Geometric Neural Network based on Phase Space for BCI decoding

Igor Carrara, Bruno Aristimunha, Marie-Constance Corsi, Raphael Y de Camargo, Sylvain Chevallier, Théodore Papadopoulo

GEOMETRIC NEURAL NETWORK BASED ON PHASE SPACE FOR BCI DECODING

Igor Carrara[1*] **Bruno Aristimunha**[2,3,*]
Marie-Constance Corsi[4] **Raphael Y. de Camargo**[3] **Sylvain Chevallier**[2] **Théodore Papadopoulo**[1]

[1] Université Côte d'Azur, Inria d'Université Côte d'Azur, Sophia Antipolis, France,
[2] Université Paris-Saclay, Inria TAU, LISN-CNRS, France,
[3] Universidade Federal do ABC, Santo André, Brazil,
[4] ARAMIS, Inria, Paris Brain Institute, Paris, France.

ABSTRACT

Objective: The integration of Deep Learning (DL) algorithms on brain signal analysis is still in its nascent stages compared to their success in fields like Computer Vision, especially in Brain-Computer Interface (BCI), where the brain activity is decoded to control external devices without requiring muscle control. Electroencephalography (EEG) is a widely adopted choice for designing BCI systems due to its non-invasive and cost-effective nature and excellent temporal resolution. Still, it comes at the expense of limited training data, poor signal-to-noise, and a large variability across and within-subject recordings. Finally, setting up a BCI system with many electrodes takes a long time, hindering the widespread adoption of reliable DL architectures in BCIs outside research laboratories. To improve adoption, we need to improve user comfort using, for instance, reliable algorithms that operate with few electrodes. **Approach:** Our research aims to develop a DL algorithm that delivers effective results with a limited number of electrodes. Taking advantage of the Augmented Covariance Method with SPDNet, we propose the SPDNet$_\psi$ architecture and analyze its performance and computational impact, as well as the interpretability of the results. The evaluation is conducted on 5-fold cross-validation, using only three electrodes positioned above the Motor Cortex. The methodology was tested on nearly 100 subjects from several open-source datasets using the Mother Of All BCI Benchmark (MOABB) framework. **Main results:** The results of our SPDNet$_\psi$ demonstrate that the augmented approach combined with the SPDNet significantly outperforms all the current state-of-the-art DL architecture in MI decoding. **Significance:** This new architecture is explainable, with a low number of trainable parameters and a reduced carbon footprint.

Keywords Brain-Computer Interfaces · Electroencephalography · Functional connectivity · SPD manifold · Riemannian optimization · Neural Network · Motor Imagery

1 Introduction

Brain-computer interface (BCI) technology allows direct communication between a user's brain activity and external devices. Originally designed to help people with disabilities [1], its uses are expanding to other fields, such as rehabilitation using virtual reality [2, 3]. Different signals acquired from brain activity can be used for such a technology, but Electroencephalography (EEG) is a widely adopted choice, as it is a non-invasive, portable, and inexpensive methodology with a very good temporal resolution. Motor imagery (MI) tasks are largely investigated among the brain activities considered when designing a BCI. As the subject is asked to mentally execute a movement without actually performing it, it provides an asynchronous and internal control scheme with no requirement for muscle capability.

The application of Deep Learning (DL) algorithms has garnered significant attention over several domains, ranging from natural language process [4] to protein structure prediction [5]. The field of EEG MI classification is no exception.

[*] Bruno and Igor are joint first authors.

However, DL algorithms have not yet established themselves in the BCI field as they have in other fields, such as Computer Vision. This is due to several problems: limited data availability, low signal-to-noise ratio in EEG signals, subject variability due to anatomical differences between individuals, and to session variability due to deviations in electrode placement [6].

Unlike other fields where DL algorithms thrive on extensive data, in EEG applications, the emphasis is on enhancing user comfort, resulting in smaller datasets with few trials and fewer electrodes. Large-scale EEG systems with numerous electrodes not only require extended calibration periods, causing user tiredness but also introduce complexity, potentially leading to increased error rates and heightened computational demands. Additionally, the increased cost associated with extensive electrode setups deters the widespread development, deployment, and accessibility of BCI technology, particularly for the patients who could benefit the most from it.

This research focuses on developing a novel DL architecture, SPDNet$_\psi$, that outperforms the state-of-the-art classification when using a limited number of electrodes. Building on the Augmented Covariance Method (ACM) [7] that is an extension of the spatial covariance and the Symmetric Positive Definite (SPD) Neural Network -SPDNet [8], we study the SPDNet$_\psi$ impact on the performance when using a reduced number of electrodes. We also conducted a comprehensive analysis of the model size, carbon emissions, and, ultimately, its explainability with respect to the standard SPDNet.

Spatial covariance, however, is not the only one that can be extracted from the EEG signal. Another potential candidate is coherence, which provides an alternative perspective on EEG signal characteristics. Historically, coherence has proven to be an inherently unstable feature, which is challenging to compute accurately, resulting in less robust results with respect to the spatial covariance. However, the use of information from both covariance and coherence has been shown to increase performance as we are considering an estimator of the interactions between brain areas/ electrodes [9]. We are thus interested to study coherence in combination with SPDNet and especially to use the ACM methodology adapted for the coherence matrix.

Our methodology is tested through a 5-fold cross-validation evaluation (*Within-Session*), utilizing only three electrodes strategically positioned above the Motor Cortex. To validate our algorithm, we test our approach on almost 100 subjects from openly available datasets using the Mother Of All BCI Benchmark (MOABB) framework [10]. This research not only contributes to the advancement of EEG MI classification but also emphasizes the importance of developing efficient, user-friendly algorithms with a minimal environmental impact in the broader context of BCI technology. Additionally, our study places a strong emphasis on ensuring that our method is both reproducible and interpretable.

The article is organized as follows: In section 2, we delve into the current state-of-the-art in Deep Learning (DL) EEG decoding, emphasizing distinctions from our approach. section 3 provides an overview of the theoretical foundations of our model and the considered datasets. The obtained results from the Within-session evaluation are presented in section 4. Subsequently, section 5 focuses on the method's impact and current limitations, specifically focusing on its explainability and environmental impact. Finally, section 6 summarizes the findings of our study.

2 Related Work

2.1 Machine Learning for EEG Decoding

In the domain of EEG decoding [11, 12], translating brain activity into meaningful data has become increasingly dependent on machine learning (ML) methods [6, 9, 13–17]. However, here, we diverge by focusing on deep learning techniques instead of a broad ML spectrum, moving beyond mere method comparison to introduce a novel EEG decoding approach.

Schirrmeister et al. [14] demonstrated that Deep Learning (DL) approaches, specifically *ShallowNet* and *DeepNet* can perform on par with conventional machine learning in decoding raw EEG data and can be trained end-to-end, eliminating several feature extraction steps. The application of DL algorithms enhances the generalization capability, enabling it to handle the inherent variability present in EEG signals effectively. This applicability of DL for EEG data is further corroborated by [15, 18–21]. Despite their effectiveness, these DL approaches demand extensive parameter tuning, high power consumption, and large datasets for thorough evaluation. In contrast, our study proposes a streamlined approach, leveraging Riemannian geometry and functional connectivity derivatives, which are less demanding in terms of parameters, enhancing both efficiency and scalability.

2.2 Deep Riemannian Networks for EEG Decoding

Incorporating non-euclidean geometry, especially the Riemannian manifold, into EEG decoding has significantly advanced the field [22–34]. A common practice is to compute spatial covariance matrices that capture signal features

with the structure of symmetric positive definite (SPD) matrices. These matrices not only enhance signal information regarding topology and amplitude but also offer increased robustness to outliers and noise with Riemannian geometry, maintaining invariance under linear transformations [24, 29].

Huang and Van Gool [22] work introduced *SPDNet*, a neural network that operates on SPD manifolds. This foundation was used by Ju et al. [35], who applied SPDNet in bio-signal classification, enhancing transfer learning. Building on this, Brooks et al. [36] adapted batch normalization for the SPD manifold, and Kobler et al. [25] further refined the batch normalization component for EEG Decoding. Pan et al. [24] added to this perspective by proposing attention components for the SPDNet zoo. Follow-up efforts included using residual layers [26, 31], filter bank inputs [23, 27, 30], mixing traditional convolution by channels [34], and constraining diffusion models [37, 38]. Contrary to our approach, these methods rely on adapting existing neural network components for the SPD manifold or preserving the dimensionality assumptions, while our method combines the dimensionality expansion of the phase space reconstruction on the SPD manifold.

2.3 Geometry Transformation in EEG data

Geometry Transformation (GT) in EEG analysis involves applying a transformation function to the data [6, 7, 15, 39]. This process commonly includes standard pre-processing transformations like resampling, band-passing, and filtering, which have been proven to enhance model performance and biomarker discovery [16, 40–43]. Data augmentation, a regularization geometry transformation, has been acknowledged for its role in enhancing brain decoding in EEG, with its benefits varying depending on the task [15, 44]. While these transformations can improve the learning processing, these steps usually do not consider the non-stationary dynamics properties of the biological signal. Our approach broadens this scope by incorporating transformations that consider phase reconstruction in terms of its components of nonlinear dynamics.

In the context of delay embedding phase geometry transformation, Chen et al. [45] demonstrated the effectiveness of phase embedding in neural networks for one-dimensional bio-signal reconstruction, but without addressing the non-euclidean nature of the signal. Our methodology, in contrast, considers the non-euclidean geometry, thereby enriching the learning process. The studies by Carrara and Papadopoulo [7] and Zhou et al. [46] align with our approach in their use of phase delays and (cross-)covariance matrices for EEG decoding. Our work, on the other hand, differentiates by fully leveraging the capabilities of the SPD matrix in Riemannian Neural Networks, allowing for more efficient, interpretable layers and achieving superior performance within challenging scenarios with a reduced number of channels.

3 Material and Methods

In this section, we describe the EEG decoding problem, the phase space reconstruction transformation, the Riemannian manifold properties, the Symmetric Positive Definite neural network components, and the datasets and baselines.

3.1 EEG Decoding

We consider EEG signals as real-valued matrices. Let us denote each bio-signal window captured during a cognitive task on a given electrode by $\mathbf{x} = x(t, t+T)$, a time serie of T elements. We denote an epoch as $\mathbf{X}_i = [\mathbf{x}_1, \ldots, \mathbf{x}_C]$, with $\mathbf{X}_i \in \mathbb{R}^{C \times T}$ and the corresponding cognitive task is denoted by $y_i \in \mathcal{Y}$. Here, C represents the number of channels (electrodes), T is the number of time points in the window, and i indexes the trial $i = 1 \ldots N$, where N is the total number of trials.

EEG decoding aims to construct a function $f : \mathcal{X} \to \mathcal{Y}$, which effectively maps each trial $\mathbf{X}_i$ to its corresponding label y_i. We can construct a neural network function represented as $f_\theta : \mathcal{X} \to \mathcal{Y}$, composing a sequence of l functions, formulated as $f_\theta = f_{\theta_1} \circ f_{\theta_2} \circ \cdots \circ f_{\theta_l}$. In this structure, θ indicates the parameters describing each layer of the network.

3.2 Phase space reconstruction for the EEG signal

When using a reduced number of electrodes, the captured signals will contain only part of the real dynamics of the brain. It is possible to recover part of this information using time delay embeddings based on the Takens's theorem [47]. This methodology allows the reconstruction of the dynamical system in an alternative space, different from the sensors one, but containing the same dynamical information as the brain.

Our method applies Takens' theorem over the minimally pre-processed epoch-cropped time series, which allows for the understanding of a system's multi-variable dynamics through a single observable in a Ψ-dimensional space by employing

a phase embedding transformation as the initial function in our neural network [47–49]. This is made by constructing a phase space using a delay vector constructed from the original signal, thereby enabling the reconstruction of the system dynamics from these observations. Consider an embedding delay τ and a delay function $d_p(\mathbf{x}) = x(t - p\tau, t + T - p\tau)$, the delay vector is defined as:

$$D_{\Psi-1}(\mathbf{X}) = [d_0(\mathbf{x}_1), d_1(\mathbf{x}_1), \ldots, d_{\Psi-1}(\mathbf{x}_1), d_0(\mathbf{x}_2), \ldots, d_{\Psi-1}(\mathbf{x}_2), \ldots, d_0(\mathbf{x}_C), \ldots, d_{\Psi-1}(\mathbf{x}_C)]^\top \tag{1}$$

where Ψ represents the embedding dimension, which dictates the order of magnitude of the phase space, and τ is the embedding delay, determining the temporal resolution of our analysis [50, 51]. As a result, $D_{\Psi-1}(\mathbf{X})$ embodies an embedding of the original phase space into a higher-dimensional space, enabling a detailed examination of the system's dynamics.

Consider EEG window signals captured during a cognitive task as $\mathbf{X}_i$, and consider this signal produced by a nonlinear dynamical system. To create the higher dimensional space, we use the methodology $f_{\text{delay}}(\mathbf{X})$ proposed by Carrara and Papadopoulo [7], consisting of sliding measures over the observable with a subsequent concatenation. We define the lag function $s_p(\mathbf{x}) = x(t + p\tau, t + T - (\Psi - p)\tau)$, and given the parameters ψ and τ, $f_{\text{delay}}(\mathbf{X})$ is defined as:

$$f_{\text{delay}}(\mathbf{X}) = [s_0(\mathbf{x_1}), \ldots, s_{\Psi-1}(\mathbf{x_1}), \ldots, s_0(\mathbf{x_C}), \ldots, s_{\Psi-1}(\mathbf{x_C})]^\top \tag{2}$$

This function is applied on each window signal $\mathbf{X}_i \in \mathcal{X} = \mathbb{R}^{C \times T}$.

Selecting the optimal τ and ψ values can be extremely time-consuming [7, 51]. To overcome this difficulty, we use an adaption of Maximizing Derivatives On Projection (MDOP) [7, 52] method for the multi-epoch context of EEG Decoding, shown in Algorithm 1. The algorithm relies on the function $mdop\_embedding$ from the package DelayEmbeddings to implement the MDOP procedure. The function aims to create a nonuniform embedding that generates a vector of different lags $\boldsymbol{\tau}_i$.

Algorithm 1 MDOP for EEG Decoding

using DelayEmbeddings: mdop_embedding
function MDOP FOR EPOCHS(X)
  $\tau = 0$
  $\psi = 0$
  for $i = 1 : \#\text{epochs}$ **do**
    $\boldsymbol{\tau}_i = \text{mdop\_embedding}(X_i)$
    $\tau = \tau + \lfloor \overline{\boldsymbol{\tau}_i} \rfloor$
    $\psi = \psi + |\text{len}(\boldsymbol{\tau}_i)|$
  end for
  $\overline{\tau}_{Epoch} = \lfloor \frac{\tau}{|\#\text{epochs}|} \rfloor$
  $\overline{\psi}_{Epoch} = \lfloor \frac{\psi}{|\#\text{epochs}|} \rfloor$
return $\overline{\tau}_{Epoch}, \overline{\psi}_{Epoch}$
end function

The MDOP algorithm emerged as a novel geometric for analyzing dynamical systems, as opposed to conventional statistical and information-theoretic methods, such as mutual information and continuity statistics. Central to MDOP is optimizing an embedding to ensure the reconstructed attractor is as expanded as possible while simultaneously reducing redundancy among the delay components. The algorithm employs a recursive approach, wherein each step of the embedding cycle identifies the lag τ that yields the highest beta statistics. This chosen lag is then utilized in the subsequent reconstruction phase. This iterative process continues until the algorithm obtains satisfying embedding dimensions ψ, selected using the method of false nearest neighbors.

3.3 Riemannian Manifold

Symmetric Positive Definite (SPD) matrices have begun to play a key role in several applications, ranging from brain imaging to Computer Vision [53, 54]. In particular, the classification approach of SPD matrices based on the Riemannian distance algorithm is the current state-of-the-art in the BCI-MI classification [43, 55].

Let's define $\mathcal{M}_n$ the space of real square matrix and $\mathcal{S}^n$ the space of symmetric matrix, where $\mathcal{S}^n = \{\mathcal{S}^n \in \mathcal{M}_n \mid \mathbf{S}^\top = \mathbf{S}\}$. It is now possible to define the space of Symmetric Positive Definite (SPD) matrices as

$$\mathcal{S}^n_{++} = \{S \in \mathcal{S}^n \mid \boldsymbol{x}^\top \boldsymbol{S} \boldsymbol{x} > 0 \quad \forall \boldsymbol{x} \in \mathbb{R}^n\} \tag{3}$$

This formulation allows us to represent matrices belonging to $\mathcal{S}_{++}^{n}$ as points on a Riemann manifold with a dimension of $n(n+1)/2$. The space of SPD matrices forms a manifold with negative curvature [56, 57], so Euclidean geometry concepts do not apply.

It is possible to define several distances between two SPD matrices $\mathcal{S}_1$ and $\mathcal{S}_2$, generally depending on the length of the geodesic connecting $\mathcal{S}_1$ and $\mathcal{S}_2$ on the Riemann manifold. Often, the affine-invariant metric is used in the context of BCI. We proceed, however, to give a mathematical formulation based on the Log-Euclidean metric [58]. This formulation reduces the computational burden associated with the affine-invariant framework while preserving robust theoretical properties [58].

The space of Symmetric Positive Definite (SPD) matrices can be endowed with a Lie group structure. For a comprehensive understanding, please refer to [58]. It becomes possible to define a distance metric between two SPD matrices $\mathcal{S}_1$ and $\mathcal{S}_2$ as the bi-variate metric on the Lie group of SPD matrices

$$d(\mathcal{S}_1, \mathcal{S}_2) = ||\log(\mathcal{S}_2) - \log(\mathcal{S}_1)|| \tag{4}$$

where $||\quad||$ is the norm associated with the metric, and $\log$ is the matrix logarithm. The Log-Euclidean metric on the Lie group of SPD matrices corresponds to an Euclidean metric within the logarithmic domain of the SPD matrices.

Distances, geodesics, and Riemannian means exhibit a more straightforward formulation within the Log-Euclidean metric than the affine-invariant case, maintaining comparable invariance properties. However, this simplification comes at the cost of more intricate formulations for exponential and logarithmic mapping. The mapping of $\mathcal{S}_2$ respect to $\mathcal{S}_1$ is defined as

$$\begin{aligned} \log_{\mathcal{S}_1}(\mathcal{S}_2) &= D_{\log(\mathcal{S}_1)} \exp \cdot (\log(\mathcal{S}_2) - \log(\mathcal{S}_1)), \\ \exp_{\mathcal{S}_1}(\mathcal{S}_2) &= \exp(\log(\mathcal{S}_1) + D_{\mathcal{S}_1} \log \cdot \mathcal{S}_2). \end{aligned} \tag{5}$$

where $D_{\log(\mathcal{S}_1)} \exp$ represent the differential at point $\mathcal{S}_1$ of the exponential function and similarly for $(D_{\log(\mathcal{S}_1)} \exp)^{-1} = D_{\mathcal{S}_1} \log$. This formulation can be simplified if the reference matrix for the mapping is the identity matrix.

In BCI, the spatial covariance is estimated from the pre-processed EEG signal $\mathbf{X} \in R^{C \times T}$. Several estimators can be used to estimate covariance [59], but the most popular is the sample covariance matrix

$$f_{\mathbb{C}\mathrm{ov}}(\mathbf{X}_i) = \frac{1}{T-1} \sum_{i=1}^{T} \mathbf{X}_i \mathbf{X}_i^{\top} \tag{6}$$

3.3.1 Symmetric Positive Definite Neural Networks Components

In the context of geometry neural networks, Huang and Van Gool [8] proposed the neural network SPDNet with three SPD layers:

BiMap Layer: The bi-linear mapping level aims at creating more compact and discriminating SPD matrices just like the convolutional level, with the complication, however, that SPD matrices live in a Riemannian space. We can describe the equation as

$$f_{\mathrm{BiMap}}(\mathbf{Z}_{k-1}) = \mathbf{Z}_k = \mathbf{W}_k \mathbf{Z}_{k-1} \mathbf{W}_k^{\top}, \tag{7}$$

where $\mathbf{Z}_{k-1} \in \mathcal{S}_{d_{k-1}}^{++}$ and $\mathbf{W}$ is, for this layer, the learnable parameter. Because the output $\mathbf{Z}_k$ must belong to the SPD space, we require that $\mathbf{W}_k \in \mathbb{R}^{(d_k, d_{k-1})}$ belongs to a compact Stiefel $St(d_k, d_{k-1})$ manifold.

ReEig Layer: This layer introduces a non-linearity with a similar approach to a Rectified Linear Unit (ReLU) level. In practice, it rectifies SPD matrices by thresholding small eigenvalues to ε.

$$f_{\mathrm{ReEig}}(\mathbf{Z}_{k-1}) = \mathbf{Z}_k = \mathbf{U}_{k-1} \max(\varepsilon I, \Sigma_{k-1}) \mathbf{U}_{k-1}^{\top}, \tag{8}$$

where $\mathbf{U}_{k-1}$ and Σ_{k-1} are not learned but obtained using eigenvalue decomposition of the previous layer, $\mathbf{Z}_{k-1} = U_{k-1} \Sigma_{k-1} U_{k-1}^{T}$. This layer does not present any trainable parameter.

LogEig: This layer aims to transport the SPD matrix obtained from the previous layers from a Riemannian space to an Euclidean one using the Log-Euclidean metric [58]. This is formally the expression (5) considering the mapping with respect to the identity, so formally, the layer is defined as

$$f_{\mathrm{LogEig}}(\mathbf{Z}_{k-1}) = \mathbf{Z}_k = \log(\mathbf{Z}_{k-1}) = \mathbf{U}_{k-1} \log(\Sigma_{k-1}) \mathbf{U}_{k-1}^{\top} \tag{9}$$

where again, the $\mathbf{U}_{k-1}$ and Σ_{k-1} are not learned but obtained using eigenvalue decomposition. This layer not present any trainable parameter. This logarithmic mapping is applied using the Identity as a reference matrix. Once this layer is applied, we can use the classical deep learning method in the Euclidean space as a Multi-Layer Perceptron.

Given the framework, we define our neural network, denoted as f, as a composition of sequential transformations:

$$f_{\mathrm{SPDNet}\psi} = f_{\mathrm{delay}} \circ f_{\mathbb{C}\mathrm{ov}} \circ f_{\mathrm{BiMap}} \circ f_{\mathrm{ReEig}} \circ f_{\mathrm{LogEig}} \circ f_{\mathrm{MLP}}, \tag{10}$$

where each f_l represents a specific transformation or layer within the network. The learning process is formalized as the mapping $f_{\mathrm{SPDNet}\psi} : \mathcal{X} \to \mathcal{Y}$, which operates on the training dataset. θ denotes the set of parameters within the parameter space Θ. The optimization objective is to minimize an average loss ℓ over the training dataset, defined as:

$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \ell(f_{\mathrm{SPDNet}\psi}(\mathbf{X}_i), \mathbf{y}_i). \tag{11}$$

The model's generalization capability is further assessed using an independent test set.

In a DL typical training process, optimization methods like Adam [60] are normally used for implementing the back-propagation procedure. However, in the context of SPDNet, we must the weights of W of the BiMap layer in such a way that the new weights still belong to the Stiefel manifolds, i.e., they are still orthonormal matrices. To solve the problem, we use the RiemannAdam [61] optimization from the geoopt [62] library. In addition to the optimization strategy, we use the standard cross-entropy loss function ℓ as a loss function, a well-established and widely used objective function in classification tasks.

To summarize, the resulting architecture SPDNet$_\psi$uses enriched SPD matrices that encapsulate a broader spectrum of information than their traditional counterparts. In addition, such an approach allows, through the use of nonlinear systems theory, the reconstruction of a phase space that contains more information with respect to the one extracted from the original signal, allowing the use of fewer electrodes. The augmented SPD matrices require an SPDNet with a larger number of parameters, which is partly counteracted by using fewer electrodes. In particular, this representation accentuates how the input SPD matrix is modified from the initial dimension $\mathcal{S}_C^{++}$ to $\mathcal{S}_{C\times\psi}^{++}$.

Overall, this enrichment of the information contained in the SPD matrices not only expands the range of discriminative information that can be extracted but also forces the network to adapt to more intricate and fuzzy patterns in the data. Figure 1 provides a graphical picture of our methodology.

In the case of SPDNet$_\psi$, we set the subspace dimension of the BiMap layer at half of the input dimension [8]. However, for standard SPDNet, we maintained the subspace dimension at the same value as the original input dimension (although in this case the BiMap layer acts only as a rotation) in order to proceed with subsequent analyses concerning the explainability of the model. Anyway, diminishing the subspace dimension of half in the case of SPDNet consistently led to a reduction in performance. Note that the current implementation of SPDNet has a ReEig layer that is not scale-independent because of the parameter $\varepsilon = 10^{-4}$. To enhance the architecture's scale independence, we have introduced a standardization procedure for the raw signal, bringing every channel to a zero mean and a unit standard deviation.

3.4 Datasets

In order to assess the replicability of our approach, we used six different open datasets from MOABB [10] consisting of almost 100 subjects as shown in Table 1.

It turns out that it is essential to employ different datasets to obtain reliable results, especially when evaluating different ML techniques in EEG decoding, in order to avoid the bias of a single data set [13].

Dataset	Subjects	Channels	Sampling Rate (Hz)	Sessions	Tasks	Trials/Class	Epoch (s)
BNCI2014001 [63]	9	22	250	2	4	144	[2, 6]
BNCI2014004 [64]	9	3	250	5	2	360	[3, 7.5]
Cho2017 [65]	52	64	512	1	2	100	[0, 3]
Schirrmeister2017 [66]	14	128	500	1	4	120	[0, 4]
Weibo2014 [67]	10	60	200	1	7	80	[3, 7]
Zhou2016 [68]	4	14	250	3	3	160	[0, 5]

Table 1: Motor Imagery datasets considered during this study

We employed standard pre-processing steps designed for all datasets described in [10]. We applied band-pass filtering with the overlap-add method between $[8 - 32]$ Hz, artifact rejection, and electrode standardization, bringing every

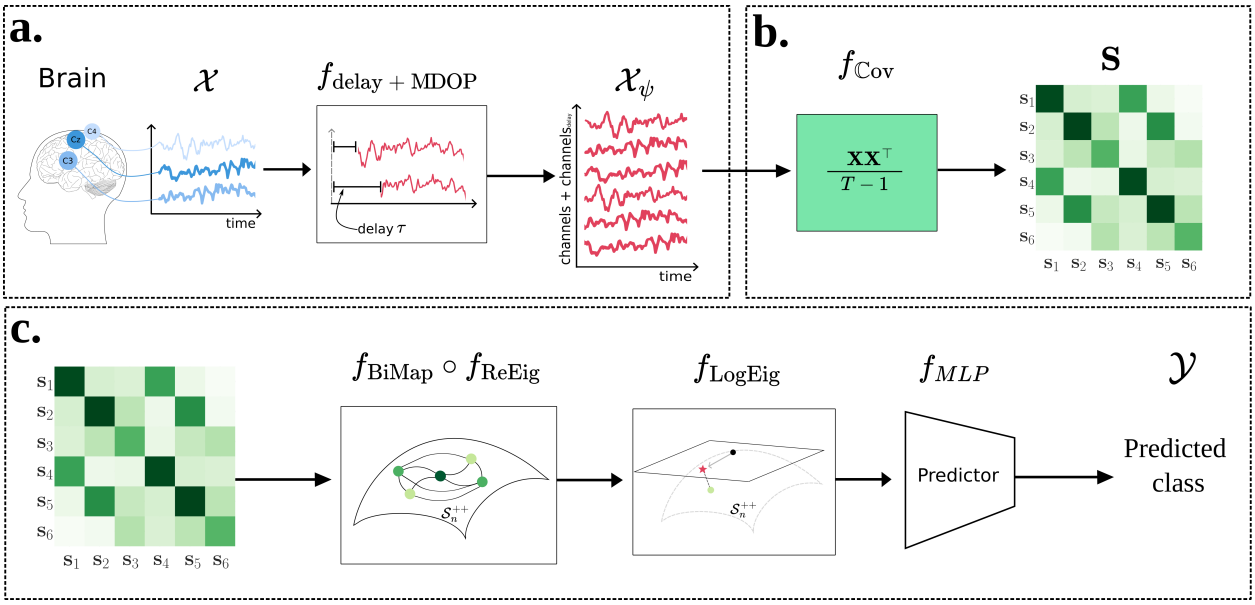


Figure 1: *Overview of our approach.* **A.** Phase space reconstruction for the EEG signal. For each trial from $\mathbf{X}$, we apply the function f_{delay} to reconstruct the phase space. During each evaluation fold, we adopted the MDOP algorithm to estimate suitable function parameters for embedding dimension ψ and the embedding delay τ. For illustrative purposes, this figure highlights the central cerebral cortex electrodes. **B.** Conversion of the phase space time-series with $f_{\mathbb{C}\text{ov}}$ for the covariance representation of the Symmetric Positive Definite Space $\mathcal{S}^n_{++}$, represented in green. **C.** The covariance feature space is then used to train the Symmetric Positive Definite Neural Network with BiMap, ReEig, LogEig, and MLP layers. Once the representation returns to the Euclidean space (red), after the LogEig, we adjust a fully connected layer to predict the associate label y for each trial.

channel to a zero mean and a unit standard deviation. Additionally, we opted to use the complete epoch duration, although this duration varies among datasets, as shown in Table 1.

We concentrate on a binary classification task, aiming to discern between imagined movements of the right and left hand using an intra-subject/within-session evaluation, which is based on a $5-$fold cross-validation is conducted in each session.

Additionally, we focus on investigating our algorithm's robustness with fewer electrode scenes. For this purpose, we selected a consistent set of three electrodes (C3, Cz, and C4) across datasets, all adhering to the 10-20 standard montage. Importantly, these electrodes are situated directly above the motor cortex, an area concentrating the most relevant information for MI-BCI classification [69].

3.5 Baseline Comparison

We compare the performance of our pipeline to that of several state-of-the-art DL Neural Networks used for BCI. All algorithms in our study were tested using the same set of three electrodes to maintain uniformity in our experimental approach. To standardize the models' parameters, we initially resample the input time series to align the signal with the state of the art in order to use the parameter's architecture of the original paper.

1. **ShallowNet** [14], a neural network architecture with independent spatial and temporal convolution steps with Relu activation, with standardized and re-sampled EEG signal at 250Hz.
2. **DeepNet** [14], an approach similar to ShallowNet with a deeper linear layer, with standardized and re-sampled EEG signal at 250Hz.
3. **EEGNet** [18], have a depth-wise convolutional layer functions as a spatial filter across channels, complemented by a separable convolution layer for features extraction designed for EEG with a sample frequency of 128Hz.
4. **EEGTCNet** [19], is a neural network that wrapper the EEGNet and includes a Temporal Convolution Network over the embedded representation, with standardized and re-sampled EEG signal at 250Hz.

5. **EEGITNet** [20], a neural network inspired by InceptionNet with parallel convolution layers with different scales. The parameters network has been designed to EEG signals at 128Hz.

6. **EEGNeX** [21], a neural network inspired by EEGNet incorporation the key components from ConvNeXt. The parameters network has been designed to EEG signals at 128Hz.

These methods were implemented and trained using MOABB version 1.0 [10]. For in-depth information about the deep learning hyper-parameters, please consult Table A2. In order to obtain a fair time comparison, all the following results are computed on the same hardware, a Dell C6420 dual-Xeon Cascade Lake SP Gold 6240 @ 2.60GHz.

4 Results

In the following section, we analyze the results produced using within-session (WS) evaluation. This method is based on a 5-fold cross-validation performed on each session independently. In particular, we report two separate analyses.

The first analysis is dedicated to contrasting the performance of SPDNet$_\psi$, with a specific focus on the exploitation of the covariance feature, against the state-of-the-art techniques in DL applied in EEG. The subsequent analysis, reported in the Appendix, shifts the focus towards the exploration of coherence as a feature for classification. Despite the ongoing investigations into coherence, we deliberately separated this analysis due to its comparatively lesser performance against the benchmarks achieved with covariance features. Nevertheless, the augmentation procedure is a significative methodology to improve the performance of both feature sets.

For the first analysis, we tested the performance of SPDNet$_\psi$ against the baseline comparison defined in 3.5 across a broad spectrum of datasets. The results are listed in Table 2, while the detailed statistical analysis can be found in Figure 2.

Table 2 shows ROC AUC results for right-hand versus left-hand classification, where bold numbers represent the best score in each dataset, revealing a consistent trend wherein our algorithm SPDNet$_\psi$ surpasses the DL state-of-the-art in the context of reduced datasets. This behavior is true for all datasets, with the notable exception of the Schirrmeister2017 dataset, where it is marginally surpassed by ShallowNet and DeepNet. It is important to note that the results of the latter two, however, are not statistically significant for this dataset. As the number of subjects per dataset is too small to reach decent statistical power for investigating the differences between pipelines, the results across datasets are aggregated to conduct a meta-analysis. Our SPDNet$_\psi$ yields statistically better results than all other approaches as shown in Fig. 2 (b), where only significant interactions are displayed ($p > 0.0.5$), the reported values and color show the t-values.

The second relevant point is to see the impact of the augmentation procedure compared to the standard SPDNet. Figure 2 (a) elucidates this aspect by illustrating the relative performance enhancements of various models against the SPDNet standard. Notably, ShallowConvNet and DeepConvNet exhibit positive performance increments across the majority of the datasets examined. However, the augmentation procedure, as implemented by SPDNet$_\psi$, not only consistently yields positive relative improvement against the standard SPDNet across all datasets but also the improvement introduced by the augmentation procedure turns out to be so significant that it even outperforms methods that initially surpassed the SPDNet benchmark. This improvement is not only demonstrated by the relative improvement with respect to the SPDNet standard but also is evident through the meta-analysis comparison(refer to Fig 2 (c), (d) and (e)).

Ultimately, the statistical analysis conducted using Figure 2 (b) allows us to demonstrate that SPDNet$_\psi$ outperforms the state-of-the-art DL methodology in a reduced dataset context. Its versatility and reliability, particularly in data-limited scenarios, position the method as a promising solution for real-world applications.

Table 2: Average AUC-ROC (%) on six datasets (BCNI2014-01, BCNI2014-04, Cho2017, Schirrmeister2017, Weibo2014, and Zhou2016), for a left vs. right motor imagery task. The highest performances are in bold-face.

Models	**BNCI2014001**	**BNCI2014004**	**Cho2017**	**Schirrmeister2017**	**Weibo2014**	**Zhou2016**
DeepNet [14]	75.80 ± 15.45	72.80 ± 19.48	63.13 ± 14.45	73.04 ± 15.67	73.97 ± 18.07	91.74 ± 7.00
ShallowNet [14]	75.85 ± 15.41	72.17 ± 18.61	64.14 ± 13.03	**73.59 ± 15.19**	75.36 ± 15.69	88.03 ± 8.55
EEGNet [18]	70.64 ± 19.87	70.27 ± 18.91	60.23 ± 14.98	69.80 ± 16.51	71.94 ± 17.80	88.95 ± 7.84
EEGTCNet [19]	65.98 ± 17.25	66.86 ± 18.41	56.43 ± 12.31	67.87 ± 17.36	65.94 ± 15.69	81.47 ± 11.66
EEGITNet [20]	66.64 ± 13.98	64.93 ± 14.49	54.68 ± 11.97	62.98 ± 16.19	56.97 ± 17.66	72.82 ± 12.78
EEGNeX [21]	68.86 ± 17.27	68.29 ± 17.85	56.64 ± 12.83	64.26 ± 17.58	58.73 ± 19.40	80.25 ± 15.55
SPDNet [8]	71.02 ± 15.64	70.15 ± 16.85	59.95 ± 12.61	67.40 ± 13.01	67.04 ± 17.62	88.85 ± 8.05
SPDNet$_\psi$ (Our)	**75.98 ± 17.00**	**80.46 ± 16.64**	**66.00 ± 13.29**	72.02 ± 13.07	**78.01 ± 19.64**	**94.92 ± 3.34**

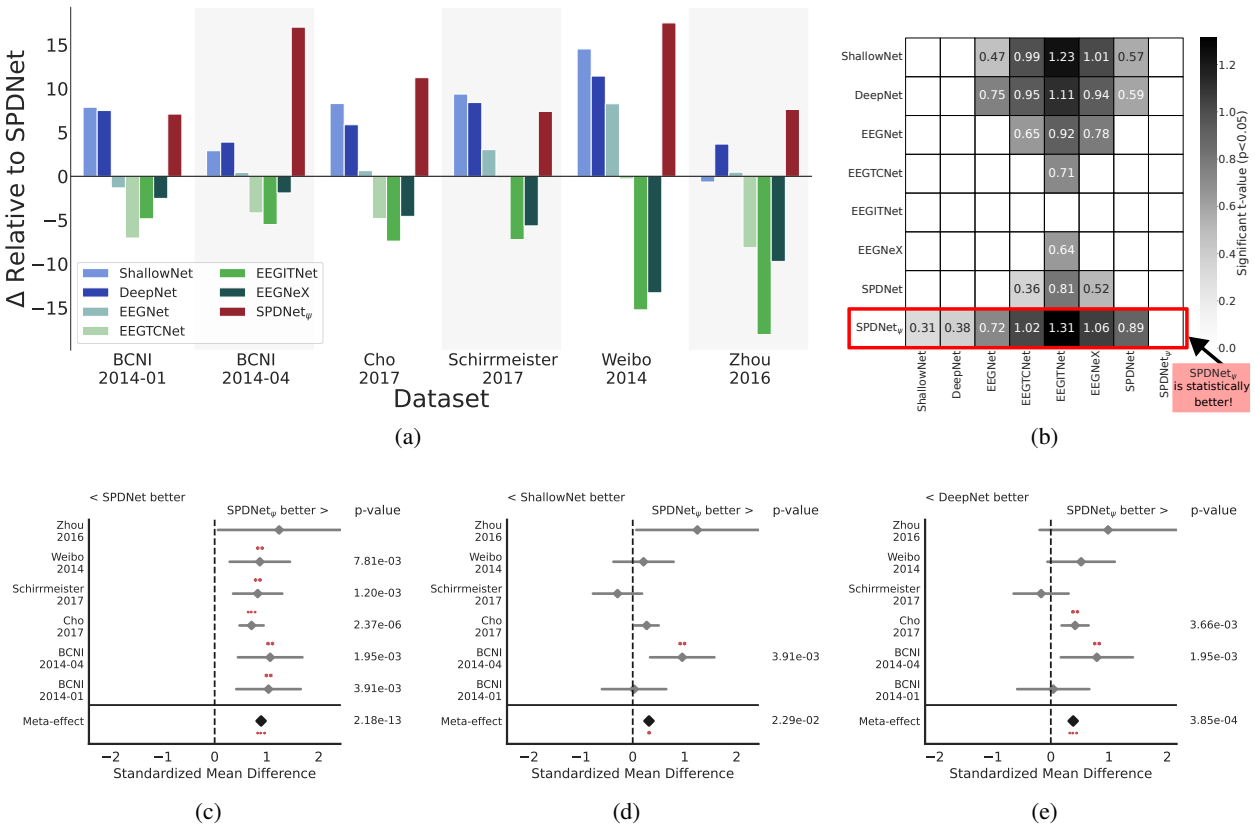


Figure 2: Results for Right vs Left-hand classification, using Within-Session evaluation. Plot (a) provides the relative improvement of the AUC-ROC in percentage of the method considered with respect to the standard SPDNet of the different pipelines considered. Plot (b) shows a combined meta-analysis (over all datasets) of the different pipelines. It shows the significance of the algorithm on the y-axis being better than the one on the x-axis. The gray level represents the significance level of the ROC-AUC difference in terms of t-values. We only show significant interactions ($p < 0.05$). Plots (c), (d), and (e) show the meta-analysis of SPDNet$_\psi$against SPDNet, ShallowNet, and DeepNet, respectively. We show the standardized mean differences of p-values computed as a one-tailed Wilcoxon signed-rank test for the hypothesis given in the plot title. The **gray** bar denotes the 95% interval. * stands for $p < 0.05$, ** for $p < 0.01$, and *** for $p < 0.001$.

5 Discussion

5.1 Impact of augmentation on most and least responsive 5 subject of Cho2017

In the previous section, we explored the outcome obtained by SPDNet$_\psi$ within the context of a reduced number of electrodes. Now, our attention shifts to a deeper exploration of this new algorithm, with a specific emphasis on the interpretability and computational performance of the method. To facilitate this comprehensive analysis, still in the context of three electrodes, we have chosen to concentrate on the examination of the five most and least responsive subjects within the Cho2017 dataset (the wider dataset in terms of the number of subjects), selected using the Minimum Distance to the Mean (MDM) algorithm [55] with covariance as a feature.

To analyze the performance, we use different pipelines, MDM [55], AugMDM [7], SPDNet, and SPDNet$_\psi$, each offering a distinct perspective on the impact of augmentation techniques on both traditional Machine Learning (ML) and advanced Deep Learning (DL) algorithms. Furthermore, we assess the performance using three different SPD features: covariance, instantaneous coherence, and imaginary coherence (Figure 3).

For the five most responsive subjects, covariance stands out as the top-performing feature as it could be expected. However, in the case of the five least responsive subjects, this dominance of covariance is less clear when considering the state-of-the-art methodology.

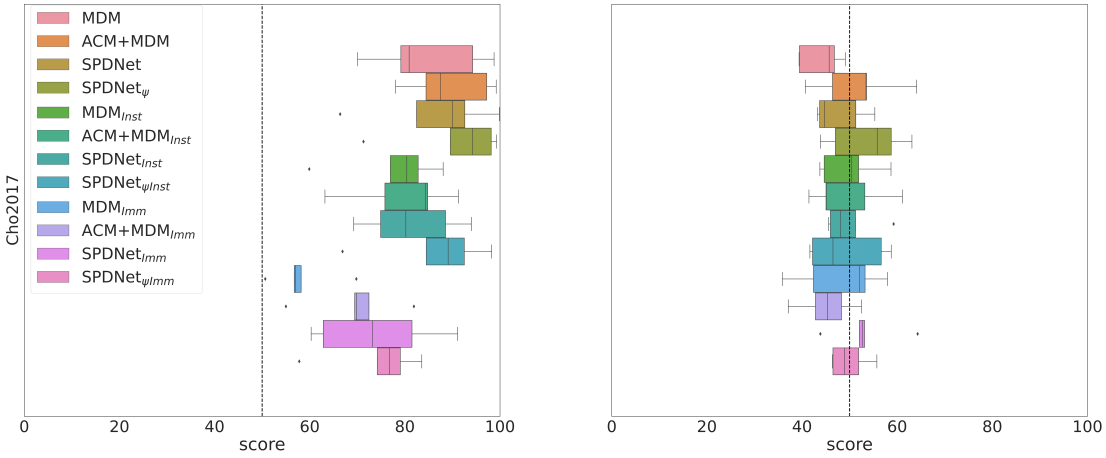


Figure 3: (a) Plot showing the box plot of the five most responsive subjects (3, 14, 35, 41, 43) of Cho2017. We see that the augmentation procedure consistently leads to notable improvements in the classification performance for all the SPD estimators considered.(b) Plot showing the box plot of the least responsive five subjects (2, 7, 29, 34, 50) of Cho2017. We see that the augmentation procedure increases the performance for the covariance feature, while the improvement is less clear for the FC estimator.

The outcomes of our study are in concordance with the state-of-the-art [9]: results achieved using Covariance as SPD feature outperform those obtained with the Functional Connectivity (FC) estimator.

It becomes evident that the augmentation procedure consistently leads to notable improvements in the classification performance for the top-performing five subjects. However, the situation becomes less definitive for the least responsive five subjects, where the augmentation effect is not as pronounced or evident. In this case, their signals exhibit a lower SNR, which can substantially impact the MDOP algorithm, leading to a poor estimation of the hyper-parameter. In this case, the augmentation procedure achieves a very positive impact on covariance while there is even a decrease in performance using coherence.

5.2 Interpretability

Creating explainable algorithms is crucial in the field of DL, and it holds even greater significance in healthcare data. In healthcare, understanding why an algorithm makes specific decisions is of great importance for clinicians and patients, ensuring transparency, accountability, bias mitigation, education, and trust-building in the collaboration between AI systems and medical experts.

For analyzing the explainability of the SPDNet$_{\psi}$ we use the GradCam++ algorithm [70], a technique designed for visualizing and explaining the decision-making process of neural networks. The GradCam++ output is essentially a visual representation that reveals which elements within the data are the most important for the model. GradCam++ has proven to be a robust method for spatial interpretability of EEG signals in a class-agnostic manner [71].

In our analysis, we focus on Subject 41 from the Cho2017 dataset, which is notable for being among the top 5 subjects that exhibit the most substantial performance improvement through the augmentation procedure. Similar findings also hold for other subjects. In particular, using covariance as a feature, we show a sample of the test dataset that was correctly classified by the SPDNet$_{\psi}$ algorithm (left hand classified as left hand) but was misclassified by SPDNet 4 using the same feature. Similar findings have been observed for the FC estimators. This figure represents the output of the GradCam++ applied to the output of the ReEig layer.

In all three scenarios, the standard SPDNet places more emphasis on the diagonal terms of SPD matrices. On the other side, when employing SPDNet$_{\psi}$, a more comprehensive picture unfolds. While diagonal terms continue to play a pivotal role, what becomes increasingly evident is the significant contribution of the off-diagonal terms. In essence, this observation emphasizes that the SPDNet$_{\psi}$ model leverages both individual electrode characteristics (diagonal terms) and the interplay between electrodes in time and space (off-diagonal terms) as crucial elements in its classification decision-making process.

To delve deeper into the interpretability of our proposed network, we employ a paired t-test to highlight statistically significant differences in the augmented covariance matrix (ACM) between left and right hands. Our analysis centers

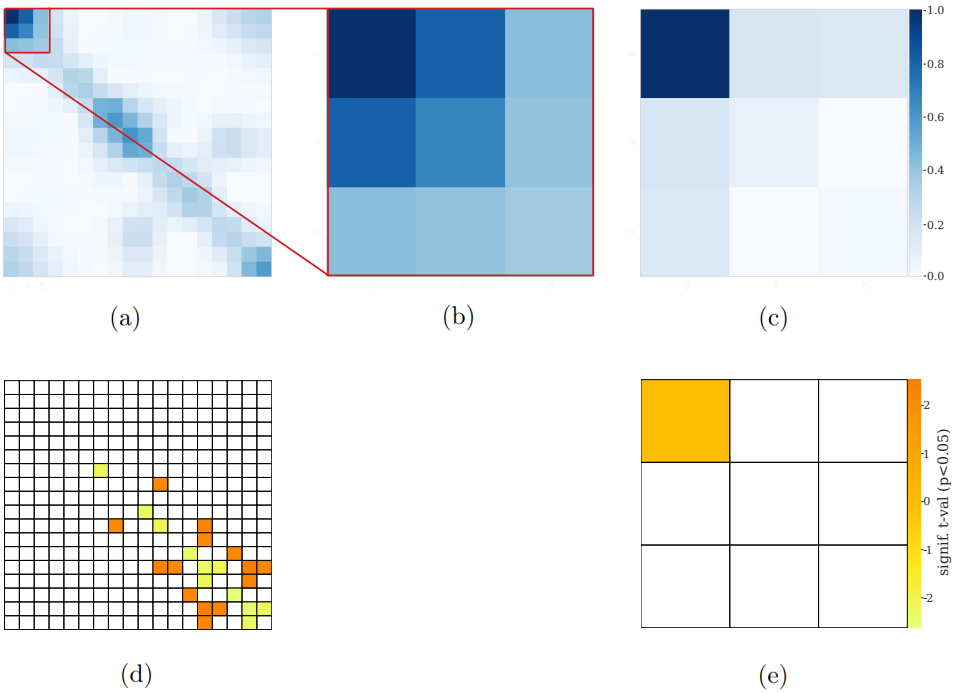


Figure 4: Interpretability parallel between GradCam++ and t-test between right hand left-hand classification for subject 41 of Cho2017. Plot (a) is the GradCam++ visualization for Augmented covariance. Plot (b) is a zoom in the region corresponding to normal Covariance. Plot (c) is the GradCam++ visualization for covariance. Plot (d) is the t-test of the Augmented covariance matrix. Plot (e) is the t-test of the covariance matrix. The t-test visualization represents only the significant interactions ($p < 0.05$).

on subject 41 from Cho2017, specifically on the test dataset of the initial fold of the cross-validation, with consistent findings observed in the other folds. Given that 20 samples are available for each imagined movement, we are in the domain of application of the t-test.

Examining Figure 4 (d) and (e), it is evident that SPDNet places emphasis primarily on the statistically significant pair of electrodes, showing a clear parallel between the GradCam++ visualization and the t-test. In contrast, SPDNet$_\psi$ not only considers significant interactions but also extends its focus to nearly significant interactions.

5.3 Convergence Behavior

To analyze convergence behavior, we again consider subject 41 of the Cho2017 dataset. Again, to get a fair comparison, all the following results are calculated on the same hardware, a Dell C6420 dual-Xeon Cascade Lake SP Gold 6240 @ 2.60GHz.

We examine the outcomes concerning both training and validation metrics, including loss and Receiver Operating Characteristic Area Under the Curve (ROC AUC) scores. The consolidated results are visually presented in Figure 5, wherein the graphs depict the average values across the considered 5-fold cross-validation.

Notably, our observations reveal that the loss function of SPDNet$_\psi$ converges to zero at a notably accelerated pace and exhibits a considerably more stable trajectory compared to standard SPDNet. Moreover, while the loss function of SPDNet$_\psi$ steadily approaches zero, the standard SPDNet experiences a more gradual descent, deviating from the path to complete convergence but ultimately plateauing around 0.4. These findings highlight that the capacity of SPDNet is limited and that the algorithm stops learning at a certain point.

Interestingly, when examining the ROC AUC graph, we observe that the validation scores of SPDNet$_\psi$ converge consistently to higher values. Notably, this convergence is achieved even faster compared to the SPDNet standard, requiring only 94% of the iterations. This implies that SPDNet$_\psi$ attains a comparable performance to SPDNet but with a significantly reduced number of iterations, making it an efficient and effective choice for real-life applications.

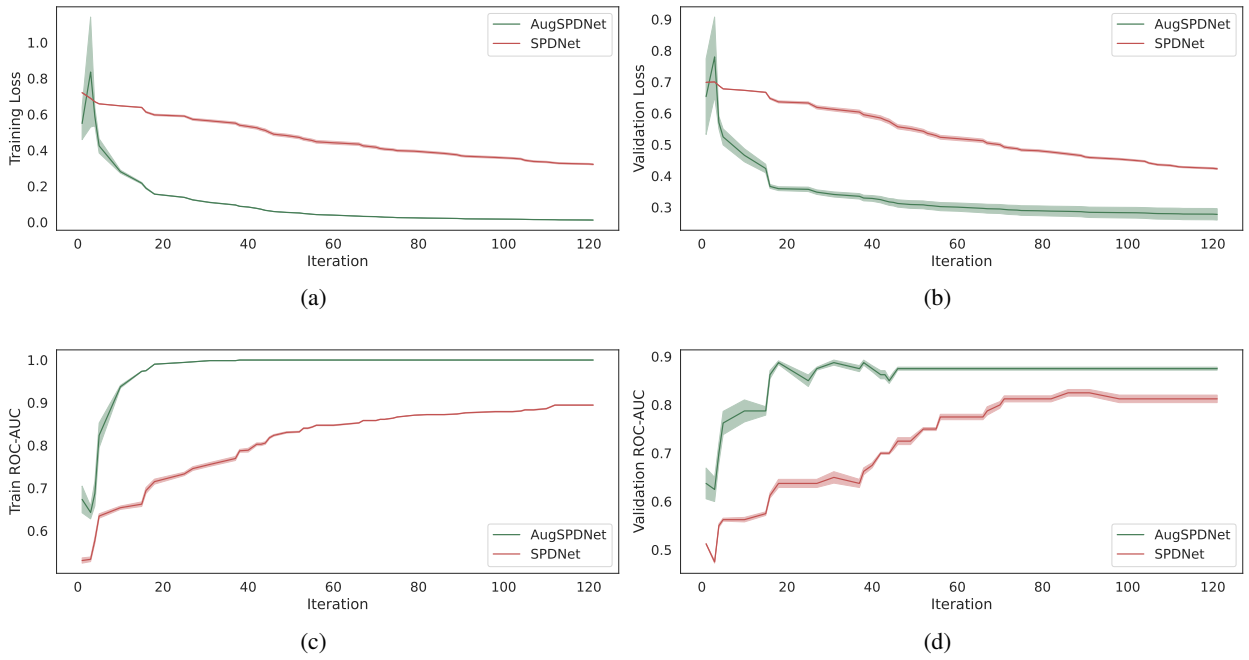


Figure 5: Convergence behavior for right-hand vs. left-hand classification using WS evaluation. The plot uses subject 41 of the Cho2017 dataset, showing the mean value for each plot computed over the 5-Fold considered in the WS evaluation. Plot (a) provides the training loss, while plot (b) is the validation one. Plot (c) provides the ROC-AUC for the training phase, while plot (d) the one for the validation.

5.4 Computational Performances

In this section, we are interested in the analysis of computational performance, with a specific focus on two key aspects: execution time and environmental impact. The results on the best 5 subjects of Cho2017 (subjects 3, 14, 35, 41, 43) are presented in Table 3.

Recently, much attention has begun to be given to the sustainability of ML and AI algorithms [72]. Since more and more models continue to proliferate and demand substantial computational resources, it is crucial to estimate and mitigate the energy consumption and emissions associated with their training and deployment. To assess the environmental impact of our algorithm, we use the CodeCarbon [73] library. CodeCarbon is an open-source library that enables developers to monitor carbon dioxide (CO_2) emissions, offering a way to analyze the carbon footprint of our work.

Interestingly our analysis shows that the SPDNet$_\psi$exhibits the longer computational time among the evaluated algorithms while simultaneously standing out as one of the most environmentally friendly in terms of its CO_2 impact (see Fig 6). This intriguing juxtaposition can be attributed to the fact that, during each fold of the cross-validation process, there's a necessity to optimize the model's hyper-parameters, which naturally augments the computational time. However, the MDOP algorithm consumes fewer resources than gradient descent procedures, thus resulting in a comparatively smaller environmental impact. In fact, the use of augmented procedure introduces the complication of two hyper-parameters but nevertheless manages to keep the number of trainable parameters of the model contained (see Table 4). It's plausible that our algorithm has yet to reach its optimal performance. Further investigation is needed to improve the optimization strategies.

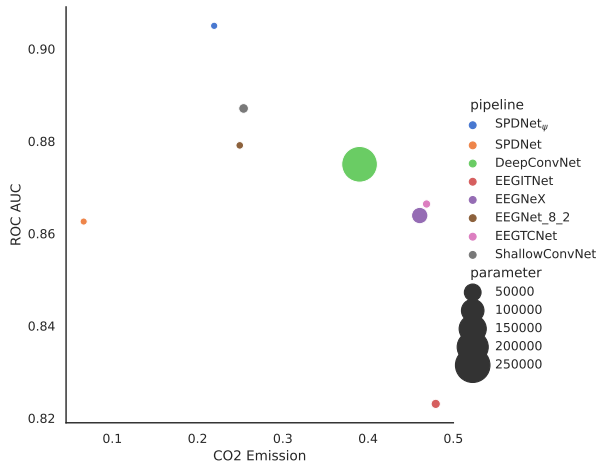


Figure 6: Performance (ROC AUC) versus CO_2 consumption (g CO_2 equivalent). The size of the blob indicates the number of trainable parameters.

	Avg. Time (s)	CO_2 Emission (g CO_2)
DeepNet	21.77	0.39
ShallowNet	15.10	0.25
EEGNet-8,2	14.96	0.25
EEGITNet	28.44	0.48
EEGTCNet	28.83	0.47
EEGNeX	28.15	0.46
SPDNet	6.80	0.07
SPDNet$_\psi$	30.16	0.22

Table 3: Average computational time and CO_2 equivalent consumption over the best five subjects of Cho2017

Stage	Layer	Output	Parameters
Input		[1, 3, 1537]	0
Augmentation	AugmentedDataset	[1, 18, 1393]	0
Covariance	Covariances	[1, 18, 18]	0
SPDNet	BiMap	[1, 9, 9]	162
	ReEig	[1, 9, 9]	0
	LogEig	[1, 45]	0
Classification	Linear	[1, 2]	45
Total			207

Table 4: Architecture of SPDNet$_\psi$ with parameter number depending on ϕ. Here, it is reported for the specific case of order $\phi = 6$.

5.5 Limitation and Future Directions

The existing SPDNet$_\psi$, as employed thus far, utilizes MDOP for hyper-parameter selection, offering computational efficiency compared to grid-search methods [7]. However, the solutions obtained with MDOP were sub-optimal compared with using a more extensive hyper-parameter selection. A potential avenue for future exploration involves streamlining and accelerating the grid-search process to harness the capabilities of the SPDNet$_\psi$ architecture fully.

Further enhancements could be achieved by adopting more intricate SPDNet architectures, such as incorporating batch normalization, exploring different BiMap layers, or experimenting with a Bottleneck architecture.

An intriguing direction arises from the insights gained in the explainability study, revealing that SPDNet$_\psi$ predominantly focuses on diagonal elements. This suggests the possibility of employing a Region of Interest approach, where the network initially prioritizes the main diagonals while treating the remainder of the matrix as background.

Continued improvements could be realized through comprehensive algorithm testing across diverse datasets, tasks, and evaluation procedures. This approach allows for a better understanding of performance in more complex scenarios, particularly those posed by intra- and inter-subject variabilities.

Using three electrodes only is advantageous for practical scenarios. In fact, this decision was strategically made to shorten the length of the experiment, thereby minimizing patient fatigue and maintaining their attention throughout the session.

Another challenge was the difficulty in classifying data from less responsive subjects. Despite our efforts, there remains a need for improved methods better to capture the subjects' intentions and their specific characteristics.

6 Conclusion

This study focuses on integrating the Augmented Covariance Matrix (ACM) with SPDNet to develop a practical algorithm intended for real-world applications, with a deliberate focus on scenarios involving a limited number of electrodes. Specifically, we restricted our electrode usage to just 3, strategically placed on the motor cortex.

In fact, the augmented methodology, based on Takens theorem, demonstrates a particularly effective performance when applied with a reduced number of electrodes. However, it is important to note that using only three electrodes can impact the results. For instance, in the BNCI2014001 dataset, employing just three electrodes led to a performance reduction of about 10% in the ShallowNet architecture compared to using all available electrodes.

Validation of our approach involves leveraging nearly 100 subjects from several open datasets, a task facilitated by the MOABB framework. The resulting SPDNet$_\psi$ algorithm outperforms existing DL algorithms in BCI-EEG classification and offers some explainability elements through GradCam++ visualization. Remarkably, this algorithm requires a modest number of trainable parameters and exhibits a lower environmental impact, measured by CO_2 equivalent emissions.

7 Acknowledgments

The work of IG and TG was partly funded by the EUR DS4H/Neuromod fellowship. The BA work was supported partly by CAPES under Grant 001 and by DATAIA Convergence Institute as part of the "Programme d'Investissement d'Avenir", (ANR-17-CONV-0003) operated by LISN. We extend our gratitude towards the OPAL infrastructure at Université Côte d'Azur for their essential resources and support. Additionally, acknowledgment is due for the support from the European Research Council (ERC) under the EU's Horizon 2020 research and innovation program (grant No. 864729), and the "Investissements d'avenir" program ANR-10-IAIHU-06.

8 Open-source availability

The code will be made public during publication time under the BSD-3 License.

References

[1] Janis J Daly and Jonathan R Wolpaw. Brain–computer interfaces in neurological rehabilitation. *The Lancet Neurology*, 7(11):1032–1043, 2008.

[2] Dong Wen, Yali Fan, Sheng-Hsiou Hsu, Jian Xu, Yanhong Zhou, Jianxin Tao, Xifa Lan, and Fengnian Li. Combining brain–computer interface and virtual reality for rehabilitation in neurological diseases: A narrative review. *Annals of physical and rehabilitation medicine*, 64(1):101404, 2021.

[3] Ctrl labs at Reality Labs, David Sussillo, Patrick Kaifosh, and Thomas Reardon. A generic noninvasive neuromotor interface for human-computer interaction. *bioRxiv*, pages 2024–02, 2024.

[4] Jianyang Deng and Yijia Lin. The benefits and challenges of chatgpt: An overview. *Frontiers in Computing and Intelligent Systems*, 2(2):81–83, 2022.

[5] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

[6] 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, 2019.

[7] Igor Carrara and Théodore Papadopoulo. Classification of BCI-EEG based on augmented covariance matrix. *arXiv preprint arXiv:2302.04508*, 2023.

[8] Zhiwu Huang and Luc Van Gool. A riemannian network for spd matrix learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

[9] Marie-Constance Corsi, Sylvain Chevallier, Fabrizio De Vico Fallani, and Florian Yger. Functional Connectivity Ensemble Method to Enhance BCI Performance (FUCONE). *IEEE Transactions on Biomedical Engineering*, 69 (9):2826–2838, 2022. doi: 10.1109/TBME.2022.3154885.

[10] Bruno Aristimunha, Igor Carrara, Pierre Guetschel, Sara Sedlar, Pedro Rodrigues, Jan Sosulski, Divyesh Narayanan, Erik Bjareholt, Barthelemy Quentin, Robin Tibor Schirrmeister, Emmanuel Kalunga, Ludovic Darmet, Cattan Gregoire, Ali Abdul Hussain, Ramiro Gatti, Vladislav Goncharenko, Jordy Thielen, Thomas Moreau, Yannick Roy, Vinay Jayaram, Alexandre Barachant, and Sylvain Chevallier. Mother of all bci benchmarks v1.0. `doi.org/10.5281/zenodo.10034223`, 2023. DOI: 10.5281/zenodo.10034223.

[11] Alexander J Casson. Wearable EEG and beyond. *Biomedical engineering letters*, 9(1):53–71, 2019.

[12] Jean-Rémi King, Laura Gwilliams, Chris Holdgraf, Jona Sassenhagen, Alexandre Barachant, Denis Engemann, Eric Larson, and Alexandre Gramfort. Encoding and Decoding Framework to Uncover the Algorithms of Cognition. In *The Cognitive Neurosciences*. The MIT Press, 05 2020. ISBN 9780262356176.

[13] Vinay Jayaram and Alexandre Barachant. MOABB: trustworthy algorithm benchmarking for BCIs. *Journal of neural engineering*, 15(6):066011, 2018.

[14] 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*, aug 2017. ISSN 1097-0193. doi: 10.1002/hbm.23730.

[15] Cédric Rommel, Joseph Paillard, Thomas Moreau, and Alexandre Gramfort. Data augmentation for learning predictive models on EEG: a systematic comparison. *Journal of Neural Engineering*, 19(6):066020, 11 2022. doi: 10.1088/1741-2552/aca220.

[16] Philipp Bomatter, Joseph Paillard, Pilar Garces, Jörg Hipp, and Denis Engemann. Machine learning of brain-specific biomarkers from EEG. *bioRxiv*, pages 2023–12, 2023. doi: 10.1101/2023.12.15.571864.

[17] Yohann Benchetrit, Hubert Banville, and Jean-Rémi King. Brain decoding: toward real-time reconstruction of visual perception. In *The Twelfth International Conference on Learning Representations*, 2024.

[18] 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.

[19] Thorir Mar Ingolfsson, Michael Hersche, Xiaying Wang, Nobuaki Kobayashi, Lukas Cavigelli, and Luca Benini. EEG-TCNet: An accurate temporal convolutional network for embedded motor-imagery brain–machine interfaces. In *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2958–2965. IEEE, 2020.

[20] Abbas Salami, Javier Andreu-Perez, and Helge Gillmeister. EEG-ITNet: An explainable inception temporal convolutional network for motor imagery classification. *IEEE Access*, 10:36672–36685, 2022.

[21] Xia Chen, Xiangbin Teng, Han Chen, Yafeng Pan, and Philipp Geyer. Toward reliable signals decoding for electroencephalogram: A benchmark study to EEGNeX. *arXiv preprint arXiv:2207.12369*, 2022.

[22] Zhiwu Huang and Luc Van Gool. A Riemannian Network for SPD Matrix Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1), Feb. 2017.

[23] Yoon-Je Suh and Byung Hyung Kim. Riemannian Embedding Banks for Common Spatial Patterns with EEG-based SPD Neural Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(1):854–862, May 2021.

[24] Yue-Ting Pan, Jing-Lun Chou, and Chun-Shu Wei. MAtt: A Manifold Attention Network for EEG Decoding. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[25] Reinmar Kobler, Jun-ichiro Hirayama, Qibin Zhao, and Motoaki Kawanabe. SPD domain-specific batch normalization to crack interpretable unsupervised domain adaptation in EEG. *Advances in Neural Information Processing Systems*, 35:6219–6235, 2022.

[26] Rui Wang, Xiao-Jun Wu, Ziheng Chen, Tianyang Xu, and Josef Kittler. DreamNet: A Deep Riemannian Manifold Network for SPD Matrix Learning. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 3241–3257, December 2022.

[27] Ce Ju and Cuntai Guan. Tensor-CSPNet: A Novel Geometric Deep Learning Framework for Motor Imagery Classification. *IEEE Transactions on Neural Networks and Learning Systems*, 34:10955–10969, 2022.

[28] Byung Hyung Kim, Jin Woo Choi, Honggu Lee, and Sungho Jo. A discriminative SPD feature learning approach on Riemannian manifolds for EEG classification. *Pattern Recognition*, 143:109751, 2023. ISSN 0031-3203.

[29] Jianchao Lu, Yuzhe Tian, Yang Zhang, Jiaqi Ge, Quan Z. Sheng, and Xianglin Zheng. LGL-BCI: A Lightweight Geometric Learning Framework for Motor Imagery-Based Brain-Computer Interfaces. *ArXiv*, abs/2310.08051, 2023.

[30] Ce Ju and Cuntai Guan. Graph Neural Networks on SPD Manifolds for Motor Imagery Classification: A Perspective From the Time–Frequency Analysis. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–15, 2023. doi: 10.1109/TNNLS.2023.3307470.

[31] Rui Wang, Xiao-Jun Wu, Tianyang Xu, Cong Hu, and Josef Kittler. U-SPDNet: An SPD manifold learning-based neural network for visual classification. *Neural Networks*, 161:382–396, 2023. ISSN 0893-6080.

[32] Zhen Peng, Hongyi Li, Di Zhao, and Chengwei Pan. Reducing the Dimensionality of SPD Matrices with Neural Networks in BCI. *Mathematics*, 11(7), 2023. ISSN 2227-7390.

[33] Yunbo Tang, Dan Chen, Jia Wu, Weiping Tu, Jessica J.M. Monaghan, Paul Sowman, and David Mcalpine. Functional connectivity learning via Siamese-based SPD matrix representation of brain imaging data. *Neural Networks*, 163:272–285, 2023. ISSN 0893-6080. doi: 10.1016/j.neunet.2023.04.004.

[34] Daniel Wilson, Robin Tibor Schirrmeister, Lukas Alexander Wilhelm Gemein, and Tonio Ball. Deep Riemannian Networks for EEG Decoding. *ArXiv*, abs/2212.10426, 2022.

[35] Ce Ju, Dashan Gao, Ravikiran Mane, Ben Tan, Yang Liu, and Cuntai Guan. Federated Transfer Learning for EEG Signal Classification. In *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 3040–3045, 2020. doi: 10.1109/EMBC44109.2020.9175344.

[36] 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.

[37] Yunchen Li, Zhou Yu, Gaoqi He, Yunhang Shen, Ke Li, Xing Sun, and Shaohui Lin. SPD-DDPM: Denoising Diffusion Probabilistic Models in the Symmetric Positive Definite Space. *arXiv preprint arXiv:2312.08200*, 2023.

[38] Nic Fishman, Leo Klarner, Valentin De Bortoli, Emile Mathieu, and Michael Hutchinson. Diffusion Models for Constrained Domains. *arXiv preprint arXiv:2304.05364*, 2023.

[39] Kay A. Robbins, Jonathan Touryan, Tim Mullen, Christian Kothe, and Nima Bigdely-Shamlo. How Sensitive Are EEG Results to Preprocessing Methods: A Benchmarking Study. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28(5):1081–1090, 2020. doi: 10.1109/TNSRE.2020.2980223.

[40] Alexandre Gramfort, Martin Luessi, Eric Larson, Denis Engemann, Daniel Strohmeier, Christian Brodbeck, Roman Goj, Mainak Jas, Teon Brooks, Lauri Parkkonen, and Matti Hämäläinen. MEG and EEG data analysis with MNE-Python. *Frontiers in Neuroscience*, 7:267, 2013. ISSN 1662-453X. doi: 10.3389/fnins.2013.00267.

[41] Mainak Jas, Denis Alexander Engemann, Yousra Bekhti, Federico Raimondo, and Alexandre Gramfort. Autoreject: Automated artifact rejection for MEG and EEG data. *NeuroImage*, 159:417–429, 2016.

[42] Pierre Ablin, Jean-François Cardoso, and Alexandre Gramfort. Faster ICA under orthogonal constraint. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4464–4468. IEEE, 2018.

[43] Fabien Lotte, Laurent Bougrain, Andrzej Cichocki, Maureen Clerc, Marco Congedo, Alain Rakotomamonjy, and Florian Yger. A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update. *Journal of neural engineering*, 15(3):031005, 2018.

[44] 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.

[45] Yen-Lin Chen, Yuan Chiang, Pei-Hsin Chiu, I-Chen Huang, Yu-Bai Xiao, Shu-Wei Chang, and Chang-Wei Huang. High-dimensional phase space reconstruction with a convolutional neural network for structural health monitoring. *Sensors*, 21(10):3514, 2021.

[46] Xueling Zhou, Bingo Wing-Kuen Ling, Waqar Ahmed, Yang Zhou, Yuxin Lin, and Hongtao Zhang. Multivariate phase space reconstruction and Riemannian manifold for sleep stage classification. *Biomedical Signal Processing and Control*, 88:105572, 2024. ISSN 1746-8094. doi: 10.1016/j.bspc.2023.105572.

[47] Floris Takens. Detecting strange attractors in turbulence. In *Dynamical systems and turbulence, Warwick 1980*, pages 366–381. Springer, 1981.

[48] Florias Takens. Detecting Nonlinearities In Stationary Time Series. *International Journal of Bifurcation and Chaos*, 03(02):241–256, 1993. doi: 10.1142/S0218127493000192.

[49] Lyle Noakes. The takens embedding theorem. *International Journal of Bifurcation and Chaos*, 01(04):867–872, 1991.

[50] Norman H Packard, James P Crutchfield, J Doyne Farmer, and Robert S Shaw. Geometry from a time series. *Physical review letters*, 45(9):712, 1980.

[51] Eugene Tan, Shannon Algar, Débora Corrêa, Michael Small, Thomas Stemler, and David Walker. Selecting embedding delays: An overview of embedding techniques and a new method using persistent homology. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 33(3):032101, 03 2023. ISSN 1054-1500. doi: 10.1063/5.0137223.

[52] Chetan Nichkawde. Optimal state-space reconstruction using derivatives on projected manifold. *Physical Review E*, 87(2):022905, 2013.

[53] Denis Le Bihan, Jean-François Mangin, Cyril Poupon, Chris A Clark, Sabina Pappata, Nicolas Molko, and Hughes Chabriat. Diffusion tensor imaging: concepts and applications. *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 13(4):534–546, 2001.

[54] Joachim Weickert and Hans Hagen. *Visualization and processing of tensor fields*. Springer Science & Business Media, 2005.

[55] Alexandre Barachant, Stéphane Bonnet, Marco Congedo, and Christian Jutten. Riemannian geometry applied to bci classification. In *International conference on latent variable analysis and signal separation*, pages 629–636. Springer, 2010.

[56] Wolfgang Förstner and Boudewijn Moonen. A metric for covariance matrices. *Geodesy-the Challenge of the 3rd Millennium*, pages 299–309, 2003.

[57] Maher Moakher. A differential geometric approach to the geometric mean of symmetric positive-definite matrices. *SIAM journal on matrix analysis and applications*, 26(3):735–747, 2005.

[58] Vincent Arsigny, Pierre Fillard, Xavier Pennec, and Nicholas Ayache. Geometric means in a novel vector space structure on symmetric positive-definite matrices. *SIAM journal on matrix analysis and applications*, 29(1): 328–347, 2007.

[59] Alexandre Barachant, Quentin Barthélemy, Jean-Rémi King, Alexandre Gramfort, Sylvain Chevallier, Pedro L. C. Rodrigues, Emanuele Olivetti, Vladislav Goncharenko, Gabriel Wagner vom Berg, Ghiles Reguig, Arthur Lebeurrier, Erik Bjäreholt, Maria Sayu Yamamoto, Pierre Clisson, and et al. Marie-Constance Corsi. pyRiemann/pyRiemann: v0.5. `doi.org/10.5281/zenodo.7547583`, 2023. DOI: 10.5281/zenodo.7547583.

[60] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[61] Gary Bécigneul and Octavian-Eugen Ganea. Riemannian adaptive optimization methods. *arXiv preprint arXiv:1810.00760*, 2018.

[62] Max Kochurov, Rasul Karimov, and Serge Kozlukov. Geoopt: Riemannian optimization in pytorch. *arXiv preprint arXiv:2005.02819*, 2020.

[63] Michael Tangermann, Klaus-Robert Müller, Ad Aertsen, Niels Birbaumer, Christoph Braun, Clemens Brunner, Robert Leeb, Carsten Mehring, Kai J Miller, Gernot Mueller-Putz, et al. Review of the BCI competition IV. *Frontiers in neuroscience*, page 55, 2012.

[64] Robert Leeb, Felix Lee, Claudia Keinrath, Reinhold Scherer, Horst Bischof, and Gert Pfurtscheller. Brain–computer communication: motivation, aim, and impact of exploring a virtual apartment. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15(4):473–482, 2007.

[65] Hohyun Cho, Minkyu Ahn, Sangtae Ahn, Moonyoung Kwon, and Sung Chan Jun. EEG datasets for motor imagery brain–computer interface. *GigaScience*, 6(7):gix034, 2017.

[66] 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.

[67] 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):e114853, 2014.

[68] Bangyan Zhou, Xiaopei Wu, Zhao Lv, Lei Zhang, and Xiaojin Guo. A fully automated trial selection method for optimization of motor imagery based brain-computer interface. *PloS one*, 11(9):e0162657, 2016.

[69] Marc Jeannerod. Mental imagery in the motor context. *Neuropsychologia*, 33(11):1419–1432, 1995.

[70] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 839–847. IEEE, 2018.

[71] Akshay Sujatha Ravindran and Jose Contreras-Vidal. An empirical comparison of deep learning explainability approaches for EEG using simulated ground truth. *Scientific Reports*, 13(1):17709, 2023.

[72] Lynn H Kaack, Priya L Donti, Emma Strubell, George Kamiya, Felix Creutzig, and David Rolnick. Aligning artificial intelligence with climate change mitigation. *Nature Climate Change*, 12(6):518–527, 2022.

[73] Benoit Courty, Victor Schmidt, Goyal-Kamal, MarionCoutarel, Boris Feld, Jérémy Lecourt, SabAmine, kngoyal, Mathilde Léval, Alexis Cruveiller, ouminasara, Franklin Zhao, Aditya Joshi, Alexis Bogroff, Hugues de Lavoreille, Niko Laskaris, LiamConnell, Amine Saboni, Douglas Blank, Ziyao Wang, inimaz, Armin Catovic, Michał Stęchły, alencon, JPW, MinervaBooks, SangamSwadiK, Hervé M., brotherwolf, and Martin Pollard. mlco2/codecarbon: v2.2.7, July 2023. DOI: 10.5281/zenodo.8181237.

[74] Fabien Lotte and Cuntai Guan. Learning from other subjects helps reducing brain-computer interface calibration time. In *2010 IEEE International conference on acoustics, speech and signal processing*, pages 614–617. IEEE, 2010.

[75] Florian Yger, Maxime Berar, and Fabien Lotte. Riemannian approaches in brain-computer interfaces: a review. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(10):1753–1762, 2016.

A Results using Coherence as feature

In this section, we shift our focus to reporting results due to the use of Imaginary and Instantaneous coherence features as input. Specifically, we aim to draw a comparison between the SPDNet$_\psi$results and the non DL state-of-the-art pipelines (usually created for covariance but here applied on coherence). It's worth noting that, in the subsequent analysis, we abstain from comparing these results with the performance of state-of-the-art DL models, given that the employment of coherence-based features typically yields lower results across the board. Nevertheless, we find it particularly intriguing to present the outcomes achieved by SPDNet$_\psi$in the context of coherence-based features since we observe that SPDNet$_\psi$demonstrates a statistically significant improvement. This shows the potential and significance of SPDNet$_\psi$in the broader landscape of signal processing and pattern recognition.

1. **CohCSP+LDA** [74], a combination of the Common spatial pattern (CSP) algorithm followed by a classification performed on a shrinkage Linear Discriminant Analysis (LDA).
2. **Coh + TS + EN** [9], a Riemannian classification method in the tangent space using an Elastic Network as classifier.
3. **CohFgMDM** [75], a Riemannian classification method by Minimum Distance to the Mean after having applied a geodesic filtering.
4. **CohMDM** [55], a Riemannian classification method by Minimum Distance to the Mean.
5. **Coh + TS + SVM** [55], a Riemannian classification method in the tangent space using a Support Vector Machine (SVM).

Results can be found in Table A1 while the detailed statistical analysis can be found in Figures A1 and A2.

Table A1: Summary of performances via average AUC-ROC (%) on six datasets (BCNI2014-01, BCNI2014-04, Cho2017, Schirrmeister2017, Weibo2014, and Zhou2016), for a left vs. right motor imagery task. Bold numbers represent the best score in each dataset. Estimators Imaginary Coherence and Instantaneous Coherence

Models	Estimator	BNCI2014001	BNCI2014004	Cho2017	Schirrmeister2017	Weibo2014	Zhou2016
SPDNet	Imm	56.55±7.41	58.12±12.20	56.16±8.94	59.15±7.84	60.59±10.78	62.89±8.97
CSP+LDA	Imm	57.66±7.19	53.09±8.55	51.63±6.30	53.06±5.20	56.77±7.62	64.47±7.47
TANG+SVM	Imm	59.39±8.86	53.66±8.26	52.45±6.24	52.93±4.83	56.09±8.89	64.19±9.49
FgMDM	Imm	60.19±7.72	52.83±8.90	51.93±6.39	53.99±4.57	57.50±7.92	65.52±8.72
MDM	Imm	60.30±7.83	54.04±9.00	52.09±6.67	53.36±5.34	57.31±7.33	66.27±8.60
Cov+EN	Imm	60.71±7.92	52.67±8.96	52.07±6.70	54.42±4.07	57.21±8.23	64.58±8.41
SPDNet$_\psi$	Imm	**60.89±10.17**	**65.70±14.06**	**56.51±8.60**	**61.69±9.81**	**64.75±11.33**	**72.16±7.40**
SPDNet	Inst	63.42±5.76	63.71±12.88	57.61±11.09	59.73±6.89	66.11±12.02	75.07±7.63
MDM	Inst	69.74±14.19	59.89±10.72	58.76±9.65	52.01±5.77	63.24±14.25	81.94±8.91
TANG+SVM	Inst	69.82±14.76	60.44±9.76	58.86±10.11	53.31±5.42	63.10±15.51	85.02±8.29
CSP+LDA	Inst	70.49±14.80	60.13±10.23	59.23±10.40	51.94±6.06	64.32±15.92	84.95±8.94
Cov+EN	Inst	71.01±14.33	59.34±10.89	59.37±9.86	52.06±5.39	64.67±14.79	84.53±8.81
FgMDM	Inst	71.61±13.86	60.64±10.24	59.45±10.29	52.53±5.32	65.09±14.73	85.17±8.36
SPDNet$_\psi$	Inst	**71.84±16.75**	**68.83±17.41**	**62.04±12.02**	**65.37±12.91**	**68.31±14.17**	**87.03±6.13**

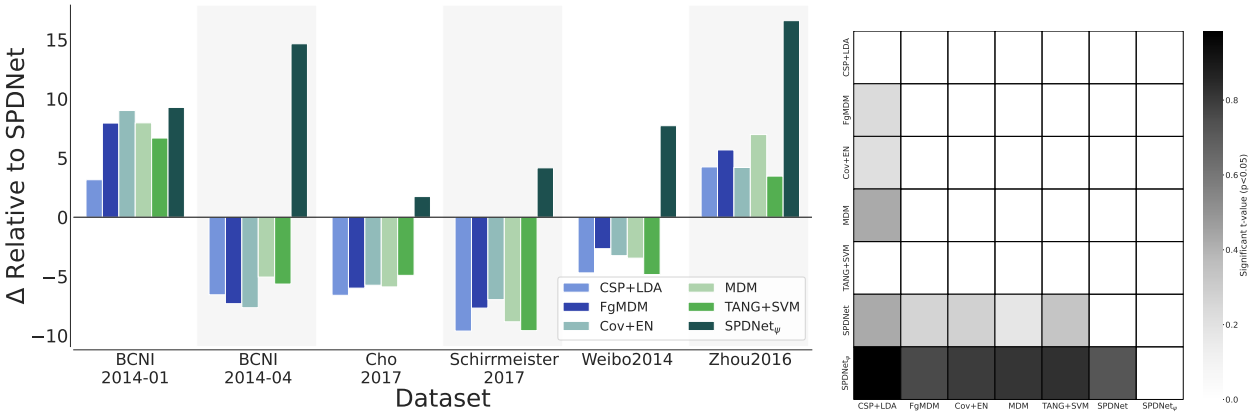
In this study, the situation is even rosier: SPDNet$_\psi$, used with either imaginary or instantaneous coherence features, emerges as the best pipeline across all datasets. When utilizing imaginary coherence, SPDNet alone already achieves statistically superior results compared to state-of-the-art approaches employing the same features. However, the augmentation procedure introduces a significant performance boost, a finding supported by the meta-effect analysis shown in Fig A1 (c).

In the case of instantaneous coherence, on the other hand, the impact of the augmentation procedure is even more significant. In fact, with this feature, the standard SPDNet fails to exhibit statistically superior performance in comparison to the state-of-the-art pipelines. SPDNet$_\psi$, on the other hand, stands out as the top-performing pipeline, showing that the augmentation procedure brings a statistically significant influence on classification outcomes (see Fig A2 (a)).

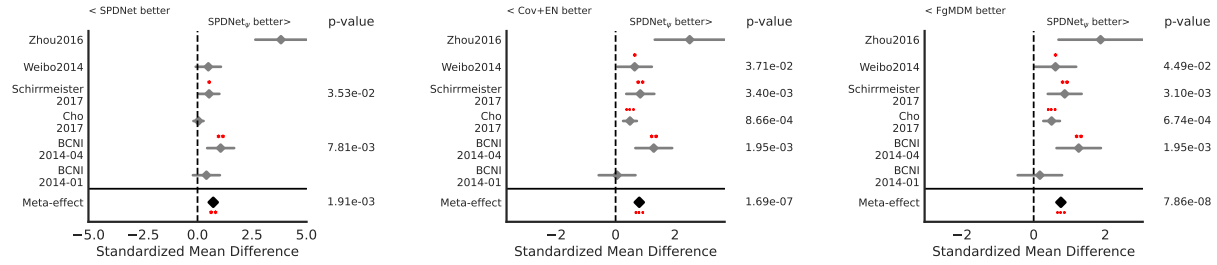
B Additional Results

Table A2: Parameters common to all DL pipelines.

Parameter	Value
Epoch	300
Batch Size	64
Validation Split	0.1
Loss	Sparse Categorical Crossentropy
Optimizer	Adam Learning Rate = 0.001
Callbacks ES	Early Stopping Patience = 75 Monitor = Validation Loss
Callbacks LR	ReduceLROnPlateau Patience = 75 Monitor = Validation Loss Factor = 0.5



(a) (b)



(c) (d) (e)

Figure A1: Result for right hand vs left hand classification, using within-session evaluation for state-of-the-art imaginary coherence pipelines. Plot (a) provides the relative improvement of the method considered with respect to the standard SPDNet of the different pipelines considered. Plot (b) shows a combined meta-analysis (over all datasets) of the different pipelines. It shows the significance of the algorithm on the y-axis being better than the one on the x-axis. The gray level represents the significance level of the ROC-AUC difference in terms of t-values. We only show significant interactions ($p < 0.05$). Plots (c), (d), and (e) show the meta-analysis of SPDNet$_\psi$against SPDNet, COV+EN, and FgMDM, respectively. We show the standardized mean differences of p-values computed as a one-tailed Wilcoxon signed-rank test for the hypothesis given in the plot title. The **gray** bar denotes the 95% interval. * stands for $p < 0.05$, ** for $p < 0.01$, and *** for $p < 0.001$.

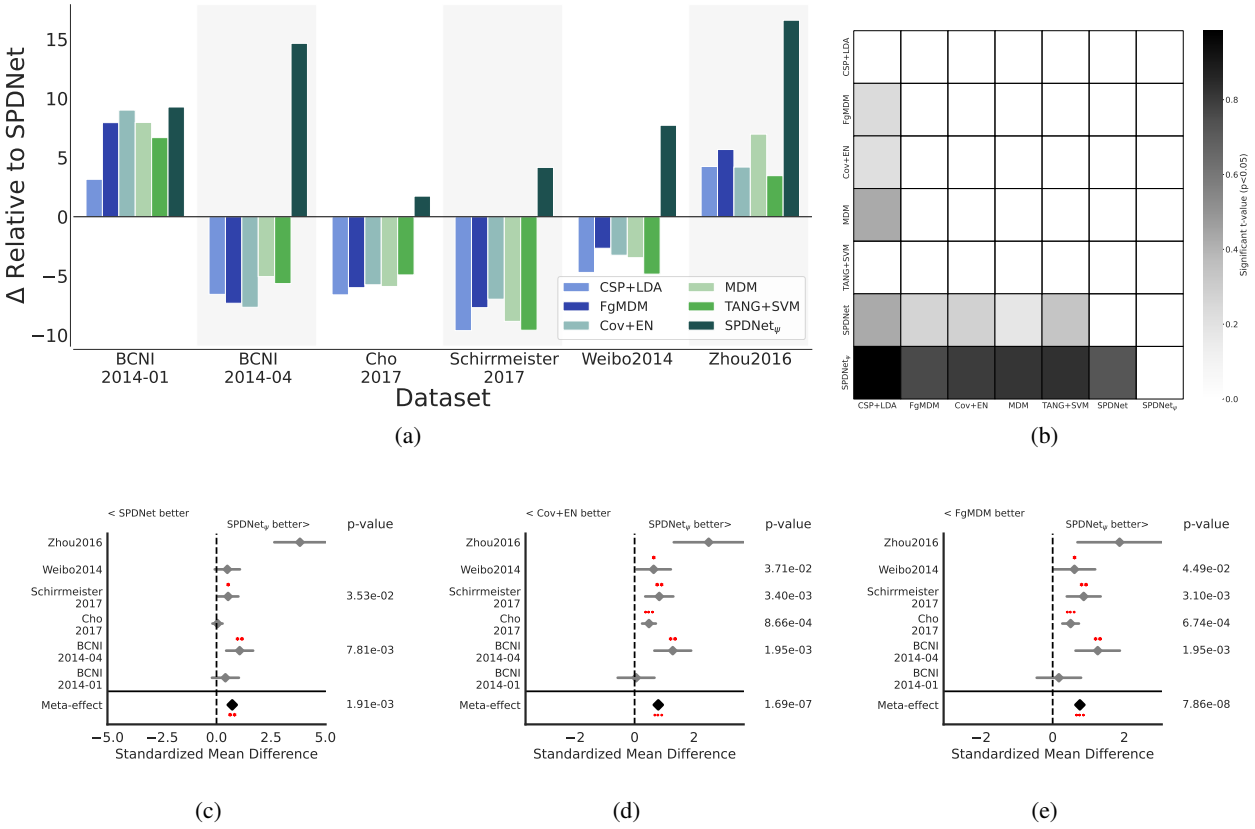


Figure A2: Result for right hand vs left hand classification, using within-session evaluation for state-of-the-art instantaneous coherence pipelines. Plot (a) provides the relative improvement of the method considered with respect to the standard SPDNet of the different pipelines considered. Plot (b) shows a combined meta-analysis (over all datasets) of the different pipelines. It shows the significance of the algorithm on the y-axis being better than the one on the x-axis. The gray level represents the significance level of the ROC-AUC difference in terms of t-values. We only show significant interactions ($p < 0.05$). Plots (c), (d), and (e) show the meta-analysis of SPDNet$_\psi$against SPDNet, COV+EN, and FgMDM, respectively. We show the standardized mean differences of p-values computed as a one-tailed Wilcoxon signed-rank test for the hypothesis given in the plot title. The **gray** bar denotes the 95% interval. * stands for $p < 0.05$, ** for $p < 0.01$, and *** for $p < 0.001$.