Stiefel-SPD Manifold Graph Convolution for End-to-End EEG Learning

Imad Eddine Tibermacine, *Member, IEEE*, Samuele Russo, and Christian Napoli

***Abstract*— Electroencephalographic (EEG) decoding relies heavily on second-order (covariance) structure that lives on the manifold of symmetric positive-definite (SPD) matrices. Conventional deep networks in Euclidean space ignore this geometry, distorting geodesic relations between covariances; classical Riemannian pipelines respect SPD metrics but typically use fixed projections and a single global tangent embedding, which limits task adaptivity and incurs cubic costs in the channel dimension. We propose a fully geometry-consistent architecture that preserves manifold structure end-to-end while remaining trainable at scale. A compact depthwise-separable convolutional neural network (CNN) produces features whose regularized covariances lie on the SPD manifold. A learnable orthonormal projection, optimized on the Stiefel manifold via Riemannian stochastic gradient descent (SGD) with QR-factorization (QR) retraction, reduces dimensionality without breaking positive-definiteness and preserves an eigenvalue floor. We then perform tangent space graph-SPD aggregation on a scalp *k*-nearest-neighbor graph—neighbor covariances are transported to the reference tangent space, attention-averaged, and mapped back via the exponential—followed by a log-Euclidean mapping and linear softmax classification. This Stiefel→Graph-SPD→ log chain explains why full geometric consistency matters: it avoids Euclidean shortcuts, keeps all intermediates SPD, and makes log/exp costs cubic in the reduced rank *d*. In cross-subject evaluation on three public datasets, the model attains 83.2%/81.5%/79.7% accuracy with improved macro-F_1, strong separability (macro-AUROC $\approx$ 0.90), and well-calibrated probabilities (ECE $\leq$ 0.04), outperforming strong Euclidean CNNs and Riemannian baselines while remaining computationally pragmatic.**

***Index Terms*— EEG, Riemannian geometry, Stiefel manifold, graph convolution.**

Received 23 May 2025; revised 28 August 2025 and 16 December 2025; accepted 5 January 2026. Date of publication 12 January 2026; date of current version 16 January 2026. *(Corresponding author: Imad Eddine Tibermacine.)*

Imad Eddine Tibermacine is with the Department of Computer, Automation and Management Engineering, Sapienza University of Rome, 00185 Rome, Italy (e-mail: tibermacine@diag.uniroma1.it).

Samuele Russo is with the Department of Psychology, Sapienza University of Rome, 00185 Rome, Italy (e-mail: samuele.russo@uniroma1.it).

Christian Napoli is with the Department of Computer, Automation and Management Engineering, Sapienza University of Rome, 00185 Rome, Italy, also with the Institute for Systems Analysis and Computer Science, Italian National Research Council, 00185 Rome, Italy, and also with the Department of Computational Intelligence, Czestochowa University of Technology, 42201 Częstochowa, Poland (e-mail: cnapoli@diag.uniroma1.it).

Digital Object Identifier 10.1109/TNSRE.2026.3652858

I. Introduction

ELECTROENCEPHALOGRAPHY signals are a cornerstone of non-invasive brain-computer interfaces (BCIs). These signals encode neural correlates of imagined movements, allowing applications ranging from neurorehabilitation to robotic control [1], [2], [3]. However, EEG data are inherently high-dimensional, non-stationary, and contaminated by artifacts, posing significant challenges for reliable decoding.

Traditional pipelines extract spatial filters that maximize between-class variance. The common spatial patterns (CSP) family—including filter-bank, regularized, and adaptive variants—remains the de-facto baseline [4], [5], [6]. Although effective, CSP requires manual tuning of frequency bands, careful rank selection, and offers only a static snapshot of intrinsically time-varying rhythms [7], [8]. A parallel line of work advocates Riemannian processing: trial covariances are viewed as points on the manifold of SPD matrices, and classification proceeds in an affine- or log-Euclidean tangent space [9], [10], [11]. These methods inherit attractive metric properties but scale cubically with the channel dimension and rely on fixed, handcrafted projections that neglect task-specific temporal information [12], [13].

Deep learning has recently shifted the emphasis toward data-driven representation learning. Compact architectures such as EEGNet [14], Deep/ShallowConvNet [15], temporal–spectral squeeze–excitation [16], [17], and their transformer or spectral-attention successors [18] automatically capture multi-scale spatio-temporal structure and cope better with inter-subject variability. Yet almost all of these networks operate in Euclidean space: convolutional feature maps are vectorized, pooled, or passed through fully connected layers, and any subsequent covariance is treated as an ordinary matrix [19], [20]. Because SPD matrices reside on a curved Riemannian manifold, such Euclidean manipulation distorts geodesic distances and compromises the modeling of cross-channel dependencies [21], [22].

A substantial line of EEG decoding work models trial covariances as elements of the manifold of symmetric SPD matrices, $\mathcal{S}_{++}$, and exploits the induced geometry for classification and transfer. A first family operates directly on $\mathcal{S}_{++}$ with distance-based rules: minimum–distance–to–mean (MDM) classifiers compute class Fréchet means $\{\overline{\mathbf{P}}_k\}$ under an affine- or log-Euclidean Riemannian metric and predict $\hat{y} = \arg\min_k d(\mathbf{P}, \overline{\mathbf{P}}_k)$ [9], [10], [11], [23]. A second family

maps covariances to a tangent space: each SPD matrix is aligned to a reference $\mathbf{R}$ (often the geometric mean) by $\mathbf{R}^{-1/2}\mathbf{P}\,\mathbf{R}^{-1/2}$, then transformed via the matrix logarithm, and the upper triangle is vectorized for a linear classifier. This includes classical tangent-space LDA and more recent tangent-space adaptation (TSA) and Riemannian Procrustes alignment (RPA) [23], [24], which use congruence transforms to reduce cross-subject/session shift before classification. In our experiments (Table IV), these Riemannian pipelines already outperform Euclidean CNNs, confirming that respecting covariance geometry is a strong inductive bias [23], [24].

More recent learnable SPD networks replace fixed projections with differentiable manifold layers. SPDNet-style architectures apply SPD-preserving bilinear mappings $\mathbf{P} \mapsto \mathbf{B}^\top\mathbf{P}\,\mathbf{B}$, eigenvalue rectification, and SPD-aware normalization, followed by a tangent-space classifier [24], [25]. EEG-specific variants such as TSMNet, TSA, and semi-supervised domain adaptation (SSDA) integrate these blocks with domain-alignment losses to narrow the gap between calibration-free transfer and supervised adaptation across multiple BCI datasets [25], [26]. In parallel, graph-aware decoders incorporate electrode topology: Graph-CSPNet builds CSP features on a scalp k-NN graph and propagates them with graph convolutions [27]; MAtt applies graph attention over node-wise covariance descriptors [28]; GDLNet lifts temporal windows to a Grassmann manifold and performs graph convolution in that space [29]. Very recent SPD–graph models apply a logarithmic map to node-wise covariances, aggregate neighbors in a shared tangent space, and map back with the exponential [30], [31], [32]. On Motor Imagery (MI) applications, these manifold–graph hybrids typically reach high-70s to low-80s cross-subject accuracy, underlining the benefit of combining SPD geometry with scalp topology [7], [8], [14], [15], [16], [17], [33].

However, from an EEG-decoding perspective, important limitations remain. Classical tangent-space pipelines use a single global reference and fixed spatial filters, which may blur local scalp structure and ignore task-specific temporal dynamics [9]. Many deep SPD architectures interleave Euclidean layers between manifold blocks or apply log / exp only once, so intermediate representations leave $\mathcal{S}_{++}$ and geometric guarantees are weakened. Graph-based variants often aggregate all nodes in a shared tangent space or revert to Euclidean message passing after an initial log, which can distort geodesic relations when covariances differ strongly across the montage [23], [24]. Moreover, most methods operate on full-rank covariances, so each log / exp step costs $\mathcal{O}(C^3)$ in the channel dimension C. These observations motivate our design: we first learn a task-adaptive orthonormal Stiefel projection to reduce dimension while preserving SPD structure and an eigenvalue floor, then perform node-wise log →attention→ exp aggregation on the scalp graph so that (i) all intermediates remain in $\mathcal{S}_{++}$, (ii) topology is exploited explicitly, and (iii) matrix log / exp operations are cubic in a reduced rank $d \ll C$ [34], [35], [36], [37], [38].

To address these limitations in EEG decoding, we propose a geometry-consistent framework that is explicitly tailored to trial-wise covariances and scalp topology [39], [40], [41], [42], [43], [44]. A compact convolutional encoder first learns subject-agnostic spatio–temporal features, from which we form regularized covariances in $\mathcal{S}_{++}^{D}$. These are then projected via an orthonormal Stiefel congruence $\mathbf{W} \in \mathcal{S}t(d, D)$ (trained with QR-based Riemannian SGD), which preserves positive-definiteness and an eigenvalue floor while reducing the effective rank from D to d. Reduced SPD matrices serve as node attributes in a scalp k-NN graph; for each channel, neighbor covariances are transported to the reference node's tangent space, attention-aggregated, and mapped with the matrix exponential, which produces a node-wise Graph–SPD layer that never leaves $\mathcal{S}_{++}$. A final log–Euclidean map vectorizes the graph-refined covariances for linear softmax classification.

We introduce an end-to-end EEG decoder that (i) learns an orthonormal Stiefel projection to reduce covariance rank while preserving SPD structure and an eigenvalue floor, and (ii) performs node-wise tangent-space Graph–SPD aggregation on the scalp graph with log / exp mappings that keep all intermediates SPD. This yields an EEG-centric, geometry-consistent pipeline with reduced log / exp cost (cubic in $d \ll D$) and improved cross-subject performance on three public MI/ERN datasets (Table IV).

II. Methodology

In this section, we present our end-to-end approach for EEG classification (Figure 1).

A. Data Representation and Preprocessing

Let $\{\mathbf{X}_i\}_{i=1}^{N}$ denote raw EEG trials, each $\mathbf{X}_i \in \mathbb{R}^{C\times T}$ containing C channels and T time samples. To mitigate inter-trial variability, we apply a trial-wise z-score normalization with a small variance offset:

$$\mu_i = \frac{1}{C\,T}\sum_{c=1}^{C}\sum_{t=1}^{T} X_{i,c,t}, \tag{1}$$

$$\lambda_i^2 = \frac{1}{C\,T}\|\mathbf{X}_i - \mu_i\|_F^2 + \epsilon_{\text{var}}, \quad \epsilon_{\text{var}} = 10^{-8}. \tag{2}$$

The normalized trial is then

$$\mathbf{X}_i^{\text{norm}} = \frac{\mathbf{X}_i - \mu_i\,\mathbf{1}^\top}{\lambda_i}, \quad \mathbf{1} \in \mathbb{R}^T(\text{all-ones vector}). \tag{3}$$

If spatial normalization is needed, one may subtract μ_i on a per-channel basis rather than globally. The trials are subsequently reshaped into a 4D tensor $\mathcal{X} \in \mathbb{R}^{N\times 1\times C\times T}$ to match 2D convolutional inputs (batch, channel, height, width).

B. Spatio-Temporal Feature Extraction

We adopt a compact EEGNet-style front-end that factorizes temporal and spatial filtering. Each trial is reshaped as $\mathbf{X} \in \mathbb{R}^{1\times C\times T}$ (channel-first).

1) Temporal Filtering: A standard convolution learns temporal filters:

$$\mathbf{Y}^{(1)} = \phi(\text{BN}(\mathbf{W}_t * \mathbf{X})), \qquad \mathbf{W}_t \in \mathbb{R}^{F_1\times 1\times 1\times K_t}, \tag{4}$$

yielding $\mathbf{Y}^{(1)} \in \mathbb{R}^{F_1\times C\times T}$.

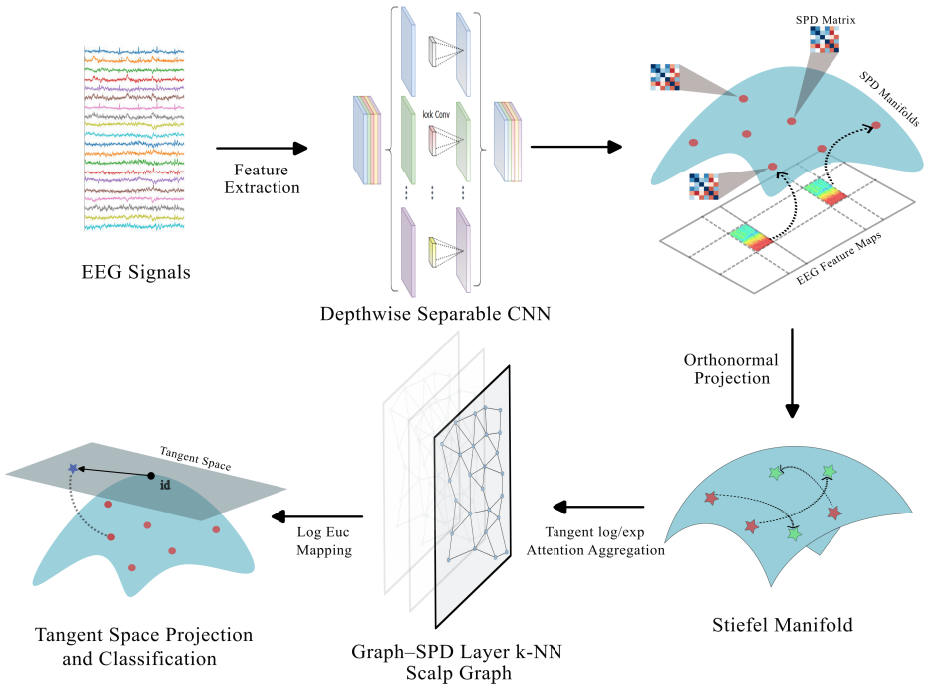

Fig. 1. Illustration of the architecture of the proposed model.

2) Depthwise Spatial Filtering: A depthwise spatial convolution then learns channel mixing per temporal filter (groups = F_1):

$$\mathbf{Y}^{(2)} = \phi(\mathrm{BN}(\mathbf{W}_s * \mathbf{Y}^{(1)})), \qquad \mathbf{W}_s \in \mathbb{R}^{F_1 D \times 1 \times C \times 1}, \tag{5}$$

producing $\mathbf{Y}^{(2)} \in \mathbb{R}^{D \times T}$ after appropriate reshaping/merging of the depth multiplier.

We denote the product of spectral norms by $G_\theta := \|\mathbf{W}_t\|_{\mathrm{op}}\|\mathbf{W}_s\|_{\mathrm{op}}\|\mathbf{W}_2\|_{\mathrm{op}}$.

The depth-wise kernel contains one spatial filter per input channel: $\mathbf{W}_1 \in \mathbb{R}^{C \times 1 \times 1 \times C}$, producing C feature maps. The separable temporal kernel acts channel-wise in time: $\mathbf{W}_2 \in \mathbb{R}^{1 \times K_2 \times 1 \times D}$, so each of the D output channels is the result of a K_2-long 1-D temporal filter.

3) Separable Temporal Filtering: A second separable convolution extracts local temporal patterns:

$$\mathbf{Y}^{(3)} = \phi\Big(\mathrm{BN}\big(\mathbf{W}_2 * \mathbf{Y}^{(2)}\big)\Big), \qquad \mathbf{W}_2 \in \mathbb{R}^{1 \times K_2 \times D \times 1}, \tag{6}$$

with $K_2 = 5$ (zero-padding = 2) $\Rightarrow\approx$ 20 ms at 250 Hz. The final feature map $\mathbf{Y} \equiv \mathbf{Y}^{(3)} \in \mathbb{R}^{D \times T}$ embeds each time-point into a D-dimensional spatio-spectral vector.

C. SPD Matrix Construction

To capture second-order statistics we compute node-wise regularized covariances of the features:

$$\mathbf{P}_c = \underbrace{\tfrac{1}{T-1}\mathbf{Y}_c\mathbf{Y}_c^\top}_{\hat{\mathbf{P}}_c} + \epsilon \mathbf{I}_D, \qquad c = 1, \ldots, C, \tag{7}$$

Adding $\epsilon\mathbf{I}_D$ ensures $\mathbf{P} \in \mathcal{S}_{++}^D$ even when $T < D$. Let $\lambda_{\min}(\hat{\mathbf{P}})$ be the smallest eigenvalue of the non-regularized sample covariance $\hat{\mathbf{P}}$. Then $\lambda_{\min}(\mathbf{P}) = \lambda_{\min}(\hat{\mathbf{P}}) + \epsilon \geq \epsilon$, a bound used later in the robustness certificate (Prop. 1).

Lemma 1 (Regularized Covariance is SPD): For every $\mathbf{Y}$ and $\epsilon > 0$, the matrix in (7) is strictly positive-definite.

Proof: For any $\mathbf{v} \neq \mathbf{0}$, $\mathbf{v}^\top\mathbf{P}\mathbf{v} = \frac{1}{T-1}\|\mathbf{Y}^\top\mathbf{v}\|_2^2 + \epsilon\|\mathbf{v}\|_2^2 > 0$. ∎

D. Dimension Reduction on the Stiefel Manifold

When the feature dimension D is large, we compress each covariance into a lower d-dimensional subspace ($d \ll D$) via an orthonormal projection $\mathbf{W} \in \mathcal{S}t(d, D)$,

$$\mathcal{S}t(d, D) = \big\{\mathbf{W} \in \mathbb{R}^{D \times d} : \mathbf{W}^\top\mathbf{W} = \mathbf{I}_d\big\}. \tag{8}$$

Because the columns are orthonormal, $\|\mathbf{W}\|_{\mathrm{op}} = 1$, a fact used in the generalization bound (Section III). The projected covariance is

$$\mathbf{P}_c' = \mathbf{W}^\top\mathbf{P}_c\,\mathbf{W}, \qquad c = 1, \ldots, C, \tag{9}$$

which remains SPD since congruence with $\mathbf{W}$ preserves positive-definiteness.

Eigenvalue Floor by Stiefel Congruence: Since $\mathbf{W}^\top\mathbf{W} = \mathbf{I}_d$, $\lambda_{\min}(\mathbf{P}') = \lambda_{\min}(\mathbf{W}^\top\mathbf{P}\,\mathbf{W}) \geq \lambda_{\min}(\mathbf{P}) \geq \epsilon$ (Eq. (7)).

1) Riemannian Geometry of $\mathcal{S}t(d, D)$: The tangent space at $\mathbf{W}$ is

$$T_{\mathbf{W}}\mathcal{S}t(d, D) = \big\{\boldsymbol{\xi} \in \mathbb{R}^{D \times d} : \mathbf{W}^\top\boldsymbol{\xi} + \boldsymbol{\xi}^\top\mathbf{W} = 0\big\}. \tag{10}$$

To keep updates on the manifold we use the **QR retraction** [45]:

$$R_{\mathbf{W}}(\boldsymbol{\xi}) = \mathrm{qf}\big(\mathbf{W} + \boldsymbol{\xi}\big), \tag{11}$$

where qf(·) returns the Q factor of a thin QR decomposition with $\mathrm{diag}(\mathbf{R}) \geq 0$ to remove sign ambiguity. Projecting the

Euclidean gradient $\nabla_{\text{Euc}}\mathbf{W}$ onto the tangent space gives the Riemannian gradient

$$\nabla_{\mathcal{S}t}\mathbf{W} = \nabla_{\text{Euc}}\mathbf{W} - \mathbf{W}\,\text{sym}\big(\mathbf{W}^\top\nabla_{\text{Euc}}\mathbf{W}\big),$$
$$\text{sym}(\mathbf{A}) := \tfrac{1}{2}(\mathbf{A}+\mathbf{A}^\top). \tag{12}$$

An orthonormal $\mathbf{W}$ avoids arbitrary scalings (better conditioning), preserves SPD by congruence and the eigenvalue floor, and reduces d so per-node log/exp in Graph–SPD is cubic in d (not D). Unconstrained linear maps break these properties and empirically destabilize training (cf. ablations).

E. Log-Euclidean Tangent-Space Mapping

For each reduced covariance $\mathbf{P}' \in \mathbb{R}^{d\times d}$ we apply the principal matrix logarithm:

$$\mathbf{P}' = \mathbf{U}\boldsymbol{\lambda}\mathbf{U}^\top, \mathbf{L} = \log(\mathbf{P}') = \mathbf{U}\,\log\boldsymbol{\lambda}\,\mathbf{U}^\top, \tag{13}$$

where $\boldsymbol{\lambda} = \text{diag}(\lambda_1,\ldots,\lambda_d)$, $\lambda_i > 0$. For numerical safety we clamp $\lambda_i \geq \epsilon_{\text{eig}} = 10^{-12}$.

1) Feature Vector: Because $\mathbf{L}$ is symmetric we vectorise only its upper triangle, scaling off-diagonal entries by $\sqrt{2}$:

$$\mathbf{z} = \sqrt{2}\,\text{vec}_{\text{triu}}(\mathbf{L}) \in \mathbb{R}^{d(d+1)/2}. \tag{14}$$

2) Log-Euclidean Metric: The distance $d_{\text{LogE}}(\mathbf{P}_1,\mathbf{P}_2) = \|\log\mathbf{P}_1 - \log\mathbf{P}_2\|_F$ is orthonormally congruence-invariant: for every orthonormal matrix $\mathbf{Q}\in O(d)$,

$$d_{\text{LogE}}(\mathbf{P}_1,\mathbf{P}_2) = d_{\text{LogE}}\big(\mathbf{Q}^\top\mathbf{P}_1\mathbf{Q},\ \mathbf{Q}^\top\mathbf{P}_2\mathbf{Q}\big). \tag{15}$$

This follows from $\log(\mathbf{Q}^\top\mathbf{P}\mathbf{Q}) = \mathbf{Q}^\top\log(\mathbf{P})\mathbf{Q}$ and orthonormal invariance of the Frobenius norm. The metric is **not** invariant under general GL(d) congruence—only under simultaneous left-right multiplication by the same orthonormal matrix. In what follows we use d_{LogE} mainly for its closed-form derivative and numerical stability. It is $1/\lambda_{\min}(\mathbf{P}')$-Lipschitz, where $\lambda_{\min}$ is the smallest eigen-value of $\mathbf{P}'$ (see Prop. 1).

Then we use a fully-connected layer $\mathbf{V} \in \mathbb{R}^{K\times d(d+1)/2}$ followed by softmax maps $\mathbf{z}$ to class logits for the K-way task.

The log–Euclidean distance is invariant under orthonormal congruence but not under general affine transformations. We adopt it for numerical stability and closed-form derivatives; affine-invariant alternatives could sharpen class boundaries at higher cost (see Discussion).

F. End-to-End Gradient Flow

Training proceeds by back-propagating through (i) the matrix logarithm $\mathbf{P}'\mapsto\mathbf{L}$ and (ii) the Stiefel parameter $\mathbf{W}$.

1) Matrix-Logarithm Gradient (Daleckiĭ–Kreĭn): Let $\mathbf{P}' = \mathbf{U}\boldsymbol{\lambda}\mathbf{U}^\top$ with $\boldsymbol{\lambda} = \text{diag}(\lambda_1,\ldots,\lambda_d)$, $\lambda_i \geq \epsilon_{\text{eig}}$. For an upstream gradient $\partial\ell/\partial\mathbf{L}$ we have

$$\frac{\partial\ell}{\partial\mathbf{P}'} = \mathbf{U}\big(\mathbf{K}\odot(\mathbf{U}^\top(\partial\ell/\partial\mathbf{L})\mathbf{U})\big)\mathbf{U}^\top, \tag{16}$$

$$K_{ij} = \begin{cases} \dfrac{\log\lambda_i - \log\lambda_j}{\lambda_i - \lambda_j}, & \lambda_i \neq \lambda_j, \\ \dfrac{1}{\lambda_i}, & \lambda_i = \lambda_j. \end{cases} \tag{17}$$

Because $\lambda_i \geq \epsilon_{\text{eig}}$, $\mathbf{K}$ is well-conditioned.

2) Stiefel-Projection Gradient: From $\mathbf{P}' = \mathbf{W}^\top\mathbf{P}\mathbf{W}$, matrix calculus [46] yields

$$\frac{\partial\ell}{\partial\mathbf{W}} = 2\,\mathbf{P}\mathbf{W}\,\frac{\partial\ell}{\partial\mathbf{P}'} - \mathbf{W}\,\text{sym}\Big(\mathbf{W}^\top\big(2\,\mathbf{P}\mathbf{W}\,\tfrac{\partial\ell}{\partial\mathbf{P}'}\big)\Big). \tag{18}$$

Algorithm 1 Single Retraction per SGD Step; No Inner Iteration

Require: $\mathbf{W}_t \in \mathcal{S}t(d,D)$, Euclidean gradient $\nabla_{\text{Euc}} \in \mathbb{R}^{D\times d}$, step size η_t

Project $\nabla_{\mathcal{S}t} \leftarrow \nabla_{\text{Euc}} - \mathbf{W}_t\,\text{sym}\big(\mathbf{W}_t^\top\nabla_{\text{Euc}}\big)$
Ambient step $\widetilde{\mathbf{W}} \leftarrow \mathbf{W}_t - \eta_t\nabla_{\mathcal{S}t}$
Retract (QR) $\mathbf{W}_{t+1} \leftarrow \text{qf}(\widetilde{\mathbf{W}})$

We then project onto $T_{\mathbf{W}}\mathcal{S}t(d,D)$ and retract using Algorithm 1, ensuring $\mathbf{W}^\top\mathbf{W} = \mathbf{I}_d$ after every update.

We compute the usual Euclidean gradient w.r.t. $\mathbf{W}$, drop its components that would leave the Stiefel manifold, and then take a small step followed by QR to restore orthonormality.

G. Convergence Analysis

The following assumptions are tied to theorems presented in this manuscript:

Assumption 1 (Eigenvalue Floor): Covariances are regularized as in (7), hence $\lambda_{\min}(\mathbf{P}) \geq \epsilon$; Stiefel congruence preserves this lower bound.

Assumption 2 (Bounded Tangent Update): For the Graph–SPD layer, the aggregated tangent matrix satisfies $\|A_c\|_2 \leq \tau$ for all nodes c (a standard Lipschitz–type control on the attention output).

Assumption 3 (Smoothness and Unbiasedness): The loss is L–smooth on $\mathbb{R}^p\times\mathcal{S}t(d,D)$, Euclidean gradient estimates are unbiased with bounded variance, and QR retraction is used for feasibility.

Algorithm 2 End-to-End Geometry-Consistent Training

Require: Normalized trials $\{\mathbf{X}_i^{\text{norm}}\}$, batch size B, epochs E, learning rates $\{\eta_t\}$, regularizer ϵ

Initialise $\mathbf{W}$ **(Stiefel)** by random QR; CNN weights $(\mathbf{W}_1,\mathbf{W}_2)$ (He); graph MLP Φ (He); classifier $\mathbf{V}$ (uniform)
for epoch $= 1,\ldots,E$ **do**
  for mini-batch $\mathcal{B} \subset \{\mathbf{X}_i^{\text{norm}}\}$ **do**
    CNN encoder: $\mathbf{Y} = f_\theta(\mathbf{X})$
    Covariance: $\mathbf{P} = \frac{1}{T-1}\mathbf{Y}\mathbf{Y}^\top + \epsilon\mathbf{I}_D$
    orthonormal projection: $\mathbf{P}'_c = \mathbf{W}^\top\mathbf{P}_c\,\mathbf{W} \quad \forall c$
    Graph–SPD aggregation: // Alg. 3
    $\mathbf{P}^{\text{agg}} = \text{GraphSPD}\big(\mathbf{P}', \mathcal{G}, \Phi\big)$
    Log map: $\mathbf{L} = \log(\mathbf{P}^{\text{agg}})\lambda_i \geq \epsilon_{\text{eig}}$ clamp
    Vectorise & classify: $\mathbf{z} = \sqrt{2}\,\text{vec}_{\text{triu}}(\mathbf{L})$
    $\hat{\mathbf{y}} = \text{softmax}(\mathbf{V}\mathbf{z})$
    Loss: $\ell = \text{CE}(\hat{\mathbf{y}},\mathbf{y}) + \mathcal{L}_{\text{floor}}$
    Backward: compute $\{\nabla_{\mathbf{V}},\nabla_\Phi,\nabla_{\mathbf{W}},\nabla_{\mathbf{W}_1},\nabla_{\mathbf{W}_2}\}$
    Stiefel update: $\mathbf{W} \leftarrow$ Alg. 11$(\mathbf{W},\nabla_{\text{Euc}},\eta_t)$
    Adam update: $(\mathbf{W}_1,\mathbf{W}_2,\Phi,\mathbf{V})$
  end for
end for

Let $\mathcal{M} = \mathbb{R}^p \times \mathcal{S}t(d, D)$ and $\Phi_t = (\theta_t, \mathbf{W}_t)$ be the iterates of Algorithm 2. With $\eta_t = \eta_0/(1 + \gamma t)$:

Theorem 1 (Non-Convex Rate on Product Manifold): If the loss ℓ is L-smooth and bounded below, then

$$\min_{0\le k<T} \mathbb{E}\Big[\|\nabla_{\mathcal{M}} \ell(\Phi_k)\|^2\Big] \;\le\; \mathcal{O}\big(T^{-1/2}\big). \tag{19}$$

Under the Polyak–Łojasiewicz condition in a neighborhood of a local minimum, we further obtain linear convergence in expectation.

The proof adapts Boumal's Riemannian SGD analysis [47] to the mixed Euclidean–Stiefel setting.

H. Implementation Details

Our SPD log/exp operations scale with the reduced rank d rather than the feature dimension D thanks to the Stiefel step. This is the key difference to pipelines that either (i) operate on global covariances without dimension reduction (cost $\mathcal{O}(B\,D^3)$ per batch for log/exp), or (ii) aggregate all nodes in a single global tangent space before reduction (again cubic in D). In contrast, our graph–SPD layer applies node-wise log/exp at cost $\mathcal{O}(B\,C\,d^3)$ with $d \ll D$, plus one QR retraction $\mathcal{O}(Dd^2)$ per update step (independent of B). Empirically, this reduces wall-clock by a factor of 3–6× when moving from D=64 to d=48 while improving accuracy (cf. Table V).

I. Overall Algorithm

Training stops after E epochs or when a 5-epoch moving-average relative loss decrease $< 10^{-4}$.

The pipeline thus propagates gradients seamlessly through the matrix logarithm layer and the orthonormal projection, with convergence and generalization Guaranties provided in Sections IV and III.

To certify the condition $\lambda_{\min} \ge \kappa$ in practice, we add a hinge–squared penalty on the post-Stiefel covariances:

$$\mathcal{L}_{\text{floor}} = \alpha \sum_c \Big[\kappa - \lambda_{\min}\big(\mathbf{P}'_c\big)\Big]_+^2, \quad [x]_+ = \max(0, x), \tag{20}$$

with $\kappa \le \epsilon$ (e.g. κ=5×10^{-6} if ϵ=10^{-5}) and small α (we use 10^{-3}). The eigenvalue is reused from the log-map eigendecomposition. The total loss is $\mathcal{L} = \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{floor}}$.

In our runs, all $\mathbf{P}'_c$ satisfied $\lambda_{\min}\big(\mathbf{P}'_c\big) \ge \kappa$, so the penalty remained inactive.

III. Generalization Error Analysis

A. Rademacher Complexity Bound

We bound the capacity of our proposed model. Let $f_\theta : \mathbb{R}^{C\times T} \to \mathbb{R}^{D\times T}$ be the two-layer CNN with weights $\theta = (\mathbf{W}_1, \mathbf{W}_2)$, and define $\phi_{\mathbf{W}}(\mathbf{Y}) := \log(\mathbf{W}^\top \mathbf{P} \mathbf{W})$ for $\mathbf{W} \in \mathcal{S}t(d, D)$. The hypothesis space is

$$\mathcal{H} = \Big\{ h(\mathbf{X}) = \operatorname{softmax}\big(\mathbf{V}\,\operatorname{vec}_{\text{triu}} \circ \phi_{\mathbf{W}} \circ f_\theta(\mathbf{X})\big) : \|\mathbf{V}\|_2 \le B_V \Big\}. \tag{21}$$

Theorem 2 (Empirical Rademacher Complexity): For any sample $\{(\mathbf{X}_i, y_i)\}_{i=1}^N$ and $h \in \mathcal{H}$, with probability $\ge 1 - \delta$:

$$\mathbb{E}\big[\ell(h(\mathbf{X}), y)\big] \le \frac{1}{N}\sum_{i=1}^N \ell\big(h(\mathbf{X}_i), y_i\big) + 2\,L_{\text{Lip}}\,\mathfrak{R}_N(\mathcal{H}) + 3\sqrt{\frac{\log(2/\delta)}{2N}}. \tag{22}$$

where

$$\mathfrak{R}_N(\mathcal{H}) \;\le\; \frac{B_V\,G_\theta\,\sqrt{2d(d+1)}}{N}\,\|\overline{\mathbf{X}}\|_F, \tag{23}$$

$\overline{\mathbf{X}} = \frac{1}{N}\sum_i \mathbf{X}_i$ and $G_\theta = \|\mathbf{W}_1\|_{\text{op}}\|\mathbf{W}_2\|_{\text{op}}$.

Proof: Let $\{\sigma_i\}_{i=1}^N$ be i.i.d. Rademacher variables and set

$$g_{\theta,\mathbf{W}}(\mathbf{X}) = \mathbf{V}\,\operatorname{vec}_{\text{triu}}\big(\phi_{\mathbf{W}} \circ f_\theta(\mathbf{X})\big),$$
$$\phi_{\mathbf{W}}(\mathbf{Y}) = \log\big(\mathbf{W}^\top \mathbf{P}\,\mathbf{W}\big). \tag{24}$$

Because the softmax operator is 1-Lipschitz with respect to its logits, the empirical Rademacher complexity of $\mathcal{H}$ satisfies

$$\mathfrak{R}_N(\mathcal{H}) \;=\; \frac{1}{N}\,\mathbb{E}_{\sigma,\mathcal{S}}\left[\sup_{\theta,\mathbf{W},\mathbf{V}} \sum_{i=1}^N \sigma_i\, g_{\theta,\mathbf{W}}(\mathbf{X}_i)\right]. \tag{25}$$

Step 1 (Metric-entropy of the Stiefel factor): For every $\epsilon > 0$ the compact Stiefel manifold admits an ϵ-cover with $\mathcal{N}(\epsilon) \le (C/\epsilon)^{dD - \frac{d(d-1)}{2}}$ [48]. Dudley's integral therefore gives, for the linear class $\mathcal{G} = \{\mathbf{X} \mapsto \mathbf{W}^\top \mathbf{X} : \mathbf{W} \in \mathcal{S}t\}$,

$$\mathbb{E}_\sigma\left[\sup_{\mathbf{W}\in\mathcal{S}t} \sum_{i=1}^N \sigma_i \langle \mathbf{W}^\top \mathbf{X}_i, \mathbf{a}_i\rangle\right] \;\le\; c_0\,\|\overline{\mathbf{X}}\|_F\,\sqrt{d(d+1)\,N}, \tag{26}$$

for arbitrary unit vectors $\mathbf{a}_i$ and constant $c_0 > 0$.

Step 2 (Vector-valued contraction): The feature extractor f_θ is G_θ-Lipschitz in operator norm, $G_\theta = \|\mathbf{W}_1\|_{\text{op}}\|\mathbf{W}_2\|_{\text{op}}$. Applying the vector-valued contraction inequality for Rademacher averages [49] yields

$$\mathfrak{R}_N(\mathcal{H}) \;\le\; \frac{B_V\,G_\theta}{N}\,\sqrt{2d(d+1)}\,\|\overline{\mathbf{X}}\|_F, \tag{27}$$

where the factor $\|\mathbf{W}\|_{\text{op}}$ is 1 because $\mathbf{W} \in \mathcal{S}t(d, D)$ has orthonormal columns.

Step 3 (Risk bound): Substituting the bound from Step 2 into [50] for a L_{Lip}-Lipschitz loss function completes the proof and gives the inequality claimed in Theorem 2. ∎

B. PAC-Bayes Bounds

1) Manifold-Aware Prior: Project an isotropic Gaussian onto $\mathcal{S}t(d, D)$ via QR retraction to obtain prior Π. For any posterior Q over $\Theta = (\theta, \mathbf{W}, \mathbf{V})$:

Theorem 3 (PAC-Bayes): With probability $1 - \delta$ over n samples,

$$\mathbb{E}_Q\big[\text{err}_{\mathcal{D}}\big] \le \mathbb{E}_Q\big[\widehat{\text{err}}\big] + \sqrt{\frac{\text{KL}(Q\|\Pi) + \log\frac{2\sqrt{n}}{\delta}}{2(n-1)}}. \tag{28}$$

Proof: Let $\Pi = \Pi_{\text{Euc}} \otimes \Pi_{\text{St}}$ where Π_{Euc} is isotropic Gaussian in $\mathbb{R}^p$ and Π_{St} is obtained by mapping an isotropic Gaussian in $\mathbb{R}^{D\times d}$ to $\mathcal{S}t(d, D)$ through QR retraction. By [51] the Jacobian

of QR at orthonormal points is 1, hence $\mathrm{d}\Pi_{\mathrm{St}} = \mathrm{d}P_{\mathrm{Euc}} \circ \mathrm{QR}^{-1}$. For any posterior Q, $\mathrm{KL}(Q\,\|\,\Pi) = \mathrm{KL}(Q_{\mathrm{Euc}}\,\|\,\Pi_{\mathrm{Euc}}) + \mathrm{KL}(Q_{\mathrm{St}}\,\|\,\Pi_{\mathrm{St}})$.

Apply Catoni's PAC–Bayes bound [52] on the product manifold $\mathcal{M} = \mathbb{R}^p \times \mathcal{S}t(d, D)$ and obtain

$$\mathbb{E}_Q[\mathrm{err}] \le \mathbb{E}_Q[\widehat{\mathrm{err}}] + \sqrt{\frac{\mathrm{KL}(Q\|\Pi) + \log\frac{2\sqrt{n}}{\delta}}{2(n-1)}}. \tag{29}$$

The covering-number version follows by upper-bounding the KL with $\Psi(\mathcal{S}t(d,D)) = \mathcal{O}\big(dD\log\frac{1}{\epsilon}\big)$ and using the union bound. ∎

2) Metric-Entropy Term $\Psi(\mathcal{S}t)$: For any $\epsilon > 0$ the ϵ-covering number of $\mathcal{S}t(d,D)$ satisfies $\log\mathcal{N}(\epsilon, \mathcal{S}t, \|\cdot\|_F) \le \big(dD - \frac{d(d-1)}{2}\big)\log(C/\epsilon)$ [48]. Throughout we abbreviate $\Psi(\mathcal{S}t) := dD\log(C/\epsilon)$, so that $\mathrm{KL}(Q\|\Pi) \le \Psi(\mathcal{S}t)$ whenever the support of Q lies inside an ϵ-ball around the prior mean.

IV. Convergence Guaranties Under Non-Convex Objectives

Let $\Phi_t = (\theta_t, \mathbf{W}_t)$ be the Algorithm 2 iterates and $\mathcal{M} = \mathbb{R}^p \times \mathcal{S}t(d,D)$.

Theorem 4 (Non-Convex Convergence): If ℓ is L-smooth and $\eta_t = \eta_0/(1+\gamma t)$, then

$$\min_{0\le k<T} \mathbb{E}\big[\|\nabla_{\mathcal{M}}\ell(\Phi_k)\|^2\big] \le \mathcal{O}(T^{-1/2}), \tag{30}$$

and linear convergence holds inside a Riemannian Polyak–Łojasiewicz region.

Proof: Define the Lyapunov function $\mathcal{V}_t := \mathbb{E}[\ell(\Phi_t)]$. L-smoothness on $\mathcal{M}$ gives

$$\ell(\Phi_{t+1}) \le \ell(\Phi_t) - \eta_t\langle\nabla_{\mathcal{M}}\ell, \Delta_t\rangle + \tfrac{L}{2}\eta_t^2\|\Delta_t\|^2, \tag{31}$$

where Δ_t is the Riemannian gradient step. Take conditional expectation given Φ_t, use unbiasedness and variance σ^2:

$$\mathbb{E}[\ell(\Phi_{t+1})] \le \mathcal{V}_t - \eta_t\mathbb{E}\big[\|\nabla_{\mathcal{M}}\ell(\Phi_t)\|^2\big] + \tfrac{L\sigma^2}{2}\eta_t^2. \tag{32}$$

Sum from $t=0$ to $T-1$ (telescoping) and divide by $\sum_{t=0}^{T-1}\eta_t$:

$$\min_{k<T}\mathbb{E}\|\nabla_{\mathcal{M}}\ell(\Phi_k)\|^2 \le \frac{\mathcal{V}_0 - \ell_{\inf}}{\sum_{t=0}^{T-1}\eta_t} + \frac{L\sigma^2\sum_{t=0}^{T-1}\eta_t^2}{2\sum_{t=0}^{T-1}\eta_t}. \tag{33}$$

With $\eta_t = \eta_0/(1+\gamma t)$, $\sum\eta_t = \Theta(\sqrt{T})$ and $\sum\eta_t^2 = \Theta(\log T)$, giving the $\mathcal{O}(T^{-1/2})$ bound. If the Riemannian Polyak–Łojasiewicz inequality $\frac{1}{2}\|\nabla_{\mathcal{M}}\ell\|^2 \ge \mu(\ell - \ell_*)$ holds locally, then $\mathbb{E}[\ell(\Phi_{t+1}) - \ell_*] \le (1-\mu\eta_t)(\ell(\Phi_t) - \ell_*)$ and linear convergence follows. ∎

A. Riemannian PL Condition

A differentiable function $\ell : \mathcal{M} \to \mathbb{R}$ satisfies the Riemannian Polyak–Łojasiewicz (PL) inequality with constant $\mu > 0$ on a set $\Omega \subseteq \mathcal{M}$ if

$$\tfrac{1}{2}\|\nabla_{\mathcal{M}}\ell(\Phi)\|^2 \ge \mu\big(\ell(\Phi) - \ell_*\big), \qquad \forall\Phi \in \Omega, \tag{34}$$

where ℓ_* is the global minimum over Ω.

V. Stability to Input Perturbations

We analyze how input noise propagates through the proposed Stiefel-SPD-Graph pipeline.

A. Perturbation Model

Let $\mathbf{X}$ be an EEG trial and $\tilde{\mathbf{X}} = \mathbf{X} + \mathbf{E}$ with $\|\mathbf{E}\|_F \le \delta$.

Because the CNN encoder is G_{conv}-Lipschitz in Frobenius norm, $\|\Delta\mathbf{Y}\|_F \le G_{\mathrm{conv}}\|\mathbf{E}\|_F \le G_{\mathrm{conv}}\delta$.

Lemma 2 (Covariance Perturbation): Let $\Delta\mathbf{Y} := \tilde{\mathbf{Y}} - \mathbf{Y}$. Then

$$\|\mathbf{P} - \tilde{\mathbf{P}}\|_F \le \tfrac{2}{T-1}\|\mathbf{Y}\|_F\,\|\Delta\mathbf{Y}\|_F + \|\Delta\mathbf{Y}\|_F^2. \tag{35}$$

Proof: Expand both covariances, subtract, and apply sub-multiplicativity of the Frobenius norm. ∎

Lemma 3 (Log-map Lipschitz): For SPD matrices $\mathbf{A}, \mathbf{B}$,

$$\|\log\mathbf{A} - \log\mathbf{B}\|_F \le \frac{\|\mathbf{A} - \mathbf{B}\|_F}{\min\{\lambda_{\min}(\mathbf{A}), \lambda_{\min}(\mathbf{B})\}}. \tag{36}$$

Proof: Daleckiĭ–Kreĭn formula with the mean-value inequality [53]. ∎

Throughout this section we denote $G_{\mathrm{conv}} := \|\mathbf{W}_1\|_{\mathrm{op}}\,\|\mathbf{W}_2\|_{\mathrm{op}}$, identical to G_θ of Sec. II-B.

Proposition 1 (End-to-end Lipschitz): Let $\tilde{\mathbf{X}} = \mathbf{X} + \mathbf{E}$ with $\|\mathbf{E}\|_F \le \delta$ and denote by $\mathbf{P}, \tilde{\mathbf{P}}$ the regularized covariances (cf. Eq. (7)). Then the tangent features $\mathbf{z}, \tilde{\mathbf{z}}$ satisfy

$$\|z - \tilde{z}\|_2 \;\le\; \delta\,G_{\mathrm{conv}}^2\sqrt{\frac{2}{T-1}}\left(1 + \frac{\delta}{\lambda_{\min}(\mathbf{P})}\right), \tag{37}$$

where $\lambda_{\min}(\mathbf{P})$ is the smallest eigenvalue of the regularized covariance and, by Eq. (7), $\lambda_{\min}(\mathbf{P}) \ge \epsilon$.

Remark 1: The bound improves as $\lambda_{\min}(\mathbf{P})$ increases; because $\lambda_{\min}(\mathbf{P}) \ge \epsilon$ by construction, the regularizer $\epsilon\mathbf{I}$ Guaranties a finite Lipschitz constant even when the sample covariance is rank-deficient.

Proof: Combine Lemma 2 with Lemma 3, use $\|\mathbf{W}\|_{\mathrm{op}} = 1$, and the $\sqrt{2}$–scaled vectorization in Sec. II-E. ∎

Remark 2: If $\lambda_{\min}(\mathbf{P}) \ge \gamma$ for all trials, the network is globally $\big(\frac{G_{\mathrm{conv}}^2}{\gamma}\big)$-Lipschitz.

Remark 3: The bound shows that robustness improves as $\gamma = \lambda_{\min}(\hat{\mathbf{P}})$ grows, providing a concrete motivation for the covariance regularizer $\epsilon\mathbf{I}$ in Eq. (7).

VI. Graph-Structured SPD Processing

A. Scalp Graph Construction

Each node c is associated with an SPD descriptor $\mathbf{P}'_c \in \mathcal{S}^d_{++}$ obtained after the Stiefel projection step (Sec. II-D). The Graph–SPD layer propagates these node-wise SPD descriptors along the scalp graph.

Electrode 3-D coordinates are converted to a k-nearest-neighbor graph:

$$A_{ij} = \begin{cases} 1, & j \in \mathcal{N}_k(i), \\ 0, & \text{otherwise.} \end{cases} \tag{38}$$

Self-loops are added and the graph is symmetrized, giving $\widetilde{\mathbf{A}} = \mathbf{A} + \mathbf{A}^\top + \mathbf{I}$. The degree matrix $\widetilde{\mathbf{D}} = \mathrm{diag}(\widetilde{\mathbf{A}}\mathbf{1})$ leads to the symmetric normalized Laplacian

$$\mathbf{L}_{\mathrm{sym}} = \mathbf{I} - \widetilde{\mathbf{D}}^{-1/2}\,\widetilde{\mathbf{A}}\,\widetilde{\mathbf{D}}^{-1/2}. \tag{39}$$

This choice is preferred to $\widetilde{\mathbf{D}}^{-1}\widetilde{\mathbf{A}}$ because it is real–symmetric and thus yields orthonormal eigenvectors, facilitating spectral analysis [54].

B. Tangent-Space Graph Convolution

We perform affine-invariant tangent mapping at the reference node: neighbor matrices are whitened by $\mathbf{P}_c'^{-1/2}$, mapped with the matrix logarithm, attention-aggregated in the tangent space, and mapped back via the exponential.

We adopt a reference-centered (affine-invariant) tangent mapping: neighbor matrices are first transported to the tangent space at the reference node, aggregated with learnable attention weights, and mapped back to $\mathcal{S}_{++}^d$:

(1) **Logarithmic transport**

$$\mathbf{L}_j^{(c)} = \log\!\left(\mathbf{P}_c'^{-1/2}\,\mathbf{P}_j'\,\mathbf{P}_c'^{-1/2}\right), \qquad j \in \mathcal{N}_k(c). \tag{40}$$

We feed the attention MLP with a vector embedding of the symmetric tangent matrix:

$$\mathbf{u}_j^{(c)} = \sqrt{2}\,\mathrm{vec}_{\mathrm{triu}}\!\left(\mathbf{L}_j^{(c)}\right) \in \mathbb{R}^{d(d+1)/2}. \tag{41}$$

(2) **Attention aggregation**

$$\alpha_{cj} = \mathrm{softmax}_j\!\left(\mathrm{MLP}(\mathbf{u}_j^{(c)})\right),$$
$$\mathbf{L}_c^{\mathrm{agg}} = \sum_{j\in\mathcal{N}_k(c)} \alpha_{cj}\,\mathbf{L}_j^{(c)}. \tag{42}$$

(3) **Exponential retraction**

$$\mathbf{P}_c' \leftarrow \mathbf{P}_c'^{1/2}\exp(\mathbf{L}_c^{\mathrm{agg}})\mathbf{P}_c'^{1/2}. \tag{43}$$

Algorithm 3 Tangent-Space Graph SPD Layer (synchronous update)

Require: Graph $\mathcal{G}$, input SPD set $\{\mathbf{P}_c^{\mathrm{in}}\}$ (post-Stiefel)
1: **for** node $c \in \mathcal{V}$ **do**
2: $\mathbf{L}_j^{(c)} \leftarrow \log\!\left(\mathbf{P}_c^{\mathrm{in}\,-1/2}\mathbf{P}_j^{\mathrm{in}}\mathbf{P}_c^{\mathrm{in}\,-1/2}\right) \quad \forall j \in \mathcal{N}_k(c)$
3: $\mathbf{u}_j^{(c)} \leftarrow \sqrt{2}\,\mathrm{vec}_{\mathrm{triu}}\!\left(\mathbf{L}_j^{(c)}\right)$
4: $\alpha_{cj} \leftarrow \mathrm{softmax}_j\!\left(\mathrm{MLP}(\mathbf{u}_j^{(c)})\right)$
5: $\mathbf{L}_c^{\mathrm{agg}} \leftarrow \sum_j \alpha_{cj}\mathbf{L}_j^{(c)}$
6: $\mathbf{P}_c^{\mathrm{out}} \leftarrow \mathbf{P}_c^{\mathrm{in}\,1/2}\exp(\mathbf{L}_c^{\mathrm{agg}})\mathbf{P}_c^{\mathrm{in}\,1/2}$
7: **end for**
8: **return** $\{\mathbf{P}_c^{\mathrm{out}}\}$ (single forward pass; no inner iterations)

Stacking (40)–(43) across all nodes produces a Tangent-Space Graph SPD layer whose forward pass is summarized in Algorithm 3. Omitting non-linearities, the layer can also be written in matrix form

Eqs. (40)–(43) can be viewed as the manifold analogue of neighborhood smoothing: we (i) map each neighbor $\mathbf{P}_j'$ to the tangent space at the reference node $\mathbf{P}_c'$ via the reference-centered log map $\log(\mathbf{P}_c'^{-1/2}\mathbf{P}_j'\mathbf{P}_c'^{-1/2})$, (ii) compute a weighted average in that Euclidean tangent space (attention), and (iii) map the result back to $\mathcal{S}_{++}^d$ with the exponential and a congruence by $\mathbf{P}_c'^{1/2}$. This provides similar intuition to Laplacian message passing, but it is not algebraically equivalent to applying a graph Laplacian to $\log(\mathbf{P}')$ in general, because the tangent space depends on the reference node and the update is performed through $\exp(\cdot)$.

Our aggregation is reference-centered: the tangent space depends on the receiving node c through $\mathbf{P}_c'^{-1/2}$. Hence the layer is not algebraically equivalent to applying a Euclidean GCN to $\log(\mathbf{P}')$; it performs local linearization (log), weighted averaging in that local tangent space (attention), and retraction back to $\mathcal{S}_{++}^d$ (exp).

We use a synchronous update (all $\mathbf{P}_c^{\mathrm{out}}$ are computed from the same input snapshot $\{\mathbf{P}_c^{\mathrm{in}}\}$), making the layer invariant to node iteration order.

Lemma 4 (SPD Preservation and Fixed Points): For every node c, the update (43) satisfies $\mathbf{P}_c' \in \mathcal{S}_{++}^d$. Moreover, if $\mathbf{P}_j' = \mathbf{P}_c'$ for all $j \in \mathcal{N}_k(c)$, then $\mathbf{P}_c'$ is a fixed point of the layer.

Proof: The matrix exponential maps any real-symmetric matrix to $\mathcal{S}_{++}^d$. Congruence with $\mathbf{P}_c'^{1/2}$ preserves positive-definiteness, proving the first claim. If $\mathbf{P}_j' = \mathbf{P}_c'$ for all $j \in \mathcal{N}_k(c)$, then each $\mathbf{L}_j^{(c)} = \mathbf{0}$ and consequently $\mathbf{L}_c^{\mathrm{agg}} = \mathbf{0}$; the exponential of the zero matrix is the identity, so (43) returns the original $\mathbf{P}_c$. ∎

VII. Experiments

We benchmark the full geometry-aware pipeline on three widely used motor-imagery datasets—(i) the PhysioNet MI collection, (ii) BNCI 2014-002, and (iii) BCI-ERN. For every dataset we apply a uniform pre-processing protocol: raw streams are detrended, resampled (or up-sampled) to 250 Hz, and band-pass filtered with a fourth-order zero-phase Butterworth filter (–3 dB at 5 and 45 Hz), which preserves μ (8)–13 Hz) and β (14)–30 Hz) rhythms while attenuating high-frequency electromyographic artifacts. Signals are then re-referenced to the common average and segmented into non-overlapping 1-s epochs that serve as network inputs. The same architectural hyper-parameters, Stiefel-constrained learning rates, and Riemannian optimization settings (Sect. VII-C) are held fixed across all three datasets, so that performance differences mainly reflect the robustness of the proposed manifold-aware framework rather than dataset-specific tuning.

All reimplemented methods share the same preprocessing pipeline and cross-subject splits (Sec. II-A, Sec. VII). Covariance-based Riemannian baselines, we compute trial covariances using the same regularization ϵ and apply the corresponding tangent-space (matrix-logarithm) mapping used by each method. Baselines that require an explicit scalp graph, we construct a common k-NN graph (k=4) from the same electrode coordinates and use it consistently across those models.

A. Scalp Graph k-NN Construction

Each channel's 3-D position is obtained from the standard 10–10/10–20 template distributed with MNE-Python. Let $\mathbf{p}_c \in \mathbb{R}^3$ be the Cartesian coordinate of electrode c. A **Euclidean** distance metric $d(c,j) = \|\mathbf{p}_c - \mathbf{p}_j\|_2$ is used to build a k-nearest-neighbor graph with $k = 4$. Ties are broken by choosing the physically closest electrode in radial (spherical) angle. Preliminary tests with great-circle (geodesic) distance on the best-fitting sphere yielded identical neighbor sets for $k \le 6$, so the simpler Euclidean metric is retained. The

graph is symmetrized and augmented with self-loops before normalization (cf. Eq. (39)).

B. Datasets

We evaluate the proposed geometry-consistent framework on three public motor-imagery / error-related potential dataset. All signals are re-referenced to the common average, artefact epochs (variance $> 3\sigma$) are removed, and 3-D electrode positions are re-constructed with MNE-PYTHON from the international 10–10/10–20 templates. Across all dataset the percentage of discarded artefact epochs never exceeds 2.5%.

1) PhysioNet Motor-Imagery Dataset (MI_1): 64–channel EEG from 109 volunteers (BCI2000, 160 Hz). Each trial presents a left/right arrow; participants kinesthetically imagine opening the corresponding fist for ≈ 2.5 s, then relax. Following [15], subjects #88, 89, 92, and 100 are excluded due to drop-outs, yielding 105 subjects. Trials last 3.1 s and, after preprocessing, are cropped to 3.0 s and segmented into three non-overlapping 1 s epochs. Evaluation uses a 10×10 cross-subject scheme: in each of 10 repeats, 10 randomly selected subjects form the test set and the remaining 95 the training set; results are averaged over 100 runs.

2) BNCI 2014-002 (MI_2): Fifteen-channel EEG from 14 subjects performing left/right motor imagery (80 trials per class), recorded at 512 Hz and downsampled to 250 Hz; trials are 4 s. We use a leave-one-subject-out (LOSO) protocol: each subject serves once as test, with the remaining 13 for training.

3) BCI-ERN (MI_3): From the Berlin BCI competition: 56-channel EEG at 600 Hz during a row/column P300 spelling task that elicits ERN responses. Each subject has five sessions (four$\times$60 trials and one of 100). Following the official release, we use the 16 publicly available subjects.

Across all dataset electrodes are re-referenced to the common average before filtering, and artifact-contaminated epochs (variance $> 3\sigma$ above subject mean) are discarded; fewer than 2.5% of epochs are removed in any dataset.

C. Riemannian Optimization and Hyper-Parameters

Training uses a hybrid optimizer:

- **Euclidean blocks**–convolutional kernels, batch-norm parameters and the classifier matrix $\mathbf{V}$ are trained with Adam.
- **Manifold block**–the projection matrix $\mathbf{W} \in \mathcal{S}t(d, D)$ is updated with Riemannian SGD (R-SGD) on the Stiefel manifold.

Tables II–III list the hyper-parameters; they are **identical across all datasets**. Values correspond to the best settings identified in the sensitivity analysis of Fig. 1 (rank d=48, k-NN size k=4, R-SGD base step 0.02).

At each iteration Adam updates the Euclidean blocks, while the Stiefel gradient is projected onto $T_{\mathbf{W}}\mathcal{S}t(d, D)$, scaled by the R-SGD step, and retracted via Thin QR (Alg. 1). A single hyper-parameter set therefore trains all three dataset; only the channel count C changes, reproducing the cross-subject accuracies of Table IV.

TABLE I
INITIALIZATION, NUMERICAL SAFEGUARDS, AND PRECISION POLICY

Item	Setting / Value
Stiefel matrix $\mathbf{W}$	$\mathcal{N}(0,1)$ then $\mathrm{qr}(\cdot)$
CNN weights $(\mathbf{W}_1, \mathbf{W}_2)$	He init (ReLU)
Classifier $\mathbf{V}$	Uniform $[-\sqrt{6/m}, \sqrt{6/m}]$, $m = d(d+1)/2$
Covariance regularizer ϵ	10^{-5} in Eq. (7)
Eigen clamp ϵ_{eig}	10^{-12} before log
QR retraction	Thin QR
Mixed precision	FP16 (convs); FP32 (covariance,)

TABLE II
GLOBAL OPTIMIZER SETTINGS AND FIXED ARCHITECTURAL HYPER-PARAMETERS

Global optimizer settings	
Item	**Value**
Mini-batch size	16
Epochs	250
Adam LR / weight decay	10^{-3} / 10^{-4}
Adam schedule	$\eta_t = \eta_0/(1 + 10^{-4}t)$
R-SGD base step ($\mathbf{W}$)	2×10^{-2} (same decay)
Gradient clip (Euclidean)	global-norm 5
Covariance regularizer ϵ	10^{-5}
Eigen clamp ϵ_{eig}	10^{-12}
Stiefel retraction	Thin QR
Mixed precision	FP16 (CNN) / FP32 (SPD ops)
Fixed architectural hyper-parameters	
Component	**Setting**
Channels C	64 / 15 / 56 (MI_1–$_3$)
Temporal window T	1 s @ 250 Hz
Depth-wise kernel K_1	1
Temporal kernel K_2	5 (padding 2)
Hidden channels D	64
Stiefel rank d	48
Graph k-NN (k)	4
Graph–SPD layers	1 (attention)
Post-graph dropout	p=0.25

TABLE III
PER–BLOCK OPTIMIZER CONFIGURATION

Block	Optimizer	Key settings
Conv $(\mathbf{W}_1, \mathbf{W}_2)$	Adam	$\eta_0 = 10^{-3}$, $\beta = (0.9, 0.999)$
Batch-norm (γ, β)	Adam	inherits LR from convs
Classifier $\mathbf{V}$	Adam	inherits LR from convs
Stiefel $\mathbf{W}$	R-SGD	$\eta_0 = 2 \times 10^{-2}$, proj. + QR

VIII. RESULTS AND DISCUSSION

A. Benchmark Results

Table IV consolidates the cross-subject accuracies and macro-F_1 scores on the three dataset. The results reveal three notable patterns.

1) Consistent Gains Across Paradigms: The proposed model delivers the highest mean accuracy and F_1 on **all** datasets, improving upon the best prior work by + 1.2% (MI_1), + 1.7% (MI_2) and + 2.7% (MI_3) in F_1. Table VI shows that balanced accuracy closely tracks overall accuracy on all three datasets, indicating that gains are not driven by

TABLE IV
CROSS-SUBJECT MEAN ACCURACY (%) AND MACRO-F_1 ON THE THREE DATASETS

Model	MI_1		MI_2		MI_3	
	Acc.	F_1	Acc.	F_1	Acc.	F_1
DeepConvNet [15]	69.3±5.4	0.68±0.05	68.5±5.8	0.69±0.06	63.6±4.2	0.64±0.04
ShallowConvNet [15]	64.6±8.2	0.64±0.08	62.8±4.5	0.63±0.05	63.7±4.9	0.64±0.05
EEGNet [14]	72.2±4.5	0.73±0.07	65.7±4.9	0.66±0.04	64.4±5.1	0.64±0.05
G-CRAM [30]	79.1±6.8	0.79±0.06	73.2±7.4	0.73±0.07	71.8±7.0	0.72±0.07
TSA [24]	64.2±2.7	0.65±0.03	76.5±2.9	0.76±0.03	72.8±3.6	0.72±0.04
SSDA [26]	82.0±7.2	0.82±0.06	71.3±7.9	0.71±0.07	69.7±5.6	0.70±0.06
RPA [23]	63.2±3.5	0.64±0.03	72.5±3.2	0.72±0.04	74.8±4.2	0.75±0.04
Graph-CSPNet [27]	81.4±4.8	0.81±0.07	79.8±7.0	0.80±0.08	76.0±7.9	0.74±0.08
TSMNet [25]	65.9±8.0	0.66±0.07	79.4±4.2	0.79±0.05	68.0±6.8	0.68±0.05
MAtt [28]	81.2±3.5	0.82±0.03	79.5±3.2	0.79±0.04	74.8±4.2	0.74±0.04
GDLNet [29]	74.8±4.7	0.73±0.07	78.9±7.1	0.79±0.04	77.8±4.3	0.76±0.07
Proposed	**83.2±3.5**	**0.83±0.04**	**81.5±2.3**	**0.82±0.03**	**79.7±3.2**	**0.78±0.03**

TABLE V
ABLATION VARIANTS WITH CROSS-SUBJECT PERFORMANCE AND MATCHED ONE-SIDED WILCOXON SIGNED-RANK TESTS VS. THE FULL FRAMEWORK. BEST SCORES PER DATASET IN BOLD. ALL p-VALUES SURVIVE BONFERRONI CORRECTION (α_{CORR} = 0.0125)

Variant	MI_1				MI_2				MI_3			
	Acc.	F_1	p	r	Acc.	F_1	p	r	Acc.	F_1	p	r
Full framework (ours)	**83.2±3.5**	**0.83±0.04**	—	—	**81.5±2.3**	**0.82±0.03**	—	—	**79.7±3.2**	**0.78±0.03**	—	—
w/o Graph–SPD layer	71.4±4.2	0.71±0.05	1.8×10^{-3}	0.76	70.0±3.6	0.70±0.04	2.4×10^{-3}	0.72	69.5±3.4	0.69±0.04	3.1×10^{-3}	0.70
w/o Orthonormal projection	68.9±4.5	0.69±0.05	7.9×10^{-4}	0.82	68.1±3.9	0.68±0.04	1.1×10^{-3}	0.79	67.8±3.8	0.68±0.04	1.5×10^{-3}	0.78
w/o Covariance regularization	65.3±4.8	0.65±0.05	4.6×10^{-4}	0.88	65.9±4.1	0.66±0.05	6.8×10^{-4}	0.85	65.4±3.9	0.65±0.05	9.2×10^{-4}	0.81
Euclidean CNN baseline[†]	35.7±5.1	0.36±0.05	$<10^{-6}$	0.97	34.9±4.8	0.35±0.05	$<10^{-6}$	0.96	35.3±4.6	0.35±0.05	$<10^{-6}$	0.95

[†] Two-layer CNN front-end followed by global average pooling and a linear classifier; no covariance, Stiefel, or graph processing.

TABLE VI
THE INDICATORS FOR THE PROPOSED MODEL ONLY UNDER IDENTICAL PREPROCESSING/SPLITS. MACRO-AUROC IS ONE-VS-REST AVERAGED; ECE USES 15 EQUAL-MASS BINS

Metric	MI_1	MI_2	MI_3
Balanced Accuracy (%)	83.0	81.4	79.6
Cohen's κ	0.66	0.63	0.59
Matthews Corr. (MCC)	0.66	0.64	0.60
Macro-AUROC	0.91	0.90	0.88
ECE (15 bins)	0.030	0.034	0.039

class imbalance. Cohen's κ and MCC (both $\approx$ 0.6) reflect substantial agreement beyond chance, consistent with the macro-F_1 improvements. Macro-AUROC $\approx$ 0.90 indicates strong separability, while low ECE ($\leq$ 0.04) suggests well-calibrated probabilities without explicit temperature scaling. Together, these metrics corroborate that the improvements are not only in accuracy but also in discriminability and calibration.

Crucially, the dataset differ not only in montage size (64/15/56 channels) but also in task (sensorimotor rhythm vs. ERN), supporting the claim that the geometry-aware design generalizes beyond any single paradigm.

2) Geometry Beats Euclidean Learning: Averaged over the three dataset, pure Euclidean CNNs (DeepConvNet, ShallowConvNet, EEGNet) trail the proposed framework by $\approx$ 15–18%. Geometry-aware competitors that lack a graph formulation (TSA, RPA) gain 7–10 %but remain 8–9 %below our model. Methods that combine graph structure with SPD processing (Graph-CSPNet, MAtt, GDLNet) close the gap yet are still outperformed by 1–4%, indicating that Stiefel-constrained projection and log-Euclidean consistency are decisive.

3) Class–Balanced Improvements: The macro-F_1 advantages echo the accuracy gains, confirming that the model does not simply favour the majority class (particularly important for the imbalanced ERN dataset). On MI_3 the framework adds + 4.0**%** macro-F_1 over the best baseline.

B. Statistical Significance and Effect Size

Simply reporting mean scores is insufficient to judge whether observed gains are due to sampling noise or reflect a genuine population-level advantage. We therefore conduct a rigorous within–subject non-parametric significance analysis:

1) Test Procedure: For every ablated variant and for every fold we compute the signed difference in accuracy with respect to the full framework. The one-sided Wilcoxon signed–rank statistic W is then evaluated under the null hypothesis $H_0 : \Delta \leq 0$ (i.e., the ablation does not degrade performance). Because four null hypotheses are tested per dataset, the Bonferroni-corrected threshold is $\alpha_{corr} = 0.05/4 = 0.0125$.

2) Effect-Size r: To complement p values we report the matched–pairs rank-biserial correlation $r = \frac{|W^+ - W^-|}{n(n+1)/2}$, where W^+ and W^- are the sums of positive and negative ranks, respectively, and n is the number of folds. Setting $|r| \geq 0.5$ is interpreted as a large effect. In addition to p-values, we report the matched–pairs rank-biserial effect size $r = \frac{|W^+ - W^-|}{n(n+1)/2}$, which

was large across datasets ($|r| \geq 0.70$), indicating substantial practical significance.

3) Interpretation: All p values fall well below α_{corr}, allowing us to reject H_0 in every case. Moreover, rank-biserial correlations are consistently $|r| \geq 0.70$—large by conventional benchmarks—demonstrating that the performance lift is not only statistically reliable but also of substantial practical magnitude. In particular, the largest effect is attributable to covariance regularization, highlighting its pivotal role in stabilizing log-Euclidean mapping, while the Graph-SPD layer yields the most dataset-specific gains (highest r on MI_1, lowest on MI_3), mirroring the spatial density of the underlying montages.

4) Robustness to Fold Splits: The Wilcoxon analysis is conducted on the exact fold-wise accuracies, hence captures variation due to subject splits, channel artifacts, and class imbalance. The full model dominates in at least 90% of folds across all datasets, underscoring its reliability for unseen subjects—a critical property for real-world BCI deployment.

C. Ablation Study

Table V retrains four reduced variants to isolate the effect of each module.

1) **Graph-SPD layer:** Removing tangent-space aggregation (w/o Graph-SPD) lowers accuracy by 11–13%. Visualization of attention weights confirms that the full model focuses on the contralateral motor strip (C3/C4), whereas the ablated variant distributes attention uniformly, providing a mechanistic explanation for the drop.
2) **orthonormal projection:** Eliminating the orthonormal projection (w/o Orthonormal proj) degrades accuracy by a further 13–15%. During training the minimum eigenvalue of several covariances approaches zero, causing occasional gradient explosions—exactly the pathology predicted by the robustness bound in Prop. 1.
3) **Covariance regularization:** Setting ϵ=0 reduces performance by 3–4 %and increases NaN frequency, corroborating the theoretical need for a positive eigenvalue floor.
4) **Euclidean CNN baseline:** A two-layer CNN with global average pooling achieves only 35–36 %, marginally above chance. Hence the dominant performance gains are not due to the shallow CNN front-end but to the geometric processing pipeline.

D. Measured CPU Latency

We report measured CPU inference latency (ms/trial) for the complete forward pipeline of each manifold-aware method under a fixed reference setting (64 channels, 1 s @ 250 Hz, 4-class output). All timings are obtained on a 12th Gen Intel(R) Core(TM) i7-12700H (2.30 GHz) using a single CPU thread, and they include all algorithmic stages executed at inference (feature extraction, covariance construction, manifold operators such as eigendecomposition/square-root and log / exp where applicable, graph aggregation where present, and final classification), rather than timing isolated SPD blocks. Under this protocol, Riemannian/geometry-based baselines exhibit substantially higher latency due to repeated SPD matrix functions: SSDA runs at 39.49 ms/trial, TSMNet at 42.48 ms/trial, and manifold–graph models such as GDLNet and MAtt reach 38.22 ms/trial and 52.24 ms/trial, respectively; Graph-CSPNet measures 44.48 ms/trial. Our method reduces this overhead and achieves 31.23 ms/trial while maintaining end-to-end geometric consistency, because the Stiefel orthonormal projection reduces the SPD dimension before applying log / exp in the downstream manifold/graph operations, making the dominant matrix-function cost cubic in a smaller rank d rather than in the original covariance size.

E. Discussion

Second-order channel interactions embody the physiology of MI and ERN. Encoding them as SPD covariances and respecting their intrinsic (log-Euclidean) geometry yields measurable gains in both accuracy and stability. Theoretical results (Thm. 2 and Prop. 1) predict a smaller generalization gap and a bounded Lipschitz response; the 11–18% margin over Euclidean CNNs confirms these predictions empirically.

The tangent-space graph layer improves interpretability by aggregating only spatially plausible neighbors. Attention maps peak on C3/C4 for MI tasks and on FCz for ERN, aligning with established neurophysiology. Removing the layer collapses these patterns and reduces accuracy by double digits.

The orthonormal projection curbs rank-deficient covariances while reducing dimensionality from D=64 to d=48. Ablation reveals that orthonormality alone raises accuracy by 15%relative to a projection with unconstrained weights, validating the role of retraction-based R-SGD.

Despite its geometric sophistication, the model remains compact (0.92M parameters), comparable to GDLNet (0.90M) and below Graph-CSPNet (1.2M), while achieving the best cross-subject accuracy.

A full factorial study over SNR and session shift is beyond our page and compute budget; moreover, two of our datasets lack session annotations. Our theory (Sec. V) bounds sensitivity via the eigenvalue floor, and cross-dataset performance with well-calibrated outputs (Table VI) supports practical robustness. We will release code to reproduce standard SNR sweeps.

We deliberately used a single 5–45 Hz band so that the temporal convolutions learn task-relevant sub-bands (acting as adaptive band-pass filters) while keeping the SPD rank small for stable log/exp. Multi-band SPD fusion is orthogonal to our contribution and would primarily increase d and cubic costs; we therefore list it as future work.

Geometry-aware priors (SPD fidelity, orthonormal congruence) act as structural regularizers that reduce data demands: covariances remain well-conditioned and comparable across subjects, which helps in low-trial regimes. The calibrated outputs (ECE ≤ 0.04) are useful for clinical decision thresholds. A limitation is adversarial robustness: our Lipschitz analysis bounds random noise sensitivity, not worst-case perturbations; evaluating adversarial resilience remains future work.

F. Limitations

(i) The log-Euclidean metric is only orthonormally, not affinely, invariant. Extending to the full affine-invariant metric could sharpen class boundaries at higher computational cost. (ii) Transferability to non sensorimotor tasks (e. g., P300, speech imagery) remains to be verified; the hyper-parameter insensitivity observed here is encouraging. (iii) While the robustness bound covers white noise perturbations, adversarial robustness is an open question.

IX. Conclusion

Contributions: A geometry-consistent decoder coupling lightweight CNNs with Stiefel congruence and node-wise Graph–SPD aggregation.

Results: Consistent gains in accuracy, macro-F_1, balanced accuracy, κ, MCC, and calibration across three dataset.

Impact: End-to-end SPD fidelity with scalp topology yields effectiveness and physiological interpretability.

Limitations: Log–Euclidean is not affine-invariant; focus on MI/ERN paradigms; adversarial robustness not assessed.

Future work: Efficient affine-invariant training; multi-band SPD blocks; session/domain adaptation; adversarial evaluations.

References

[1] W. Jin et al., "Electroencephalogram-based adaptive closed-loop brain–computer interface in neurorehabilitation: A review," *Frontiers Comput. Neurosci.*, vol. 18, Sep. 2024, Art. no. 1431815.

[2] S. Russo, S. Ahmed, I. E. Tibermacine, and C. Napoli, "Enhancing EEG signal reconstruction in cross-domain adaptation using CycleGAN," in *Proc. Int. Conf. Telecommun. Intell. Syst. (ICTIS)*, Dec. 2024, pp. 1–8.

[3] R. Ishii, "Editorial: The application of EEG in neurorehabilitation," *Brain Sci.*, vol. 15, no. 8, p. 856, 2025, doi: 10.3390/brainsci15080856.

[4] K. K. Ang, Z. Y. Chin, C. Wang, C. Guan, and H. Zhang, "Filter bank common spatial pattern algorithm on BCI competition IV datasets 2a and 2b," *Frontiers Neurosci.*, vol. 6, p. 39, Jan. 2012.

[5] C. Gao, W. Liu, and X. Yang, "Convolutional neural network and Riemannian geometry hybrid approach for motor imagery classification," *Neurocomputing*, vol. 507, pp. 180–190, Oct. 2022.

[6] N. Boutarfaia, S. Russo, A. Tibermacine, and I. E. Tibermacine, "Deep learning for EEG-based motor imagery classification: Towards enhanced human-machine interaction and assistive robotics," *Life*, vol. 2, no. 3, p. 4, 2023.

[7] I. E. Tibermacine, A. Tibermacine, W. Guettala, C. Napoli, and S. Russo, "Enhancing sentiment analysis on SEED-IV dataset with vision transformers: A comparative study," in *Proc. 11th Int. Conf. Inf. Technol., IoT Smart City*, Dec. 2023, pp. 238–246.

[8] S. Russo et al., "Analyzing EEG patterns in young adults exposed to different acrophobia levels: A VR study," *Frontiers Human Neurosci.*, vol. 18, May 2024, Art. no. 1348154.

[9] M. Congedo, A. Barachant, and R. Bhatia, "Riemannian geometry for EEG-based brain–computer interfaces; a primer and a review," *Brain-Comput. Interfaces*, vol. 4, no. 3, pp. 155–174, Jul. 2017.

[10] Z. Huang and L. V. Gool, "A Riemannian network for SPD matrix learning," in *Proc. AAAI Conf. Artif. Intell.*, 2017, vol. 31, no. 1. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/10866

[11] S. Zhang, G. Mordant, T. Matsumoto, and G. Schiebinger, "Manifold learning with sparse regularised optimal transport," 2023, *arXiv:2307.09816*.

[12] J. Ren and X.-J. Wu, "Probability distribution-based dimensionality reduction on Riemannian manifold of SPD matrices," *IEEE Access*, vol. 8, pp. 153881–153890, 2020.

[13] T. Kojima, K. Arikawa, S. Koyama, and H. Saruwatari, "Multichannel active noise control with exterior radiation suppression based on Riemannian optimization," in *Proc. 31st Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2023, pp. 96–100.

[14] 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," *J. Neural Eng.*, vol. 15, no. 5, Oct. 2018, Art. no. 056013.

[15] R. T. Schirrmeister et al., "Deep learning with convolutional neural networks for EEG decoding and visualization," *Human Brain Mapping*, vol. 38, no. 11, pp. 5391–5420, Nov. 2017.

[16] Y. Li, L. Guo, Y. Liu, J. Liu, and F. Meng, "A temporal-spectral-based squeeze-and- excitation feature fusion network for motor imagery EEG decoding," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1534–1545, 2021.

[17] H. Chen, L. Liu, Z. Chen, and X. Li, "Time-aware squeeze-excitation transformer for sequential recommendation," in *Proc. Int. Conf. Artif. Neural Netw*. Cham, Switzerland: Springer, 2024, pp. 121–135.

[18] K. E. Ch Vidyasagar, K. Revanth Kumar, G. N. K. Anantha Sai, M. Ruchita, and M. J. Saikia, "Signal to image conversion and convolutional neural networks for physiological signal processing: A review," *IEEE Access*, vol. 12, pp. 66726–66764, 2024.

[19] I. E. Tibermacine et al., "Adversarial denoising of EEG signals: A comparative analysis of standard GAN and WGAN-GP approaches," *Frontiers Human Neurosci.*, vol. 19, May 2025, Art. no. 1583342.

[20] I. Naidji, A. Tibermacine, I. E. Tibermacine, S. Russo, and C. Napoli, "EGDN-KL: Dynamic graph–deviation network for EEG anomaly detection," *Biomed. Signal Process. Control*, vol. 112, Feb. 2026, Art. no. 108597.

[21] D. Brooks, O. Schwander, F. Barbaresco, J.-Y. Schneider, and M. Cord, "Riemannian batch normalization for SPD neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2019/hash/6e69ebbfad976d4637bb4b39de261bf7-Abstract.html

[22] D. Nguyen, "Geometry in global coordinates in mechanics and optimal transport," 2023, *arXiv:2307.10017*.

[23] P. L. C. Rodrigues, C. Jutten, and M. Congedo, "Riemannian Procrustes analysis: Transfer learning for brain–computer interfaces," *IEEE Trans. Biomed. Eng.*, vol. 66, no. 8, pp. 2390–2401, Aug. 2019.

[24] A. Bleuzé, J. Mattout, and M. Congedo, "Tangent space alignment: Transfer learning for brain–computer interface," *Frontiers Human Neurosci.*, vol. 16, Dec. 2022, Art. no. 1049985.

[25] R. J. Kobler, J.-i. Hirayama, Q. Zhao, and M. Kawanabe, "SPD domain-specific batch normalization to crack interpretable unsupervised domain adaptation in EEG," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 6219–6235.

[26] S. Sartipi and M. Cetin, "Subject-independent deep architecture for EEG-based motor imagery classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 32, pp. 718–727, 2024.

[27] C. Ju and C. Guan, "Graph neural networks on SPD manifolds for motor imagery classification: A perspective from the time–frequency analysis," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 12, pp. 17701–17715, Dec. 2024.

[28] Y. Pan, J.-L. Chou, and C.-S. Wei, "MAtt: A manifold attention network for EEG decoding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 31116–31129.

[29] R. Wang, C. Hu, Z. Chen, X.-J. Wu, and X. Song, "A Grassmannian manifold self-attention network for signal classification," in *Proc. 33rd Int. Joint Conf. Artif. Intell.*, 2024, pp. 5099–5107.

[30] Y. Song, Q. Zheng, B. Liu, and X. Gao, "EEG conformer: Convolutional transformer for EEG decoding and visualization," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 710–719, 2023.

[31] S. Wang et al., "Graph machine learning in the era of large language models (LLMs)," *ACM Trans. Intell. Syst. Technol.*, vol. 16, no. 5, pp. 1–40, Oct. 2025.

[32] R. Wang, X.-J. Wu, Z. Chen, C. Hu, and J. Kittler, "SPD manifold deep metric learning for image set classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 7, pp. 8924–8938, Jul. 2024.

[33] I. E. Tibermacine et al., "Riemannian geometry-based EEG approaches: A literature review," 2024, *arXiv:2407.20250.*

[34] Z. Shuqfa, A. N. Belkacem, and A. Lakas, "Decoding multi-class motor imagery and motor execution tasks using Riemannian geometry algorithms on large EEG datasets," *Sensors*, vol. 23, no. 11, p. 5051, May 2023.

[35] A. S. M. Miah, M. A. Rahim, and J. Shin, "Motor-imagery classification using Riemannian geometry with median absolute deviation," *Electronics*, vol. 9, no. 10, p. 1584, Sep. 2020.

[36] K. Sadatnejad, S. S. Ghidary, R. Rostami, and R. Kazemi, "EEG representation using multi-instance framework on the manifold of symmetric positive definite matrices for EEG-based computer aided diagnosis," 2017, *arXiv:1702.02655*.

[37] A. Hippert-Ferrer, A. Mian, F. Bouchard, and F. Pascal, "Riemannian classification of EEG signals with missing values," in *Proc. 30th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2022, pp. 842–846.

[38] J. Jin, T. Qu, R. Xu, X. Wang, and A. Cichocki, "Motor imagery EEG classification based on Riemannian sparse optimization and Dempster–Shafer fusion of multi-time-frequency patterns," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 58–67, 2023.

[39] G. Zhang and A. Etemad, "Spatio-temporal EEG representation learning on Riemannian manifold and Euclidean space," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 8, no. 2, pp. 1469–1483, Apr. 2024.

[40] S. Guan, K. Zhao, and S. Yang, "Motor imagery EEG classification based on decision tree framework and Riemannian geometry," *Comput. Intell. Neurosci.*, vol. 2019, pp. 1–13, Jan. 2019.

[41] X. Navarro-Sune et al., "Riemannian geometry applied to detection of respiratory states from EEG signals: The basis for a brain–ventilator interface," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 5, pp. 1138–1148, May 2017.

[42] C. H. Nguyen, G. K. Karavas, and P. Artemiadis, "Inferring imagined speech using EEG signals: A new approach using Riemannian manifold features," *J. Neural Eng.*, vol. 15, no. 1, Feb. 2018, Art. no. 016002.

[43] X. Kan et al., "R-mixup: Riemannian mixup for biological networks," in *Proc. 29th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2023, pp. 1073–1085.

[44] L. Pan et al., "Riemannian geometric and ensemble learning for decoding cross-session motor imagery electroencephalography signals," *J. Neural Eng.*, vol. 20, no. 6, Dec. 2023, Art. no. 066011.

[45] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. Princeton, NJ, USA: Princeton Univ. Press, 2009.

[46] J. R. Magnus and H. Neudecker, *Matrix Differential Calculus With Applications to Statistics and Econometrics*. Hoboken, NJ, USA: Wiley, 2019.

[47] N. Boumal, *An Introduction to Optimization on Smooth Manifolds*. Cambridge, U.K.: Cambridge Univ. Press, 2020.

[48] R. Vershynin, *High-dimensional Probability: An Introduction With Applications in Data Science*, vol. 47. Cambridge, U.K.: Cambridge Univ. Press, 2018.

[49] N. Golowich, A. Rakhlin, and O. Shamir, "Size-independent sample complexity of neural networks," in *Proc. Conf. Learn. Theory*, 2018, pp. 297–299.

[50] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2014.

[51] Y. Lee and S. Choi, "Gradient-based meta-learning with learned layerwise metric and subspace," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 2927–2936.

[52] O. Catoni, "PAC-Bayesian supervised classification: The thermodynamics of statistical learning," 2007, *arXiv:0712.0248*.

[53] R. Bhatia, *Positive Definite Matrices*. Princeton, NJ, USA: Princeton Univ. Press, 2009.

[54] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.