Structure-Preserving Transformers for Sequences of SPD Matrices

Mathieu Seraphim
Normandie Univ, UNICAEN,
ENSICAEN, CNRS, GREYC
14000 Caen, France
mathieu.seraphim@unicaen.fr

Alexis Lechervy
Normandie Univ, UNICAEN,
ENSICAEN, CNRS, GREYC
14000 Caen, France
alexis.lechervy@unicaen.fr

Florian Yger
LAMSADE, CNRS,
PSL Univ. Paris-Dauphine
75016 Paris, France
florian.yger@lamsade.dauphine.fr

Luc Brun
Normandie Univ, ENSICAEN,
UNICAEN, CNRS, GREYC
14000 Caen, France
luc.brun@ensicaen.fr

Olivier Etard
Normandie Université, UNICAEN, INSERM,
COMETE, CYCERON, CHU Caen
14000 Caen, France
olivier.etard@unicaen.fr

***Abstract*—In recent years, Transformer-based auto-attention mechanisms have been successfully applied to the analysis of a variety of context-reliant data types, from texts to images and beyond, including data from non-Euclidean geometries. In this paper, we present such a mechanism, designed to classify sequences of Symmetric Positive Definite matrices while preserving their Riemannian geometry throughout the analysis. We apply our method to automatic sleep staging on timeseries of EEG-derived covariance matrices from a standard dataset, obtaining high levels of stage-wise performance.**

***Index Terms*—Transformers, SPD Matrices, Structure-Preserving, Electroencephalography, Sleep Staging**

I. Introduction

When analyzing the relationship between concurrent signals, covariance matrices are a useful tool, with applications in fields like Brain-Computer Interfaces (BCI) [1] and evolutionary computation [2]. By construction, they are rich in information, illustrating the relationship between signals while still encoding for signal-wise information on their diagonal. Such matrices are at least Positive Semi-Definite, and often fully Symmetric Positive Definite (SPD). The set of $n \times n$ SPD matrices ($SPD(n)$) is a non-Euclidean, Riemannian (i.e. metric) manifold, and the regular Euclidean operations of most Neural Network (NN)-based models seldom preserve that geometric structure, introducing deformations such as the "swelling effect" [3]. Structure-preserving NN-based approaches have been introduced [4], [5], deriving their layers from one of two geodesic-defining metrics on $SPD(n)$. Affine invariant metrics offer the best properties, but present computational challenges (e.g. no closed-form formula for averaging) [6]. LogEuclidean metrics are less isotropic, but still prevent swelling while being easier to compute [3].

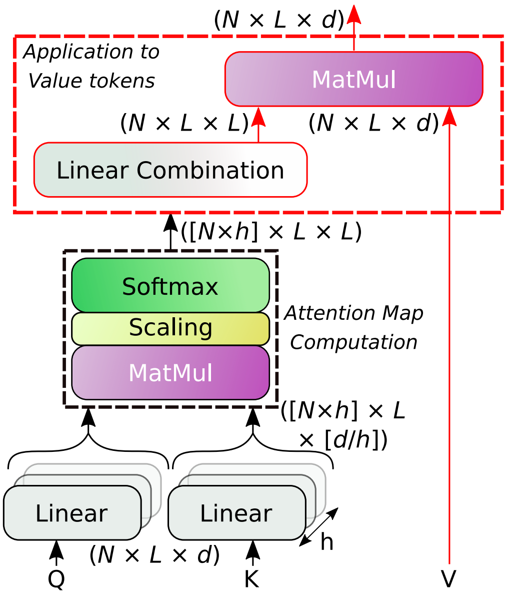

Fig. 1. The SP-MHA architecture. In parentheses are tensor dimensions at every step, with N the batch size.

In this paper, we present a structure-preserving self-attention mechanism applicable to sequences of SPD matrices, derived from such a LogEuclidean metric. We embed said mechanism into a Transformer-based architecture, and apply it to a biomedical classification problem. Transformer-based technology has exploded in popularity ever since its introduction in [7], with self-attention mechanisms being applied to very different problems. With regards to Riemannian geometry, innovations seem centered around the computation and application of attention maps, specifically. For instance, Konstantinidis et al. [8] combine the standard attention maps with

MS is supported by the French National Research Agency (ANR) and Region Normandie under grant HAISCoDe. This work was granted access to the HPC resources of IDRIS under the allocation 2022-AD010613618 made by GENCI, and to the computing resources of CRIANN (Normandy, France). This work was partially carried out during a CNRS leave at GREYC of FY.

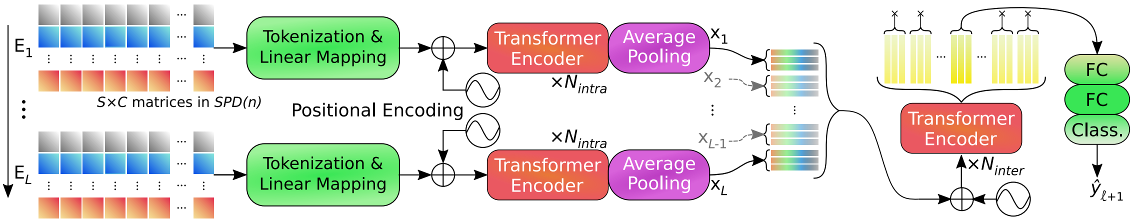

Fig. 2. SPDTransNet global architecture, with $t = 3$ feature tokens per epoch.

Grassmann and SPD manifold-valued maps, to enrich their computer vision model's descriptive capabilities. By contrast, both He et al. [9] and Li et al. [10] developed architectures to analyze 2D-manifold-valued data in 3D space, the former achieving rotational equivariance with respect to surfaces on the manifold and the latter developing two geodesic distances applicable to point clouds, and building attention maps from these distances. More generally, Kratsios et al. [11] provide a mathematical framework to apply attention mechanisms on a variety of constrained sets, including manifolds. While the latter approaches share our interest in preserving geometric information, little to no focus is given to a Transformer's other components. Although simple single-head attention modules for SPD-valued data have been recently developed [12], [13], to the best of our knowledge, our approach is the only one utilizing full structure-preserving Transformer encoders in this context.

II. SPD Structure-Preserving Attention

The LogEuclidean distance (Section I) can be written as:

$$\delta_{LE}(A, B) = \|log_{mat}(A) - log_{mat}(B)\|_2 \qquad (1)$$

with $A, B \in SPD(n)$. Here, $\|X\|_2$ (with $X \in Sym(n)$) is the $\mathcal{L}_2$ norm applied to the upper triangular of X, and $log_{mat}(\cdot)$ is the matrix logarithm, bijectively mapping $SPD(n)$ onto $Sym(n)$, the vector space of $n \times n$ symmetric matrices (with $exp_{mat}(\cdot)$ being its inverse). Euclidean operations on $Sym(n)$ are thus equivalent to LogEuclidean (and therefore Riemannian) operations on the corresponding SPD matrices.

Let $B_n = \{e_{i,j}\}_{0<i\leq j} \subset \mathbb{R}^{n\times n}$ be the the canonical basis of $Sym(n)$, with $(e_{i,j})_{i,j} = (e_{i,j})_{j,i} = 1$, and all other coefficients at 0. Let the triangular number $d(n) = \frac{n(n+1)}{2}$ be the dimension of $Sym(n)$. Any matrix M of $Sym(n)$ can be written in the basis B_n as a vector of coordinates in $\mathbb{R}^{d(n)}$.

In accordance with convention surrounding Transformer-based architectures, we refer to these vectors as "tokens". In this paper, any token of $\mathbb{R}^{d(n)}$ is thus equivalent to a matrix in $SPD(n)$, and linear combinations of such tokens would equate to a LogEuclidean weighted sum in $SPD(n)$, preserving their underlying manifold structure.

A. Structure-Preserving Multihead Attention (SP-MHA)

In the original Linear Multihead Attention (L-MHA) component of Transformers [7], the input tokens in the Q, K and V tensors are processed in parallel in h attention heads, then recombined through concatenation. There is no guarantee that any underlying SPD structure in our tokens would survive this concatenation. Echoing the similar concerns, Li et al. [10] decided to forego having multiple heads. Likewise, Pan et al. [12] and Qin et al. [13] restricted themselves to single-head SPD-valued attention modules. By contrast, we design our Multihead Attention block to retain the parallel attention maps computation of the original L-MHA without sacrificing our data's structure.

Let $d(m)$ be the dimension of input tokens. As seen in Figure 1, our SP-MHA block does the following:

$$MHA_{SP}(Q, K, V) = C\left(sm\left(\frac{\mathcal{L}_Q(Q)\cdot\mathcal{L}_K(K)^T}{\sqrt{d(m)/h}}\right)\right)\cdot V \qquad (2)$$

with $\mathcal{L}_Q(\cdot)$ and $\mathcal{L}_K(\cdot)$ banks of h linear maps from $\mathbb{R}^{d(m)}$ to $\mathbb{R}^{\frac{d(m)}{h}}$, $sm(\cdot)$ the softmax function, and $C(\cdot)$ the weighted linear combination of the h post-softmax attention maps. Here, the computation of attention maps (small-dashed black rectangle in Figure 1) remains identical to L-MHA. However, their application to V (large-dashed red rectangle in the figure) only requires a matrix multiplication, i.e. linear combinations of V's tokens weighted by the combined attention map. As such, the SP-MHA block does not compromise our tokens' vector space geometry.

B. Triangular linear maps

Let $Sym(n)$ and $Sym(m)$ have the canonical bases B_n and B_m, respectively. Let $\mathcal{L}_{n,m}(\cdot)$ be a linear map from $Sym(n)$ to $Sym(m)$, represented by the matrix W in $\mathbb{R}^{d(m)\times d(n)}$ with respect to the bases (implemented in code through a fully connected NN layer between tokenized matrices). We shall refer to such a map as a "triangular" linear map.

Let A^*, B^* be in $SPD(n)$, mapped to $A, B \in Sym(n)$ through $log_{mat}(\cdot)$. As $\mathcal{L}_{n,m}(\cdot)$ is a continuous linear map:

$$\|\mathcal{L}_{n,m}(A) - \mathcal{L}_{n,m}(B)\|_2 \leq \|W\|_* \cdot \|A - B\|_2 \qquad (3)$$

$$\delta_{LE}(\mathcal{L}^{\mathcal{R}}_{n,m}(A^*), \mathcal{L}^{\mathcal{R}}_{n,m}(B^*)) \leq \|W\|_* \cdot \delta_{LE}(A^*, B^*) \qquad (4)$$

with $\|\cdot\|_*$ the matrix norm induced by the norm $\|\cdot\|_2$, and $\mathcal{L}^{\mathcal{R}}_{n,m}(\cdot) = exp_{mat} \circ \mathcal{L}_{n,m} \circ log_{mat}(\cdot)$ mapping $SPD(n)$ onto $SPD(m)$. By definition of δ_{LE} (Equation 1), Equations 3 and 4 are strictly identical. Hence, applying $\mathcal{L}_{n,m}(\cdot)$ on our tokens is equivalent to applying $\mathcal{L}^{\mathcal{R}}_{n,m}(\cdot)$ on matrices in $SPD(n)$. The output tokens exhibit the Riemannian structure of $SPD(m)$, and relations of proximity are preserved. Therefore, so is the overall structure of our data.

Note that while other SPD-to-SPD NN-based mappings have been proposed [4], [14], they rely on full-rank weight tensors, whereas $\mathcal{L}^{\mathcal{R}}_{n,m}(\cdot)$ does not require special constraints.

III. Application to EEG Sleep Staging

The study of sleep most often requires the analysis of electrophysiological - including electroencephalographic (EEG) - signals, subdivided into fixed-length windows ("epochs") and manually labeled with the appropriate sleep stages, inferred from properties of the signal in and around each epoch [15].

As seen in a recent survey by Phan et al. [16], state-of-the-art automatic sleep staging models typically use two-step architectures - given a sequence of epochs, epoch-wise features are extracted before being compared at the sequence-wise level, utilizing this contextual information to improve classification. Since epochs often contain markers indicative of multiple stages, two-step architectures tend to subdivide them further, extracting features from subwindows using convolutional NNs [17], [18] and/or recurrent NNs [19]–[21] - the latter utilizing RNNs for both steps. Multiple authors have adapted this context-inclusive approach to Transformer-based architectures [22]–[24], with auto-attention mechanisms at both the intra- and inter-epoch levels, taking advantage of the high performance they offer when applied to sequence-based data.

A. The stakes of automatic sleep staging

According to the aforementioned survey [16], current sleep staging models have attained a sufficient performance level to replace manual staging in some contexts. However, we have found that class-wise performance was often lacking, particularly with regards to the N1 sleep stage [15], universally difficult to classify. Most EEG datasets are heavily imbalanced, with the N1 stage often underrepresented (Section IV) - models optimized for high overall accuracy may thus sacrifice N1 classification if it improves global performance. To account for this, recent approaches [24], [26] elected to primarily evaluate their performance through the macro-averaged F1 (MF1) score, a class-wise balanced metric widely used in the literature. They also rebalance their training sets through oversampling, so that all stages within have the same number of classification targets. While the survey states that a sequence-to-sequence classification scheme (classifying each epoch in the input sequence) might lead to better performance, having multilabel inputs is nonsensical for this rebalancing - hence their use of a sequence-to-epoch scheme (classifying one epoch per sequence).

Beyond sleep staging, EEG signals are also utilized in BCI (Section I), where they are often analyzed through the lens of functional connectivity - the activation correlations between different brain regions [27]. Automatic sleep staging through functional connectivity was first investigated by Jia et al. [25], using epoch-wise graph learning to estimate said connectivity and sequence-wise spatio-temporal graph NNs to compare them. By contrast, Seraphim et al. [24] estimate it through covariance matrices, as is commonly done in BCI [1]. Their two-step model uses standard Transformer encoders at each step, reminiscent of [23]. Each input epoch is described as a multichannel timeseries of SPD matrices, which are then tokenized bijectively. However, their approach does not guarantee the preservation of their data's SPD structure, as they operate a channel-wise concatenation of their tokens, in addition to the concatenations found within their encoders (Section II-A). Hence, we propose a Transformer-based model capable of analyzing EEG-derived functional connectivity through SPD matrices *without* sacrificing the SPD structure of our data throughout the analysis.

B. Our preprocessing

To estimate functional connectivity from EEG signals, we apply the same preprocessing pipeline as [24][1]. We first select n EEG signals. Each signal is then filtered along C frequency bands, divided into epochs, and further subdivided into S subwindows per epoch. A covariance matrix is computed per channel and subwindow, resulting in $S \times C$ covariance matrices in $SPD(n)$ for each epoch. We then augment our matrices with signal-derived information before whitening them[1], leading to more uniformly distributed matrices in $SPD(n+1)$. Said whitening requires the computation of average covariance matrices per recording and channel, which was done in [24] by computing the covariances over the entire recording. Instead, we average all relevant matrices using the standard affine invariant metric [6], improving performance.

C. The SPDTransNet model

As can be seen in Figure 2, our SPDTransNet model takes as input a sequence of L epochs, composed of a central epoch to classify and surrounding epochs to provide context. Given ℓ the context size, we have $L = 2 \cdot \ell + 1$.

Our preprocessing yields $S \times C$ matrices of $SPD(n+1)$ per epoch (Section III-B). Each of these matrices is mapped onto $Sym(n+1)$ through $log_{mat}(\cdot)$ and tokenized (Section II). Each input token of $\mathbb{R}^{d(n+1)}$ thus encodes the covariance of each signal pair, along with signal-specific information (the variance and augmentation features).

These tokens are linearly mapped onto $\mathbb{R}^{d(m)}$ (with $m > n+1$, as we have found that larger tokens improve performance). The $S \times C$ grid of tokens is then arranged into a sequence, with the S tokens in the channel 1 followed by the S tokens in channel 2, etc.

At the intra-epoch level, a first positional encoding is applied to the tokens, which pass through the first Transformer encoder. The $S \times C$ output tokens are then uniformly divided into t groups, with each group averaged into a single feature token. The L sets of t epoch-wise feature tokens are then regrouped at the inter-epoch level, and passed through another positional encoding and Transformer encoder pair. Finally, the feature tokens corresponding to the central epoch (of index

[1] More details at github.com/MathieuSeraphim/SPDTransNet.

	Model	MF1	Macro Acc.	N1 F1	Valid. metric	Token dim. $d(m)$	# Feat. Tokens t
1	SPDTransNet, $L = 13$	81.06 ± 3.49	**84.87** ± 2.47	60.39 ± 6.77	MF1	351 ($m = 26$)	7
2	SPDTransNet, $L = 21$	**81.24** ± 3.29	84.40 ± 2.61	**60.50** ± 6.18	MF1	351 ($m = 26$)	10
3	SPDTransNet, $L = 29$	80.83 ± 3.40	84.29 ± 2.65	60.35 ± 6.01	N1 F1	351 ($m = 26$)	5
4	Classic MHA	80.82 ± 3.40	84.60 ± 2.95	60.16 ± 7.20	MF1	351 ($m = 26$)	10
5	DeepSleepNet [17]	78.14 ± 4.12	80.05 ± 3.47	53.52 ± 8.24	N/A	N/A	N/A
6	IITNet [18]	78.48 ± 3.15	81.88 ± 2.89	56.01 ± 6.54	N/A	N/A	N/A
7	GraphSleepNet [25]	75.58 ± 3.75	79.75 ± 3.41	50.80 ± 8.06	N/A	N/A	N/A
8	Dequidt et al. [26]	81.04 ± 3.26	82.59 ± 3.45	58.42 ± 6.09	N/A	N/A	N/A
9	Seraphim et al. [24]	79.78 ± 4.56	81.76 ± 4.61	58.43 ± 6.41	MF1	Concatenation	1

TABLE I

Results obtained from both our model and the re-trained literature. Best results are in **bold**.

$\ell + 1$ in Figure 2) go through two FC blocks (fully connected layers followed by ReLU activation and a dropout layer), and are mapped onto $\hat{y}_{\ell+1} \in \mathbb{R}^c$ by a final classification linear map, with c the number of classes.

We ensure structure preservation by using the SP-MHA block in all Transformer encoders, and choosing all linear maps within said encoders' Feed-Forward (FF) components [7] to be triangular (Section II-B). The ReLU and dropout layers in the FF blocks do not cause issue, as setting a values within a token to 0 won't remove the corresponding matrix from $Sym(m)$. Same for the positional encodings, average poolings and in-encoder layer normalizations, which all qualify as linear combinations.

As such, our model preserves the SPD structure of its input up to the final classification layers, and every token throughout the model remains equivalent to an SPD matrix obtained through Riemannian operations (Section II).

IV. Experiments & Results

We utilize the MASS SS3 dataset [28] due to its large number of available EEG electrode-derived signals and widespread use in the literature. It is composed of 62 full-night recordings of healthy subjects, segmented into 30s epochs. Due to its nature, it is unbalanced, with the largest and smallest of its $c = 5$ classes (stages N2 and N1) composed of 50.24% and 8.16% of the dataset, respectively. Out of 20 available electrodes, we selected the n = 8 electrodes F3, F4, C3, C4, T3, T4, O1 and O2, providing us with a good coverage of the brain while limiting redundancies. As in [24], we filter our signals to obtain C = 7 channels, and subdivide each epoch into S = 30 one-second windows[1], yielding us 30×7 matrices in $SPD(9)$ after preprocessing (Section III-B).

To maximize class-wise performance, we operate a hyperparameter research per configuration, followed by a 31-fold cross-validation. As do [24], [26] (Section III-A), we rebalance all training sets and maximize the MF1 score. To explore the importance of the context length ℓ (Section III-C) within our model, we ran hyperparameter researches with ℓ = 6, 10 or 14 (i.e. L = 13, 21 or 29), with hyperparameter research configuration unchanged between them.

Our hyperparameter researches use the Optuna tool [29], with 5 simultaneous runs and 50 total runs per configuration. Hyperparameters include[1] the token size $d(m)$, set by the first linear map (Section III-C) and chosen in $\{351,\ 378\}$ (i.e. $m \in \{26,\ 27\}$)[2]; the h parameter of each Transformer encoder, in $\{3,\ 9\}$[2]; and the number of epoch feature tokens t (Section III-C), chosen among $\{1,\ 3,\ 5,\ 7,\ 10\}$ - with in particular $t = 1$ akin to describing each epoch with a single token, and $t = 7$ corresponding to one token being preserved per channel. We train all folds on the hyperparameters giving the best validation MF1, as well as those with the best F1 score for the N1 stage. Out of those two sets, the results from the set yielding the best average test MF1 is presented in lines 1 to 3 of Table I, with the corresponding hyperparameter set, $d(m)$ and t in the final three columns.

We obtain the best MF1 and N1 F1 scores for $L = 21$, whereas the best macro-averaged accuracy is obtained for $L = 13$. For all values of L, we outperform the state-of-the-art on the considered metrics (except for the MF1 score for $L = 29$). Moreover, all three configurations have around a two-point lead in both macro accuracy and N1 F1 score. While our model favors the smaller token size of $d(m) = 351$ for all values of L, it seems that having a large number of tokens to describe each epoch (at least $t = 5$) is necessary for best performance. Overall, $L = 21$ seems to be a good compromise to capture enough contextual information without burdening our model with irrelevant data. We also investigate the impact of our strict structural preservation by replacing the SP-MHA block of SPDTransNet model with the classic L-MHA (Section 2), all other things being equal (with $L = 21$). Results for this configuration are displayed in line 4 of the table.

We compare ourselves to five models: DeepSleepNet [17], often used as a benchmark, with a pre-trained epoch-wise global feature map submodel followed by a sequence-to-sequence RNN; IITNet [18], the source of our 31 folds, extracting multiple features per epoch through CNNs and comparing them through sequence-wise RNNs; GraphSleepNet [25], expliciting epoch-wise functional connectivity through graph learning; Dequidt et al. [26], utilizing a single-step pretrained visual CNN, who both maximize MF1 performance and rebalance training sets; and Seraphim et al. [24], with a similar approach to ours, though utilizing an alternative whitening (Section III-B) and lacking in structural preservation (cf. line 4 of Table I). These models were re-trained using our methodology - except for oversampling in DeepSleepNet's sequence-to-sequence submodel - using their published

[2] Since $\frac{d(m)}{h}$ must be an integer, potential values for those are limited.

hyperparameters. Finally, as test sets vary between models due to recording-wise border effects, we trim test set borders to enforce uniformity. These results, averaged over all folds, are displayed in lines 5 to 9 of Table I. As we can see, SPDTransNet outperforms all tested State-of-the-Art models, though our lead on Dequidt et al. is minor.

Furthermore, comparing our best results (line 2) to those of lines 4 and 9 indicate that the structural preservation of our SP-MHA improves our model's performance, with or without the influence of our new whitening (Section III-B).

V. Conclusion

We presented SP-MHA, a novel, structure-preserving Multihead Attention bloc, and integrated it into our SPDTransNet model, designed to analyze SPD matrix sequences. We proved said model's capabilities through automatic EEG sleep staging, obtaining a high level of per-stage performance relative to the literature. Beyond this two-step analysis, SPDTransNet can be easily adapted to a variety of problems, for instance by using only a single encoder step and/or implementing a sequence-to-sequence classification scheme.

References

[1] F. Yger, M. Berar, and F. Lotte, "Riemannian approaches in brain-computer interfaces: A review," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 10, pp. 1753–1762, 2017.

[2] N. Hansen, S. D. Müller, and P. Koumoutsakos, "Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES)," *Evolutionary Computation*, vol. 11, no. 1, pp. 1–18, 2003.

[3] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, "Log-Euclidean metrics for fast and simple calculus on diffusion tensors," *Magnetic Resonance in Medicine*, vol. 56, no. 2, pp. 411–421, 2006.

[4] Z. Huang and L. Van Gool, "A riemannian network for SPD matrix learning," in *Association for the Advancement of Artificial Intelligence (AAAI)*, 2017.

[5] R. Chakraborty, J. Bouza, J. H. Manton, and B. C. Vemuri, "ManifoldNet: A deep neural network for manifold-valued data with applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 2, pp. 799–810, 2022.

[6] X. Pennec, P. Fillard, and N. Ayache, "A riemannian framework for tensor computing," *International Journal of Computer Vision*, vol. 66, no. 1, pp. 41–66, Jan 2006.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017.

[8] D. Konstantinidis, I. Papastratis, K. Dimitropoulos, and P. Daras, "Multi-manifold attention for vision Transformers," *IEEE Access*, vol. 11, pp. 123 433–123 444, 2023.

[9] L. He, Y. Dong, Y. Wang, D. Tao, and Z. Lin, "Gauge equivariant Transformer," in *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34. Curran Associates, Inc., 2021, pp. 27 331–27 343.

[10] Z. Li, X. TANG, Z. Xu, X. Wang, H. Yu, M. Chen, and X. Wei, "Geodesic self-attention for 3D point clouds," in *Advances in Neural Information Processing Systems*, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022.

[11] A. Kratsios, B. Zamanlooy, T. Liu, and I. Dokmanić, "Universal approximation under constraints is possible with Transformers," in *International Conference on Learning Representations*, 2022.

[12] Y.-T. Pan, J.-L. Chou, and C.-S. Wei, "MAtt: A manifold attention network for EEG decoding," in *Advances in Neural Information Processing Systems*, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022.

[13] R. Qin, Z. Song, H. Ren, Z. Pei, L. Zhu, X. Shi, Y. Guo, H. Liu, M. Zhang, and Z. Zhang, "BNMTrans: A brain network sequence-driven manifold-based transformer for cognitive impairment detection using EEG," in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 2016–2020.

[14] M. T. Harandi, M. Salzmann, and R. Hartley, "From manifold to manifold: Geometry-aware dimensionality reduction for SPD matrices," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 17–32.

[15] R. B. Berry, R. Brooks, C. Gamaldo, S. M. Harding, R. M. Lloyd, S. F. Quan, M. T. Troester, and B. V. Vaughn, "AASM scoring manual updates for 2017 (version 2.4)," *Journal of Clinical Sleep Medicine*, vol. 13, no. 05, pp. 665–666, 2017. [Online]. Available: https://jcsm.aasm.org/doi/abs/10.5664/jcsm.6576

[16] H. Phan and K. Mikkelsen, "Automatic sleep staging of EEG signals: recent development, challenges, and future directions," *Physiological Measurement*, vol. 43, no. 4, p. 04TR01, apr 2022. [Online]. Available: https://dx.doi.org/10.1088/1361-6579/ac6049

[17] A. Supratak, H. Dong, C. Wu, and Y. Guo, "DeepSleepNet: a model for automatic sleep stage scoring based on raw single-channel EEG," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 11, pp. 1998–2008, Nov 2017.

[18] H. Seo, S. Back, S. Lee, D. Park, T. Kim, and K. Lee, "Intra- and inter-epoch temporal context network (IITNet) using sub-epoch features for automatic sleep scoring on raw single-channel EEG," *Biomedical Signal Processing and Control*, vol. 61, p. 102037, 2020.

[19] H. Phan, O. Y. Chén, M. C. Tran, P. Koch, A. Mertins, and M. De Vos, "XSleepNet: Multi-view sequential model for automatic sleep staging," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 5903–5915, 2022.

[20] H. Phan, K. P. Lorenzen, E. Heremans, O. Y. Chén, M. C. Tran, P. Koch, A. Mertins, M. Baumert, K. B. Mikkelsen, and M. De Vos, "L-SeqSleepNet: Whole-cycle long sequence modelling for automatic sleep staging," *IEEE Journal of Biomedical and Health Informatics*, pp. 1–10, 2023.

[21] A. Guillot and V. Thorey, "RobustSleepNet: Transfer learning for automated sleep staging at scale," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 1441–1451, 2021.

[22] T. Zhu, W. Luo, and F. Yu, "Convolution-and Attention-Based Neural Network for Automated Sleep Stage Classification," *Int J Environ Res Public Health*, vol. 17, no. 11, Jun 2020.

[23] H. Phan, K. Mikkelsen, O. Y. Chén, P. Koch, A. Mertins, and M. De Vos, "SleepTransformer: Automatic sleep staging with interpretability and uncertainty quantification," *IEEE Transactions on Biomedical Engineering*, vol. 69, no. 8, pp. 2456–2467, 2022.

[24] M. Seraphim, P. Dequidt, A. Lechervy, F. Yger, L. Brun, and O. Etard, "Temporal sequences of EEG covariance matrices for automated sleep stage scoring with attention mechanisms," in *Computer Analysis of Images and Patterns*, N. Tsapatsoulis, A. Lanitis, M. Pattichis, C. Pattichis, C. Kyrkou, E. Kyriacou, Z. Theodosiou, and A. Panayides, Eds. Cham: Springer Nature Switzerland, 2023, pp. 67–76.

[25] Z. Jia, Y. Lin, J. Wang, R. Zhou, X. Ning, Y. He, and Y. Zhao, "GraphSleepNet: Adaptive spatial-temporal graph convolutional networks for sleep stage classification." in *IJCAI*, 2020, pp. 1324–1330.

[26] P. Dequidt, M. Seraphim, A. Lechervy, I. I. Gaez, L. Brun, and O. Etard, "Automatic sleep stage classification on EEG signals using time-frequency representation," in *Artificial Intelligence in Medicine*, J. M. Juarez, M. Marcos, G. Stiglic, and A. Tucker, Eds. Cham: Springer Nature Switzerland, 2023, pp. 250–259.

[27] S. Eickhoff and V. Müller, "Functional connectivity," in *Brain Mapping*, A. W. Toga, Ed. Waltham: Academic Press, 2015, pp. 187–201.

[28] C. O'reilly, N. Gosselin, J. Carrier, and T. Nielsen, "Montreal archive of sleep studies: an open-access resource for instrument benchmarking and exploratory research," *Journal of sleep research*, vol. 23, no. 6, pp. 628–635, 2014.

[29] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 2623–2631.