Building Transformation Layers for Riemannian Neural Networks

Ziheng Chen
University of Trento; MPI for Intelligent Systems, Tübingen
ziheng_ch@163.com

Abstract

Recently, deep neural networks on manifold-valued representations have garnered significant attention across various machine learning applications. One recent focus is the generalization of Euclidean fully connected (FC) and convolutional layers to non-Euclidean geometries. However, previous approaches typically focus on a few selected manifolds and rely on specific properties of the target manifold. In contrast, this work proposes a framework for constructing FC and convolutional layers over computationally tractable Riemannian spaces. This framework incorporates several previous FC layers across different geometries as special cases and is instantiated on ten representative manifolds, including three hyperbolic models, five geometries of the symmetric positive definite (SPD) manifold, and two Grassmannian perspectives. Experiments on different manifolds demonstrate the effectiveness and applicability of our approach. Code can be found at https://github.com/GitZH-Chen/RieTrans.

1 Introduction

Deep neural networks on Riemannian manifolds have achieved remarkable success across various applications [34, 28, 67, 47, 33, 77, 13, 39, 61, 44, 45, 32, 37, 21, 22]. Commonly encountered manifolds include Symmetric Positive Definite (SPD) [59], Grassmannian [5], matrix Lie group [31], and hyperbolic [72] manifolds. These manifolds admit *computationally tractable tools* such as geodesics, exponential and logarithmic maps, and parallel transport, which have enabled the extension of fundamental deep learning components, including normalization [8, 10, 48, 41, 16, 19, 79], attention [30, 58, 76, 78], residual blocks [74, 38], and classification layers [28, 55, 15, 17, 4].

Yet, the extension of the most basic layers, Fully Connected (FC) and convolutional layers, remains particularly challenging. Early works targeted specific manifolds: Huang and Van Gool [34], Huang et al. [35, 36] proposed layers for SPD, special orthogonal, and Grassmann manifolds. Later, Ganea et al. [28], Mao et al. [51] developed hyperbolic counterparts via tangent spaces, and Chen et al. [14] introduced LFC via spacetime transformations. Fan et al. [26] further introduced nested hyperbolic spaces and a Lorentz-specific feature transformation that supports dimensionality reduction. To better respect geometry, Shimizu et al. [66] extended FC and convolutional layers on the Poincaré model, while Nguyen et al. [56, 57] proposed SPD counterparts based on gyrovector structures and symmetric spaces. However, these methods largely rely on specific properties, such as Poincaré geometry, gyro or symmetric structures, which restricts their generality. In another direction, Chakraborty et al. [11] introduced a convolution based on the weighted Fréchet mean, but unlike the Euclidean convolution, its output manifold dimension is restricted to match the input, limiting flexibility. Consequently, a general and flexible framework for constructing FC and convolutional layers across different geometries remains unsolved.

We address this challenge by proposing a principled framework for building Riemannian FC and convolutional layers on computationally tractable manifolds. Our framework relies solely on Riemannian

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

operators such as exponential and logarithmic maps. It thus applies broadly to different geometries, such as hyperbolic, SPD, and Grassmannian spaces. Our contributions are summarized as follows.

- **Riemannian FC and convolutional layers.** We introduce a principled generalization of FC and convolutional layers to Riemannian spaces. In contrast to previous approaches, our framework requires only tractable Riemannian operators, ensuring broad applicability. Moreover, several existing Riemannian FC layers are subsumed as special cases.
- **Ten concrete instantiations.** We instantiate our framework on three hyperbolic models, five SPD geometries, and two Grassmannian perspectives. Our approach enables direct variation of the latent geometry under a consistent network architecture.
- **Empirical validation.** We validate our approach on benchmark tasks across hyperbolic, SPD, and Grassmannian manifolds, demonstrating both effectiveness and versatility.

2 Preliminaries

Notations. For the Euclidean space $\mathbb{R}^n$ or $\mathbb{R}^{n\times n}$, we denote the standard inner product by $\langle\cdot,\cdot\rangle$ and the induced norm by $\|\cdot\|$, *i.e.*, the L_2-norm for vectors and the Frobenius norm for matrices. A Riemannian manifold $(\mathcal{M}, g)$ with the Riemannian metric g is abbreviated as $\mathcal{M}$. Its tangent space at $P \in \mathcal{M}$ is denoted by $T_P\mathcal{M}$. The Riemannian logarithmic map, exponential map, and metric at $P \in \mathcal{M}$ are denoted by Log_P, Exp_P, and $\langle\cdot,\cdot\rangle_P = g_P(\cdot,\cdot)$, respectively. The parallel transport along the geodesic connecting $P, Q \in \mathcal{M}$ is $\Gamma_{P\to Q}$. A table of notation is summarized in App. C.

Hyperbolic manifold. There are five isometric hyperbolic models [9]. We focus on the Poincaré ball $\mathbb{P}^n_K = \left\{x \in \mathbb{R}^n \mid \|x\|^2 < -1/K\right\}$, the Beltrami–Klein ball $\mathbb{K}^n_K = \left\{x \in \mathbb{R}^n \mid \|x\|^2 < -1/K\right\}$, and the hyperboloid (or Lorentz) model $\mathbb{H}^n_K = \left\{x \in \mathbb{R}^{n+1} \mid \|x\|^2_{\mathcal{L}} = 1/K, x_1 > 0\right\}$, where $\|x\|^2_{\mathcal{L}} = \sum_{i=2}^{n+1} x_i^2 - x_1^2$ is the Lorentz inner product. Here, $K < 0$ is the constant curvature. The Poincaré and Beltrami–Klein balls admit gyrovector spaces, known as the Möbius and Einstein gyrovector spaces [73], respectively. The Möbius gyroaddition and scalar gyromultiplication are denoted by $\oplus_{\mathrm{M}}$ and $\otimes_{\mathrm{M}}$, while the Einstein counterparts are $\oplus_{\mathrm{E}}$ and $\otimes_{\mathrm{E}}$. App. D.1 summarizes the associated Riemannian and gyro operators.

SPD manifold. The set of $n \times n$ SPD matrices, denoted $\mathcal{S}^n_{++}$, forms a smooth manifold, called the SPD manifold [2]. It admits five widely used Riemannian metrics: Affine-Invariant Metric (AIM) [59], Log-Euclidean Metric (LEM) [2], Power-Euclidean Metric (PEM) [24], Log-Cholesky Metric (LCM) [46], and Bures–Wasserstein Metric (BWM) [6]. Each metric provides closed-form expressions for Riemannian operators, which are summarized in App. D.2.

Grassmannian. The Grassmannian is the manifold of p-dimensional subspaces of the n-dimensional vector space [71, Prob. 7.8]. It has two common matrix representations [5]. The Projector Perspective (PP) embeds each element as an $n \times n$ symmetric matrix: $\widetilde{\mathrm{Gr}}(p, n) = \{P \in \mathcal{S}^n \mid P^2 = P, \mathrm{rank}(P) = p\}$, where $\mathcal{S}^n$ is the Euclidean space of symmetric matrices. The OrthoNormal Basis (ONB) perspective is the quotient of the Stiefel manifold $\mathrm{St}(p, n)$: $\mathrm{Gr}(p, n) = \mathrm{St}(p, n)/\mathrm{O}(p) = \{[U] \mid [U] = \{\widetilde{U} \in \mathrm{St}(p, n) \mid \widetilde{U} = UR, R \in \mathrm{O}(p)\}\}$, where $\mathrm{O}(p)$ is the $p \times p$ orthogonal group. By abuse of notation, we use $[U]$ and U interchangeably. The associated Riemannian structures are summarized in App. D.3.

The considered manifolds admit multiple geometries, including isometric hyperbolic models, distinct SPD metrics, and diffeomorphic Grassmannian perspectives, whose empirical performance often varies across tasks [54, 38, 20]. This motivates a unified framework to flexibly handle such variants. Moreover, exponential and logarithmic maps may encounter singular cases, but these cases can be handled numerically (Rmks. D.1 and D.2), and the maps are assumed well-defined.

3 Proposed Framework

3.1 Riemannian fully connected layers

Our method for building FC layers over Riemannian manifolds relies on the point-to-hyperplane distance, which has shown success in building hyperbolic and SPD networks [66, 55, 4, 15, 17, 56, 57].

The hyperplane in the Riemannian manifold $\mathcal{N}$ [17, Eq. 5] is $H_{A,P} = \{X \in \mathcal{N} \mid \langle \mathrm{Log}_P(X), A\rangle_P = 0\}$, with $P \in \mathcal{N}$ and $A \in T_P\mathcal{N}$. When $\mathcal{N} = \mathbb{R}^n$, it recovers the Euclidean hyperplane, $H_{a,p} = \{x \in \mathbb{R}^n \mid \langle a, x - p\rangle = 0\}$.

The Euclidean FC layer is defined as $y = Ax + b$ with $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$. It can be expressed element-wise as $y_k = \langle a_k, x\rangle - b_k = \langle a_k, x - p_k\rangle$ with $a_k, p_k \in \mathbb{R}^n$ and $\langle p_k, a_k\rangle = b_k$. As shown by Shimizu et al. [66, Sec. 3.2], the LHS y_k is the signed distance from y to the hyperplane passing through the origin and orthogonal to the k-th axis of the output space, which can be formulated as

$$\mathrm{sign}(\langle e_k, y - \mathbf{0}\rangle) d(y, H_{e_k,\mathbf{0}}) = \langle a_k, x - p_k\rangle, \quad \forall 1 \le k \le m, \tag{1}$$

where $\mathbf{0} \in \mathbb{R}^m$ is the zero vector, and $\{e_k\}_{k=1}^m$ forms an orthonormal basis over $\mathbb{R}^m$ with e_k denoting the vector whose k-th element is 1 and all others are 0. Here, the LHS of Eq. (1) equals y_k.

Given a point-to-hyperplane distance, Eq. (1) can be readily generalized to manifolds. Noting that $\mathrm{Log}_p(x) = x - p$ under the Euclidean geometry and that $T_{\mathbf{0}}\mathbb{R}^m \cong \mathbb{R}^m$, the counterparts of $H_{e_k,\mathbf{0}}$ on an m-dimensional Riemannian manifold $\mathcal{M}$ are defined as

$$H_{B_k,E} = \{S \in \mathcal{M} \mid \langle \mathrm{Log}_E S, B_k\rangle_E = 0\}, \quad \forall 1 \le k \le m, \tag{2}$$

where $E \in \mathcal{M}$ is the predefined origin, and $\{B_k\}_{k=1}^m$ is an orthogonal basis over $\{T_E\mathcal{M}, g_E\}$. Eq. (2) characterizes the hyperplane containing the origin E and orthogonal to the geodesic starting from E with the initial velocity B_k, which recovers the Euclidean $H_{e_k,\mathbf{0}}$ as $\mathcal{M} = \mathbb{R}^m$. With all the above ingredients, we define the Riemannian FC layer in the following.

Definition 3.1. Given an n-dimensional manifold $\mathcal{N}$ and an m-dimensional manifold $\mathcal{M}$, the Riemannian FC layer $\mathcal{F} : \mathcal{N} \to \mathcal{M}$ for the input $X \in \mathcal{N}$ returns the output $Y \in \mathcal{M}$ by solving m equations:

$$\mathrm{sign}\left(\left\langle \mathrm{Log}^{\mathcal{M}}_E(Y), B_k\right\rangle^{\mathcal{M}}_E\right) \mathrm{d}^{\mathcal{M}}(Y, H^{\mathcal{M}}_{B_k,E^{\mathcal{M}}}) = \langle A_k, \mathrm{Log}^{\mathcal{N}}_{P_k}(X)\rangle^{\mathcal{N}}_{P_k}, 1 \le k \le m, \tag{3}$$

where $E^{\mathcal{M}} \in \mathcal{M}$ is the origin, and $\{B_k\}_{k=1}^m$ is an orthonormal basis over $T_{E^{\mathcal{M}}}\mathcal{M}$. Here, $\mathrm{d}^{\mathcal{M}}$ denotes a chosen point-to-hyperplane distance, which can be either the true distance $\inf_{S\in H^{\mathcal{M}}_{B_k,E^{\mathcal{M}}}} d_{\mathcal{M}}(Y, S)$ or a surrogate. Meanwhile, $\mathrm{Log}^{\mathcal{N}}_{P_i}$ and $\langle\cdot,\cdot\rangle^{\mathcal{N}}_{P_i}$ are the Riemannian logarithm and metric over $\mathcal{N}$. The pairs $P_k \in \mathcal{N}$ and $A_k \in T_{P_k}\mathcal{N}$ are the FC parameters.

Our Def. 3.1 naturally extends previous FC layers over different geometries.

Proposition 3.2. [↓] *When $\mathcal{N} = \mathbb{R}^n$ and $\mathcal{M} = \mathbb{R}^m$, Def. 3.1 with the true point-to-hyperplane distance reduces to the Euclidean FC layer. When $\mathcal{N} = \mathbb{P}^n_K$, $\mathcal{M} = \mathbb{P}^m_K$, the true point-to-hyperplane distance follows Ganea et al. [28, Thm. 5], and the LHS of Eq.* (3) *follows $v_k(\cdot)$ from Shimizu et al. [66, Eq. (3)]. In this case, Def. 3.1 yields the Poincaré FC layer [66, Sec. 3.2]. When $\mathcal{N} = \mathcal{S}^n_{++}$, $\mathcal{M} = \mathcal{S}^m_{++}$, and the point-to-hyperplane distances are pseudo-gyrodistances [55, Thms. 2.23–2.25], Def. 3.1 recovers the corresponding gyro SPD FC layers [56, Props. 3.4–3.6].*

The crux of Def. 3.1 lies in the point-to-hyperplane distance and solving the resulting m equations. Using the true point-to-hyperplane distance presents three difficulties: it has no general closed-form expression on arbitrary Riemannian manifolds; on a specific manifold, evaluating its infimum may require a difficult geometry-specific nonconvex optimization [17, Sec. 3.1]; and having a computable distance alone does not ensure that the m equations in Def. 3.1 admit a common output Y, as illustrated by the counterexample in Chen et al. [21, App. D]. To avoid these difficulties, we adopt the Riemannian-trigonometric point-to-hyperplane pseudo-distance of Chen et al. [17, Def. 3.1 and Thm. 3.2]: $\mathrm{d}(X, H_{A_k,P_k}) = \frac{|\langle \mathrm{Log}_{P_k}(X), A_k\rangle_{P_k}|}{\|A_k\|_{P_k}}$, where $\|\cdot\|_{P_k}$ denotes the norm induced by $\langle\cdot,\cdot\rangle_{P_k}$. Under this pseudo-distance, the implicit Def. 3.1 admits the following explicit solution.

Theorem 3.3 (Riemannian FC Layers)**.** [↓] *Following the notation in Def. 3.1 the Riemannian FC layer $\mathcal{F}(\cdot) : \mathcal{N} \to \mathcal{M}$ for the input $X \in \mathcal{N}$ is $Y = \mathrm{Exp}^{\mathcal{M}}_E\left(\sum_{i=1}^m \langle \mathrm{Log}^{\mathcal{N}}_{P_i}(X), A_i\rangle^{\mathcal{N}}_{P_i} B_i\right)$, where $\mathrm{Exp}^{\mathcal{M}}_E$ is the Riemannian exponential map on $\mathcal{M}$.*

The tangent FC layer [28, Lem. 6] uses a single tangent space and is written as $\mathrm{Exp}_E(f(\mathrm{Log}_E(X)))$, with f being a Euclidean FC layer. In contrast, our formulation involves multiple tangent spaces, where each $\langle \mathrm{Log}^{\mathcal{N}} P_i(X), A_i\rangle^{\mathcal{N}}_{P_i} = 0$ corresponds to a Riemannian hyperplane H_{A_i,P_i}. Moreover, our formulation naturally incorporates prior Riemannian FC layers without requiring additional geometric or algebraic structures, as summarized in Tab. 2.

Remark 3.4. The point-to-hyperplane pseudo-distance adopted above coincides with the true geodesic point-to-hyperplane distance when the manifold is isometric to an Euclidean space. Specifically, if $\phi : (\mathcal{N}, g) \to (\mathbb{R}^n, \langle\cdot,\cdot\rangle)$ is an isometry, extending Chen et al. [15, Lem. 3.5] gives $\inf_{Q\in H_{A,P}} \mathrm{d}(X,Q) = \frac{|\langle \mathrm{Log}_P(X), A\rangle_P|}{\|A\|_P}$. This includes Euclidean space as well as LEM and LCM on SPD manifolds.

Remark 3.5. Theoretically, the origin $E^{\mathcal{M}}$ in Def. 3.1 can be chosen arbitrarily. On a homogeneous space, any two choices and the FC layers defined through them are identified by an isometry (Thm. E.1). In practice, we choose canonical origins that simplify the Riemannian operators and the resulting FC expressions. The examples in Sec. 4 follow this computational criterion.

3.2 Riemannian convolutional layers

As shown in [66, Sec. 3.4], Euclidean convolution first concatenates the features within each receptive field and then applies an FC transformation. The concatenated feature vector $x \in (\mathbb{R}^n)^c$ is an element of the product space $(\mathbb{R}^n)^c$, and its k-th output is the affine transformation $y_k = \langle a_k, x\rangle - b_k$. This view naturally extends to manifolds: concatenating manifold-valued features forms an element of the product manifold, on which we apply the Riemannian FC layer in Sec. 3.1.

Riemannian convolution. In each receptive field, the product-manifold element $X = (X_1, \ldots, X_c) \in (\mathcal{N})^c$ is fed into k Riemannian FC layers, where k is the number of kernels. Here, each Riemannian FC layer is implemented under the product geometry $(\mathcal{N})^c = \Pi_{i=1}^c \mathcal{N}$, which is detailed in App. E.3. Fig. 1 illustrates the above process.

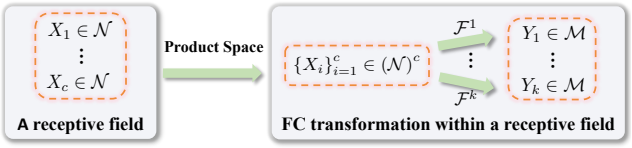


Figure 1: Riemannian convolution within a receptive field. Here, $\mathcal{F}^k(\cdot)$ denotes the k-th FC transformation.

3.3 Parameter trivialization

As convolution uses the FC layer as its prototype, we focus on the FC parameters. Since P_i varies during training, $A_i \in T_{P_i}\mathcal{N}$ cannot be updated directly by a Euclidean optimizer. As shown by Chen et al. [17, Eqs. (12)–(13)], it can be determined from the fixed tangent space at the origin $E^{\mathcal{N}} \in \mathcal{N}$ by[1] $A_i = \Gamma_{E^{\mathcal{N}}\to P_i}(Z_i)$ with $Z_i \in T_{E^{\mathcal{N}}}\mathcal{N}$. Moreover, as shown by Shimizu et al. [66, Sec. 3.1], p_k in Eq. (1) may be over-parameterized, since there are countless p_k satisfying $\langle a_k, p_k\rangle = b_k$. Therefore, following Shimizu et al. [66], each P_i in the Riemannian FC layer is parameterized as $\mathrm{Exp}^{\mathcal{N}}_{E^{\mathcal{N}}}(\gamma_i[Z_i])$, where $\gamma_i \in \mathbb{R}$, $Z_i \neq 0$, and $[Z_i]$ is the unit vector of Z_i. This use of the exponential map to optimize manifold-valued parameters is known as trivialization [43, Sec. 4.1].

Table 1: Comparison of hyperbolic FC layers. An extended table can be found in App. F.

Method	Space	Mechanism	References
Möbius	$\mathbb{P}^n_K$	Tangent	[28, Def. 3.2]
Einstein	$\mathbb{K}^n_K$	Tangent	[51, Thm. 9]
LFC	$\mathbb{H}^n_K$	Spacetime	[14, Eq. (1)]
NestFC	$\mathbb{H}^n_K$	Nested projection	[26, Eq. (14)]
Poincaré FC	$\mathbb{P}^n_K$	Poincaré	[66, Sec. 3.2]
Ours	$\mathbb{P}^n_K, \mathbb{K}^n_K, \mathbb{H}^n_K$	Riemannian	Thms. 4.1 and 4.2

Thus, instead of independently optimizing the two n-dimensional parameters P_i and A_i, we parsimoniously optimize one tangent vector Z_i and one scalar γ_i for each output dimension. This reduces the number of parameters from $2n$ to $n+1$ and allows them to be updated by a Euclidean optimizer, instead of expensive Riemannian optimizers.

[1] Although Γ could be flexibly replaced by other maps between tangent spaces, such as vector transport and differential of group translation, we mainly use parallel transport.

4 Examples

Although Thm. 3.3 is geometry-agnostic, its instantiation can be further simplified under a specific geometry. We now instantiate the FC layer in Thm. 3.3 over different geometries, including three hyperbolic models, five SPD geometries, and two Grassmannian perspectives.

4.1 Hyperbolic vector manifolds

We focus on three hyperbolic models: the Poincaré ball, the Beltrami–Klein model, and the hyperboloid model. The resulting FC layers are denoted by HFC-P, HFC-K, and HFC-H, respectively. Tab. 1 compares our hyperbolic FC layers against prior layers, where we refer to the hyperbolic feature transformation of Fan et al. [26, Eq. (14)] as NestFC.

Poincaré & Beltrami–Klein. These two models admit Möbius and Einstein gyrovector spaces [73], respectively. These structures further simplify the concrete HFC layers.

Theorem 4.1 (HFC-P & HFC-K)**.** [↓] *Let $\mathcal{H}^n \in \{\mathbb{P}^n_K, \mathbb{K}^n_K\}$. Given $x \in \mathcal{H}^n$, the Riemannian FC layer $\mathcal{F}(\cdot) : \mathcal{H}^n \to \mathcal{H}^m$ is $y = \operatorname{Exp}_{\mathbf{0}}\Big((v_1(x), \cdots, v_m(x))^\top\Big)$. Here, $v_i(x) = \langle \operatorname{Log}_{\mathbf{0}}(-p_i \oplus_{\mathcal{H}} x), z_i\rangle$, with the zero vector $\mathbf{0}$ as the origin and $p_i = \operatorname{Exp}_{\mathbf{0}}(\gamma_i[z_i])$. The FC parameters are $\{\gamma_i \in \mathbb{R}\}_{i=1}^m$ and $\{z_i \in \mathbb{R}^n\}_{i=1}^m$. The gyroaddition and Riemannian exponential and logarithmic maps can be found in App. D.1, where $\operatorname{Exp}_{\mathbf{0}}$ ($\operatorname{Log}_{\mathbf{0}}$) shares the same expression in the two models.*

Interestingly, the only difference between the HFC-P and HFC-K layers lies in the gyroaddition. Moreover, HFC-P takes an expression different from that of the Poincaré FC layer [66, Sec. 3.2], since their point-to-hyperplane distances and LHSs of Eq. (3) are different.

Hyperboloid. The origin is defined as $e = (1/\sqrt{|K|}, 0, \cdots, 0)^\top$, which corresponds to the Poincaré origin under the stereographic projection [67, Sec. 2.1]. Then, we have the following.

Theorem 4.2 (HFC-H FC layer)**.** [↓] *The Riemannian FC layer $\mathcal{F}(\cdot) : \mathbb{H}^n_K \to \mathbb{H}^m_K$ for $x \in \mathbb{H}^n_K$ is $y = \operatorname{Exp}_e\Big((0, v_1(x), \cdots, v_m(x))^\top\Big)$, where $v_i(x) = \big\langle \operatorname{Log}_{p_i}(x), \Gamma_{e\to p_i}(0, z_i)\big\rangle$, and $p_i = \operatorname{Exp}_e(\gamma_i[(0, z_i^\top)^\top])$, with $\gamma_i \in \mathbb{R}$ and $z_i \in \mathbb{R}^n$ as parameters.*

Substituting the hyperbolic operators into the preceding HFC layers gives the following expressions.

Theorem 4.3 (Simplification)**.** [↓] *For the hyperboloid model, we write $x = (x_1, x_s)$. For each HFC layer, $v_i(x)$ can be further simplified:*

$$\mathbb{P}^n_K : v_i(x) = \|z_i\| \frac{\tanh^{-1}\left(\sqrt{|K|}Q_i^{\mathrm{P}}\right)}{\sqrt{|K|}Q_i^{\mathrm{P}}} \frac{\left[1 + \tanh^2(\sqrt{|K|}\gamma_i)\right]\langle x, [z_i]\rangle - \frac{\tanh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}\left(1 + |K|\|x\|^2\right)}{D_i^{\mathrm{P}}},$$

$$\mathbb{K}^n_K : v_i(x) = \|z_i\| \frac{\tanh^{-1}\left(\sqrt{|K|}Q_i^{\mathrm{K}}\right)}{\sqrt{|K|}Q_i^{\mathrm{K}}} \frac{\langle x, [z_i]\rangle - \frac{\tanh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}}{D_i^{\mathrm{K}}},$$

$$\mathbb{H}^n_K : v_i(x) = \|z_i\| \frac{\cosh^{-1}\left(\sqrt{|K|}\left[\cosh(\sqrt{|K|}\gamma_i)x_1 - \sinh(\sqrt{|K|}\gamma_i)\langle x_s, [z_i]\rangle\right]\right)}{\sqrt{|K|\left[\cosh(\sqrt{|K|}\gamma_i)x_1 - \sinh(\sqrt{|K|}\gamma_i)\langle x_s, [z_i]\rangle\right]^2 - 1}} \left[\cosh(\sqrt{|K|}\gamma_i)\langle x_s, [z_i]\rangle - \sinh(\sqrt{|K|}\gamma_i)x_1\right],$$

$$D_i^{\mathrm{P}} = 1 - 2\sqrt{|K|}\tanh(\sqrt{|K|}\gamma_i)\langle x, [z_i]\rangle + |K|\tanh^2(\sqrt{|K|}\gamma_i)\|x\|^2,$$

$$Q_i^{\mathrm{P}} = \sqrt{\frac{\|x\|^2 + \frac{\tanh^2(\sqrt{|K|}\gamma_i)}{|K|} - \frac{2\tanh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}\langle x, [z_i]\rangle}{D_i^{\mathrm{P}}}},$$

$$D_i^{\mathrm{K}} = 1 - \sqrt{|K|}\tanh(\sqrt{|K|}\gamma_i)\langle x, [z_i]\rangle,$$

$$Q_i^{\mathrm{K}} = \sqrt{\frac{\|x\|^2 + \frac{\tanh^2(\sqrt{|K|}\gamma_i)}{|K|} - \frac{2\tanh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}\langle x, [z_i]\rangle - \tanh^2(\sqrt{|K|}\gamma_i)\left(\|x\|^2 - \langle x, [z_i]\rangle^2\right)}{(D_i^{\mathrm{K}})^2}}.$$

Although Thm. 4.3 appears more complicated than the original formulations, it enables efficient batched computation and greatly reduces GPU memory consumption. Directly implementing Thms. 4.1 and 4.2 requires either looping over the m output dimensions or materializing a large $B \times m \times n$ intermediate tensor. As shown in Thm. 4.3, the formulas mainly depend on $\langle x, [z_i] \rangle$ for HFC-P and HFC-K and $\langle x_s, [z_i] \rangle$ for HFC-H. Stacking B inputs into $X \in \mathbb{R}^{B \times n}$ (or their spatial components into $X_s \in \mathbb{R}^{B \times n}$) and the unit directions into $U = ([z_1], \dots, [z_m])^\top \in \mathbb{R}^{m \times n}$ allows all these inner products to be computed at once as $XU^\top$ (or $X_s U^\top$). The remaining terms can be evaluated elementwise.

4.2 SPD matrix manifolds

We focus on five popular Riemannian metrics, *i.e.*, LEM, AIM, PEM, LCM, and BWM. We define the identity matrix I as the origin, since it corresponds to the zero matrix under the matrix logarithm.

Theorem 4.4 (SPD FC Layers)**.** [↓] *Given an SPD matrix $S \in \mathcal{S}^n_{++}$, the outputs of the SPD FC layers $\mathcal{F}(\cdot) : \mathcal{S}^n_{++} \to \mathcal{S}^m_{++}$ under different Riemannian metrics are*

$$\textit{LEM}: Y = \exp\left(V^{\mathrm{LE}}\right), V^{\mathrm{LE}}_{ij} = \begin{cases} \frac{1}{\sqrt{\alpha}} v^{\mathrm{LE}}_{ii}(S) + \mu \sum_{k=1}^m v^{\mathrm{LE}}_{kk}(S), & \textit{if } i = j \\ \frac{1}{\sqrt{2\alpha}} v^{\mathrm{LE}}_{ij}(S), & \textit{if } i > j \\ V^{\mathrm{LE}}_{ji}, & \textit{otherwise} \end{cases} \tag{4}$$

$$\textit{AIM}: Y = \exp\left(V^{\mathrm{AI}}\right), V^{\mathrm{AI}}_{ij} = \begin{cases} \frac{1}{\sqrt{\alpha}} v^{\mathrm{AI}}_{ii}(S) + \mu \sum_{k=1}^m v^{\mathrm{AI}}_{kk}(S), & \textit{if } i = j \\ \frac{1}{\sqrt{2\alpha}} v^{\mathrm{AI}}_{ij}(S), & \textit{if } i > j \\ V^{\mathrm{AI}}_{ji}, & \textit{otherwise} \end{cases} \tag{5}$$

$$\textit{PEM}: Y = \left(I + V^{\mathrm{PE}}\right)^{\frac{1}{\theta}}, V^{\mathrm{PE}}_{ij} = \begin{cases} \frac{1}{\sqrt{\alpha}} v^{\mathrm{PE}}_{ii}(S) + \mu \sum_{k=1}^m v^{\mathrm{PE}}_{kk}(S), & \textit{if } i = j \\ \frac{1}{\sqrt{2\alpha}} v^{\mathrm{PE}}_{ij}(S), & \textit{if } i > j \\ V^{\mathrm{PE}}_{ji}, & \textit{otherwise} \end{cases} \tag{6}$$

$$\textit{LCM}: Y = V^{\mathrm{LC}} (V^{\mathrm{LC}})^\top, V^{\mathrm{LC}}_{ij} = \begin{cases} \exp\left(v^{\mathrm{LC}}_{ii}(S)\right), & \textit{if } i = j \\ v^{\mathrm{LC}}_{ij}(S), & \textit{if } i > j \\ 0, & \textit{otherwise} \end{cases} \tag{7}$$

$$\textit{BWM}: Y = \left(I + \frac{1}{2} V^{\mathrm{BW}}\right)^2, V^{\mathrm{BW}}_{ij} = \begin{cases} v^{\mathrm{BW}}_{ii}(S), & \textit{if } i = j \\ \frac{1}{\sqrt{2}} v^{\mathrm{BW}}_{ij}(S), & \textit{if } i > j \\ V^{\mathrm{BW}}_{ji}, & \textit{otherwise} \end{cases} \tag{8}$$

Here, $v_{ij}(S)$ under different metrics is defined as

$$\textit{LEM}: \langle \log(S) - \log(P_{ij}), Z_{ij} \rangle^{(\alpha,\beta)}, \quad \textit{AIM}: \left\langle \log(P_{ij}^{-\frac{1}{2}} S P_{ij}^{-\frac{1}{2}}), Z_{ij} \right\rangle^{(\alpha,\beta)},$$

$$\textit{PEM}: \left\langle S^\theta - P_{ij}^\theta, Z_{ij} \right\rangle^{(\alpha,\beta)}, \quad \textit{LCM}: \left\langle \lfloor K \rfloor - \lfloor L_{ij} \rfloor + \mathrm{Dlog}(\mathbb{K} \mathbb{L}_{ij}^{-1}), \lfloor Z_{ij} \rfloor + \frac{1}{2} \mathbb{Z}_{ij} \right\rangle,$$

$$\textit{BWM}: \left\langle (P_{ij} S)^{\frac{1}{2}} + (S P_{ij})^{\frac{1}{2}} - 2 P_{ij}, \mathcal{L}_{P_{ij}}(L_{ij} Z_{ij} L_{ij}^\top) \right\rangle,$$

The above notations are defined in the following.

- $Z_{ij} \in T_I \mathcal{S}^n_{++} \cong \mathcal{S}^n$ *and* $P_{ij} \in \mathcal{S}^n_{++}$ *are the parameters for* $1 \le i \le j \le m$,
- $\log(\cdot)$ *is the matrix logarithm.* $\mathrm{Dlog}(\cdot)$ *is the diagonal element-wise logarithm.* $\lfloor \cdot \rfloor$ *is the strictly lower part of a square matrix.* $\mathrm{Chol}(\cdot)$ *is the Cholesky decomposition.* $\mathbb{V}$ *is a diagonal matrix with diagonal elements of the square matrix* V. $\mathcal{L}_P(V)$ *is the solution to the matrix linear system* $\mathcal{L}_P[V]P + P\mathcal{L}_P[V] = V$*, known as the Lyapunov operator.* $\mu = \frac{1}{n}\left(\frac{1}{\sqrt{\alpha + n\beta}} - \frac{1}{\sqrt{\alpha}}\right)$, $K = \mathrm{Chol}(S)$ *and* $L_{ij} = \mathrm{Chol}(P_{ij})$.
- $\langle \cdot, \cdot \rangle$ *and* $\langle \cdot, \cdot \rangle^{(\alpha,\beta)}$ *are the Frobenius inner product and the* $\mathrm{O}(n)$*-invariant inner product defined in Eq.* (26)*.*
- *Due to the incompleteness of PEM and BWM, there are constraints on* V^{PE} *and* V^{BW}*:* $I + \theta V^{\mathrm{PE}} \in \mathcal{S}^m_{++}$ *and* $I + \frac{1}{2} V^{\mathrm{BW}} \in \mathcal{S}^n_{++}$*. Both constraints can be solved by numerical regularization, as detailed in Rmk. G.4.*

The Euclidean affine FC $y = Ax + b$ incorporates the linear map $y = Ax$, the most natural map between linear spaces. As shown by Arsigny et al. [2, Sec. 4.4] and Chen et al. [18, Thm. 1], the SPD

manifold admits two vector space structures with respect to LEM and LCM. Similar to the Euclidean FC layer, our SPD FC layer also incorporates linear maps over these vector structures. Denoting the addition and scalar product by $\oplus^{\mathrm{LE}}$ ($\oplus^{\mathrm{LC}}$) and $\odot^{\mathrm{LE}}$ ($\odot^{\mathrm{LC}}$), respectively, which are detailed in App. J.7, we have the following result.

Proposition 4.5. [↓] *The LEM- and LCM-SPD FC layers incorporate the linear homomorphisms over the vector spaces* $\{\mathcal{S}^n_{++}, \oplus^{\mathrm{LE}}, \odot^{\mathrm{LE}}\}$ *and* $\{\mathcal{S}^n_{++}, \oplus^{\mathrm{LC}}, \odot^{\mathrm{LC}}\}$*, respectively.*

Comparison. As summarized in Tab. 2, the three gyro SPD FC layers from Nguyen et al. [56, Props. 3.4-3.6] and the two flat SPD FC layers [57] are incorporated by our SPD FC layers.

Table 2: Comparison with the existing SPD FC layers.

SPD FC Layer	Geometries	Requirement	Incorporated by Ours
Gyro FC [56]	AIM, LEM & LCM	Gyrovector	✓(App. G.1)
Flat FC [57]	LEM & LCM	Flat geometry	✓(App. G.2)
Symmetric FC [57]	AIM	Invariant metric Symmetric space	N/A
Ours	Riemannian spaces	Riemannian	N/A

Simplification and convolution. Following the trivialization in Sec. 3.3, the SPD FC layers under LEM, AIM, LCM, and PEM can be further simplified, as detailed in App. G.3. The convolution is defined as Sec. 3.2, with $\mathcal{M} = \mathcal{S}^m_{++}$ and $\mathcal{N} = \mathcal{S}^n_{++}$.

4.3 Grassmannian matrix manifolds

We instantiate our FC layers over the ONB and PP Grassmannian and define Grassmannian Convolution (GrConv) according to Sec. 3.2. We then compare GrConv with existing popular Grassmannian transformations, concluding that GrConv is more flexible in both dimensionality and geometry.

ONB. We denote by $I_{p,n} = [I_p, \mathbf{0}]^\top \in \mathbb{R}^{n\times p}$, with I_p being the $p \times p$ identity matrix. We define it as the Grassmannian origin, since it corresponds to $I_n \in \mathrm{O}(n)$ in the quotient structure [5, Sec. 2.2]. As in Sec. 3.3, the FC parameters are modeled by parallel transport and the Riemannian exponential map at $I_{p,n}$. The concrete ONB Grassmannian FC layer can be further simplified.

Theorem 4.6 (ONB)**.** [↓] *Given* $U \in \mathrm{Gr}(p,n)$*, the ONB Grassmannian FC layer* $\mathcal{F}(\cdot) : \mathrm{Gr}(p,n) \to \mathrm{Gr}(q,m)$ *is* $Y = \begin{pmatrix} R\cos(\Sigma)R^\top \\ O\sin(\Sigma)R^\top \end{pmatrix}$*, with* $B^{\mathrm{ONB}} \overset{SVD}{:=} O\Sigma R^\top \in \mathbb{R}^{(m-q)\times q}$*. Each* (i,j) *element of* $B^{\mathrm{ONB}} \in \mathbb{R}^{(m-q)\times q}$ *is* $\left\langle \mathrm{Log}^{\mathrm{ONB}}_{P_{ij}}(U), T_{ij}B_{Z_{ij}} \right\rangle$*, with* $T_{ij} = \begin{pmatrix} -R_{ij}\sin(\Sigma_{ij})O_{ij}^\top \\ O_{ij}\cos(\Sigma_{ij})O_{ij}^\top + I_{n-p} - O_{ij}O_{ij}^\top \end{pmatrix}$*. Here,* $\gamma_{ij}[B_{Z_{ij}}] \overset{SVD}{:=} O_{ij}\Sigma_{ij}R_{ij}^\top$ *is the SVD decomposition. The FC parameters are* $B_{Z_{ij}} \in \mathbb{R}^{(n-p)\times p}$ *and* $\gamma_{ij} \in \mathbb{R}$ *for* $1 \le i \le m-q$ *and* $1 \le j \le q$*.*

PP. We define the PP origin as $\widetilde{I}_{p,n} = I_{p,n}I_{p,n}^\top$, since it corresponds to $I_{p,n}$ [5, Eq. 2.11]. Similarly, we model the FC parameters by parallel transport and the Riemannian exponential map at $\widetilde{I}_{p,n}$. The PP Grassmannian FC layer can be further simplified. In addition, the Riemannian logarithm under the PP Grassmannian can be calculated using the ONB logarithm to support auto-differentiation [56, Prop. 3.12]. For more details, please refer to the proof of the following theorem.

Theorem 4.7 (PP)**.** [↓] *Given* $X \in \widetilde{\mathrm{Gr}}(p,n)$*, the PP Grassmannian FC layer* $\mathcal{F}(\cdot) : \widetilde{\mathrm{Gr}}(p,n) \to \widetilde{\mathrm{Gr}}(q,m)$ *is* $Y = \widetilde{U}\widetilde{U}^\top$*, with* $\widetilde{U} = \left(\exp\left(\begin{pmatrix} 0 & -(B^{\mathrm{PP}})^T \\ B^{\mathrm{PP}} & 0 \end{pmatrix}\right)\right)_{1:q}$*, where* $(\cdot)_{1:q}$ *returns the first-*q *columns of the input square matrix. Each* (i,j) *element of* $B^{\mathrm{PP}} \in \mathbb{R}^{(m-q)\times q}$ *is defined as* $\frac{1}{2}\left\langle \pi_{*,\pi(P)}\left(\mathrm{Log}^{\mathrm{ONB}}_{(O_{ij})_{1:p}}(\pi^{-1}(X))\right), O_{ij}Z_{ij}O_{ij}^\top \right\rangle$*, with* $O_{ij} = \exp\left(\begin{pmatrix} 0 & -(\gamma_{ij}[B_{Z_{ij}}])^T \\ \gamma_{ij}[B_{Z_{ij}}] & 0 \end{pmatrix}\right)$*, where* $\pi(U) = UU^\top$*, and* $\pi_{*,U}(V) = UV^\top + VU^\top$ *is the differential map for all* $U \in \mathrm{Gr}(p,n)$ *and* $V \in T_U\mathrm{Gr}(p,n)$*. The FC parameters are* $B_{Z_{ij}} \in \mathbb{R}^{(n-p)\times p}$ *and* $\gamma_{ij} \in \mathbb{R}$ *for* $1 \le i \le m-q$ *and* $1 \le j \le q$*.*

Comparison. Huang et al. [36] proposed FRMap + ReOrth layers to perform transformations over the ONB Grassmannian via left matrix product (FRMap) and QR decomposition (ReOrth). Nguyen [54] proposed PP scaling for the PP Grassmannian using the tangent space at the identity. Nguyen and Yang [55] extended PP scaling to the ONB Grassmannian.

Table 4: Comparison of hyperbolic FC layers under the same two-layer HNN, reporting AUC (%), full-model registered parameter elements, and peak allocated memory (MiB). The top 3 AUC results are highlighted with **red**, **blue**, and **cyan**. The worst value for each resource is underlined.

Method	Mechanism	Space	Disease ($\delta = 0$)			Airport ($\delta = 1$)			Pubmed ($\delta = 3.5$)			Cora ($\delta = 11$)		
			AUC	#Param	MiB	AUC	#Param	MiB	AUC	#Param	MiB	AUC	#Param	MiB
Möbius [28]	Tangent	$\mathbb{P}^n_K$	76.73 ± 4.86	464	77.20	93.26 ± 0.43	480	123.84	**94.95 ± 0.06**	8,288	3273.19	90.75 ± 0.47	23,216	176.57
Einstein [51]	Tangent	$\mathbb{K}^n_K$	77.34 ± 2.56	464	77.32	92.72 ± 0.07	480	125.70	**94.99 ± 0.13**	8,288	3273.19	89.73 ± 0.21	23,216	176.57
LorentzTan	Tangent	$\mathbb{H}^n_K$	76.54 ± 0.83	464	82.03	92.70 ± 0.19	480	125.46	**94.99 ± 0.13**	8,288	3653.47	89.73 ± 0.21	23,216	326.61
LFC [14]	Spacetime	$\mathbb{H}^n_K$	78.00 ± 0.60	563	77.28	92.63 ± 0.27	580	118.01	94.22 ± 0.11	8,876	3653.48	**91.74 ± 0.12**	24,737	326.62
NestFC [26]	Nested projection	$\mathbb{H}^n_K$	71.30 ± 3.50	811	78.35	83.19 ± 0.90	850	119.20	88.07 ± 0.75	258,514	3659.20	86.08 ± 0.84	2,076,931	1234.25
Poincaré FC [66]	Riemannian	$\mathbb{P}^n_K$	79.19 ± 2.05	528	78.72	**94.21 ± 0.43**	544	127.80	94.30 ± 0.19	8,352	3294.38	87.16 ± 0.95	23,280	176.94
HFC-P	Riemannian	$\mathbb{P}^n_K$	**80.66 ± 1.35**	496	83.42	94.13 ± 0.47	512	133.88	94.77 ± 0.28	8,320	3314.73	**90.76 ± 0.57**	23,248	176.57
HFC-K	Riemannian	$\mathbb{K}^n_K$	**80.58 ± 1.59**	496	85.10	**94.32 ± 0.27**	512	137.20	94.61 ± 0.27	8,320	3329.29	89.94 ± 0.49	23,248	176.57
HFC-H	Riemannian	$\mathbb{H}^n_K$	**82.93 ± 0.89**	496	89.82	**95.18 ± 0.18**	512	134.76	93.99 ± 0.25	8,320	3653.48	**92.20 ± 0.25**	23,248	326.61

In addition, Nguyen and Yang [55] used gyrogroup left translation (GrTrans) as a transformation. These layers are briefly recapped in App. H. However, these previous layers fail to faithfully respect the Grassmannian geometries and lack flexibility regarding dimensions and perspectives. In contrast, given a c-channel Grassmannian $\mathrm{Gr}(p, n)$ (or $\widetilde{\mathrm{Gr}}(p, n)$) input, our GrConv can adjust all dimensions across both perspectives, enabling more flexibility. Tab. 3 summarizes the above discussion.

Table 3: Our GrConv against the existing Grassmannian transformation layers.

Methods	Perspective	Flexible dimensions		
		Subspace p	Ambient n	Channel
FRMap + ReOrth [36, Eqs. (2-4)]	ONB	✗	✓	✗
PP Scaling [54, Sec. 4.2.2]	PP	✗	✗	✗
ONB Scaling [55, Sec. 3.2]	ONB	✗	✗	✗
GrTrans [55, Sec. 2.3.2]	ONB + PP	✗	✗	✗
GrConv	ONB + PP	✓	✓	✓

4.4 Manifold embedding

In several applications [12, 47, 81, 56], Euclidean features are embedded into the manifold via $\mathrm{Exp}_E(Ax + b)$. As detailed in App. E.4, our framework implies that this operation follows the Riemannian FC layer between the Euclidean space and the target manifold, *i.e.*, $\mathcal{F}(\cdot) : \mathbb{R}^n \to \mathcal{M}$.

5 Experiments

We evaluate the effectiveness of our layers on different manifolds. We refer the reader to Apps. I.1 to I.3 for experimental details on the hyperbolic, SPD, and Grassmannian spaces, respectively.

5.1 Experiments on the hyperbolic manifold

We compare our HFC layers against other hyperbolic transformation layers, including Möbius [28] and Einstein [51] transformations via tangent spaces, LFC [14] via spacetime, NestFC [26] via nested projections, and the Poincaré FC layer [66]. Following Chami et al. [12], we adopt four graph datasets for the link prediction task: Disease [1], Airport [80], Pubmed [53], and Cora [64].

Results on HNN. Following the HNN implementations [28, 12, 51], we compare different transformation layers using the same two-layer HNN backbone. For every method, the first transformation maps the input feature dimension n_{in} to 16, and the second maps 16 to 16. Mimicking Möbius and Einstein transformations, we further implement the tangent transformation on the hyperboloid model, $\mathrm{Log}_e(M\,\mathrm{Log}_e(x))$, referred to as LorentzTan. Tab. 4 presents the 5-fold average test AUC, number of parameters, and peak allocated GPU memory. We have the following key observations. (1) **Effectiveness:** Our HFC layers generally achieve superior performance over prior hyperbolic layers. (2) **Hyperbolicity:** Riemannian transformations outperform tangent or spacetime transformations on low-δ Disease and Airport. On the higher-δ datasets, tangent-space transformations remain competitive on Pubmed, whereas HFC-H performs best on Cora. These results suggest that the effectiveness of tangent-space approximations depends on data hyperbolicity. (3) **Representation power & metrics:** The optimal models vary across datasets. Among our HFC variants, HFC-P performs best on Pubmed, whereas HFC-H performs best on Disease, Airport, and Cora, demonstrating the benefit of adapting the FC layer across hyperbolic models. (4) **Parameter and memory efficiency.** NestFC is the least parameter-efficient and the least memory-efficient for high-dimensional inputs, due to its matrix-manifold parameters. On Cora, it uses $\mathbf{89.3}\times$ as many parameters and $\mathbf{3.8}\times$ as much peak memory as HFC-H. Further analysis in App. I.1.3 shows that our HFC layers have the same $O(Bnm)$ asymptotic complexity and comparable parameter dimensions as most existing hyperbolic FC layers, while NestFC has the highest asymptotic complexity and the largest parameter dimension.

Table 5: Comparison of hyperbolic transformations under different settings of Poincaré RResNet.

Dataset	Number of horospheres	50			250		
	Dim	8	16	32	8	16	32
Disease	RResNet [38]	76.0 ± 1.7	78.0 ± 2.2	77.4 ± 2.2	71.5 ± 5.1	78.1 ± 3.3	76.5 ± 2.5
	Möbius+RResNet	74.6 ± 1.9	74.6 ± 5.7	75.1 ± 2.1	74.0 ± 2.7	71.0 ± 5.2	73.3 ± 3.4
	Poincaré FC+RResNet	80.4 ± 0.7	79.1 ± 1.8	79.1 ± 1.6	80.6 ± 0.8	79.1 ± 0.7	80.1 ± 1.4
	HFC-P+RResNet	**81.1 ± 0.6**	**80.0 ± 0.4**	**81.0 ± 0.6**	**80.9 ± 0.6**	**82.3 ± 0.6**	**82.1 ± 0.3**
Airport	RResNet [38]	93.4 ± 1.1	92.6 ± 1.1	93.0 ± 0.2	93.0 ± 0.4	93.0 ± 1.6	89.6 ± 4.7
	Möbius+RResNet	92.9 ± 0.5	93.0 ± 0.3	92.6 ± 0.3	92.9 ± 0.1	93.2 ± 0.2	92.9 ± 0.6
	Poincaré FC+RResNet	92.8 ± 0.6	93.4 ± 0.6	93.8 ± 0.4	93.5 ± 0.4	93.1 ± 0.4	93.8 ± 0.7
	HFC-P+RResNet	**94.1 ± 0.5**	**93.5 ± 0.3**	**94.8 ± 0.5**	**94.1 ± 0.6**	**94.0 ± 0.4**	**94.3 ± 0.4**
Cora	RResNet [38]	**86.7 ± 1.2**	87.2 ± 1.4	82.4 ± 3.5	82.7 ± 3.0	84.0 ± 3.7	83.3 ± 1.6
	Möbius+RResNet	84.6 ± 2.9	86.8 ± 2.1	83.1 ± 2.5	84.1 ± 2.4	83.2 ± 1.6	83.9 ± 2.9
	Poincaré FC+RResNet	83.8 ± 2.4	84.6 ± 0.9	83.3 ± 2.7	82.8 ± 3.3	82.8 ± 3.6	83.3 ± 3.3
	HFC-P+RResNet	85.6 ± 0.8	**87.6 ± 0.8**	**87.2 ± 1.8**	**87.68 ± 1.81**	**86.08 ± 1.72**	**86.97 ± 1.04**

Results on RResNet. We conduct ablations on the RResNet backbone [38]. Since hyperbolic RResNet is built on the Poincaré ball, we compare Poincaré transformation layers, *i.e.*, Möbius, Poincaré FC, and our HFC-P. In the vanilla RResNet, inputs are first projected to the target dimension using a Euclidean linear layer, followed by mapping to the hyperbolic space and processing with hyperbolic residual blocks. In contrast, we first map the input to the hyperbolic space and then apply a hyperbolic transformation layer before feeding it into the residual blocks. This transformation layer can be instantiated as Möbius, Poincaré FC, or our HFC-P layer. We perform experiments across various configurations of the residual blocks, varying both the hidden dimensions and the number of horospheres. Tab. 5 presents the 5-fold average AUC results. Our HFC-P generally outperforms other hyperbolic transformations, demonstrating its effectiveness.

Results on NHGCN. We further evaluate HFC within NHGCN [26], a hyperbolic graph convolutional network. Appendix I.1.4 shows that HFC improves NHGCN with substantially fewer parameters.

Table 6: Comparison of our SPDNNs against other SPD networks. The ones highlighted with are our special cases, while those marked with * are reproduced by us because official code is unavailable.

Methods	Radar	HDM05	FPHA	NTU60
SPDNet [34]	93.25 ± 1.10	64.57 ± 0.61	85.59 ± 0.72	66.36 ± 0.72
SPDNetBN [8]	94.85 ± 0.99	71.28 ± 0.79	89.33 ± 0.49	69.38 ± 0.84
RResNet-AIM [38]	95.71 ± 0.37	64.95 ± 0.82	86.63 ± 0.55	70.70 ± 3.81
RResNet-LEM [38]	95.89 ± 0.86	70.12 ± 2.45	85.07 ± 0.99	74.67 ± 2.89
SPDNetLieBN-AIM [16]	95.47 ± 0.90	71.83 ± 0.69	90.39 ± 0.66	73.34 ± 0.40
SPDNetLieBN-LCM [16]	94.80 ± 0.71	71.78 ± 0.44	86.33 ± 0.43	72.54 ± 1.09
SPDNetMLR [17]	95.64 ± 0.83	65.90 ± 0.93	85.67 ± 0.69	74.18 ± 1.24
GyroLE* [55]	97.31 ± 0.68	73.17 ± 0.37	90.73 ± 0.92	82.65 ± 0.20
GyroLC* [55]	95.23 ± 0.66	67.53 ± 0.85	76.10 ± 0.63	78.32 ± 0.92
GyroAI* [55]	97.17 ± 0.67	72.34 ± 1.06	89.60 ± 0.37	**83.71 ± 0.32**
GyroSPD++-AIM* [56]	97.01 ± 0.40	69.82 ± 1.79	89.50 ± 0.37	83.14 ± 0.87
GyroSPD++-LEM* [56]	98.08 ± 0.26	77.63 ± 1.01	88.23 ± 0.62	**85.48 ± 1.10**
GyroSPD++-LCM* [56]	97.28 ± 0.78	75.36 ± 1.08	81.83 ± 0.93	74.64 ± 2.49
SPDNN-LEM	**98.27 ± 0.48**	**81.16 ± 0.93**	**91.83 ± 0.41**	**86.72 ± 0.14**
SPDNN-AIM	97.63 ± 0.50	**80.12 ± 0.78**	**91.57 ± 0.40**	82.44 ± 0.18
SPDNN-PEM	**98.43 ± 0.44**	**78.77 ± 0.45**	90.33 ± 0.37	82.61 ± 0.37
SPDNN-LCM	97.65 ± 0.75	75.42 ± 0.95	**91.33 ± 0.24**	83.39 ± 0.10
SPDNN-BWM	**98.72 ± 0.14**	72.49 ± 2.02	87.80 ± 0.56	82.64 ± 0.35

5.2 Experiments on the SPD manifold

Following Huang et al. [35], Brooks et al. [8], Katsman et al. [38], we use the Radar dataset [8] for radar classification, and the HDM05 [52], FPHA [29], and NTU60 [65] datasets for human action recognition. In line with Nguyen et al. [56], we focus on the mutual action in NTU60. Following Wang et al. [76], Nguyen et al. [56], we model each sample sequence as multi-channel SPD covariance matrices of shape $[c, n, n]$.

SPDNN. Our SPDNN has an MLR layer [17] stacked on top of a convolutional layer. We denote by SPDNN-[Metric] the SPDNN using convolution under the specified metric. For SPDNN-LEM, -PEM, and -LCM, the MLR is based on the same metric as the convolution. Since the MLRs under AIM and BWM are less efficient [17], we apply LEM MLR for SPDNN-AIM and -BWM. Moreover,

we trivialize the SPD parameters in the MLR as Sec. 3.3, which can be further simplified (detailed in App. G.4). Consequently, all parameters in the SPDNNs can be optimized by a Euclidean optimizer. We compare our networks against the following SOTA SPD networks: SPDNet [35], SPDNetBN [8], LieBN [16], RResNet [38], MLR [17], Gyro [55], and GyroSPD++ [56].

Results. Tables 6 and 7 report the classification results and training efficiency, respectively. Our SPDNNs consistently outperform other SPD models. Specifically, SPDNNs exceed the classic SPDNet by up to **5.47%, 16.59%, 6.24%, and 20.36%** on the four datasets, respectively. Despite sharing the same architecture, SPDNN generally outperforms GyroSPD++ in both accuracy and efficiency across LEM, LCM, and AIM. This advantage arises because our trivialization not only simplifies the expression of the FC and MLR layers but also mitigates the over-parameterization in GyroSPD++. In GyroSPD++, each output dimension of the FC layer requires two matrix parameters, whereas our approach uses only one matrix and one scalar parameter. This reduction in parameter complexity leads to improved training efficiency and generalization. Furthermore, the variation in optimal metrics across datasets underscores the flexibility of our methods.

Table 7: The average training time per epoch of our SPDNNs compared with GyroSPD++. A full comparison of efficiency can be found in App. I.2.5.

Geometry	Method	Radar	HDM05	FPHA	NTU60
AIM	GyroSPD++	5.09	103.57	66.35	125.05
	SPDNN	4.84	101.80	65.42	124.41
LEM	GyroSPD++	0.99	0.95	0.66	7.58
	SPDNN	0.86	0.74	0.63	5.79
LCM	GyroSPD++	0.66	0.70	0.37	5.74
	SPDNN	0.65	0.59	0.35	3.72

5.3 Experiments on the Grassmannian

We compare our Grassmannian convolutional layer against previous transformation layers, such as FRMap + ReOrth [36], scaling [55], and GrTrans [55]. Compared with these previous layers, our transformation can more faithfully respect Grassmannian geometries while allowing greater flexibility with respect to dimensions and geometries. Following Nguyen and Yang [55], each network consists of one transformation layer followed by classification. The corresponding models are denoted by GrNet [36], GyroGr-Scaling [55], GyroGr [55], GrNN-ONB, and GrNN-PP, respectively. Since our GrConv allows a more flexible change in dimensionality, we also perform ablations on the subspace and ambient dimensions of the output of the FC transformation. The experiments are conducted on the Radar dataset. Following Wang et al. [76], we model each radar signal as a multi-channel Grassmannian tensor, *i.e.*, $[c, n, p]$ for ONB and $[c, n, n]$ for PP. Tab. 8 reports the five-fold averages: GrConv outperforms the baselines, while varying the subspace dimension yields the two best results.

Table 8: Comparison of GrNNs against other Grassmannian networks on the Radar dataset. Those marked with * are reproduced by us because official code is unavailable.

Method	Subspace dims	Ambient dims	Mean ± Std
GrNet [36]	4	20–>16	90.48 ± 0.76
GyroGr-Scaling* [55]	4	20–>20	88.88 ± 1.52
GyroGr* [55]	4	20–>20	90.64 ± 0.57
GrNN-ONB	4–>4	20–>16	93.92 ± 0.74
	4–>4	20–>20	92.83 ± 0.66
	4–>6	20–>16	**95.23 ± 0.96**
	4–>8	20–>16	**94.77 ± 0.81**
GrNN-PP	4–>4	20–>16	94.35 ± 0.42
	4–>4	20–>20	**94.56 ± 0.58**
	4–>6	20–>16	94.51 ± 0.53
	4–>8	20–>16	94.11 ± 0.58

6 Conclusion

This paper extends fundamental FC and convolutional layers to operate on Riemannian manifolds. Our approach offers a naturally geometry-aware generalization that is more broadly applicable than previous work. Several existing Riemannian FC layers are subsumed within our framework as special cases. Empirically, we instantiate our framework across ten different geometries, including three hyperbolic models, five SPD geometries, and two Grassmannian formulations. Extensive experiments on radar classification, human action recognition, and graph link prediction demonstrate the effectiveness and flexibility of our approach. We expect this work to facilitate further advances in deep learning on Riemannian spaces.

Acknowledgments and Disclosure of Funding

This work was supported by the Google PhD Fellowship, a DAAD Research Grant in Germany (No. 57811724), an ELIZA PhD Mobility Scholarship, an ELSA Mobility Grant, and an ELIAS Mobility Grant. The author also acknowledges the support of CINECA and the EuroHPC Joint Undertaking for granting access to Leonardo at CINECA, Italy. The author declares no competing interests.

References

[1] Roy M Anderson and Robert M May. *Infectious diseases of humans: dynamics and control*. Oxford University Press, 1991.

[2] Vincent Arsigny, Pierre Fillard, Xavier Pennec, and Nicholas Ayache. Fast and simple computations on tensors with log-Euclidean metrics. Technical Report RR-5584, INRIA, 2005.

[3] Ekkehard Batzies, Knut Hüper, Luis Machado, and F Silva Leite. Geometric mean and geodesic regression on Grassmannians. *Linear Algebra and its Applications*, 2015.

[4] Ahmad Bdeir, Kristian Schwethelm, and Niels Landwehr. Fully hyperbolic convolutional neural networks for computer vision. In *ICLR*, 2024.

[5] Thomas Bendokat, Ralf Zimmermann, and P-A Absil. A Grassmann manifold handbook: Basic geometry and computational aspects. *Advances in Computational Mathematics*, 2024.

[6] Rajendra Bhatia, Tanvi Jain, and Yongdo Lim. On the Bures-Wasserstein distance between positive definite matrices. *Expositiones Mathematicae*, 37(2):165–191, 2019.

[7] Jose J. Bouza, Chun-Hao Yang, David E. Vaillancourt, and Baba C. Vemuri. MVC-Net: A convolutional neural network architecture for manifold-valued images with applications. *arXiv preprint arXiv:2003.01234*, 2020.

[8] Daniel Brooks, Olivier Schwander, Frédéric Barbaresco, Jean-Yves Schneider, and Matthieu Cord. Riemannian batch normalization for SPD neural networks. In *NeurIPS*, 2019.

[9] James W Cannon, William J Floyd, Richard Kenyon, Walter R Parry, et al. Hyperbolic geometry. *Flavors of geometry*, 31(59-115):2, 1997.

[10] Rudrasis Chakraborty. ManifoldNorm: Extending normalizations on Riemannian manifolds. *arXiv preprint arXiv:2003.13869*, 2020.

[11] Rudrasis Chakraborty, Jose Bouza, Jonathan H Manton, and Baba C Vemuri. Manifoldnet: A deep neural network for manifold-valued data with applications. *IEEE TPAMI*, 2020.

[12] Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. In *NeurIPS*, 2019.

[13] Ricky T. Q. Chen and Yaron Lipman. Flow matching on general geometries. In *ICLR*, 2024.

[14] Weize Chen, Xu Han, Yankai Lin, Hexu Zhao, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. Fully hyperbolic neural networks. In *ACL*, 2022.

[15] Ziheng Chen, Yue Song, Gaowen Liu, Ramana Rao Kompella, Xiaojun Wu, and Nicu Sebe. Riemannian multinomial logistics regression for SPD neural networks. In *CVPR*, 2024.

[16] Ziheng Chen, Yue Song, Yunmei Liu, and Nicu Sebe. A Lie group approach to Riemannian batch normalization. In *ICLR*, 2024.

[17] Ziheng Chen, Yue Song, Xiaojun Wu, and Nicu Sebe. RMLR: Extending multinomial logistic regression into general geometries. In *NeurIPS*, 2024.

[18] Ziheng Chen, Yue Song, Tianyang Xu, Zhiwu Huang, Xiao-Jun Wu, and Nicu Sebe. Adaptive Log-Euclidean metrics for SPD matrix learning. *IEEE TIP*, 2024.

[19] Ziheng Chen, Yue Song, Xiaojun Wu, and Nicu Sebe. Gyrogroup batch normalization. In *ICLR*, 2025.

[20] Ziheng Chen, Xiao-Jun Wu, and Nicu Sebe. Riemannian batch normalization: A gyro approach. *arXiv preprint arXiv:2509.07115*, 2025.

[21] Ziheng Chen, Bernhard Schölkopf, and Nicu Sebe. Hyperbolic Busemann neural networks. In *CVPR*, 2026.

[22] Ziheng Chen, Xiaojun Wu, Bernhard Schölkopf, and Nicu Sebe. Riemannian networks over full-rank correlation matrices. In *ICML*, 2026.

[23] Manfredo Perdigao Do Carmo and J Flaherty Francis. *Riemannian Geometry*, volume 6. Springer, 1992.

[24] Ian L Dryden, Xavier Pennec, and Jean-Marc Peyrat. Power Euclidean metrics for covariance matrices with application to diffusion tensor imaging. *arXiv preprint arXiv:1009.3045*, 2010.

[25] Alan Edelman, Tomás A Arias, and Steven T Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 1998.

[26] Xiran Fan, Chun-Hao Yang, and Baba C. Vemuri. Nested hyperbolic spaces for dimensionality reduction and hyperbolic NN design. In *CVPR*, 2022.

[27] Xingcheng Fu, Yisen Gao, Yuecen Wei, Qingyun Sun, Hao Peng, Jianxin Li, and Xianxian Li. Hyperbolic geometric latent diffusion model for graph generation. In *ICML*, 2024.

[28] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic neural networks. In *NeurIPS*, 2018.

[29] Guillermo Garcia-Hernando, Shanxin Yuan, Seungryul Baek, and Tae-Kyun Kim. First-person hand action benchmark with RGB-D videos and 3D hand pose annotations. In *CVPR*, 2018.

[30] Caglar Gulcehre, Misha Denil, Mateusz Malinowski, Ali Razavi, Razvan Pascanu, Karl Moritz Hermann, Peter Battaglia, Victor Bapst, David Raposo, Adam Santoro, and Nando de Freitas. Hyperbolic attention networks. In *ICLR*, 2019.

[31] Brian C Hall. *Lie groups, Lie algebras, and representations*. Springer, 2013.

[32] Chen Hu, Ziheng Chen, Rui Wang, Yefeng Zheng, and Nicu Sebe. Riemannian high-order pooling for brain foundation models. In *ICLR*, 2026.

[33] Chin-Wei Huang, Milad Aghajohari, Joey Bose, Prakash Panangaden, and Aaron C Courville. Riemannian diffusion models. In *NeurIPS*, 2022.

[34] Zhiwu Huang and Luc Van Gool. A Riemannian network for SPD matrix learning. In *AAAI*, 2017.

[35] Zhiwu Huang, Chengde Wan, Thomas Probst, and Luc Van Gool. Deep learning on Lie groups for skeleton-based action recognition. In *CVPR*, 2017.

[36] Zhiwu Huang, Jiqing Wu, and Luc Van Gool. Building deep networks on Grassmann manifolds. In *AAAI*, 2018.

[37] Shaocheng Jin, Tao Zhou, Rui Wang, Ziheng Chen, Xiaoqing Luo, Xiao-Jun Wu, and Josef Kittler. Towards robust EEG decoding based on riemannian self-attention. In *KDD*, 2026.

[38] Isay Katsman, Eric Chen, Sidhanth Holalkere, Anna Asch, Aaron Lou, Ser Nam Lim, and Christopher M De Sa. Riemannian residual neural networks. In *NeurIPS*, 2024.

[39] Raiyan R. Khan, Philippe Chlenski, and Itsik Pe'er. Hyperbolic genome embeddings. In *ICLR*, 2025.

[40] Diederik P Kingma. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[41] Reinmar Kobler, Jun-ichiro Hirayama, Qibin Zhao, and Motoaki Kawanabe. SPD domain-specific batch normalization to crack interpretable unsupervised domain adaptation in EEG. In *NeurIPS*, 2022.

[42] John M Lee. *Riemannian manifolds: an introduction to curvature*, volume 176. Springer Science & Business Media, 2006.

[43] Mario Lezcano Casado. Trivializations for gradient-based optimization on manifolds. In *NeurIPS*, 2019.

[44] Shanglin Li, Motoaki Kawanabe, and Reinmar J Kobler. SPDIM: Source-free unsupervised conditional and label shift adaptation in EEG. In *ICLR*, 2025.

[45] Shanglin Li, Shiwen Chu, Okan Koç, Yi Ding, Qibin Zhao, Motoaki Kawanabe, and Ziheng Chen. HEEGNet: Hyperbolic embeddings for EEG. In *ICLR*, 2026.

[46] Zhenhua Lin. Riemannian geometry of symmetric positive definite matrices via Cholesky decomposition. *SIAM Journal on Matrix Analysis and Applications*, 40(4):1353–1370, 2019.

[47] Federico López, Beatrice Pozzetti, Steve Trettel, Michael Strube, and Anna Wienhard. Vector-valued distance and Gyrocalculus on the space of symmetric positive definite matrices. In *NeurIPS*, 2021.

[48] Aaron Lou, Isay Katsman, Qingxuan Jiang, Serge Belongie, Ser-Nam Lim, and Christopher De Sa. Differentiating through the Fréchet mean. In *ICML*, 2020.

[49] Miroslav Lovrić, Maung Min-Oo, and Ernst A Ruh. Multivariate normal distributions parametrized as a riemannian symmetric space. *Journal of Multivariate Analysis*, 74(1):36–48, 2000.

[50] Luigi Malagò, Luigi Montrucchio, and Giovanni Pistone. Wasserstein Riemannian geometry of Gaussian densities. *Information Geometry*, 2018.

[51] Yidan Mao, Jing Gu, Marcus C Werner, and Dongmian Zou. Klein model for hyperbolic neural networks. *arXiv preprint arXiv:2410.16813*, 2024.

[52] Meinard Müller, Tido Röder, Michael Clausen, Bernhard Eberhardt, Björn Krüger, and Andreas Weber. Documentation mocap database HDM05. Technical report, Universität Bonn, 2007.

[53] Galileo Namata, Ben London, Lise Getoor, Bert Huang, and U Edu. Query-driven active surveying for collective classification. In *10th International Workshop on Mining and Learning with Graphs*, volume 8, page 1, 2012.

[54] Xuan Son Nguyen. The gyro-structure of some matrix manifolds. In *NeurIPS*, 2022.

[55] Xuan Son Nguyen and Shuo Yang. Building neural networks on matrix manifolds: A Gyrovector space approach. In *ICML*, 2023.

[56] Xuan Son Nguyen, Shuo Yang, and Aymeric Histace. Matrix manifold neural networks++. In *ICLR*, 2024.

[57] Xuan Son Nguyen, Shuo Yang, and Aymeric Histace. Neural networks on symmetric spaces of noncompact type. In *ICLR*, 2025.

[58] Yue-Ting Pan, Jing-Lun Chou, and Chun-Shu Wei. MAtt: A manifold attention network for EEG decoding. In *NeurIPS*, 2022.

[59] Xavier Pennec, Pierre Fillard, and Nicholas Ayache. A Riemannian framework for tensor computing. *IJCV*, 2006.

[60] Peter Petersen. *Riemannian geometry*. Springer, 2006.

[61] Can Pouliquen, Mathurin Massias, and Titouan Vayer. Schur's positive-definite network: Deep learning in the SPD cone with structure. In *ICLR*, 2025.

[62] Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of Adam and beyond. In *ICLR*, 2018.

[63] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

[64] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93–93, 2008.

[65] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. NTU RGB+ D: A large scale dataset for 3D human activity analysis. In *CVPR*, 2016.

[66] Ryohei Shimizu, Yusuke Mukuta, and Tatsuya Harada. Hyperbolic neural networks++. In *ICLR*, 2021.

[67] Ondrej Skopek, Octavian-Eugen Ganea, and Gary Bécigneul. Mixed-curvature variational autoencoders. In *ICLR*, 2020.

[68] Yann Thanwerdas and Xavier Pennec. Is affine-invariance well defined on SPD matrices? a principled continuum of metrics. In *Geometric Science of Information: 4th International Conference*, 2019.

[69] Yann Thanwerdas and Xavier Pennec. Theoretically and computationally convenient geometries on full-rank correlation matrices. *SIAM Journal on Matrix Analysis and Applications*, 43(4): 1851–1872, 2022.

[70] Yann Thanwerdas and Xavier Pennec. O (n)-invariant Riemannian metrics on SPD matrices. *Linear Algebra and its Applications*, 661:163–201, 2023.

[71] Loring W.. Tu. *An introduction to manifolds*. Springer, 2011.

[72] Abraham Ungar. *A gyrovector space approach to hyperbolic geometry*. Springer Nature, 2022.

[73] Abraham Albert Ungar. *Analytic hyperbolic geometry and Albert Einstein's special theory of relativity (Second Edition)*. World Scientific, 2022.

[74] Max van Spengler, Erwin Berkhout, and Pascal Mettes. Poincaré ResNet. In *ICCV*, 2023.

[75] Raviteja Vemulapalli, Felipe Arrate, and Rama Chellappa. Human action recognition by representing 3D skeletons as points in a Lie group. In *CVPR*, 2014.

[76] Rui Wang, Chen Hu, Ziheng Chen, Xiao-Jun Wu, and Xiaoning Song. A Grassmannian manifold self-attention network for signal classification. In *IJCAI*, 2024.

[77] Rui Wang, Xiao-Jun Wu, Ziheng Chen, Cong Hu, and Josef Kittler. SPD manifold deep metric learning for image set classification. *IEEE TNNLS*, 2024.

[78] Rui Wang, Chen Hu, Xiaoning Song, Xiao-Jun Wu, Nicu Sebe, and Ziheng Chen. Towards a general attention framework on gyrovector spaces for matrix manifolds. In *NeurIPS*, 2025.

[79] Rui Wang, Shaocheng Jin, Ziheng Chen, Xiaoqing Luo, and Xiao-Jun Wu. Learning to normalize on the SPD manifold under Bures-Wasserstein geometry. In *CVPR*, 2025.

[80] Muhan Zhang and Yixin Chen. Link prediction based on graph neural networks. In *NeurIPS*, 2018.

[81] Wei Zhao, Federico Lopez, J Maxwell Riestenberg, Michael Strube, Diaaeldin Taha, and Steve Trettel. Modeling graphs beyond hyperbolic: Graph neural networks in symmetric positive definite matrices. In *ECML PKDD*, 2023.

Appendix Contents

List of acronyms **17**

A Use of large language models **17**

B Limitations **17**

C Glossary of symbols **17**

D Geometries of the involved vector and matrix manifolds **17**

D.1 Geometries of the hyperbolic space 17

D.2 Geometries of the SPD manifold 20

D.3 Geometries of the Grassmannian 20

E Discussions on the Riemannian FC and convolutional layer **22**

E.1 Additional discussions on the orthogonal basis 22

E.2 Riemannian fully connected layers under isometric geometry 22

E.3 Riemannian fully connected layers under product geometry 23

E.4 Riemannian fully connected layers and manifold embedding 24

E.5 Relation with the convolution in ManifoldNet 24

F Comparison of our hyperbolic FC layers against previous ones **25**

G Additional details on the SPD fully connected layers **25**

G.1 Relation with the gyro SPD fully connected layers 25

G.2 Relation with the flat SPD fully connected layers 26

G.3 Trivialized SPD fully connected layers 26

G.4 Trivialized SPD multinomial logistic regression 27

G.5 Covariant equivariance 27

H Review of previous Grassmannian transformation layers **28**

I Additional experimental details and results **28**

I.1 Hyperbolic spaces 28

I.1.1 Datasets 28

I.1.2 Implementation details 29

I.1.3 Complexity and parameter analysis 29

I.1.4 Comparison under the NHGCN architecture 30

I.2 SPD manifolds 30

I.2.1 Datasets 30

I.2.2 SPD modeling 31

I.2.3 Implementation details 32

I.2.4 Reproduction Fidelity and Controlled Comparisons 32
I.2.5 Training efficiency 33
I.2.6 Comparison with ManifoldNet and MVC-Net 33
I.3 Grassmannian manifolds 35
I.4 Hardware 35

J Proofs **35**
J.1 Proof of Prop. 3.2 35
J.2 Proof of Thm. 3.3 36
J.3 Proof of Thm. 4.1 36
J.4 Proof of Thm. 4.2 37
J.5 Proof of Thm. 4.3 38
J.6 Proof of Thm. 4.4 43
J.7 Proof of Prop. 4.5 46
J.8 Proof of Thm. 4.6 47
J.9 Proof of Thm. 4.7 48

List of acronyms

ONB	OrthoNormal Basis 2
PP	Projector Perspective 2
FC	Fully Connected 1
GrConv	Grassmannian Convolution 7
AIM	Affine-Invariant Metric 2
BWM	Bures–Wasserstein Metric 2
LCM	Log-Cholesky Metric 2
LEM	Log-Euclidean Metric 2
PEM	Power-Euclidean Metric 2
SPD	Symmetric Positive Definite 1

A Use of large language models

Large Language Models (LLMs) were used primarily for language polishing and minor text editing. In limited cases, they also assisted in translating certain mathematical formulations into PyTorch code. All generated outputs were carefully reviewed and, where necessary, corrected by the authors. The authors take full responsibility for the final content of this paper.

B Limitations

Our framework is designed for computationally tractable Riemannian manifolds, where closed-form expressions for exponential and logarithmic maps are available. This includes many commonly used manifolds, such as hyperbolic, SPD, and Grassmannian spaces. However, in cases where the underlying manifold structure is unknown or lacks tractable Riemannian operators, our approach may not be directly applicable. In such scenarios, future work could explore numerical approximations of Riemannian operators or develop new paradigms for constructing transformation layers for intractable geometries.

C Glossary of symbols

Tab. 10 summarizes all the notation in the main paper.

D Geometries of the involved vector and matrix manifolds

D.1 Geometries of the hyperbolic space

There are five models over the hyperbolic space [9]. We focus on the Poincaré ball, Beltrami–Klein, and hyperboloid models:

$$\text{Poincaré ball: } \mathbb{P}_K^n = \left\{ x \in \mathbb{R}^n \mid \|x\|^2 < -\frac{1}{K} \right\}, \tag{9}$$

$$\text{Beltrami–Klein: } \mathbb{K}_K^n = \left\{ x \in \mathbb{R}^n \mid \|x\|^2 < -\frac{1}{K} \right\}, \tag{10}$$

$$\text{Hyperboloid: } \mathbb{H}_K^n = \left\{ x \in \mathbb{R}^{n+1} \mid \|x\|_{\mathcal{L}}^2 = \frac{1}{K}, x_1 > 0 \right\}, \tag{11}$$

where $\|x\|_{\mathcal{L}}^2 = \sum_{i=2}^{n+1} x_i^2 - x_1^2$ is the Lorentz inner product, and $\|\cdot\|$ is the standard L_2 norm induced by the standard inner product $\langle \cdot, \cdot \rangle$. Here, $K < 0$ is the constant curvature. Although the set of the Poincaré ball is identical to that of the Beltrami–Klein model, their Riemannian metrics are different. In fact, each of the above models has its own Riemannian metric:

$$g_x^{\mathbb{P}}(v, w) = (\lambda_x^K)^2 \langle v, w \rangle, \tag{12}$$

Table 10: Summary of notation.

Notation	Explanation
$\{\mathcal{N}, g^{\mathcal{N}}\}$	Riemannian manifold $\mathcal{N}$ with Riemannian metric $g^{\mathcal{N}}$
$\{\mathcal{M}, g^{\mathcal{M}}\}$	Riemannian manifold $\mathcal{M}$ with Riemannian metric $g^{\mathcal{M}}$
E	Origin of the manifold of interest
$T_P\mathcal{M}$	Tangent space at $P \in \mathcal{M}$
$g_p(\cdot,\cdot)$ or $\langle\cdot,\cdot\rangle_P$	Riemannian metric at P
$\|\cdot\|_P$	The norm induced by $\langle\cdot,\cdot\rangle_P$ on $T_P\mathcal{M}$
$\mathrm{d}(X, Y)$	Geodesic distance between points X and Y
$\mathrm{d}(X, H)$	Point-to-hyperplane pseudo-distance from X to H
Log_P	Riemannian logarithm at P
Exp_P	Riemannian exponential map at P
$\Gamma_{P\to Q}$	Parallel transport from P to Q along the geodesic
$f_{*,P}$	Differential map of the smooth map f at $P \in \mathcal{M}$
$\{B_i\}_{i=1}^m$	Standard orthonormal basis over the m-dimensional $T_E\mathcal{M}$
$\mathbb{P}_K^n$, $\mathbb{K}_K^n$ and $\mathbb{H}_K^n$	Hyperbolic models of Poincaré ball, Beltrami–Klein, and hyperboloid ($K < 0$)
$\mathbb{R}^n$	Euclidean space of n-dimensional vectors
$\langle\cdot,\cdot\rangle_{\mathcal{L}}$	Lorentz inner product
$\oplus_{\mathrm{M}}$ and $\otimes_{\mathrm{M}}$	Möbius gyro addition and scalar product
$\oplus_{\mathrm{E}}$ and $\otimes_{\mathrm{E}}$	Einstein gyro addition and scalar product
$\pi_{\mathbb{K}_K^n\to\mathbb{P}_K^n}$ and $\pi_{\mathbb{P}_K^n\to\mathbb{K}_K^n}$	Riemannian isometries between Beltrami–Klein and Poincaré ball
$\mathcal{S}_{++}^n$	Space of $n \times n$ SPD matrices
$\mathcal{S}^n$	Euclidean space of $n \times n$ symmetric matrices
$\mathcal{L}^n$	Euclidean space of $n \times n$ lower triangular matrices
$\langle\cdot,\cdot\rangle$	Standard Frobenius inner product
$\langle\cdot,\cdot\rangle^{(\alpha,\beta)}$	$\mathrm{O}(n)$-invariant Euclidean metric on $\mathcal{S}^n$ s.t. $\min(\alpha, \alpha + n\beta) > 0$
$\|\cdot\|_{\mathrm{F}}$	Frobenius Norm
$\log$	Matrix logarithm
$\exp$	Matrix exponential
P^θ	Matrix power for SPD matrix P
$\mathcal{L}_P[\cdot]$	Lyapunov operator by $P \in \mathcal{S}_{++}^n$
$\mathscr{L}$	Cholesky decomposition
Dlog	Diagonal element-wise logarithm
$\lfloor\cdot\rfloor$	Strictly lower triangular part of a square matrix
$\mathbb{D}(\cdot)$	A diagonal matrix with diagonal elements from a square matrix
$\mathrm{Gr}(p, n)$	Grassmannian under the ONB perspective
$\widetilde{\mathrm{Gr}}(p, n)$	Grassmannian under the projector perspective
$\mathcal{Q}(\cdot)$	Return an orthogonal matrix by QR decomposition
$[\cdot,\cdot]$	Matrix commutator
$I_{p,n}$	Grassmannian identity under the ONB perspective
$\widetilde{I}_{p,n}$	Grassmannian identity under the projector perspective
I_n	$n \times n$ identity matrix
π	Riemannian isometry from $\mathrm{Gr}(p, n)$ onto $\widetilde{\mathrm{Gr}}(p, n)$
$\overline{(\cdot)}$	$\overline{(\cdot)} = \widetilde{\mathrm{Log}}_{\widetilde{I}_{p,n}}(\cdot)$ with $\widetilde{\mathrm{Log}}$ being the Riemannian logarithm on $\widetilde{\mathrm{Gr}}(p, n)$
$\mathbf{0}$	Zero matrix or vector
$\mathrm{St}(p, n)$	Stiefel manifold of $n \times p$ column-wise orthogonal matrices
$\mathrm{GL}(n)$	General linear group of $n \times n$ invertible matrices
$\mathrm{O}(n)$	Orthogonal group of $n \times n$ orthogonal matrices

$$g_x^{\mathbb{K}}(v, w) = \frac{\langle v, w\rangle}{1 + K\|x\|^2} - \frac{K\langle x, v\rangle\langle x, w\rangle}{\left(1 + K\|x\|^2\right)^2}, \tag{13}$$

$$g_x^{\mathbb{H}}(v, w) = \langle v, w\rangle_{\mathcal{L}} = \sum_{i=2}^{n+1} v_i w_i - v_1 w_1, \tag{14}$$

where $\lambda_x^K = \frac{2}{(1+K\|x\|^2)}$ is a conformal factor.

As shown by Ungar [73], both the Poincaré ball and Beltrami–Klein models admit gyrovector structures, which are the natural manifold counterparts of vector spaces. The Poincaré ball admits a Möbius gyrovector space [73, Ch. 6.14], while the Beltrami–Klein model admits an Einstein gyrovector space [73, Ch. 6.18]. Denoting $\mathcal{H} \in \{\mathbb{P}_K^n, \mathbb{K}_K^n\}$, for any $x, y \in \mathcal{H}$ and $r \in \mathbb{R}$, the gyro

Table 11: Riemannian operators on the Poincaré ball and hyperboloid ($K < 0$).

Operator	$\mathbb{P}^n_K = \left\{x \in \mathbb{R}^n \mid \|x\|^2 < -\frac{1}{K}\right\}$	$\mathbb{H}^n_K = \left\{x \in \mathbb{R}^{n+1} \mid \|x\|^2_{\mathcal{L}} = \frac{1}{K}, x_1 > 0\right\}$, with $\|x\|^2_{\mathcal{L}} = \sum_{i=2}^{n+1} x_i^2 - x_1^2$
$g_x(v, w)$	$(\lambda_x^K)^2 \langle v, w\rangle$, $\lambda_x^K = \frac{2}{(1+K\|x\|^2)}$	$\langle v, w\rangle_{\mathcal{L}} = \sum_{i=2}^{n+1} v_i w_i - v_1 w_1$
$\mathrm{d}(x, y)$	$\frac{2}{\sqrt{\lvert K\rvert}} \tanh^{-1}\left(\sqrt{\lvert K\rvert}\|-x \oplus_{\mathrm{M}} y\|\right)$	$\frac{1}{\sqrt{\lvert K\rvert}} \cosh^{-1}\left(K \langle x, y\rangle_{\mathcal{L}}\right)$
$\mathrm{Log}_x(y)$	$\frac{2}{\sqrt{\lvert K\rvert}\lambda_x^K} \tanh^{-1}\left(\sqrt{\lvert K\rvert}\|-x \oplus_{\mathrm{M}} y\|\right) \frac{-x\oplus_{\mathrm{M}} y}{\|-x\oplus_{\mathrm{M}} y\|}$	$\frac{\cosh^{-1}(K\langle x,y\rangle_{\mathcal{L}})}{\sinh\left(\cosh^{-1}(K\langle x,y\rangle_{\mathcal{L}})\right)}\left(y - K\langle x, y\rangle_{\mathcal{L}} x\right)$
$\Gamma_{x\to y}(v)$	$\frac{\lambda_x^K}{\lambda_y^K} \mathrm{gyr}[y, -x]v$	$v - \frac{K\langle y,v\rangle_{\mathcal{L}}}{1+K\langle x,y\rangle_{\mathcal{L}}}(x + y)$
$\mathrm{Exp}_x(v)$	$x \oplus_{\mathrm{M}} \left(\tanh\left(\sqrt{\lvert K\rvert}\frac{\lambda_x^K\|v\|}{2}\right)\frac{v}{\sqrt{\lvert K\rvert}\|v\|}\right)$	$\cosh\left(\sqrt{\lvert K\rvert}\|v\|_{\mathcal{L}}\right) x + \sinh\left(\sqrt{\lvert K\rvert}\|v\|_{\mathcal{L}}\right)\frac{v}{\sqrt{\lvert K\rvert}\|v\|_{\mathcal{L}}}$
References	[28, 67, 72]	[60, 67]

Table 12: Riemannian operators on the Beltrami–Klein model ($K < 0$).

Operators	$\mathbb{K}^n_K = \{x \in \mathbb{R}^n \mid \|x\|^2 < -\frac{1}{K}\}$
$g_x(v, w)$	$\frac{\langle v,w\rangle}{1+K\|x\|^2} - \frac{K\langle x,v\rangle\langle x,w\rangle}{\left(1+K\|x\|^2\right)^2}$
$\mathrm{d}(x, y)$	$\frac{2}{\sqrt{-K}} \tanh^{-1}\left(\sqrt{-K}\frac{\|-x\oplus_{\mathrm{E}} y\|}{1+\sqrt{1+K\|-x\oplus_{\mathrm{E}} y\|^2}}\right)$
$\mathrm{Exp}_x(v)$	$x \oplus_{\mathrm{E}} \mathrm{Exp}_{\mathbf{0}}\left(\frac{1}{\sqrt{1+K\|x\|^2}}v - \frac{K\langle x,v\rangle}{(1+\sqrt{1+K\|x\|^2})(1+K\|x\|^2)}x\right)$
$\mathrm{Log}_x(y)$	$\frac{1}{\lambda_{\tilde{x}}^K}(\pi_{\mathbb{P}^n_K\to\mathbb{K}^n_K})_{*,\tilde{x}}\left(\mathrm{Log}_{\mathbf{0}}(-x \oplus_{\mathrm{E}} y)\right), \quad \tilde{x} = \pi_{\mathbb{K}^n_K\to\mathbb{P}^n_K}(x)$
References	[73, 20]

operations are defined as

$$\text{Möbius addition} : x \oplus_{\mathrm{M}} y = \frac{\left(1 - 2K\langle x, y\rangle - K\|y\|^2\right)x + \left(1 + K\|x\|^2\right)y}{1 - 2K\langle x, y\rangle + K^2\|x\|^2\|y\|^2}, \quad (15)$$

$$\text{Möbius scalar multiplication} : r \otimes_{\mathrm{M}} x = \frac{\tanh\left(r \tanh^{-1}\left(\sqrt{-K}\|x\|\right)\right)}{\sqrt{-K}}\frac{x}{\|x\|}, \quad (16)$$

$$\text{Einstein addition} : x \oplus_{\mathrm{E}} y = \frac{1}{1 - K\langle x, y\rangle}\left(x + \frac{1}{\gamma_x}y - K\frac{\gamma_x}{1+\gamma_x}\langle x, y\rangle x\right), \quad (17)$$

$$\text{Einstein scalar multiplication} : r \otimes_{\mathrm{E}} x = \frac{\tanh\left(r \tanh^{-1}\left(\sqrt{-K}\|x\|\right)\right)}{\sqrt{-K}}\frac{x}{\|x\|}. \quad (18)$$

where $\gamma_x = 1/\sqrt{1 + K\|x\|^2}$ is called the gamma factor. Interestingly, the scalar gyromultiplications are identical under the Möbius and Einstein gyrovector spaces.

The Poincaré ball and hyperboloid admit closed-form Riemannian operators, as summarized in Tab. 11. The parallel transport over the Poincaré ball requires the notion of gyration [73]:

$$\mathrm{gyr}[x, y]z = \ominus_{\mathrm{M}}(x \oplus_{\mathrm{M}} y) \oplus_{\mathrm{M}} (x \oplus_{\mathrm{M}} (y \oplus_{\mathrm{M}} z)), \forall x, y, z \in \mathbb{P}^n_K. \quad (19)$$

Chen et al. [19, Sec. 5.6] studied the Riemannian structure over the Beltrami–Klein ball. The Beltrami–Klein ball is isometric to the Poincaré ball by

$$\pi_{\mathbb{K}^n_K\to\mathbb{P}^n_K} : x \in \mathbb{K}^n_K \longmapsto \frac{1}{1 + \sqrt{1 + K\|x\|^2}}x \in \mathbb{P}^n_K, \quad (20)$$

$$\pi_{\mathbb{P}^n_K\to\mathbb{K}^n_K} : x \in \mathbb{P}^n_K \longmapsto \frac{2}{1 - K\|x\|^2}x \in \mathbb{K}^n_K. \quad (21)$$

Using the above isometries, Chen et al. [19, Sec. 5.6] introduced closed-form expressions for the Riemannian operators on the Beltrami–Klein ball, as summarized in Tab. 12. In particular, the Riemannian exponential and logarithmic maps at the zero vector $\mathbf{0}$ are identical under the Beltrami–Klein and Poincaré ball models:

$$\operatorname{Exp}_{\mathbf{0}}(v) = \tanh(\sqrt{|K|}\|v\|)\frac{v}{\sqrt{|K|}\|v\|}, \quad \forall v \in T_{\mathbf{0}}\mathcal{H}, \tag{22}$$

$$\operatorname{Log}_{\mathbf{0}}(x) = \tanh^{-1}(\sqrt{|K|}\|x\|)\frac{x}{\sqrt{|K|}\|x\|}, \quad \forall x \in \mathcal{H}, \tag{23}$$

with $\mathcal{H} \in \{\mathbb{K}^n_K, \mathbb{P}^n_K\}$.

As shown by Chen et al. [20, Secs. 5.4 and 5.6], both the Möbius and Einstein gyrovector operations can be expressed by their Riemannian geometries

$$x \oplus_{\mathcal{H}} y = \operatorname{Exp}_x\left(\Gamma_{\mathbf{0}\to x}(\operatorname{Log}_{\mathbf{0}}(y))\right), \tag{24}$$

$$t \otimes_{\mathcal{H}} x = \operatorname{Exp}_{\mathbf{0}}(t \operatorname{Log}_{\mathbf{0}}(x)), \tag{25}$$

where $\oplus_{\mathcal{H}}$ and $\otimes_{\mathcal{H}}$ are the gyroaddition and gyromultiplication under the corresponding model.

D.2 Geometries of the SPD manifold

Tabs. 13 and 14 summarizes the associated Riemannian operators and properties. Following Tab. 10, we further make the following notation. Given any SPD points $P, Q \in \mathcal{S}^n_{++}$ and tangent vectors $V, W \in T_P\mathcal{S}^n_{++}$, we denote $\widetilde{V} = \operatorname{Chol}_{*,P}(V)$, $\widetilde{W} = \operatorname{Chol}_{*,P}(W)$, $L = \operatorname{Chol} P$, and $K = \operatorname{Chol} Q$. The corresponding diagonal matrices with their diagonal elements are denoted by $\widetilde{\mathbb{V}}, \widetilde{\mathbb{W}}, \mathbb{L}$, and $\mathbb{K}$, respectively. For the parallel transport under the BWM, we only present the case where P, Q are commuting matrices, *i.e.* $P = U\Sigma U^\top$ and $Q = U\Delta U^\top$.

The $\mathrm{O}(n)$-invariant Euclidean metric on $\mathcal{S}^n$ [70] is

$$\langle V, W\rangle^{(\alpha,\beta)} = \alpha\langle V, W\rangle + \beta \operatorname{tr}(V)\operatorname{tr}(W), \quad \text{with } \min(\alpha, \alpha + n\beta) > 0. \tag{26}$$

Remark D.1. We make the following remarks with respect to the geometries on the SPD manifold.

- **PEM & EM.** When the power equals 1, the associated PEM is reduced to the Euclidean Metric (EM) [70, Sec. 3.1].
- **Incompleteness & Riemannian exponential maps.** As PEM and BWM are incomplete, their Riemannian exponential maps are locally defined. As shown by Malagò et al. [50, Prop. 9] and implied by Chen et al. [17], Thanwerdas and Pennec [70], the restricted domains are

$$\begin{aligned} &\text{PEM: } P^\theta + P_{\theta *,P}(V) \in \mathcal{S}^n_{++}, \\ &\text{BWM: } \mathcal{L}_P[V] + I \in \mathcal{S}^n_{++}. \end{aligned} \tag{27}$$

  The above restriction can be addressed numerically, for example by ReEig [35]:

$$\widetilde{S} = U \max(\epsilon I, \Sigma) U^\top, \tag{28}$$

  where $S \overset{\text{Eig}}{:=} U\Sigma U^\top$ is the Eigendecomposition.

D.3 Geometries of the Grassmannian

As the set of linear subspaces, the Grassmannian can naturally be represented by any orthonormal basis, which is called the OrthoNormal Basis (ONB) perspective. Under this perspective, the Grassmannian is the quotient of the Stiefel manifold [5], denoted by $\operatorname{Gr}(p, n) \cong \operatorname{St}(p, n)/\mathrm{O}(p)$. Each point is an equivalence class:

$$\operatorname{Gr}(p, n) = \{[U] \mid [U] := \{\widetilde{U} \in \operatorname{St}(p, n) \mid \widetilde{U} = UR, R \in \mathrm{O}(p)\}\}. \tag{29}$$

By abuse of notation, we use $[U]$ and U interchangeably for elements of $\operatorname{Gr}(p, n)$. Each tangent space can be identified as a subspace of a corresponding tangent space on the Stiefel manifold, which is called the horizontal space. Therefore, every tangent vector can be identified with a tangent

Table 13: The Riemannian operators under LEM, AIM, and PEM on the SPD manifold.

Operators	LEM	AIM	PEM
$g_P(V,W)$	$\langle \log_{*,P}(V), \log_{*,P}(W)\rangle^{(\alpha,\beta)}$	$\langle P^{-1}V, WP^{-1}\rangle^{(\alpha,\beta)}$	$\frac{1}{\theta^2}\langle \mathrm{P}_{\theta*,P}(V), \mathrm{P}_{\theta*,P}(W)\rangle^{(\alpha,\beta)}$
$\mathrm{Log}_P Q$	$(\log_{*,P})^{-1}\left[\log(Q)-\log(P)\right]$	$P^{\frac{1}{2}}\log\left(P^{-\frac{1}{2}}QP^{-\frac{1}{2}}\right)P^{\frac{1}{2}}$	$(P_{\theta*,P})^{-1}\left(Q^\theta - P^\theta\right)$
$\Gamma_{P\to Q}(V)$	$(\log_{*,Q})^{-1}\circ\log_{*,P}(V)$	$(QP^{-1})^{\frac{1}{2}}V(P^{-1}Q)^{\frac{1}{2}}$	$(\mathrm{P}_{\theta*,Q})^{-1}\circ\mathrm{P}_{\theta*,P}(V)$
$\mathrm{Exp}_P(V)$	$\exp\left(\log(P)+\log_{*,P}(V)\right)$	$P^{\frac{1}{2}}\exp\left(P^{-\frac{1}{2}}VP^{-\frac{1}{2}}\right)P^{\frac{1}{2}}$	$\left(P^\theta + P_{\theta*,P}(V)\right)^{\frac{1}{\theta}}$
Invariance	Lie group bi-invariance $\mathrm{O}(n)$-invariance	Lie group left-invariance $\mathrm{GL}(n)$-invariance	$\mathrm{O}(n)$-invariance
References	[2, 70]	[59, 68]	[24, 70, 17]

Table 14: The Riemannian operators under BWM and LCM on the SPD manifold.

Operators	LCM	BWM
$g_P(V,W)$	$\langle \lfloor\widetilde{V}\rfloor, \lfloor\widetilde{W}\rfloor\rangle + \langle \widetilde{\mathbb{V}}\widetilde{\mathbb{L}}^{-1}, \widetilde{\mathbb{W}}\widetilde{\mathbb{L}}^{-1}\rangle$	$\frac{1}{2}\langle \mathcal{L}_P[V], W\rangle$
$\mathrm{Log}_P Q$	$(\mathrm{Chol}^{-1})_{*,L}\left[\lfloor K\rfloor - \lfloor L\rfloor + \mathbb{L}\,\mathrm{Dlog}(\mathbb{L}^{-1}\mathbb{K})\right]$	$(PQ)^{\frac{1}{2}} + (QP)^{\frac{1}{2}} - 2P$
$\Gamma_{P\to Q}(V)$	$(\mathrm{Chol}^{-1})_{*,K}\left[\lfloor\widetilde{V}\rfloor + \mathbb{K}\mathbb{L}^{-1}\widetilde{\mathbb{V}}\right]$	$U\left[\sqrt{\frac{\delta_i+\delta_j}{\sigma_i+\sigma_j}}\left[U^\top VU\right]_{ij}\right]U^\top$
$\mathrm{Exp}_P(V)$	$\mathrm{Chol}^{-1}\left[\lfloor L\rfloor + \lfloor\widetilde{V}\rfloor + \mathbb{L}\,\mathrm{Dexp}(\mathbb{L}^{-1}\widetilde{V})\right]$	$P + V + \mathcal{L}_P[V]P\mathcal{L}_P[V]$
Invariance	Lie group bi-invariance	$\mathrm{O}(n)$-invariance
References	[46]	[6, 70]

vector in the horizontal space, called a horizontal lift[2]. Under this identification, each tangent vector $V \in T_P\mathrm{Gr}(p,n)$ can be represented as

$$V = P_\perp B, \text{ with } B \in \mathbb{R}^{(n-p)\times p}, \tag{30}$$

where $P_\perp \in \mathrm{St}(n-p,n)$ is the orthogonal complement of P.

Another perspective is called the Projector Perspective (PP). As shown by Bendokat et al. [5], the Grassmannian is an embedded submanifold of $\mathcal{S}^n$:

$$\widetilde{\mathrm{Gr}}(p,n) = \{P \in \mathcal{S}^n \mid P^2 = P, \mathrm{rank}(P) = p\}. \tag{31}$$

Therefore, each point can be represented as an $n \times n$ symmetric matrix. Under this perspective, any tangent vector $V \in T_P\widetilde{\mathrm{Gr}}(p,n)$ at $P \in \widetilde{\mathrm{Gr}}(p,n)$ can be represented as

$$V = Q\begin{pmatrix} 0 & B^T \\ B & 0 \end{pmatrix}Q^T, \text{ with } B \in \mathbb{R}^{(n-p)\times p}, \tag{32}$$

where $Q\widetilde{I}_{p,n}Q^\top = P$.

Supposing P and Q are the points on the Grassmannian $\mathrm{Gr}(p,n)$ ($\widetilde{\mathrm{Gr}}(p,n)$), and V and W are the tangent vectors over $T_P\mathrm{Gr}(p,n)$ ($T_P\widetilde{\mathrm{Gr}}(p,n)$), Tab. 15 summarizes the associated Riemannian operators following the notation in Tab. 10.

Remark D.2. We make the following remarks with respect to the Riemannian operators over the Grassmannian.

- **Cut locus & logarithm.** The Grassmannian Riemannian logarithm does not exist for every pair of P and Q. As shown by Bendokat et al. [5, Sec. 5], $\mathrm{Log}_P(Q)$ exists only if P and Q are not in each other's cut locus. However, this can be addressed numerically, for example by Bendokat et al. [5, Alg. 5.3] or by using the Moore–Penrose inverse for the inverse in the ONB logarithm [54].
- **PP & ONB logarithm.** The matrix logarithm shown in the PP logarithm does not support backpropagation, as it cannot be calculated by the SVD like the SPD matrix. However, the

[2]In this paper, the tangent vector under the ONB perspective is always considered as the horizontal lift.

Table 15: Riemannian operators on the Grassmannian.

Operators	$\mathrm{Gr}(p,n)$	$\widetilde{\mathrm{Gr}}(p,n)$
$g_P(V,W)$	$\langle V,W\rangle$	$\frac{1}{2}\langle V,W\rangle$
$\mathrm{Log}_P Q$	$O\arctan(\Sigma)R^\top$ $(I_n-PP^\top)Q(P^\top Q)^{-1}\overset{\text{SVD}}{:=}O\Sigma R^\top$	$\frac{1}{2}[\log\left((I_n-2Q)\left(I_n-2P\right)\right),P]$
$\Gamma_{P\to Q}(V)$	$\left(\begin{pmatrix} PR & O\end{pmatrix}\begin{pmatrix}-\sin(\Sigma)\\ \cos(\Sigma)\end{pmatrix}O^T+\left(I-OO^T\right)\right)V$ $\mathrm{Log}_P(Q)\overset{\text{SVD}}{:=}O\Sigma R^\top$	$\exp([\log_P(Q),P])V\exp(-[\log_P(Q),P])$
$\mathrm{Exp}_P V$	$\begin{pmatrix} PR & O\end{pmatrix}\begin{pmatrix}\cos(\Sigma)\\ \sin(\Sigma)\end{pmatrix}R^\top$ $V\overset{\text{SVD}}{:=}O\Sigma R^\top$	$\exp([V,P])P\exp(-[V,P])$
References	[25, 5]	[3, 5]

PP logarithm can be calculated via the ONB logarithm [56, Prop. 3.12]. The latter can be backpropagated through the SVD. In this way, the PP logarithm can be integrated into the PyTorch deep learning framework.

E Discussions on the Riemannian FC and convolutional layer

E.1 Additional discussions on the orthogonal basis

When the inner product g_E on $T_E\mathcal{M}$ is the standard inner product, we use the familiar orthonormal basis $\{e_i\}_{i=1}^m$. However, when g_E is not standard, $\{e_i\}_{i=1}^m$ might not be orthonormal. In this case, we can always find a corresponding basis associated with $\{e_i\}_{i=1}^m$ by a linear isometry. We rewrite the inner product g_E as

$$g_E(V,W)=\langle f(V),f(W)\rangle=f(V)^\top f(W),\forall V,W\in T_E\mathcal{M}\cong\mathbb{R}^m, \tag{33}$$

where f is the linear isometry that pulls back the standard inner product $\langle\cdot,\cdot\rangle$ to g_E. Then, $\{B_i\}_{i=1}^m=\{f^{-1}(e_i)\}_{i=1}^m$ is the standard orthonormal basis over $\{T_E\mathcal{M},g_E\}$.

E.2 Riemannian fully connected layers under isometric geometry

As isometric Riemannian metrics commonly arise in various geometries [69, 18, 5], we discuss the construction of Riemannian FC layers under isometries. The following theorem demonstrates that a Riemannian FC layer under isometric metrics can be computed by the following procedure: mapping, applying the Riemannian FC layer, and remapping. This result will be applied in our concrete examples of the SPD and Grassmannian FC layers.

We denote the FC transformation by $Y=\mathcal{F}(X;\mathbf{A},\mathbf{P})$, with $\mathbf{P}=\{P_i\in\mathcal{N}\}_{i=1}^m$ and $\mathbf{A}=\{A_i\in T_{P_i}\mathcal{N}\}_{i=1}^m$ as the FC parameters.

Theorem E.1 (Isometric FC Layers)**.** *Given n-dimensional Riemannian manifolds $\left\{\widetilde{\mathcal{N}},g^{\widetilde{\mathcal{N}}}\right\}$ and $\left\{\mathcal{N},g^{\mathcal{N}}\right\}$ with a Riemannian isometry $\phi^{\mathcal{N}}:\widetilde{\mathcal{N}}\to\mathcal{N}$, and m-dimensional Riemannian manifolds $\left\{\widetilde{\mathcal{M}},g^{\widetilde{\mathcal{M}}}\right\}$ and $\left\{\mathcal{M},g^{\mathcal{M}}\right\}$ with $\phi^{\mathcal{M}}:\widetilde{\mathcal{M}}\to\mathcal{M}$ as a Riemannian isometry mapping the origin $E^{\widetilde{\mathcal{M}}}\in\widetilde{\mathcal{M}}$ to the origin $E\in\mathcal{M}$, the Riemannian FC layer $\widetilde{\mathcal{F}}:\widetilde{\mathcal{N}}\to\widetilde{\mathcal{M}}$ can be calculated by $\mathcal{F}:\mathcal{N}\to\mathcal{M}$:*

$$\widetilde{\mathcal{F}}\left(\widetilde{X};\widetilde{\mathbf{P}},\widetilde{\mathbf{A}}\right)=\left(\phi^{\mathcal{M}}\right)^{-1}\left(\mathcal{F}\left(\phi^{\mathcal{N}}(\widetilde{X});\mathbf{P},\mathbf{A}\right)\right), \tag{34}$$

where $\widetilde{\mathbf{P}}=\left\{\widetilde{P}_i\in\widetilde{\mathcal{N}}\right\}_{i=1}^m$ and $\widetilde{\mathbf{A}}=\left\{\widetilde{A}_i\in T_{\widetilde{P}_i}\widetilde{\mathcal{N}}\right\}_{i=1}^m$ are the FC parameters of $\widetilde{\mathcal{F}}$, while $\mathbf{P}=\left\{\phi^{\mathcal{N}}(\widetilde{P}_i)\right\}_{i=1}^m$ and $\mathbf{A}=\left\{\phi^{\mathcal{N}}_{*,\widetilde{P}_i}(\widetilde{A}_i)\right\}_{i=1}^m$ are the FC parameters of $\mathcal{F}$.

Proof. First, we show the correspondence between the standard orthonormal bases $\{\widetilde{B}_i\in\widetilde{\mathcal{M}}\}$ and $\{B_i\in\mathcal{M}\}$. The set $\{\widetilde{B}_i\in\widetilde{\mathcal{M}}\}$ is orthonormal iff $\{B_i\in\mathcal{M}\}$ is orthonormal. We only need to show

standardness. The Riemannian metric $g^{\widetilde{\mathcal{M}}}$ satisfies

$$\begin{aligned} g^{\widetilde{\mathcal{M}}}_{\widetilde{E}}(V,W) &\overset{(1)}{=} g^{\mathcal{M}}_{E}\left(\phi^{\mathcal{M}}_{*,\widetilde{E}}(V), \phi^{\mathcal{M}}_{*,\widetilde{E}}(V)\right) \\ &= \left\langle f \circ \phi^{\mathcal{M}}_{*,\widetilde{E}}(V), f \circ \phi^{\mathcal{M}}_{*,\widetilde{E}}(V) \right\rangle, \end{aligned} \tag{35}$$

where f is the linear isomorphism that pulls back the standard Frobenius inner product to $g^{\mathcal{M}}_{E}$. Here, (1) comes from the isometry. Therefore, for each i, we have

$$\begin{aligned} \widetilde{B}_i &= (f \circ \phi^{\mathcal{M}}_{*,\widetilde{E}})^{-1}(E_i) \\ &\overset{(1)}{=} \left(\phi^{\mathcal{M}}_{*,\widetilde{E}}\right)^{-1}(B_i), \end{aligned} \tag{36}$$

where (1) comes from $B_i = f^{-1}(E_i), \forall i = 1, \cdots, n$.

We now demonstrate the correspondence between the FC layers as follows:

$$\begin{aligned} Y &= \mathrm{Exp}^{\widetilde{\mathcal{M}}}_{\widetilde{E}}\left(\sum_{i=1}^{m}\left(\langle \mathrm{Log}^{\widetilde{\mathcal{N}}}_{\widetilde{P}_i}(\widetilde{X}), \widetilde{A}_i\rangle^{\widetilde{\mathcal{N}}}_{\widetilde{P}_i}\widetilde{B}_i\right)\right) \\ &\overset{(1)}{=} \left(\phi^{\mathcal{M}}\right)^{-1}\left(\mathrm{Exp}^{\mathcal{M}}_{E}\left(\phi^{\mathcal{M}}_{*,\widetilde{E}}\left[\sum_{i=1}^{m}\left(\langle \mathrm{Log}^{\mathcal{N}}_{P_i}(X), A_i\rangle^{\mathcal{N}}_{P_i}\widetilde{B}_i\right)\right]\right)\right) \\ &\overset{(2)}{=} \left(\phi^{\mathcal{M}}\right)^{-1}\left(\mathrm{Exp}^{\mathcal{M}}_{E}\left(\sum_{i=1}^{m}\left(\langle \mathrm{Log}^{\mathcal{N}}_{P_i}(X), A_i\rangle^{\mathcal{N}}_{P_i}B_i\right)\right)\right), \end{aligned} \tag{37}$$

where $B_i = \phi^{\mathcal{M}}_{*,\widetilde{E}}(\widetilde{B}_i)$, $A_i = \phi^{\mathcal{N}}_{*,\widetilde{P}_i}(\widetilde{A}_i)$, $X = \phi^{\mathcal{N}}(\widetilde{X})$, and $P_i = \phi^{\mathcal{N}}(\widetilde{P}_i)$. The above derivation comes from the following.

(1) The isometry of $\phi^{\mathcal{M}}$ and $\phi^{\mathcal{N}}$;
(2) The linearity of $\phi^{\mathcal{M}}_{*,\widetilde{E}}$.

□

E.3 Riemannian fully connected layers under product geometry

Now, we discuss Thm. 3.3 under product geometry.

Theorem E.2. *Following the notation in Thm. 3.3, the Riemannian FC layer $\mathcal{F}(\cdot) : (\mathcal{N})^c \to \mathcal{M}$ for the input $(X_1 \in \mathcal{N}, \cdots, X_c \in \mathcal{N}) = X \in (\mathcal{N})^c$ is*

$$Y = \mathrm{Exp}^{\mathcal{M}}_{E}\left(\sum_{i=1}^{m}\sum_{j=1}^{c}\langle \mathrm{Log}^{\mathcal{N}}_{P_{ij}}(X), A_{ij}\rangle^{\mathcal{N}}_{P_{ij}}B_i\right), \tag{38}$$

where $P_{ij} \in \mathcal{N}$ and $A_{ij} \in T_{P_{ij}}\mathcal{N}$ are the FC parameters.

Proof. By product geometry, we have

$$(\mathcal{N})^c \ni P_i = (P_{i1} \in \mathcal{N}, \cdots, P_{ic} \in \mathcal{N}), \tag{39}$$

$$T_{P_i}(\mathcal{N})^c \ni A_i = (A_{i1} \in T_{P_{i1}}\mathcal{N}, \cdots, A_{ic} \in T_{P_{ic}}\mathcal{N}). \tag{40}$$

The above implies that

$$\langle \mathrm{Log}^{(\mathcal{N})^c}_{P_i}(X), A_i\rangle^{(\mathcal{N})^c}_{P_i} = \sum_{j=1}^{c}\langle \mathrm{Log}^{\mathcal{N}}_{P_{ij}}(X), A_{ij}\rangle^{\mathcal{N}}_{P_{ij}}. \tag{41}$$

□

E.4 Riemannian fully connected layers and manifold embedding

In several applications [12, 47, 81, 56], embedding Euclidean features into non-Euclidean manifolds often yields superior results. A common approach can be expressed as $\mathrm{Exp}_E(Ax+b)$, which maps Euclidean features to the tangent space at the origin via a linear layer, followed by applying the exponential map at the origin. This method has been adopted in various embeddings, including hyperbolic [12, 27], SPD [81], and Grassmannian spaces [56, Sec. 3.4.2]. Our framework offers a novel intrinsic interpretation, showing that this operation respects the Riemannian FC layer between the Euclidean space and the target manifold.

Proposition E.3. *The Riemannian FC layer from a standard Euclidean space $\mathbb{R}^n$ to an m-dimensional target manifold $\mathcal{M}$, namely $\mathcal{F}(\cdot):\mathbb{R}^n\to\mathcal{M}$, is given by*

$$\mathcal{F}(x)=\mathrm{Exp}_E(Ax+b), \tag{42}$$

where $A\in\mathbb{R}^{n\times m}$ and $b\in\mathbb{R}^m$ are the transformation matrix and bias vector, respectively.

Proof. By Thm. 3.3, we have the following

$$\begin{aligned}
Y &\overset{(1)}{=} \mathrm{Exp}_E^{\mathcal{M}}\left(\sum_{i=1}^{m}\left(\langle \mathrm{Log}_{p_i}^{\mathrm{Euc}}(x),a_i\rangle_{p_i}^{\mathrm{Euc}}B_i\right)\right),\\
&\overset{(2)}{=} \mathrm{Exp}_E^{\mathcal{M}}\left(\sum_{i=1}^{m}\left(\langle x-p_i,a_i\rangle B_i\right)\right),\\
&\overset{(3)}{=} \mathrm{Exp}_E^{\mathcal{M}}\left(\sum_{i=1}^{m}\left(\langle x-p_i,a_i\rangle f^{-1}(e_i)\right)\right),\\
&\overset{(4)}{=} \mathrm{Exp}_E^{\mathcal{M}}\left(f^{-1}\left(\sum_{i=1}^{m}\langle x-p_i,a_i\rangle e_i\right)\right),\\
&\overset{(5)}{=} \mathrm{Exp}_E^{\mathcal{M}}\left(f^{-1}\left(\bar{A}x+\bar{b}\right)\right),\\
&\overset{(6)}{=} \mathrm{Exp}_E^{\mathcal{M}}\left(Ax+b\right).
\end{aligned} \tag{43}$$

The above follows from the following.

(1) $p_i,a_i\in\mathbb{R}^n$, and $\{B_i\}$ is an orthonormal basis over $\{T_E\mathcal{M},g_E\}$;
(2) The Euclidean logarithm and metric become the familiar vector operation:

$$\begin{aligned}
\mathrm{Log}_{p_i}^{\mathrm{Euc}}(x)&=x-p_i\\
\langle v,w\rangle_p^{\mathrm{Euc}}&=\langle v,w\rangle,\forall p\in\mathbb{R}^n,\forall v,w\in T_p\mathbb{R}^n;
\end{aligned}$$

(3) f is the linear isomorphism pulling the standard inner product back to g_E; $\{e_i\}$ is the standard orthonormal basis over the standard inner product;
(4) Linearity of f^{-1};
(5) $\sum_{i=1}^m\langle x-p_i,a_i\rangle e_i$ has the form of an affine transformation;
(6) As f^{-1} has matrix representation, $f^{-1}(x)=\tilde{A}x$, we have

$$\begin{aligned}
f^{-1}\left(\bar{A}x+\bar{b}\right)&=\tilde{A}\left(\bar{A}x+\bar{b}\right)\\
&=\tilde{A}\bar{A}x+\tilde{A}\bar{b}.
\end{aligned} \tag{44}$$

Setting $A=\tilde{A}\bar{A}$ and $b=\tilde{A}\bar{b}$, one can obtain the result.

□

E.5 Relation with the convolution in ManifoldNet

Chakraborty et al. [11] also proposed a convolution operation for manifolds. However, since its formulation is based on the weighted Fréchet mean, it is unable to alter the manifold dimension, for example through dimensionality reduction. In contrast, our framework allows modifications in both the channel and manifold dimensions, providing greater flexibility.

F Comparison of our hyperbolic FC layers against previous ones

Tab. 16 extends Tab. 1 by comparing our hyperbolic FC layers against previous hyperbolic linear layers.

Table 16: Comparison of hyperbolic linear layers. Here, we consider the transformation from an n-dimensional hyperbolic space to an m-dimensional one.

Method	Model	Mechanism	Formulation	Parameters	References
Möbius	$\mathbb{P}^n_K$	Tangent	$\operatorname{Exp}_{\mathbf{0}}(M \operatorname{Log}_{\mathbf{0}}(x))$	$M \in \mathbb{R}^{m \times n}$	[28, Def. 3.2]
Klein	$\mathbb{K}^n_K$	Tangent	$\operatorname{Exp}_{\mathbf{0}}(M \operatorname{Log}_{\mathbf{0}}(x))$	$M \in \mathbb{R}^{m \times n}$	[51, Thm. 9]
LFC	$\mathbb{H}^n_K$	Spacetime	$\begin{bmatrix} \frac{\sqrt{\|Wx\|^2 - 1/K}}{v^\top x} v^\top \\ W \end{bmatrix} x$	$M \in \mathbb{R}^{m \times (n+1)}$ $v \in \mathbb{R}^{n+1}$	[14, Sec 3.1]
NestFC	$\mathbb{H}^n_K$	Nested projection	$y = \frac{Wx}{\|Wx\|_L}, \quad W J_n W^\top = J_m$	$Q \in \mathrm{SO}(n)$ $\widetilde{P} \in \mathrm{St}(m, n)$ $\alpha \in \mathbb{R}$	[26, Eq. (14) and Sec. 3.3]
Poincaré FC	$\mathbb{P}^n_K$	Poincaré	$w \left(1 + \sqrt{1 - K\|w\|^2}\right)^{-1}$ $w = \left((-K)^{-\frac{1}{2}} \sinh\left(\sqrt{-K} v_k(x)\right)\right)_{k=1}^{m}$ v_k is defined by Shimizu et al. [66, Eq. (6)]	$\{z_i \in \mathbb{R}^n\}_{i=1}^m$ $\{\gamma_i \in \mathbb{R}\}_{i=1}^m$	[66, Sec. 3.2]
Ours	$\mathbb{P}^n_K, \mathbb{K}^n_K, \mathbb{H}^n_K$	Riemannian	Thms. 4.1 and 4.2	$\{z_i \in \mathbb{R}^n\}_{i=1}^m$ $\{\gamma_i \in \mathbb{R}\}_{i=1}^m$	Thms. 4.1 and 4.2

G Additional details on the SPD fully connected layers

G.1 Relation with the gyro SPD fully connected layers

This subsection demonstrates that our SPD FC layers subsume three gyro SPD FC layers under LEM, AIM, and LCM. This follows directly from Prop. 3.2, as one can readily verify that the point-to-hyperplane pseudo-distance we used is identical to the corresponding pseudo-gyrodistances under these three metrics. To clarify this relationship more clearly, we compare the final expressions.

We first review some related SPD gyro structures [55]. Given P, Q in $\{\mathcal{S}^n_{++}, g\}$ with g being AIM, LEM, or LCM, and $t \in \mathbb{R}$, the gyro structures induced by g are defined as follows:

$$\text{Gyro addition: } P \oplus Q = \operatorname{Exp}_P \left(\Gamma_{I \to P} \left(\operatorname{Log}_I(Q)\right)\right), \tag{45}$$

$$\text{Scalar gyromultiplication: } t \otimes P = \operatorname{Exp}_I \left(t \operatorname{Log}_I(P)\right), \tag{46}$$

$$\text{Gyro inverse: } \ominus P = -1 \otimes P = \operatorname{Exp}_I \left(-\operatorname{Log}_I(P)\right), \tag{47}$$

$$\text{Gyro inner product: } \langle P, Q \rangle_{\mathrm{gr}} = \langle \operatorname{Log}_I(P), \operatorname{Log}_I(Q) \rangle_I, \tag{48}$$

where Log_I and $\langle \cdot, \cdot \rangle_I$ are the Riemannian logarithm and metric at the identity matrix I. As shown by Nguyen [54], the gyro addition and scalar product under AIM, LEM, and LCM form gyrovector spaces.

Based on these gyro structures, Nguyen et al. [56] introduced the gyro SPD FC layers under AIM, LEM, and LCM, respectively. We review their results in the following.

Theorem G.1 (Gyro SPD FC Layers [56])**.** *The gyro SPD FC layers under standard LEM, AIM, and LCM are*

$$\textit{LEM}: Y = \exp\left(V^{\mathrm{LE}}\right), V^{\mathrm{LE}}_{ij} = \begin{cases} v^{\mathrm{LE}}_{ii}(S), & \textit{if } i = j \\ \frac{1}{\sqrt{2}} v^{\mathrm{LE}}_{ij}(S), & \textit{if } i > j \\ V^{\mathrm{LE}}_{ji}, & \textit{otherwise} \end{cases} \tag{49}$$

$$\textit{AIM}: Y = \exp\left(V^{\mathrm{AI}}\right), V^{\mathrm{AI}}_{ij} = \begin{cases} v^{\mathrm{AI}}_{ii}(S) + \eta \sum_{k=1}^{m} v^{\mathrm{AI}}_{kk}(S), & \textit{if } i = j \\ \frac{1}{\sqrt{2}} v^{\mathrm{AI}}_{ij}(S), & \textit{if } i > j \\ V^{\mathrm{AI}}_{ji}, & \textit{otherwise} \end{cases} \tag{50}$$

$$\textit{LCM}: Y = V^{\mathrm{LC}} (V^{\mathrm{LC}})^\top, V^{\mathrm{LC}}_{ij} = \begin{cases} \exp\left(v^{\mathrm{LC}}_{ii}(S)\right), & \textit{if } i = j \\ v^{\mathrm{LC}}_{ij}(S), & \textit{if } i > j \\ 0, & \textit{otherwise} \end{cases} \tag{51}$$

where $\eta = \frac{1}{n}\left(\frac{1}{\sqrt{1+n\beta}} - 1\right)$*, and* $v_{ij}^g = \langle \ominus P_{ij} \oplus S, W_{ij} \rangle_{\mathrm{gr}}$ *with* g *as LEM, AIM, or LCM. Here,* $P_{ij}, W_{ij} \in \mathcal{S}_{++}^n, \forall i \geq j, i, j = 1, \cdots, m$.

Proposition G.2. *Our LEM (*$(\alpha, \beta) = (1, 0)$*), AIM (*$(\alpha, \beta) = (1, \beta)$*), and LCM SPD FC layers incorporate the LEM, AIM, and LCM gyro SPD FC layers, respectively.*

Proof. Comparing Thm. G.1 with our Thm. 4.4, we only need to show the equality of v_{ij} in the gyro and our framework:

$$v_{ij}^g \overset{(1)}{=} \left\langle \mathrm{Log}_{P_{ij}}(S), \Gamma_{I \to P_{ij}}\left(\mathrm{Log}_I(W_{ij})\right) \right\rangle_{P_{ij}}, \tag{52}$$

where (1) has been proved in Prop. 3.2. Setting $A_{ij} = \Gamma_{I \to P}\left(\mathrm{Log}_I(W_{ij})\right) \in T_{P_{ij}}\mathcal{S}_{++}^n$, we recover Eqs. (99), (100) and (102) for each metric. □

G.2 Relation with the flat SPD fully connected layers

Nguyen et al. [57] proposed two SPD FC layers based on flat LEM and LCM. However, as shown by Nguyen et al. [57, App. B. 2.2], they have the same formulations as the LEM and LCM gyro SPD FC layers, respectively.

G.3 Trivialized SPD fully connected layers

Theorem G.3 (Trivialized SPD FC Layers)**.** *Trivializing each* P_{ij} *in Thm. 4.4 as* $\mathrm{Exp}_I(\gamma_{ij}[Z_{ij}])$*,* $v_{ij}(S)$ *under different metrics can be further simplified:*

$$\textit{LEM}: \langle \log(S), Z_{ij} \rangle^{(\alpha,\beta)} - \gamma_{ij} \left\| Z_{ij} \right\|^{(\alpha,\beta)}, \tag{53}$$

$$\textit{AIM}: \left\langle \log\left(\exp\left(-\frac{\gamma_{ij}}{2}[Z_{ij}]\right) S \exp\left(-\frac{\gamma_{ij}}{2}[Z_{ij}]\right)\right), Z_{ij} \right\rangle^{(\alpha,\beta)}, \tag{54}$$

$$\textit{PEM}: \left\langle S^\theta - (I + \theta\gamma_{ij}[Z_{ij}]), Z_{ij} \right\rangle^{(\alpha,\beta)}, \tag{55}$$

$$\textit{LCM}: \left\langle \lfloor K \rfloor + \mathrm{Dlog}(\mathbb{K}) - \left(\gamma_{ij}\lfloor [Z_{ij}] \rfloor + \frac{1}{2}\gamma_{ij}\mathbb{D}([Z_{ij}])\right), \lfloor Z_{ij} \rfloor + \frac{1}{2}\mathbb{Z}_{ij} \right\rangle, \tag{56}$$

where $\|\cdot\|^{(\alpha,\beta)}$ *is the norm induced by* $\langle \cdot, \cdot \rangle^{(\alpha,\beta)}$*, and* $\mathbb{D}(\cdot)$ *returns a diagonal matrix with diagonal elements from the input square matrix.*

Proof. **LEM:**

$$\begin{aligned} \langle \log(S) - \log(P_{ij}), Z_{ij} \rangle^{(\alpha,\beta)} &\overset{(1)}{=} \langle \log(S) - \gamma_{ij}[Z_{ij}], Z_{ij} \rangle^{(\alpha,\beta)} \\ &\overset{(2)}{=} \langle \log(S), Z_{ij} \rangle^{(\alpha,\beta)} - \gamma_{ij} \left\| Z_{ij} \right\|^{(\alpha,\beta)}, \end{aligned} \tag{57}$$

The above comes from the following.

(1) Eq. (115);
(2) $[Z_{ij}] = \frac{Z_{ij}}{\|Z_{ij}\|^{(\alpha,\beta)}}$.

AIM: This can be obtained by the following:

$$\exp\left(\gamma_{ij}[Z_{ij}]\right)^{-\frac{1}{2}} = \exp\left(-\frac{\gamma_{ij}}{2}[Z_{ij}]\right). \tag{58}$$

PEM: This can be obtained by Eq. (116).

LCM:

$$\begin{aligned} &\left\langle \lfloor K \rfloor - \lfloor L_{ij} \rfloor + \mathrm{Dlog}(\mathbb{K}\mathbb{L}_{ij}^{-1}), \lfloor Z_{ij} \rfloor + \frac{1}{2}\mathbb{Z}_{ij} \right\rangle \\ &= \left\langle \lfloor K \rfloor + \mathrm{Dlog}(\mathbb{K}) - \left(\lfloor L_{ij} \rfloor + \mathrm{Dlog}(\mathbb{L}_{ij})\right), \lfloor Z_{ij} \rfloor + \frac{1}{2}\mathbb{Z}_{ij} \right\rangle \\ &\overset{(1)}{=} \left\langle \lfloor K \rfloor + \mathrm{Dlog}(\mathbb{K}) - \left(\gamma_{ij}\lfloor [Z_{ij}] \rfloor + \frac{1}{2}\gamma_{ij}\mathbb{D}([Z_{ij}])\right), \lfloor Z_{ij} \rfloor + \frac{1}{2}\mathbb{Z}_{ij} \right\rangle, \end{aligned} \tag{59}$$

where (2) comes from Eq. (117). □

Remark G.4. Due to the incompleteness of PEM and BWM, their exponential maps at I, $\mathrm{Exp}_I(V)$, are well-defined locally:

$$\begin{aligned} &\text{PEM: } I + \theta V \in \mathcal{S}_{++}^n, \\ &\text{BWM: } I + \frac{1}{2} V \in \mathcal{S}_{++}^n. \end{aligned} \tag{60}$$

The above restriction can be addressed numerically, for example by ReEig [35]:

$$\widetilde{S} = U \max(\epsilon I, \Sigma) U^\top, \tag{61}$$

where $S \overset{\text{Eig}}{:=} U \Sigma U^\top$ is the eigendecomposition.

G.4 Trivialized SPD multinomial logistic regression

In our implementation, we trivialize the SPD parameters in the SPD MLR as in Sec. 3.3. The SPD MLRs proposed by Chen et al. [17] under five geometries can be further simplified. For simplicity, we do not involve the power deformation [17].

Theorem G.5 (Trivialized SPD MLRs)**.** *Given C classes and an SPD feature S, the SPD MLRs, $p(y = k \mid S \in \mathcal{S}_{++}^n)$, are proportional to*

$$\textit{LEM}: \exp\left[\langle \log(S), Z_k \rangle^{(\alpha,\beta)} - \gamma_k \left\| Z_k \right\|^{(\alpha,\beta)}\right], \tag{62}$$

$$\textit{AIM}: \left[\exp\left\langle \log\left(\exp\left(-\frac{\gamma_k}{2}[Z_k]\right) S \exp\left(-\frac{\gamma_k}{2}[Z_k]\right)\right), Z_k \right\rangle^{(\alpha,\beta)}\right], \tag{63}$$

$$\textit{PEM}: \frac{1}{\theta} \exp\left[\left\langle S^\theta - (I + \theta\gamma_k[Z_k]), Z_k \right\rangle^{(\alpha,\beta)}\right], \tag{64}$$

$$\textit{LCM}: \exp\left[\left\langle \lfloor K \rfloor + \mathrm{Dlog}(\mathbb{K}) - \left(\gamma_k \lfloor [Z_k] \rfloor + \frac{1}{2}\gamma_k \mathbb{D}([Z_k])\right), \lfloor Z_k \rfloor + \frac{1}{2}\mathbb{Z}_k \right\rangle\right], \tag{65}$$

$$\textit{BWM:} \exp\left[\frac{1}{2}\left\langle (P_k S)^{\frac{1}{2}} + (S P_k)^{\frac{1}{2}} - 2P_k, \mathcal{L}_{P_k}(L_k Z_k L_k^\top) \right\rangle\right], \tag{66}$$

where $Z_k \in T_I \mathcal{S}_{++}^n \backslash \{0\}$ is a symmetric matrix, $L_k = \mathrm{Chol}(P_k)$ is the Cholesky factor of P_k with $P_k = (I + \frac{1}{2}\gamma_k[Z_k])^2$. Here $\{Z_k \in \mathcal{S}^n\}_{k=1}^C$ and $\{\gamma_k \in \mathbb{R}\}_{k=1}^C$ are the MLR parameters.

Proof. For each class k, the expression of v_k in the SPD MLR [17, Thm. 4.2] has been reviewed in App. J.6. For MLR under each metric g, we parameterize each parameter $P_k \in \mathcal{S}_{++}^n$ by Z_k and γ_k by

$$P_k = \mathrm{Exp}_I^g(\gamma_k[Z_k]), \tag{67}$$

with $[Z_k]$ as the unit vector of Z_k. Under this parameterization, the MLRs under LEM, AIM, PEM, and LCM can be further simplified, which has been implied by Thm. G.3. □

Remark G.6. Similar to the SPD FC layer, due to the incompleteness of PEM and BWM, the associated parameterization should follow

$$\text{PEM: } I + \theta\gamma_k[Z_k] \in \mathcal{S}_{++}^n, \tag{68}$$

$$\text{BWM: } I + \frac{1}{2}\gamma_k[Z_k] \in \mathcal{S}_{++}^n. \tag{69}$$

G.5 Covariant equivariance

Equivariance in ManifoldNet [11]. For a learned scalar kernel w with positive weights that sum to one, ManifoldConv [11, Eq. (8)] is defined as

$$(f * w)(y) = \underset{Z \in \mathcal{M}}{\mathrm{argmin}} \sum_{x \in \mathcal{K}_y} w(x - y) d_{\mathcal{M}}^2(f(x), Z), \tag{70}$$

where $\mathcal{K}_y$ is the receptive field centered at y. Since an isometry preserves Riemannian distances and hence commutes with the weighted Fréchet mean, ManifoldConv satisfies the fixed-parameter equivariance

$$((\phi \circ f) * w)(y) = \phi((f * w)(y)). \tag{71}$$

Equivariance of our layers. Our FC and convolutional layers are defined through Riemannian operators that are compatible with isometries. In contrast to ManifoldNet's fixed-parameter equivariance, they admit parameter-covariant equivariance. Since our convolutional layer is composed of local FC transformations, we only consider the FC layer below for notational convenience. Consider the equal-manifold and equal-dimension setting $\mathcal{N} = \mathcal{M}$ in Thm. 3.3. For an isometry $\phi : \mathcal{M} \to \mathcal{M}$, we have

$$\mathcal{F}_{A,P,B,E}(\phi(X)) = \operatorname{Exp}_E \left(\sum_{i=1}^{m} \left\langle \operatorname{Log}_{P_i}(\phi(X)), A_i \right\rangle_{P_i} B_i \right) = \phi \left(\mathcal{F}_{\bar{A},\bar{P},\bar{B},\bar{E}}(X) \right), \tag{72}$$

with

$$\bar{P}_i = \phi^{-1}(P_i), \quad \bar{A}_i = (\phi^{-1})_{*,P_i}(A_i), \quad \bar{E} = \phi^{-1}(E), \quad \bar{B}_i = (\phi^{-1})_{*,E}(B_i). \tag{73}$$

Since ϕ is an isometry, $\{\bar{B}_i\}_{i=1}^m$ remains an orthonormal basis of $T_{\bar{E}}\mathcal{M}$. If $\phi(E) = E$ further fixes the origin E, and the equivariance identity simplifies to

$$\mathcal{F}_{A,P}(\phi(X)) = \phi \left(\mathcal{F}_{\bar{A},\bar{P}}(X) \right). \tag{74}$$

H Review of previous Grassmannian transformation layers

This section briefly reviews several popular Grassmannian transformation layers.

FRMap + ReOrth. Given an input Grassmannian $X \in \operatorname{Gr}(p, q)$, Huang et al. [36] used Full Rank Map (FRMap) to transform the input orthonormal matrices of subspaces into new matrices through a linear mapping function, and then applied QR decomposition to recover orthogonality:

$$Y = \mathcal{Q}(WX), \tag{75}$$

where $W \in \mathbb{R}^{m \times n}$ is a row-wise orthogonal parameter, and $\mathcal{Q}(\cdot)$ returns the orthogonal matrix in the QR decomposition.

PP & ONB Scaling. Nguyen [54], Nguyen and Yang [55] proposed matrix scaling for the PP and ONB Grassmannian, respectively. Given $P = XX^\top \in \widetilde{\operatorname{Gr}}(p, n)$ with $X \in \operatorname{Gr}(p, n)$, the operations are defined as

$$\textbf{PP: } Y = \exp \left(\begin{bmatrix} 0 & W * B \\ -(W * B)^T & 0 \end{bmatrix} \right) \widetilde{I}_{p,n} \exp \left(- \begin{bmatrix} 0 & W * B \\ -(W * B)^T & 0 \end{bmatrix} \right), \tag{76}$$

$$\textbf{ONB: } Y = \exp \left(\begin{bmatrix} 0 & W * B \\ -(W * B)^T & 0 \end{bmatrix} \right) I_{p,n}, \tag{77}$$

where $*$ denotes the Hadamard product and $B \in \mathbb{R}^{(n-p) \times p}$ is a Euclidean parameter. Here, $X = \exp \left(\begin{bmatrix} 0 & B \\ -B^T & 0 \end{bmatrix} \right) I_{p,n}$.

GrTrans. Nguyen and Yang [55] adopted Grassmannian gyrogroup translation (GrTrans) to transform the ONB and PP Grassmannian features. Given $X \in \widetilde{\operatorname{Gr}}(p, n)$ (or $X \in \operatorname{Gr}(p, n)$), the operation is defined as

$$Y = W \oplus X, \tag{78}$$

where $\oplus$ is the Grassmannian PP (ONB) gyro addition [55, Sec. 2.3], and $W \in \widetilde{\operatorname{Gr}}(p, n)$ (or $W \in \operatorname{Gr}(p, n)$) is a Grassmannian parameter.

I Additional experimental details and results

I.1 Hyperbolic spaces

I.1.1 Datasets

Disease [1]. It represents a disease propagation tree, simulating the SIR disease transmission model, with each node representing either an infection or a non-infection state.

Airport [80]. It is a transductive dataset where nodes represent airports and edges represent airline routes from OpenFlights.org.

Pubmed [53]. This is a standard benchmark describing citation networks where nodes represent scientific papers in the area of medicine, edges are citations between them, and node labels are academic (sub)areas.

Cora [64]. It is a citation network where nodes represent scientific papers in the area of machine learning, edges are citations between them, and node labels are academic (sub)areas.

I.1.2 Implementation details

We follow the official implementations of HNN[3] [28], HNN++[4] [66], and HyboNet [5] [14] to conduct the experiments. For the Einstein transformation in the Beltrami–Klein model, we carefully implement it according to the original paper [51]. We adopt the HGCN[6] settings [12] for the link prediction task.

Details on main experiments. Following the HNN implementations [28, 12, 51], the baseline encoder consists of two transformation layers: the first maps the input feature dimension to 16, and the second maps 16 to 16. The transformation layers can be our HFC layers or alternatives such as Möbius, Einstein, Poincaré FC, LorentzTan, or LFC. On Disease, Airport, and Pubmed, each transformation is followed by an activation layer $\mathrm{Exp}_o(\mathrm{ReLU}(\mathrm{Log}_o(x)))$, where o is the origin in each model. On Cora, no activation layer is used for any method. Following HNN, we also adopt the bias translation after each HFC layer, *i.e.*, $x \oplus b = \mathrm{Exp}_x(\Gamma_{o\to x}\,\mathrm{Log}_o(b))$, except NestFC. We use the Adam optimizer [40] with a learning rate of $1e^{-2}$. Within each dataset, all methods use the same hyperparameters except for weight decay and dropout, which are tuned for each method. The weight decay and dropout configurations for our HFC models are reported in Tab. 17.

Table 17: Weight decay and dropout configurations for the HFC models. Each entry reports weight decay / dropout.

Method	Disease	Airport	Pubmed	Cora
HFC-P	$1e^{-5}/0$	$1e^{-3}/0.1$	$1e^{-5}/0$	$1e^{-4}/0$
HFC-K	$5e^{-5}/0$	$1e^{-3}/0$	$0/0$	$5e^{-5}/0.1$
HFC-H	$0/0$	$1e^{-3}/0$	$1e^{-5}/0.1$	$1e^{-3}/0.1$

Details on ablations on the RResNet. We employ a hyperbolic transformation layer to map each input vector into an 8-dimensional vector in the Poincaré ball. The network consists of two residual blocks, each configured with different hidden dimensions and varying numbers of horospheres. We use the Adam optimizer [40] and fine-tune hyperparameters, such as the learning rate and weight decay.

I.1.3 Complexity and parameter analysis

We compare the computational complexity and parameter dimensions of the hyperbolic FC layers.

Analysis. We consider a hyperbolic FC layer that maps a batch of B samples from an n-dimensional hyperbolic space to an m-dimensional hyperbolic space in the reduction setting $m \le n$. NestFC uses two matrix-manifold parameters, $Q \in \mathrm{SO}(n)$ and $\widetilde{P} \in \mathrm{St}(m, n)$ [26, Sec. 3.3], which introduce additional complexity and parameters.

- **Asymptotic complexity.** As summarized in Tab. 18, all other FC layers have complexity $O(Bnm)$, whereas NestFC has the highest complexity, $O(n^3 + Bn^2)$. Its additional $O(n^3)$ term comes from the matrix exponentials used to construct its matrix-manifold parameters.
- **Parameter dimension.** As summarized in Tab. 18, NestFC has the largest and fastest-growing parameter dimension among the compared FC layers when m is fixed and n increases. Specifically, Q contributes $n(n-1)/2$ dimensions and $\widetilde{P}$ contributes $mn - m(m+1)/2$ dimensions.

[3] https://github.com/dalab/hyperbolic_nn

[4] https://github.com/mil-tokyo/hyperbolic_nn_plusplus

[5] https://github.com/chenweize1998/fully-hyperbolic-nn

[6] https://github.com/HazyResearch/hgcn

Table 18: Forward complexity and parameter dimensions of an n-to-m hyperbolic FC layer for a batch of B samples in the reduction setting $m \leq n$. The worst entry in each metric is underlined.

Method	Space	Complexity	Parameter dimension
Möbius [28]	$\mathbb{P}^n_K$	$O(Bnm)$	nm
Einstein [51]	$\mathbb{K}^n_K$	$O(Bnm)$	nm
LorentzTan	$\mathbb{H}^n_K$	$O(Bnm)$	nm
LFC [14]	$\mathbb{H}^n_K$	$O(Bnm)$	$m(n+1)+m+(n+1)+2$
NestFC [26]	$\mathbb{H}^n_K$	$\underline{O(n^3+Bn^2)}$	$\underline{\frac{n(n-1)}{2}+mn-\frac{m(m+1)}{2}+1}$
Poincaré FC [66]	$\mathbb{P}^n_K$	$O(Bnm)$	$nm+m$
HFC-P	$\mathbb{P}^n_K$	$O(Bnm)$	$nm+m$
HFC-K	$\mathbb{K}^n_K$	$O(Bnm)$	$nm+m$
HFC-H	$\mathbb{H}^n_K$	$O(Bnm)$	$nm+m$

Table 19: Hyperparameter-search procedure.

Step	Hyperparameter	Candidates
1	Feature normalization (`normalize_feats`)	$0, 1$
2	Activation	None, ReLU, LeakyReLU, ELU, Tanh
3	Learning rate	$0.001, 0.005, 0.008, 0.01, 0.02$
4	Gyro bias	$0, 1$ for HFC-H only. NestFC skips this step
5	Gradient clipping	None, $0.5, 1.0$
6	Attention and local aggregation	$(0,0), (1,1)$ for (`use_att`, `local_agg`)
7	Weight decay	$0, 0.0001, 0.0005, 0.001, 0.002, 0.005, 0.01, 0.05, 0.1$
8	Dropout	$0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$

I.1.4 Comparison under the NHGCN architecture

The main experiments compare different hyperbolic FC layers within the HNN framework, and we further compare HFC-H and NestFC within the NHGCN [26] framework, which is based on the hyperboloid model.

Setting. We replace only NestFC in NHGCN with HFC-H for feature transformation [26, Sec. 3.2]. All other NHGCN modules remain unchanged. Both methods use the same two-layer NHGCN architecture with a hidden dimension of 16 and identical input and output dimensions in each layer. For link prediction, the public paper and code[7] specify neither the hyperparameter-search procedure nor the final dataset-specific hyperparameters. For a fair comparison, we conduct two rounds of hyperparameter search for both methods, as shown in Tab. 19.

Results. Table 20 compares the best HFC-H and NestFC results.

- **Better performance.** HFC-H exceeds NestFC by 19.09, 1.65, and 1.48 AUC points on Disease, Airport, and Pubmed, respectively. On Cora, the two selected outcomes differ by only 0.26 point.
- **Lower resource use.** Although both methods use the same two-layer architecture and identical input and output dimensions in every layer, HFC-H uses fewer trainable parameters than NestFC. On the high-dimensional Pubmed and Cora inputs, it also requires less peak memory and FitTime. Particularly on Cora, NestFC uses $\mathbf{95.3}\times$ as many parameters, $\mathbf{5.0}\times$ as much peak memory, and $\mathbf{3.9}\times$ as much FitTime as HFC-H.

I.2 SPD manifolds

I.2.1 Datasets

Radar[8] **[8].** It consists of 3,000 synthetic radar signals equally distributed across 3 classes.

[7] `https://github.com/cvgmi/Nested-Hyperbolic-DimReduc-and-HNN`
[8] `https://www.dropbox.com/s/dfnlx2bnyh3kjwy/data.zip?dl=0`

Table 20: Link-prediction test AUC (%).

Method	Disease	Airport	Pubmed	Cora
NHGCN-NestFC	78.28 ± 0.73	94.49 ± 0.04	94.70 ± 0.21	**93.11 ± 0.40**
NHGCN-HFC-H	**97.37 ± 0.22**	**96.14 ± 0.27**	**96.18 ± 0.06**	92.85 ± 0.24

Table 21: Full-model trainable parameters, peak memory (MiB), and FitTime (ms/epoch) for NHGCN-NestFC and NHGCN-HFC-H.

Method	Disease			Airport			Pubmed			Cora		
	#Param	MiB	FitTime	#Param	MiB	FitTime	#Param	MiB	FitTime	#Param	MiB	FitTime
NHGCN-NestFC	738	**81.10**	27.039	776	**120.29**	**43.292**	257,952	3496.40	105.094	2,075,436	1266.51	94.458
NHGCN-HFC-H	**450**	92.88	**19.854**	**465**	134.34	44.638	**7,785**	**3444.32**	**91.064**	**21,780**	**253.68**	**24.007**

HDM05[9] **[52].** It consists of 2,343 skeleton-based motion capture sequences executed by different actors. Each frame consists of 3D coordinates of 31 joints. We remove the under-represented clips, trimming the dataset down to 2,326 instances scattered throughout 122 classes. We randomly select 50% of the samples from each category for training and the remaining 50% for testing.

FPHA[10] **[29].** It includes 1,175 skeleton-based first-person hand gesture videos of 45 different categories with 600 clips for training and 575 for testing. Each frame contains the 3D coordinates of 21 hand joints.

For the HDM05 and FPHA datasets, we preprocess each sequence using the code[11] provided by Vemulapalli et al. [75] to normalize body part lengths and ensure invariance to scale and view.

I.2.2 SPD modeling

For our SPDNNs, we follow Wang et al. [76], Nguyen et al. [56] to model each sample into a multi-channel SPD tensor. For the Radar dataset, we follow Wang et al. [76] to use the temporal convolution followed by a covariance pooling layer to obtain a multi-channel covariance $[c, 20, 20]$ tensor. For the HDM05 and FPHA datasets, we follow Nguyen et al. [56, Sec. D.2.2] to model each skeleton sequence into a multi-channel covariance tensor $[c, n, n]$. Specifically, we first identify the closest left (right) neighbor of every joint based on their distance to the hip (wrist) joint, and then combine the 3D coordinates of each joint and those of its left (right) neighbor to create a feature vector for the joint. For a given frame t, we compute its Gaussian embedding [49]:

$$Y_t = (\det \Sigma_t)^{-\frac{1}{n+1}} \begin{bmatrix} \Sigma_t + \mu_t (\mu_t)^T & \mu_t \\ (\mu_t)^T & 1 \end{bmatrix}, \tag{79}$$

where μ_t and Σ_t are the mean vector and covariance matrix computed from the set of feature vectors within the frame. The lower part of matrix $\log(Y_t)$ is flattened to obtain a vector $\tilde{v}_t$. All vectors $\tilde{v}_t$ within a time window $[t, t+c-1]$, where c is determined from a temporal pyramid representation of the sequence (the number of temporal pyramids is set to 2 in our experiments), are used to compute a covariance matrix as

$$Z_t = \frac{1}{c} \sum_{i=t}^{t+c-1} (\tilde{v}_i - \overline{v}_t)(\tilde{v}_i - \overline{v}_t)^T, \tag{80}$$

where $\overline{v}_t = \frac{1}{c}\sum_{i=t}^{t+c-1} \tilde{v}_i$. The resulting $\{Z_t\}$ is the input covariance tensor. On the FPHA dataset, we generate the covariance based on three sets of neighbors: left, right, and vertical (bottom) neighbors.

For GyroLE, GyroAI, GyroLC, and GyroSPD++, the inputs are similar to those of our SPDNNs. For other SPD baselines, such as SPDNet, SPDNetBN, LieBN, MLR, and RResNet, each sequence is represented by a global covariance representation [34, 8]. The sizes of the covariance matrices are 20×20, 93×93, and 63×63 for the Radar, HDM05, and FPHA datasets, respectively.

[9] https://resources.mpi-inf.mpg.de/HDM05/

[10] https://github.com/guiggh/hand_pose_action

[11] https://ravitejav.weebly.com/kbac.html

Table 22: Training hyperparameters in SPDNNs.

Dataset	Model	θ	Optimizer	Learning Rate
Radar	SPDNN-LEM	N/A	AMSGrad	$5e^{-3}$
	SPDNN-AIM	0.25	AMSGrad	$5e^{-4}$
	SPDNN-PEM	N/A	AMSGrad	$1e^{-2}$
	SPDNN-LCM	0.25	AMSGrad	$5e^{-4}$
	SPDNN-BWM	N/A	AMSGrad	$5e^{-4}$
HDM05	SPDNN-LEM	N/A	SGD	$5e^{-3}$
	SPDNN-AIM	N/A	SGD	$5e^{-3}$
	SPDNN-PEM	N/A	AMSGrad	$1e^{-3}$
	SPDNN-LCM	N/A	AMSGrad	$1e^{-3}$
	SPDNN-BWM	N/A	AMSGrad	$1e^{-3}$
FPHA	SPDNN-LEM	N/A	AMSGrad	$1e^{-4}$
	SPDNN-AIM	N/A	AMSGrad	$1e^{-4}$
	SPDNN-PEM	N/A	AMSGrad	$1e^{-3}$
	SPDNN-LCM	-0.25	AMSGrad	$1e^{-3}$
	SPDNN-BWM	-0.25	AMSGrad	$1e^{-4}$
NTU60	SPDNN-LEM	N/A	SGD	$1e^{-3}$
	SPDNN-AIM	N/A	AMSGrad	$1e^{-4}$
	SPDNN-PEM	N/A	AMSGrad	$5e^{-4}$
	SPDNN-LCM	0.25	AMSGrad	$5e^{-4}$
	SPDNN-BWM	0.25	AMSGrad	$1e^{-3}$

I.2.3 Implementation details

Comparative methods. We follow the official PyTorch code of SPDNetBN[12] to implement SPDNet and SPDNetBN. For LieBN[13], we focus on the instantiations under AIM and LCM, while for RResNet[14], we implement the variants induced by LEM and AIM. For SPD MLR[15], we implement the variant induced by LCM. For GyroLE, GyroAI, GyroLC, and GyroSPD++, we re-implemented them based on the original papers [55, 56].

SPDNNs. On all datasets, we employ a single convolutional kernel for global convolution, *i.e.*, applying a global receptive field across the channel dimension. The output dimensions of the SPD convolutional layer are 8×8, 34×34, 22×22, and 11×11 for the Radar, HDM05, FPHA, and NTU60 datasets, respectively. We primarily use the AMSGrad [62] optimizer, except for SPDNN-LEM and SPDNN-AIM on the HDM05 dataset and SPDNN-LEM on NTU60, where SGD [63] is employed. Weight decay is set to zero except for SPDNN-PEM and SPDNN-BWM on the FPHA dataset, where it is $5e^{-4}$ and $1e^{-4}$, respectively. The matrix power in SPDNN-PEM is set to 0.5 for Radar and 0.25 for the other three datasets. Since matrix power can deform the latent Riemannian metric [17, Fig. 1], we also apply matrix power $(\cdot)^\theta$ before the convolutional layer in SPDNN-AIM, -LCM, and -BWM to activate the latent geometries. The batch size is set to 30, and training runs for 150 epochs with early stopping. Tab. 22 summarizes the training hyperparameters.

I.2.4 Reproduction Fidelity and Controlled Comparisons

Gyro [55] and GyroSPD++ [56] baselines are reproduced due to the unavailability of their official code.

Reproduction fidelity. Two factors may explain gaps between our reproduced and published results:

[12] https://proceedings.neurips.cc/paper_files/paper/2019/file/6e69ebbfad976d4637bb4b39de261bf7-Supplemental.zip

[13] https://github.com/GitZH-Chen/LieBN

[14] https://github.com/CUAI/Riemannian-Residual-Neural-Networks

[15] https://github.com/GitZH-Chen/SPDMLR

- **Input preprocessing.** We follow the original papers [55, 56], but the authors did not release processed inputs, and the papers omit the raw-skeleton preprocessing, ambiguous joint-to-neighbor cases [54, Supp. Sec. 6.1], and the temporal-pyramid partition. Thus, our inputs may not exactly match theirs:

$$\text{raw skeletons} \rightarrow \boxed{\text{unreported raw-data preprocessing}} \rightarrow \boxed{\text{underdetermined joint-neighbor selection}} \rightarrow \boxed{\text{unspecified temporal-pyramid partition}} \rightarrow \text{SPD inputs.}$$

- **Network implementation.** Because code for Gyro [55] and GyroSPD++ [56] is unavailable, we reimplemented them. For a controlled comparison, these baselines and our SPDNNs share inputs, pipeline, and code for common matrix operations, including Riemannian operators and matrix functions. We also tested the official training settings in Gyro [55, App. A.1] and GyroSPD++ [56, App. D.2.1], but they did not recover the published accuracies. We therefore report each baseline's best selected configuration in this shared pipeline.

Published vs. reproduced results. Among the three datasets, only FPHA uses the same official split. On HDM05, they use 130 classes and a subject split, whereas we follow Chen et al. [17, App. G.2.2] to use 122 classes and a per-class 50/50 split. On NTU60, they use cross-subject, whereas we use cross-view. Tab. 23 therefore reports only FPHA. All published FPHA results exceed our reproduced results. Even SPDNet and SPDNetBN, which have official code, show gaps of 3.20 and 1.69 points.

Table 23: FPHA accuracy (%): Source [55, 56], Ours, and Gap (Ours − Source; percentage points).

Method	FPHA		
	Source	Ours	Gap
SPDNet (official code)	88.79 ± 0.36	85.59 ± 0.72	-3.20
SPDNetBN (official code)	91.02 ± 0.25	89.33 ± 0.49	-1.69
GyroLE	94.61	90.73 ± 0.92	-3.88
GyroLC	82.43	76.10 ± 0.63	-6.33
GyroAI	93.39	89.60 ± 0.37	-3.79
GyroSPD++-AIM (source: AI-LE)	96.84 ± 0.27	89.50 ± 0.37	-7.34
GyroSPD++-LEM (source: LE-LE)	94.72 ± 0.25	88.23 ± 0.62	-6.49

Controlled fairness. This shared pipeline controls our internal comparison, and the Gyro [55] and GyroSPD++ [56] architectures follow the original papers. SPDNN and GyroSPD++ also use the same architecture, consisting of one SPD convolutional layer followed by an SPD MLR classifier.

I.2.5 Training efficiency

Tab. 24 presents the average training time per epoch of each SPD network. We have the following observations:

- **The efficiency of SPDNN varies across metrics.** The most efficient metric is LCM, where our model even achieves comparable efficiency to the vanilla SPDNet. However, AIM and BWM demonstrate significant computational burden, primarily due to their complex Riemannian computations.
- **Our trivialization improves efficiency.** Compared with the LCM-based SPDNetMLR, SPDNN-LCM achieves much lower training time. This improvement can be partially attributed to our trivialization, which simplifies the final expression of MLR (App. G.4) and eliminates the need for computationally expensive Riemannian optimization. Moreover, SPDNN consistently outperforms GyroSPD++ under LEM, LCM, and AIM in terms of efficiency. This advantage arises because our trivialization not only simplifies the expression of the FC and MLR layers, but also reduces the number of parameters.

I.2.6 Comparison with ManifoldNet and MVC-Net

To explicitly evaluate local receptive fields and kernel sharing, we conduct additional 1D convolutional experiments on Radar and compare our method with ManifoldNet and MVC-Net. Our main SPD

Table 24: Training efficiency (seconds per epoch).

Method	Geometry	Radar	HDM05	FPHA	NTU60
SPDNet	N/A	0.66	0.50	0.28	3.08
SPDNetBN	AIM	1.25	0.94	0.58	6.14
SPDResNet-AIM	AIM	0.96	1.23	0.69	6.84
SPDResNet-LEM	LEM	0.77	0.55	0.30	3.17
SPDNetLieBN-AIM	AIM	1.21	1.15	0.97	8.85
SPDNetLieBN-LCM	LCM	1.10	1.11	0.59	5.96
SPDNetMLR	LCM	0.66	5.46	0.88	4.94
GyroLE	LEM	0.79	2.86	1.59	10.57
GyroLC	LCM	0.66	1.49	0.78	5.99
GyroAI	AIM	0.99	22.80	12.62	26.76
GyroSPD++-AIM	AIM	5.09	103.57	66.35	125.05
GyroSPD++-LEM	LEM	0.99	0.95	0.66	7.58
GyroSPD++-LCM	LCM	0.66	0.70	0.37	5.74
SPDNN-LEM	LEM	0.86	0.74	0.63	5.79
SPDNN-AIM	AIM	4.84	101.80	65.42	124.41
SPDNN-PEM	PEM	1.09	7.10	1.57	8.71
SPDNN-LCM	LCM	0.65	0.59	0.35	3.72
SPDNN-BWM	BWM	6.07	110.51	71.67	139.48

Table 25: SPD CNN architectures with local receptive fields and shared kernels.

Input shape	#Conv	Kernels	Kernel sizes	Strides	Shape changes	Final SPD features
$[40, 3, 3]$	3	$[4, 8, 16]$	$[3, 3, 3]$	$[2, 2, 2]$	$[40, 3, 3] \to [19, 4, 3, 3] \to [9, 8, 3, 3] \to [4, 16, 3, 3]$	$[64, 3, 3]$
$[20, 6, 6]$	3	$[4, 8, 16]$	$[2, 2, 2]$	$[2, 2, 1]$	$[20, 6, 6] \to [10, 4, 6, 6] \to [5, 8, 6, 6] \to [4, 16, 6, 6]$	$[64, 6, 6]$
$[5, 12, 12]$	2	$[8, 20]$	$[2, 2]$	$[1, 1]$	$[5, 12, 12] \to [4, 8, 12, 12] \to [3, 20, 12, 12]$	$[60, 12, 12]$

experiments follow Gyro [55] and GyroSPD++ [56], representing each sample with only a few SPD descriptors and applying a single global receptive field across them. In this regime, convolution reduces to an FC layer on the product manifold and therefore does not directly evaluate local convolution. ManifoldNet and MVC-Net instead operate on manifold-valued fields $f : U \to \mathcal{M}$, where $U \subset \mathbb{Z}^d$ indexes spatial or temporal positions and each position stores a point on the manifold $\mathcal{M}$ [11, 7]. Their convolutional layers slide local windows over these positions and share kernel weights across positions.

Settings. Local receptive fields and kernel sharing require sufficiently many spatial or temporal positions. We therefore use Radar and split each $[2, 1000]$ signal into C consecutive temporal blocks. Covariance pooling within each block produces a 1D CNN input of shape $[C, n, n]$, where C is the number of temporal positions and each $[n, n]$ slice is an SPD matrix. We focus on three settings: $[40, 3, 3]$, $[20, 6, 6]$, and $[5, 12, 12]$. Tab. 25 gives the corresponding convolutional architectures. We refer to our SPD convolutional network as SPDConvNet and apply tangent ReLU after every convolutional layer. Following its original implementation, ManifoldNet uses no activation. MVC-Net includes tangent ReLU in its original formulation [7], but in our preliminary runs it caused numerical instability and yielded NaNs. We therefore omit tangent ReLU for MVC-Net.

Experiments. We compare ManifoldNet [11] and MVC-Net [7] with SPDConvNet instantiated with LEM, LCM, and PEM. All methods use the matched architectures in Tab. 25. As shown in Tab. 26, all three SPDConvNet variants achieve higher accuracy and train faster across the three settings. The efficiency difference arises from the repeated computation of weighted Fréchet means in ManifoldNet [11, Eq. (8)] and Fréchet means in MVC-Net [7, Def. 1] during convolution.

Learning-rate ablation. We additionally scan eight learning rates for ManifoldNet under the $[40, 3, 3]$ input setting. Its best result is 77.89%, confirming that the comparison is not caused by an untuned learning rate.

Table 26: Radar classification accuracy (%) and FitTime (s/epoch) under different architectures.

Method	$[40, 3, 3]$		$[20, 6, 6]$		$[5, 12, 12]$	
	Accuracy	FitTime	Accuracy	FitTime	Accuracy	FitTime
ManifoldNet	77.89 ± 1.56	13.12	73.27 ± 0.68	11.58	80.75 ± 1.23	11.50
MVC-Net	81.04 ± 1.72	7.88	78.05 ± 0.23	8.18	81.15 ± 0.34	7.87
SPDConvNet-LEM	**97.68 ± 0.32**	1.58	**95.86 ± 0.95**	1.55	96.91 ± 0.18	**1.03**
SPDConvNet-LCM	96.51 ± 0.66	**1.54**	94.40 ± 0.50	**1.22**	**97.20 ± 0.65**	1.25
SPDConvNet-PEM	97.44 ± 0.56	1.82	94.45 ± 0.78	1.53	96.16 ± 0.91	2.34

Table 27: ManifoldNet learning-rate ablation under the $[40, 3, 3]$ input setting.

Learning rate	5×10^{-5}	10^{-4}	2×10^{-4}	5×10^{-4}	10^{-3}	2×10^{-3}	5×10^{-3}	10^{-2}
Accuracy (%)	55.20 ± 2.58	59.73 ± 1.31	63.52 ± 1.44	69.55 ± 2.46	73.28 ± 1.42	76.08 ± 1.08	**77.89 ± 1.56**	77.12 ± 1.30

I.3 Grassmannian manifolds

Grassmannian modeling. As Grassmannian descriptors can be derived by the SVD of the covariance [36, 55], we map the multi-channel Radar covariance into a $[c, n, p]$ ONB Grassmannian tensor via the SVD. The PP Grassmannian features can be derived from the ONB Grassmannian features via the isometry $\pi(\cdot) : \mathrm{Gr}(p, n) \to \widetilde{\mathrm{Gr}}(p, n)$:

$$\pi(U) = UU^\top, \forall U \in \mathrm{Gr}(p, n). \tag{81}$$

Implementation details. Since GrNet [36] is officially implemented in Matlab, we carefully re-implemented it using PyTorch. Additionally, since both GyroGr and GyroGr-Scaling do not release official code, we re-implemented them based on the original papers [55]. For all comparative methods, we use SGD with a learning rate of $5e^{-2}$. For training our ONB and PP GrNNs, we use AMSGrad with a learning rate of $5e^{-3}$. The batch size is set to 30, and training runs for 150 epochs.

I.4 Hardware

On the HDM05 and FPHA datasets, SPDNet, RResNet, SPDNetBN, SPDNetLieBN, and MLR require SVD operations on relatively large matrices, which are more efficiently executed on a CPU. As a result, these methods are implemented on a CPU, whereas all other cases are executed on a single A6000 GPU.

J Proofs

J.1 Proof of Prop. 3.2

Proof. **Euclidean spaces.** We first review the following facts about Euclidean space:

- the origin is the zero vector $\mathbf{0}$;
- the standard orthonormal basis over $T_0\mathbb{R}^m \cong \mathbb{R}^m$ is $\{e_i\}_{i=1}^m$;
- $\mathrm{Log}_p(x) = x - p$ and $\langle \cdot, \cdot \rangle_p = \langle \cdot, \cdot \rangle$ for any $x, p \in \mathbb{R}^n$;
- point-to-hyperplane distance is $d(x, H_{a,p}) = \frac{|\langle x-p, a \rangle|}{\|a\|}$.

Putting the above together, one can recover Eq. (1).

Poincaré balls. This exactly corresponds to the derivation of the Poincaré FC layer [66, App. D.3].

SPD gyrovector spaces. Nguyen et al. [56] proposed three gyro SPD FC layers based on the gyrovector structures under LEM, LCM, and AIM, respectively. The discussion below summarizes the proof in Nguyen et al. [56, Apps. J-L].

In the SPD gyro FC layer, the origin of the SPD manifold is the identity matrix I. Given a metric among LEM, LCM, and AIM, let $\{B_i\}_{i=1}^d$ be an orthonormal basis over $T_I\mathcal{S}_{++}^m$, where $d = {}^{n(n+1)}/_2$ is the dimension of $\mathcal{S}_{++}^n$. The SPD gyro FC layer is defined by solving the following equations

$$\mathrm{sign}\left(\langle \mathrm{Log}_I(Y), B_k \rangle_I\right) \mathrm{d}_{\mathrm{gyo}}(Y, H_{B_k,I}) = \langle W_k, \ominus P_k \oplus X \rangle_{gr}, \quad 1 \le k \le m, \tag{82}$$

where $\mathrm{d}_{\mathrm{gyo}}$ denotes the corresponding pseudo-gyrodistance, $\ominus$ and $\oplus$ are gyro operations [56, Apps. G.2-G.4], and $\langle \cdot, \cdot \rangle_{gr}$ is the gyro inner product [56, App. G.7]. Here, each $W_k \in \mathcal{S}_{++}^n$ and $P_k \in \mathcal{S}_{++}^n$ are FC parameters. By Prop. 3.2 in Nguyen et al. [56], the RHS of Eq. (82) is equal to $\left\langle \mathrm{Log}_{P_k}(X), \Gamma_{I \to P_k}\left(\mathrm{Log}_I(W_k)\right) \right\rangle_{P_k}$. Setting $A_k = \Gamma_{I \to P_k}\left(\mathrm{Log}_I(W_k)\right) \in T_{P_k}\mathcal{S}_{++}^n$, one can recover Eq. (3). □

J.2 Proof of Thm. 3.3

Proof. The point-to-hyperplane pseudo-distance from a point $Y \in \mathcal{M}$ to a Riemannian hyperplane over $\mathcal{M}$ is

$$\mathrm{d}(Y, H_{A,P}) = \frac{|\langle \mathrm{Log}_P^{\mathcal{M}} Y, A \rangle_P^{\mathcal{M}}|}{\|A\|_P^{\mathcal{M}}}, \tag{83}$$

where $H_{A,P}$ is a Riemannian hyperplane parameterized by $P \in \mathcal{M}$ and $A \in T_P\mathcal{M}$. Therefore, the signed point-to-hyperplane pseudo-distance from Y to $H_{B_i,E}$ is

$$\begin{aligned} \mathrm{sign}\left(\left\langle \mathrm{Log}_E^{\mathcal{M}}(Y), B_i \right\rangle_E^{\mathcal{M}}\right) \mathrm{d}(Y, H_{B_i,E}) &= \frac{\langle \mathrm{Log}_E^{\mathcal{M}}(Y), B_i \rangle_E^{\mathcal{M}}}{\|B_i\|_E^{\mathcal{M}}} \\ &\overset{(1)}{=} \langle \mathrm{Log}_E^{\mathcal{M}}(Y), B_i \rangle_E^{\mathcal{M}} \end{aligned} \tag{84}$$

where (1) comes from the orthonormality of B_i.

Setting Eq. (84) equal to $v_i(X)$, we have

$$\langle \mathrm{Log}_E^{\mathcal{M}}(Y), B_i \rangle_E^{\mathcal{M}} = \langle \mathrm{Log}_{P_i}^{\mathcal{N}}(X), A_i \rangle_{P_i}^{\mathcal{N}}. \tag{85}$$

The above equation indicates

$$\mathrm{Log}_E^{\mathcal{M}}(Y) = \sum_{i=1}^{m} \left(\langle \mathrm{Log}_{P_i}^{\mathcal{N}}(X), A_i \rangle_{P_i}^{\mathcal{N}} B_i \right). \tag{86}$$

□

J.3 Proof of Thm. 4.1

To simplify the Riemannian FC layer with the gyro structure, we first prove a useful lemma.

Lemma J.1. *We assume that the manifold $\mathcal{M}$ admits a gyrogroup [54, Def. 2.2] defined by*[16]

$$x \oplus y = \mathrm{Exp}_x\left(\Gamma_{e \to x}\left(\mathrm{Log}_e\left(y\right)\right)\right), \forall x, y \in \mathcal{M}, \tag{87}$$

where $e \in \mathcal{M}$ is the origin of the manifold. Then, we have the following

$$\left\langle \mathrm{Log}_p(x), a \right\rangle_p = \left\langle \mathrm{Log}_e(\ominus p \oplus x), \Gamma_{p \to e}(a) \right\rangle_e, \quad \forall x, p \in \mathcal{M} \text{ and } \forall a \in T_p\mathcal{M}. \tag{88}$$

Proof. **Credit of the proof:** Eq. (87) comes from Nguyen and Yang [55, Eq. (1)], who demonstrated that several geometries admit gyrogroups based on this definition. The prototype of Eq. (88) comes from App. I by Nguyen et al. [56], which only deals with SPD matrices. Here, we further extend the result into general gyrogroups.

Denoting $\ominus p$ as the gyro inverse of p ($\ominus p \oplus p = e$), we have

$$\begin{aligned} x &\overset{(1)}{=} p \oplus (\ominus p \oplus x) \overset{(2)}{=} \mathrm{Exp}_p\left(\Gamma_{e \to p}\left(\mathrm{Log}_e\left(\ominus p \oplus x\right)\right)\right) \\ &\overset{(3)}{\Rightarrow} \mathrm{Log}_p(x) = \Gamma_{e \to p}\left(\mathrm{Log}_e\left(\ominus p \oplus x\right)\right). \end{aligned} \tag{89}$$

The above follows from the following.

(1) Left cancellation law of the gyrogroup [72, Thms. 1.13].
(2) Definition of gyro addition.
(3) Applying $\mathrm{Log}_p(\cdot)$ to both sides.

[16] We assume all the involved Riemannian operators are well-defined.

By the last equation, we have

$$\begin{aligned}\left\langle \operatorname{Log}_p(x), a\right\rangle_p &= \left\langle \Gamma_{e\to p}\left(\operatorname{Log}_e\left(\ominus p \oplus x\right)\right), a\right\rangle_p \\ &\overset{(1)}{=} \left\langle \operatorname{Log}_e\left(\ominus p \oplus x\right), \Gamma_{p\to e}(a)\right\rangle_e,\end{aligned} \tag{90}$$

where (1) comes from

- Parallel transport preserves the norm [23, Sec. 3.1].
- $\Gamma_{p\to e}\circ\Gamma_{e\to p}(v) = v, \forall v \in T_e\mathcal{M}$.

□

Now we begin to prove Thm. 4.1.

Proof of Thm. 4.1. In both geometries, the origin is defined as the zero vector, as it is the identity element in each gyrovector space. We first deal with the Poincaré ball, followed by the Beltrami–Klein model.

Poincaré ball: The Riemannian metric at the identity element is

$$\langle v, w\rangle_{\mathbf{0}} = 4\langle v, w\rangle, \forall v, w \in T_{\mathbf{0}}\mathbb{P}_K^m. \tag{91}$$

Obviously, $\{\frac{1}{2}e_i\}_{i=1}^m$ is an orthonormal basis. Lem. J.1 implies

$$\begin{aligned}\left\langle \operatorname{Log}_{p_i}(x), a_i\right\rangle_{p_i}\frac{1}{2}e_i &\overset{(1)}{=} \left\langle \operatorname{Log}_{\mathbf{0}}(-p_i\oplus_{\mathrm{M}} x), \Gamma_{p_i\to\mathbf{0}}(a_i)\right\rangle_{\mathbf{0}}\frac{1}{2}e_i \\ &\overset{(2)}{=} 2\left\langle \operatorname{Log}_{\mathbf{0}}(-p_i\oplus_{\mathrm{M}} x), \Gamma_{p_i\to\mathbf{0}}(a_i)\right\rangle e_i \\ &\overset{(3)}{=} \left\langle \operatorname{Log}_{\mathbf{0}}(-p_i\oplus_{\mathrm{M}} x), z_i\right\rangle e_i.\end{aligned} \tag{92}$$

The above comes from the following.

(1) Lem. J.1 and $\ominus_{\mathrm{M}} p = -p, \quad \forall p \in \mathbb{P}_K^n$.
(2) Eq. (91).
(3) $a_i = \Gamma_{\mathbf{0}\to p_i}(z_i/2)$. This reparameterization preserves p_i because $[z_i/2] = [z_i]$.

Beltrami–Klein model: The Riemannian metric at the identity element is

$$\langle v, w\rangle_{\mathbf{0}} = \langle v, w\rangle, \forall v, w \in T_{\mathbf{0}}\mathbb{K}_K^m. \tag{93}$$

Obviously, $\{e_i\}_{i=1}^m$ is an orthonormal basis. Lem. J.1 and Eq. (24) implies that the above reasoning for the Poincaré ball can be transferred into the Beltrami–Klein model:

$$\begin{aligned}\left\langle \operatorname{Log}_{p_i}(x), a_i\right\rangle_{p_i} e_i &\overset{(1)}{=} \left\langle \operatorname{Log}_{\mathbf{0}}(-p_i\oplus_{\mathrm{E}} x), \Gamma_{p_i\to\mathbf{0}}(a_i)\right\rangle_{\mathbf{0}} e_i \\ &= \left\langle \operatorname{Log}_{\mathbf{0}}(-p_i\oplus_{\mathrm{E}} x), \Gamma_{p_i\to\mathbf{0}}(a_i)\right\rangle e_i \\ &\overset{(2)}{=} \left\langle \operatorname{Log}_{\mathbf{0}}(-p_i\oplus_{\mathrm{E}} x), z_i\right\rangle e_i,\end{aligned} \tag{94}$$

The above comes from the following.

(1) Lem. J.1, Eq. (24), and $\ominus_{\mathrm{E}} p = -p$, .
(2) $a_i = \Gamma_{\mathbf{0}\to p_i}(z_i)$.

□

J.4 Proof of Thm. 4.2

Proof. We only need to show the origin, the tangent space at the origin, and the inner product and an orthonormal basis over the tangent space at the origin.

The hyperboloid is isometric to the Poincaré ball by the following diffeomorphism [42]:

$$\pi_{\mathbb{P}_K^n\to\mathbb{H}_K^n}(x) = \left(\frac{1}{\sqrt{|K|}}\frac{1-K\|x\|^2}{1+K\|x\|^2}; \frac{2x^T}{1+K\|x\|^2}\right)^\top. \tag{95}$$

The origin of hyperboloid is therefore defined as

$$e := \pi_{\mathbb{P}^n_K \to \mathbb{H}^n_K}(\mathbf{0}) = \left(\frac{1}{\sqrt{|K|}}, 0 \cdots, 0 \right)^\top. \tag{96}$$

The Riemannian metric and tangent space at e are

$$T_e \mathbb{H}^n_K = \{(0, v^\top)^\top | v \in \mathbb{R}^n\}, \tag{97}$$

$$\langle (0, v^\top)^\top, (0, w^\top)^\top \rangle_e = \langle v, w \rangle, \quad \forall (0, v^\top)^\top, (0, w^\top)^\top \in T_e \mathbb{H}^n_K. \tag{98}$$

Likewise, $\{(0, e_i^\top)^\top\}_{i=1}^m$ is an orthonormal basis of $T_e \mathbb{H}^m_K$ with $e_i \in \mathbb{R}^m$.

Combining the above with Tab. 11, we can instantiate Thm. 3.3 in the hyperboloid geometry. □

J.5 Proof of Thm. 4.3

Proof. **Poincaré ball.**

$$\begin{aligned}
v_i(x) &\overset{(1)}{=} \langle \mathrm{Log}_{\mathbf{0}}((-p_i) \oplus_{\mathrm{M}} x), z_i \rangle \\
&\overset{(2)}{=} \frac{\tanh^{-1}\left(\sqrt{|K|}\|(-p_i) \oplus_{\mathrm{M}} x\|\right)}{\sqrt{|K|}\|(-p_i) \oplus_{\mathrm{M}} x\|} \langle (-p_i) \oplus_{\mathrm{M}} x, z_i \rangle \\
&\overset{(3)}{=} \frac{\tanh^{-1}\left(\sqrt{|K|}\left\|\frac{(1+K\|p_i\|^2)x - \left(1+2K\langle p_i, x\rangle - K\|x\|^2\right)p_i}{1+2K\langle p_i,x\rangle + K^2\|p_i\|^2\|x\|^2}\right\|\right)}{\sqrt{|K|}\left\|\frac{(1+K\|p_i\|^2)x - (1+2K\langle p_i,x\rangle - K\|x\|^2)p_i}{1+2K\langle p_i,x\rangle + K^2\|p_i\|^2\|x\|^2}\right\|} \left\langle \frac{(1+K\|p_i\|^2)x - \left(1+2K\langle p_i,x\rangle - K\|x\|^2\right)p_i}{1+2K\langle p_i,x\rangle + K^2\|p_i\|^2\|x\|^2}, z_i \right\rangle \\
&\overset{(4)}{=} \frac{\tanh^{-1}\left(\sqrt{\frac{|K|\|x-p_i\|^2}{1+2K\langle p_i,x\rangle + K^2\|p_i\|^2\|x\|^2}}\right)}{\sqrt{\frac{|K|\|x-p_i\|^2}{1+2K\langle p_i,x\rangle + K^2\|p_i\|^2\|x\|^2}}} \frac{(1+K\|p_i\|^2)\langle x, z_i\rangle - \left(1+2K\langle p_i,x\rangle - K\|x\|^2\right)\langle p_i, z_i\rangle}{1+2K\langle p_i,x\rangle + K^2\|p_i\|^2\|x\|^2} \\
&\overset{(5)}{=} \frac{\tanh^{-1}\left(\sqrt{\frac{|K|\|x-p_i\|^2}{1+2K\langle p_i,x\rangle + K^2\|p_i\|^2\|x\|^2}}\right)}{\sqrt{\frac{|K|\|x-p_i\|^2}{1+2K\langle p_i,x\rangle + K^2\|p_i\|^2\|x\|^2}}} \frac{(1-K\|p_i\|^2)\langle x, z_i\rangle - (1-K\|x\|^2)\langle p_i, z_i\rangle}{1+2K\langle p_i,x\rangle + K^2\|p_i\|^2\|x\|^2} \\
&\overset{(6)}{=} \frac{\tanh^{-1}\left(\sqrt{\frac{|K|\|x-p_i\|^2}{1+2K\langle p_i,x\rangle + K^2\|p_i\|^2\|x\|^2}}\right)\left[(1-K\|p_i\|^2)\langle x, z_i\rangle - (1-K\|x\|^2)\langle p_i, z_i\rangle\right]}{\sqrt{|K|\|x-p_i\|^2\left(1+2K\langle p_i,x\rangle + K^2\|p_i\|^2\|x\|^2\right)}} \\
&\overset{(7)}{=} \|z_i\| \frac{\tanh^{-1}\left(\sqrt{|K|}Q_i^{\mathrm{P}}\right)}{\sqrt{|K|}Q_i^{\mathrm{P}}} \frac{\left[1+\tanh^2(\sqrt{|K|}\gamma_i)\right]\langle x, [z_i]\rangle - \frac{\tanh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}\left(1+|K|\|x\|^2\right)}{D_i^{\mathrm{P}}}.
\end{aligned}$$

The above comes from the following.

(1) The first equality is the HFC-P expression in Thm. 4.1.

(2) From Eq. (23),

$$\mathrm{Log}_{\mathbf{0}}((-p_i) \oplus_{\mathrm{M}} x) = \frac{\tanh^{-1}(\sqrt{|K|}\|(-p_i) \oplus_{\mathrm{M}} x\|)}{\sqrt{|K|}\|(-p_i) \oplus_{\mathrm{M}} x\|}((-p_i) \oplus_{\mathrm{M}} x).$$

Taking its inner product with z_i gives the second equality.

(3) Substituting $-p_i$ and x into the Möbius addition in Eq. (15) gives

$$(-p_i) \oplus_{\mathrm{M}} x = \frac{(1+K\|p_i\|^2)x - \left(1+2K\langle p_i,x\rangle - K\|x\|^2\right)p_i}{1+2K\langle p_i,x\rangle + K^2\|p_i\|^2\|x\|^2}.$$

Inserting this vector into both occurrences of $(-p_i) \oplus_{\mathrm{M}} x$ gives the third equality.

(4) Expanding the squared norm of the numerator in (3) gives

$$\begin{aligned}
&\left\|(1+K\|p_i\|^2)x - \left(1+2K\langle p_i,x\rangle - K\|x\|^2\right)p_i\right\|^2 \\
&= (1+K\|p_i\|^2)^2\|x\|^2 + \left(1+2K\langle p_i,x\rangle - K\|x\|^2\right)^2\|p_i\|^2 \\
&\quad - 2(1+K\|p_i\|^2)\left(1+2K\langle p_i,x\rangle - K\|x\|^2\right)\langle p_i, x\rangle \\
&= \|x-p_i\|^2\left(1+2K\langle p_i,x\rangle + K^2\|p_i\|^2\|x\|^2\right).
\end{aligned}$$

Therefore,

$$\|(-p_i) \oplus_{\mathrm{M}} x\|^2 = \frac{\|x - p_i\|^2}{1 + 2K\langle p_i, x\rangle + K^2\|p_i\|^2\|x\|^2}.$$

The corresponding inner product is

$$\langle (-p_i) \oplus_{\mathrm{M}} x, z_i\rangle = \frac{(1 + K\|p_i\|^2)\langle x, z_i\rangle - \left(1 + 2K\langle p_i, x\rangle - K\|x\|^2\right)\langle p_i, z_i\rangle}{1 + 2K\langle p_i, x\rangle + K^2\|p_i\|^2\|x\|^2}.$$

Substituting these two identities into (3) gives the fourth equality.

(5) Since $p_i = \mathrm{Exp}_{\mathbf{0}}(\gamma_i[z_i])$, we have $p_i \parallel z_i$ and hence

$$\langle p_i, x\rangle\langle p_i, z_i\rangle = \|p_i\|^2\langle x, z_i\rangle.$$

It follows that

$$\begin{aligned}&(1 + K\|p_i\|^2)\langle x, z_i\rangle - \left(1 + 2K\langle p_i, x\rangle - K\|x\|^2\right)\langle p_i, z_i\rangle \\ &= (1 - K\|p_i\|^2)\langle x, z_i\rangle - (1 - K\|x\|^2)\langle p_i, z_i\rangle,\end{aligned}$$

which gives the fifth equality.

(6) The scalar denominators satisfy

$$\begin{aligned}&\frac{1}{\sqrt{\frac{|K|\|x-p_i\|^2}{1+2K\langle p_i,x\rangle+K^2\|p_i\|^2\|x\|^2}}}\frac{1}{1 + 2K\langle p_i, x\rangle + K^2\|p_i\|^2\|x\|^2} \\ &= \frac{1}{\sqrt{|K|\|x - p_i\|^2\left(1 + 2K\langle p_i, x\rangle + K^2\|p_i\|^2\|x\|^2\right)}}.\end{aligned}$$

This yields the sixth equality.

(7) The exponential map at the origin gives

$$p_i = \mathrm{Exp}_{\mathbf{0}}(\gamma_i[z_i]) = \frac{\tanh\left(\sqrt{|K|}\gamma_i\right)}{\sqrt{|K|}}[z_i].$$

Since $[z_i] = z_i/\|z_i\|$, we have

$$\begin{aligned}\|p_i\|^2 &= \frac{\tanh^2(\sqrt{|K|}\gamma_i)}{|K|}, \\ \langle p_i, x\rangle &= \frac{\tanh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}\langle [z_i], x\rangle, \\ \langle p_i, z_i\rangle &= \frac{\tanh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}\|z_i\|, \\ \langle x, z_i\rangle &= \|z_i\|\langle x, [z_i]\rangle, \\ \|x - p_i\|^2 &= \|x\|^2 + \frac{\tanh^2(\sqrt{|K|}\gamma_i)}{|K|} - \frac{2\tanh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}\langle x, [z_i]\rangle.\end{aligned}$$

Define

$$\begin{aligned}D_i^{\mathrm{P}} &= 1 - 2\sqrt{|K|}\tanh(\sqrt{|K|}\gamma_i)\langle x, [z_i]\rangle + |K|\tanh^2(\sqrt{|K|}\gamma_i)\|x\|^2, \\ Q_i^{\mathrm{P}} &= \sqrt{\frac{\|x\|^2 + \frac{\tanh^2(\sqrt{|K|}\gamma_i)}{|K|} - \frac{2\tanh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}\langle x, [z_i]\rangle}{D_i^{\mathrm{P}}}}.\end{aligned}$$

When $Q_i^{\mathrm{P}} = 0$, the quotient $\tanh^{-1}(\sqrt{|K|}Q_i^{\mathrm{P}})/(\sqrt{|K|}Q_i^{\mathrm{P}})$ is defined as 1 by continuity. Substituting these identities into (6) and using $K = -|K|$ gives the seventh equality.

Beltrami–Klein model.

$$
\begin{aligned}
v_i(x) &\overset{(1)}{=} \langle \mathrm{Log}_{\mathbf{0}}((-p_i) \oplus_{\mathrm{E}} x), z_i \rangle \\
&\overset{(2)}{=} \frac{\tanh^{-1}\left(\sqrt{|K|}\|(-p_i) \oplus_{\mathrm{E}} x\|\right)}{\sqrt{|K|}\|(-p_i) \oplus_{\mathrm{E}} x\|} \langle (-p_i) \oplus_{\mathrm{E}} x, z_i \rangle \\
&\overset{(3)}{=} \frac{\tanh^{-1}\left(\sqrt{|K|}\left\|\frac{\sqrt{1+K\|p_i\|^2}x - \left(1+\frac{K\langle p_i,x\rangle}{1+\sqrt{1+K\|p_i\|^2}}\right)p_i}{1+K\langle p_i,x\rangle}\right\|\right)}{\sqrt{|K|}\left\|\frac{\sqrt{1+K\|p_i\|^2}x - \left(1+\frac{K\langle p_i,x\rangle}{1+\sqrt{1+K\|p_i\|^2}}\right)p_i}{1+K\langle p_i,x\rangle}\right\|} \times \left\langle \frac{\sqrt{1+K\|p_i\|^2}x - \left(1+\frac{K\langle p_i,x\rangle}{1+\sqrt{1+K\|p_i\|^2}}\right)p_i}{1+K\langle p_i,x\rangle}, z_i \right\rangle \\
&\overset{(4)}{=} \frac{\tanh^{-1}\left(\frac{\sqrt{|K|[\|x-p_i\|^2+K(\|p_i\|^2\|x\|^2-\langle p_i,x\rangle^2)]}}{1+K\langle p_i,x\rangle}\right)}{\frac{\sqrt{|K|[\|x-p_i\|^2+K(\|p_i\|^2\|x\|^2-\langle p_i,x\rangle^2)]}}{1+K\langle p_i,x\rangle}} \times \frac{1}{1+K\langle p_i,x\rangle}\left[\sqrt{1+K\|p_i\|^2}\langle x,z_i\rangle - \left(1+\frac{K\langle p_i,x\rangle}{1+\sqrt{1+K\|p_i\|^2}}\right)\langle p_i,z_i\rangle\right] \\
&\overset{(5)}{=} \frac{\tanh^{-1}\left(\frac{\sqrt{|K|[\|x-p_i\|^2+K(\|p_i\|^2\|x\|^2-\langle p_i,x\rangle^2)]}}{1+K\langle p_i,x\rangle}\right)}{\frac{\sqrt{|K|[\|x-p_i\|^2+K(\|p_i\|^2\|x\|^2-\langle p_i,x\rangle^2)]}}{1+K\langle p_i,x\rangle}} \frac{\langle x-p_i, z_i\rangle}{1+K\langle p_i,x\rangle} \\
&\overset{(6)}{=} \frac{\tanh^{-1}\left(\frac{\sqrt{|K|[\|x-p_i\|^2+K(\|p_i\|^2\|x\|^2-\langle p_i,x\rangle^2)]}}{1+K\langle p_i,x\rangle}\right)}{\sqrt{|K|\left[\|x-p_i\|^2+K\left(\|p_i\|^2\|x\|^2-\langle p_i,x\rangle^2\right)\right]}}\langle x-p_i, z_i\rangle \\
&\overset{(7)}{=} \|z_i\|\frac{\tanh^{-1}\left(\sqrt{|K|}Q_i^{\mathrm{K}}\right)}{\sqrt{|K|}Q_i^{\mathrm{K}}} \frac{\langle x,[z_i]\rangle - \frac{\tanh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}}{D_i^{\mathrm{K}}}.
\end{aligned}
$$

The above comes from the following.

(1) The first equality is the HFC-K expression in Thm. 4.1.
(2) Applying the logarithmic map at $\mathbf{0}$ in Eq. (23) to $(-p_i) \oplus_{\mathrm{E}} x$ gives the second equality.
(3) The gamma factor of $-p_i$ is

$$\gamma_{-p_i} = \frac{1}{\sqrt{1+K\|p_i\|^2}}.$$

Substituting $-p_i$ and x into the Einstein addition in App. D.1 and replacing γ_{-p_i} gives

$$(-p_i) \oplus_{\mathrm{E}} x = \frac{\sqrt{1+K\|p_i\|^2}x - \left(1+\frac{K\langle p_i,x\rangle}{1+\sqrt{1+K\|p_i\|^2}}\right)p_i}{1+K\langle p_i,x\rangle}.$$

Inserting this vector into both occurrences of $(-p_i) \oplus_{\mathrm{E}} x$ gives the third equality.
(4) Expanding the squared norm of the numerator in (3) gives

$$
\begin{aligned}
&\left\|\sqrt{1+K\|p_i\|^2}x - \left(1+\frac{K\langle p_i,x\rangle}{1+\sqrt{1+K\|p_i\|^2}}\right)p_i\right\|^2 \\
&= (1+K\|p_i\|^2)\|x\|^2 + \left(1+\frac{K\langle p_i,x\rangle}{1+\sqrt{1+K\|p_i\|^2}}\right)^2\|p_i\|^2 \\
&\quad - 2\sqrt{1+K\|p_i\|^2}\left(1+\frac{K\langle p_i,x\rangle}{1+\sqrt{1+K\|p_i\|^2}}\right)\langle p_i,x\rangle \\
&= \|x-p_i\|^2 + K\left(\|p_i\|^2\|x\|^2 - \langle p_i,x\rangle^2\right).
\end{aligned}
$$

Therefore,

$$\|(-p_i) \oplus_{\mathrm{E}} x\|^2 = \frac{\|x-p_i\|^2 + K\left(\|p_i\|^2\|x\|^2 - \langle p_i,x\rangle^2\right)}{(1+K\langle p_i,x\rangle)^2}.$$

Expanding the inner product of the vector in (3) with z_i gives

$$\langle(-p_i)\oplus_{\mathrm{E}} x, z_i\rangle = \frac{1}{1+K\langle p_i,x\rangle}\left[\sqrt{1+K\|p_i\|^2}\langle x,z_i\rangle - \left(1+\frac{K\langle p_i,x\rangle}{1+\sqrt{1+K\|p_i\|^2}}\right)\langle p_i,z_i\rangle\right].$$

Substituting these two identities into (3) gives the fourth equality.

(5) Since $p_i \parallel z_i$,

$$\langle p_i,x\rangle\langle p_i,z_i\rangle = \|p_i\|^2\langle x,z_i\rangle.$$

Using this identity, the directional term in (4) becomes

$$\begin{aligned}&\sqrt{1+K\|p_i\|^2}\langle x,z_i\rangle - \left(1+\frac{K\langle p_i,x\rangle}{1+\sqrt{1+K\|p_i\|^2}}\right)\langle p_i,z_i\rangle\\ &= \left(\sqrt{1+K\|p_i\|^2} - \frac{K\|p_i\|^2}{1+\sqrt{1+K\|p_i\|^2}}\right)\langle x,z_i\rangle - \langle p_i,z_i\rangle\\ &= \langle x,z_i\rangle - \langle p_i,z_i\rangle = \langle x-p_i,z_i\rangle.\end{aligned}$$

Here, the second equality uses

$$\sqrt{1+K\|p_i\|^2} - \frac{K\|p_i\|^2}{1+\sqrt{1+K\|p_i\|^2}} = 1.$$

This gives the fifth equality.

(6) The remaining scalar factors satisfy

$$\frac{1}{\frac{\sqrt{|K|[\|x-p_i\|^2+K(\|p_i\|^2\|x\|^2-\langle p_i,x\rangle^2)]}}{1+K\langle p_i,x\rangle}}\frac{1}{1+K\langle p_i,x\rangle} = \frac{1}{\sqrt{|K|\left[\|x-p_i\|^2+K\left(\|p_i\|^2\|x\|^2-\langle p_i,x\rangle^2\right)\right]}}.$$

This gives the sixth equality.

(7) The exponential map at the origin gives

$$p_i = \frac{\tanh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}[z_i]$$

Define

$$D_i^{\mathrm{K}} = 1-\sqrt{|K|}\tanh(\sqrt{|K|}\gamma_i)\langle x,[z_i]\rangle,$$

$$Q_i^{\mathrm{K}} = \sqrt{\frac{\|x\|^2+\frac{\tanh^2(\sqrt{|K|}\gamma_i)}{|K|}-\frac{2\tanh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}\langle x,[z_i]\rangle - \tanh^2(\sqrt{|K|}\gamma_i)\left(\|x\|^2-\langle x,[z_i]\rangle^2\right)}{(D_i^{\mathrm{K}})^2}}.$$

When $Q_i^{\mathrm{K}}=0$, the quotient $\tanh^{-1}(\sqrt{|K|}Q_i^{\mathrm{K}})/(\sqrt{|K|}Q_i^{\mathrm{K}})$ is defined as 1 by continuity. Substituting p_i into (6) gives

$$\begin{aligned}1+K\langle p_i,x\rangle &= D_i^{\mathrm{K}},\\ \langle x-p_i,z_i\rangle &= \|z_i\|\left(\langle x,[z_i]\rangle - \frac{\tanh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}\right),\\ \|(-p_i)\oplus_{\mathrm{E}} x\| &= Q_i^{\mathrm{K}}.\end{aligned}$$

These identities give the seventh equality.

Hyperboloid model. For $x = (x_1, x_s)$ and $p_i = (p_{i1}, p_{is})$, we have

$$
\begin{aligned}
v_i(x) &\overset{(1)}{=} \left\langle \mathrm{Log}_{p_i}(x), \Gamma_{e\to p_i}(0, z_i) \right\rangle_{\mathcal{L}} \\
&\overset{(2)}{=} \frac{\cosh^{-1}\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)}{\sqrt{\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)^2 - 1}} \left\langle x - K\langle p_i, x\rangle_{\mathcal{L}} p_i, \Gamma_{e\to p_i}(0, z_i) \right\rangle_{\mathcal{L}} \\
&\overset{(3)}{=} \frac{\cosh^{-1}\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)}{\sqrt{\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)^2 - 1}} \left\langle x, \Gamma_{e\to p_i}(0, z_i) \right\rangle_{\mathcal{L}} \\
&\overset{(4)}{=} \frac{\cosh^{-1}\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)}{\sqrt{\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)^2 - 1}} \left\langle x, \left(\sqrt{|K|}\langle p_{is}, z_i\rangle, z_i + \frac{|K|\langle p_{is}, z_i\rangle}{1 + \sqrt{|K|}p_{i1}} p_{is} \right) \right\rangle_{\mathcal{L}} \\
&\overset{(5)}{=} \frac{\cosh^{-1}\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)}{\sqrt{\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)^2 - 1}} \left[\langle x_s, z_i\rangle + \frac{|K|\langle p_{is}, z_i\rangle\langle x_s, p_{is}\rangle}{1 + \sqrt{|K|}p_{i1}} - \sqrt{|K|}x_1\langle p_{is}, z_i\rangle \right] \\
&\overset{(6)}{=} \frac{\cosh^{-1}\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)}{\sqrt{\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)^2 - 1}} \left[\left(1 + \frac{|K|\|p_{is}\|^2}{1 + \sqrt{|K|}p_{i1}} \right) \langle x_s, z_i\rangle - \sqrt{|K|}x_1\langle p_{is}, z_i\rangle \right] \\
&\overset{(7)}{=} \sqrt{|K|}\frac{\cosh^{-1}\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)}{\sqrt{\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)^2 - 1}} \left(p_{i1}\langle x_s, z_i\rangle - x_1\langle p_{is}, z_i\rangle \right) \\
&\overset{(8)}{=} \|z_i\| \frac{\cosh^{-1}\left(\sqrt{|K|}\left[\cosh(\sqrt{|K|}\gamma_i)x_1 - \sinh(\sqrt{|K|}\gamma_i)\langle x_s, [z_i]\rangle\right]\right)}{\sqrt{|K|\left[\cosh(\sqrt{|K|}\gamma_i)x_1 - \sinh(\sqrt{|K|}\gamma_i)\langle x_s, [z_i]\rangle\right]^2 - 1}} \left[\cosh(\sqrt{|K|}\gamma_i)\langle x_s, [z_i]\rangle - \sinh(\sqrt{|K|}\gamma_i)x_1\right].
\end{aligned}
$$

The above comes from the following.

(1) The first equality is the HFC-H expression in Thm. 4.2.

(2) From Tab. 11,

$$
\mathrm{Log}_{p_i}(x) = \frac{\cosh^{-1}\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)}{\sinh\left(\cosh^{-1}\left(K\langle p_i, x\rangle_{\mathcal{L}}\right)\right)} \left(x - K\langle p_i, x\rangle_{\mathcal{L}} p_i\right).
$$

Since $\sinh(\cosh^{-1}(t)) = \sqrt{t^2 - 1}$ for $t \geq 1$, substituting this logarithmic map into (1) gives the second equality.

(3) Parallel transport maps $(0, z_i) \in T_e\mathbb{H}^n_K$ to $T_{p_i}\mathbb{H}^n_K$. Therefore,

$$
\langle p_i, \Gamma_{e\to p_i}(0, z_i)\rangle_{\mathcal{L}} = 0.
$$

Consequently,

$$
\begin{aligned}
&\langle x - K\langle p_i, x\rangle_{\mathcal{L}} p_i, \Gamma_{e\to p_i}(0, z_i)\rangle_{\mathcal{L}} \\
&= \langle x, \Gamma_{e\to p_i}(0, z_i)\rangle_{\mathcal{L}},
\end{aligned}
$$

which gives the third equality.

(4) The parallel transport in Tab. 11 gives

$$
\Gamma_{e\to p_i}(0, z_i) = (0, z_i) - \frac{K\langle p_i, (0, z_i)\rangle_{\mathcal{L}}}{1 + K\langle e, p_i\rangle_{\mathcal{L}}}(e + p_i).
$$

Using $e = (1/\sqrt{|K|}, 0)$, we have

$$
\langle p_i, (0, z_i)\rangle_{\mathcal{L}} = \langle p_{is}, z_i\rangle, \qquad 1 + K\langle e, p_i\rangle_{\mathcal{L}} = 1 + \sqrt{|K|}p_{i1}.
$$

Hence,

$$
\Gamma_{e\to p_i}(0, z_i) = \left(\sqrt{|K|}\langle p_{is}, z_i\rangle, z_i + \frac{|K|\langle p_{is}, z_i\rangle}{1 + \sqrt{|K|}p_{i1}} p_{is} \right),
$$

which gives the fourth equality.

(5) Expanding the Lorentz inner product in (4) gives

$$\left\langle x, \left(\sqrt{|K|}\langle p_{is}, z_i\rangle, z_i + \frac{|K|\langle p_{is}, z_i\rangle}{1+\sqrt{|K|}p_{i1}} p_{is} \right) \right\rangle_{\mathcal{L}}$$
$$= \langle x_s, z_i\rangle + \frac{|K|\langle p_{is}, z_i\rangle\langle x_s, p_{is}\rangle}{1+\sqrt{|K|}p_{i1}} - \sqrt{|K|}x_1\langle p_{is}, z_i\rangle.$$

This gives the fifth equality.

(6) Since $p_i = \mathrm{Exp}_e(\gamma_i[(0, z_i^\top)^\top])$, the spatial component satisfies $p_{is} \parallel z_i$. Thus,

$$\langle p_{is}, z_i\rangle\langle x_s, p_{is}\rangle = \|p_{is}\|^2\langle x_s, z_i\rangle,$$

which gives the sixth equality.

(7) Since $p_i \in \mathbb{H}^n_K$,

$$\|p_{is}\|^2 - p_{i1}^2 = \frac{1}{K} = -\frac{1}{|K|}.$$

Therefore,

$$1 + \frac{|K|\|p_{is}\|^2}{1+\sqrt{|K|}p_{i1}} = 1 + \frac{|K|p_{i1}^2 - 1}{1+\sqrt{|K|}p_{i1}} = \sqrt{|K|}p_{i1}.$$

Substituting this identity into (6) gives the seventh equality.

(8) The exponential map at e gives

$$p_i = \left(\frac{\cosh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}, \frac{\sinh(\sqrt{|K|}\gamma_i)}{\sqrt{|K|}}[z_i] \right).$$

Hence,

$$K\langle p_i, x\rangle_{\mathcal{L}} = \sqrt{|K|}\left[\cosh(\sqrt{|K|}\gamma_i)x_1 - \sinh(\sqrt{|K|}\gamma_i)\langle x_s, [z_i]\rangle\right],$$
$$\sqrt{|K|}\left(p_{i1}\langle x_s, z_i\rangle - x_1\langle p_{is}, z_i\rangle\right) = \|z_i\|\left[\cosh(\sqrt{|K|}\gamma_i)\langle x_s, [z_i]\rangle - \sinh(\sqrt{|K|}\gamma_i)x_1\right].$$

Substituting these two identities into (7) gives the eighth equality.

When $x = p_i$, we have $K\langle p_i, x\rangle_{\mathcal{L}} = 1$. The quotient $\cosh^{-1}(K\langle p_i, x\rangle_{\mathcal{L}})/\sqrt{(K\langle p_i, x\rangle_{\mathcal{L}})^2 - 1}$ is defined as 1 by continuity. The remaining factor in (7) vanishes, and hence $v_i(p_i) = 0$. □

J.6 Proof of Thm. 4.4

Proof. In the following proof, we first present the expressions of several operators under different metrics, including $v_{ij}(S)$, standard orthonormal bases, and Riemannian exponential maps at the origin. Then, we prove the theorem. In this proof, we follow the notation of the theorem.

$v_{ij}(S)$ **under different metrics:** The expressions are implied by Chen et al. [17, Thm. 4.2]:

$$\text{LEM} : \langle \log(S) - \log(P_{ij}), Z_{ij}\rangle^{(\alpha,\beta)}, \tag{99}$$
$$\text{AIM} : \left\langle \log(P_{ij}^{-\frac{1}{2}} S P_{ij}^{-\frac{1}{2}}), Z_{ij} \right\rangle^{(\alpha,\beta)}, \tag{100}$$
$$\text{PEM} : \frac{1}{\theta}\left\langle S^\theta - P_{ij}^\theta, Z_{ij}\right\rangle^{(\alpha,\beta)}, \tag{101}$$
$$\text{LCM} : \left\langle \lfloor K\rfloor - \lfloor L_{ij}\rfloor + \mathrm{Dlog}(\mathbb{K}\mathbb{L}_{ij}^{-1}), \lfloor Z_{ij}\rfloor + \frac{1}{2}\mathbb{Z}_{ij} \right\rangle, \tag{102}$$
$$\text{BWM} : \frac{1}{2}\left\langle (P_{ij}S)^{\frac{1}{2}} + (SP_{ij})^{\frac{1}{2}} - 2P_{ij}, \mathcal{L}_{P_{ij}}(L_{ij}Z_{ij}L_{ij}^\top) \right\rangle. \tag{103}$$

Standard orthonormal bases: Next, we show the standard orthonormal bases over $T_I\mathcal{S}^n_{++}$ under different metrics. As indicated by Tabs. 13 and 14, the inner products for any $V, W \in T_I\mathcal{S}^n_{++}$ are

$$\text{LEM, AIM, and PEM} : \langle V, W\rangle^{(\alpha,\beta)}, \tag{104}$$

$$\text{LCM}: \langle \lfloor V \rfloor + \frac{1}{2}\mathbb{V}, \lfloor W \rfloor + \frac{1}{2}\mathbb{W} \rangle, \tag{105}$$

$$\text{BWM}: \frac{1}{4}\langle V, W \rangle \tag{106}$$

The above comes from the following.

(1) Eq. (104) comes from $\log_{*,I}(V) = V$ and $\mathrm{P}_{\theta *,I}(V) = \theta V$;
(2) Eq. (105) comes from $\mathrm{Chol}_{*,I}(V) = \lfloor V \rfloor + \frac{1}{2}\mathbb{V}$;
(3) Eq. (106) comes from $\mathcal{L}_I[V] = \frac{1}{2}V$.

As shown by Thanwerdas and Pennec [70, Thm.2.1], $F_{\sqrt{\alpha+n\beta},\sqrt{\alpha}} : \{\mathcal{S}^n, \langle \cdot, \cdot \rangle^{(\alpha,\beta)}\} \to \{\mathcal{S}^n, \langle \cdot, \cdot \rangle\}$ is the linear isometry pulling the standard inner product back to the $\mathrm{O}(n)$-invariant one:

$$F_{\sqrt{\alpha+n\beta},\sqrt{\alpha}}(X) = \sqrt{\alpha}X + \frac{\sqrt{\alpha+n\beta} - \sqrt{\alpha}}{n} \operatorname{tr}(X) I_n, \forall X \in \mathcal{S}^n. \tag{107}$$

Given any $Y \in \mathcal{S}^n$, its inverse map is

$$\begin{aligned} \left(F_{\sqrt{\alpha+n\beta},\sqrt{\alpha}}\right)^{-1}(Y) &= \frac{1}{\sqrt{\alpha}} \left\{ Y - \left(\frac{\sqrt{1+n\frac{\beta}{\alpha}} - 1}{n} \frac{1}{\sqrt{1+n\frac{\beta}{\alpha}}} \right) \operatorname{tr}(Y) I \right\} \\ &= \frac{1}{\sqrt{\alpha}} \left\{ Y - \frac{1}{n} \left(1 - \frac{1}{\sqrt{1+n\frac{\beta}{\alpha}}} \right) \operatorname{tr}(Y) I \right\} \\ &= \frac{1}{\sqrt{\alpha}} Y - \frac{1}{n} \left(\frac{1}{\sqrt{\alpha}} - \frac{1}{\sqrt{\alpha+n\beta}} \right) \operatorname{tr}(Y) I. \end{aligned} \tag{108}$$

The standard orthonormal bases over the Euclidean spaces $\{\mathcal{S}^n, \langle \cdot, \cdot \rangle\}$ and $\{\mathcal{L}^n, \langle \cdot, \cdot \rangle\}$ are

$$\{\mathcal{S}^n, \langle \cdot, \cdot \rangle\} : U_{ij}^{\mathrm{sym}} = \begin{cases} E_{ii}, & \text{if } i = j, \\ \frac{E_{ij}+E_{ji}}{\sqrt{2}}, & \text{if } i > j. \end{cases} \tag{109}$$

$$\{\mathcal{L}^n, \langle \cdot, \cdot \rangle\} : U_{ij}^{\mathrm{tril}} = E_{ij}, \forall i \geq j \tag{110}$$

where $i \geq j, i, j = 1, \cdots, n$, and $\{E_{ij}\}_{i,j=1}^n$ are standard basis matrices, with the (k, l) element defined by

$$(E_{ij})_{kl} = \begin{cases} 1 & \text{if } k = i \text{ and } l = j, \\ 0 & \text{otherwise.} \end{cases} \tag{111}$$

The standard orthonormal bases with respect to Eqs. (104) to (106) are

$$\text{LEM, AIM, PEM}: U_{ij}^{(\alpha,\beta)} \overset{(1)}{=} \begin{cases} \frac{1}{\sqrt{\alpha}} E_{ii} - \frac{1}{n}\left(\frac{1}{\sqrt{\alpha}} - \frac{1}{\sqrt{\alpha+n\beta}}\right) I, & \text{if } i = j, \\ \frac{E_{ij}+E_{ji}}{\sqrt{2\alpha}}, & \text{if } i > j. \end{cases} \tag{112}$$

$$\text{LCM}: U_{ij}^{\mathrm{LC}} \overset{(2)}{=} \begin{cases} 2E_{ii}, & \text{if } i = j, \\ E_{ij}, & \text{if } i > j. \end{cases} \tag{113}$$

$$\text{BWM}: U_{ij}^{\mathrm{BW}} \overset{(3)}{=} \begin{cases} 2E_{ii}, & \text{if } i = j, \\ \sqrt{2}(E_{ij} + E_{ji}), & \text{if } i > j. \end{cases} \tag{114}$$

Here, $i \geq j, i, j = 1, \cdots, n$. The above comes from the following.

(1) $U_{ij}^{(\alpha,\beta)} = \left(F_{\sqrt{\alpha+n\beta},\sqrt{\alpha}}\right)^{-1}\left(U_{ij}^{\mathrm{sym}}\right)$, with $F_{\sqrt{\alpha+n\beta},\sqrt{\alpha}} : \mathcal{S}^n \to \mathcal{S}^n$ as the linear isometry pulling back the Frobenius inner product to the $\mathrm{O}(n)$-invariant inner product;
(2) $f^{\mathrm{LC}}(V) = \lfloor V \rfloor + \frac{1}{2}\mathbb{V} : \mathcal{L}^n \to \mathcal{L}^n$ is the linear isometry pulling the Frobenius inner product to Eq. (105);
(3) $f^{\mathrm{BW}}(V) = \frac{1}{2}V : \mathcal{S}^n \to \mathcal{S}^n$ is the linear isometry pulling the Frobenius inner product back to Eq. (106);

Riemannian exponential maps: Next, we show Exp_I under different metrics

$$\text{LEM and AIM}: \mathrm{Exp}_I(V) \overset{(1)}{=} \exp(V), \tag{115}$$

$$\text{PEM}: \mathrm{Exp}_I(V) \overset{(2)}{=} (I+\theta V)^{\frac{1}{\theta}}, \tag{116}$$

$$\text{LCM}: \mathrm{Exp}_I(V) \overset{(3)}{=} \left(\lfloor V \rfloor + \mathrm{Dexp}\left(\frac{1}{2}\mathbb{V}\right)\right)\left(\lfloor V \rfloor + \mathrm{Dexp}\left(\frac{1}{2}\mathbb{V}\right)\right)^{\top}, \tag{117}$$

$$\text{BWM}: \mathrm{Exp}_I(V) \overset{(4)}{=} I + V + \frac{1}{4}V^2 = \left(I + \frac{1}{2}V\right)^2, \tag{118}$$

The above comes from the following.

(1) $\log_{*,I}(V) = V$ and $\log I = \mathbf{0}$;
(2) $\mathrm{P}_{\theta *,I}(V) = \theta V$;
(3) $\mathrm{Chol}_{*,I}(V) = \lfloor V \rfloor + \frac{1}{2}\mathbb{V}$;
(4) $\mathcal{L}_I[V] = \frac{1}{2}V$.

Now, we can prove the results metric by metric.

LEM:

$$\begin{aligned}&\mathrm{Exp}_I\left(\sum_{i,j=1,i\geq j}^{m} v_{ij}^{\mathrm{LE}}(S)U_{ij}^{(\alpha,\beta)}\right)\\ &= \exp\left(\sum_{i,j=1,i\geq j}^{m}\left(\log(S) - \log(P_{ij}), Z_{ij}\right\rangle^{(\alpha,\beta)}U_{ij}^{(\alpha,\beta)}\right)\right).\end{aligned} \tag{119}$$

AIM:

$$\begin{aligned}&\mathrm{Exp}_I\left(\sum_{i,j=1,i\geq j}^{m} v_{ij}^{\mathrm{AI}}(S)U_{ij}^{(\alpha,\beta)}\right)\\ &= \exp\left(\sum_{i,j=1,i\geq j}^{m}\left(\langle\log(P_{ij}^{-\frac{1}{2}}SP_{ij}^{-\frac{1}{2}}), Z_{ij}\right\rangle^{(\alpha,\beta)}U_{ij}^{(\alpha,\beta)}\right)\right).\end{aligned} \tag{120}$$

PEM:

$$\begin{aligned}&\mathrm{Exp}_I\left(\sum_{i,j=1,i\geq j}^{m} v_{ij}^{\mathrm{PE}}(S)U_{ij}^{(\alpha,\beta)}\right)\\ &= \left(I + \theta\sum_{i,j=1,i\geq j}^{m}\left(\frac{1}{\theta}\langle S^{\theta} - P_{ij}^{\theta}, Z_{ij}\rangle^{(\alpha,\beta)}U_{ij}^{(\alpha,\beta)}\right)\right)^{\frac{1}{\theta}}\\ &= \left(I + \sum_{i,j=1,i\geq j}^{m}\left(\langle S^{\theta} - P_{ij}^{\theta}, Z_{ij}\rangle^{(\alpha,\beta)}U_{ij}^{(\alpha,\beta)}\right)\right)^{\frac{1}{\theta}}.\end{aligned} \tag{121}$$

LCM:

$$\begin{aligned}&\mathrm{Exp}_I\left(\sum_{i,j=1,i\geq j}^{m} v_{ij}^{\mathrm{LC}}(S)U_{ij}^{\mathrm{LC}}\right)\\ &= \left(\lfloor V^{\mathrm{LC}} \rfloor + \mathrm{Dexp}\left(\frac{1}{2}\mathbb{V}^{\mathrm{LC}}\right)\right)\left(\lfloor V^{\mathrm{LC}} \rfloor + \mathrm{Dexp}\left(\frac{1}{2}\mathbb{V}^{\mathrm{LC}}\right)\right)^{\top},\end{aligned} \tag{122}$$

with

$$\begin{aligned} V^{\mathrm{LC}} &= \sum_{i,j=1,i\geq j}^{m} v_{ij}^{\mathrm{LC}}(S) U_{ij}^{\mathrm{LC}} \\ &= \sum_{i,j=1,i\geq j}^{m} \left(\left\langle \lfloor K \rfloor - \lfloor L_{ij} \rfloor + \mathrm{Dlog}(\mathbb{K}\mathbb{L}_{ij}^{-1}), \lfloor Z_{ij} \rfloor + \frac{1}{2}\mathbb{Z}_{ij} \right\rangle \right) U_{ij}^{\mathrm{LC}} \end{aligned} \tag{123}$$

BWM:

$$\begin{aligned} &\mathrm{Exp}_I \left(\sum_{i,j=1,i\geq j}^{m} v_{ij}^{\mathrm{BW}}(S) U_{ij}^{\mathrm{BW}} \right) \\ &= \left(I + \frac{1}{2} V^{\mathrm{BW}} \right)^2, \end{aligned} \tag{124}$$

with V^{BW} defined as

$$V^{\mathrm{BW}} = \sum_{i,j=1,i\geq j}^{m} \left\{ \frac{1}{2} \left\langle (P_{ij}S)^{\frac{1}{2}} + (SP_{ij})^{\frac{1}{2}} - 2P_{ij}, \mathcal{L}_{P_{ij}}(L_{ij} Z_{ij} L_{ij}^{\top}) \right\rangle U_{ij}^{\mathrm{BW}} \right\}. \tag{125}$$

□

J.7 Proof of Prop. 4.5

We begin by recalling two vector structures on the SPD manifold. Next, we identify the expression for the linear homomorphisms. Finally, we present our proof.

We define a map $\phi(\cdot) : \mathcal{S}_{++}^n \to \mathcal{L}^n$ as

$$\phi(S) = \lfloor L \rfloor + \mathrm{Dlog}(\mathbb{L}), \tag{126}$$

where $P = LL^{\top}$ is the Cholesky decomposition. For any $P, Q \in \mathcal{S}_{++}^n$ and $t \in \mathbb{R}$, the vector structures over the SPD manifold are defined as

$$P \oplus^{\mathrm{LE}} Q = \exp(\log(P) + \log(Q)) \tag{127}$$

$$t \odot^{\mathrm{LE}} P = \exp(t \log(P)) = P^t \tag{128}$$

$$P \oplus^{\mathrm{LC}} Q = \phi^{-1}(\phi(P) + \phi(Q)) \tag{129}$$

$$t \odot^{\mathrm{LC}} P = \phi^{-1}(t\phi(P)) = P^t \tag{130}$$

As shown by Arsigny et al. [2], Chen et al. [18], $\{\mathcal{S}_{++}^n, \oplus^{\mathrm{LE}}, \odot^{\mathrm{LE}}\}$ and $\{\mathcal{S}_{++}^n, \oplus^{\mathrm{LC}}, \odot^{\mathrm{LC}}\}$ forms vector spaces. We further present the associated linear homomorphisms.

Lemma J.2 (SPD Homomorphisms)**.** *Given any homomorphisms*

$$\zeta^{\mathrm{LE}}(\cdot) : \{\mathcal{S}_{++}^n, \oplus^{\mathrm{LE}}, \odot^{\mathrm{LE}}\} \to \{\mathcal{S}_{++}^m, \oplus^{\mathrm{LE}}, \odot^{\mathrm{LE}}\}, \tag{131}$$

$$\zeta^{\mathrm{LC}}(\cdot) : \{\mathcal{S}_{++}^n, \oplus^{\mathrm{LC}}, \odot^{\mathrm{LC}}\} \to \{\mathcal{S}_{++}^m, \oplus^{\mathrm{LC}}, \odot^{\mathrm{LC}}\}, \tag{132}$$

they can be expressed as

$$\zeta^{\mathrm{LE}} = \exp \circ g \circ \log, \tag{133}$$

$$\zeta^{\mathrm{LC}} = \phi^{-1} \circ f \circ \phi, \tag{134}$$

where $f : \mathcal{L}^n \to \mathcal{L}^m$ and $g : \mathcal{S}^n \to \mathcal{S}^m$ are linear homomorphisms over the Euclidean space $\mathcal{L}^n$ and $\mathcal{S}^n$, respectively.

Proof. As shown by Chen et al. [18], $\log(\cdot)$ is the linear isomorphism from $\{\mathcal{S}_{++}^n, \oplus^{\mathrm{LE}}, \odot^{\mathrm{LE}}\}$ to the Euclidean space $\mathcal{S}^n$ and ϕ is the linear isomorphism from $\{\mathcal{S}_{++}^n, \oplus^{\mathrm{LC}}, \odot^{\mathrm{LC}}\}$ to the Euclidean space $\mathcal{L}^n$. Therefore, any linear homomorphisms over these two linear spaces have the following forms:

$$\zeta^{\mathrm{LE}} = \log^{-1} f \circ \log, \tag{135}$$

$$\zeta^{\mathrm{LC}} = \phi^{-1} g \circ \phi, \tag{136}$$

where $f : \mathcal{S}^n \to \mathcal{S}^m$ and $g : \mathcal{L}^n \to \mathcal{L}^m$ are linear homomorphisms over the Euclidean space $\mathcal{S}^n$ and $\mathcal{L}^n$, respectively. □

With all the above theoretical preparation, we begin to present our proof.

Proof. Given an SPD matrix $S \in \mathcal{S}_{++}^n$, Eq. (135) can be rewritten as

$$\begin{aligned} \zeta^{\mathrm{LE}}(S) &\overset{(1)}{=} \exp\left(\sum_{i,j=1,i\geq j}^{m} \langle \log(S), A_{ij}\rangle\, U_{ij}^{\mathrm{sym}}\right) \\ &\overset{(2)}{=} \exp\left(\sum_{i,j=1,i\geq j}^{m} \langle \log(S), A_{ij}\rangle\, U_{ij}^{(1,0)}\right) \\ &\overset{(3)}{=} \mathcal{F}^{\mathrm{LE}}(S;\mathbf{A},\mathbf{I}) \end{aligned} \tag{137}$$

where $\mathbf{A} = \{A_{ij} \in \mathcal{S}^n\}_{i,j=1,i\geq j}^m$ and $\mathbf{I} = \{I, \cdots, I\}$. The above comes from the following.

(1) The linear map f can be represented by $\{A_{ij} \in \mathcal{S}^n\}_{i,j=1,i\geq j}^m$ under the bases $\{U_{ij}^{\mathrm{sym}}\}_{i,j=1,i\geq j}^n$ over $\mathcal{S}^n$ and $\{U_{ij}^{\mathrm{sym}}\}_{i,j=1,i\geq j}^m$ over $\mathcal{S}^m$;
(2) $\{U_{ij}^{\mathrm{sym}}\}_{i,j=1,i\geq j}^m = \{U_{ij}^{(1,0)}\}_{i,j=1,i\geq j}^m$;
(3) $\mathrm{Exp}_I = \exp$ under LEM.

Following the above logic, we have the following for $\{\mathcal{S}_{++}^n, \oplus^{\mathrm{LC}}, \odot^{\mathrm{LC}}\}$:

$$\begin{aligned} \zeta^{\mathrm{LC}}(S) &\overset{(1)}{=} \phi^{-1}\left(\sum_{i,j=1,i\geq j}^{m} \langle \phi(S), A_{ij}\rangle\, U_{ij}^{\mathrm{tril}}\right) \\ &\overset{(2)}{=} \mathcal{F}^{\mathrm{LC}}(S;\mathbf{Z},\mathbf{I}), \end{aligned} \tag{138}$$

where $A_{ij} \in \mathcal{L}^n$ for $i,j = 1,\cdots,m, i \geq j$, $\mathbf{Z} = \{Z_{ij} = A_{ij} + \mathbb{D}(A_{ij}) \in \mathcal{L}^n\}_{i,j=1,i\geq j}^m$ and $\mathbf{I} = \{I, \cdots, I\}$. The above comes from the following.

(1) The linear map g can be represented by $\{A_{ij}\}_{i,j=1,i\geq j}^m$;
(2) Eq. (7) and v_{ij}^{LC}.

□

J.8 Proof of Thm. 4.6

Before presenting our proof, we first discuss some basic facts about the ONB Grassmannian FC layer. As implied by Eq. (30), any tangent vector $V \in T_{I_{p,n}}\mathrm{Gr}(p,n)$ can be expressed as

$$V = \begin{pmatrix} \mathbf{0} \\ I_{n-p} \end{pmatrix} B_V = \begin{pmatrix} \mathbf{0} \\ B_V \end{pmatrix}, \text{ with } B_V \in \mathbb{R}^{(n-p)\times p}. \tag{139}$$

According to Thm. 3.3 and Eq. (139), the ONB Grassmannian FC layer $\mathcal{F}(\cdot) : \mathrm{Gr}(p,n) \to \mathrm{Gr}(q,m)$ has the following form:

$$Y = \mathrm{Exp}_{I_{q,m}}\left(\sum_{\substack{i=1,\cdots,m-q \\ j=1,\cdots,m}} \Big(\langle \mathrm{Log}_{P_{ij}}(X), A_{ij}\rangle_{P_{ij}} U_{ij}\Big)\right), \tag{140}$$

where $\{U_{ij}\}$ is an orthonormal basis over $T_{I_{q,m}}\mathrm{Gr}(q,m)$. As discussed in Sec. 3.3, we model the FC parameters by parallel transport and the Riemannian exponential map:

$$A_{ij} = \Gamma_{I_{p,n}\to P_{ij}}(Z_{ij}), \tag{141}$$

$$P_{ij} = \mathrm{Exp}_{I_{p,n}}(\gamma_{ij}[Z_{ij}]), \tag{142}$$

where $Z_{ij} = \begin{pmatrix} \mathbf{0} \\ B_{Z_{ij}} \end{pmatrix} \in T_{I_{p,n}}\mathrm{Gr}(p,n)$. Therefore, we can model each P_{ij} and A_{ij} by $B_{Z_{ij}} \in \mathbb{R}^{(n-p)\times p}$ and $\gamma_{ij} \in \mathbb{R}$. With the above ingredient, we present the proof in the following.

Proof. **The standard orthonormal basis:** As the inner product over $T_{I_{q,m}}\mathrm{Gr}(q,m)$ is the Frobenius matrix inner product [5, Eq. 3.2], the standard orthonormal basis over $T_{I_{q,m}}\mathrm{Gr}(q,m)$ is

$$U_{ij} = \begin{pmatrix} \mathbf{0} \\ E_{ij} \end{pmatrix}, 1 \leq i \leq m-q \wedge 1 \leq j \leq q, \tag{143}$$

where $\{E_{ij}\}$ are standard basis matrices over $\mathbb{R}^{(m-q)\times q}$

The Riemannian exponential map at the origin: The SVD of $V \in T_{I_{p,n}}\mathrm{Gr}(p,n)$ can be calculated via the SVD of B_V:

$$V = \begin{pmatrix} \mathbf{0} \\ B_V \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ O \end{pmatrix} \Sigma R^\top = \begin{pmatrix} \mathbf{0} \\ O\Sigma R^\top \end{pmatrix}, \tag{144}$$

where $B_V \overset{\text{SVD}}{:=} O\Sigma R^\top$. Therefore, the Riemannian exponential map at $I_{p,n}$ can be simplified as

$$\begin{aligned} \mathrm{Exp}_{I_{p,n}}(V) &= \begin{pmatrix} I_p \\ \mathbf{0} \end{pmatrix} R\cos(\Sigma)R^T + \begin{pmatrix} \mathbf{0} \\ O \end{pmatrix} \sin(\Sigma)R^T \\ &= \begin{pmatrix} R\cos(\Sigma)R^T \\ O\sin(\Sigma)R^T \end{pmatrix} \end{aligned} \tag{145}$$

$v_{ij}(U)$ **under the ONB perspective:** The ONB parallel transport can be further simplified. Given $P \in \mathrm{Gr}(p,n)$, we have the following for the Riemannian logarithm

$$\mathrm{Log}_{I_{p,n}}(P) = \begin{pmatrix} \mathbf{0} \\ B_P \end{pmatrix} \overset{\text{SVD}}{:=} \begin{pmatrix} \mathbf{0} \\ O_P\Sigma_P R_P^\top \end{pmatrix}, \tag{146}$$

with $B_P \overset{\text{SVD}}{:=} O_P\Sigma_P R_P^\top$. For $P \in \mathrm{Gr}(p,n)$ and $Z \in T_{I_{p,n}}\mathrm{Gr}(p,n)$, the parallel transport can be further simplified:

$$\begin{aligned} &\Gamma_{I_{p,n}\to P}(Z) \\ &= \left(\begin{pmatrix} I_{p,n}R_P & \begin{pmatrix} \mathbf{0} \\ O_P \end{pmatrix} \end{pmatrix} \begin{pmatrix} -\sin(\Sigma_P) \\ \cos(\Sigma_P) \end{pmatrix} \begin{pmatrix} \mathbf{0} \\ O_P \end{pmatrix}^T + \left(I - \begin{pmatrix} \mathbf{0} \\ O_P \end{pmatrix} \begin{pmatrix} \mathbf{0} \\ O_P \end{pmatrix}^T\right)\right) Z \\ &= \left(\left(-\begin{pmatrix} I_p \\ \mathbf{0} \end{pmatrix} R_P \sin(\Sigma_P) + \begin{pmatrix} \mathbf{0} \\ O_P \end{pmatrix} \cos(\Sigma_P)\right) \begin{pmatrix} \mathbf{0} \\ O_P \end{pmatrix}^T + \begin{pmatrix} I_p & \mathbf{0} \\ \mathbf{0} & I_{n-p} - O_P O_P^\top \end{pmatrix}\right) Z \\ &= \left(\begin{pmatrix} -R_P\sin(\Sigma_P) \\ O_P\cos(\Sigma_P) \end{pmatrix} \begin{pmatrix} \mathbf{0} & O_P^\top \end{pmatrix} + \begin{pmatrix} I_p & \mathbf{0} \\ \mathbf{0} & I_{n-p} - O_P O_P^\top \end{pmatrix}\right) Z \\ &= \left(\begin{pmatrix} \mathbf{0} & -R_P\sin(\Sigma_P)O_P^\top \\ \mathbf{0} & O_P\cos(\Sigma_P)O_P^\top \end{pmatrix} + \begin{pmatrix} I_p & \mathbf{0} \\ \mathbf{0} & I_{n-p} - O_P O_P^\top \end{pmatrix}\right) Z \\ &= \begin{pmatrix} I_p & -R_P\sin(\Sigma_P)O_P^\top \\ \mathbf{0} & I_{n-p} + O_P\cos(\Sigma_P)O_P^\top - O_P O_P^\top \end{pmatrix} Z \\ &= \begin{pmatrix} I_p & -R_P\sin(\Sigma_P)O_P^\top \\ \mathbf{0} & I_{n-p} + O_P\cos(\Sigma_P)O_P^\top - O_P O_P^\top \end{pmatrix} \begin{pmatrix} \mathbf{0} \\ B_Z \end{pmatrix} \\ &= \begin{pmatrix} -R_P\sin(\Sigma_P)O_P^\top B_Z \\ \left(O_P\cos(\Sigma_P)O_P^\top + I_{n-p} - O_P O_P^\top\right) B_Z \end{pmatrix}. \end{aligned}$$

Combining all the above results, one can directly obtain the results. □

J.9 Proof of Thm. 4.7

Proof. First, $v_{ij}(X)$ over the Grassmannian $\widetilde{\mathrm{Gr}}(p,n)$ takes the following form:

$$\begin{aligned} v_{ij}(X) &= \left\langle \mathrm{Log}_{P_{ij}}(X), \Gamma_{\widetilde{I}_{p,n}\to P_{ij}}(Z_{ij}) \right\rangle_{P_{ij}} \\ &\overset{(1)}{=} \frac{1}{2}\left\langle \mathrm{Log}_{P_{ij}}(X), \Gamma_{\widetilde{I}_{p,n}\to P_{ij}}(Z_{ij}) \right\rangle \end{aligned} \tag{147}$$

where (1) comes from Tab. 15. Here, each $Z_{ij} \in T_{\widetilde{I}_{p,n}} \widetilde{\mathrm{Gr}}(p,n)$ and $P_{ij} \in \widetilde{\mathrm{Gr}}(p,n)$.

Riemannian logarithm. As shown by Nguyen et al. [56, Prop. 3.12], the PP Grassmannian logarithm can be calculated using the ONB logarithm:

$$\mathrm{Log}^{\mathrm{PP}}_P(X) = \pi_{*,\pi(P)} \left(\mathrm{Log}^{\mathrm{ONB}}_{\pi^{-1}(P)}(\pi^{-1}(X)) \right), \tag{148}$$

where $\pi(U) = UU^\top : \mathrm{Gr}(p,n) \to \widetilde{\mathrm{Gr}}(p,n)$ is the Riemannian isometry, and $\pi_{*,U}(V) = UV^\top + VU^\top$ is the differential map for all $U \in \mathrm{Gr}(p,n)$ and $V \in T_U \mathrm{Gr}(p,n)$.

Tangent vector and Riemannian exponential map at the identity. As implied by Eq. (32), any tangent vector at the identity has the following form:

$$V = \begin{pmatrix} 0 & B^T \\ B & 0 \end{pmatrix} \in T_{\widetilde{I}_{p,n}} \widetilde{\mathrm{Gr}}(p,n) \text{ with } B \in \mathbb{R}^{(n-p)\times p}. \tag{149}$$

The Riemannian exponential map at the identity can also be simplified:

$$\begin{aligned} \mathrm{Exp}_{\widetilde{I}_{p,n}}(V) &= \exp([V, \widetilde{I}_{p,n}]) \widetilde{I}_{p,n} \exp(-[V, \widetilde{I}_{p,n}]) \\ &= \exp\left(\begin{pmatrix} 0 & -B^T \\ B & 0 \end{pmatrix} \right) \widetilde{I}_{p,n} \exp\left(\begin{pmatrix} 0 & -B^T \\ B & 0 \end{pmatrix} \right)^\top \\ &= \left(\exp\left(\begin{pmatrix} 0 & -B^T \\ B & 0 \end{pmatrix} \right) \right)_{1:p} \left(\left(\exp\left(\begin{pmatrix} 0 & -B^T \\ B & 0 \end{pmatrix} \right) \right)_{1:p} \right)^\top \end{aligned} \tag{150}$$

with $(\cdot)_{1:p}$ being the first-p columns of the input square matrix.

Parallel transport starting at the identity. The parallel transport along geodesic from $\widetilde{I}_{p,n}$ to $P \in \widetilde{\mathrm{Gr}}(p,n)$ can also be simplified. For any $V \in T_{\widetilde{I}_{p,n}} \widetilde{\mathrm{Gr}}(p,n)$, denoting $\bar{P} = \mathrm{Log}_{\widetilde{I}_{p,n}}(P)$, we have the following:

$$\begin{aligned} \Gamma_{\widetilde{I}_{p,n} \to P}(V) &\overset{(1)}{=} \exp\left(\left[\bar{P}, \widetilde{I}_{p,n} \right] \right) V \exp\left(- \left[\bar{P}, \widetilde{I}_{p,n} \right] \right) \\ &\overset{(2)}{=} \exp\left(\begin{pmatrix} 0 & -B_P^T \\ B_P & 0 \end{pmatrix} \right) V \exp\left(\begin{pmatrix} 0 & -B_P^T \\ B_P & 0 \end{pmatrix} \right)^\top \end{aligned} \tag{151}$$

The above derivation comes from the following.

(1) Tab. 15;

(2) $\bar{P} = \begin{pmatrix} 0 & B_P^T \\ B_P & 0 \end{pmatrix}$

Trivialization and simplification Combining Eqs. (147) and (149) to (151), we model each P_{ij} such that

$$P_{ij} = \exp\left(\begin{pmatrix} 0 & -B_{P_{ij}}^T \\ B_{P_{ij}} & 0 \end{pmatrix} \right) \widetilde{I}_{p,n} \exp\left(\begin{pmatrix} 0 & -B_{P_{ij}}^T \\ B_{P_{ij}} & 0 \end{pmatrix} \right)^\top \tag{152}$$

where $B_{P_{ij}} = \gamma_{ij}[B_{Z_{ij}}]$ with $Z_{ij} = \begin{pmatrix} 0 & B_{Z_{ij}}^T \\ B_{Z_{ij}} & 0 \end{pmatrix}$ and $B_{Z_{ij}} \in \mathbb{R}^{(n-p)\times p}$.

Denoting $O_{ij} = \exp\left(\begin{pmatrix} 0 & -B_{P_{ij}}^T \\ B_{P_{ij}} & 0 \end{pmatrix} \right)$, $v_{ij}(X)$ can be simplified as

$$v_{ij}(X) = \frac{1}{2} \left\langle \pi_{*,\pi(P)} \left(\mathrm{Log}^{\mathrm{ONB}}_{(O_{ij})_{1:p}}(\pi^{-1}(X)) \right), O_{ij} Z_{ij} O_{ij}^\top \right\rangle \tag{153}$$

Orthonormal bases. Finally, let us handle the orthonormal bases over $T_{\widetilde{I}_{q,m}} \widetilde{\mathrm{Gr}}(q,m)$. For any tangent vector $V_1, V_2 \in T_{\widetilde{I}_{q,m}} \widetilde{\mathrm{Gr}}(q,m)$, we have the following:

$$\begin{aligned} \langle V_1, V_2 \rangle_{\widetilde{I}_{p,n}} &= \frac{1}{2} \langle V_1, V_2 \rangle \\ &= \frac{1}{2} \left\langle \begin{pmatrix} 0 & B_{V_1}^T \\ B_{V_1} & 0 \end{pmatrix}, \begin{pmatrix} 0 & B_{V_2}^T \\ B_{V_2} & 0 \end{pmatrix} \right\rangle \\ &= \langle B_{V_1}, B_{V_2} \rangle \end{aligned} \tag{154}$$

Therefore, the orthonormal basis is

$$U_{ij} = \begin{pmatrix} 0 & E_{ij}^\top \\ E_{ij} & 0 \end{pmatrix}, \forall i = 1, \cdots, m-q \wedge j = 1, \cdots, q \tag{155}$$

where $E_{ij} \in \mathbb{R}^{(m-q)\times q}$ is the standard basis matrix.

Combining Eqs. (150), (153) and (155), one can readily obtain the results. □

NeurIPS Paper Checklist

1. **Claims**
   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?
   Answer: [Yes]
   Justification: The abstract and Sec. 1 state the scope, contributions, and empirical claims. The main method, examples, and experiments in Secs. 3.1, 3.2 and 5 support these claims across hyperbolic, SPD, and Grassmannian manifolds.
   Guidelines:
   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.
2. **Limitations**
   Question: Does the paper discuss the limitations of the work performed by the authors?
   Answer: [Yes]
   Justification: The limitations are discussed in App. B. The paper states that the framework applies to computationally tractable Riemannian manifolds and may not directly apply when tractable Riemannian operators are unavailable.
   Guidelines:
   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.
3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?
Answer: [Yes]
Justification: The theoretical statements are presented in the main text and appendix with assumptions such as well-defined Riemannian operators discussed in Rmks. D.1 and D.2. Complete proofs are provided in App. J.
Guidelines:
- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**
Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?
Answer: [Yes]
Justification: The experimental setup, datasets, modeling choices, architectures, optimizers, and hyperparameters are described in Sec. 5 and App. I. The method definitions and proofs provide the information needed to reproduce the proposed layers.
Guidelines:
- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**
Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?
Answer: [No]

Justification: Dataset sources and several baseline code sources are cited in App. I. The code will be released after the review process.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Training and testing details are described in Apps. I.1 to I.3. The appendix reports dataset splits where applicable, optimizers, learning rates, batch size, epochs, and model-specific settings.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The experimental tables report values with $\pm$ and describe K-fold averages.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?
Answer: [Yes]
Justification: App. I.4 summarizes the hardware information.
Guidelines:
- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**
Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?
Answer: [Yes]
Justification: The paper uses publicly available benchmark datasets and there is no ethical concern.
Guidelines:
- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**
Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?
Answer: [N/A]
Justification: The work is primarily methodological and evaluated on benchmark tasks.
Guidelines:
- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**
Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?
Answer: [N/A]
Justification: The paper does not release high-risk pretrained models, image generators, language models, or scraped datasets. The experiments use existing benchmark datasets.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Existing datasets and baseline implementations are cited and several URLs are provided in App. I.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The paper does not introduce or release a new dataset, benchmark, or pretrained model asset. The proposed layers and networks are documented as methods in the paper.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not conduct crowdsourcing experiments or new human-subject studies. It uses existing benchmark datasets.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**
    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?
    Answer: [N/A]
    Justification: The paper does not report crowdsourcing or new human-subject research requiring IRB approval. The experiments rely on existing benchmark datasets.
    Guidelines:
    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.
16. **Declaration of LLM usage**
    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.
    Answer: [N/A]
    Justification: App. A describes their use for language polishing, minor editing, and limited assistance in translating mathematical formulations into PyTorch code.
    Guidelines:
    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.