MAtt: A Manifold Attention Network for EEG Decoding

Anonymous Author(s)
Affiliation
Address
email

Abstract

Recognition of electroencephalographic (EEG) signals highly affect the efficiency of non-invasive brain-computer interfaces (BCIs). While recent advances of deep-learning (DL)-based EEG decoders offer improved performances, the development of geometric learning (GL) has attracted much attention for offering exceptional robustness in decoding noisy EEG data. However, there is a lack of studies on the merged use of deep neural networks (DNNs) and geometric learning for EEG decoding. We herein propose a manifold attention network (mAtt), a novel geometric deep learning (GDL)-based model, featuring a manifold attention mechanism that characterizes spatiotemporal representations of EEG data fully on a Riemannian symmetric positive definite (SPD). The evaluation of the proposed mAtt on both time-synchronous and -asyncronous EEG datasets suggests its superiority over other leading DL methods for general EEG decoding. Furthermore, analysis of model interpretation reveals the capability of mAtt in capturing informative EEG features and handling the non-stationarity of brain dynamics.

1 Introduction and related works

A brain-computer interface (BCI) is a type of human-machine interaction that bridges a pathway from brain to external devices. Electroencephalogram (EEG), a non-invasive neuromonitoring modality with high portability and affordability, has been widely used to explore practical applications of BCI in the real world [40, 19, 37]. For instance, disabled users can type messages through an EEG-based BCI that recognizes the steady-state visual evoked potential (SSVEP) induced by flickering visual targets presented on a screen [1, 44, 12]. Stroke patients who need restoration of motor function undergo motor-imagery (MI) BCI-controlled rehabilitation as an active training [3, 2]. Most EEG-based BCI systems are designed to detect/recognize reproducible time-asynchronous or time-synchronous EEG patterns of interest, depending on the schemes of BCI [28]. For example, the MI EEG pattern is an endogenous oscillatory perturbation without an explicit onset time sources from the motor cortex [15]. On the other hand, a time-synchronous EEG pattern is time-locked to a specific event. For example, the pattern of SSVEP is synchronized to the change of brightness on a flickering visual target. The efficiency of BCI systems largely relies on the accuracy and robustness of the EEG decoder. However, due to the low signal-to-noise ratio (SNR) [20] and non-stationarity [17] of EEG, translating perplexing EEG signals into meaningful information has been a grand technology and scientific challenge in the field.

Recent advances in deep learning (DL) have contributed to the rapid development of DL-based EEG decoding techniques [35]. DL models are capable of extracting features automatically according to

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

given training data. Convolutional neural network (CNN) is one type of the most common DL models and has achieved remarkable performance in tasks such as image recognition and object detection [26, 16, 39]. CNN models newly designed for EEG decoding use convolutional kernels that serve as spatial and temporal filters but with extra flexibility to optimize the transformation of EEG data automatically through model training [25, 36, 45]. In addition to the fast growth of DL-based EEG decoders, geometric learning (GL) approaches, mostly based on Riemannian geometry (RG), have been adopted in the field of BCI [27]. RG is a type of non-Euclidean geometry that has a different interpretation of Euclid's fifth postulate (i.e. parallel postulate) [41]. In GL, geodesic between points on the manifold is a critical feature for classification tasks in BCI. The power and spatial distribution of a segment of multi-channel EEG signals can be coded into a covariance matrix that is symmetric positive definite (SPD) in general. This allows mapping of EEG data directly onto a Riemannian manifold where Riemannian metric is insensitive to extreme outliers and noise [14, 27]. In 2010, Barachant et al. [6] proposed Minimum Distance to Mean (MDM) that maps target EEG data onto the SPD manifold to find the nearest class center. Later on, they developed TSLDA [7] that projects data from the manifold to a specific tangent space where Euclidean classifiers are applicable. RG-based classification for EEG decoding has shown extra robustness as the relationship between data samples can be stably preserved, leading to success in recent data competitions in the BCI field such as 'DecMEG2014'[1] and the 'BCI challenge'[2].

The nascent field of geometric deep learning (GDL) has expanded by emerging techniques to generalize the use of deep neural networks to non-Euclidean spaces. Efforts have been made to transitioning useful operations from Euclidean to Riemannian spaces, including convolution [18, 11, 8], activation function [18, 11], batch normalization [9, 13], that facilitate the ongoing development of GDL tools. SPDNet [18] is a Riemannian network for non-linear SPD-based learning on Riemannian manifolds using bi-linear mapping that mimics Euclidean convolution for visual classification tasks. ManifoldNet [11] offers high performance in medical image classification with manifold autoencoder. [30] characterizes 3D movement via the manifold polar coordinate with a geodesic CNN. [32] performs convolution on the manifold as a generalization of local graph or manifold pseudo-coordinate for vertex classification on graph and shape correspondence task. In contrast of the vast develop of geometric deep learning (GDL) in many other scientific fields, only few studies focus on decoding EEG data with a merge use of GL and DL. [46] proposed a network architecture that integrates fusion of Euclidean-based module and manifold-based module with multiple LSTM and attention structures to extract spatiotemporal information of EEG. [38] proposes a Riemannian-embedding-banks method that separates the entire embeddings into multiple sub-problems for learning spatial patterns of MI EEG signals based on the features extracted from the SPDNet. [21] combines federated learning and transfer learning on Riemannian manifold using the spatial information of EEG. [22] proposes deep optimal transport on the manifold to minimize the cost of domain adaptation from the source domain to the target domain. [23] extracts multi-view representations of EEG. These studies have established cornerstones toward the field of future GDL for EEG decoding, but the increment of performance is yet marginal. Most of the above-mentioned techniques can not map the temporal information of EEG onto the manifold, or still rely on Euclidean tools to handle EEG features. We herein propose a manifold attention network, a novel GDL framework, which maps EEG features on a Riemannian SPD manifold where the spatiotemporal EEG patterns are fully characterized. The main contributions of the present study are the following:

- a manifold attention network proposed for decoding general types of EEG data.
- a lightweight, interpretable, and efficient GDL framework that is capable of capturing spatiotemporal EEG features across Euclidean and Riemannian spaces.
- an empirical validation of our proposed model demonstrating its generalizable superiority over leading DL approaches in EEG decoding.
- neuroscientific insights interpreted by the model that not only echo prior knowledge but also offer a new look into the dynamical brain.

[1] DecMEG2014: https://www.kaggle.com/competitions/decoding-the-human-brain/leaderboard
[2] BCI challenge: https://www.kaggle.com/c/inria-bci-challenge

This article is organized as follows: we first brief the essential background of RG and manifold attention mechanism; next, we describe the proposed mAtt architecture with details of model design and training; we then validate our proposed model experimentally; lastly, we interpret our proposed model with neuroscientific insights.

2 Preliminary

A manifold is considered as the expansion of curve and surface in Euclidean space. It is a topological space that can locally regarded as an open set in Hilbert space. Suppose a manifold is endowed with a differential structure (i.e. a collection of charts satisfying transition mapping, which is defined on the overlap of charts), it is then the so-called differential manifold [24]. Riemannian geometry is a differential manifold equipped with Riemannian metric. We consider the Symmetric positive definite (SPD) manifold, which allows us to manipulate manifold-valued data on the manifold directly. The spatial information of EEG signal can be represented as a specific covariance matrix, which records the relationship between channels, and is a critical index for us to understand EEG signal. However, the solution of the Riemannian mean doesn't have a close form, thus we need to calculate the approximate mean in an iteration manner [6, 11] until convergence conditions are satisfied. Riemannian mean is not suitable for being applied in deep learning because of the high computational complexity. Therefore, we have the Log-Euclidean metric below.

2.1 Notations

$GL(n, \mathbb{R}) := \{A \in \mathbb{R}^{n\times n} \mid determinant(A) \neq 0\}$ is a general linear group, which is the set of all real non-singular sqaure matrices. $(\mathcal{M}, g)$ denotes connected Riemannian manifold. $Sym(n) := \{S \in M_{n\times n}(\mathbb{R}) \mid S^T = S\}$ is the space of all $n \times n$ real symmetric matrices, where $M_{n\times n}(\mathbb{R})$ specifies the space of all real square matrices, $(.)^T$ is the *transpose* operator, and $Sym^+(n) := \{P \in M_{n\times n}(\mathbb{R}) \mid P = P^T, v^T P v > 0, \forall v \in \mathbb{R}^n - \{0\}\}$ is the set of all $n \times n$ symmetric positive definite(SPD) matrices.$< A, B >_F$ means the Frobenius inner product, defined as $Tr(A^T B)$, where $Tr(.)$ is the *trace* operator. $Log(.)$ and $Exp(.)$ are the *principle logrithm* operator for SPD matrix[31] and *exponential* operator for symmetric matrix respectively. Both of them can be computed using the *orthogonal diagonalization*. $Log : Sym^+(n) \mapsto Sym(n)$ is an operator that maps a SPD matrix $P \in Sym^+(n)$ to $Sym(n)$ by:

$$Log(P) = Udiag(log(\sigma_1), ..., log(\sigma_n))U^T \tag{1}$$

where U is the matrix of eigenvectors of P, since $P \in Sym^+(n)$, $\sigma_i > 0, i = 1, ..., n$

The inverse projection is Exp of symmetric matrix: $Exp : Sym(n) \mapsto Sym^+(n)$, an operator maps a symmetric matrix $S \in Sym(n)$ to $Sym(n)$ by:

$$Exp(S) = Vdiag(exp(\sigma_1), ..., exp(\sigma_n))V^T$$

where V is the matrix of eigenvectors of S.

2.2 Log-Euclidean metric

Log-Euclidean metric (LEM) offers a more simple, similar, and efficient generalization to calculate the center on the SPD manifold than AIM. LEM is a bi-invariant metric on the Lie group on the SPD manifold [4]. The geodesic distance from P_1 to P_2 on the $Sym^+(n)$ is also given by [4]:

$$\delta_L(P_1, P_2) = \|Log(P_1) - Log(P_2)\|_F \tag{2}$$

Furthermore, we can also define the Log-Euclidean mean($\mathcal{G}$) via the Log-Euclidean distance:

$$\mathcal{G}(P_1, ...P_k) = \underset{P\in Sym^+(n)}{\arg\min} \sum_{l=1}^{k} \delta_L^2(P, P_l) \text{ where } P_1, ..., P_k \in Sym^+(n).$$

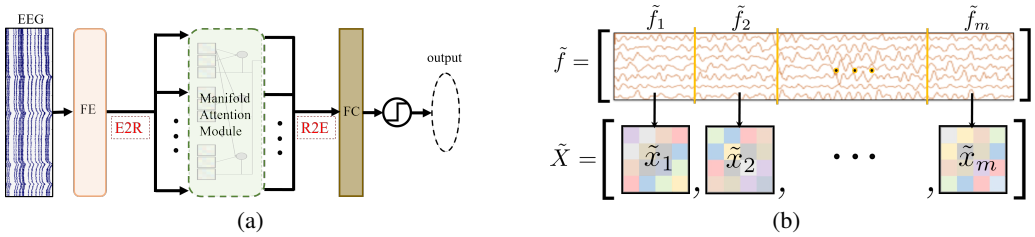


Figure 1: (a) The overview of the proposed model architecture. (b) E2R operation: split latent feature into several epochs, and convert each one to a specific SPD matrix.

Fortunately, the solution to the formula above has a closed form to follow, given by [5]:

$$\mathcal{G} = Exp\left(\frac{1}{k}\sum_{l=1}^{k} Log(P_l)\right)$$

In this work, we are going to use the weighted Log-Euclidean mean that is endowed with different weights in different P_l. We denote the weight of each P_l as w_l, where $\forall l \in \{1, 2, ..., k\}$. Here, $\{w_l\}_{l=1}^{k}$ satisfies the *convexity constraint* definition (i.e. $\sum_{l=1}^{k} w_l = 1$, and $w_l > 0$). The definition and the corresponding weighted Log-Euclidean mean can be defined and derived as:

$$\mathcal{G}(P_1, ...P_k) = \underset{P \in Sym^+(n)}{\arg\min} \sum_{l=1}^{k} w_l \delta_L^2(P, P_l)$$

and

$$\mathcal{G} = Exp\left(\sum_{l=1}^{k} w_l Log(P_l)\right)$$

respectively.

3 Methodology

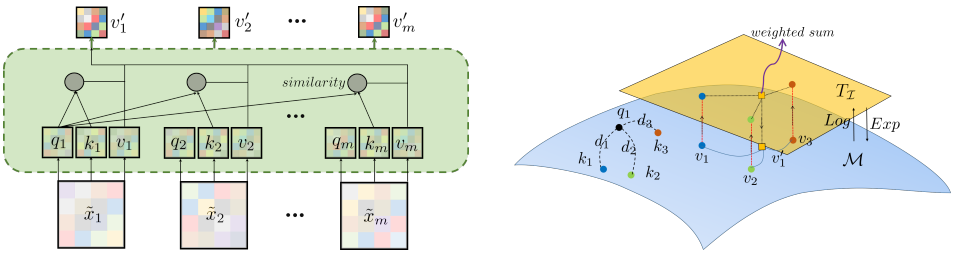
As shown in Figure 1(a), the architecture of mAtt includes components of the feature extraction (FE), the manifold attention module, transitioning from Euclidean to Riemannian space (E2R), and transitioning from Riemannian to Euclidean space (R2E).

3.1 Feature extraction of EEG signals

We use two convolutional layers to extract information of the raw EEG signals, where the first convolutional layer performs spatial filtering to the multi-channel EEG signals and the second convolutional layer extracts spatiotemporal features. Our parameter setting follows [45].

3.2 From Euclidean space to SPD manifold(E2R operation)

As illustrated in 1 (b), we convert the embeddings from the feature extraction stage to the SPD data and map the feature embeddings from Euclidean space to the SPD manifold. Suppose $\tilde{f}$ denotes the embeddings after the feature extraction stage, we divide the whole embeddings into several epochs $\tilde{f}_1, \tilde{f}_2, ..., \tilde{f}_m$, and calculate the sample covariance matrix(SCM) of each $\tilde{f}_i, \forall i \in \{1, 2, ..., m\}$. By doing so, we get a sequence of covariance matrices that present the temporal information of the embeddings $\tilde{f}$ in the form of SPD data, called $SCM_{\tilde{f}_1}, SCM_{\tilde{f}_2}, ..., SCM_{\tilde{f}_m}$. After we get some datapoints, we do trace-normalization and add a small number ϵ on each main diagonal element for



(a) (b)

Figure 2: (a) The architecture of manifold attention module. (b) Illustration of the operation of Log-Euclidean mean used in mAtt.

each $SCM_{\tilde{f}_i}$(i.e. $\frac{SCM_{\tilde{f}_i}}{tr\left(SCM_{\tilde{f}_i}\right)} + \epsilon I$) where $i \in \{1, 2, ..., m\}$, I is the identity matrix, and we set ϵ as 1e-5 in our source code. The resulting SPD sequence is denoted as $\tilde{X} = [\tilde{x_1}, \tilde{x_2}, ..., \tilde{x_m}]$. We add a small identity matrix on them to promise $\tilde{x_i}$ to be a well-defined SPD matrix.

3.3 Manifold attention module

The input of this layer is a sequence of SPD data. The overview of the manifold attention module is illustrated in Figure2 (a). Motivated by [18] and [43], we capture the spatiotemporal information on the manifold. Suppose the module takes a sequence of SPD matrices $[\tilde{x_1}, \tilde{x_2}, ..., \tilde{x_m}]$, denoted as $\tilde{X}$.

Here we have the query, key, and value in the form of SPD matrices on the manifold. We convert the $\tilde{x_i}$ to the q_i, k_i, and v_i via bilinear mapping and exploit non-linear and valid features from each segment. Suppose the shape of $\tilde{x_i}$ is $d_c \times d_c$, and h_q, h_k, and h_v is the mapping from $\tilde{x_i}$ to q_i, k_i, and v_i respectively:

$$
\begin{aligned}
q_i &= h_q(\tilde{x_i}; W_q) = W_q \tilde{x_i} W_q^T \\
k_i &= h_k(\tilde{x_i}; W_k) = W_k \tilde{x_i} W_k^T \\
v_i &= h_v(\tilde{x_i}; W_v) = W_v \tilde{x_i} W_v^T
\end{aligned}
$$

where $\tilde{x_i} \in Sym^+(d_c)$, W_q, W_k, and $W_v \in \mathbb{R}^{d_u \times d_c} (d_u < d_c)$ denotes transformation matrices. To make sure the output q_i, k_i, and v_i are also SPD matrices, transition matrices W_q, W_k, and W_v are row-full rank matrices.

After we got q_i, k_i, and v_i by bilinear mapping, we define the similarity for measuring the q_i and k_j SPD matrices. In Euclidean space, there are several ways to define the similarity. A most common way is to use dot-product[43] to measure the similarity of query and key. However, our query, key, and value are SPD matrices instead of vectors as regular attention. We define the similarity based on the Log-Euclidean distance (equation(2)) between query and key. Suppose we have q_i and k_j, for some $i, j \in \{1, ..., m\}$. The similarity $sim(.)$ is a $strictly\ decreasing\ function$ of distance $[0, \infty) \mapsto [0, 1]$ and is defined as: $sim(q_i, k_j) = \frac{1}{1+log(1+\delta_L(q_i, k_j))} := \alpha_{ij}$. Then, the attention matrix is:

$$
\mathcal{A} = [\alpha_{ij}]_{m \times m}
$$

We then use $Softmax$ function to shrink the range along the row direction, making values in row have $convexity\ constraint$ property. The final attention probability matrix $\mathcal{A}'$ is:

$$
\mathcal{A}' = Softmax(\mathcal{A}) = Softmax([\alpha_{ij}]_{m \times m}) = [\alpha'_{ij}]_{m \times m}
$$

where $\alpha'_{ij} = \frac{exp(\alpha_{ij})}{\sum_{k=1}^{m} exp(\alpha_{ik})}$, $\forall i, j \in 1, \cdots, m$. Finally, we combine the attention probability matrix and $v_1, v_2, ..., v_m$ to get the final output $v'_1, v'_2, ..., v'_m$ and define the output $v'_i (\forall i = 1, 2, ..., m)$ via Log-Euclidean mean as:

$$v'_i = Exp\left(\sum_{l=1}^{m} \alpha'_{il} Log(v_l)\right)$$

The whole mAtt procedure is illustarted in Algorithm1.

Algorithm 1 Manifold attention module

Require: A sequence of SPD data $\{\tilde{x_i}\}_{i=1}^{m}$ **, transformation matrices** W_q, W_k, W_v

for $i = 1 : m$ **do**
  $q_i = W_q \tilde{x_i} W_q^T$
  $k_i = W_k \tilde{x_i} W_k^T$
  $v_i = W_v \tilde{x_i} W_v^T$
end for
$\forall i, j \in \{1, \cdots, m\}, \mathcal{A} := [\alpha_{ij}]_{m \times m} = \frac{1}{1+log(1+\delta_L(q_i,k_j))}$
$\mathcal{A}' = Softmax(\mathcal{A})$
for $i = 1 : m$ **do**
  $v'_i = Exp\left(\sum_{l=1}^{m} \alpha'_{il} Log(v_l)\right)$
end for
return a sequence of SPD data $\{v'_i\}_{i=1}^{m}$

As shown in Figure2(b), suppose $i = 1, m = 3$, we compute the similarity $\alpha'_{11}, \alpha'_{12}, \alpha'_{13}$ via the distance from k_1 to q_1, k_2 to q_1, and k_3 to q_1 respectively. The points on the SPD manifold $\mathcal{M}$ are averaged after being projected into the tangent space based on an identity matrix $\mathcal{I}$ (denoted as $T_{\mathcal{I}}$), and then the weighted sum is mapped back to the SPD manifold.

3.4 From Riemannian manifold to Euclidean space(R2E) and loss layers

After passing through attention module, ReEig layer is used to imitate the ReLU function. But it is different from the ReLU function that sets the threshold to the value of input, ReEig sets a threshold to the eigenvalue of the input, can be defined in [18]. R2E operation aims to map the SPD data back to the Euclidean space for executing the final classification, which is composed of a Log layer and regular flatten layer in [18] sequentially. Log layer is the most common skill in geometric deep learning to project the SPD data to the Euclidean space. By doing so, we reduce the manifold to a flat space by $Log(.)$ operation. We take $Log(.)$ operation on the output from the attention module layer $v'_1, v'_2, ..., v'_m \in Sym^+(d_u)$. Denote the whole R2E operation as $h_L : Sym^+(d_u) \mapsto \mathbb{R}^{d_u \times (d_u+1)/2}$:

$$h_L(v'_i) = flatten\left(Log(v'_i)\right) = flatten\left(S(diag(log(\sigma_1), \cdots, log(\sigma_{d_u})))S^T\right)$$

where S is the eigenvector-matrix of v'_i, and $\sigma_1, \cdots, \sigma_{d_u}$ are the eigenvalues of v'_i. The $Log(.)$ operation is the same as equation(1), and the $flatten(A)$ operation flatten the upper triangle of the arbitrary symmetric matrix A .

Finally, we set a fully connected layer and regular softmax operation on embeddings after R2E operation. Suppose the output from the whole model stream is $\hat{y}$, the groundtruth is y, we define the loss $\mathcal{L}$ as the cross-entropy loss of $\hat{y}$ and y.

3.5 Evaluation

Time-asynchronous EEG data: We employ the BCIC-IV-2a as a representative dataset providing time-asynchronous EEG data. The BCIC-IV-2a dataset is one of the most commonly used public EEG dataset released for the BCI Competition IV in 2008 [10]. It contains EEG recordings in a four-class motor-imagery task from nine subjects with two repeated session each on different days. During the task, the subjects were instructed to perform imagination of one of the four types of movements (right hand, left hand, feet, and tongue) for four seconds after an instructional cue. Each

session consists of a total of 288 trials with 72 trials for each type of the motor imagery. The EEG signals were recorded by 22 Ag/AgCl EEG electrodes at the central and surrounding regions in a sampling rate of 250 Hz. We performed standard preprocessing procedures for the 22-channel EEG signals, including 1) Down-sampling from 256 Hz to 128 Hz, 2) Band-pass filtering at 4-38 Hz, and 3) Segmenting EEG signals at 0.5s-4s of the onset of cue for each trial.

Time-synchronous EEG data: The MAMEM-SSVEP-II dataset was incoporated as the representative dataset of time-synchronous EEG data. It consists of EEG recordings from 11 subjects performing a SSVEP-based task where the subjects were instructed to gaze at one of the five visual stimuli flickering at different frequencies (6.66, 7.50, 8.57, 10.00, and 12.00 Hz) for five seconds. Each subject executed five runs of five cue-based trials for each of the five stimulation frequencies. The EEG signals were acquired using the EGI 300 Geodesic EEG System (GES 300) equipped with 256 channels in a sampling rate of 250 Hz. The preprocessing procedures for this dataset were 1) Band-pass filtering at 1-50 Hz, 2) Selecting eight channels (PO7, PO3, PO, PO4, PO8, O1, Oz, and O2) in the occipital area, the location of visual cortex, and 3) Segmenting each trial into four 1-second segments at 1s-5s of the onset of cue, yielding a total of 500 trials of 1-second 8-channel SSVEP signals for each subject.

Model training and validation: A series of experiments were conducted to evaluate the performances of the mAtt and other EEG decoders with the scenario of real-world BCI usage into account. In the real-world usage of BCI, a user usually needs to go through a training session to collect a sufficient amount of individual EEG data for training the decoding model before executing the BCI system. To stick with the practical scenario, we performed an individual training scheme where a chunk of trials within a subject are assigned to training data and the left-over trials within the same subject are used for testing [45, 29]. For the BCIC-IV-2a dataset, we used the first session of a subject to the training set where one out of eight was used for validation for mAtt with $m = 3$. The model with the lowest validation loss within 350 iterations was used for testing on the second session of the same subject. Similarly, for the MAMEM-SSVEP-II dataset, we assigned the first four sessions of a subject to the training set where one out of four was used for validation for mAtt with $m = 7$. The model with the lowest validation loss within 180 iterations was used for testing on the fifth session of the same subject. As all models were initialized with random weights, the classification performance was estimated by the mean across ten repeats for each subject.

4 Experiments

Here we evaluate the proposed mAtt using both time-asynchronous and time-synchronous EEG data to giveempirical evidence of the advantages. The performance in a general use for EEG decoding is compared against leading GL- and DL-based techniques. We incorporate the BCI Competition IV 2a Dataset (BCIC-IV-2a) [10] to assess the performance on time-asynchronous motor-imagery EEG decoding and the MAMEM EEG SSVEP Dataset II (MAMEM-SSVEP-II) [33] to assess the performance on time-synchronous SSVEP EEG decoding. Previous and current state-of-the-art DL-based models listed for comparison with mAtt include FBCNet[29], SCCNet[45], EEGNet[25], and ShallowConvNet[36].

4.1 Performance comparison

We show the performance comparison on MI EEG decoding. As shown in Table 1, mAtt outperforms all other SOTA DL methods on both time-synchronous (SSVEP) and -asynchronous (MI) EEG decoding. We conclude that mAtt has a generalizable superiority in decoding EEG for all types of BCI systems. The classification result at the level of individual subject is provided in the appendix.

4.2 Model interpretation

Through analysis of model interpretation, we are able to uncover the underlying characteristics learnt from the data. Figure 3 illustrates the gradient response for MI EEG decoding across channel and

Table 1: Performance comparison between mAtt and baseline DL methods on MI (BCIC-IV-2a) and SSVEP (MAMEM-SSVEP-II) datasets. Bold fonts mark the highest overall performance. The asterisk indicates that mAtt significantly outperforms the given baseline method with $p < 0.05$ (Wilcoxon-sign rank test with Bonferroni correction).

Dataset	EEGNet [25]	ShallowConvNet [36]	FBCNet [29]	SCCNet [45]	mAtt
MI	57.43*±6.3	61.84*±6.4	71.45*±4.5	71.95±5.1	**74.71**±5.0
SSVEP	53.72*±7.2	56.93*±7.0	53.09±5.7	62.11±7.7	**65.50**±8.2

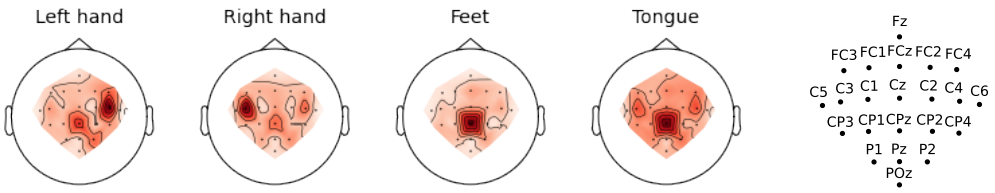


Figure 3: Spatial topomaps for the mean absolute gradient response across time from the visualization of the model S3 in the BCIC-IV-2a dataset for the four motor-imagery classes (left hand, right hand, feet, and tongue). Dark red marks the brains region presenting strong gradient activation at C4 (over right motor cortex) for the left hand, C3 (over left motor cortex) for the right hand, CPz (over motor cortex) for the feet and the tongue motor imagery.

across time. We can see left/right hand MI responses are strong at C4 and C3 corresponding to right/left motor cortices that control the lateral motor functions of the contralateral side of the body [34]. Both feet and tongue MI, that are not lateral movements, presents strong responses at CPz above the midline of motor cortex. The spatial distribution clearly exhibit asymmetric pattern for

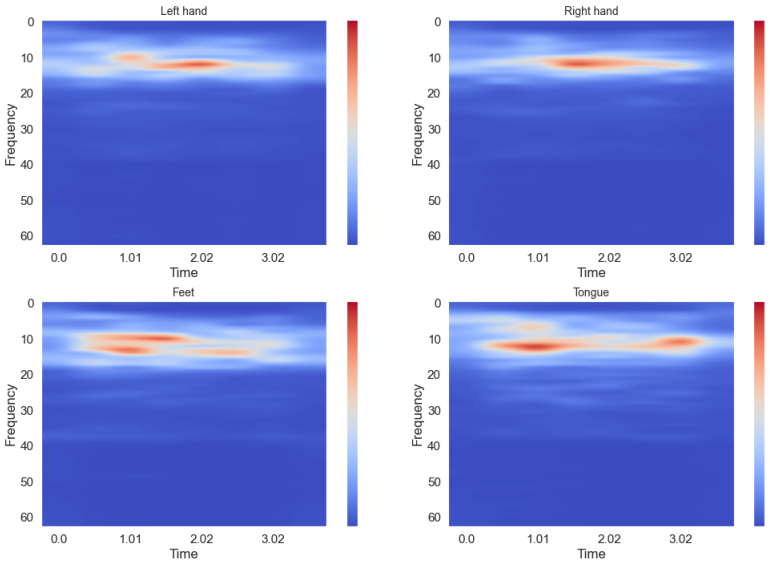


Figure 4: Time-frequency spectrograms from the visualization of the model S3 in the BCIC-IV-2a dataset for the four motor-imagery classes (left hand, right hand, feet, and tongue). Strong response of motor imagery is marked by dark red at specific frequency bands and time intervals. Increased response of motor imagery is found at mu band (10 Hz) for all classes. The strong response of left/right hand motor imagery occurs at 1-2 seconds, the feet motor imagery is most vivid at 0.5-1 seconds, and the tongue motor imagery induced two peaks at 1 second and 3 seconds.

left/right hand MI and symmetric pattern for feet/tongue MI. The temporal information is available in Figure 4, where all four types of MI induce strong response at mu band around 10 Hz. This result is in line with the well-known association between motor function and mu rhythm in EEG recordings [34]. The visualization of our model for SSVEP decoding is exhibit in the appendix.

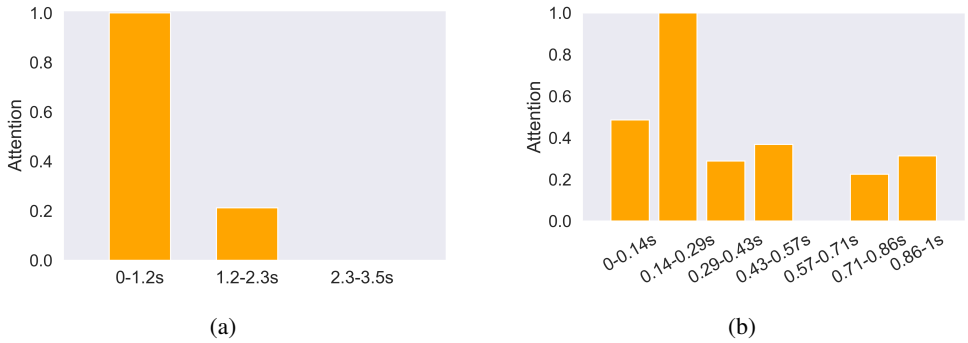


Figure 5: (a) The distribution of attention scores across three epochs within a trial gained from the model interpretation of the S3 in the BCIC-IV-2a dataset. (b) The distribution of attention scores across three epochs within a trial gained from the model interpretation of the S11 in the MAMEM-SSVEP-II dataset.

Figure 5 depicts the distribution of attention scores across epochs within the (a) MI and (b) SSVEP EEG signals. For MI EEG signals, attention score is the highest at the first epoch and decreases in the following epochs, which implies that the beginning of the motor imagery may contribute a higher importance determined by the manifold attention module. The profile of attention score for SSVEP EEG signals presents a similar traits that the earlier epochs relate to higher importance. This analysis reveals the capability of mAtt in handling the non-stationarity of the dynamical brain activity.

4.3 Limitation

In our framework, vacuum permittivity ϵ is added on all main diagonal elements of covariance $cx_ix_i^T$ to ensure the rigor of SPD matrix. But the operation may cause the repeated singular value ϵ in S_i. Therefore, we proposed possible solutions for this issue:1) Let $m < n$ when dividing the embeddings into several time segments, reducing the possibility of getting low-rank S_i; 2) Let ϵ be randomly drawn from a specific distribution, such as $Uniform\ (1e-8, 1e-4)$ to solve this issue, which is also a practicable solution; 3) Use the derivative of a low-rank matrix[42] to cope with this issue.

Conclusion

We propose a manifold attention network as a novel GDL framework for decoding both time-synchronous and -asynchronous EEG decoding. Using back propagation based on the Stiefel manifold, the proposed mAtt is capable of mapping EEG features onto a Riemannian manifold, where spatiotemporal EEG patterns are captured and characterized, within a lightweight architecture. The experimental results suggest the superiority of mAtt over current SOTA DL methods for both time-synchronous and -asynchronous EEG decoding. With the interpretability of mAtt, we visualize the spatial and temporal EEG patterns, which are in line with prior neuroscientific knowledge and shed light on potential possibility of tracking the brain dynamics. In sum, our privileged method, mAtt, improves the SOTA performance of EEG decoding, and is expected to impact on GDL-based EEG processing with generalizable efficiency and robustness for future development of various BCI systems.

References

[1] Z. Al-Qaysi, B. Zaidan, A. Zaidan, and M. Suzani. A review of disability eeg based wheelchair control system: Coherent taxonomy, open challenges and recommendations. *Computer methods and programs in biomedicine*, 164:221–237, 2018.

[2] L. M. Alonso-Valerdi, R. A. Salido-Ruiz, and R. A. Ramirez-Mendoza. Motor imagery based brain–computer interfaces: An emerging technology to rehabilitate motor deficits. *Neuropsychologia*, 79:354–363, 2015.

[3] K. K. Ang, K. S. G. Chua, K. S. Phua, C. Wang, Z. Y. Chin, C. W. K. Kuah, W. Low, and C. Guan. A randomized controlled trial of eeg-based motor imagery brain-computer interface robotic rehabilitation for stroke. *Clinical EEG and neuroscience*, 46(4):310–320, 2015.

[4] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache. Log-euclidean metrics for fast and simple calculus on diffusion tensors. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 56(2):411–421, 2006.

[5] V. Arsigny, P. Fillard, X. Pennec, and N. 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.

[6] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten. Riemannian geometry applied to bci classification. In *International conference on latent variable analysis and signal separation*, pages 629–636. Springer, 2010.

[7] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten. Multiclass brain–computer interface classification by riemannian geometry. *IEEE Transactions on Biomedical Engineering*, 59(4):920–928, 2011.

[8] J. J. Bouza, C.-H. Yang, D. Vaillancourt, and B. C. Vemuri. Mvc-net: A convolutional neural network architecture for manifold-valued images with applications. *arXiv preprint arXiv:2003.01234*, 2020.

[9] D. Brooks, O. Schwander, F. Barbaresco, J.-Y. Schneider, and M. Cord. Riemannian batch normalization for spd neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

[10] C. Brunner, R. Leeb, G. Müller-Putz, A. Schlögl, and G. Pfurtscheller. Bci competition 2008–graz data set a. *Institute for Knowledge Discovery (Laboratory of Brain-Computer Interfaces), Graz University of Technology*, 16:1–6, 2008.

[11] R. Chakraborty, J. Bouza, J. Manton, and B. C. Vemuri. Manifoldnet: A deep neural network for manifold-valued data with applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[12] X. Chen, Y. Wang, M. Nakanishi, X. Gao, T.-P. Jung, and S. Gao. High-speed spelling with a noninvasive brain–computer interface. *Proceedings of the national academy of sciences*, 112(44):E6058–E6067, 2015.

[13] M. Cho and J. Lee. Riemannian approach to batch normalization. *Advances in Neural Information Processing Systems*, 30, 2017.

[14] M. Congedo, A. Barachant, and R. Bhatia. Riemannian geometry for eeg-based brain-computer interfaces; a primer and a review. *Brain-Computer Interfaces*, 4(3):155–174, 2017.

[15] M. Feurra, P. Pasqualetti, G. Bianco, E. Santarnecchi, A. Rossi, and S. Rossi. State-dependent effects of transcranial oscillatory currents on the motor system: what you think matters. *Journal of Neuroscience*, 33(44):17483–17489, 2013.

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

[17] G. E. Hine, E. Maiorana, and P. Campisi. Resting-state eeg: A study on its non-stationarity for biometric applications. In *2017 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pages 1–5. IEEE, 2017.

[18] Z. Huang and L. Van Gool. A riemannian network for spd matrix learning. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[19] I. Iturrate, J. Antelis, and J. Minguez. Synchronous eeg brain-actuated wheelchair with automated navigation. In *2009 IEEE International Conference on Robotics and Automation*, pages 2318–2325. IEEE, 2009.

[20] D. H. Johnson. Signal-to-noise ratio. *Scholarpedia*, 1(12):2088, 2006.

[21] C. Ju, D. Gao, R. Mane, B. Tan, Y. Liu, and C. 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. IEEE, 2020.

[22] C. Ju and C. Guan. Deep optimal transport on spd manifolds for domain adaptation. *arXiv preprint arXiv:2201.05745*, 2022.

[23] C. Ju and C. Guan. Tensor-cspnet: A novel geometric deep learning framework for motor imagery classification. *arXiv preprint arXiv:2202.02472*, 2022.

[24] W. P. Klingenberg. *Riemannian geometry*, volume 1. Walter de Gruyter, 2011.

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

[26] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

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

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

[29] R. Mane, E. Chew, K. Chua, K. K. Ang, N. Robinson, A. P. Vinod, S.-W. Lee, and C. Guan. Fbcnet: A multi-view convolutional neural network for brain-computer interface. *arXiv preprint arXiv:2104.01233*, 2021.

[30] J. Masci, D. Boscaini, M. Bronstein, and P. Vandergheynst. Geodesic convolutional neural networks on riemannian manifolds. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 37–45, 2015.

[31] M. 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.

[32] F. Monti, D. Boscaini, J. Masci, E. Rodola, J. Svoboda, and M. M. Bronstein. Geometric deep learning on graphs and manifolds using mixture model cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5115–5124, 2017.

[33] S. Nikolopoulos. MAMEM EEG SSVEP Dataset II (256 channels, 11 subjects, 5 frequencies presented simultaneously). 5 2021.

[34] G. Pfurtscheller, C. Brunner, A. Schlögl, and F. L. Da Silva. Mu rhythm (de) synchronization and eeg single-trial classification of different motor imagery tasks. *NeuroImage*, 31(1):153–159, 2006.

[35] Y. Roy, H. Banville, I. Albuquerque, A. Gramfort, T. H. Falk, and J. Faubert. Deep learning-based electroencephalography analysis: a systematic review. *Journal of neural engineering*, 16(5):051001, 2019.

[36] R. T. Schirrmeister, J. T. Springenberg, L. D. J. Fiederer, M. Glasstetter, K. Eggensperger, M. Tangermann, F. Hutter, W. Burgard, and T. Ball. Deep learning with convolutional neural networks for eeg decoding and visualization. *Human brain mapping*, 38(11):5391–5420, 2017.

[37] D. P. Subha, P. K. Joseph, R. Acharya U, C. M. Lim, et al. Eeg signal analysis: a survey. *Journal of medical systems*, 34(2):195–212, 2010.

[38] Y.-J. Suh and B. H. Kim. Riemannian embedding banks for common spatial patterns with eeg-based spd neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 854–862, 2021.

[39] M. Tan and Q. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[40] M. Teplan et al. Fundamentals of eeg measurement. *Measurement science review*, 2(2):1–11, 2002.

[41] R. Torretti. *Philosophy of geometry from Riemann to Poincaré*, volume 7. Springer Science & Business Media, 2012.

[42] J. Townsend. Differentiating the singular value decomposition. Technical report, Technical Report 2016, https://j-towns. github. io/papers/svd-derivative . . . , 2016.

[43] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[44] Y.-T. Wang, M. Nakanishi, Y. Wang, C.-S. Wei, C.-K. Cheng, and T.-P. Jung. An online brain-computer interface based on ssveps measured from non-hair-bearing areas. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(1):14–21, 2016.

[45] C.-S. Wei, T. Koike-Akino, and Y. Wang. Spatial component-wise convolutional network (sccnet) for motor-imagery eeg classification. In *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 328–331. IEEE, 2019.

[46] G. Zhang and A. Etemad. Rfnet: Riemannian fusion network for eeg-based brain-computer interfaces. *arXiv preprint arXiv:2008.08633*, 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] **We elaborated the contribution and scope of present study in Section1.**
   (b) Did you describe the limitations of your work? [Yes] **We discussed not only limitation, but also the possible solutions for our limitation in Section4.3.**

(c) Did you discuss any potential negative societal impacts of your work? [N/A]

(d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...

(a) Did you state the full set of assumptions of all theoretical results? [Yes] **See Section3**

(b) Did you include complete proofs of all theoretical results? [Yes] **See Section3**

3. If you ran experiments...

(a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] **We provide the instructions, code, and dataset in the supplemental material for everyone can reproduce our result easily.**

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] **We roughly describe the training scheme in Experiments??, and the detailed parameters already set in our source code(including the learning rate, batch size...).**

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] **We randomly repeated every experiment ten time to get an unbiased result, thus we also provide the std for each cell except for Riemannian ML methods that have no stochasticity.**

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] **We elaborate it in the appendix in supplemental materials.**

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [Yes] **We already elaborated evaluation datasets cited in Experiments.**

(b) Did you mention the license of the assets? **[TODO]**MAMEM CC BY 4.0

(c) Did you include any new assets either in the supplemental material or as a URL? [Yes]

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]