corresponding to 60% and 59% of CS improvement relative to the multilinear model. Comparisons using the T-test showed significant differences ($p < 0.05$) for both hands between CNN2D + LSTM + MT and CNN2D + LSTM, MLP, and multilinear model. Additionally, in the case of the RH dataset, CNN2D + LSTM + MT was also significantly better than CNN3D + FC and CNN2D + tConv/LSTM + FC. All proposed DL methods performed better than the multilinear model. Generally, all models decoded LH movements more accurately than RH movements.

Correlation coefficients between predicted and desired outputs are presented in table 6. CNN2D + LSTM + MT achieved the highest average correlation among all models. A significant decrease of performance can be observed in the X axis (forward/backward movements) in comparison to other directions for all models, reaching correlation close to zero for the MLP. No correlation was observed between the norm of predicted outputs and the distance to the target (maximum 0.047 for REW-NPLS, average over all models: 0.0021 ± 0.26).

Table 6. Pearson's r correlation coefficient between predicted and desired outputs on the test dataset. Bold text indicates the highest correlation for each hand and axis.

	Correlation coefficient							
	Left hand				Right hand			
	X axis	*Y* axis	*Z* axis	Average	*X* axis	*Y* axis	*Z* axis	Average
CNN2D + LSTM + MT	**0.09 ± 0.015**	0.313 ± 0.005	0.292 ± 0.014	**0.232 ± 0.021**	**0.072 ± 0.004**	**0.269 ± 0.008**	0.253 ± 0.011	**0.198 ± 0.014**
CNN2D + tConv/LSTM + FC	0.067 ± 0.028	**0.318 ± 0.009**	**0.294 ± 0.015**	0.226 ± 0.033	0.051 ± 0.015	0.23 ± 0.007	0.246 ± 0.011	0.175 ± 0.02
CNN2D + LSTM	0.054 ± 0.003	0.29 ± 0.01	0.274 ± 0.006	0.206 ± 0.012	0.061 ± 0.012	0.247 ± 0.007	**0.254 ± 0.012**	0.188 ± 0.018
CNN2D + FC	0.065 ± 0.035	0.304 ± 0.007	0.264 ± 0.007	0.211 ± 0.0366	0.063 ± 0.04	0.254 ± 0.034	0.169 ± 0.074	0.162 ± 0.091
CNN3D + FC	0.047 ± 0.023	0.302 ± 0.006	0.265 ± 0.013	0.205 ± 0.027	0.046 ± 0.008	0.232 ± 0.006	0.217 ± 0.011	0.165 ± 0.015
MLP	−0.013 ± 0.01	0.182 ± 0.043	0.197 ± 0.011	0.122 ± 0.045	−0.003 ± 0.011	0.173 ± 0.012	0.199 ± 0.012	0.123 ± 0.02
REW-NPLS	0.036	0.1256	0.2186	0.126	0.046	0.174	0.119	0.113

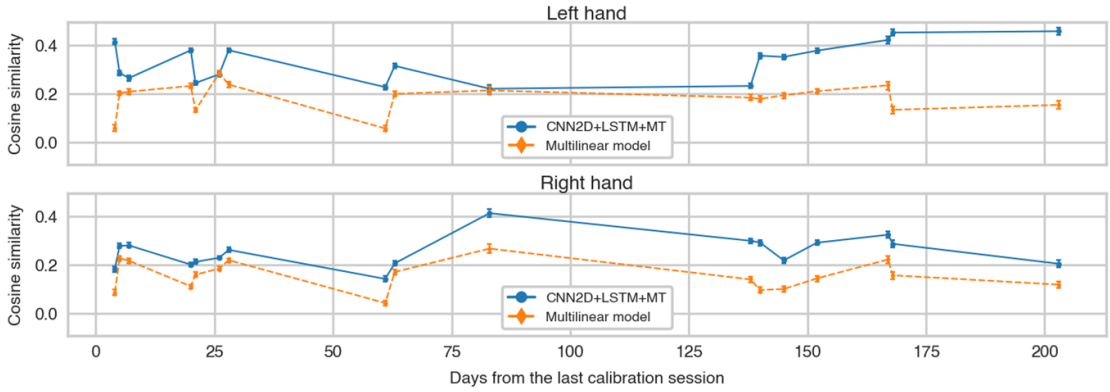


Figure 8. The course of cosine similarity of CNN2D + LSTM + MT and multilinear model in time. Error bars denote 95% mean confidence interval.

3.3. Detailed performance comparison

To understand why CNN2D + LSTM + MT outperformed the multilinear models, we performed a detailed analysis of the accuracy of both methods. The CNN2D + LSTM + MT model used for comparison was randomly selected from the five models trained for that study. Then, the models' performance was compared over time, depending on the distance to the targets and depending on the desired trajectory direction.

Firstly, we compared the stability of the performance over time. The average CS on each day of the experiment is given in figure 8. The DL model obtained a higher CS for a majority of points (15/17) in the case of the LH dataset and all in the RH dataset. For the LH and in comparison to the multilinear model, the CS obtained by the DL models was better (except the 26th day) until the 83rd day, similar on days 83rd and 138th, and even much better until the end of the test set. The improvement is more uniform for the right hand, with an increase starting from the 83rd day after the last calibration session. The CS of CNN2D + LSTM + MT varies similarly to the multilinear REW-NPLS model in the case of the RH dataset. CS was better than zero for all recording days for both models.

The accuracy depending on the distance between the hand cursor and the target is shown for both models in figure 9. The DL model achieved a higher CS than the multilinear model for both hands in almost the whole range of distance (>90%). For both hands

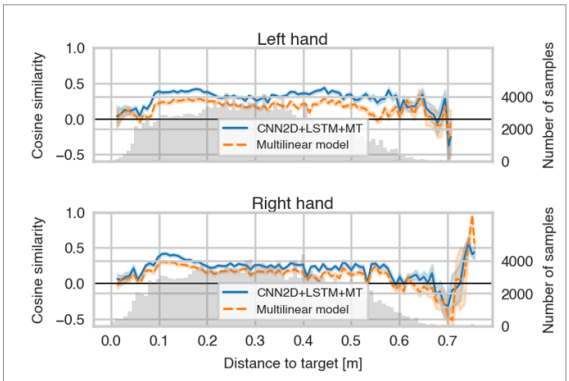


Figure 9. Cosine similarity as a function of the distance between cursor and target. The gray histograms represent the number of samples in the test dataset. Error bands denote 95% mean confidence interval.

and models, a drop in performance occurred when the distance to the target was inferior to 10 cm. The CS variance increased strongly when this distance was superior to 60 cm. There are significantly fewer observations for the edge values of distance to the target, so the performance estimate is noisier.

To determine if the predictions were more accurate in a given direction, we plotted CS on 2D planes (figure 10). CS has a non-uniform distribution—observations for which the patient was asked to move hand backward have a lower CS. CNN2D + LSTM + MT and multilinear model performed better in different directions. There is no wide angle in which performance is below 0.

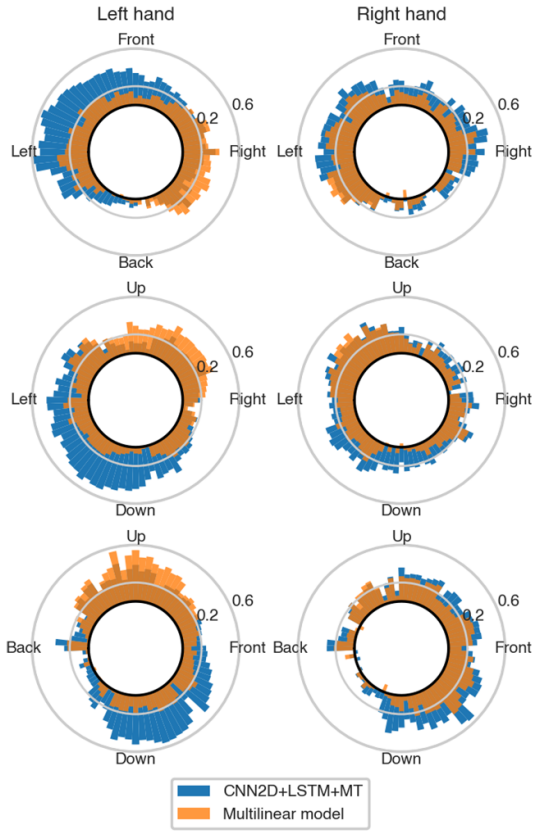


Figure 10. Cosine similarity directional performance of multilinear and CNN2D + LSTM + MT models projected on 2D planes.

3.4. Sensitivity analysis

To determine the inputs that had the strongest influence on the prediction, we visualized the average sensitivity of the DL model on the test set (figure 11). We put together spatial feature importance maps and the approximate shape of the central sulcus (CnS). CnS separates the primary motor cortex and the primary somatosensory cortex. For each hand, we can see that the DL model outputs were more sensitive to changes in ECoG features from the contralateral implant. The highest sensitivity can be observed for the electrodes that are close to the CnS. For the LH movements decoding, the most important electrodes are located in the center of the implant surrounding the CnS of the right cortex, while in the case of the RH, they are posterior to the CnS of the left cortex. The lowest absolute gradient values can be found at the top and bottom edges of the implants.

The most important features in the frequency domain correspond to 20 and 30 Hz (beta rhythm). An increase in importance can also be observed in frequencies higher than 130 Hz.

In the time domain, the importance of input features increases when getting closer to the current time step, with a decrease observed for the features computed with the last 100 ms of the ECoG signal for the RH movements prediction.

3.5. Multiple trajectory influence

To understand better the influence of the MT variant on the performance, we modified CL_{MT} loss function. At time step $j \in [i-N-1, i]$ instead of comparing LSTM output $\hat{\mathbf{y}}_j$ to the corresponding desired trajectory $\mathbf{y}_j$, it was compared to the desired trajectory at the last time step $\mathbf{y}_i$ (the one that is used when only one time step is predicted). Finally, the modified loss function was defined as:

$$\mathrm{CL}_{\mathcal{MT}}(\mathbf{y}_i, \hat{\mathbf{y}}_i) = \sum_{j=i-N-1}^{i} \mathrm{CL}(\mathbf{y}_i, \hat{\mathbf{y}}_j). \tag{4}$$

This enabled us to isolate the influence of the information about desired trajectory variation on the performance from other factors as changes in the optimization process due to providing explicit gradient to the LSTM cell at each time step. Models trained with $\mathrm{CL}_{\mathcal{MT}}$ loss function obtained cosine similarity of 0.285 ± 0.012 and 0.224 ± 0.0077 for the left and right hand respectively. This result is not as good as in the case of standard MT modification (LH: 0.302 ± 0.009, RH: 0.246 ± 0.011).

3.6. Influence of convolutional block design

The convolution blocks had the following chosen structure: dropout ($p = 0.5$), batch normalization, ReLU activation function, and no-padding. To assess the influence of each particular design choice in the convolutional block, we individually removed dropout, batch normalization, replaced no-padding with max pooling, and replaced ReLU with ELU. We compared CS obtained over the test set by each DL model with and without the particular design choices. Results are presented in figure 12. It enabled us to separately estimate the deterioration/improvement related to each design choice. When CS values are negative, it means that not applying the design choice decreased the accuracy. The dropout layer brought the biggest improvement overall for both hands and all CNN-based models except CNN2D + LSTM, where batch normalization had a stronger impact. We observed much smaller improvements in the case of no-padding, batch normalization, and activation function.

As the biggest improvement in CS was due to the dropout, we investigated its optimal value. The CNN2D + LSTM + MT model was trained with different probabilities of zeroing a channel. CS for each hand for different dropout values is given in figure 13. Results look similar for both hands: optimal dropout values were 0.5 and 0.65. High (0.8) or low ($\leqslant$0.15)

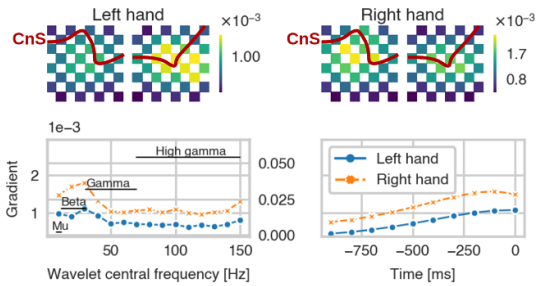


Figure 11. Average sensitivity of the CNN2D + LSTM + MT outputs on the test dataset. Absolute gradient values are plotted as a function of electrode positions (top plots), central wavelet frequency (left bottom plot), and time (right bottom plot). Gradient integrals (averaged over hands) in mu (8–12 Hz), beta (12–31 Hz), gamma (31–70 Hz), and high gamma (70–150 Hz) frequency bands are indicated on the plot with black horizontal lines with the corresponding axis on the right. CnS corresponds to the red line on the heatmaps.

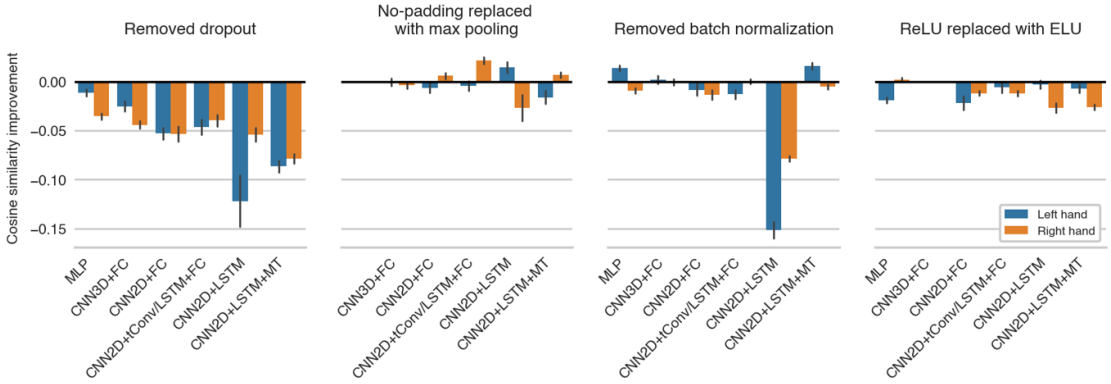


Figure 12. Influence of the architecture choices on the model performance. Models were trained with and without modification for each architecture choice (dropout, no-padding, batch normalization). The difference between the two obtained models was computed to evaluate the influence of particular modifications on the model performance. The lower the cosine similarity difference is, the bigger deterioration is caused by removing the proposed modification from the model.

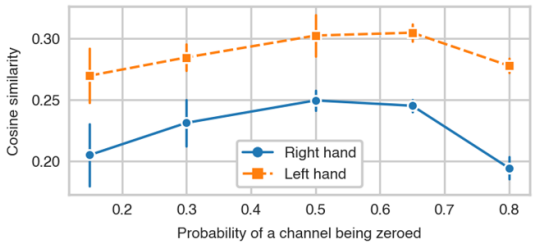


Figure 13. Cosine similarity obtained with CNN2D + LSTM + MT model with two convolutional blocks as a function of dropout levels.

dropout value strongly decreased the network accuracy.

4. Discussion

We evaluated several DL-based methods to predict 3D hand translation. Our offline results show that both MLP and CNN-based models outperform the multilinear model proposed in [66]. A significant improvement was obtained with CNN-based methods in comparison to both MLP and multilinear models. At the same time, CNN2D + LSTM + MT uniformly improved cosine similarity, independently to the distance to the target, and provided better performance than multilinear models in the majority of directions. Increased correlation coefficients for all the axes were observed, indicating higher performance in predicting desired outputs in all the directions. No correlation was observed between the norm of the prediction and the magnitude of the desired output. This is contradictory to the studies analyzing real hand movements in able-bodied population [12, 89]. However, it can be expected because of the cosine loss function that does not depend on the vector's magnitude and the mental task, which consisted of imagined repeated movements encoding better the direction. In our analysis, we did not use the LFC, which had high importance in both studies decoding speed of movement [12, 89].

We demonstrated that CNNs are a reasonable choice to analyze time-frequency ECoG features and may predict complex 3D hand translation. All the presented results were computed offline, without interaction between model and patient. This allowed us to separate decoder performance and patient adaptation influence and compare the models. Our dataset was recorded in a closed-loop experiment, which includes patient's corrections after erroneous

movements and control feedback that influence the user's brain activity. Thus, our estimate of models' performance should be closer to the one obtained during online control compared to open-loop experiments. However, training and evaluation on previously recorded trajectories may lead to overfitting and accommodation to the model used during the experiment and the models' errors inducing additional biases. This can impact the online results, but it is impossible to measure those factors' influence on the results computed in an offline study.

In our dataset, we did not have access to real hand movements. Compared to the open-loop experiment with actual movement recorded, our output variable can be distorted and influenced by the patient's attention level, tiredness, or inexactness of imagination. Moreover, in a closed-loop, a patient has to correct the erroneous movements, complicating the trajectory. As a result, we train models on a distorted and noisy dataset. It also complicates model evaluation and makes the problem more challenging than decoding overt movements.

4.1. Generalization of the results

This study included only one participant. It is then impossible to generalize these results to other patients. However, the proposed models were tested on two different datasets. CNN2D + LSTM + MT model achieved a similar improvement compared to the state-of-the-art multilinear model and MLP for both hands.

The feature importance maps demonstrated that the neuronal patterns for each hand were different. Thus, it showed the DL-based models' capacity to determine relevant features on two datasets. Nevertheless, our results must be confirmed with additional tests (including online evaluation) of proposed models on a bigger group of patients. This is planned in the clinical trial as the next steps after enrollment of new patients.

The sensitivity analysis also revealed that beta and high gamma band activity has a high impact on the predictions, which is coherent with studies analyzing real arm movements in epileptic patients [22, 89]. However, these studies indicated the strongest influence of low-frequency potential on hand movement predictions. Our feature extraction pipeline was not designed to analyze the LFC, so further analysis should use the information transmitted in frequencies lower than 10 Hz.

4.2. Differences in cosine similarity

Our results showed a significant difference in cosine similarity and correlation depending on hands and directions. For example, right hand translations on the test dataset were predicted with significantly lower accuracy than the left hand translations. The reason for that relationship is unclear, especially taking into account an inverse relationship showed by the cross-validation results on the calibration dataset. We hypothesize that the MI patterns used by the patient to control left hand translations are more stable in time than the right hand patterns. This may be caused by differences in the level of residual motor functions in the hands—the patient had better control of his left hand and used it daily to operate a wheelchair. Considering that the test sessions were spread over more than 200 days while calibration was recorded within 3 days, it could explain why cosine similarity was better for left hand on the test sessions and not on the calibration.

Interestingly, movements in the X axis (forward/-backward) were decoded with significantly decreased performance in comparison to Y (left/right) and Z (up/down) axes. The possible amplitude of movements was significantly lower for the X axis ($\sim$30 cm) compared to Y and Z axes ($\sim$55 cm) which may require from the patient more precise movements closer to the target. Additionally, in the virtual environment, the hands of the avatar were displayed on a flat screen which may cause problems in the perception of the depth and may affect visual feedback. A combination of those two factors might influence the recordings. Thus it should be taken into account while designing the next experimental sessions.

4.3. Online training

REW-NPLS enables incremental training of multilinear models from newly recorded data chunks. This is particularly important in a closed-loop experiment since it enables co-adaptation between the patient and the models. Such kind of training may be hard to use in the case of DL. Backpropagation may fail to find a reasonable solution [90], especially when provided with small chunks (e.g. 150 samples corresponding to 15 s of signal in the case of REW-NPLS) that are likely to be biased towards an overrepresented direction. Such circumstances may lead to a drastic decrease in DL model performance. To train DL models incrementally over the ECoG data stream, one may increase the data chunk size or keep the whole or part of the dataset in the memory and mix it with new data. Schwemmer *et al* [51] updated a pretrained model using a part of their training dataset combined with newly recorded data. One can also utilize more sophisticated methods like hedge backpropagation [90]. An alternative solution could be keeping all the data in memory and retraining from scratch each time a new data chunk is available. This solution is memory inefficient and may not be possible to perform in real-time as training from scratch may take more than 2 min (for CNN2D + LSTM + MT and around 40 min of signal). More experiments must be conducted to study the possibility of training DL models incrementally. Nevertheless, our study shows that a standard optimization of the DL-based models

from data recorded during multilinear model incremental training enables obtaining significantly better models.

4.4. 2D CNN

Methods processing data with 2D CNN obtained higher cosine similarity compared to solutions based on flattened feature vector. Input representation for 2D CNN enabled exploitation of local correlation between electrodes in contrast to MLP and multilinear model. 2D CNN model with several convolutional blocks also has significantly fewer trainable parameters than MLP. In 2D CNN architecture, the data processing is separated for each implant. We also considered a scenario in which the features from both implants would be concatenated along the width dimension as if there was no space between the inner edges of the implants. However, we decided to keep a separate data processing for each implant to avoid introducing an additional spatial distortion that would have broken the spatial consistency between neighboring electrodes. Then, the same set of weights was used for both implants. This enabled us to halve the number of trainable parameters (see table B1) and learn features that generalize across implants. Nevertheless, MI imagery patterns recorded by each ECoG implant are different since they are not positioned in the same cerebral hemisphere. From neuroscience, we also know that most of the brain activity correlated to unilateral hand MI/movements occurs in the contralateral brain hemisphere. However, brain activity correlated to this kind of movement can also be found in the ipsilateral motor cortex [89]. Due to the weight sharing property of CNNs, the network is oriented towards extracting low-level features that are implant invariant. On the other hand, extracting implant-specific features remains possible because each convolutional layer has multiple independent filters with parameters adjusted to the data.

4.5. Temporal information processing

We tested five methods to aggregate temporal information. CNN2D + LSTM + MT obtained the highest average cosine similarity. This confirmed that LSTMs could decode ECoG signals into complex hand trajectories. As the hidden state of the LSTM cell was the output of the network, the final predicted trajectory could be influenced by the input data from each time step. 3D CNNs that also analyze temporal information do not have this memory, so convolutional filters in the first layers are not aware of a longer temporal context than the length of the kernel. LSTM's memory can increase the network's ability to predict the desired trajectory as the target position is constant through the analyzed one second of the signal. Therefore, using several even similar target vectors to train the network may create a more robust and precise final estimation. However, the improvement compared to the CNN2D + FC model was not statistically significant, so one can also use this model, which has fewer parameters and performs similarly.

4.6. Multiple trajectories

CNN2D + LSTM + MT model achieved significantly higher cosine similarity than the CNN2D + LSTM. In CNN2D + LSTM models, the LSTM state encodes previous trajectories. It can memorize past hand and target positions and then modify the memory based on the next steps' data. Finally, the state of the LSTM is a summary of past desired trajectories. Providing LSTM with additional information about earlier desired trajectories enables a more accurate representation of the system state with a broader context. LSTMs at each time step decide what to memorize and what to forget. This can be especially useful in the case of repeated imagined MI patterns and varying patient concentration levels, resulting in temporal changes in the level of information contained in the data. Hence, we anticipate that the attention mechanism may be a reasonable way to extend the CNN2D + LSTM + MT model. Attention modules can highlight parts of the input containing the most relevant information for the prediction. Therefore, we expect that models incorporating more advanced attention can further improve hand movement decoding. Another way to extend the LSTM context may be increasing input signal length and taking into account long temporal relationships that occur inside one trial (average trial length $\approx$25 s). Our results show that including even a one-second-long time series of desired trajectory variations improves decoding accuracy. Having access to previous output variables gives more awareness, to the ML models, about the processes taking place in the experiment. Thus, extending the length of the input signal combined with the attention mechanism could improve overall BCI performance.

4.7. Design choices

We searched for the design choices that had the strongest influence on the accuracy of the DL-based models. Our results show that the most impactful component of the CNN architecture is the dropout layer. It enabled us to create a significant difference between the accuracy obtained by the MLP and the CNN2D + LSTM + MT. Our dataset has a few samples compared to the number of trainable parameters, so it is especially prone to overfitting. Dropout can be a remedy for overfitting as it is a regularization method [70]. Surprisingly, the CNN2D + LSTM + MT model achieved a high performance even when the dropout value was 0.65. It corresponds to a strong regularization with more than half of the network switched off. This value is different from the one suggested for computer vision. Cai *et al* [91] reported a decrease in accuracy for channel-wise dropout rates higher than 0.1. Higher dropout levels

limit model capacity, distort information stronger, and impose representation to be shared between various neurons. We hypothesize that the difference in optimal dropout rate between computer vision and ECoG signals originates from the difference in signal to noise ratio but may also be influenced by the network's size. A large network regarding the complexity of the problem can favor stronger overfitting. Therefore, we suggest that it may be possible to reduce the number of convolutional filters and the dropout rate conjointly. This would decrease the number of model parameters and reduce training and inference time.

The other design choices had a much smaller influence on the performance. We can notice a slight but general downward trend in performance with these design choices for almost all tested architectures. Our results concluded that ReLU is a better choice than ELU for our problem. This is in contradiction to the conclusions of Schirrmeister *et al* [73], who reported a higher decoding accuracy using ELU. However, Schirrmeister *et al* [73] analyzed raw EEG signals, while our networks were trained from ECoG time-frequency features. Therefore, both analyses consider signals represented in different domains, which certainly influences the choice of architecture.

Regarding pooling method selection, no particular trend was observed when the no-padding option was replaced by max pooling. One of our arguments for the use of no-padding is the specific arrangement of ECoG electrodes. The 64 ECoG electrodes are placed on two 4×4 cm grids that record neural signals from a small and specific area of the brain. The spatial resolution of the recording is higher than in the case of EEG by orders of magnitude. We see an opportunity for ML models to take advantage of this fact. However, max pooling may prevent the detection of small signal variations and make models invariant to small translations. Those may be important due to the nature of the observed phenomenon and the fact that we try to decode precise 3D movements that can be coded in tiny signal variations. Moreover, ECoG implants are centered on the sensorimotor cortex responsible for hand movements. Then, the central features are expected to be the most informative. No-padding naturally directs the attention towards the center of the implants, as fewer convolution operations are performed on the edges of the implants.

In our study, we also analyzed the influence of model depth on cosine similarity. Using more than two convolutional layers can decrease the models' accuracy or give only a slight improvement, depending on the dataset. This result might seem counter-intuitive, as stacking more layers in the DL architectures increases the number of trainable parameters and their capacity for representation. Nevertheless, our input data corresponded to hand-crafted features, which might not enable the extraction of high-level features. Worse, increasing the depth of the networks might have resulted in difficulties in estimating the optimal value of the model parameters. This might explain why we do not need to use dozens of layers, like computer vision [75], for this application. Finally, our best proposed model CNN2D + LSTM + MT consists of four layers—two convolutional blocks and two LSTM layers. Our results are consistent with the vast majority of BCI studies that used DL for EEG [92], ECoG [19, 62–65] and applied less than ten layers inside the models.

In the experiment, we chose hand-crafted time-frequency features that may not be optimal since they are not optimized for this specific problem. Thus, other successful methods of feature extraction should be evaluated, e.g. LFC [22, 93]. Most importantly, end-to-end DL-based models can learn features specific to this problem, which could bring an even more significant improvement than building upon already existing solutions.

Data availability statement

The data generated and/or analysed during the current study are not publicly available for legal/ethical reasons but are available from the corresponding author on reasonable request.

Acknowledgments

The authors would like to thank Thomas Costecalde for designing and recording the dataset used in this study, Serpil Karakas, Félix Martel, and Alexandre Moly for designing and implementing the software platform used to perform the experiments, Alim Louis Benabid and Stephan Chabardes for directing the clinical trial and the Clinatec BCI project, Guillaume Charvet for providing advice and direction for the study, and finally and most importantly the patient for his strong involvement in this research.

Clinatec is a Laboratory of CEA-Leti at Grenoble and has statutory links with the University Hospital of Grenoble (CHUGA) and with University Grenoble Alpes (UGA). This study was funded by CEA (recurrent funding) and the French Ministry of Health (Grant PHRC-15-15-0124), Institut Carnot, Fonds de Dotation Clinatec. Matthieu Martin was supported by the cross-disciplinary program on Numerical Simulation of CEA. Maciej Śliwowski was supported by the CEA NUMERICS program, which has received funding from European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie Grant Agreement No. 800945. Fondation Philanthropique Edmond J Safra is a major founding institution of the Clinatec Edmond J Safra Biomedical Research Center.

Appendix A. Targets positions

In figure A1, we presented positions of targets that were used during calibration and testing sessions.

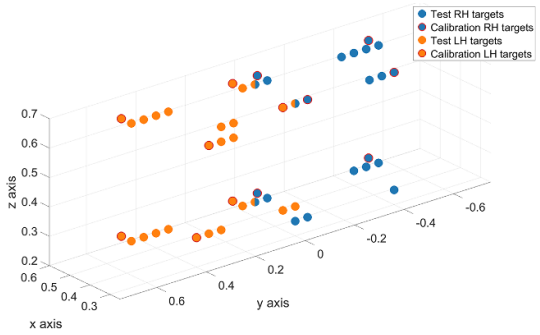


Figure A1. Positions of targets that patient was asked to reach during experiment.

Appendix B. Number of parameters

Table B1. Number of trainable parameters for evaluated methods.

	Parameters
REW-NPLS	28 800
CNN2D + FC	53 635
CNN3D + FC	86 851
CNN2D + LSTM + MT	238 772
CNN2D + LSTM	238 772
MLP	482 953
CNN2D + tConv/LSTM + FC	943 523

Appendix C. Hyperparameters values

Table C1. Values of hyperparameters selected using TPE.

Model	Learning rate	Weight decay	Batch size
MLP	0.00031	0.018	592
CNN3D + FC	0.00064	0.3	928
CNN2D + LSTM + MT	0.00023	0.24	96
CNN2D + LSTM	0.0089	0.12	352
CNN2D + FC	0.003	0.22	560
CNN2D + LSTM/ tConv + FC	0.00029	0.42	96

Table C2. Hyperparameters values used to estimate optimal number of layers.

Learning rate	Weight decay	Batch size
0.001	0.01	200

ORCID iDs

Maciej Śliwowski https://orcid.org/0000-0001-6744-1714
Matthieu Martin https://orcid.org/0000-0001-5954-8087

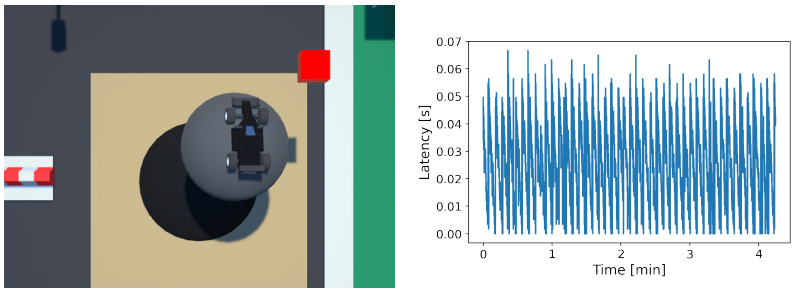
References

[1] Guger C, Kapeller C, Ortner R and Kamada K 2015 Motor imagery with brain-computer interface neurotechnology *Motor Imagery* (New York, NY: Nova Science Publishers, Inc) pp 61–80
[2] Benabid A L *et al* 2019 An exoskeleton controlled by an epidural wireless brain–machine interface in a tetraplegic patient: a proof-of-concept demonstration *Lancet Neurol.* **18** 1112–22
[3] Simpson L A, Eng J J, Hsieh J T and Wolfe D L (The Spinal Cord Injury Rehabilitation Evidence (SCIRE) Research Team) 2012 The health and life priorities of individuals with spinal cord injury: a systematic review *J. Neurotrauma* **29** 1548–55
[4] Anderson K D 2004 Targeting recovery: priorities of the spinal cord-injured population *J. Neurotrauma* **21** 1371–83
[5] Hochberg L R *et al* 2012 Reach and grasp by people with tetraplegia using a neurally controlled robotic arm *Nature* **485** 372–5
[6] Collinger J L *et al* 2013 High-performance neuroprosthetic control by an individual with tetraplegia *Lancet* **381** 557–64
[7] Wodlinger B, Downey J E, Tyler-Kabara E C, Schwartz A B, Boninger M L and Collinger J L 2015 Ten-dimensional antropomorphic arm control in a human brain-machine interface: difficulties, solutions and limitations *J. Neural Eng.* **12** 016011
[8] Nakanishi Y *et al* 2014 Decoding fingertip trajectory from electrocorticographic signals in humans *Neurosci. Res.* **85** 20–27
[9] Flint R D, Rosenow J M, Tate M C and Slutzky M W 2017 Continuous decoding of human grasp kinematics using epidural and subdural signals *J. Neural Eng.* **14** 016005
[10] Lawhern V J, Solon A J, Waytowich N R, Gordon S M, Hung C P and Lance B J 2018 EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces *J. Neural Eng.* **15** 056013
[11] Schwarz A, Ofner P, Pereira J, Sburlea A I and Müller-Putz G R 2018 Decoding natural reach-and-grasp actions from human EEG *J. Neural Eng.* **15** 016005
[12] Mondini V, Kobler R J, Sburlea A I and Müller-Putz G R 2020 Continuous low-frequency EEG decoding of arm movement for closed-loop, natural control of a robotic arm *J. Neural Eng.* **17** 046031
[13] Korik A, Sosnik R, Siddique N and Coyle D 2018 Decoding Imagined 3D hand movement trajectories from EEG: evidence to support the use of mu, beta and low gamma oscillations *Front. Neurosci.* **12** 130
[14] Larzabal C, Bonnet S, Costecalde T, Auboiroux V, Charvet G, Chabardes S, Aksenova T and Sauter-Starace F 2021 Long-term stability of the chronic epidural wireless recorder WIMAGINE in tetraplegic patients *J. Neural Eng.* **18** 056026
[15] Schalk G *et al* 2007 Decoding two-dimensional movement trajectories using electrocorticographic signals in humans *J. Neural Eng.* **4** 264–75
[16] Chin C M, Popovic M R, Thrasher A, Cameron T, Lozano A and Chen R 2007 Identification of arm movements using correlation of electrocorticographic spectral components and kinematic recordings *J. Neural Eng.* **4** 146–58
[17] Sanchez J C, Gunduz A, Carney P R and Principe J C 2008 Extraction and localization of mesoscopic motor control signals for human ECoG neuroprosthetics *J. Neurosci. Methods* **167** 63–81
[18] Liang N and Bougrain L 2012 Decoding finger flexion from band-specific ECoG signals in humans *Front. Neurosci.* **6** 91
[19] Du A, Yang S, Liu W and Huang H 2018 Decoding ECoG signal with deep learning model based on LSTM *TENCON 2018—2018 IEEE Region 10th Conf. Jeju, Korea (South)* (IEEE) pp 430–5
[20] Pistohl T, Ball T, Schulze-Bonhage A, Aertsen A and Mehring C 2008 Prediction of arm movement trajectories from ECoG-recordings in humans *J. Neurosci. Methods* **167** 105–14

[21] Acharya S, Fifer M S, Benz H L, Crone N E and Thakor N V 2010 Electrocorticographic amplitude predicts finger positions during slow grasping motions of the hand *J. Neural Eng.* **7** 046002
[22] Hammer J *et al* 2016 Predominance of movement speed over direction in neuronal population signals of motor cortex: intracranial EEG data and a simple explanatory model *Cereb. Cortex* **26** 2863–81
[23] Chauhan N K and Singh K 2018 A review on conventional machine learning vs deep learning *2018 Int. Conf. on Computing, Power and Communication Technologies (GUCON) Greater Noida, India* (IEEE) pp 347–52
[24] Ajiboye A B *et al* 2017 Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration *Lancet* **389** 1821–30
[25] Pistohl T, Schulze-Bonhage A, Aertsen A, Mehring C and Ball T 2012 Decoding natural grasp types from human ECoG *NeuroImage* **59** 248–60
[26] Milekovic T *et al* 2012 An online brain–machine interface using decoding of movement direction from the human electrocorticogram *J. Neural Eng.* **9** 046003
[27] Gunduz A, Brunner P, Sharma M, Leuthardt E C, Ritaccio A L, Pesaran B and Schalk G 2016 Differential roles of high gamma and local motor potentials for movement preparation and execution *Brain-Comput. Interfaces* **3** 88–102
[28] Kapeller C, Schneider C, Kamada K, Ogawa H, Kunii N, Ortner R, Pruckl R and Guger C 2014 Single trial detection of hand poses in human ECoG using CSP based feature extraction *2014 36th Annual Int. Conf. IEEE Engineering in Medicine and Biology Society Chicago, IL* (IEEE) pp 4599–602
[29] Li Y, Zhang S, Jin Y, Cai B, Controzzi M, Zhu J, Zhang J and Zheng X 2017 Gesture decoding using ECoG signals from human sensorimotor cortex: a pilot study *Behav. Neurol.* **2017** 3435686
[30] Yanagisawa T *et al* 2012 Electrocorticographic control of a prosthetic arm in paralyzed patients *Ann. Neurol.* **71** 353–61
[31] Wang W *et al* 2009 Human motor cortical activity recorded with micro-ECoG electrodes, during individual finger movements *2009 Annual Int. Conf. IEEE Engineering in Medicine and Biology Society Minneapolis, MN* (IEEE) pp 586–9
[32] Liu Y, Sharma M, Gaona C, Breshears J, Roland J, Freudenburg Z, Leuthardt E and Weinberger K Q 2010 Decoding ipsilateral finger movements from ECoG signals in humans *Advances in Neural Information Processing Systems* vol 23, ed J Lafferty, C Williams, J Shawe-Taylor, R Zemel and A Culotta (Curran Associates, Inc.) pp 1468–76
[33] Spüler M, Walter A, Ramos-Murguialday A, Naros G, Birbaumer N, Gharabaghi A, Rosenstiel W and Bogdan M 2014 Decoding of motor intentions from epidural ECoG recordings in severely paralyzed chronic stroke patients *J. Neural Eng.* **11** 066008
[34] Fangzhou X 2020 Decoding spectro-temporal representation for motor imagery recognition using ECoG-based brain-computer interfaces *J. Integr. Neurosci.* **19** 259
[35] Chestek C A, Gilja V, Blabe C H, Foster B L, Shenoy K V, Parvizi J and Henderson J M 2013 Hand posture classification using electrocorticography signals in the gamma band over human sensorimotor brain areas *J. Neural Eng.* **10** 026002
[36] Bleichner M G, Freudenburg Z V, Jansma J M, Aarnoutse E J, Vansteensel M J and Ramsey N F 2016 Give me a sign: decoding four complex hand gestures based on high-density ECoG *Brain Struct. Funct.* **221** 203–16
[37] Branco M P, Freudenburg Z V, Aarnoutse E J, Bleichner M G, Vansteensel M J and Ramsey N F 2017 Decoding hand gestures from primary somatosensory cortex using high-density ECoG *NeuroImage* **147** 130–42
[38] Nakanishi Y *et al* 2013 Prediction of three-dimensional arm trajectories based on ECoG signals recorded from human sensorimotor cortex *PLoS One* **8** e72085
[39] Nakanishi Y *et al* 2017 Mapping ECoG channel contributions to trajectory and muscle activity prediction in human sensorimotor cortex *Sci. Rep.* **7** 45486
[40] Kubánek J, Miller K J, Ojemann J G, Wolpaw J R and Schalk G 2009 Decoding flexion of individual fingers using electrocorticographic signals in humans *J. Neural Eng.* **6** 066001
[41] Flamary R and Rakotomamonjy A 2012 Decoding finger movements from ECoG signals using switching linear models *Front. Neurosci.* **6** 29
[42] Chen W, Liu X and Litt B 2014 Logistic-weighted regression improves decoding of finger flexion from electrocorticographic signals *36th Annual Int. Conf. IEEE Engineering in Medicine and Biology Society Chicago, IL* (IEEE) pp 2629–32
[43] Bundy D T, Pahwa M, Szrama N and Leuthardt E C 2016 Decoding three-dimensional reaching movements using electrocorticographic signals in humans *J. Neural Eng.* **13** 026021
[44] Silversmith D B, Abiri R, Hardy N F, Natraj N, Tu-Chan A, Chang E F and Ganguly K 2020 Plug-and-play control of a brain–computer interface through neural map stabilization *Nat. Biotechnol.* **39** 326–35
[45] Kellis S, Hanrahan S, Davis T, House P A, Brown R and Greger B 2012 Decoding hand trajectories from micro-electrocorticography in human patients *2012 Annual Int. Conf. IEEE Engineering in Medicine and Biology Society* (IEEE) pp 4091–4
[46] Krizhevsky A, Sutskever I and Hinton G E 2012 Imagenet classification with deep convolutional neural networks *Advances in Neural Information Processing Systems* **vol 25**, ed F Pereira, C J C Burges, L Bottou and K Q Weinberger (Curran Associates, Inc.) pp 1097–105
[47] Devlin J, Chang M-W, Lee K and Toutanova K 2019 BERT: pre-training of deep bidirectional transformers for language understanding *Proc. 2019 Conf. North American Chapter of the Association for Computational Linguistics: Human Language Technologies* vol 1 (Minneapolis, MN: Association for Computational Linguistics) pp 4171–86
[48] Willett F R, Avansino D T, Hochberg L R, Henderson J M and Shenoy K V 2021 High-performance brain-to-text communication via handwriting *Nature* **593** 249–54
[49] Pandarinath C *et al* 2018 Inferring single-trial neural population dynamics using sequential auto-encoders *Nat. Methods* **15** 805–15
[50] Golshan H M, Hebb A O and Mahoor M H 2020 LFP-Net: a deep learning framework to recognize human behavioral activities using brain STN-LFP signals *J. Neurosci. Methods* **335** 108621
[51] Schwemmer M A, Skomrock N D, Sederberg P B, Ting J E, Sharma G, Bockbrader M A and Friedenberg D A 2018 Meeting brain–computer interface user performance expectations using a deep neural network decoding framework *Nat. Med.* **24** 1669–76
[52] Roy S, Chowdhury A, McCreadie K and Prasad G 2020 Deep learning based inter-subject continuous decoding of motor imagery for practical brain-computer interfaces *Front. Neurosci.* **14** 918
[53] Tabar Y R and Halici U 2017 A novel deep learning approach for classification of EEG motor imagery signals *J. Neural Eng.* **14** 016003
[54] Tang Z, Li C and Sun S 2017 Single-trial EEG classification of motor imagery using deep convolutional neural networks *Optik* **130** 11–18
[55] Xu B *et al* 2019 Wavelet transform time-frequency image and convolutional network-based motor imagery EEG classification *IEEE Access* **7** 6084–93
[56] Nakagome S, Luu T P, He Y, Ravindran A S and Contreras-Vidal J L 2020 An empirical comparison of neural networks and machine learning algorithms for EEG gait decoding *Sci. Rep.* **10** 4372
[57] Ma X, Qiu S, Du C, Xing J and He H 2018 Improving EEG-based motor imagery classification via spatial and

temporal recurrent neural networks *2018 40th Annual Int. Conf. IEEE Engineering in Medicine and Biology Society (EMBC) (Honolulu, HI)* (IEEE) pp 1903–6
[58] Wang P, Jiang A, Liu X, Shang J and Zhang L 2018 LSTM-based EEG classification in motor imagery tasks *IEEE Trans. Neural Syst. Rehabil. Eng.* **26** 2086–95
[59] Bashivan P, Rish I, Yeasin M and Codella N 2016 Learning representations from EEG with deep recurrent-convolutional neural networks (arXiv:1511.06448 [cs])
[60] Zhang R, Zong Q, Dou L and Zhao X 2019 A novel hybrid deep learning scheme for four-class motor imagery classification *J. Neural Eng.* **16** 066004
[61] Garcia-Moreno F M, Bermudez-Edo M, Rodriguez-Fortiz M J and Garrido J L 2020 A CNN-LSTM deep learning classifier for motor imagery EEG detection using a low-invasive and low-cost BCI headband *2020 16th Int. Conf. on Intelligent Environments (IE) (Madrid, Spain)* (IEEE) pp 84–91
[62] Elango V, Patel A N, Miller K J and Gilja V 2017 Sequence transfer learning for neural decoding (accessed 01 September 2021)
[63] Pan G, Li J-J, Qi Y, Yu H, Zhu J-M, Zheng X-X, Wang Y-M and Zhang S-M 2018 Rapid decoding of hand gestures in electrocorticography using recurrent neural networks *Front. Neurosci.* **12** 555
[64] Rashid M, Islam M, Sulaiman N, Bari B S, Saha R K and Hasan M J 2020 Electrocorticography based motor imagery movements classification using long short-term memory (LSTM) based on deep learning approach *SN Appl. Sci.* **2** 211
[65] Xie Z, Schwartz O and Prasad A 2018 Decoding of finger trajectory from ECoG using deep learning *J. Neural Eng.* **15** 036009
[66] Eliseyev A, Auboiroux V, Costecalde T, Langar L, Charvet G, Mestais C, Aksenova T and Benabid A-L 2017 Recursive exponentially weighted N-way partial least squares regression with recursive-validation of hyper-parameters in brain-computer interface applications *Sci. Rep.* **7** 16281
[67] Moly A, Costecalde T, Martel F, Martin M, Larzabal C, Karakas S, Verney A, Charvet G, Benabid AL and Aksenova T 2022 An adaptive closed-loop ECoG decoder for long-term and stable bimanual control of an exoskeleton by a tetraplegic *J. Neural Eng.* (Accepted)
[68] Mestais C S, Charvet G, Sauter-Starace F, Foerster M, Ratel D and Benabid A L 2015 Wimagine: wireless 64-channel ECoG recording implant for long term clinical applications *IEEE Trans. Neural Syst. Rehabil. Eng.* **23** 10–21
[69] Ioffe S and Szegedy C 2015 Batch normalization: accelerating deep network training by reducing internal covariate shift *Proc. 32nd Int. Conf. on Machine Learning (Lille, France, 7–9 July 2015)* vol 37, ed F Bach and D Blei (PMLR) pp 448–56
[70] Srivastava N, Hinton G, Krizhevsky A, Sutskever I and Salakhutdinov R 2014 Dropout: a simple way to prevent neural networks from overfitting *J. Mach. Learn. Res.* **15** 1929–58
[71] Goodfellow I, Bengio Y and Courville A 2016 *Deep Learning* (Cambridge, MA: MIT Press)
[72] Simonyan K and Zisserman A 2015 Very deep convolutional networks for large-scale image recognition (arXiv:1409.1556 [cs])
[73] Schirrmeister R T *et al* 2017 Deep learning with convolutional neural networks for EEG decoding and visualization: convolutional neural networks in EEG analysis *Hum. Brain Mapp.* **38** 5391–420
[74] Chen G, Chen P, Shi Y, Hsieh C-Y, Liao B and Zhang S 2019 Rethinking the usage of batch normalization and dropout in the training of deep neural networks (arXiv:1905.05928 [cs, stat])
[75] He K, Zhang X, Ren S and Sun J 2015 Deep residual learning for image recognition (arXiv:1512.03385 [cs])
[76] Clevert D-A, Unterthiner T and Hochreiter S 2016 Fast and accurate deep network learning by exponential linear units (ELUS) (accessed 01 September 2021) (arXiv:1512.03385 [cs])
[77] Szegedy C, Vanhoucke V, Ioffe S, Shlens J and Wojna Z 2016 Rethinking the inception architecture for computer vision *2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (Las Vegas, NV, USA)* (IEEE) pp 2818–26
[78] Hochreiter S and Schmidhuber J 1997 Long short-term memory *Neural Comput.* **9** 1735–80
[79] Olah C 2015 Understanding LSTM networks
[80] Ji S, Xu W, Yang M and Yu K 2013 3D convolutional neural networks for human action recognition *IEEE Trans. Pattern Anal. Mach. Intell.* **35** 221–31
[81] Bergstra J S, Bardenet R, Bengio Y and Kégl B 2011 Algorithms for hyper-parameter optimization *Advances in Neural Information Processing Systems* **24**
[82] Loshchilov I and Hutter F 2017 SGDR: stochastic gradient descent with warm restarts *International Conf. on Learning Representations*
[83] Paszke A *et al* 2019 PyTorch: an imperative style, high-performance deep learning library *Advances in Neural Information Processing Systems* vol 32 (Curran Associates, Inc.) pp 8024–35
[84] Tietz M, Fan T J, Nouri D and Bossan B (skorch Developers) 2017 skorch: a scikit-learn compatible neural network library that wraps PyTorch (https://skorch.readthedocs.io/en/stable/)
[85] MATLAB 2017 MATLAB 9.3.0.713579 (R2017b) (Natick, MA: The MathWorks Inc.)
[86] Waskom M L 2021 seaborn: statistical data visualization *J. Open Source Softw.* **6** 3021
[87] Hunter J D 2007 Matplotlib: a 2D graphics environment *Comput. Sci. Eng.* **9** 90–95
[88] The pandas Development Team 2021 pandas-dev/pandas: pandas 1.2.1 *Zenodo* (available at: https://doi.org/10.5281/zenodo.4452601)
[89] Bundy D T, Szrama N, Pahwa M and Leuthardt E C 2018 Unilateral, 3D arm movement kinematics are encoded in ipsilateral human cortex *J. Neurosci.* **38** 10042–56
[90] Sahoo D, Pham Q, Lu J and Hoi S C H 2018 Online deep learning: learning deep neural networks on the fly *Proc. 27th Int. Conf. on Artificial Intelligence (IJCAI-18)* pp 2660–6
[91] Cai S, Shu Y, Chen G, Ooi B C, Wang W and Zhang M 2020 Effective and efficient dropout for deep convolutional neural networks (accessed 01 September 2021) (arXiv:1904.03392 [cs])
[92] Roy Y, Banville H, Albuquerque I, Gramfort A, Falk T H and Faubert J 2019 Deep learning-based electroencephalography analysis: a systematic review *J. Neural Eng.* **16** 051001
[93] Kobler R J, Almeida I, Sburlea A I and Müller-Putz G R 2020 Using machine learning to reveal the population vector from EEG signals *J. Neural Eng.* **17** 026002

Supplementary implementation test



(a) Screenshot from F1 game. The joystick is controlled with motor imagery commands.

(b) Latency of the system with DL model prediction for one expert.

Figure 2.1: Online DL model evaluation.

The goal of this study was to propose DL-based models that could be used in online experiments. As a part of the thesis, we prepared Python-package for analysis of data recorded in the clinical trial. Prepared framework was integrated into the Matlab-based real-time signal processing system.

To validate our implementation, DL-based model, i.e., CNN+LSTM+MT, was used during an experiment with another tetraplegic subject enrolled in the Clinatec 'BCI and Tetraplegia' clinical trial. The experimental paradigm considered for DL model evaluation was formula one (F1) virtual environment game in which patient controlled 2D joystick movements using hand motor imagery (see figure 2.1a). First, model was trained using database of prerecorded signals. Then, during an experiment, the model was used for approximately 4 minutes to provide 2D control over virtual effector to the the patient. Only CPU was used to compute predictions which provided enough computational power for our application. Thanks to that, we could remove GPU from computer configuration used for experiments.

Our online evaluation validates possibility to use DL models for real-life application in the clinical trial without additional latency (< 100 ms, see figure 2.1b). Our experiment focused only on the infrastructure integration and potential for real-life online application of DL-based models. The main reason why we could not analyze models prediction to estimate decoding performance was the level of control the patient achieved in the clinical trial procedure that did not allow for efficient multidimensional control at the moment of the experiment.

3

Impact of dataset size and long-term ECoG-based BCI usage on deep learning decoders performance

This chapter contains pre-print version of article submitted to Frontiers in Human Neuroscience **(accepted)**: Śliwowski, M., Martin, M., Souloumiac, A., Blanchart, P., and Aksenova, T. Impact of dataset size and long-term ECoG-based BCI usage on deep learning decoders performance. https://arxiv.org/abs/2209.03789

Impact of dataset size and long-term ECoG-based BCI usage on deep learning decoders performance

Maciej Śliwowski[1,2], Matthieu Martin[1], Antoine Souloumiac[2], Pierre Blanchart[2] and Tetiana Aksenova[1]

1 Univ. Grenoble Alpes, CEA, LETI, Clinatec, F-38000 Grenoble, France
2 Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France

E-mail: tetiana.aksenova@cea.fr

August 2022

Abstract. *Objective.* In brain-computer interfaces (BCI) research, recording data is time-consuming and expensive, which limits access to big datasets. This may influence the BCI system performance as machine learning methods depend strongly on the training dataset size. Important questions arise: taking into account neuronal signal characteristics (e.g., non-stationarity), can we achieve higher decoding performance with more data to train decoders? What is the perspective for further improvement with time in the case of long-term BCI studies? In this study, we investigated the impact of long-term recordings on motor imagery decoding from two main perspectives: model requirements regarding dataset size and potential for patient adaptation.

Approach. We evaluated the multilinear model and two deep learning (DL) models on a long-term clinical trial dataset containing 43 sessions of ECoG recordings performed with a tetraplegic patient. In the experiment, a participant executed 3D virtual hand translation using motor imagery patterns. We designed multiple computational experiments in which training datasets were increased or translated to investigate the relationship between models' performance and different factors influencing recordings.

Main results. For all tested decoders, our analysis showed that adding more data to the training dataset may not instantly increase performance for datasets already containing 40 minutes of the signal. DL decoders showed similar requirements regarding the dataset size compared to the multilinear model while demonstrating higher decoding performance. Moreover, high decoding performance was obtained with relatively small datasets recorded later in the experiment, suggesting motor imagery patterns improvement and patient adaptation during the long-term experiment. Finally, we proposed UMAP embeddings and local intrinsic dimensionality as a way to visualize the data and potentially evaluate data quality.

Significance. DL-based decoding is a prospective approach in BCI which may be efficiently applied with real-life dataset size. Patient–decoder co-adaptation is an important factor to consider in long-term clinical BCI.

Keywords: ECoG, motor imagery, deep learning, tetraplegia, adaptation, dataset size, learning curve

1. Introduction

Permanent motor deficits as a result of a spinal cord injury (SCI) affect hundreds of thousands of people worldwide each year (12,000 people each year just in the United States [1]). In this case, the motor cortex is preserved, but neuronal signals can no longer be transmitted to the muscles. Then, the use of a brain-computer interface (BCI), which enables interacting with an effector by thought, could enable these patients to regain a certain autonomy in everyday life. For example, motor imagery based BCI has been used for the control of prostheses or exoskeletons of upper limbs [2–4], lower limbs [5–8] and four limbs [9] in subjects with paraplegia or tetraplegia following an SCI. In this study, we focus on electrocorticography (ECoG)-based motor BCIs, promising tools that may enable continuous 3D hand trajectory decoding for neuroprosthesis control while reducing the risk of implantation compared to more invasive approaches [10].

BCIs record neuronal activity and decode it into control commands for effectors. Decoders are generally trained using machine learning algorithms in a supervised manner. In the vast majority of studies, the training dataset is strongly restricted due to limited access to recordings. At the same time, dataset size is an important factor in machine learning analysis and can influence overall system performance drastically. In contrast to recent computer vision and natural language processing studies [11–13], the optimal quantity of training data, i.e., the quantity at which decoder's performance reaches a plateau for a given application, is rarely studied for BCI [14]. Especially, learning curves, providing insight into the relationship between model performance and training set size, are rarely presented. Learning curves can be used for model selection, decreasing the computational load of model training, or estimating the theoretical influence of adding more data to training datasets [15]. The last point is particularly important in BCI, considering limited access to datasets recorded with humans. Without knowing the relationship between system performance and dataset size, it is hard to determine the strategy to improve the accuracy of decoders: increase the amount of training data or increase the capacity of the models. In the case of ECoG-based motor BCI, most models have a limited capacity. The decoders used are Kalman filters [16, 17] and mostly variants of linear models [18–24]. In most of these studies, decoder optimization has been carried out on databases containing a few minutes or tens of minutes of the signal. This results in usable models but does not provide any information on the performance gain that could be achieved with more data, nor does it compare the data quantity/performance relationship between several decoders.

Model characteristics and learning curves are not the only factors influencing decoders' performance in the case of BCI. The human ability to generate distinct brain signal patterns is crucial for a BCI system to work. Research in recent years has focused mainly on the development of increasingly efficient decoders, for example DL [25–38] rather than on patient learning or co-adaptation [39, 40], even though several studies have shown the crucial importance of patient learning [41–45]. Thanks to recording device developments and clinical trial advances, long-term studies of chronic BCI enable recording bigger datasets than ever before. Current techniques for recording brain activity, such as the ElectroCorticoGram (ECoG), provide stable recordings for at least 2 years [46]. It offers the possibility to train and test a decoder over several months. It also enables studying potential patient learning and provides insight into the optimal quantity of data necessary to get the best out of a decoder. These questions have largely been put aside [14].

Closed-loop learning allows for short-term patient-model co-adaptation through the visual feedback received by the patient. This feedback leads to a modification of the brain activity and has shown capabilities for improving the control of neuroprostheses [44, 47–50]. Nevertheless, motor learning is a process that takes place in the short term and in the long term [51, 52]. This long-term learning is little studied in BCI and most studies in humans are limited to a few sessions (< 15) [53–56] to show that a fast and efficient calibration of the proposed decoders is possible. Several studies with a larger number of sessions (> 20) were nevertheless carried out: [4, 8, 45, 57–63]. Some have focused on patient learning [8, 45, 56–58, 61] by seeking an improvement in performance coming from changes in the signal or the characteristics extracted from it. The last point is required to distinguish between performance improvement due to patient learning, increased data available for decoder optimization, or changes in the experimental environment [14, 64].

This study investigated the relationship between BCI decoders' performance predicting 3D upper-limb movements from ECoG signals and the training dataset size used to optimize model parameters. Learning curves obtained in different offline computational experiments showed that multilinear and DL models saturate at a similar amount of data, between 30 and 90 minutes of ECoG signal, depending on the scenario and hand. Moreover, learning curves revealed characteristics that were unlikely caused by just the dataset increase. Extended analysis using unsupervised ML methods showed dataset characteristic changes with time, suggesting that long-term patient learning may play an important role in achieving higher

BCI performance. This kind of analysis was possible thanks to the access to a rare database of ECoG signals [63] containing imagined hand movements performed by a tetraplegic patient to control upper-limb 3D translation in a virtual environment. The dataset contains 43 sessions recorded over 9 months (approximately 6 hours of data for each hand).

2. Methods

2.1. Clinical trial and patient

The data was recorded and analyzed as a part of the "BCI and Tetraplegia" (ClinicalTrials.gov identifier: NCT02550522) clinical trial, which was approved by the Agency for the Safety of Medicines and Health Products (Agence nationale de sécurité du médicament et des produits de santé—ANSM) with the registration number: 2015-A00650-49 and the ethical Committee for the Protection of Individuals (Comité de Protection des Personnes—CPP) with the registration number: 15-CHUG-19.

The participant was a 28-year-old right-handed man following tetraplegia after a C4-C5 spinal cord injury. He had residual control over upper limbs with American Spinal Injury Association Impairment (ASIA) scores of 4 (right hand), 5 (left hand) at the level of the elbow, and 0 (right hand), 3 (left hand) at the extensors of the wrist. All motor functions below were completely lost (ASIA score of 0). [65]

Two WIMAGINE implants [66], recording ECoG signal at 586 Hz sampling rate, were implanted above left and right primary motor and sensory cortex responsible for upper limb movements. The implants consisted of an 8×8 electrode's grid. Due to the data transfer limit, only 32 electrodes (organized on a chessboard-like grid) were used.

The data recordings used in this study started after 463 days post-implantation. The subject was already experienced in the BCI setup as the clinical trial experiments began shortly after the surgery. During the clinical trial, the participant gradually learned how to control the BCI, starting by using discrete/1D effectors and finally achieving control of up to 8D movements in one experimental session.

2.2. Data and experimental paradigm

The dataset analyzed in this study contains 43 experimental sessions in which tetraplegic patient was asked to perform motor imagery tasks in order to move virtual exoskeleton effectors (see the virtual environment in figure 1). In particular, the patient used an MI strategy in which he repeatedly imagined/attempted fingers, hands, and arm movements to control 8 dimensions (3D left and right hand translation and 1D left and right wrist rotation). In every trial, the patient's goal was to reach the target displayed on the screen, one after another, without returning to the center position. [63]

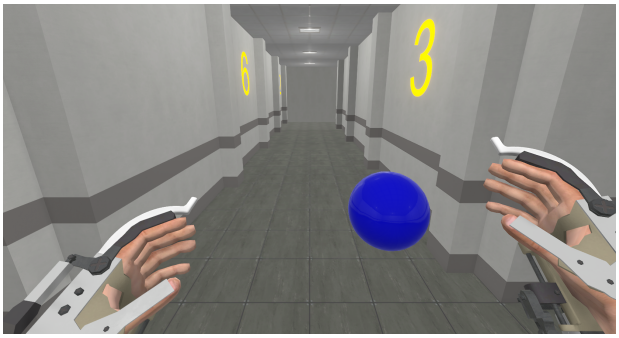


Figure 1. Screenshot from the virtual environment. The patient was asked to reach the blue sphere with his right hand.

Table 1. Datasets size in the number of trials and length of the recordings.

	Left hand	Right hand
Trials	811	756
Duration [min]	300	284

During the experimental sessions, 1 out of 5 states (idle state, left hand translation, right hand translation, left wrist rotation, right wrist rotation) was decoded from the recorded ECoG signal with a multilinear gate model. Accordingly to the gate predictions, an appropriate multilinear expert was selected to provide a trajectory of hand movement or direction of wrist rotation. For further analysis, we selected only left and right hand translation datasets.

Multilinear model parameters were optimized online during the recordings using recursive exponentially weighted n-way partial least squares (REW-NPLS) [67]. Models were trained on the first six sessions, further referred to as the calibration dataset. For the next 37 sessions, models' weights were fixed and used for the performance evaluation. In our computational experiments, we concatenated calibration and test sessions to perform offline simulations in different scenarios, studying the dataset and model characteristics in-depth. Datasets sizes are reported in table 1.

2.3. Preprocessing and feature extraction

Raw ECoG signal was processed with a feature extraction pipeline creating time-frequency representation. Continuous complex wavelet transform was used with 15 Morlet wavelets with central frequencies in the range of 10-150 Hz (10 Hz interval). Every 100 ms, one second of signal (90% overlap) was selected and convolved with the set of wavelets coefficients. Then modulus of

the convolved complex signal was averaged over 0.1s fragments. Finally, every i-th window of the signal was represented with time-frequency representation in the form of a tensor $\underline{\mathbf{X}}_i \in \mathbb{R}^{64\times15\times10}$ with dimensions corresponding to ECoG channels, frequency bands, and time steps.

In this study, samples for which predicted and desired states did not match each other were removed. By removing the gate errors, we minimize the influence of low gate model performance on the visual feedback and thus on the patient imagination patterns. In addition, one session was removed from the dataset as during the online experiment patient reached a highly negative cosine similarity (outliers compared to other sessions) which may as well influence recorded signals by providing erroneous visual feedback to the patient.

2.4. UMAP embeddings and artifacts identification

High-dimensional datasets are almost not possible to visualize without any dimensionality reduction before. What can be trivial to observe in low-dimensional space may easily stay hidden in the noise in high-dimensional representations. Due to the curse of dimensionality, understanding the topology of distributions or even noticing outliers is challenging. The main goal of the visualization was to see the evolution of data distributions between sessions. To map time-frequency representation into lower-dimensional space, an unsupervised learning algorithm, namely Uniform Manifold Approximation and Projection (UMAP) [68] was used. We decided to apply UMAP as it preserves the global manifold structure similarly to t-SNE [69] but has a lower computational load according to [68,70]. Thanks to that, we could avoid additional dimensionality reduction (e.g., PCA), which is usually done before feeding high-dimensional datasets into t-SNE [71]. We used flattened time-frequency features $\underline{\mathbf{X}}_i \in \mathbb{R}^{64\times15\times10} \rightarrow \mathbb{R}^{9600}$ (the same as for motor imagery decoding) as the input to UMAP. Every tenth observation from the dataset was selected for UMAP to avoid redundancy in the data (90% overlap between samples) and decrease the computational load. UMAP was fitted on three datasets to all the sessions together, i.e., one UMAP for both hands optimized together and one per hand trained individually. The first scenario lets us better see the data distributions within the state classification framework, with samples being colored due to the state they belong to. This gave us a global overview of the dataset. In the per hand scenario, we focused more locally on the structure of each dataset. This may have a bigger influence on the decoding performance while being harder to analyze due to the lack of explicit labels for visualization (like states in the previous case).

In the case of UMAP optimized together for both hands, we proposed an indirect indicator of data quality reflecting the separability of the left and right hand clusters. This was assessed using linear support-vector machine (SVM). SVM was fitted to every session separately. Then every sample in the session was classified into two categories, i.e., left hand or right hand movement. Accuracy of the state classification was further used as a state separability indicator. We did not perform any cross-validation as we focused on the separability of the clusters and not on the state classification performance itself. On the UMAP embeddings we visualized also SVM decision boundary dividing the space between categories of movements.

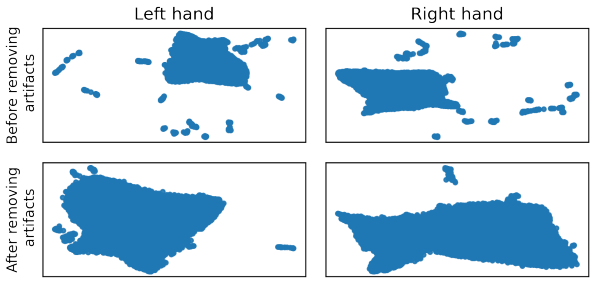


Figure 2. Per hand embeddings before (top row) and after (bottom row) artifacts removal.

UMAP as a dataset visualization method may also be used for an overall sanity check of the dataset, especially for artifacts that are easy to spot when the dataset is small, but it is impossible to review every sample individually when analyzing thousands of observations. In our case, UMAP helped us to observe artifacts coming from connection loss resulting in singular outliers samples that were not caught during recording. Those, on the UMAP plots, created suspicious clusters of observations (figure 2). The clusters of artifacts after recognition on the UMAP plots and further manual review were fixed by interpolation of points in the raw signal domain.

2.5. Evaluated models

Multilinear model optimized with REW-NPLS algorithm [67] was used as a 'traditional' ML benchmark to predict 3D hand translation. The same algorithm was also used for providing online control to the patient during recordings. PLS models embed both high-dimensional input features and output variables into lower-dimensional latent space, aiming to extract latent variables with the highest correlation between input and output. REW-NPLS model can be updated online thanks to low-computational cost, recursive validation of the number of latent factors, and model parameters being updated with only chunks of the dataset. Online training ease performing the experiments and makes it possible to use ECoG decoders almost from

Table 2. MLP architecture from [72].

Layer	Kernel Shape	Output Shape
Flatten	–	[200, 9600]
Fully connected	[9600, 50]	[200, 50]
BatchNorm	[50]	[200, 50]
ReLU	–	[200, 50]
Dropout	–	[200, 50]
Fully connected	[50, 50]	[200, 50]
BatchNorm	[50]	[200, 50]
ReLU	–	[200, 50]
Dropout	–	[200, 50]
Fully connected	[50, 3]	[200, 3]

Table 3. CNN+LSTM+MT architecture from [72].

Layer	Kernel Shape	Output Shape
Input		[200, 15, 8, 8, 10]
Input per implant		[200, 15, 8, 4, 10]
Conv space	[15, 32, 3, 3, 1]	[200, 32, 6, 4, 10]
ReLU	–	[200, 32, 6, 4, 10]
BatchNorm	[32]	[200, 32, 6, 4, 10]
Dropout	–	[200, 32, 6, 4, 10]
Conv space	[32, 64, 3, 3, 1]	[200, 64, 4, 2, 10]
ReLU	–	[200, 64, 4, 2, 10]
Dropout	–	[200, 64, 4, 2, 10]
LSTM	–	[200, 10, 50]
LSTM	–	[200, 10, 3]

the beginning of the first recording session. Even if decoders may show unstable performance at the beginning of the experiment due to the small amount of data used for training, it provides visual feedback to the patient. For our offline computational experiments, multilinear models were trained in pseudo-online mode, simulating real-life experiments with updates based on 15 seconds-long chunks of data.

The second group of models used deep learning to predict the desired hand translation. In particular, methods proposed and described in detail in [72] were evaluated—i.e., multilayer perceptron (MLP—simple approach) and mix of CNN and LSTM (CNN+LSTM+MT) providing the best performance for a given dataset [72]. MLP was built from two fully-connected layers with 50 neurons with dropout and batch normalization in-between (see table 2). CNN-based method exploited the spatial correlation between electrodes by analyzing data organized on a grid reflecting the electrodes' arrangement with convolutional layers. As the CNN-based method utilizes data structure, it has fewer parameters while maintaining similar capabilities to MLP. In CNN+LSTM+MT, LSTMs were used to aggregate temporal information extracted by convolutional layers into desired translation trajectory (see table 3). The DL models were trained to maximize cosine similarity (CS) between predicted and optimal trajectories. We used early stopping to limit the overfitting with a validation dataset consisting of the last 10% of the calibration dataset. The best model on the validation dataset was used for further evaluations. The procedure was repeated five times for every scenario and model to limit the influence of the stochasticity of the training process on our results. To train DL models, we used a fixed set of hyperparameters, i.e., learning rate equals 0.001, weight decay (L2 regularization) equals 0.01, and batch size equals 200.

2.6. Computational experiments

Multiple offline computational experiments were performed on the prerecorded ECoG BCI dataset to assess the impact of training dataset size on decoding performance. The results computed on a real-life dataset may be impacted by multiple factors that cannot be observed directly. Thus, we proposed several ways of increasing the training dataset as well as iterating over it. By modifying the training datasets in different manners, we aimed to isolate different factors that can potentially influence learning curves. In every scenario, all the models were trained on a different subset of the database and then evaluated on test datasets accordingly to the experiment.

2.6.1. Forward increase The forward increase (FI) scenario measured the change of cosine similarity when adding more recording sessions to the dataset. This experiment corresponds to a real-life situation where more data is collected during the experiment. The sessions were incrementally added (session by session) to the training dataset. After every step, all the decoders were trained from scratch and evaluated on the following 22 sessions (see Figure 3).

2.6.2. Backward increase An important factor influencing model training may be the nonstationarity of signal in time originating from the plasticity of the brain as well as the patient's adaptation. To assess the influence of this factor, an inverse of forward increase was performed, further referred to as backward increase (BI). Similar to the FI simulation, the training dataset was increased session by session. However, the increase was started from the 21st session and the previous sessions were added until including the first calibration session. After every training, models were evaluated on a fixed test set consisting of 22 last recordings (see Figure 3).

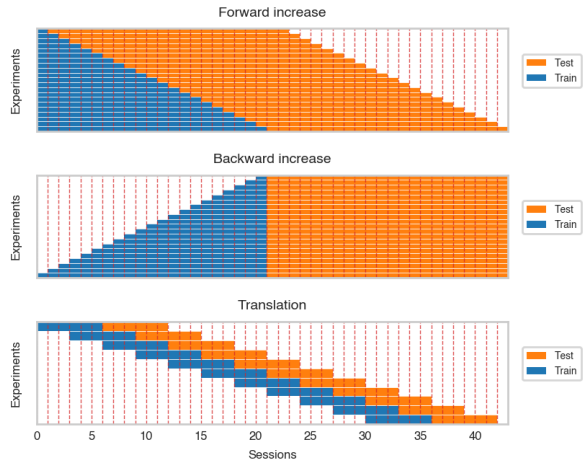


Figure 3. Visualization of forward and backward increase and translation over the dataset. For clarity, we ignored differences in session length.

2.6.3. Random increase An alternative way of assessing the influence of training dataset size on the decoder performance is random dataset increase (RI). Instead of maintaining the temporal order of recorded samples, we artificially removed the connection between neighboring observations, i.e., for every dataset size, a respective number of observations was selected from the first 22 sessions, and then the model was trained. This may reduce the effects of neuronal signal nonstationarity and/or patient adaptation and provide results closer to theoretical learning curves when assumptions about the stationarity of observations are fulfilled. Evaluations were performed on the same test set as in BI.

2.6.4. Dataset translation As data may change over time, we trained models on approximately the same amount of data but recorded in different periods of the experiment. This enabled us to rule out the effect of the increased dataset and focus on data shift and potential patient adaptation that may modify the data representation and influence the performance of trained decoders. The training dataset was translated over the whole dataset and evaluated on the test dataset consisting of the following six sessions (see figure 3).

2.7. Learning curve

The learning curve ‡ describes the relationship between model performance and the training dataset size l [73]. It can be used, for example, to infer a potential change in the performance from adding more data to the system. This can be particularly efficient in application to BCI because we can estimate the hypothetical performance of decoders when recording more data without actually performing the experiments. Learning curves can also be used to select an appropriate model for a specific dataset size. For example, Strang et al. [74] showed that non-linear models are more likely to outperform linear models for bigger datasets. On the other hand, Hoiem et al. [12] showed that models with more parameters can be more efficient in the case of small datasets despite the higher potential for overfitting.

The learning curve may be formulated with power law [73,75]. In our case, the relationship between cosine similarity and training dataset size may be expressed as:

$$\mathrm{CS}(l; a, b, c) = a - b \cdot l^{-c} \tag{1}$$

where b and c can be interpreted as learning rate and decay rate, respectively. a corresponds to theoretical asymptotic performance when $l \to \infty$. Parameters a, b, and c were fitted to the results obtained in RI experiment with non-linear least squares using Trust Region Reflective algorithm with bounds $a \in [-1, 1]$, $b > 0$, and $c > 0$.

2.8. Intrinsic dimensionality estimation

The idea of patient adaptation and improving BCI skills using visual feedback is based on the assumption that the patient can modify/adjust motor imagery patterns to solve the task better. As a result, the data distribution and the shape of the data manifold may change. To estimate the data distribution changes, intrinsic dimensionality (ID) estimation methods may be used. ID reflects the minimum number of variables needed to represent the dataset without a significant information loss. Thus, the ID indicator is strictly connected to a dataset's true dimensionality, which is an important factor in data analysis, influencing the performance and changing the number of samples needed to train models. Intuitively, in a typical case, higher-dimensional manifolds are harder to learn due to the 'curse of dimensionality.' ID is better studied for images that, although have thousands of pixels, lie on a lower-dimensional manifold (e.g., less than 50 for ImageNet [76]). We use ID as a potential data quality indicator, which may vary with different recording sessions. ID estimates were computed for every session, and values from the respective sessions were averaged to obtain training dataset estimates for the dataset translation experiment. To compute ID, we used current state-of-the-art methods, namely expected simplex skewness (ESS) [77] estimating local ID in data neighborhoods (in our case 100 points) and TwoNN

‡ In this context, the learning curve does not refer to the relationship between the number of training epochs and model performance which the name learning curve is also commonly used for.

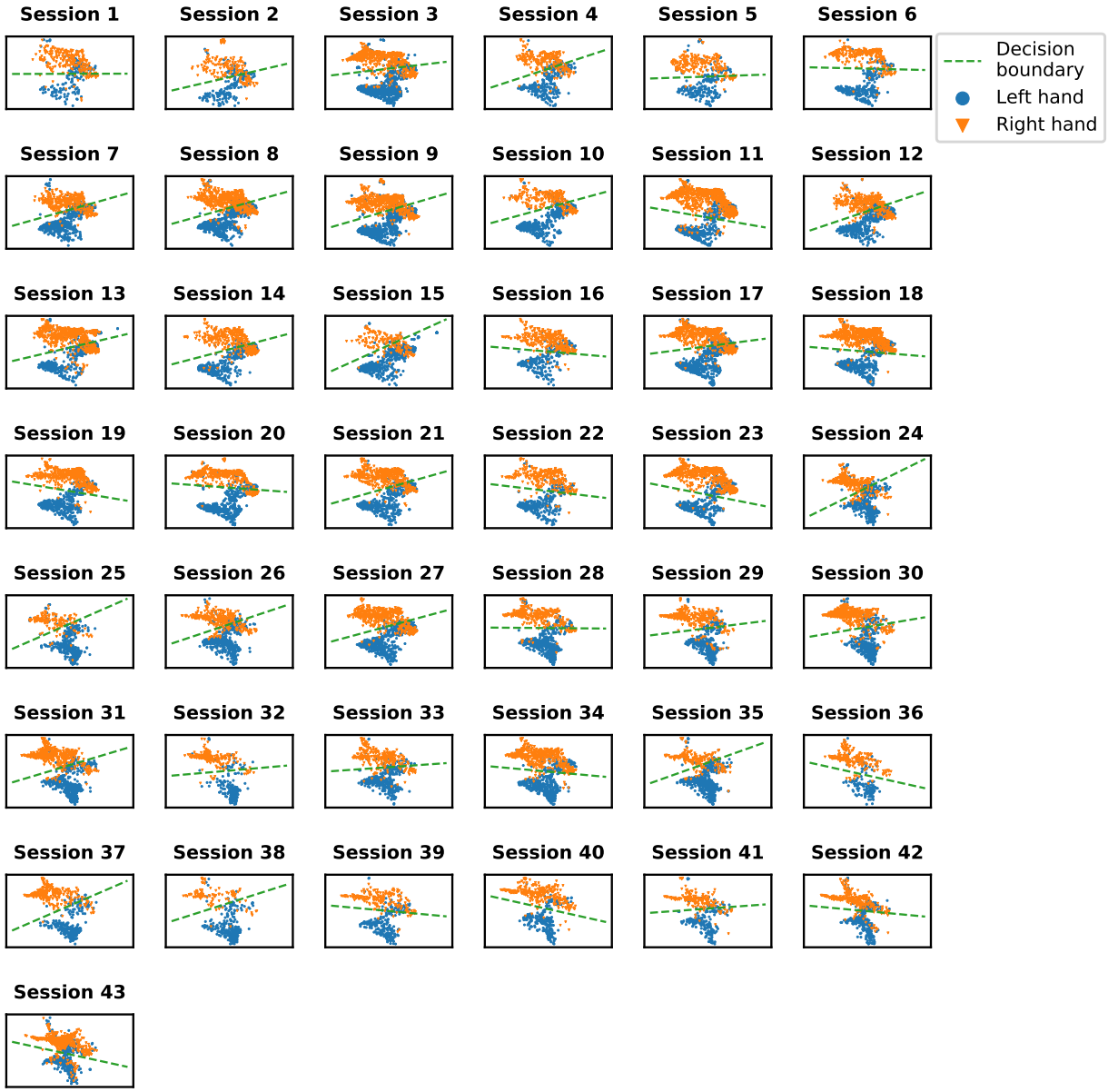


Figure 4. Visualization of 2D embedding of left and right hand data obtained using UMAP. Green dashed line shows SVM decision boundary.

[78] estimating global dataset ID. ESS, according to [79] provides better estimates for high ID values, while most of the methods tend to underestimate the ID (e.g., TwoNN [78]). It is especially important because our preliminary analysis showed that ECoG data is high dimensional, with ECoG features' mean local ID being significantly higher than the mean local ID for images (around 300 for ECoG, below 15 for MNIST, EMNIST, FMNIST [80]). For ID computations we used scikit-dimensions package [81].

3. Results

3.1. UMAP

Data distributions for every session were shown in figure 4 with colors indicating left and right hand states. With time, clusters of states get better separated from each other. We quantified separability of different states with SVM classification accuracy (figure 5). An increase in accuracy can be observed for sessions recorded later in the experiment, with

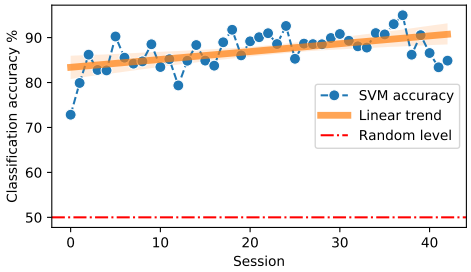


Figure 5. Accuracy of left vs. right hand state classification using SVM classifier. The orange line indicates a linear trend fitted to the points.

a maximum accuracy of 95% for session 37. Note that UMAP, similarly to t-SNE, does not preserve the density of points when mapping to the lower dimensional space and may, in some cases, create sub-clusters that originally may not exist in the input space.

3.2. Forward and backward increase

Forward increase results (figure 6) show learning curves in a situation close to a real-life scenario when more recordings are performed in the experiment. For all the models, a sharp increase in performance can be observed for small datasets. After 30-40 minutes of data, the curves become flat, reaching 70-80% of maximum FI performance (except 100% for the multilinear right-hand model) until 100-120 minutes of the signal. For datasets with more data than 100-120 minutes, a slow performance increase can be noticed. In the case of the left hand dataset, it starts earlier and is also visible for the multilinear model, while for the right hand, REW-NPLS performance stays stable. Overall, multilinear and DL models have similar learning curves and reach a performance plateau after including the same amount of data. However, multilinear models usually perform worse than DL models for the same amount of data.

Extending the dataset backward, starting from the middle of the recorded dataset, does not correspond to any real-life scenario. However, by doing this, we were able to assess the potential influence of data quality change on the results computed in the FI computational experiment. In the case of backward increase (figure 7), high performance can be observed for relatively small datasets—with just 3 (left hand) and 2 (right hand) sessions. For bigger datasets, the performance stabilizes or slightly decreases. The curves for all the models behave similarly. Performance of DL models starts to increase for > 130 minutes of signal for the right hand and achieves the best cosine similarity. When comparing FI and BI, in the case of the left hand, the best performance can be observed for BI and only 3 sessions of data in the training dataset. In the case of the right hand, the highest performance is achieved for the biggest dataset, suggesting that recording more data may improve the cosine similarity. The small amount of data needed to achieve high performance (2-3 sessions) in the BI experiment may suggest neuronal patterns improvement resulting in dataset quality increase (the amount of data required to reach a given performance).

3.3. Random increase

In the RI experiment, the influence of patient adaptation and signal nonstationarity is reduced as all the links between neighboring samples are destroyed when selecting data for the training dataset. Results for RI are more similar to theoretical learning curves of DL models, with a sharp increase in performance in the beginning and saturation when the model's maximum capacity is achieved. The performance is saturated after adding approximately 60-90 minutes of data to the training dataset at 95% of maximum cosine similarity for RI experiment. Only a small improvement can be observed from using more data. For the multilinear model, we can observe that saturated best performance is lower than in the case of DL models. DL methods are able to learn more complex functions and thus can reach higher performance. Fitted learning curves show the relationship between cosine similarity and dataset size within a theoretical framework, emphasizing the bigger capabilities of DL methods. The best models trained in the RI experiment showed lower performance compared to the best models from other experiments (dataset translation for both hands and BI for the left hand). However, in every experiment except BI and RI models were evaluated on different test datasets (see figure 3).

3.4. Dataset translation

The Dataset translation experiment shows the change in performance while maintaining approximately the same amount of data (six sessions) in the training dataset. Generally, all models show similar trends. For the left hand, we can observe an increase in cosine similarity for datasets recorded later in the experiment suggesting an improvement in data quality. The increase is less visible for the right hand dataset. This is confirmed by the slope of the linear trend fitted to the average performance of all the models (table 5). Expected cosine similarity improvement from training a model on the dataset recorded later was equal to 0.0069 per session and 0.0044 per session for left and right hand datasets, respectively. For both

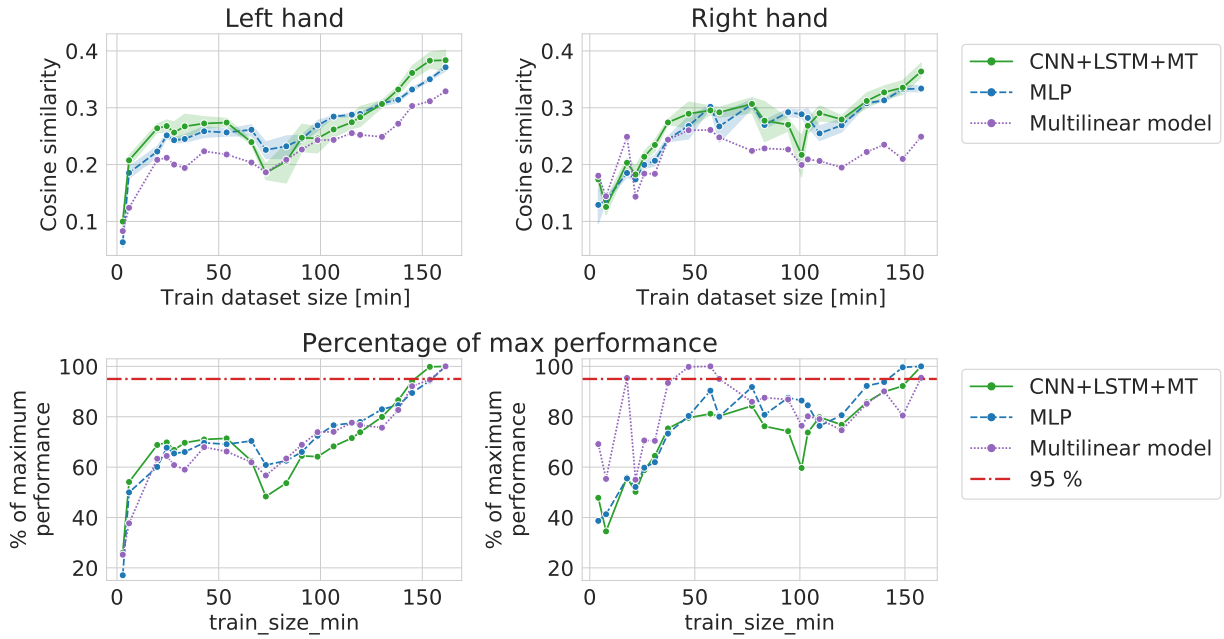


Figure 6. Cosine similarity computed in forward increase experiment, i.e., different training dataset sizes when starting from the first session.

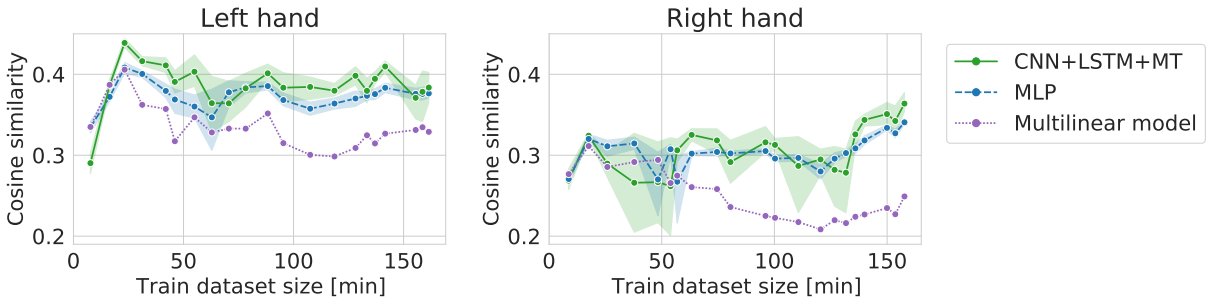


Figure 7. Cosine similarity for backward increase experiment, i.e., different training dataset sizes when starting from the 21st session and going backward.

datasets, the most significant performance increase between the first and last evaluation can be observed for the multilinear model (table 4). It may suggest that patient, to some extent, adapted specifically to the linear model family. The multilinear model does not follow the same fluctuations as the DL methods. The difference could be caused by the way of validating models (10% validation set for DL, recursive validation on last 15 seconds of data at every step for pseudo-online REW-NPLS).

In figure 10, the relationship between the local ID of the training dataset computed with ESS and the cosine similarity of different models for the translation experiment is presented. A statistically significant ($\alpha < 0.05$) correlation between local ID and models'

Table 4. Differences between models trained on sessions 0-6 and 30-36 in the dataset translation experiment.

	Left hand	Right hand
CNN+LSTM+MT	0.203	0.157
MLP	0.175	0.167
Multilinear model	0.274	0.239

Table 5. Parameters of trend lines fitted to the dataset translation results. Statistically significant p-values for the correlation coefficient are marked with asterisks.

	Slope	Intercept	R	p-value
Left hand	0.0069	0.2612	0.8816	0.0003*
Right hand	0.0044	0.2744	0.6999	0.0165*

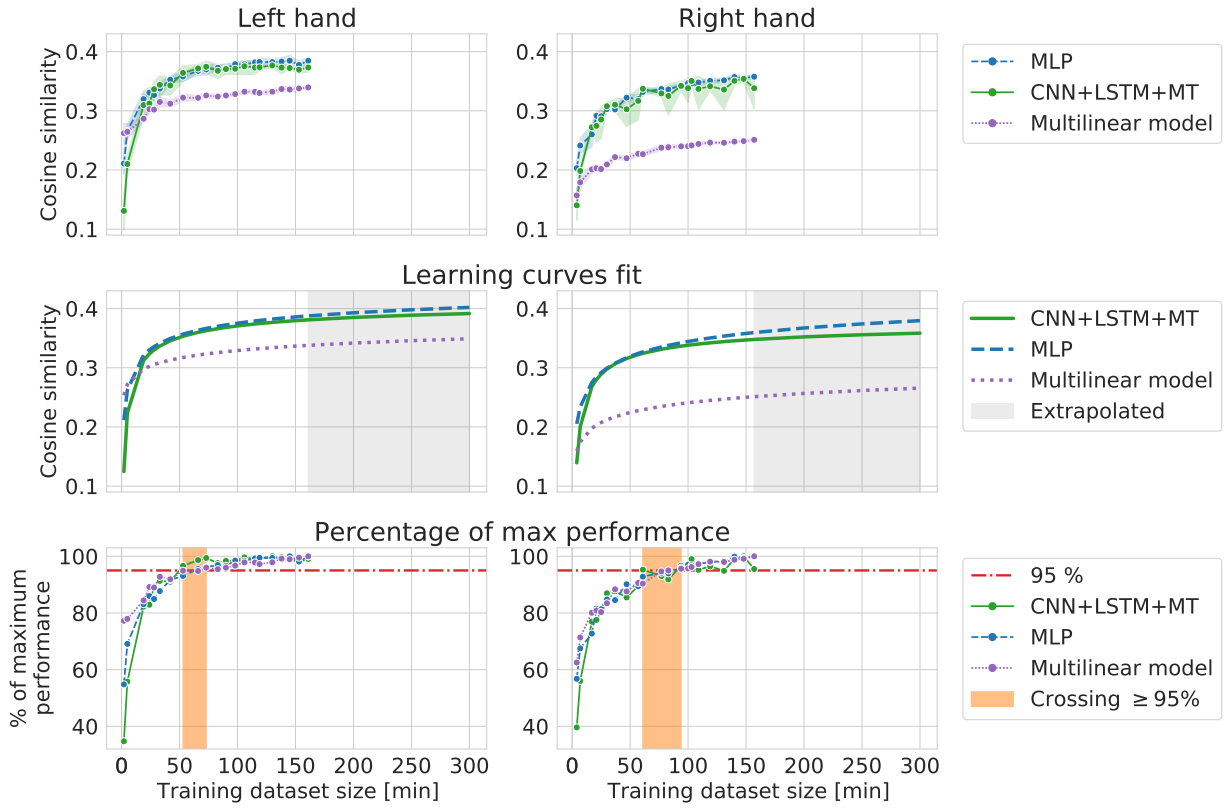


Figure 8. Cosine similarity for random increase experiment, i.e., different training dataset sizes when randomly selecting a subset of observations from the first 22 sessions. Every evaluation was performed 10 times.

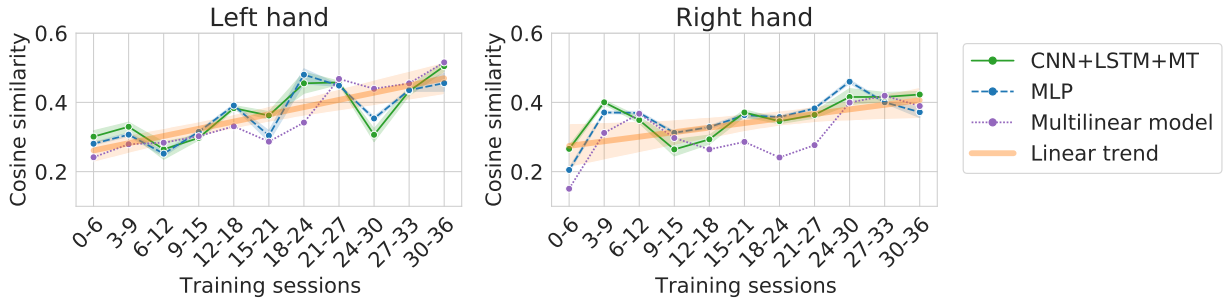


Figure 9. Cosine similarity for dataset translation, i.e., different training datasets (always 6 sessions for training and following 6 sessions for testing) translated over the dataset. The orange line indicates a linear trend line fitted to the models' average.

performance was observed for all the methods, reaching up to 0.66 of the r correlation coefficient for the multilinear model. An overall trend of achieving higher cosine similarity can be observed for training datasets with a higher ID. ID for the analyzed datasets varies between 250 and 330, which is much more compared to less than 15 reported for MNIST, EMNIST, FMNIST [80].

4. Discussion

Our results showed that including more data in the training dataset for ECoG BCI may not be immediately visible on the performance metrics if already having access to 40 minutes of the signal. Indeed, a drastic increase in performance can be noticed for datasets smaller than 40 minutes. This justifies the current experimental paradigms in which 40-50 minutes of the signal is collected (corresponding

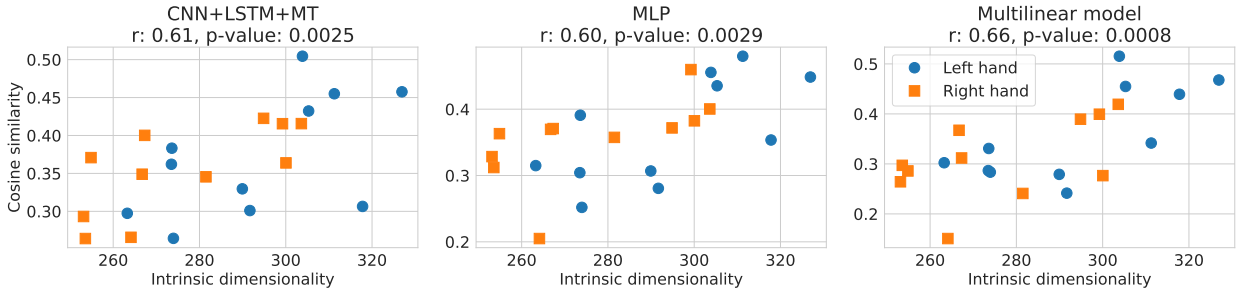


Figure 10. Relationship between cosine similarity and local ID of the training dataset computed with ESS for dataset translation experiment. In the plot titles, Pearson correlation coefficient r and p-value (the probability of two uncorrelated inputs obtaining r at least as extreme as obtained in this case) are presented.

to achieving approximately 70-80% of maximum performance achieved with datasets up to 160 minutes of data) for training 3D hand translation models.

Theoretically, models with bigger capacity can benefit stronger from having access to more data. One of the indicators of model capacity can be the number of trainable parameters. In our case, MLP had the biggest number of trainable parameters (482 953), followed by CNN+LSTM+MT (238 772) and the multilinear model (28 800). The difference in potential performance gains can be visible in figure 8. For small datasets, a multilinear model outperforms DL-based approaches (left hand) or provides approximately the same cosine similarity. However, the multilinear model saturates at a lower level of cosine similarity, resulting in a performance gap which could be explained by the difference in model capacity. Multilinear models are more likely to provide high performance compared to DL for small datasets, which is consistent with the ML theory of less complex functions being less prone to overfitting. RI results and fitted theoretical learning curves revealed models' characteristics while limiting the influence of other factors like distribution shifts or patient adaptation on the decoding performance. Finally, all models saturate for relatively small training datasets (50-90 minutes for RI, 50 minutes for FI, 30 minutes for BI) with only slight improvement from adding more data ($\sim$ 5%). This amount of data is similar to the usual amount of data used in BCI studies.

While this result validates previously developed data processing and experimental pipelines, a question arises whether it is an actual property/characteristic of brain signals or the shape of the curve is influenced by the previous years of research in which a relatively small amount of data was usually used to develop pipelines. There are hundreds of hyperparameters influencing data processing characteristics, starting from recording devices (e.g., number of electrodes, mental task design), signal processing pipelines (e.g., electrodes montage, filtering, standardization), ending on hyperparameters of machine learning models of all kinds (e.g., models' capacity, regularization weight, the architecture of models). The lack of huge improvement from increasing the dataset may be caused just because we reached the level of decoding close to maximum due to a lack of information in the data needed for prediction. However, from another perspective, one can hypothesize that we observe an effect of researchers overfitting to the specific conditions observed so far.

4.1. Models optimization for big datasets

All offline experiments were performed with a fixed set of hyperparameters. At the same time, different dataset sizes may require a change in the hyperparameters. For example, regularization limits overfitting, which should be less severe when the training dataset is big. Similar logic applies to dropout, which limits overfitting but on the other hand, it decreases models' capacity by introducing redundancy in the network representation. In the BI experiment, we observed a decrease in performance when adding more data for the left hand dataset. Hypothetically, increasing models' capacity may solve this problem (assuming it is caused by adding samples from different distributions to the dataset) because models with bigger capacity might not have to 'choose' on which motor imagery patterns they should focus. However, hyperparameters search is time and resource-consuming, so performing hyperparameters search for every dataset size may not be reasonable. In the future, DL architectures with bigger capacities in terms of the number of layers, number of neurons, etc., should be evaluated.

Datasets can also be artificially extended by using data augmentation methods. A variety of beneficial data augmentation methods exist for brain signals, especially EEG [82], that might improve decoding accuracy for the 3D hand movement control. Hoiem et al. [12] showed, for computer vision datasets, that data augmentation may act as a multiplier of the number of examples used for training. In the light of recent

advancements in EEG data augmentation, i.e., class-wise automatic differentiable data augmentation [83], it can be interesting to investigate how the reported results generalize to ECoG signals and influence, presented here, learning curves.

4.2. Patient training

UMAP embeddings may reveal interesting data manifold structures. In our case, we observed signs of distribution change on the embeddings visualization and the separability of left/right hand observations. Points start to be distributed denser in some regions of the plots and align along lines (see for examples sessions 35, 42, 43 for the left hand in figure A1 or sessions 31, 41, 42 in figure A2). Additionally, in the dataset translation experiment, we can see an increase in cosine similarity, stronger for the left hand. Moreover, the overall best performance for the left hand was achieved with only 3 sessions ($\sim$ 25 minutes of signal), outperforming models trained on much bigger datasets. This suggests improvements in patient BCI skills by adapting motor imagery patterns to the ML pipeline used in the study but non-specific to the multilinear model because trends are visible for all the evaluated approaches. At the same time, adding more data with noisy and changing patterns may not be profitable for the predictions. Thus, more focus should be placed on obtaining high-quality and well-separable motor imagery patterns in the signal. Patient adaptation is possible thanks to the visual feedback provided to the participant during recordings. The potential for patient adaptation creates a perspective for further improvements of BCI performance with the experience gained by the patient in long-term usage of the system. However, the reason adaptation is visible only for the left hand remains unknown. We hypothesize that the motor imagery patterns are easier to adapt for the left hand thanks to the remaining residual control resulting in better cortex preservation.

Our results showed a correlation between the local ID of the training dataset and the models' performance. This may indicate that models achieve better results when trained on more complicated manifolds. However, this hypothesis is counterintuitive and contradictory to research in computer vision. Thus, we hypothesize that higher ID may also indicate more diverse motor imagery patterns, better representing those found in the test set. Diversity of patterns may be harmful to models with a too-small capacity to learn them all. However, to some extent, it may be helpful as it creates a more diverse dataset that better reflect/cover the real manifold of all motor imagery patterns. Finally, we hypothesize that another hidden factor affects both the local ID and the amount of information needed for prediction, like the diversity of motor imagery patterns, so a change in local ID may not cause the increase in the performance itself. For example, local ID can also be increased by adding Gaussian noise to the signal, decreasing cosine similarity instead. Investigation of this kind of relationship is especially challenging in the case of brain signals due to a lack of data understanding with the 'naked eye,' which would significantly ease finding a correct interpretation of observed phenomena. As a next step, more ID estimation methods could be evaluated as statistically significant correlations for DL models were observed only for local ID computed with ESS. In the case of TwoNN, global ID did not show a significant correlation for DL approaches (see figure B1). This could be caused by worse TwoNN precision for high ID values as well as a lack of local per-sample ID information in the global ID dataset estimate. The relationship between local ID and performance should be further analyzed on different brain signal datasets.

4.3. Interpretation limitations

All the computational experiments analyzed in this study were obtained offline using data recorded with only one patient. Thus the learning curves and potential of patient adaptation should be further investigated in a bigger population with online experiments verifying our conclusions. Specifically, an online experimental protocol aiming to isolate patient training (with or without visual feedback) and decoders update influence on performance should be designed.

Our results were computed on a real-life dataset recorded with a tetraplegic patient. Analyzing this kind of dataset allows us to draw conclusions about the population in real need of assistive technology. However, interpretation of results is even more challenging than in the case of healthy subjects because we do not have access to solid ground-truth labels to train ML models. This increases the already long list of factors that can affect the performance of the decoders and may not be easily noticed when analyzing ECoG signals. For example, in the ideal ML world, one could analyze the learning curve and draw conclusions about the required dataset size to effectively train ML models. In our case, other factors like the nonstationarity of the signal play an important role in the process. In some cases, we may add more data to the dataset (e.g., BI experiment) and decrease the performance because we also 'extend' the data manifold with samples from a shifted distribution. A remedy for distribution shifts between sessions may be methods used for domain adaptation, for example, in EEG transfer learning [84]. Part of the aforementioned issues limiting our interpretation capabilities might be addressed with generative models [85] that are a popular tool in computer vision. In the case of

brain signals, the ability to produce signals with the given parameters and characteristics may be used to verify and understand phenomena observed in real-life experiments. First attempts to train GANs for EEG [86] data analysis were made, but a significant amount of work still has to be done to create a consistent framework for easier hypothesis evaluation.

5. Conclusions

Our analysis showed that adding more data to the training dataset may not be instantly profitable, starting from datasets containing 30-50 minutes of the signal in real-life scenarios. Instead, improvement may be achieved by creating a high-quality dataset that can be recorded after participant training. Furthermore, we showed the importance of patient adaptation in the human-in-the-loop system that enabled obtaining high-performance models with relatively small training datasets. Finally, we propose UMAP embeddings and local intrinsic dimensionality as a way to visualize the data and potentially evaluate data quality.

Data availability statement

The data analyzed during the current study are not publicly available for legal/ethical reasons but are available from the corresponding author on reasonable request.

Acknowledgments

Clinatec is a Laboratory of CEA-Leti at Grenoble and has statutory links with the University Hospital of Grenoble (CHUGA) and with University Grenoble Alpes (UGA). This study was funded by CEA (recurrent funding) and the French Ministry of Health (Grant PHRC-15-15-0124), Institut Carnot, Fonds de Dotation Clinatec. Matthieu Martin was supported by the cross-disciplinary program on Numerical Simulation of CEA. Maciej Śliwowski was supported by the CEA NUMERICS program, which has received funding from European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant agreement No 800945.

Appendix A. UMAP embeddings

Appendix B. Intrinsic dimensionality computed with TwoNN

References

[1] Laureen D. Hachem, Christopher S. Ahuja, and Michael G. Fehlings. Assessment and management of acute spinal cord injury: From point of injury to rehabilitation. *The Journal of Spinal Cord Medicine*, 40(6):665–675, November 2017.

[2] Bradley J Edelman, Jianjun Meng, Daniel Suma, C Zurn, E Nagarajan, BS Baxter, Christopher C Cline, and Bin He. Noninvasive neuroimaging enhances continuous neural tracking for robotic device control. *Science robotics*, 4(31), 2019.

[3] Leigh R Hochberg, Daniel Bacher, Beata Jarosiewicz, Nicolas Y Masse, John D Simeral, Joern Vogel, Sami Haddadin, Jie Liu, Sydney S Cash, Patrick Van Der Smagt, et al. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, 485(7398):372–375, 2012.

[4] B Wodlinger, JE Downey, EC Tyler-Kabara, AB Schwartz, ML Boninger, and JL Collinger. Ten-dimensional anthropomorphic arm control in a human brain- machine interface: difficulties, solutions, and limitations. *Journal of neural engineering*, 12(1):016011, 2014.

[5] Eduardo López-Larraz, Fernando Trincado-Alonso, Vijaykumar Rajasekaran, Soraya Pérez-Nombela, Antonio J Del-Ama, Joan Aranda, Javier Minguez, Angel Gil-Agudo, and Luis Montesano. Control of an ambulatory exoskeleton with a brain–machine interface for spinal cord injury gait rehabilitation. *Frontiers in neuroscience*, 10:359, 2016.

[6] Yongtian He, David Eguren, José M Azorín, Robert G Grossman, Trieu Phat Luu, and Jose L Contreras-Vidal. Brain–machine interfaces for controlling lower-limb powered robotic systems. *Journal of neural engineering*, 15(2):021004, 2018.

[7] No-Sang Kwak, Klaus-Robert Müller, and Seong-Whan Lee. A lower limb exoskeleton control system based on steady state visual evoked potentials. *Journal of neural engineering*, 12(5):056009, 2015.

[8] Jennifer L Collinger, Brian Wodlinger, John E Downey, Wei Wang, Elizabeth C Tyler-Kabara, Douglas J Weber, Angus JC McMorland, Meel Velliste, Michael L Boninger, and Andrew B Schwartz. High-performance neuroprosthetic control by an individual with tetraplegia. *The Lancet*, 381(9866):557–564, 2013.

[9] Alim Louis Benabid, Thomas Costecalde, Andrey Eliseyev, Guillaume Charvet, Alexandre Verney, Serpil Karakas, Michael Foerster, Aurélien Lambert, Boris Morinière, Neil Abroug, et al. An exoskeleton controlled by an epidural wireless brain–machine interface in a tetraplegic patient: a proof-of-concept demonstration. *The Lancet Neurology*, 18(12):1112–1122, 2019.

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

[11] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *CoRR*, abs/2001.08361, 2020.

[12] Derek Hoiem, Tanmay Gupta, Zhizhong Li, and Michal Shlapentokh-Rothman. Learning curves for analysis of deep networks. In *ICML*, 2021.

[13] Jonathan S. Rosenfeld, Amir Rosenfeld, Yonatan Belinkov, and Nir Shavit. A constructive prediction of the generalization error across scales. In *International Conference on Learning Representations*, 2020.

[14] Serafeim Perdikis and Jose del R Millan. Brain-machine interfaces: A tale of two learners. *IEEE Systems, Man, and Cybernetics Magazine*, 6(3):12–19, 2020.

[15] Tom J. Viering and Marco Loog. The shape of learning curves: a review. *CoRR*, abs/2103.10948, 2021.

[16] Daniel B. Silversmith, Reza Abiri, Nicholas F. Hardy, Nikhilesh Natraj, Adelyn Tu-Chan, Edward F. Chang, and Karunesh Ganguly. Plug-and-play control of a

Figure A1. Left hand dataset embedding showed for every session separately.

brain–computer interface through neural map stabilization. *Nature Biotechnology*, September 2020.

[17] Tobias Pistohl, Andreas Schulze-Bonhage, Ad Aertsen, Carsten Mehring, and Tonio Ball. Decoding natural grasp types from human ECoG. *NeuroImage*, 59(1):248–260, January 2012. Number: 1.

[18] Nanying Liang and Laurent Bougrain. Decoding Finger Flexion from Band-Specific ECoG Signals in Humans. *Frontiers in Neuroscience*, 6, 2012.

[19] Yasuhiko Nakanishi, Takufumi Yanagisawa, Duk Shin, Ryohei Fukuma, Chao Chen, Hiroyuki Kambara, Natsue Yoshimura, Masayuki Hirata, Toshiki Yoshimine, and Yasuharu Koike. Prediction of Three-Dimensional Arm Trajectories Based on ECoG Signals Recorded from Human Sensorimotor Cortex. *PLoS ONE*, 8(8):e72085, August 2013. Number: 8.

[20] Yasuhiko Nakanishi, Takufumi Yanagisawa, Duk Shin, Hiroyuki Kambara, Natsue Yoshimura, Masataka Tanaka, Ryohei Fukuma, Haruhiko Kishima, Masayuki Hirata, and Yasuharu Koike. Mapping ECoG channel contributions to trajectory and muscle activity prediction in human sensorimotor cortex. *Scientific Reports*, 7(1):45486, April 2017. Number: 1.

[21] Rémi Flamary and Alain Rakotomamonjy. Decoding Finger Movements from ECoG Signals Using Switching Linear Models. *Frontiers in Neuroscience*, 6, 2012.

[22] Weixuan Chen, Xilin Liu, and Brian Litt. Logistic-weighted regression improves decoding of finger flexion

Figure A2. Right hand dataset embedding showed for every session separately.

from electrocorticographic signals. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 2629–2632, Chicago, IL, August 2014. IEEE.

[23] David T Bundy, Mrinal Pahwa, Nicholas Szrama, and Eric C Leuthardt. Decoding three-dimensional reaching movements using electrocorticographic signals in humans. *Journal of Neural Engineering*, 13(2):026021, April 2016. Number: 2.

[24] Andrey Eliseyev, Vincent Auboiroux, Thomas Costecalde, Lilia Langar, Guillaume Charvet, Corinne Mestais, Tetiana Aksenova, and Alim-Louis Benabid. Recursive Exponentially Weighted N-way Partial Least Squares Regression with Recursive-Validation of Hyper-Parameters in Brain-Computer Interface Applications. *Scientific Reports*, 7(1):16281, December 2017. Number: 1.

[25] 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. *Journal of neural engineering*, 15(5):056013, 2018.

[26] Venkatesh Elango, Aashish N Patel, Kai J Miller, and Vikash Gilja. Sequence transfer learning for neural decoding. *bioRxiv*, page 210732, 2017.

[27] 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

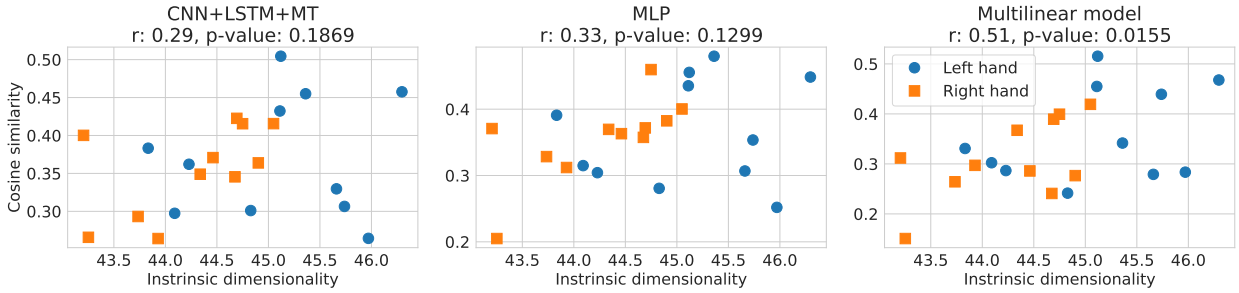


Figure B1. Relationship between cosine similarity and ID of the training dataset computed with TwoNN for dataset translation experiment. In the plot titles, Pearson correlation coefficient r and p-value (the probability of two uncorrelated inputs obtaining r at least as extreme as obtained in this case) are presented.

with convolutional neural networks for eeg decoding and visualization. *Human brain mapping*, 38(11):5391–5420, 2017.

[28] Ruilong Zhang, Qun Zong, Liqian Dou, and Xinyi Zhao. A novel hybrid deep learning scheme for four-class motor imagery classification. *Journal of neural engineering*, 16(6):066004, 2019.

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

[30] 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. *Frontiers in neuroscience*, page 555, 2018.

[31] Mamunur Rashid, Minarul Islam, Norizam Sulaiman, Bifta Sama Bari, Ripon Kumar Saha, and Md Jahid Hasan. Electrocorticography based motor imagery movements classification using long short-term memory (lstm) based on deep learning approach. *SN Applied Sciences*, 2(2):1–7, 2020.

[32] Ziqian Xie, Odelia Schwartz, and Abhishek Prasad. Decoding of finger trajectory from ecog using deep learning. *Journal of neural engineering*, 15(3):036009, 2018.

[33] 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, Jeju, Korea (South), October 2018. IEEE.

[34] Venkatesh Elango, Aashish N. Patel, Kai J. Miller, and Vikash Gilja. Sequence Transfer Learning for Neural Decoding. *bioRxiv*, page 210732, December 2017. Publisher: Cold Spring Harbor Laboratory Section: New Results.

[35] 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. *Frontiers in Neuroscience*, 12:555, August 2018.

[36] Mamunur Rashid, Minarul Islam, Norizam Sulaiman, Bifta Sama Bari, Ripon Kumar Saha, and Md Jahid Hasan. Electrocorticography based motor imagery movements classification using long short-term memory (LSTM) based on deep learning approach. *SN Applied Sciences*, 2(2):211, February 2020. Number: 2.

[37] Ziqian Xie, Odelia Schwartz, and Abhishek Prasad. Decoding of finger trajectory from ECoG using deep learning. *Journal of Neural Engineering*, 15(3):036009, June 2018. Number: 3.

[38] Maciej Śliwowski, Matthieu Martin, Antoine Souloumiac, Pierre Blanchart, and Tetiana Aksenova. Decoding ecog signal into 3d hand translation using deep learning. *Journal of Neural Engineering*, 19(2):026023, 2022.

[39] Jdel R Millan. On the need for on-line learning in brain-computer interfaces. In *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541)*, volume 4, pages 2877–2882. IEEE, 2004.

[40] Jonathan R Wolpaw, Niels Birbaumer, Dennis J McFarland, Gert Pfurtscheller, and Theresa M Vaughan. Brain–computer interfaces for communication and control. *Clinical neurophysiology*, 113(6):767–791, 2002.

[41] Jose M Carmena. Advances in neuroprosthetic learning and control. *PLoS biology*, 11(5):e1001561, 2013.

[42] Ricardo Chavarriaga, Melanie Fried-Oken, Sonja Kleih, Fabien Lotte, and Reinhold Scherer. Heading for new shores! overcoming pitfalls in bci design. *Brain-Computer Interfaces*, 4(1-2):60–73, 2017.

[43] Fabien Lotte, Florian Larrue, and Christian Mühl. Flaws in current human training protocols for spontaneous brain-computer interfaces: lessons learned from instructional design. *Frontiers in human neuroscience*, 7:568, 2013.

[44] 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. *Neuron*, 82(6):1380–1393, 2014.

[45] Camille Benaroch, Khadijeh Sadatnejad, Aline Roc, Aurélien Appriou, Thibaut Monseigne, Smeety Pramij, Jelena Mladenovic, Léa Pillette, Camille Jeunet, and Fabien Lotte. Long-term bci training of a tetraplegic user: Adaptive riemannian classifiers and user training. *Frontiers in Human Neuroscience*, 15, 2021.

[46] Ewan S Nurse, Sam E John, Dean R Freestone, Thomas J Oxley, Hoameng Ung, Samuel F Berkovic, Terence J O'Brien, Mark J Cook, and David B Grayden. Consistency of long-term subdural electrocorticography in humans. *IEEE Transactions on Biomedical Engineering*, 65(2):344–352, 2017.

[47] John P Cunningham, Paul Nuyujukian, Vikash Gilja, Cindy A Chestek, Stephen I Ryu, and Krishna V Shenoy. A closed-loop human simulator for investigating the role of feedback control in brain-machine interfaces. *Journal of neurophysiology*, 105(4):1932–1949, 2011.

[48] Beata Jarosiewicz, Nicolas Y Masse, Daniel Bacher, Sydney S Cash, Emad Eskandar, Gerhard Friehs, John P Donoghue, and Leigh R Hochberg. Advantages of closed-loop calibration in intracortical brain–computer interfaces for people with tetraplegia. *Journal of neural engineering*, 10(4):046012, 2013.

[49] Ranganatha Sitaram, Tomas Ros, Luke Stoeckel, Sven

Haller, Frank Scharnowski, Jarrod Lewis-Peacock, Nikolaus Weiskopf, Maria Laura Blefari, Mohit Rana, Ethan Oblak, et al. Closed-loop brain training: the science of neurofeedback. *Nature Reviews Neuroscience*, 18(2):86–100, 2017.

[50] Maryam M Shanechi, Amy L Orsborn, Helene G Moorman, Suraj Gowda, Siddharth Dangi, and Jose M Carmena. Rapid control and feedback rates enhance neuroprosthetic control. *Nature communications*, 8(1):1–10, 2017.

[51] Eran Dayan and Leonardo G Cohen. Neuroplasticity subserving motor skill learning. *Neuron*, 72(3):443–454, 2011.

[52] John W Krakauer, Alkis M Hadjiosif, Jing Xu, Aaron L Wong, and Adrian M Haith. Motor learning. *Compr Physiol*, 9(2):613–663, 2019.

[53] Elisa Mira Holz, Johannes Höhne, Pit Staiger-Sälzer, Michael Tangermann, and Andrea Kübler. Brain–computer interface controlled gaming: Evaluation of usability by severely motor restricted end-users. *Artificial intelligence in medicine*, 59(2):111–120, 2013.

[54] Johannes Höhne, Elisa Holz, Pit Staiger-Sälzer, Klaus-Robert Müller, Andrea Kübler, and Michael Tangermann. Motor imagery for severely motor-impaired patients: evidence for brain-computer interfacing as superior control solution. *PloS one*, 9(8):e104854, 2014.

[55] Jianjun Meng, Shuying Zhang, Angeliki Bekyo, Jaron Olsoe, Bryan Baxter, and Bin He. Noninvasive electroencephalogram based control of a robotic arm for reach and grasp tasks. *Scientific Reports*, 6(1):1–15, 2016.

[56] Robert Leeb, Luca Tonin, Martin Rohm, Lorenzo Desideri, Tom Carlson, and Jose del R Millan. Towards independence: a bci telepresence robot for people with severe motor disabilities. *Proceedings of the IEEE*, 103(6):969–982, 2015.

[57] Christa Neuper, Gernot R Müller, Andrea Kübler, Niels Birbaumer, and Gert Pfurtscheller. Clinical application of an eeg-based brain–computer interface: a case study in a patient with severe motor impairment. *Clinical neurophysiology*, 114(3):399–409, 2003.

[58] Dennis J McFarland, William A Sarnacki, and Jonathan R Wolpaw. Electroencephalographic (eeg) control of three-dimensional movement. *Journal of neural engineering*, 7(3):036007, 2010.

[59] Leigh R Hochberg, Mijail D Serruya, Gerhard M Friehs, Jon A Mukand, Maryam Saleh, Abraham H Caplan, Almut Branner, David Chen, Richard D Penn, and John P Donoghue. Neuronal ensemble control of prosthetic devices by a human with tetraplegia. *Nature*, 442(7099):164–171, 2006.

[60] A Bolu Ajiboye, Francis R Willett, Daniel R Young, William D Memberg, Brian A Murphy, Jonathan P Miller, Benjamin L Walter, Jennifer A Sweet, Harry A Hoyen, Michael W Keith, et al. Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration. *The Lancet*, 389(10081):1821–1830, 2017.

[61] Serafeim Perdikis, Luca Tonin, Sareh Saeedi, Christoph Schneider, and José del R Millán. The cybathlon bci race: Successful longitudinal mutual learning with two tetraplegic users. *PLoS biology*, 16(5):e2003787, 2018.

[62] Jonathan R Wolpaw and Dennis J McFarland. Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *Proceedings of the national academy of sciences*, 101(51):17849–17854, 2004.

[63] Alexandre Moly, Thomas Costecalde, Félix Martel, Matthieu Martin, Christelle Larzabal, Serpil Karakas, Alexandre Verney, Guillaume Charvet, Stephan Chabardes, Alim Louis Benabid, et al. An adaptive closed-loop ecog decoder for long-term and stable bimanual control of an exoskeleton by a tetraplegic. *Journal of Neural Engineering*, 19(2):026021, 2022.

[64] Fabien Lotte and Camille Jeunet. Defining and quantifying users' mental imagery-based bci skills: a first step. *Journal of neural engineering*, 15(4):046030, 2018.

[65] 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. *The Lancet Neurology*, 18(12):1112–1122, December 2019. Number: 12.

[66] C. S. Mestais, G. Charvet, F. Sauter-Starace, M. Foerster, D. Ratel, and A. L. Benabid. Wimagine: Wireless 64-channel ecog recording implant for long term clinical applications. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(1):10–21, 2015.

[67] Andrey Eliseyev, Vincent Auboiroux, Thomas Costecalde, Lilia Langar, Guillaume Charvet, Corinne Mestais, Tetiana Aksenova, and Alim-Louis Benabid. Recursive exponentially weighted n-way partial least squares regression with recursive-validation of hyper-parameters in brain-computer interface applications. *Scientific reports*, 7(1):1–15, 2017.

[68] Leland McInnes, John Healy, and James Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv e-prints*, page arXiv:1802.03426, February 2018.

[69] Dmitry Kobak and George C. Linderman. Initialization is critical for preserving global data structure in both t-SNE and UMAP. *Nature Biotechnology*, 39(2):156–157, February 2021. Number: 2 Publisher: Nature Publishing Group.

[70] Performance comparison of dimension reduction implementations, 2018.

[71] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.

[72] Maciej Śliwowski, Matthieu Martin, Antoine Souloumiac, Pierre Blanchart, and Tetiana Aksenova. Decoding ECoG signal into 3d hand translation using deep learning. *Journal of Neural Engineering*, 19(2):026023, mar 2022.

[73] Corinna Cortes, L. D. Jackel, Sara Solla, Vladimir Vapnik, and John Denker. Learning curves: Asymptotic values and rate of convergence. In J. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems*, volume 6. Morgan-Kaufmann, 1993.

[74] Benjamin Strang, Peter van der Putten, Jan N. van Rijn, and Frank Hutter. Don't rule out simple models prematurely: A large scale benchmark comparing linear and non-linear classifiers in openml. In Wouter Duivesteijn, Arno Siebes, and Antti Ukkonen, editors, *Advances in Intelligent Data Analysis XVII*, pages 303–315, Cham, 2018. Springer International Publishing.

[75] Baohua Gu, Feifang Hu, and Huan Liu. Modelling classification performance for large data sets. In X. Sean Wang, Ge Yu, and Hongjun Lu, editors, *Advances in Web-Age Information Management*, pages 317–328, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[76] Phil Pope, Chen Zhu, Ahmed Abdelkader, Micah Gold-

blum, and Tom Goldstein. The intrinsic dimension of images and its impact on learning. In *International Conference on Learning Representations*, 2021.

[77] Kerstin Johnsson, Charlotte Soneson, and Magnus Fontes. Low bias local intrinsic dimension estimation from expected simplex skewness. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(1):196–202, 2015.

[78] Elena Facco, Maria d'Errico, Alex Rodriguez, and Alessandro Laio. Estimating the intrinsic dimension of datasets by a minimal neighborhood information. *Scientific Reports*, 7, 2017.

[79] Piotr Tempczyk, Rafał Michaluk, Lukasz Garncarek, Przemysław Spurek, Jacek Tabor, and Adam Golinski. LIDL: Local intrinsic dimension estimation using approximate likelihood. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 21205–21231. PMLR, 17–23 Jul 2022.

[80] Jonathan Bac and Andrei Zinovyev. Local intrinsic dimensionality estimators based on concentration of measure. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2020.

[81] Jonathan Bac, Evgeny M. Mirkes, Alexander N. Gorban, Ivan Tyukin, and Andrei Zinovyev. Scikit-dimension: A python package for intrinsic dimension estimation. *Entropy*, 23(10), 2021.

[82] Cédric Rommel, Joseph Paillard, Thomas Moreau, and Alexandre Gramfort. Data augmentation for learning predictive models on eeg: a systematic comparison. *ArXiv*, abs/2206.14483, 2022.

[83] Cédric Rommel, Thomas Moreau, Joseph Paillard, and Alexandre Gramfort. CADDA: Class-wise automatic differentiable data augmentation for EEG signals. In *International Conference on Learning Representations*, 2022.

[84] He Zhao, Qingqing Zheng, Kai Ma, Huiqi Li, and Yefeng Zheng. Deep representation-based domain adaptation for nonstationary eeg classification. *IEEE Transactions on Neural Networks and Learning Systems*, 32(2):535–545, 2021.

[85] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[86] Kay Gregor Hartmann, Robin Tibor Schirrmeister, and Tonio Ball. Eeg-gan: Generative adversarial networks for electroencephalograhic (eeg) brain signals, 2018.

4

Deep learning for ECoG brain-computer interface: end-to-end vs. hand-crafted features

This chapter contains pre-print version of article submitted to the 21st International Conference of the Italian Association for Artificial Intelligence (AIxIA 2022) (accepted): Śliwowski, M., Martin, M., Souloumiac, A., Blanchart, P., and Aksenova, T. Deep learning for ECoG brain-computer interface: end-to-end vs. hand-crafted features. URL https://arxiv.org/abs/2210.02544

Deep learning for ECoG brain-computer interface: end-to-end vs. hand-crafted features

Maciej Śliwowski[1,2][0000−0001−6744−1714], Matthieu Martin[1][0000−0001−5954−8087], Antoine Souloumiac[2], Pierre Blanchart[2], and Tetiana Aksenova[1][0000−0003−4007−2343]

[1] Univ. Grenoble Alpes, CEA, LETI, Clinatec, F-38000 Grenoble, France
[2] Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France

Abstract. In brain signal processing, deep learning (DL) models have become commonly used. However, the performance gain from using end-to-end DL models compared to conventional ML approaches is usually significant but moderate, typically at the cost of increased computational load and deteriorated explainability. The core idea behind deep learning approaches is scaling the performance with bigger datasets. However, brain signals are temporal data with a low signal-to-noise ratio, uncertain labels, and nonstationary data in time. Those factors may influence the training process and slow down the models' performance improvement. These factors' influence may differ for end-to-end DL model and one using hand-crafted features.
As not studied before, this paper compares models that use raw ECoG signal and time-frequency features for BCI motor imagery decoding. We investigate whether the current dataset size is a stronger limitation for any models. Finally, obtained filters were compared to identify differences between hand-crafted features and optimized with backpropagation. To compare the effectiveness of both strategies, we used a multilayer perceptron and a mix of convolutional and LSTM layers that were already proved effective in this task. The analysis was performed on the long-term clinical trial database (almost 600 minutes of recordings) of a tetraplegic patient executing motor imagery tasks for 3D hand translation.
For a given dataset, the results showed that end-to-end training might not be significantly better than the hand-crafted features-based model. The performance gap is reduced with bigger datasets, but considering the increased computational load, end-to-end training may not be profitable for this application.

Keywords: deep learning · ECoG · brain-computer interfaces · dataset size · motor imagery · end-to-end

1 Introduction

In the last decade, deep learning (DL) models achieved extraordinary performance in a variety of complex real-life tasks, e.g., computer vision [4], natural language processing [2], compared to previously developed models. This

was possible mainly thanks to the improvements of data processing units and, most importantly, increased dataset sizes [4]. Generally, in brain-computer interfaces (BCI) research, access to large databases of brain signals is limited due to the experimental and medical constraints as well as the immensity of paradigms/hardware combinations. Given limited datasets, can we still train end-to-end (E2E) DL models for the medical BCI application as effectively as in computer vision?

In 2019, Roy et al. [12] reported that the number of studies classifying EEG signals with deep learning using hand-crafted features (mainly frequency domain) and raw EEG signals (end-to-end) was similar. This indicates that decoding EEG from raw signals is indeed possible. However, in many articles, researchers decided to use harder to design hand-crafted features. While end-to-end models dominated computer vision, in brain signals processing, it is still common to use features extracted as an input to the DL models. It is unclear whether specific signal characteristics cause this, e.g., nonstationarity in time making the creation of a homogeneous dataset impractical, low signal-to-noise ratio complicating the optimization process and favoring overfitting, labels uncertainty originating from human-in-the-loop experimental setup, or researchers' bias toward solutions better understood and more explainable.

Most studies do not directly compare DL using end-to-end and hand-crafted features approaches. Usually, DL architectures are compared with each other and with an additional 'traditional' ML pipeline, e.g., filter-bank common spatial pattern (FBCSP) in [15], xDAWN and FBCSP in [5], SVM and FBCSP in [17]. In figure 1, we presented accuracy improvement of the best proposed DL model compared to the 'traditional' baseline for articles analyzed in [12] [3] depending on the recording time and the number of examples in the dataset. The gap between performance improvement of DL compared to the 'traditional' baseline increases with the dataset size (except for the last points on the plot, which contain significantly fewer studies). In the right plot, the difference between models using raw EEG and frequency domain features increases which may exhibit a boost of end-to-end models with access to bigger datasets compared to hand-crafted features. As the proposed DL models are usually compared to the baseline, the boost of end-to-end models cannot be clearly stated because the accuracy difference depends strongly on the 'traditional' baseline model performance and the particular task tackled in the study.

While EEG and ECoG signals share many characteristics—both are multi-channel temporal signals with information encoded in frequency and space, with low signal-to-noise ratio and noisy labels—there are also differences, e.g., a higher spatial resolution of ECoG, higher signal-to-noise ratio and higher contribution of informative high gamma band (> 70Hz). In motor imagery ECoG decoding, end-to-end DL is not commonly used. 'Traditional' ML classifiers are usually preceded by a feature extraction step creating brain signals representation, typically in the form of time-frequency features, containing information about power

[3] limited to the articles that contained all the required information, code adapted from [12]

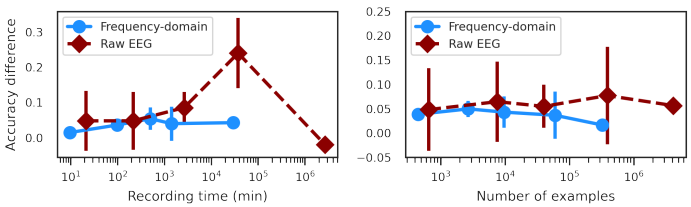


Fig. 1: Binned average accuracy difference between best proposed DL model and 'traditional' baseline on EEG datasets. Error bars denote one standard deviation of the values in the bin. Bins are equal in size on a logarithmic scale. Points x-axis position denotes the average dataset size in a bin.

time course in several frequency bands [8, 14] or focused only on low-frequency component (LFC)/Local Motor Potential (LMP) [14] (detailed analysis can be found in [19]).

However, a successful application of an end-to-end DL model to motor imagery decoding of finger movements trajectory from ECoG was performed with convolutional layers filtering the raw signal both in temporal and spatial domains followed by LSTM layers [20]. Smart weights initialization was helpful in achieving high performance. Nevertheless, an average improvement from training the weights can be estimated as 0.022 ± 0.0393 of Pearson r correlation coefficient, which is relatively small, with 66% of cases noticeable improvement from end-to-end training (at the level of subjects/fingers). As this was not studied before, we investigated the differences in data requirements between an end-to-end model and one using hand-crafted features on a long-term clinical trial BCI dataset of 3D target reach task. Unique long-term recordings (several months of experiments, more than 600 min duration in total, compared to few minutes of ECoG recording available in previous studies, e.g., [20]) allowed us to explore the relationship between dataset size and the type of feature used for ECoG signal decoding. In this study, we used architectures previously applied to the ECoG dataset for decoding motor imagery signals with hand-crafted time-frequency features as input [16]. In addition, we optimized the temporal filtering layer with backpropagation seeking a more efficient set of filters that were initialized to reproduce continuous wavelet transform. We also investigated whether both approaches react differently to training dataset perturbations which may be the case due to distinct model properties and may influence the choice of optimal data processing pipeline for ECoG BCI.

2 Methods

2.1 Dataset

The dataset used in this study was collected as a part of the clinical trial 'BCI and Tetraplegia' (ClinicalTrials.gov identifier: NCT02550522, details in [1]) approved

by the ethical Committee for the Protection of Individuals (Comité de Protection des Personnes—CPP) with the registration number: 15-CHUG-19 and the Agency for the Safety of Medicines and Health Products (Agence nationale de sécurité du médicament et des produits de santé—ANSM) with the registration number: 2015-A00650-49 and the ethical Committee for the Protection of Individuals (Comité de Protection des Personnes—CPP) with the registration number: 15-CHUG-19.

In the experiment, a 28-years-old tetraplegic patient after spinal cord injury was asked to move the hands of a virtual avatar displayed on a screen (see figure 2) using motor imagery patterns—by imaging/attempting hand movements that influence brain activity in the motor cortex. These changes were then recorded with two WIMAGINE [10] implants placed over the primary motor and sensory cortex bilaterally. Both implants consisted of 8×8 grid of electrodes with recording performed using 32 electrodes selected in a chessboard-like manner due to limited data transfer with a sampling frequency equal to 586 Hz. Signals from implants were transferred to the decoding system that performed online predictions. First, one out of 5 possible states (idle, left and right hand translation, left and right wrist rotation) was selected with a state decoder. Then, for every state (except idle), a multilinear REW-NPLS model [3] updated online was used to predict 3D movements or 1D wrist rotation. The dataset consisted of 44 experimental sessions recorded over more than 200 days. It constitutes 300 and 284 minutes for left and right hand translation, respectively.

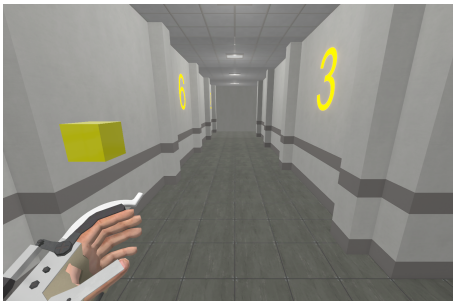


Fig. 2: Screenshot from the virtual environment. The patient is asked to reach the yellow square (target) with the left hand (effector) using motor imagery.

2.2 Data representation and problem

Based on the collected database, we extracted two datasets for left and right hand translation. The raw signal representation was created from 1-second long windows of ECoG signal with 90% overlap. Every observation $\mathbf{X}_i \in \mathbb{R}^{64\times590}$ contained 590 samples for each of the 64 channels corresponding to the number of electrodes recording the signal.

Every signal window $\mathbf{X}_i$ was paired with the corresponding desired trajectory $\mathbf{y}_i \in \mathbb{R}^3$ that the patient was asked to follow, i.e., the straight line connecting the tip of the hand to the target. The trajectories were computed in the 3D virtual avatar coordinate system mounted in the pelvis of the effector.

Before feeding the data to the models, datasets were cleaned from data loss artifacts that were not caught during the online recordings. Additionally, observations for which the predicted and desired state did not match due to state decoder errors were also removed to reduce the number of mislabelled observations (e.g., when the patient was asked to control left hand translation but instead left wrist was rotating).

Then, all the models were trained to find the mapping between $\mathbf{X}_i$ ECoG signal and $\mathbf{y}_i$ desired trajectories that the exoskeleton hand should follow in the case of optimal prediction. As a performance metric we used cosine similarity (equation 1) measuring cosine of the angle α_i between prediction $\hat{\mathbf{y}}_i$ and the desired trajectory $\mathbf{y}_i$.

$$\mathrm{CS}(\mathbf{y}_i, \hat{\mathbf{y}}_i) = \frac{\mathbf{y}_i \cdot \hat{\mathbf{y}}_i}{\|\mathbf{y}_i\| \cdot \|\hat{\mathbf{y}}_i\|} = \cos \alpha_i \tag{1}$$

Cosine loss defined as $\mathrm{CL}(\mathbf{y}_i, \hat{\mathbf{y}}_i) = 1 - \mathrm{CS}(\mathbf{y}_i, \hat{\mathbf{y}}_i)$ was used as optimization objective.

2.3 'Traditional' feature extraction and DL optimization

Hand-crafted features were extracted using complex continuous wavelet transform (CWT). CWT was performed with Morlet wavelets with central frequencies ranging from 10 to 150 Hz with a step of 10 Hz. Each wavelet support consisted of 118 samples (0.2s) centered on its maximum value. Features were obtained by applying CWT on one-second-long signals, computing the module of the complex signals, and performing an average pooling of 0.1 second. The resulting feature tensor was of shape $64 \times 15 \times 10$, with dimensions corresponding to channels, frequency bands, and time steps.

CWT can be represented as a convolution between a set of filters and a signal in the temporal domain. In the standard case, the filters are fixed and constitute a basis for feature extraction where every filter detects brain activity in a different frequency band. As every spatial channel is convolved separately in time, we obtained a time-frequency-space representation of the ECoG signal (see table 1 for feature extractor architecture specification).

Here, we propose to adjust the filters during backpropagation together with all other parameters of the models. In the first scenario, the filters were initialized to Morlet wavelets with 15 central frequencies, resulting in 30 kernels (real and imaginary parts). Note that at the beginning of training, the first layer reproduces 'traditional' feature extraction. The filters were fixed for 5 epochs of so-called pre-training, then they were unfreezed and optimized for the following 50 epochs. The pre-training was used to not distort the wavelets drastically in the first epochs when parameters of the rest of the network are randomly initialized. We also evaluated random weights initialization from uniform distribution as a solution that does not incorporate prior knowledge about the system.

In the second scenario, an alternative approach was used to maintain the wavelet structure by optimizing only the parameters used to generate the wavelets

instead of modifying all filters' parameters. In our case, the function generating the wavelets was defined as:

$$\Psi(t,f) = \frac{1}{\sqrt{\pi}} \frac{1}{\sqrt{\frac{f_s}{f}}} e^{-(tf)^2} e^{2i\pi tf} \tag{2}$$

where central frequency parameter f defines the center of the frequency band analyzed by the wavelet and f_s is the signal sampling frequency. In the central frequency optimization (CFO) scenario, we optimized only the central frequency f parameters (one per wavelet), so the filters after training are still from the Morlet wavelets family.

Table 1: The architecture used to reproduce hand-crafted feature extraction with CWT. Only one convolutional layer (conv time) was used in computations according to the performed experiment E2E/E2E CFO.

Layer	Kernel Shape	Output Shape	Param #	Mult-Adds
Input	–	[200, 1, 590, 8, 8]	–	–
Conv time	[1, 30, 118, 1, 1]	[200, 30, 590, 8, 8]	3,570	27,006,336,000
Conv time CFO	[1, 30, 118, 1, 1]	[200, 30, 590, 8, 8]	15	27,006,336,000
Square	–	[200, 30, 590, 8, 8]	–	–
Sum real and imaginary	–	[200, 15, 590, 8, 8]	–	–
Square root	–	[200, 15, 590, 8, 8]	–	–
Dropout	–	[200, 15, 590, 8, 8]	–	–
AvgPool	–	[200, 15, 10, 8, 8]	–	–
BatchNorm	[15]	[200, 15, 10, 8, 8]	30	6,000

2.4 DL architectures

In this study, we used two architectures proposed in [16], i.e., CNN+LSMT+MT, which showed the best performance, and MLP, which was the simplest approach. In the baseline approach, the 'traditional' feature extraction was followed with fully connected or convolutional layers. When optimizing the first convolutional layer, we kept the rest of the network the same to isolate the influence of the training feature extraction step. Details of the tested DL architectures are described below and in [16]. Additionally, we used ShallowFBCSPNet and Deep4Net [15] as end-to-end DL baseline.

MLP The most basic DL architecture evaluated in the study was multilayer perceptron (MLP), consisting of two fully connected layers. Dropout and batch normalization layers were placed between fully connected layers for stronger regularization (see table 2).

Table 2: MLP architecture from [16].

Layer	Kernel Shape	Output Shape	Param #	Mult-Adds
Flatten	–	[200, 9600]	–	–
Fully connected	[9600, 50]	[200, 50]	480,050	96,010,000
BatchNorm	[50]	[200, 50]	100	20,000
ReLU	–	[200, 50]	–	–
Dropout	–	[200, 50]	–	–
Fully connected	[50, 50]	[200, 50]	2,550	510,000
ReLU	–	[200, 50]	–	–
Dropout	–	[200, 50]	–	–
Fully connected	[50, 3]	[200, 3]	153	30,600

CNN+LSTM+MT In the CNN+LSTM+MT architecture, CWT features were further analyzed with 3×3 convolutional layers in space (electrodes organized on an array 4×8 reflecting positions of electrodes on implants). After two convolutional layers, two LSTM layers were applied to analyze temporal information from 10 timesteps. Finally, every output of the last LSTM layer was used for training to compute loss based on all predicted and ground truth trajectories corresponding to 1 second (10 timesteps) of signal analyzed (see table 3).

Table 3: CNN+LSTM+MT architecture from [16].

Layer	Kernel Shape	Output Shape	Param #	Mult-Adds
Input		[200, 15, 8, 8, 10]	–	
Input per implant		[200, 15, 8, 4, 10]	–	
Conv space	[15, 32, 3, 3, 1]	[200, 32, 6, 4, 10]	4,352	208,896,000
ReLU	–	[200, 32, 6, 4, 10]	–	–
BatchNorm	[32]	[200, 32, 6, 4, 10]	64	12,800
Dropout	–	[200, 32, 6, 4, 10]	–	–
Conv space	[32, 64, 3, 3, 1]	[200, 64, 4, 2, 10]	18,496	295,936,000
ReLU	–	[200, 64, 4, 2, 10]	–	–
Dropout	–	[200, 64, 4, 2, 10]	–	–
LSTM	–	[200, 10, 50]	215,200	430,400,000
LSTM	–	[200, 10, 3]	660	

Models training and hyperparameters For every model evaluation, we used 90% and 10% of the training dataset for training and validation, respectively. The validation dataset was used for early stopping after 20 epochs without improvement. All the models used a learning rate of 0.001, weight decay of 0.01,

batch size of 200, and ADAM optimizer [9]. To train DL models we used PyTorch [11], skorch [18], and braindecode [15].

2.5 Offline experiments

First, we computed results in a classical scenario, i.e., train/valid/test split. We used the calibration dataset (first six sessions) as the training dataset. The rest of the data (online evaluation dataset) was used as the test set.

Additionally, we gradually increased the training dataset size from one session up to 22 with a step of 2. As different models may have different dataset requirements, we wanted to verify whether collecting more data can be more profitable for one of the evaluated optimization/architecture combinations.

To investigate the possible influence of end-to-end learning on models' robustness against data mislabelling, we perturbed the dataset to make training more challenging. In the BCI, part of observations is often mistakenly labeled due to lack of subject attention, tiredness, experimental setup, etc. Therefore, we randomly selected a fraction of observations in which targets were shuffled between samples so they no longer have a connection with the ECoG signal while preserving the same distribution. At the same time, we kept the test set unchanged.

3 Results

Table 4: Cosine similarity computed in the train-valid-test split scenario. Values are sorted by average performance and represent the mean and standard deviation of 5 runs.

	Left hand	Right hand
E2E CNN+LSTM+MT CFO	$\mathbf{0.304 \pm 0.005}$	0.266 ± 0.020
CNN+LSTM+MT	0.297 ± 0.008	0.270 ± 0.011
E2E CNN+LSTM+MT	0.289 ± 0.007	$\mathbf{0.273 \pm 0.015}$
E2E MLP CFO	0.254 ± 0.012	0.230 ± 0.013
MLP	0.247 ± 0.023	0.232 ± 0.005
E2E MLP	0.243 ± 0.014	0.234 ± 0.020
ShallowFBCSPNet [15]	0.235 ± 0.010	0.236 ± 0.011
E2E CNN+LSTM+MT random init	0.216 ± 0.008	0.230 ± 0.020
E2E MLP random init	0.181 ± 0.029	0.223 ± 0.008
Deep4Net [15]	0.111 ± 0.021	0.259 ± 0.013

We started the analysis by comparing different model training scenarios when trained on the first six sessions (online calibration dataset). The results for the train/test split can be found in table 4. Differences between scenarios are rather

small, with only small performance improvement coming from full end-to-end optimization. The best performance was achieved by models using CFO. However, the gap between the hand-crafted features approach and CFO is relatively small, considering standard deviations of the computed values. The worst performance was achieved for Deep4Net (especially low performance for the left hand dataset) and both MLP and CNN+LSTM+MT models with random weights initialization, suggesting the high importance of the prior signal processing knowledge used to define the wavelet shape of the filters at the beginning of the training.

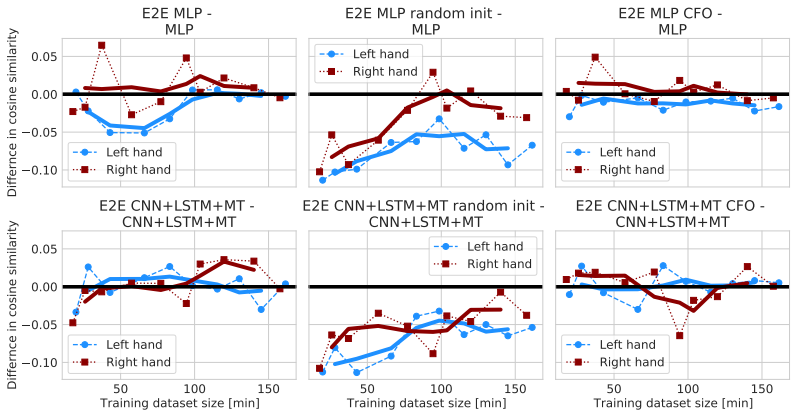


Fig. 3: Difference between end-to-end model and its counterpart using hand-crafted features. The bold line denotes the moving average with a window of size 3.

We did not notice significant improvements coming from end-to-end optimization, so we wanted to verify the hypothesis of different dataset size requirements for different optimization methods. Therefore, the differences between end-to-end models and their hand-crafted features counterparts for several training dataset sizes are presented in figure 3. In some cases, end-to-end models increase the cosine similarity faster than the models using fixed features, so the gap between models can be reduced for approaches using random weights initialization. However, only for models initialized to wavelets and optimized directly, an improvement over hand-crafted features can be observed for some points (up to 0.05 of cosine similarity for the right hand dataset).

When comparing CFO and standard E2E optimization in figure 4, higher effectiveness of CFO for small training datasets can be observed. CFO may limit overfitting as the functions represented by the convolutional filters are constrained to the wavelet family. It may be interpreted as an additional optimization constraint imposed on model parameters. Note that diminished gap

between CFO and standard end-to-end in figure 4 show only relative decrease of CFO performance.

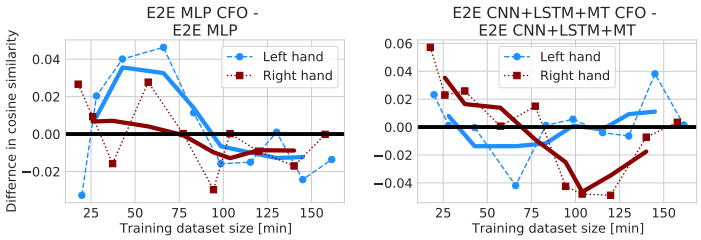


Fig. 4: Difference between the CFO model and its counterpart using constraint-free end-to-end optimization. The bold line denotes the moving average with a window of size 3.

3.1 Filters visualization

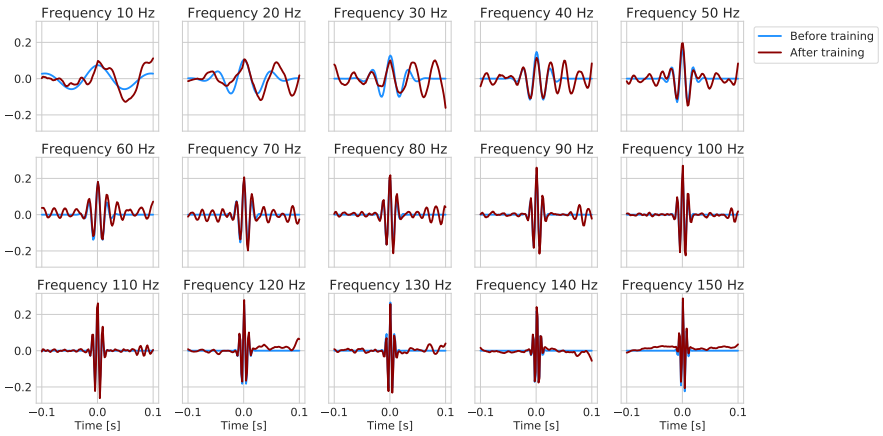


Fig. 5: Visualized filters before (blue) and after (orange) training for the models with parameters optimized freely. Note that only real part of the wavelet was visualized for clarity. Plot titles denote central wavelet frequency at initialization.

We visualized the filters before and after training to analyze the characteristics of learned feature extraction. In figure 5, we presented the filters modified

without additional constraints. The biggest change can be observed in the central frequencies between 30 Hz and 80 Hz. In most cases, the initial central frequency was maintained, while the wavelets got extended with a signal similar to the sine wave in the central wavelet frequency. This could indicate the importance of information about frequencies from which the signal is composed. At the same time, extending wavelets reduces the temporal resolution of the signals. The changes in the high-frequency wavelets (> 100Hz) are less significant, and the pattern of extending wavelets is no longer visible. Instead, components of significantly lower frequencies and smaller amplitude were added.

In figure 6, we visualized filters before and after optimization when the convolutional layer was initialized to random. As random filters were much harder to analyze visually, we presented them in the form of power spectra, so the changes in the filtered frequencies could be better visible. All filters have a maximum power peak lower than 65 Hz with 40% of maxima contained in the frequency range 25-30Hz. Compared to hand-crafted features, end-to-end filters initialized to random covered only approximately half of the frequency band analyzed by the traditional feature extraction pipeline. However, in the higher frequencies, there are smaller peaks which can also contribute to the extracted representation and may cover the missing frequency band.

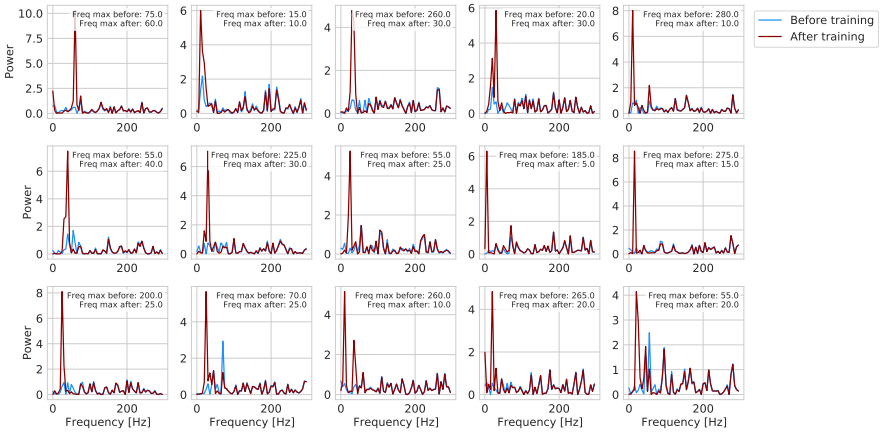
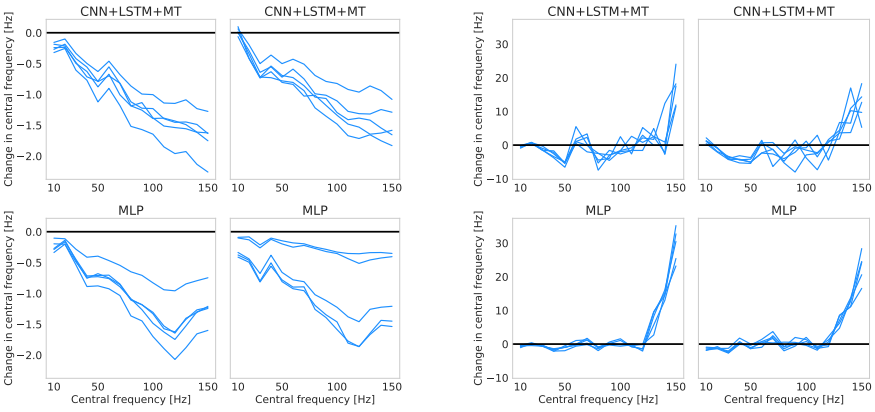


Fig. 6: Power spectra of filters before (blue) and after (orange) training for convolutional layer initialized to random. The plots denoted frequencies for which maximum power spectra were observed before and after training.

In figure 7.a, we presented the difference between initialized central wavelet frequency and the one obtained after the training. We observed a decrease in almost all frequencies when training the models. The decrease was higher for higher frequencies. This may suggest that more information can be extracted from lower frequencies. However, in our preliminary results, we noticed that

adapting the learning rate for the convolutional layer may significantly change the frequency behavior (see figure 7.b), which should be taken into account when analyzing the results. This may lead to different changes in the central frequencies than in the base model. The gradient was increased 150 times by squeezing central frequencies from 10-150Hz to 0-1. In the case of initialization to wavelet, a network may start the training near a local minimum found by the manual design of feature extraction that can be hard to move out. Setting a higher learning rate may enable reaching different regions on the loss function surface. The performance achieved with a higher learning rate was similar to the standard CFO results with a cosine similarity of 0.283 ± 0.014 (left hand) and 0.270 ± 0.011 (right hand) for CNN+LSTM+MT and 0.262 ± 0.01 (left hand) and 0.227 ± 0.007 (right hand) for MLP.



(a) standard central frequencies in range 10-150

(b) central frequencies squeezed to range 0-1

Fig. 7: Difference between central wavelet frequencies before and after CFO. Models for left hand translation are presented in the left column, for the right hand in the right column. Note that the scale is different for the (a) and (b) figures.

3.2 Target perturbation

In the case of perturbed ground-truth (figure 8), CNN+LSTM+MT models were more robust to noise in the targets with increased stability (especially for the left hand) of hand-crafted features and CFO models compared to models optimized freely. On the other hand, in the case of MLP models, almost no differences between different optimization methods in the influence of noise on the performance were noticed.

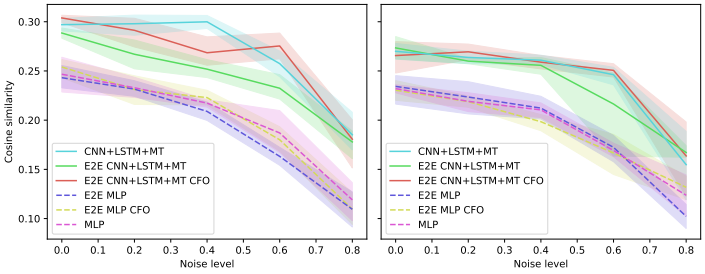


Fig. 8: Influence of noise in the targets on models' performance. Noise level indicates the fraction of observations with perturbed labels.

4 Discussion

We proposed several approaches for the end-to-end optimization of deep learning ECoG decoders. However, in this study, we did not observe improvement from end-to-end optimization, especially when no prior knowledge was used for filter initialization. This confirms the usefulness of hand-crafted features and years of neuroscientific signal processing while leaving doors open to more sophisticated end-to-end models. Firstly, deeper models with more advanced DL mechanisms [6, 13] should be evaluated as they may allow for the extraction of more complex representations and thus outperform hand-crafted features. Secondly, machine learning methods for robust learning may be evaluated, e.g., learning from noisy input data, noisy labels, and out-of-distribution samples [7]. Those can particularly tackle problems coming from specific recording/experimental circumstances.

The reasoning behind our study is focused on the specificity of ECoG brain signals and the adequacy of selected DL methods to the problem. The specificity originates from experimental constraints caused by the presence of a human in the loop but also signals characteristics, hardware capabilities, etc. It results in a distorted dataset with a low signal-to-noise ratio, short signal stationarity interval, and uncertain labels. This is quite different from computer vision problems, usually with well-defined labels and images understandable with a naked eye. Improving information extraction from noisy data may be especially important in the light of increased robustness to noise in targets shown by the CNN+LSTM+MT model compared to MLP. Using all 10 targets recorded during a 1-second window decreases the influence of single perturbed points on the performance because the information can be efficiently extracted even for 40% or 60% of perturbed targets. In this case, the CNN+LSTM+MT model using hand-crafted features maintains high performance for a higher noise level than the end-to-end model. However, an important point in the discussion is that our dataset, even after data cleaning, still contains a significant amount of observations with incorrect labels. Thus, in figure 8, a noise level equal to zero corresponds to an unknown noise level in labels originating from the experimental

setup. Thus, generative models should be used to create datasets with a known level of noise and analyze the influence of perturbations on the performance in the case of less distorted datasets.

All the results were computed offline on datasets recorded with only one patient. These kinds of datasets are hardly accessible due to experimental and legal constraints. It makes the generalization of the results to other patients and datasets hard to estimate. Thus, more simulations should be performed to confirm our conclusions, ideally with more patients and tasks. This should also include hyperparameters search, like learning rate, batch size, weight decay, as those could vary between different approaches. However, performing hundreds of evaluations is time-consuming, and the problem is magnified in the case of end-to-end models due to increased computational load. Our study focused on feature extraction based on wavelet transform, which was previously used in this problem. As we optimized the parameters of the wavelet transform without changing other parts of the model, we isolated the influence of end-to-end optimization on models' performance. While this simplified the problem, our study did not evaluate other feature extraction pipelines that could behave differently. Thus, an extended analysis of several feature extraction pipelines compared to their end-to-end counterparts would allow for broader generalization and therefore is of great interest.

While this article and [20] analyzed ECoG signals, targets used for training models in [20] were actual fingers trajectories recorded while subjects performed real movements. In our case, targets are much noisier due to the lack of labeling based on the hand movements of a tetraplegic patient. This may favor hand-crafted features, as could be seen for CNN+LSTM+MT in figure 8. Finally, our conclusions are in line with [20] who observed relatively small improvement from optimizing hand-crafted features and worse performance/longer training time when initializing the model to random. In our case, end-to-end models achieved the same performance as models using CWT features only with smart weights initialization, which emphasizes the importance of prior signal processing knowledge in designing DL for ECoG analysis.

Acknowledgements

Clinatec is a Laboratory of CEA-Leti at Grenoble and has statutory links with the University Hospital of Grenoble (CHUGA) and University Grenoble Alpes (UGA). This study was funded by CEA (recurrent funding) and the French Ministry of Health (Grant PHRC-15-15-0124), Institut Carnot, Fonds de Dotation Clinatec. Matthieu Martin was supported by the cross-disciplinary program on Numerical Simulation of CEA. Maciej Śliwowski was supported by the CEA NUMERICS program, which has received funding from European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant agreement No 800945 — NUMERICS — H2020-MSCA-COFUND-2017.

References

1. Benabid, A.L., Costecalde, T., Eliseyev, A., Charvet, G., Verney, A., Karakas, S., Foerster, M., Lambert, A., Morinière, B., Abroug, N., Schaeffer, M.C., Moly, A., Sauter-Starace, F., Ratel, D., Moro, C., Torres-Martinez, N., Langar, L., Oddoux, M., Polosan, M., Pezzani, S., Auboiroux, V., Aksenova, T., Mestais, C., Chabardes, S.: An exoskeleton controlled by an epidural wireless brain–machine interface in a tetraplegic patient: a proof-of-concept demonstration. The Lancet Neurology **18**(12), 1112–1122 (Dec 2019). https://doi.org/10.1016/S1474-4422(19)30321-7, number: 12 3
2. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Burstein, J., Doran, C., Solorio, T. (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers). pp. 4171–4186. Association for Computational Linguistics (2019). https://doi.org/10.18653/v1/n19-1423 1
3. Eliseyev, A., Auboiroux, V., Costecalde, T., Langar, L., Charvet, G., Mestais, C., Aksenova, T., Benabid, A.L.: Recursive Exponentially Weighted N-way Partial Least Squares Regression with Recursive-Validation of Hyper-Parameters in Brain-Computer Interface Applications. Scientific Reports **7**(1), 16281 (Dec 2017). https://doi.org/10.1038/s41598-017-16579-9, number: 1 4
4. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Pereira, F., Burges, C., Bottou, L., Weinberger, K. (eds.) Advances in Neural Information Processing Systems. vol. 25. Curran Associates, Inc. (2012), `https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf` 1, 2
5. Lawhern, V.J., Solon, A.J., Waytowich, N.R., Gordon, S.M., Hung, C.P., Lance, B.J.: EEGNet: a compact convolutional neural network for EEG-based brain–computer interfaces. Journal of Neural Engineering **15**(5), 056013 (jul 2018). https://doi.org/10.1088/1741-2552/aace8c 2
6. Lee, Y.E., Lee, S.H.: Eeg-transformer: Self-attention from transformer architecture for decoding eeg of imagined speech (2021). https://doi.org/10.48550/ARXIV.2112.09239 13
7. Li, J., Xiong, C., Hoi, S.C.: Learning from noisy data with robust representation learning. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9465–9474 (2021). https://doi.org/10.1109/ICCV48922.2021.00935 13
8. Liang, N., Bougrain, L.: Decoding Finger Flexion from Band-Specific ECoG Signals in Humans. Frontiers in Neuroscience **6** (2012). https://doi.org/10.3389/fnins.2012.00091 3
9. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations (2019), `https://openreview.net/forum?id=Bkg6RiCqY7` 8
10. Mestais, C.S., Charvet, G., Sauter-Starace, F., Foerster, M., Ratel, D., Benabid, A.L.: Wimagine: Wireless 64-channel ecog recording implant for long term clinical applications. IEEE Transactions on Neural Systems and Rehabilitation Engineering **23**(1), 10–21 (2015). https://doi.org/10.1109/TNSRE.2014.2333541 4
11. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L.,

Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems 32, pp. 8024–8035. Curran Associates, Inc. (2019), http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf 8

12. Roy, Y., Banville, H., Albuquerque, I., Gramfort, A., Falk, T.H., Faubert, J.: Deep learning-based electroencephalography analysis: a systematic review. Journal of Neural Engineering **16**(5), 051001 (aug 2019). https://doi.org/10.1088/1741-2552/ab260c 2
13. Santamaría-Vázquez, E., Martínez-Cagigal, V., Vaquerizo-Villar, F., Hornero, R.: Eeg-inception: A novel deep convolutional neural network for assistive erp-based brain-computer interfaces. IEEE Transactions on Neural Systems and Rehabilitation Engineering **28**(12), 2773–2782 (2020). https://doi.org/10.1109/TNSRE.2020.3048106 13
14. Schalk, G., Kubánek, J., Miller, K.J., Anderson, N.R., Leuthardt, E.C., Ojemann, J.G., Limbrick, D., Moran, D., Gerhardt, L.A., Wolpaw, J.R.: Decoding two-dimensional movement trajectories using electrocorticographic signals in humans. Journal of Neural Engineering **4**(3), 264–275 (Sep 2007). https://doi.org/10.1088/1741-2560/4/3/012, number: 3 3
15. Schirrmeister, R.T., Springenberg, J.T., Fiederer, L.D.J., Glasstetter, M., Eggensperger, K., Tangermann, M., Hutter, F., Burgard, W., Ball, T.: Deep learning with convolutional neural networks for eeg decoding and visualization. Human Brain Mapping **38**(11), 5391–5420 (2017). https://doi.org/https://doi.org/10.1002/hbm.23730 2, 6, 8
16. Śliwowski, M., Martin, M., Souloumiac, A., Blanchart, P., Aksenova, T.: Decoding ECoG signal into 3d hand translation using deep learning. Journal of Neural Engineering **19**(2), 026023 (mar 2022). https://doi.org/10.1088/1741-2552/ac5d69 3, 6, 7
17. Tabar, Y.R., Halici, U.: A novel deep learning approach for classification of EEG motor imagery signals. Journal of Neural Engineering **14**(1), 016003 (nov 2016). https://doi.org/10.1088/1741-2560/14/1/016003 2
18. Tietz, M., Fan, T.J., Nouri, D., Bossan, B., skorch Developers: skorch: A scikit-learn compatible neural network library that wraps PyTorch (Jul 2017), https://skorch.readthedocs.io/en/stable/ 8
19. Volkova, K., Lebedev, M.A., Kaplan, A., Ossadtchi, A.: Decoding movement from electrocorticographic activity: A review. Frontiers in Neuroinformatics **13** (2019). https://doi.org/10.3389/fninf.2019.00074 3
20. Xie, Z., Schwartz, O., Prasad, A.: Decoding of finger trajectory from ECoG using deep learning. Journal of Neural Engineering **15**(3), 036009 (Jun 2018). https://doi.org/10.1088/1741-2552/aa9dbe, number: 3 3, 14

5

Limitations and perspective

As our main contribution, we showed that deep learning could be used for decoding 3D hand translation from ECoG signal in a tetraplegic patient to provide higher prediction performance than multilinear models. The data analyzed in this thesis was recorded with a tetraplegic patient. The subject did not perform any actual hand movement during the experiments. This is opposite to the situation in which labels for training are inferred from real movements of non-disabled subjects (e.g., BCI IV competition ECoG dataset [Miller, 2019]). The ability to decode motor commands without having access to solid ground-truth labels is especially important for motor BCIs, as the primary goal of many current studies is to compensate for lost motor functions in the motor-impaired population [Shih et al., 2012]. Furthermore, we conclude that the proposed methods are suitable for BCI from the dataset size needs point of view. We showed that DL models have similar learning curve characteristic to the multilinear model and require 40-50 minutes of signal in the case of analyzed dataset. For real-life BCI applications, data efficiency is an important model characteristic allowing for maintaining reasonable duration of calibration. We also found signs of data quality increase throughout the experiments, indicating patient adaptation, influencing BCI performance. We also created a new Python infrastructure that can be used for clinical trial datasets analysis. It works in real-time with the current signal processing system. The code implementation can speed up future deep learning research in the context of the 'BCI and Tetraplegia' clinical trial.

The main limitation of the study is that all our analysis was performed using a prerecorded dataset with only one tetraplegic patient. Therefore, we are not able to estimate how these results generalize to the wider population. The observed models' behavior may be specific to the patient and experimental setup. For example, a mental task performed by the patient plays an important role in the BCI loop. The motor imagery patterns may vary between subjects [Saha and Baumert, 2019], thus requiring an adjusted signal processing pipeline. We were able to analyze the data only from one patient because of the clinical trial progress, which was also affected by the COVID-19 health emergency state. COVID-19 restrictions resulted in experiment cancellation, making it impossible to perform planned recordings with the tetraplegic subject. An additional challenge in the case of invasive BCI is concerns connected to the risk of implantation and further ethical questions. These factors limit access to the data that could be used for this analysis. However, additional experiments will be performed in the future as the clinical trial is planned to include a total of five tetraplegic patients in the study, potentially enabling analysis in the wider population.

All our experiments are fully computational, offline simulations, raising concern about the validity of the results in the case of real-life applications. In online experiments, models interact with patients, which may influence the results. However, comparing decoders in an online experiment is not feasible due to limited recording time and data variations, making it challenging to identify the source of performance change. On the other hand, offline experiments separate different factors and evaluate models on the same datasets in the same conditions. While the influence of the feedback on the models' performance remains unknown, offline simulation allows comparing different models in identical settings.

In the analysis, we only focused on predicting 3D hand translation movements, ignoring states classification and other simpler effectors' commands. We focused on this particular problem due to the clinical trial's need to increase decoding performance for hand translation. It can be very interesting to evaluate proposed methods on other tasks to show how the results generalize to other experimental paradigms and identify potential gains for other system components. This should be relatively straightforward as all the models use the same input features/signal and only differ due to the output type. Models evaluation on a different experimental paradigm may be possible in another clinical

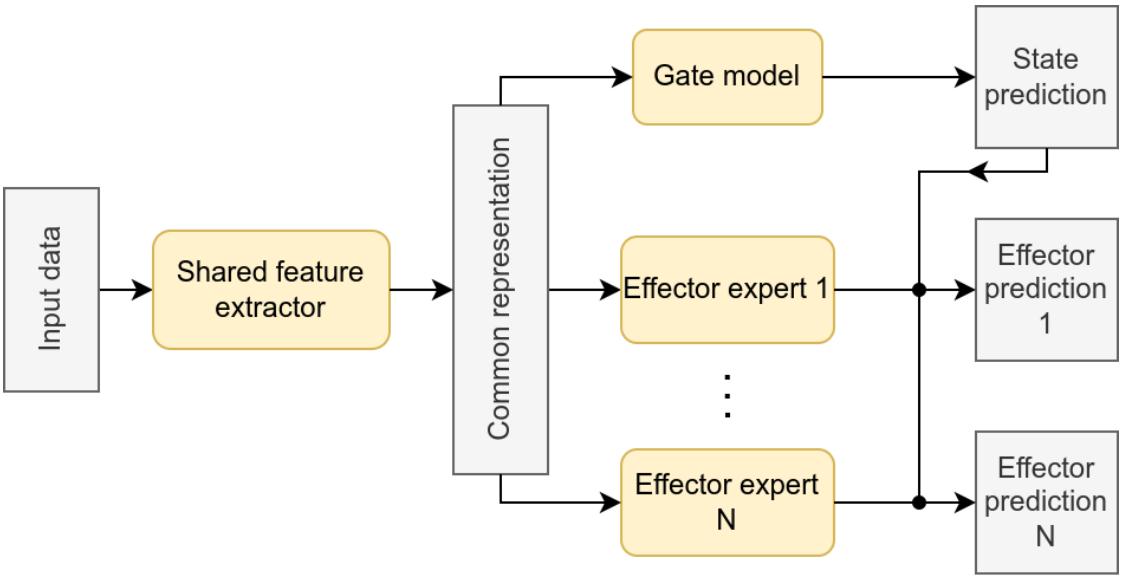


Figure 5.1: Schematic view of the potential architecture with shared feature extractor and multiple predictions heads, individual for each effector. Note that gate model can also be used before effectors to select appropriate expert which could limit computational load.

trial (STIMO-BSI, ClinicalTrials.gov identifier: NCT04632290), carried out jointly between EPFL and Clinatec, started recently aiming to restore lost motor functions, i.e., walking, in patients after spinal cord injury using ECoG-based BCI and spinal cord stimulation. In this ambitious project, ECoG signals are decoded into spinal cord stimulation commands to move patients' legs. It is possible to adjust the proposed DL models to the new experimental paradigm, which can potentially improve decoding performance similarly to the upper-limb decoding problem.

In this thesis, each model was trained to solve one particular problem, i.e., an independent model for each state (left hand translation, right hand translation). However, multi-task learning (simultaneously training a shared model to solve multiple tasks [Crawshaw, 2020]) can be a way to effectively increase dataset size by combining datasets of different effectors (e.g., translation, rotation, grasping, walking) in the case of BCI. This can also be useful when starting experiments for new tasks, so using a pretrained model can decrease calibration time and potentially improve decoding performance. General purpose models with shared parameters may extract complex representation more efficiently by integrating knowledge learned from related tasks [Crawshaw, 2020]. Additionally, sharing parameters in the first layers would allow for computations speed up required in the case of BCI operating with higher frequency. The concept of a gate model

and multiple experts controlling various effectors could be implemented with a shared general-purpose feature extractor followed by individual prediction heads, one for state prediction and each effector (see figure 5.1). On the other hand, multi-task learning may be challenging due to conflicting needs of different tasks and may require special task selection for joint training and several architecture/approach adjustments [Crawshaw, 2020]. Thus, offline analysis on the clinical trial dataset could be performed to assess possible improvements from using a general-purpose model.

Deep learning covers a wide variety of architectures and decoding strategies that can be potentially applied to this problem. Moreover, it is a fast-developing field with new approaches proposed every day that may be suitable for this application. In this thesis, we evaluated only a limited set of DL approaches. We can assume that incorporating further deep learning advancements will increase decoding performance. In the next steps, more DL architectures could be tested, for example, models that use attention mechanisms [Eldele et al., 2021; Vaswani et al., 2017], inception blocks [Santamaría-Vázquez et al., 2020], etc.

Besides different DL architectures, more types of neural networks could be efficient in this task. For example, bayesian neural networks (BNNs) are data efficient and provide uncertainty estimates [Jospin et al., 2022] which could be important information allowing for creating real-life BCI systems resistant to the noise in labels and allowing for high precision control thanks to the ability to deal with the variation of data quality. Another potential neural network framework could be spiking neural networks (SNNs) [Maass, 1997], which are more closely inspired by biological neurons than the methods evaluated in this thesis. SNNs were also successfully applied to EEG data [Demin and Nekhaev, 2018; Faghihi et al., 2022; Virgilio G. et al., 2020], in a particular case for decoding hand movements kinematics [Kumarasinghe et al., 2021], which makes SNN a reasonable choice for further ECoG data analysis. In the last decade, methods based on Riemannian geometry were shown effective for EEG multichannel temporal data classification using signal covariance matrices [Barachant et al., 2013]. In our case, ECoG data is also a multichannel and temporal signal. However, recording electrodes are placed much closer to each other than EEG, increasing the correlation between channels. Electrodes are also placed closer to the cortex, increasing high-frequency components' amplitude. It can be interesting to verify how Riemannian geometry can work with covariance matrices of ECoG

signal to predict continuous hand trajectories, especially considering that first attempts for classification were made [Larzabal et al., 2021] and new Riemannian DL methods for analyzing SPD matrices were proposed recently [Brooks et al., 2019; Huang and Gool, 2017] that were applied to EEG data [Ju et al., 2020; Wei et al., 2022].

Our analysis revealed that regularization methods play an important role in obtaining efficient DL models. In our approach, we used dropout, L2 regularization, early stopping, and batch normalization. These are among the popular regularization choices in DL research. However, more regularization methods can be useful in alleviating overfitting. For example, in the proposed approach, we did not use any data augmentation method, which is common in computer vision and recently also in EEG analysis [Rommel et al., 2022]. Data augmentation is a way to regularize the model and artificially increase dataset size. Thus, it can be profitable in this application with the potential to improve decoding accuracy. Another strategy to regularize the model can be, for example, GradAug [Yang et al., 2020], which showed effectiveness in the case of computer vision datasets.

Deep learning models typically do not support continual/online learning in which models are updated on chunks of newly recorded data. Usually, observations from the whole dataset must be used for every update to avoid catastrophic forgetting [Parisi et al., 2019]. Our experiments (forward increase in chapter 3) provide preliminary model training results after recording new sessions. This can be a starting point for a more detailed analysis of model behavior in the case of updating the model in a pseudo-online scenario, simulating a real-life experiment and updating the model on chunks of data, and finally, in an online experiment with a patient.

In some of our experiments, we used a fixed set of models' hyperparameters without performing any hyperparameters search. To estimate the models' capabilities more accurately, some hyperparameters, e.g., the learning rate, weight decay, size of the network, can be optimized specifically for different tasks and bigger datasets. This can potentially provide an improvement for datasets consisting of many recording sessions (proposed models were created and evaluated on 40 minutes of data in the chapter 2). However, hyperparameter search is time-consuming, especially for big datasets and end-to-end models. In this case, the experiments should be performed on a powerful server and focus on a particular

dataset, for example, recorded with the patient shortly before online evaluation.

Our results did not show particular improvement from using end-to-end optimization to train DL models. This is counterintuitive as, in theory, end-to-end training increases the model's capabilities to learn complicated patterns by removing the limitation of hand-crafted feature extraction. The reason for this behavior is not fully explained. While our analysis showed that dataset size and noise in the labels might have some minor influence, an extended study should be performed to understand this behavior and address particular issues preventing end-to-end models from achieving higher performance. Generative adversarial networks (GANs) [Goodfellow et al., 2014] and, more generally, generative models can help understand this behavior by providing control over the data generation process, opposite to real-life datasets where data understanding is limited. This could lead to the identification of patterns that could not be detected by the proposed architecture. Learned insights could potentially be used to improve DL architecture and training strategy.

Considering a broader perspective of invasive BCI progress, ECoG-based BCI research is affected by several factors that are general field problems. First, open-access datasets are not easily accessible. This makes comparing models and the development of new methods difficult. Currently, a dataset most similar to the clinical trial dataset analyzed here is the BCI IV competition dataset with finger flexion recording in non-disabled patients [Miller, 2019]. In this experiment, fingers' trajectories were recorded using a glove, so ground-truth labels represent actual fingers' movement. Besides the BCI IV competition dataset, most datasets provide only discrete targets or data recorded with monkeys. The datasets created in clinical trials are rarely accessible due to legal/ethical/strategic reasons. As ML describes a family of data-driven models, methodological advancements are hard to achieve without proper data and validation on multiple datasets. Thus sharing more datasets with the community is very important and may increase the number of studies analyzing BCI data as well as put more attention on developing machine learning methods that could be better evaluated and easier to use for the whole community.

One of the major steps in invasive BCI signals analysis could be setting proper benchmarks on multiple datasets. The community could distinguish between two types of contributions, i.e., case studies showing the effectiveness of a method in particular conditions and studies focusing on a machine learning

advancement that should be evaluated in a broader context of different clinical trials and various datasets. The latter could be more beneficial for long-term ML model development, bringing conclusions to the community that can generalize better to new data. This can be done by creating competitions, well-known in the ML community, less popular for BCI, but already greatly impacting the research directions [Blankertz et al., 2004; Tangermann et al., 2012; Wei et al., 2022]. Competitions can be a way to unify data processing pipelines between groups and introduce a better-structured model evaluation process. Easy access to the data and increased popularity of BCI can significantly increase the quality of research, allow for more in-depth analysis of specific data processing steps, and be profitable to society.

Current machine learning research is primarily conducted using Python, which provides a large ML ecosystem of packages with enormous capabilities. It is based on an open-source paradigm, in which the whole community can contribute to creating a greater project. This paradigm also influenced how research is done in machine learning, i.e., many studies share implementations of articles, significantly increasing reproducibility and, what is also essential, reducing the redundancy between different studies. In the BCI community, a shift toward Python-based systems can be visible. However, closed-source programming languages are still dominant in invasive BCI applications, increasing costs and limiting transparency and reproducibility. This also makes it difficult to unify data processing pipelines due to a lack of a general, shared between groups scheme. Generally, a shift toward open-source software can start a stronger collaboration in the invasive BCI field and ease research reproducibility in the future.

Bibliography

Alexandre Barachant, Stéphane Bonnet, Marco Congedo, and Christian Jutten. Classification of covariance matrices using a riemannian-based kernel for bci applications. *Neurocomputing*, 112:172–178, 2013. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2012.12.039. URL https://www.sciencedirect.com/science/article/pii/S0925231213001574. Advances in artificial neural networks, machine learning, and computational intelligence.

B. Blankertz, K.-R. Muller, G. Curio, T.M. Vaughan, G. Schalk, J.R. Wolpaw, A. Schlogl, C. Neuper, G. Pfurtscheller, T. Hinterberger, M. Schroder, and N. Birbaumer. The bci competition 2003: progress and perspectives in detection and discrimination of eeg single trials. *IEEE Transactions on Biomedical Engineering*, 51(6):1044–1051, 2004. doi: 10.1109/TBME.2004.826692.

Daniel Brooks, Olivier Schwander, Frederic Barbaresco, Jean-Yves Schneider, and Matthieu Cord. Riemannian batch normalization for spd neural networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/6e69ebbfad976d4637bb4b39de261bf7-Paper.pdf.

Michael Crawshaw. Multi-task learning with deep neural networks: A survey. *ArXiv*, abs/2009.09796, 2020.

Vyacheslav Demin and Dmitry Nekhaev. Recurrent spiking neural network learning based on a competitive maximization of neuronal activity. *Frontiers in Neuroinformatics*, 12, 2018. ISSN 1662-5196. doi: 10.3389/fninf.2018.00079. URL https://www.frontiersin.org/articles/10.3389/fninf.2018.00079.

Emadeldeen Eldele, Zhenghua Chen, Chengyu Liu, Min Wu, Chee-Keong Kwoh, Xiaoli Li, and Cuntai Guan. An attention-based deep learning approach for sleep stage classification with single-channel eeg. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 29:809–818, 2021. doi: 10.1109/TNSRE.2021.3076234.

Faramarz Faghihi, Siqi Cai, and Ahmed A. Moustafa. A neuroscience-inspired spiking neural network for eeg-based auditory spatial attention detection. *Neural Networks*, 152:555–565, 2022. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2022.05.003. URL https://www.sciencedirect.com/science/article/pii/S0893608022001757.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and

K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf.

Zhiwu Huang and Luc Van Gool. A riemannian network for spd matrix learning. *ArXiv*, abs/1608.04233, 2017.

Laurent Valentin Jospin, Hamid Laga, Farid Boussaid, Wray Buntine, and Mohammed Bennamoun. Hands-on bayesian neural networks—a tutorial for deep learning users. *IEEE Computational Intelligence Magazine*, 17(2):29–48, 2022. doi: 10.1109/MCI.2022.3155327.

Ce Ju, Dashan Gao, Ravikiran Mane, Ben Tan, Yang Liu, and Cuntai Guan. Federated transfer learning for eeg signal classification. *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 3040–3045, 2020.

Kaushalya Kumarasinghe, Nikola K. Kasabov, and Denise Taylor. Brain-inspired spiking neural networks for decoding and understanding muscle activity and kinematics from electroencephalography signals during hand movements. *Scientific Reports*, 11, 2021.

Christelle Larzabal, Vincent Auboiroux, Serpil Karakas, Guillaume Charvet, Alim-Louis Benabid, Stephan Chabardes, Thomas Costecalde, and Stéphane Bonnet. The riemannian spatial pattern method: mapping and clustering movement imagery using riemannian geometry. *Journal of Neural Engineering*, 18(5):056014, apr 2021. doi: 10.1088/1741-2552/abf291. URL https://doi.org/10.1088/1741-2552/abf291.

Wolfgang Maass. Networks of spiking neurons: The third generation of neural network models. *Neural Networks*, 10(9):1659–1671, 1997. ISSN 0893-6080. doi: https://doi.org/10.1016/S0893-6080(97)00011-7. URL https://www.sciencedirect.com/science/article/pii/S0893608097000117.

Kai J. Miller. A library of human electrocorticographic data and analyses. *Nature Human Behaviour*, pages 1–11, 2019.

German I. Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2019.01.012. URL https://www.sciencedirect.com/science/article/pii/S0893608019300231.

C'edric Rommel, Joseph Paillard, Thomas Moreau, and Alexandre Gramfort. Data augmentation for learning predictive models on eeg: a systematic comparison. *ArXiv*, abs/2206.14483, 2022.

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

Eduardo Santamaría-Vázquez, Víctor Martínez-Cagigal, Fernando Vaquerizo-Villar, and Roberto Hornero. Eeg-inception: A novel deep convolutional neural network for assistive erp-based brain-computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28(12):2773–2782, 2020. doi: 10.1109/TNSRE.2020.3048106.

Jerry J. Shih, Dean J. Krusienski, and Jonathan R. Wolpaw. Brain-computer interfaces in medicine. *Mayo Clinic proceedings*, 87 3:268–79, 2012.

Michael Tangermann, Klaus-Robert Müller, Ad Aertsen, Niels Birbaumer, Christoph Braun, Clemens Brunner, Robert Leeb, Carsten Mehring, Kai J. Miller, Gernot R. Müller-Putz, Guido Nolte, Gert Pfurtscheller, Hubert Preissl, Gerwin Schalk, Alois Schlögl, Carmen Vidaurre, Stephan Waldert, and Benjamin Blankertz. Review of the bci competition iv. *Frontiers in Neuroscience*, 6, 2012.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Carlos D. Virgilio G., Juan H. Sossa A., Javier M. Antelis, and Luis E. Falcón. Spiking neural networks applied to the classification of motor tasks in eeg signals. *Neural Networks*, 122:130–143, 2020. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2019.09.037. URL https://www.sciencedirect.com/science/article/pii/S0893608019303193.

Xiaoxi Wei, A. Aldo Faisal, Moritz Grosse-Wentrup, Alexandre Gramfort, Sylvain Chevallier, Vinay Jayaram, Camille Jeunet, Stylianos Bakas, Siegfried Ludwig, Konstantinos Barmpas, Mehdi Bahri, Yannis Panagakis, Nikolaos Laskaris, Dimitrios A. Adamos, Stefanos Zafeiriou, William C. Duong, Stephen M. Gordon, Vernon J. Lawhern, Maciej Śliwowski, Vincent Rouanne, and Piotr Tempczyk. 2021 beetl competition: Advancing transfer learning for subject independence & heterogenous eeg data sets. In Douwe Kiela, Marco Ciccone, and Barbara Caputo, editors, *Proceedings of the NeurIPS 2021 Competitions and Demonstrations Track*, volume 176 of *Proceedings of Machine Learning Research*, pages 205–219. PMLR, 06–14 Dec 2022. URL https://proceedings.mlr.press/v176/wei22a.html.

Taojiannan Yang, Sijie Zhu, and Chen Chen. Gradaug: A new regularization method for deep neural networks. *Advances in Neural Information Processing Systems*, 33, 2020.

Résumé en français

Chapitre 1 - Introduction

Interfaces cerveau-machine (ICMs) Une interface cerveau-machine (ICM) est un dispositif qui permet une connexion directe entre le cerveau et un ordinateur sans nécessiter d'activation musculaire. Dans cette thèse, nous nous concentrons sur une ICM motrice utilisée pour restaurer/compenser les fonctionnalités perdues en fournissant un contrôle sur divers effecteurs, comme par exemple un curseur d'ordinateur [Wolpaw and McFarland, 2004], un bras robotique [Hochberg et al., 2012], un exosquelette [Benabid et al., 2019], ou un stimulation électrique [Ajiboye et al., 2017; Lorach et al., 2022]. Dans le cas des ICM motrices, de nombreux patients perdent leurs fonctions motrices mais conservent un cortex entièrement fonctionnel, ce qui leur permet de générer des schémas d'imagerie motrice distincts. Les ICM motrices pourraient potentiellement traiter la paralysie en permettant le contrôle volontaire de prothèses [Volkova et al., 2019], la recherche dans ce domaine se développe donc rapidement.

Généralement, le système ICM moteur est constitué de plusieurs éléments qui communiquent entre eux et constituent un pipeline de traitement des données. L'élément prérequis le plus important d'un système d'ICM moteur est évidemment le sujet humain qui effectue une tâche mentale induisant des changements dans l'activité cérébrale. La présence d'un humain dans la boucle de l'ICM est un facteur essentiel qui influence presque tous les aspects du système. Ensuite, les modifications de l'activité cérébrale causées par la modulation de l'activité neuronale dans le cortex sont saisies à l'aide d'un dispositif d'enregistrement, par exemple l'électroencéphalographie (EEG), l'électrocorticographie (ECoG) ou les réseaux de microélectrodes intracorticaux (MEA). Bien que les dispositifs d'enregistrement fournissent des signaux de haute qualité, les données ne peuvent pas être directement déchiffrées en commandes compréhensibles. Par conséquent, les systèmes ICM typiques nécessitent une étape de traitement du signal (basée sur l'apprentissage automatique (ML)) qui crée une correspondance entre les signaux cérébraux et les instructions pour les effecteurs. Enfin, les commandes décodées représentant l'intention du participant sont exécutées

par un effecteur, par exemple un bras robotique [Collinger et al., 2013; Wodlinger et al., 2014] ou un exosquelette [Benabid et al., 2019].

Essai clinique Cette thèse a été réalisée dans le cadre de l'essai clinique "BCI et tétraplégie" (ClinicalTrials.gov identifier : NCT02550522) mené à Clinatec. L'objectif principal de cet essai clinique est de réaliser la preuve de concept permettant à des sujets tétraplégiques d'interagir avec leur environnement en contrôlant des effecteurs ICM complexes, tels qu'un exosquelette à 4 membres [Eliseyev et al., 2014]. Ceci est réalisé en décodant les intentions des sujets tétraplégiques à partir du signal ECoG enregistré à la surface du cerveau. Le système ICM proposé est conçu pour une application à long terme afin de fournir le contrôle de multiples effecteurs à des sujets handicapés en utilisant la modulation de l'activité cérébrale. Dans le cadre du projet, des développements substantiels de tous les composants de l'ICM ont été réalisés, c'est-à-dire la création d'implants ECoG adaptés aux enregistrements chroniques [Mestais et al., 2015], la construction d'une plateforme de traitement du signal pour le décodage des signaux en temps réel [Eliseyev et al., 2017b], et enfin la conception d'un ensemble d'effecteurs comprenant à la fois un exosquelette robotique [Morinière et al., 2015] et des exécuteurs de commandes virtuels. En conséquence, le projet peut constituer une étape vers l'amélioration de la qualité de vie des sujets tétraplégiques en leur fournissant des outils qui peuvent compenser une partie de la fonction motrice perdue [Benabid et al., 2019].

L'apprentissage profond (DL) L'apprentissage profond (DL) a prouvé son efficacité dans diverses tâches, montrant des capacités accrues par rapport aux méthodes linéaires. Les modèles DL sont construits à partir d'une série de transformations non linéaires appelées couches. Cela permet de créer une représentation complexe des données d'entrée. Les méthodes DL semblent être un bon candidat pour fournir des performances de décodage plus élevées et remplacer le modèle multilinéaire actuellement utilisé.

L'objectif principal de la thèse était de chercher à améliorer les performances en examinant si les modèles basés sur DL sont adaptés au décodage de la translation de la main en 3D dans le contexte de l'ICM basée sur l'ECoG. Dans le cas d'une amélioration des performances, les modèles DL avec des capacités plus importantes pourraient remplacer les modèles multilinéaires utilisés dans l'essai clinique. Le deuxième objectif de la thèse était d'étudier la taille optimale du jeu

de données d'entraînement pour les modèles multilinéaires et les modèles DL. La relation entre la taille du jeu de données et la performance de décodage est particulièrement importante dans le cas des ICM car elle définit la durée des expériences de calibration.

Chapitre 2 - Décodage du signal ECoG en une translation en 3D de la main avec de l'apprentissage profond

La plupart des décodeurs de signaux ECoG utilisés pour prédire les mouvements continus de la main sont des modèles linéaires. Ces modèles ont une capacité de représentation limitée et peuvent échouer à capturer la relation entre les caractéristiques du signal ECoG et les mouvements continus de la main. Les modèles d'apprentissage profond (DL), qui sont à la pointe du progrès dans de nombreux problèmes, pourraient être une solution pour mieux capturer cette relation. Dans ce chapitre, nous avons testé plusieurs architectures basées sur l'apprentissage profond pour prédire la translation continue de la main en 3D à l'aide de caractéristiques temps-fréquence extraites des signaux ECoG. Le jeu de données utilisé dans l'analyse fait partie d'un essai clinique à long terme (identifiant ClinicalTrials.gov : NCT02550522) et a été acquis lors d'une expérience en boucle fermée avec un sujet tétraplégique. Les architectures proposées comprennent des perceptrons multicouches (MLP), des réseaux neuronaux convolutionnels (CNN) et des réseaux long short-term memory (LSTM). La précision des modèles DL et multilinéaires a été comparée hors ligne à l'aide de la similitude en cosinus.

Nos résultats montrent que les architectures basées sur les CNN sont plus performantes que le modèle multilinéaire actuel. La meilleure architecture exploite la corrélation spatiale entre les électrodes voisines avec CNN et bénéficie du caractère séquentiel de la trajectoire de la main souhaitée en utilisant LSTMs. Dans l'ensemble, l'approche DL a obtenu jusqu'à 60 % d'augmentation de la similarité cosinus moyenne par rapport au modèle multilinéaire, de 0.189 à 0.302 et de 0.157 à 0.249 pour la main gauche et la main droite, respectivement. Ce chapitre montre que les modèles basés sur le DL pourraient augmenter la précision des systèmes ICM dans le cas de la prédiction de la translation de la main en 3D chez un sujet tétraplégique.

Chapitre 3 - Impact de la taille du jeu de données et de l'utilisation à long terme d'une ICM basée sur l'ECoG sur la performance des décodeurs d'apprentissage profond

Dans la recherche sur les ICM, l'enregistrement des données est long et coûteux, ce qui limite l'accès aux grands jeux de données. Cela peut influer sur les performances du système d'ICM, car les méthodes d'apprentissage automatique dépendent fortement de la taille du jeu de données d'apprentissage. Des questions importantes se posent : en tenant compte des caractéristiques des signaux neuronaux (par exemple, la non-stationnarité), peut-on obtenir des performances de décodage plus élevées avec davantage de données pour entraîner les décodeurs ? Quelles sont les perspectives d'amélioration dans le temps dans le cas d'études ICM à long terme ? Dans ce chapitre, nous avons étudié l'impact des enregistrements à long terme sur le décodage de l'imagerie motrice sous deux angles principaux : les exigences du modèle concernant la taille du jeu de données et le potentiel d'adaptation du patient.

Nous avons évalué le modèle multilinéaire et deux modèles d'apprentissage profond (DL) sur un jeu de données de l'essai clinique « BCI & Tetraplegia » (NCT02550522) contenant 43 sessions d'enregistrements ECoG réalisées avec un patient tétraplégique. Au cours de l'expérience, un participant exécutait une translation virtuelle de la main en 3D en utilisant de l'imagerie motrice. Nous avons conçu plusieurs simulations dans lesquelles les jeux de données d'entraînement ont été augmentés ou translatés afin d'étudier la relation entre les performances des modèles et différents facteurs influençant les enregistrements.

Pour tous les décodeurs testés, notre analyse a montré que l'ajout de données supplémentaires au jeu de données d'entraînement peut ne pas augmenter instantanément les performances pour les jeux de données contenant déjà 40 minutes de signal. Les décodeurs DL ont montré des exigences similaires concernant la taille du jeu de données par rapport au modèle multilinéaire tout en démontrant une meilleure performance de décodage. De plus, des performances de décodage élevées ont été obtenues avec des jeux de données relativement petits enregistrés plus tard dans l'expérience, ce qui suggère une amélioration des modèles d'imagerie motrice et une adaptation du patient au cours de l'expérience à long terme. Enfin, nous avons proposé des incorporations UMAP et de la dimensionnalité intrinsèque locale comme moyen de visualiser les données et d'évaluer potentiellement la qualité des données. Dans ce chapitre, nous avons montré que le décodage basé sur le DL est une approche ICM potentielle qui

peut être appliquée efficacement avec des jeux de données de taille réelle. En outre, la co-adaptation patient-décodeur est un facteur important à prendre en compte dans les ICM cliniques à long terme.

Chapitre 4 - Apprentissage profond pour l'interface cerveau-machine ECoG : caractéristiques de bout en bout contre caractéristiques créées à la main

Les modèles d'apprentissage profond (DL) sont devenus couramment utilisés dans le traitement des signaux cérébraux. Cependant, le gain de performance obtenu par l'utilisation de modèles d'apprentissage profond de bout en bout par rapport aux approches ML classiques est généralement significatif mais modéré, au prix d'une augmentation de la charge de calcul et d'une détérioration de l'explicabilité. L'idée centrale des approches d'apprentissage profond est de faire évoluer les performances avec des jeux de données plus importants. Cependant, les signaux cérébraux sont des données temporelles avec un faible rapport signal/bruit, des étiquettes incertaines et des données non stationnaires dans le temps. Ces facteurs peuvent influencer le processus d'apprentissage et ralentir l'amélioration des performances des modèles. L'influence de ces facteurs peut différer entre un modèle DL de bout en bout et un modèle utilisant des caractéristiques créées à la main.

Dans ce chapitre, nous avons comparé les modèles qui utilisent le signal ECoG brut à ceux qui utilisent les caractéristiques temps-fréquence pour le décodage de l'imagerie motrice des ICM. En outre, nous cherchons à savoir si la taille actuelle du jeu de données est une limitation plus forte pour tous les modèles. Enfin, les filtres obtenus ont été comparés pour identifier les différences entre les caractéristiques créées à la main et les filtres optimisés par rétropropagation. Pour comparer l'efficacité des deux stratégies, nous avons utilisé un perceptron multicouche et un mélange de couches convolutionnelles et LSTM qui se sont déjà avérés efficaces dans cette tâche. L'analyse a été réalisée sur la base de données d'essais cliniques à long terme (près de 600 minutes d'enregistrements) d'un patient tétraplégique exécutant des tâches d'imagerie motrice pour la translation de la main en 3D.

Pour un jeu de données donné, les résultats ont montré que l'entraînement de bout en bout n'est pas nécessairement plus performant que le modèle basé sur les caractéristiques élaborées à la main. L'écart de performance est réduit avec des jeux de données plus importants, mais compte tenu de la charge de calcul accrue, l'entraînement de bout en bout peut ne pas être rentable pour cette ap-

plication particulière.

Chapitre 5 - Limitations et perspectives

Notre principale contribution a consisté à montrer que l'apprentissage profond pouvait être utilisé pour décoder la translation de la main en 3D à partir du signal ECoG chez un patient tétraplégique. Il s'agit d'une étape importante vers l'amélioration des systèmes de BCI basés sur l'ECoG, car elle peut fournir aux patients un niveau de contrôle plus élevé. Comme nous avons analysé les données enregistrées chez un sujet tétraplégique, les modèles n'ont pas eu accès aux trajectoires réelles de la main pendant l'entraînement et ont été optimisés uniquement sur la base des instructions fournies au patient. Ceci est particulièrement important pour les BCI motrices, car l'objectif principal de nombreuses études actuelles est de compenser la perte des fonctions motrices chez les personnes atteintes de déficience motrice. En outre, nous concluons que les méthodes proposées sont adaptées aux BCI du point de vue de la taille des jeux de données.

Dans notre analyse, nous nous sommes concentrés sur un seul sujet tétraplégique. C'est la principale limite de l'étude, car nous ne sommes pas en mesure d'estimer comment ces résultats se généralisent à une population plus large. En outre, nous avons analysé des jeux de données préenregistrées dans des expériences de calcul hors ligne, de sorte qu'une évaluation en ligne devrait être effectuée pour confirmer nos résultats dans l'expérience réelle. Heureusement, à l'avenir, d'autres expériences sont prévues qui pourraient évaluer les conclusions présentées dans cette thèse. De plus, certaines de nos expériences de calcul étaient limitées à des hyperparamètres sélectionnés manuellement, donc dans les études suivantes, une recherche d'hyperparamètres qui prend du temps pourrait être effectuée pour estimer plus précisément les capacités des modèles. Enfin, d'autres architectures d'apprentissage profond pourraient être étudiées, par exemple, les mécanismes d'attention et les autoencodeurs, qui pourraient fournir des performances de décodage plus élevées.

Bibliography

A Bolu Ajiboye, Francis R Willett, Daniel R Young, William D Memberg, Brian A Murphy, Jonathan P Miller, Benjamin L Walter, Jennifer A Sweet, Harry A Hoyen, Michael W Keith, P Hunter Peckham, John D Simeral, John P Donoghue, Leigh R Hochberg, and Robert F Kirsch. Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration. *The Lancet*, 389(10081):1821–1830, May 2017. ISSN 01406736. doi: 10.1016/S0140-6736(17)30601-3. URL https://linkinghub.elsevier.com/retrieve/pii/S0140673617306013. Number: 10081.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2015.

Tonio Ball, Markus Kern, Isabella Mutschler, Ad Aertsen, and Andreas Schulze-Bonhage. Signal quality of simultaneously recorded invasive and non-invasive eeg. *NeuroImage*, 46(3):708–716, 2009. ISSN 1053-8119. doi: https://doi.org/10.1016/j.neuroimage.2009.02.028. URL https://www.sciencedirect.com/science/article/pii/S1053811909001827.

Alexandre Barachant, Stéphane Bonnet, Marco Congedo, and Christian Jutten. Classification of covariance matrices using a riemannian-based kernel for bci applications. *Neurocomputing*, 112:172–178, 2013. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2012.12.039. URL https://www.sciencedirect.com/science/article/pii/S0925231213001574. Advances in artificial neural networks, machine learning, and computational intelligence.

Ali Bashashati, Mehrdad Fatourechi, Rabab K Ward, and Gary E Birch. A survey of signal processing algorithms in brain–computer interfaces based on electrical brain signals. *Journal of Neural Engineering*, 4(2):R32–R57, mar 2007. doi: 10.1088/1741-2560/4/2/r03. URL https://doi.org/10.1088/1741-2560/4/2/r03.

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. *The Lancet Neurology*, 18(12):1112–1122, December 2019. ISSN 14744422. doi: 10.1016/S1474-4422(19)30321-7. URL https://linkinghub.elsevier.com/retrieve/pii/S1474442219303217. Number: 12.

Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994. doi: 10.1109/72.279181.

Nils Bjorck, Carla P Gomes, Bart Selman, and Kilian Q Weinberger. Understanding batch normalization. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/36072923bfc3cf47745d704feb489480-Paper.pdf.

B. Blankertz, K.-R. Muller, G. Curio, T.M. Vaughan, G. Schalk, J.R. Wolpaw, A. Schlogl, C. Neuper, G. Pfurtscheller, T. Hinterberger, M. Schroder, and N. Birbaumer. The bci competition 2003: progress and perspectives in detection and discrimination of eeg single trials. *IEEE Transactions on Biomedical Engineering*, 51(6):1044–1051, 2004. doi: 10.1109/TBME.2004.826692.

Y-Lan Boureau, Jean Ponce, and Yann LeCun. A theoretical analysis of feature pooling in visual recognition. In *ICML*, 2010.

Denny Britz, Anna Goldie, Minh-Thang Luong, and Quoc V. Le. Massive exploration of neural machine translation architectures. *ArXiv*, abs/1703.03906, 2017.

Daniel Brooks, Olivier Schwander, Frederic Barbaresco, Jean-Yves Schneider, and Matthieu Cord. Riemannian batch normalization for spd neural networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/6e69ebbfad976d4637bb4b39de261bf7-Paper.pdf.

Clemens Brunner, Niels Birbaumer, Benjamin Blankertz, Christoph Guger, Andrea Kübler, Donatella Mattia, José del R. Millán, Felip Miralles, Anton Nijholt, Eloy Opisso, Nick Ramsey, Patric Salomon, and Gernot R. Müller-Putz. Bnci horizon 2020: towards a roadmap for the bci community. *Brain-Computer Interfaces*, 2(1):1–10, 2015. doi: 10.1080/2326263X.2015.1008956. URL https://doi.org/10.1080/2326263X.2015.1008956.

Anthony N. Burkitt. A review of the integrate-and-fire neuron model: I. homogeneous synaptic input. *Biological Cybernetics*, 95:1–19, 2006.

Borís Burle, Laure Spieser, Clémence Roger, Laurence Casini, Thierry Hasbroucq, and Franck Vidal. Spatial and temporal resolutions of EEG: Is it really black and white? A scalp current density view. *International Journal of Psychophysiology*, 97(3):210–220, September 2015. ISSN 0167-8760. doi: 10.1016/j.ijpsycho.2015.05.004. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4548479/.

György Buzsáki, Costas A. Anastassiou, and Christof Koch. The origin of extracellular fields and currents — EEG, ECoG, LFP and spikes. *Nature Reviews Neuroscience*, 13(6):407–420, June 2012. ISSN 1471-0048. doi: 10.1038/nrn3241.

URL https://www.nature.com/articles/nrn3241. Number: 6 Publisher: Nature Publishing Group.

Grégoire Cattan. The use of brain–computer interfaces in games is not ready for the general public. *Frontiers in Computer Science*, 3, 2021. ISSN 2624-9898. doi: 10.3389/fcomp.2021.628773. URL https://www.frontiersin.org/articles/10.3389/fcomp.2021.628773.

Anna Choromańska, Mikael Henaff, Michaël Mathieu, Gérard Ben Arous, and Yann LeCun. The loss surfaces of multilayer networks. In *AISTATS*, 2015.

Junyoung Chung, Çaglar Gülçehre, Kyunghyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *ArXiv*, abs/1412.3555, 2014.

Junyoung Chung, Çaglar Gülçehre, Kyunghyun Cho, and Yoshua Bengio. Gated feedback recurrent neural networks. In *ICML*, 2015.

Giulia Cisotto, Alessio Zanga, Joanna Chlebus, Italo Zoppis, Sara Manzoni, and Urszula Markowska-Kaczmar. Comparison of attention-based deep learning models for eeg classification. *ArXiv*, abs/2012.01074, 2020.

Jennifer L. Collinger, Brian Wodlinger, John E. Downey, Wei Wang, Elizabeth C. Tyler-Kabara, Douglas J. Weber, Angus JC McMorland, Meel Velliste, Michael L. Boninger, and Andrew B. Schwartz. High-performance neuroprosthetic control by an individual with tetraplegia. *The Lancet*, 381:557–564, 2013.

Michael Crawshaw. Multi-task learning with deep neural networks: A survey. *ArXiv*, abs/2009.09796, 2020.

Vyacheslav Demin and Dmitry Nekhaev. Recurrent spiking neural network learning based on a competitive maximization of neuronal activity. *Frontiers in Neuroinformatics*, 12, 2018. ISSN 1662-5196. doi: 10.3389/fninf.2018.00079. URL https://www.frontiersin.org/articles/10.3389/fninf.2018.00079.

Marian Dovgialo, Anna Chabuda, Anna Duszyk, Magdalena Zieleniewska, Marcin Pietrzak, Piotr Różański, and Piotr Durka. Assessment of statistically significant command-following in pediatric patients with disorders of consciousness, based on visual, auditory and tactile event-related potentials. *International Journal of Neural Systems*, 29(03):1850048, 2019. doi: 10.1142/S012906571850048X. URL https://doi.org/10.1142/S012906571850048X. PMID: 30606086.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(61):2121–2159, 2011. URL http://jmlr.org/papers/v12/duchi11a.html.

Emadeldeen Eldele, Zhenghua Chen, Chengyu Liu, Min Wu, Chee-Keong Kwoh, Xiaoli Li, and Cuntai Guan. An attention-based deep learning approach for sleep stage classification with single-channel eeg. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 29:809–818, 2021. doi: 10.1109/TNSRE.2021.3076234.

Andrey Eliseyev, Corinne Mestais, Guillaume Charvet, Fabien Sauter, Neil Abroug, Nana Arizumi, Serpil Cokgungor, Thomas Costecalde, Michael Foerster, Louis Korczowski, Boris Morinière, Jean Porcherot, Jérémy Pradal, David Ratel, Nicolas Tarrin, Napoleon Torres-Martinez, Alexandre Verney, Tetiana Aksenova, and Alim-Louis Benabid. Clinatec® bci platform based on the ecog-recording implant wimagine® and the innovative signal-processing: Preclinical results. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 1222–1225, 2014. doi: 10.1109/EMBC.2014.6943817.

Andrey Eliseyev, Vincent Auboiroux, Thomas Costecalde, Lilia Langar, Guillaume Charvet, Corinne Mestais, Tetiana Aksenova, and Alim-Louis Benabid. Recursive Exponentially Weighted N-way Partial Least Squares Regression with Recursive-Validation of Hyper-Parameters in Brain-Computer Interface Applications. *Scientific Reports*, 7(1):16281, December 2017a. ISSN 2045-2322. doi: 10.1038/s41598-017-16579-9. URL http://www.nature.com/articles/s41598-017-16579-9. Number: 1.

Andrey Eliseyev, Vincent Auboiroux, Thomas Costecalde, Lilia Langar, Guillaume Charvet, Corinne Mestais, Tetiana Aksenova, and Alim-Louis Benabid. Recursive Exponentially Weighted N-way Partial Least Squares Regression with Recursive-Validation of Hyper-Parameters in Brain-Computer Interface Applications. *Scientific Reports*, 7:16281, 2017b. doi: 10.1038/s41598-017-16579-9.

Nathan Evans, Steven Gale, Aaron Schurger, and Olaf Blanke. Visual feedback dominates the sense of agency for brain-machine actions. *PLOS ONE*, 10(6): 1–17, 06 2015. doi: 10.1371/journal.pone.0130019. URL https://doi.org/10.1371/journal.pone.0130019.

Faramarz Faghihi, Siqi Cai, and Ahmed A. Moustafa. A neuroscience-inspired spiking neural network for eeg-based auditory spatial attention detection. *Neural Networks*, 152:555–565, 2022. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2022.05.003. URL https://www.sciencedirect.com/science/article/pii/S0893608022001757.

Sharlene N. Flesher, Jennifer L. Collinger, Stephen T. Foldes, Jeffrey M. Weiss, John E. Downey, Elizabeth C. Tyler-Kabara, Sliman J. Bensmaia, Andrew B. Schwartz, Michael L. Boninger, and Robert A. Gaunt. Intracortical microstimulation of human somatosensory cortex. *Science Translational Medicine*, 8 (361):361ra141–361ra141, 2016. doi: 10.1126/scitranslmed.aaf8083. URL https://www.science.org/doi/abs/10.1126/scitranslmed.aaf8083.

Francisco M. Garcia-Moreno, Maria Bermudez-Edo, María José Rodríguez-Fórtiz, and José Luis Garrido. A cnn-lstm deep learning classifier for motor imagery eeg detection using a low-invasive and low-cost bci headband. In *2020 16th International Conference on Intelligent Environments (IE)*, pages 84–91, 2020. doi: 10.1109/IE49459.2020.9155016.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

Ian J. Goodfellow, Yoshua Bengio, and Aaron C. Courville. Deep learning. *Nature*, 521:436–444, 2015.

A. Gramfort, D. Strohmeier, J. Haueisen, M.S. Hämäläinen, and M. Kowalski. Time-frequency mixed-norm estimates: Sparse m/eeg imaging with non-stationary source activations. *NeuroImage*, 70:410–422, 2013. ISSN 1053-8119. doi: https://doi.org/10.1016/j.neuroimage.2012.12.051. URL https://www.sciencedirect.com/science/article/pii/S1053811912012372.

Bhagya Gunasekera, Tarun Saxena, Ravi Bellamkonda, and Lohitash Karumbaiah. Intracortical recording interfaces: Current challenges to chronic recording function. *ACS Chemical Neuroscience*, 6(1):68–83, 2015. doi: 10.1021/cn5002864. URL https://doi.org/10.1021/cn5002864. PMID: 25587704.

Laureen D. Hachem, Christopher S. Ahuja, and Michael G. Fehlings. Assessment and management of acute spinal cord injury: From point of injury to rehabilitation. *The Journal of Spinal Cord Medicine*, 40(6):665–675, November 2017. ISSN 1079-0268, 2045-7723. doi: 10.1080/10790268.2017.1329076. URL https://www.tandfonline.com/doi/full/10.1080/10790268.2017.1329076.

Matti Hämäläinen, Riitta Hari, Risto J. Ilmoniemi, Jukka Knuutila, and Olli V. Lounasmaa. Magnetoencephalography—theory, instrumentation, and applications to noninvasive studies of the working human brain. *Rev. Mod. Phys.*, 65:413–497, Apr 1993. doi: 10.1103/RevModPhys.65.413. URL https://link.aps.org/doi/10.1103/RevModPhys.65.413.

Leigh R. Hochberg, Daniel Bacher, Beata Jarosiewicz, Nicolas Y. Masse, John D. Simeral, Joern Vogel, Sami Haddadin, Jie Liu, Sydney S. Cash, Patrick van der Smagt, and John P. Donoghue. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, 485:372 – 375, 2012. doi: 10.1038/nature11076.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 11 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.

Zhiwu Huang and Luc Van Gool. A riemannian network for spd matrix learning. *ArXiv*, abs/1608.04233, 2017.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015. URL http://arxiv.org/abs/1502.03167.

Beata Jarosiewicz, Nicolas Y Masse, Daniel Bacher, Sydney S Cash, Emad Eskandar, Gerhard Friehs, John P Donoghue, and Leigh R Hochberg. Advantages of closed-loop calibration in intracortical brain–computer interfaces for people with tetraplegia. *Journal of Neural Engineering*, 10(4):046012, jul 2013. doi: 10.1088/1741-2560/10/4/046012. URL https://doi.org/10.1088/1741-2560/10/4/046012.

Ji-Hoon Jeong, Kyung-Hwan Shim, Dong-Joo Kim, and Seong-Whan Lee. Brain-controlled robotic arm system based on multi-directional cnn-bilstm network using eeg signals. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28(5):1226–1238, 2020. doi: 10.1109/TNSRE.2020.2981659.

Laurent Valentin Jospin, Hamid Laga, Farid Boussaid, Wray Buntine, and Mohammed Bennamoun. Hands-on bayesian neural networks—a tutorial for deep learning users. *IEEE Computational Intelligence Magazine*, 17(2):29–48, 2022. doi: 10.1109/MCI.2022.3155327.

Ce Ju, Dashan Gao, Ravikiran Mane, Ben Tan, Yang Liu, and Cuntai Guan. Federated transfer learning for eeg signal classification. *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 3040–3045, 2020.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

Kaushalya Kumarasinghe, Nikola K. Kasabov, and Denise Taylor. Brain-inspired spiking neural networks for decoding and understanding muscle activity and kinematics from electroencephalography signals during hand movements. *Scientific Reports*, 11, 2021.

Zhen Lan, Chao Yan, Zixing Li, Dengqing Tang, and Xiaojia Xiang. Macro: Multi-attention convolutional recurrent model for subject-independent erp detection. *IEEE Signal Processing Letters*, 28:1505–1509, 2021. doi: 10.1109/LSP.2021.3095761.

Christelle Larzabal, Vincent Auboiroux, Serpil Karakas, Guillaume Charvet, Alim-Louis Benabid, Stephan Chabardes, Thomas Costecalde, and Stéphane Bonnet. The riemannian spatial pattern method: mapping and clustering movement imagery using riemannian geometry. *Journal of Neural Engineering*, 18(5):056014, apr 2021a. doi: 10.1088/1741-2552/abf291. URL https://doi.org/10.1088/1741-2552/abf291.

Christelle Larzabal, Stéphane Bonnet, Thomas Costecalde, Vincent Auboiroux, Guillaume Charvet, Stéphan Chabardes, Tetiana Aksenova, and Fabien Sauter-Starace. Long-term stability of the chronic epidural wireless recorder WIMAGINE in tetraplegic patients. *Journal of Neural Engineering*, 18(5):056026, sep 2021b. doi: 10.1088/1741-2552/ac2003. URL https://doi.org/10.1088/1741-2552/ac2003.

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. *Journal of Neural Engineering*, 15(5):056013, jul 2018. doi: 10.1088/1741-2552/aace8c. URL https://doi.org/10.1088/1741-2552/aace8c.

Young-Eun Lee and Seo-Hyun Lee. Eeg-transformer: Self-attention from transformer architecture for decoding eeg of imagined speech. In *2022 10th International Winter Conference on Brain-Computer Interface (BCI)*, pages 1–4, 2022. doi: 10.1109/BCI53720.2022.9735124.

Eric C. Leuthardt, Daniel W. Moran, and Tim R. Mullen. Defining surgical terminology and risk for brain computer interface technologies. *Frontiers in Neuroscience*, 15, 2021. ISSN 1662-453X. doi: 10.3389/fnins.2021.599549. URL https://www.frontiersin.org/articles/10.3389/fnins.2021.599549.

Jesse A Livezey and Joshua I Glaser. Deep learning approaches for neural decoding across architectures and recording modalities. *Briefings in Bioinformatics*, 22(2):1577–1591, 12 2020. ISSN 1477-4054. doi: 10.1093/bib/bbaa355. URL https://doi.org/10.1093/bib/bbaa355.

Henri Lorach, Guillaume Charvet, Jocelyne Bloch, and Grégoire Courtine. Brain–spine interfaces to reverse paralysis. *National Science Review*, 01 2022. ISSN 2095-5138. doi: 10.1093/nsr/nwac009. URL https://doi.org/10.1093/nsr/nwac009. nwac009.

Wolfgang Maass. Networks of spiking neurons: The third generation of neural network models. *Neural Networks*, 10(9):1659–1671, 1997. ISSN 0893-6080. doi: https://doi.org/10.1016/S0893-6080(97)00011-7. URL https://www.sciencedirect.com/science/article/pii/S0893608097000117.

Ravikiran Mane, Tushar Chouhan, and Cuntai Guan. BCI for stroke rehabilitation: motor and beyond. *Journal of Neural Engineering*, 17(4):041001, aug 2020. doi: 10.1088/1741-2552/aba162. URL https://doi.org/10.1088/1741-2552/aba162.

A R Marathe and D M Taylor. The impact of command signal power distribution, processing delays, and speed scaling on neurally-controlled devices. *Journal of Neural Engineering*, 12(4):046031, jul 2015. doi: 10.1088/1741-2560/12/4/046031. URL https://doi.org/10.1088/1741-2560/12/4/046031.

Suzanne Martens, Michael Bensch, Sebastian Halder, Jeremy Hill, Femke Nijboer, Ander Ramos-Murguialday, Bernhard Schoelkopf, Niels Birbaumer, and Alireza Gharabaghi. Epidural electrocorticography for monitoring of arousal

in locked-in state. *Frontiers in Human Neuroscience*, 8, 2014. ISSN 1662-5161. doi: 10.3389/fnhum.2014.00861. URL https://www.frontiersin.org/articles/10.3389/fnhum.2014.00861.

John McCarthy. WHAT IS ARTIFICIAL INTELLIGENCE? page 14, 2004.

George C McConnell, Howard D Rees, Allan I Levey, Claire-Anne Gutekunst, Robert E Gross, and Ravi V Bellamkonda. Implanted neural electrodes cause chronic, local inflammation that is correlated with local neurodegeneration. *Journal of Neural Engineering*, 6(5):056003, aug 2009. doi: 10.1088/1741-2560/6/5/056003. URL https://doi.org/10.1088/1741-2560/6/5/056003.

Corinne S. Mestais, Guillaume Charvet, Fabien Sauter-Starace, Michael Foerster, David Ratel, and Alim Louis Benabid. Wimagine: Wireless 64-channel ecog recording implant for long term clinical applications. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(1):10–21, 2015. doi: 10.1109/TNSRE.2014.2333541.

Kai J. Miller. A library of human electrocorticographic data and analyses. *Nature Human Behaviour*, pages 1–11, 2019.

Alexandre Moly, Thomas Costecalde, Félix Martel, Matthieu Martin, Christelle Larzabal, Serpil Karakas, Alexandre Verney, Guillaume Charvet, Stephan Chabardes, Alim Louis Benabid, and Tetiana Aksenova. An adaptive closed-loop ECoG decoder for long-term and stable bimanual control of an exoskeleton by a tetraplegic. *Journal of Neural Engineering*, 19(2):026021, mar 2022. doi: 10.1088/1741-2552/ac59a0. URL https://doi.org/10.1088/1741-2552/ac59a0.

B. Morinière, A. Verney, N. Abroug, P. Garrec, and Y. Perrot. Emy: a dual arm exoskeleton dedicated to the evaluation of brain machine interface in clinical trials. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5333–5338, 2015. doi: 10.1109/IROS.2015.7354130.

Luis Fernando Nicolas-Alonso and Jaime Gomez-Gil. Brain computer interfaces, a review. *Sensors*, 12(2):1211–1279, 2012. ISSN 1424-8220. doi: 10.3390/s120201211. URL https://www.mdpi.com/1424-8220/12/2/1211.

Christopher Olah. Understanding lstm networks, 2015. URL http://colah.github.io/posts/2015-08-Understanding-LSTMs/.

German I. Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2019.01.012. URL https://www.sciencedirect.com/science/article/pii/S0893608019300231.

János A Perge, Mark L Homer, Wasim Q Malik, Sydney Cash, Emad Eskandar, Gerhard Friehs, John P Donoghue, and Leigh R Hochberg. Intra-day signal instabilities affect decoding performance in an intracortical neural interface system. *Journal of Neural Engineering*, 10(3):036004, apr 2013. doi: 10.1088/

1741-2560/10/3/036004. URL https://doi.org/10.1088/1741-2560/10/3/036004.

Claudia Perlich, Foster Provost, and Jeffrey S. Simonoff. Tree induction vs. logistic regression: a learning-curve analysis. *The Journal of Machine Learning Research*, 4(null):211–255, 2003. ISSN 1532-4435. doi: 10.1162/153244304322972694. URL https://doi.org/10.1162/153244304322972694.

C'edric Rommel, Joseph Paillard, Thomas Moreau, and Alexandre Gramfort. Data augmentation for learning predictive models on eeg: a systematic comparison. *ArXiv*, abs/2206.14483, 2022.

Andreas Rowald, Salif Komi, Robin Demesmaeker, Edeny Baaklini, Sergio Daniel Hernandez-Charpak, Edoardo Paoles, Hazael Montanaro, Antonino Cassara, Fabio Becce, Bryn Lloyd, Taylor Newton, Jimmy Ravier, Nawal Kinany, Marina D'Ercole, Aurélie Paley, Nicolas Hankov, Camille Varescon, Laura McCracken, Molywan Vat, Miroslav Caban, Anne Watrin, Charlotte Jacquet, Léa Bole-Feysot, Cathal Harte, Henri Lorach, Andrea Galvez, Manon Tschopp, Natacha Herrmann, Moïra Wacker, Lionel Geernaert, Isabelle Fodor, Valentin Radevich, Katrien Van Den Keybus, Grégoire Eberle, Etienne Pralong, Maxime Roulet, Jean-Baptiste Ledoux, Eleonora Fornari, Stefano Mandija, Loan Mattera, Roberto Martuzzi, Bruno Nazarian, Stefan Benkler, Simone Callegari, Nathan Greiner, Benjamin Fuhrer, Martijn Froeling, Nik Buse, Tim Denison, Rik Buschman, Christian Wende, Damien Ganty, Jurriaan Bakker, Vincent Delattre, Hendrik Lambert, Karen Minassian, Cornelis A. T. van den Berg, Anne Kavounoudias, Silvestro Micera, Dimitri Van De Ville, Quentin Barraud, Erkan Kurt, Niels Kuster, Esra Neufeld, Marco Capogrosso, Leonie Asboth, Fabien B. Wagner, Jocelyne Bloch, and Grégoire Courtine. Activity-dependent spinal cord neuromodulation rapidly restores trunk and leg motor functions after complete paralysis. *Nature Medicine*, 28(2):260–271, February 2022. ISSN 1546-170X. doi: 10.1038/s41591-021-01663-5. URL https://www.nature.com/articles/s41591-021-01663-5. Number: 2 Publisher: Nature Publishing Group.

Yannick Roy, Hubert Banville, Isabela Albuquerque, Alexandre Gramfort, Tiago H Falk, and Jocelyn Faubert. Deep learning-based electroencephalography analysis: a systematic review. *Journal of Neural Engineering*, 16(5):051001, aug 2019. doi: 10.1088/1741-2552/ab260c. URL https://doi.org/10.1088/1741-2552/ab260c.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

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

Eduardo Santamaría-Vázquez, Víctor Martínez-Cagigal, Fernando Vaquerizo-Villar, and Roberto Hornero. Eeg-inception: A novel deep convolutional neural network for assistive erp-based brain-computer interfaces. *IEEE Transactions*

on Neural Systems and Rehabilitation Engineering, 28(12):2773–2782, 2020. doi: 10.1109/TNSRE.2020.3048106.

Gerwin Schalk and Eric C. Leuthardt. Brain-computer interfaces using electrocorticographic signals. *IEEE Reviews in Biomedical Engineering*, 4:140–154, 2011. doi: 10.1109/RBME.2011.2172408.

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. *Human brain mapping*, 38(11):5391–5420, 2017.

Marc Sebastián-Romagosa, Woosang Cho, Rupert Ortner, Nensi Murovec, Tim Von Oertzen, Kyousuke Kamada, Brendan Z. Allison, and Christoph Guger. Brain computer interface treatment for motor rehabilitation of upper extremity of stroke patients—a feasibility study. *Frontiers in Neuroscience*, 14, 2020. ISSN 1662-453X. doi: 10.3389/fnins.2020.591435. URL https://www.frontiersin.org/articles/10.3389/fnins.2020.591435.

Jerry J. Shih, Dean J. Krusienski, and Jonathan R. Wolpaw. Brain-computer interfaces in medicine. *Mayo Clinic proceedings*, 87 3:268–79, 2012a.

Jerry J. Shih, Dean J. Krusienski, and Jonathan R. Wolpaw. Brain-computer interfaces in medicine. *Mayo Clinic Proceedings*, 87(3):268–279, 2012b. ISSN 0025-6196. doi: https://doi.org/10.1016/j.mayocp.2011.12.008. URL https://www.sciencedirect.com/science/article/pii/S0025619612001231.

Nicholas D. Skomrock, Michael A. Schwemmer, Jordyn E. Ting, Hemang R. Trivedi, Gaurav Sharma, Marcia A. Bockbrader, and David A. Friedenberg. A characterization of brain-computer interface performance trade-offs using support vector machines and deep neural networks to decode movement intent. *Frontiers in Neuroscience*, 12, 2018. ISSN 1662-453X. doi: 10.3389/fnins.2018.00763. URL https://www.frontiersin.org/articles/10.3389/fnins.2018.00763.

Yonghao Song, Xueyu Jia, Lie Yang, and Longhan Xie. Transformer-based spatial-temporal feature learning for eeg decoding. *ArXiv*, abs/2106.11170, 2021.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014. URL http://jmlr.org/papers/v15/srivastava14a.html.

Jiayao Sun, Jin Xie, and Huihui Zhou. Eeg classification with transformer-based models. In *2021 IEEE 3rd Global Conference on Life Sciences and Technologies (LifeTech)*, pages 92–93, 2021. doi: 10.1109/LifeTech52111.2021.9391844.

Michael Tangermann, Klaus-Robert Müller, Ad Aertsen, Niels Birbaumer, Christoph Braun, Clemens Brunner, Robert Leeb, Carsten Mehring, Kai J. Miller,

Gernot R. Müller-Putz, Guido Nolte, Gert Pfurtscheller, Hubert Preissl, Gerwin Schalk, Alois Schlögl, Carmen Vidaurre, Stephan Waldert, and Benjamin Blankertz. Review of the bci competition iv. *Frontiers in Neuroscience*, 6, 2012.

Tijmen Tieleman, Geoffrey Hinton, et al. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

Mariska J. Vansteensel, Elmar G.M. Pels, Martin G. Bleichner, Mariana P. Branco, Timothy Denison, Zachary V. Freudenburg, Peter Gosselaar, Sacha Leinders, Thomas H. Ottens, Max A. Van Den Boom, Peter C. Van Rijen, Erik J. Aarnoutse, and Nick F. Ramsey. Fully implanted brain–computer interface in a locked-in patient with als. *New England Journal of Medicine*, 375(21):2060–2066, 2016. doi: 10.1056/NEJMoa1608085. URL https://doi.org/10.1056/NEJMoa1608085. PMID: 27959736.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Carlos D. Virgilio G., Juan H. Sossa A., Javier M. Antelis, and Luis E. Falcón. Spiking neural networks applied to the classification of motor tasks in eeg signals. *Neural Networks*, 122:130–143, 2020. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2019.09.037. URL https://www.sciencedirect.com/science/article/pii/S0893608019303193.

Ksenia Volkova, Mikhail A. Lebedev, Alexander Kaplan, and Alexei Ossadtchi. Decoding movement from electrocorticographic activity: A review. *Frontiers in Neuroinformatics*, 13, 2019. ISSN 1662-5196. doi: 10.3389/fninf.2019.00074. URL https://www.frontiersin.org/articles/10.3389/fninf.2019.00074.

Ping Wang, Aimin Jiang, Xiaofeng Liu, Jing Shang, and Li Zhang. Lstm-based eeg classification in motor imagery tasks. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(11):2086–2095, 2018. doi: 10.1109/TNSRE.2018.2876129.

Xiaoxi Wei, A. Aldo Faisal, Moritz Grosse-Wentrup, Alexandre Gramfort, Sylvain Chevallier, Vinay Jayaram, Camille Jeunet, Stylianos Bakas, Siegfried Ludwig, Konstantinos Barmpas, Mehdi Bahri, Yannis Panagakis, Nikolaos Laskaris, Dimitrios A. Adamos, Stefanos Zafeiriou, William C. Duong, Stephen M. Gordon, Vernon J. Lawhern, Maciej Śliwowski, Vincent Rouanne, and Piotr Tempczyk. 2021 beetl competition: Advancing transfer learning for subject independence & heterogenous eeg data sets. In Douwe Kiela, Marco Ciccone, and Barbara Caputo, editors, *Proceedings of the NeurIPS 2021 Competitions and Demonstrations Track*, volume 176 of *Proceedings of Machine Learning Research*, pages 205–219. PMLR, 06–14 Dec 2022. URL https://proceedings.mlr.press/v176/wei22a.html.

Francis R. Willett, Donald T. Avansino, Leigh R. Hochberg, Jaimie M. Henderson, and Krishna V. Shenoy. High-performance brain-to-text communication via handwriting. *Nature*, 593 7858:249–254, 2021.

D. Randall Wilson and Tony R. Martinez. The general inefficiency of batch training for gradient descent learning. *Neural networks : the official journal of the International Neural Network Society*, 16 10:1429–51, 2003.

B Wodlinger, J E Downey, E C Tyler-Kabara, A B Schwartz, M L Boninger, and J L Collinger. Ten-dimensional anthropomorphic arm control in a human brain-machine interface: difficulties, solutions, and limitations. *Journal of Neural Engineering*, 12(1):016011, dec 2014. doi: 10.1088/1741-2560/12/1/016011. URL https://doi.org/10.1088/1741-2560/12/1/016011.

Jonathan R. Wolpaw and Dennis J. McFarland. Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *Proceedings of the National Academy of Sciences*, 101(51):17849–17854, 2004. doi: 10.1073/pnas.0403504101. URL https://www.pnas.org/doi/abs/10.1073/pnas.0403504101.

Ziqian Xie, Odelia Schwartz, and Abhishek Prasad. Decoding of finger trajectory from ECoG using deep learning. *Journal of Neural Engineering*, 15(3): 036009, feb 2018. doi: 10.1088/1741-2552/aa9dbe. URL https://doi.org/10.1088/1741-2552/aa9dbe.

Taojiannan Yang, Sijie Zhu, and Chen Chen. Gradaug: A new regularization method for deep neural networks. *Advances in Neural Information Processing Systems*, 33, 2020.

Weibo Yi, Shuang Qiu, Kun Wang, Hongzhi Qi, Lixin Zhang, Peng Zhou, Feng He, and Dong Ming. Evaluation of eeg oscillatory patterns and cognitive process during simple and compound limb motor imagery. *PLOS ONE*, 9(12): 1–19, 12 2014. doi: 10.1371/journal.pone.0114853. URL https://doi.org/10.1371/journal.pone.0114853.

Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *CoRR*, abs/1511.07122, 2016.

Dalin Zhang, Lina Yao, Kaixuan Chen, and Jessica Monaghan. A convolutional recurrent attention model for subject-independent eeg signal analysis. *IEEE Signal Processing Letters*, 26(5):715–719, 2019. doi: 10.1109/LSP.2019.2906824.

Ruilong Zhang, Qun Zong, Liqian Dou, Xinyi Zhao, Yifan Tang, and Zhiyu Li. Hybrid deep neural network using transfer learning for eeg motor imagery decoding. *Biomedical Signal Processing and Control*, 63:102144, 2021. ISSN 1746-8094. doi: https://doi.org/10.1016/j.bspc.2020.102144. URL https://www.sciencedirect.com/science/article/pii/S1746809420302901.

Jie Zhou, Ming Meng, Yunyuan Gao, Yuliang Ma, and Qizhong Zhang. Classification of motor imagery eeg using wavelet envelope analysis and lstm networks. In *2018 Chinese Control And Decision Conference (CCDC)*, pages 5600–5605, 2018. doi: 10.1109/CCDC.2018.8408108.

Artificial intelligence for real-time decoding of motor commands from ECoG of disabled subjects for chronic brain-computer interfacing

Brain-computer interfaces (BCIs) may significantly improve tetraplegic patients' quality of life by creating an alternative communication path between humans and the environment, potentially compensating for motor function loss. This thesis focuses on ECoG-based BCI systems that showed a high potential to provide efficient communication while being less invasive than intracortical recordings. In particular, we explored problem of continuous 3D hand translation decoding in a tetraplegic patient. In this case, most studies use linear models that may be too simple to analyze brain processes and may suffer from low decoding accuracy. Models based on deep learning (DL) have been proven effective in various tasks and thus emerge as a potential solution to create a robust brain signals representation. In this thesis, we studied the potential of DL-based methods for hand translation decoding from ECoG signals.

First, we evaluated several DL models to predict 3D hand translation from ECoG time-frequency features. The analysis was performed on a dataset recorded with a tetraplegic subject in the BCI and Tetraplegia clinical trial (NCT02550522). We started the investigation with a multilayer perceptron taking vectorized features as input. Then, we proposed convolutional neural networks (CNN), which take matrix-organized inputs approximating the spatial arrangement of the electrodes. In addition, we investigated the usefulness of long short-term memory (LSTM) to analyze temporal information. Results showed that CNN-based architectures performed better than the current state-of-the-art multilinear model on the analyzed ECoG dataset. The best architecture used a CNN-based model to analyze the spatial representation of time-frequency features followed by LSTM exploiting the sequential character of the desired hand trajectory. Compared to the multilinear model, DL-based solutions increased average cosine similarity by up to 60%.

In the case of BCI, access to large datasets is limited because recordings are time-consuming and tiring. To investigate the influence of the dataset size on the decoding performance, we compared the learning curve characteristics of DL and multilinear models evaluated in the previous step. The training dataset size was gradually increased from 5 to 140 minutes of signal in different computational experiments, focusing on dataset size requirements and patient adaptation effects. Our results revealed that DL models have a learning curve profile similar to the multilinear model, increasing performance for almost all training dataset sizes. This result validates the DL-based models as a good candidate for real-life applications. We also observed increased data quality for recordings performed later, indicating improved patient ability to generate meaningful patterns.

DL proved its usefulness for computer vision, primarily in the case of end-to-end learning. It enabled extracting more powerful representations trained for a specific task and removed the step of hand-crafted feature extraction. We evaluated methods using raw ECoG signals as a natural extension of hand-crafted feature analysis. In the data processing pipeline evaluated so far, continuous wavelet transform was used to extract time-frequency representation, which can be seen as a convolution between a set of wavelet filters and the ECoG signal. In this setup, the gradient w.r.t. filters coefficients can be computed, and thus the whole network can be trained within an end-to-end scheme. The parameters of wavelet filters were optimized end-to-end to see potential profit from adjusting the parameters to this specific problem. The results showed only minor or no benefit from training the wavelets in terms of cosine similarity, while end-to-end models require more computational power. This may suggest that training first layer parameters may be less beneficial and more challenging in the case of ECoG-based BCI.

Keywords: brain-computer interface, BCI, ECoG, tetraplegia, deep learning, convolutional neural networks, multilayer perceptron, LSTM, dataset size, learning curve, end-to-end, hand-crafted features, time-frequency representation, brain signals

Intelligence artificielle pour le décodage de commandes motrices de sujets handicapés, grâce à des interfaces cerveau-machine à usage chronique

Les interfaces cerveau-machines (ICMs) peuvent améliorer considérablement la qualité de vie des patients tétraplégiques en créant une voie de communication alternative et en compensant la perte de la fonction motrice. Cette thèse se concentre sur les systèmes ICM basés sur l'ECoG qui ont montré un fort potentiel pour fournir une communication efficace, tout en étant moins invasif que les enregistrements intracorticaux. Cependant, la plupart des systèmes actuels souffrent d'une faible précision de décodage. Les modèles basés sur l'apprentissage profond (DL) se sont avérés efficaces dans de nombreuses tâches et apparaissent donc comme une solution possible pour créer une représentation robuste des signaux cérébraux. Dans cette thèse, nous avons étudié le potentiel des méthodes basées sur l'apprentissage profond pour le décodage de la translation de la main à partir de signaux ECoG.

Tout d'abord, nous avons évalué plusieurs modèles de DL sur des données de l'essai clinique BCI et Tétraplégie (NCT02550522) enregistrées avec un sujet tétraplégique, pour prédire la translation de la main en 3D à partir des caractéristiques temps-fréquence des signaux ECoG. Nous avons commencé l'étude avec un perceptron multicouche prenant en entrée des caractéristiques vectorisées. Ensuite, nous avons proposé des réseaux neuronaux convolutifs (CNN), qui prennent des entrées organisées en matrices se rapprochant de la disposition spatiale des électrodes, et des cellulles long short-term memory (LSTM) pour analyser les informations temporelles. Les résultats ont montré que les architectures basées sur les réseaux neuronaux convolutifs sont plus performantes que le modèle multilinéaire actuel sur l'ensemble des données ECoG analysées, avec une augmentation de la similarité cosinus moyenne allant jusqu'à 60 %. La meilleure architecture utilisait un modèle CNN pour analyser la représentation spatiale des caractéristiques temps-fréquence, suivi d'un LSTM exploitant le caractère séquentiel des données.

Dans le cas des ICM, l'accès à de grands jeux de données est limité car les enregistrements sont longs et fatigants pour le patient. Pour étudier l'impact de la taille du jeu de données sur les performances de décodage, la quantité de données d'entraînement a été progressivement augmentée de 5 à 140 minutes de signal dans différentes expériences de calcul, en se concentrant sur les exigences de taille du jeu de données et les effets d'adaptation du patient. Nos résultats ont révélé que les modèles de DL augmentent les performances pour presque toutes les tailles de jeu de données d'entraînement. Ce résultat valide les modèles de DL comme bons candidats pour les applications de la vie réelle. Nous avons également observé une meilleure qualité des données pour les enregistrements effectués plus tardivement, ce qui indique un progrès du patient pour générer des signaux informatifs.

Le DL a prouvé son utilité pour la vision par ordinateur, principalement dans le cas de l'apprentissage de bout en bout, permettant l'extraction de représentations plus adaptées à une tâche spécifique. Nous avons évalué des méthodes utilisant des signaux ECoG bruts en tant qu'extension naturelle de l'analyse des caractéristiques créées à la main. Les paramètres de la transformée en ondelettes continue utilisée pour extraire la représentation temps-fréquence ont été optimisés de bout en bout pour voir le bénéfice potentiel de l'ajustement des paramètres à ce problème spécifique. Les résultats ont montré qu'il n'y avait que peu ou pas de bénéfice à entraîner les ondelettes en termes de similarité en cosinus, alors que les modèles de bout en bout nécessitent plus de puissance de calcul. Cela peut suggérer que l'entraînement des paramètres de la première couche peut être moins bénéfique et plus difficile dans le cas d'une BCI basée sur l'ECoG.

Mots clés : interface cerveau machine, ICM, ECoG, tétraplégie, apprentissage profond, réseaux neuronaux convolutifs, perceptron multicouche, LSTM, taille de jeu de données, courbe d'apprentissage, de bout en bout, caractéristiques créées à la main, représentation temps-fréquence, signaux cérébraux

This thesis was typeset using LaTeX, originally developed by Leslie Lamport and based on Donald Knuth's TeX. A template that can be used to format a PhD dissertation with this look & feel has been released under the permissive AGPL license, and can be found online at github.com/suchow/Dissertate or from its lead author, Jordan Suchow, at suchow@post.harvard.edu.