CONSTANT-CURVATURE SLICED GROMOV–WASSERSTEIN FOR HETEROGENEOUS CROSS-CURVATURE ALIGNMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advances in representation learning have highlighted the utility of constant-curvature models, such as hyperbolic and spherical spaces, for modeling complex data. Mixed-curvature models further enhance this by integrating multiple constant-curvature components. However, these models typically learn each component space independently because spaces with different curvatures are inherently heterogeneous and lack a unified metric. Consequently, they lack explicit mechanisms to enforce geometric consistency across various spaces. Moreover, the problem of comparing probability distributions across mixed-curvature spaces remains unexplored. To compare distributions on heterogeneous spaces, Gromov–Wasserstein (GW) distances provide a principled framework by aligning their intra-space geometries. Building on this, we propose constant-curvature sliced Gromov–Wasserstein (CCSGW), a novel divergence for aligning distributions supported on heterogeneous constant-curvature spaces. We first introduce the missing geodesic-based one-dimensional projections for spherical spaces, and then extend sliced GW to constant-curvature spaces, enabling efficient and principled comparison across manifolds with different curvatures. This formulation preserves intrinsic geometric relationships while avoiding the high computational cost. We provide theoretical analysis showing that CCSGW controls intrinsic geometric discrepancy across heterogeneous spaces, promoting distribution-level geometric consistency. By integrating CCSGW into existing mixed-curvature learning tasks, including graph anomaly detection, graph node classification, and multimodal learning, we observe consistent performance gains across diverse settings.

1 INTRODUCTION

Euclidean geometry has long been the standard representation space in machine learning. Recently, non-Euclidean spaces, particularly hyperbolic and spherical spaces, have gained attention by providing better representations for certain types of structured data (Gu et al., 2018). Hyperbolic space is well-suited to hierarchical organization (Khrulkov et al., 2020; Peng et al., 2021; Li et al., 2026b), while spherical space effectively captures periodic patterns (Cohen et al., 2018; Meng et al., 2019). These spaces can be viewed as special cases of constant-curvature geometries, i.e., geometric spaces whose curvature is uniform everywhere. Hyperbolic, spherical, and Euclidean spaces are the three types of constant-curvature geometries, corresponding to negative, positive, and zero curvature, respectively (Lee, 2006).

Changing the geometry of the underlying latent space can lead to more expressive representations (Nickel & Kiela, 2017), with curvature playing a central role. However, many real-world datasets do not admit a single geometric structure (Peng et al., 2021). As a result, any single constant-curvature geometry, Euclidean, hyperbolic, or spherical, may be insufficient to capture all structural nuances simultaneously (Chlenski et al., 2025a). To model real-world data with complex topologies, various mixed-curvature models have been proposed that operate across multiple constant-curvature spaces (Skopek et al., 2020; Grover et al., 2025c; He et al., 2025; Grover et al., 2025a; Chen et al., 2025; Chlenski et al., 2025b; Gao et al., 2025; Gu et al., 2018). Mixed-curvature models partition the space into multiple component spaces and learn or fix a curvature for each part, aiming to capture the complex geometric topology (Gu et al., 2018).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

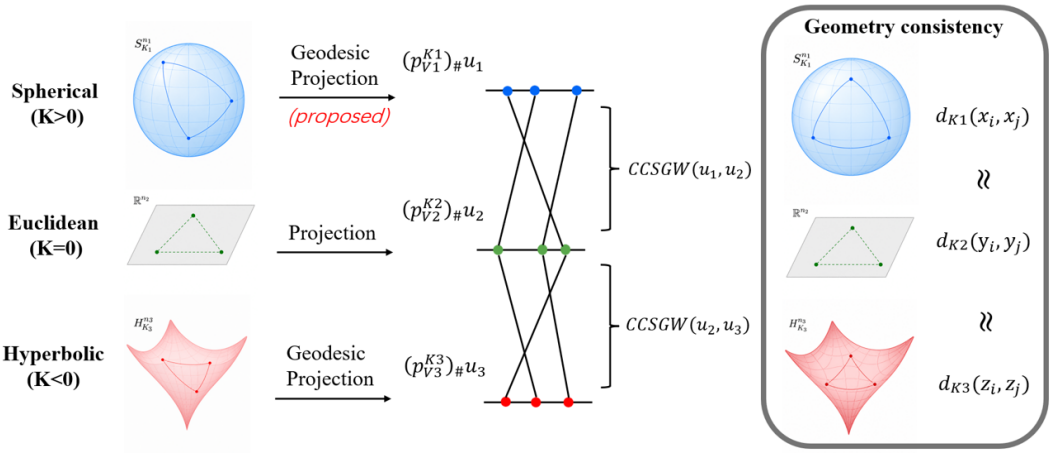


Figure 1: **CCSGW Overview.** Distributions supported on spherical, Euclidean, and hyperbolic spaces are projected onto one-dimensional coordinates using geometry-aware projections. CCSGW then compares the projected distributions across heterogeneous constant-curvature spaces, encouraging consistency of their intrinsic geometric structures despite differences in curvature.

However, these approaches typically learn each component space independently and lack an explicit mechanism to enforce geometric consistency across spaces (Gu et al., 2018). Enhancing geometric consistency imposes an additional inductive prior to regularize the representation learning, in cases such as knowledge distillation (Park et al., 2019), graph matching (Xu et al., 2019), and cross-lingual word matching (Alvarez-Melis & Jaakkola, 2018). This mismatch can lead to distortions of the global relational structure, hindering coherent representation learning and potentially degrading downstream performance. Since data are embedded into heterogeneous metric spaces with different curvatures, distances computed independently within each space are generally not directly comparable. To address this issue, a principled way to encourage geometric consistency across heterogeneous spaces is needed.

In this study, we propose constant-curvature sliced-Gromov Wasserstein (CCSGW), a new divergence for comparing distributions on heterogeneous constant-curvature spaces. CCSGW is an extension of sliced-Gromov–Wasserstein (SGW) (Vayer et al., 2019) to constant-curvature spaces, allowing explicit alignment across manifolds to enforce geometric consistency. SGW compares distriptions that do not lie in the same metric space based on their internal pairwise distance structures, while making the comparison computationally efficient by reducing it to a collection of one-dimensional GW problems on the real line. This property makes SGW particularly well suited to mixed-curvature representations, where components with different curvatures naturally constitute heterogeneous metric spaces and their intrinsic distances are not directly comparable across spaces. Building on this, we first derive geodesic-based one-dimensional projections for spherical spaces, completing the one-dimensional slicing framework across hyperbolic, Euclidean, and spherical geometries, and then introduce CCSGW. CCSGW enables meaningful comparison and alignment of distributions across mixed-curvature spaces to preserve distribution-level geometric consistency across mixed-curvature representations (Fig. 1). Through theoretical analysis, we show that CCSGW controls the intrinsic Gromov–Wasserstein discrepancy between heterogeneous constant-curvature spaces up to projection error. Therefore, minimizing CCSGW encourages approximately distance-preserving alignment for most sample pairs, providing a theoretical justification for distribution-level geometric consistency. Extensive experiments on simulations and mixed-curvature models for graph anomaly detection (Grover et al., 2025a), graph node classification (Grover et al., 2025c), and multimodal learning (Zhou et al., 2026) demonstrate that integrating CCSGW achieves superior performance.

We make three main contributions:

- We introduce CCSGW, a geometry-aware divergence for aligning distributions across heterogeneous constant-curvature spaces, together with geodesic-based one-dimensional projections for spherical spaces.

- We provide theoretical guarantees connecting CCSGW to intrinsic Gromov–Wasserstein discrepancy and approximate pairwise distance preservation.
- Experiments on graph anomaly detection, node classification, and multimodal learning show consistent improvements when CCSGW is integrated into existing mixed-curvature models.

2 PRELIMINARIES

2.1 RIEMANNIAN MANIFOLD

Constant curvature spaces. A constant-curvature space can be viewed as a smooth Riemannian manifold with curvature $K \in \mathbb{R}$. The K-radius model provides an extrinsic representation of such spaces (Skopek et al., 2020), including the sphere $\mathbb{S}^n_K$, Euclidean space $\mathbb{R}^n$, and hyperboloid $\mathbb{H}^n_K$:

$$\mathcal{M}^n_K = \begin{cases} \mathbb{S}^n_K = \left\{x \in \mathbb{R}^{n+1} : \|x\|^2 = \frac{1}{K}\right\} & \text{for } K > 0, \text{ spherical geometry,} \\ \mathbb{R}^n & \text{for } K = 0, \text{ Euclidean geometry,} \\ \mathbb{H}^n_K = \left\{x \in \mathbb{R}^{n+1} : \|x\|^2_L = \frac{1}{K},\ x_t > 0\right\} & \text{for } K < 0, \text{ hyperbolic geometry.} \end{cases} \tag{1}$$

where $\|x\|^2_L = \|x_s\|^2 - x_t^2$ is the Lorentz inner product. Following the conventions of the hyperboloid, we write $x = (x_t, x_s^\top)^\top$, with $x_t \in \mathbb{R}$ as the time component and $x_s \in \mathbb{R}^n$ as the spatial component (Ratcliffe, 2006).

A generalization of straight lines in Euclidean spaces to Riemannian manifolds is given by geodesics, which are smooth curves connecting two points $x, y \in \mathcal{M}$ with minimal length, i.e., continuous mappings $\gamma : [0,1] \to \mathcal{M}$ such that $\gamma(0) = x$, $\gamma(1) = y$. Following (Bonet et al., 2023b), we focus on geodesically complete Riemannian manifolds, in which case there is always a geodesic between two points $x, y \in \mathcal{M}$. Furthermore, in this specific case, all geodesics are actually geodesic lines, i.e., they can be extended to $\mathbb{R}$. Let $x, y \in \mathcal{M}$, and $\gamma : [0,1] \to \mathcal{M}$ be a geodesic between x and y such that $\gamma(0) = x$ and $\gamma(1) = y$. $\mathrm{PT}_{x \to y}$ denotes the parallel transport used to transport tangent vectors from the tangent space at x to the tangent space at y.

Mixed-curvature spaces. A mixed-curvature space is defined as the Cartesian product of constant-curvature manifolds (Skopek et al., 2020). Let each component $\mathcal{M}^{n_j}_{K_j}$ be a space of constant curvature K_j and dimension d_j. The product space is

$$\mathcal{M} = \prod_{j=1}^{m} \mathcal{M}^{n_j}_{K_j}. \tag{2}$$

For a point $u = (u_1, \ldots, u_m) \in \mathcal{M}$, tangent spaces and maps decompose component wise:

$$\begin{aligned} T_u\mathcal{M} &= \prod_{j=1}^{m} T_{u_j}\mathcal{M}^{n_j}_{K_j}, \\ \mathrm{Exp}^{\mathcal{M}}_u(q) &= \left(\exp^{K_1}_{u_1}(q_1), \ldots, \exp^{K_m}_{u_m}(q_m)\right), \\ \mathrm{Log}^{\mathcal{M}}_u(x) &= \left(\log^{K_1}_{u_1}(x_1), \ldots, \log^{K_m}_{u_m}(x_m)\right). \end{aligned} \tag{3}$$

2.2 THE SLICED-WASSERSTEIN

The Wasserstein distance is a powerful tool to compare probability distributions, but its computational cost can become prohibitive in high dimensions. A key observation is that in one dimension, the Wasserstein distance admits a closed-form solution that can be computed efficiently by sorting samples, leading to a complexity of $O(n \log n)$. The sliced Wasserstein (SW) distance leverages this property by projecting high-dimensional distributions onto one-dimensional subspaces. In practice, the integral is approximated using a finite number of random projections, leading to an efficient approximation with complexity $O(L\, n \log n)$, where L is the number of sampled directions.

The Sliced-Gromov Wasserstein. We first recall the definition of the Gromov–Wasserstein (GW) distance. Let $\mu \in \mathcal{P}(\mathbb{R}^p)$ and $\nu \in \mathcal{P}(\mathbb{R}^q)$ be two probability measures. The squared GW discrepancy

is defined as

$$GW_2^2(d^2, \mu, \nu) := \min_{\pi \in \Pi(\mu,\nu)} \int |d(x, x') - d(y, y')|^2 \, d\pi(x, y) \, d\pi(x', y'), \tag{4}$$

where $\Pi(\mu, \nu)$ denotes the set of couplings between μ and ν, and d is the Euclidean distance. Unlike classical optimal transport, which compares points across domains, the GW distance aligns distributions by matching their intrinsic geometries. More precisely, it seeks a coupling π such that pairwise distances within μ are preserved when mapped to ν, i.e., pairs of points (x, x') in the source are matched to pairs (y, y') in the target with similar distances. This allows comparing distributions supported on different spaces or of different dimensions.

The computational cost of equation 4 is prohibitive for large-scale problems ($O(n^3)$ in the discrete case). Inspired by the sliced Wasserstein (SW) distance, (Vayer et al., 2019) proposed the SGW discrepancy. Let $\mathbb{S}^{q-1}$ denote the unit sphere equipped with the uniform measure λ_{q-1}. Let $\mu \in \mathcal{P}(\mathbb{R}^p)$ and $\nu \in \mathcal{P}(\mathbb{R}^q)$, and let $\Delta \in \mathbb{R}^{q \times p}$ be a linear map. The SGW divergence is defined as

$$\mathrm{SGW}_\Delta(\mu, \nu) := \int_{\mathbb{S}^{q-1}} \mathrm{GW}_2^2 \left(d^2, \, (P_\theta)_\sharp \mu_\Delta, \, (P_\theta)_\sharp \nu\right) \, d\lambda_{q-1}(\theta), \tag{5}$$

where $P_\theta(x) := \langle x, \theta \rangle$ denotes the one-dimensional projection along $\theta \in \mathbb{S}^{q-1}$. Here, the inner GW is computed in one dimension, where it admits a closed-form solution based on sorting and evaluating only two candidate permutations (identity or anti-identity) (Vayer et al., 2019). This leads to a computational cost of $O(n \log n)$ per projection. In practice, the integral in equation 5 is approximated by averaging over L random directions, yielding an overall complexity of $O(L\, n \log n)$.

Slicing on geodesics. The Wasserstein distance can be naturally defined on a Riemannian manifold by replacing the Euclidean distance with the geodesic distance (Bonet et al., 2023b). More precisely, given a manifold $(\mathcal{M}, d)$, the optimal transport cost between two probability measures is defined using $d(x, y)$, the length of the shortest geodesic connecting x and y. This provides a natural extension of Wasserstein distances to curved spaces.

To generalize slicing to hyperbolic spaces, points are projected onto geodesics, which play the role of one-dimensional subspaces (Bonet et al., 2023b). Given a direction v, let γ_v denote the geodesic passing through the origin. The projection of a point x onto γ_v is defined as

$$P_v(x) := \arg\min_{y \in \gamma_v} d(x, y), \tag{6}$$

and its one-dimensional coordinate is given by the signed geodesic distance to the origin:

$$t_v(x) := \mathrm{sign}(\langle x, v \rangle) \, d(x, x_0), \tag{7}$$

where x_0 denotes the origin. This mapping yields a scalar representation on $\mathbb{R}$ for each point.

3 Methods

3.1 Sphere geodesic projections

Recent work has extended the sliced-Wasserstein framework to the sphere by projecting measures onto great circles and computing optimal transport directly on the circle (Bonet et al., 2023a). In particular, spherical sliced-Wasserstein relies on closed-form formulations of Wasserstein distances on $\mathcal{S}^1$, which require solving a shift-invariant optimal transport problem due to the periodic structure of the circle. In our work, we aim to project points onto geodesics and further map them to $\mathbb{R}$ via signed geodesic distances. This reduces the problem to standard one-dimensional Wasserstein distances, avoiding the need for circular optimal transport and shift alignment.

To extend slicing to spherical spaces, we derive geodesic projection onto a fixed geodesic on the Sphere (Radius Model). Let $K > 0$ and consider the sphere (radius model) $S_K^n = \{x \in \mathbb{R}^{n+1} : \|x\|^2 = 1/K\}$ For a fixed reference point $x_0 \in S_K^n$ and a fixed unit tangent vector $v \in T_{x_0} S_K^n$ with $\|v\| = 1$. The geodesic starting from x_0 in direction v is given by the exponential map

$$\gamma(t) = \mathrm{Exp}_{x_0}(tv) = \cos(\sqrt{K}\, t)\, x_0 + \frac{\sin(\sqrt{K}\, t)}{\sqrt{K}}\, v, \qquad t \in \mathbb{R}. \tag{8}$$

For any $x \in S_K^n$, we define the geodesic projection parameter $t^\star(x) \in \arg\min_{t\in\mathbb{R}} d(x, \gamma(t))$, where the geodesic distance on S_K^n is $d(x,y) = \frac{1}{\sqrt{K}} \arccos\big(K\langle x, y\rangle\big)$. Let $x \in S_K^n$, the optimal projection parameter $t^\star(x)$ satisfies the following proposition, which gives a closed-form solution.

Proposition 1. For $x \in \mathcal{S}_K^n$ with $\langle x, x_0\rangle \neq 0$, the one-dimensional projection coordinate is given by

$$t_v^*(x) = \frac{1}{\sqrt{K}} \operatorname{atan2}\left(\langle x, v\rangle, \sqrt{K}\langle x, x_0\rangle\right). \tag{9}$$

The proof and equatorial and degenerate cases can be handled similarly and are provided in Appendix Sec. A.

3.2 Constant-curvature sliced Gromov–Wasserstein

We now define a sliced Gromov–Wasserstein discrepancy for distributions supported on constant-curvature spaces with possibly different curvatures. Although the heterogeneous spaces and their metrics differ through the curvature parameter K, each distribution can be projected onto a one-dimensional geodesic coordinate, yielding a probability measure on the line $\mathbb{R}$.

Let $\mu \in \mathcal{P}(\mathcal{M}_{K_1}^{n_1})$ and $\nu \in \mathcal{P}(\mathcal{M}_{K_2}^{n_2})$, with $n_1 \le n_2$. At the origins, $T_{o_{K_j}}\mathcal{M}_{K_j}^{n_j} \cong \mathbb{R}^{n_j}$ for $j = 1, 2$. Following (Vayer et al., 2019), let $\Delta : \mathbb{R}^{n_1} \to \mathbb{R}^{n_2}$ be the zero-padding map $\Delta(u) = (u, 0_{n_2-n_1})$. For $\theta \in \mathbb{S}^{n_2-1}$, let $p_\theta^{K_j} : \mathcal{M}_{K_j}^{n_2} \to \mathbb{R}$ be the geodesic projection coordinate associated with the unit tangent vector represented by θ at the origin. Let λ_{n_2-1} be the uniform probability measure on $\mathbb{S}^{n_2-1}$. We define

$$\mathrm{CCSGW}_\Delta(\mu,\nu) = \int_{v\sim\lambda_{K_1}} \mathrm{GW}_2^2\Big(d_{\mathbb{R}}^2,\ (p_v^{K_1})_{\#}\mu,\ (p_{\Delta v}^{K_2})_{\#}\nu\Big)\, d\lambda_{n_1}(v), \tag{10}$$

Here, $d_{\mathbb{R}}(s,t) = |s-t|$, and both measures are sliced using the same direction θ after dimension alignment. When $K_1 = K_2 = 0$, this definition reduces to Euclidean SGW. When $n_1 = n_2$, the zero-padding map and its induced embedding are the identity.

3.3 Theoretical Analysis

We next connect the sliced objective in Eq. (10) to the intrinsic geometry of the original constant-curvature spaces. We show that CCSGW controls the intrinsic GW discrepancy up to an explicit error induced by geodesic projection. Since CCSGW averages discrepancies over one-dimensional projections, the resulting guarantee is naturally an average one: a cross-space map approximately preserves pairwise distances for most pairs, rather than uniformly over all pairs.

Let $\mathcal{X} = (X, d_X, \mu)$ and $\mathcal{Y} = (Y, d_Y, \nu)$ be two constant-curvature metric-measure spaces, with origins o_X and o_Y. As in Eq. (10), directions $v \sim \lambda_X$ and $w \sim \lambda_Y$ are drawn independently and uniformly from the unit tangent spheres at the origins, and p_v^X, p_w^Y denote the one-dimensional geodesic projection coordinates. Throughout, GW_2 denotes the square root of the discrepancy in Eq. (4) evaluated with the intrinsic geodesic distances, $\mathrm{GW}_2(\mathcal{X},\mathcal{Y}) = \inf_{\pi\in\Pi(\mu,\nu)} \|d_X(x,x') - d_Y(y,y')\|_{L_2(\pi\otimes\pi)}$, so that $S := \mathrm{CCSGW}(\mu,\nu) = \mathbb{E}_{v,w}\,\mathrm{GW}_2^2\left((p_v^X)_{\#}\mu, (p_w^Y)_{\#}\nu\right)$.

We assume that the distributions are supported within a finite radius R of the corresponding origins and, for a spherical component with curvature $K > 0$, that $R < \frac{\pi}{2\sqrt{K}}$. Under this condition the geodesic projection is unique and its signed coordinate is an isometry onto the real line: the difference of two coordinates equals the geodesic distance between the projected points. Without it, coordinates on a great circle can wrap around, and the sliced objective no longer reflects the spherical geometry. For $K \le 0$ geodesics do not wrap, and only bounded support is required. Let $P_v^X(x)$ denote the projection of x onto the geodesic through o_X in direction v, and define the average projection displacement

$$H_X := \left(\mathbb{E}_v \mathbb{E}_{x\sim\mu} d_X\big(x, P_v^X(x)\big)^2\right)^{1/2}, \tag{11}$$

with H_Y defined analogously using P_w^Y. Intuitively, H_X and H_Y quantify how much intrinsic geometric information may be lost by reducing the original spaces to one-dimensional geodesic slices, i.e. the information loss of slice projection.

Theorem 3.1. *Let $S := \mathrm{CCSGW}(\mu, \nu)$. Under the above conditions,*

$$\mathrm{GW}_2(\mathcal{X}, \mathcal{Y}) \le 2H_X + \sqrt{S} + 2H_Y.$$

Thm. 3.1 suggests that a small CCSGW value certifies a small intrinsic GW discrepancy whenever the projection displacements H_X and H_Y are small. These two terms are properties of the data and the projection family; they are explicit and measurable, but they are not reduced by minimizing S alone. The proof lifts, for each pair of directions, the optimal matching between the two one-dimensional slices to a matching between the original spaces, and bounds its distortion by the triangle inequality: the intrinsic discrepancy is at most the distortion introduced by projecting $\mathcal{X}$, plus the one-dimensional GW discrepancy computed by CCSGW, plus the distortion introduced by projecting $\mathcal{Y}$. Averaging over directions yields Thm. 3.1.

We further translate this distribution-level guarantee into a pairwise distance-preservation result, the average analogue of a Lipschitz bound used in (Li et al., 2026a). Let $\pi^\star \in \Pi(\mu, \nu)$ attain $\mathrm{GW}_2(\mathcal{X}, \mathcal{Y})$, let S_X be the X-projection of $\operatorname{supp} \pi^\star$, and assume

$$r := \inf_{\substack{x, x' \in S_X \\ x \neq x'}} d_X(x, x') > 0,$$

which holds, e.g., for the empirical measures used in training. For a measurable cross-space map $g : X \to Y$, define its L_2 matching error with respect to $\pi^\star$ as

$$\rho_{\pi^\star}(g) := \left(\int d_Y(g(x), y)^2 \, d\pi^\star(x, y) \right)^{1/2}.$$

Theorem 3.2. *Under the same assumptions, let $B := 2H_X + \sqrt{S} + 2H_Y + 2\rho_{\pi^\star}(g)$,*

$$\left(\int_{x \neq x'} \left| \frac{d_Y(g(x), g(x'))}{d_X(x, x')} - 1 \right|^2 d\pi^\star(x, y) \, d\pi^\star(x', y') \right)^{1/2} \le \frac{B}{r}.$$

Consequently, for every $\epsilon \in (0, 1)$, all pairs with $x \neq x'$ except a set of $\pi^\star \otimes \pi^\star$-mass at most ϵ satisfy

$$1 - \frac{B}{r\sqrt{\epsilon}} \le \frac{d_Y(g(x), g(x'))}{d_X(x, x')} \le 1 + \frac{B}{r\sqrt{\epsilon}}.$$

Thm. 3.1 and Thm. 3.2 give a direct geometric interpretation of the CCSGW objective. A small sliced discrepancy, together with small geodesic projection displacement and small matching error, implies that the cross-space map g is approximately distance-preserving on most sample pairs. Thus CCSGW enforces distribution-level geometric consistency across heterogeneous constant-curvature components, while allowing the components to retain different curvatures. The guarantee characterizes what a small objective certifies and is informative in the near-isometric regime $B < r$, later in experiment sections we will showcase the effectiveness of enforcing such consistency from empirical perspective.

4 Experiments

We conduct experiments to evaluate the effectiveness of CCSGW in promoting geometric consistency across heterogeneous constant-curvature spaces. We first perform a controlled simulation to visualize the cross-curvature alignment induced by CCSGW. Mixed-curvature models have achieved state-of-the-art performance across diverse tasks, including graph anomaly detection (Grover et al., 2025a), graph node classification (Grover et al., 2025c), and multimodal learning (Zhou et al., 2026). Building on these strong backbones, we incorporate CCSGW as an additional alignment objective in the training to explicitly enforce geometric consistency across heterogeneous component spaces.

4.1 Simulations

We first conduct a synthetic experiment to provide an intuitive visualization of cross-curvature alignment with CCSGW. Since the deformation of point configurations is more directly visualized in Euclidean space, we optimize a set of points in Euclidean space to align with a fixed target distribution

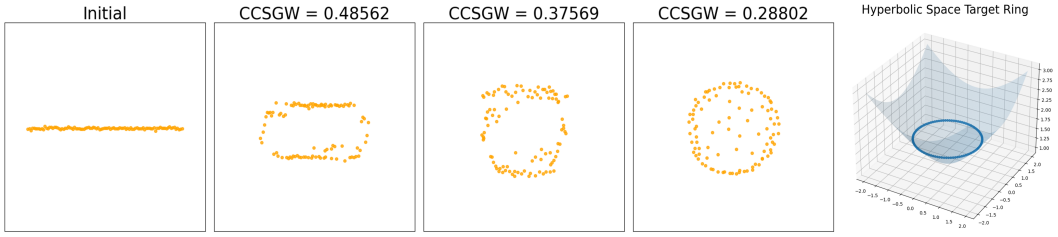


Figure 2: **Simulation of cross-curvature alignment.** A Euclidean point cloud initialized as a line is optimized to align with a fixed ring-shaped distribution in hyperbolic space using CCSGW. As optimization proceeds, the Euclidean points gradually form a circular structure while the CCSGW discrepancy decreases, illustrating improved geometric consistency.

Table 1: Graph anomaly detection performance (AUROC %(↑) ± 95% confidence interval). **First**, second, and third best are highlighted. Δ **CCSGW** is the relative gain of CCSGW.

Model	Reddit	Weibo	Amazon	Elliptic	Questions
GCN	65.25±0.89	97.03±0.57	82.48±0.43	81.42±1.83	70.93±1.34
GAT	65.38±0.19	95.35±1.47	96.75±1.06	84.93±1.98	71.17±1.63
GraphSAGE	62.86±0.14	94.94±1.37	89.65±5.42	85.35±0.72	72.15±1.73
BernNet	65.82±1.75	94.96±1.56	96.21±1.43	83.18±1.49	68.82±2.98
ChebyNet	64.11±1.49	95.02±0.93	96.59±2.01	79.34±3.07	69.23±1.55
HGCN	64.13±0.98	94.23±1.86	79.94±0.93	78.66±2.22	67.16±1.74
HGAT	63.45±0.23	92.24±0.98	85.98±3.65	85.45±0.32	68.93±1.27
κGCN	61.77±1.54	97.87±1.24	97.34±1.19	76.13±2.13	73.15±3.84
$\mathcal{Q}$GCN	66.45±1.12	97.34±0.34	96.71±5.93	85.55±2.03	71.74±2.26
DCI	66.54±3.38	94.25±1.73	94.63±0.92	82.86±1.51	69.21±1.34
PCGNN	53.25±2.19	90.27±1.51	97.31±0.82	85.89±1.85	69.91±1.43
BWGNN	65.43±4.35	97.38±0.93	97.05±0.76	85.29±1.14	71.85±2.23
DOMINANT	62.32±2.65	87.43±2.01	76.23±5.99	79.34±0.94	64.92±1.74
AnomalyDAE	65.75±1.63	93.54±1.88	91.99±0.97	83.69±0.29	62.45±0.66
GADNR	67.01±1.02	96.92±0.43	93.46±2.03	82.57±0.84	70.11±0.06
ADAGAD	66.45±0.99	98.01±0.01	96.05±1.30	86.01±0.45	71.06±1.04
CurvGAD	67.76±1.54	99.04±0.34	97.50±0.15	83.90±1.25	73.89±0.81
CurvGAD + CCSGW	**70.55±1.12**	**99.27±0.09**	**97.73±0.20**	**89.01±0.46**	**76.40±0.70**
Δ **(CCSGW)**	2.79	0.23	0.23	5.11	2.51

in hyperbolic space using CCSGW as the alignment objective. The Euclidean points are initialized along a line, while the target distribution forms a ring-like structure in the hyperbolic space.

As optimization proceeds (Fig. 2), the Euclidean point cloud gradually deforms from the initial line into a circular configuration that better matches the intrinsic relational structure of the hyperbolic target. Meanwhile, the CCSGW discrepancy consistently decreases, indicating that the pairwise geometric structures of the two distributions become increasingly aligned despite being supported on spaces with different curvatures. This simulation provides a direct illustration that minimizing CCSGW can promote distribution-level geometric consistency across heterogeneous spaces.

4.2 Main Experiments

Graph anomaly detection. Wegrate CCSGW into CurvGAD (Grover et al., 2025a), a state-of-the-art graph anomaly detector based on mixed-curvature representations, and evaluate on five GAD-Bench datasets (Tang et al., 2023): Reddit (Kumar et al., 2019; Liu et al., 2022), Weibo (Zhao et al., 2020), Amazon (McAuley & Leskovec, 2013), Elliptic (Weber et al., 2019) and Questions (Platonov et al., 2023). CCSGW aligns the constant-curvature components of CurvGAD's product-manifold embedding. At every epoch, CCSGW draws 256 random nodes and, for each constant-curvature component, projects their embeddings along random geodesic directions (number of directions

Table 2: Node classification performance (weighted F1-score %(↑) ± 95% confidence interval). **First**, second, and third best are highlighted. The last row reports the gain of CCSGW.

	Homophilic			Heterophilic	
Method	Cora	CiteSeer	PubMed	Chameleon	Actor
GCN	75.21±0.28	67.30±1.05	83.75±0.07	61.16±0.23	31.12±0.96
GAT	76.70±0.13	66.23±0.85	82.83±0.22	63.10±0.77	32.65±0.23
SAGE	71.88±0.91	70.01±0.64	81.09±0.13	59.99±0.89	36.73±0.01
HGCN	78.50±0.14	69.55±0.39	83.72±0.21	60.18±0.57	35.89±0.29
HGAT	77.12±0.01	70.12±0.92	84.02±0.19	62.43±0.59	35.12±0.27
κGCN	78.71±1.37	68.14±0.34	85.18±0.52	62.12±0.49	34.57±0.26
$\mathcal{Q}$GCN	79.64±0.38	71.15±1.11	84.76±0.13	61.83±1.01	32.24±0.65
SelfMGNN	80.19±0.60	70.91±0.38	82.81±0.34	64.97±0.54	38.99±0.23
ChebyNet	71.09±0.91	66.67±0.38	83.83±0.42	59.96±1.51	38.02±0.01
BernNet	73.34±0.53	62.12±2.09	82.15±0.13	62.03±0.12	33.55±0.24
GPRGNN	79.49±0.31	67.61±0.38	84.07±0.09	65.09±0.43	37.43±1.09
FiGURe	80.01±0.09	71.26±0.41	83.89±0.11	67.18±0.02	38.31±0.36
H2GCN	82.70±0.90	71.10±0.80	84.60±0.50	60.10±2.20	35.90±1.20
FAGCN	82.50±0.50	71.30±0.60	84.30±0.40	66.90±1.80	35.80±1.10
CurvDrop	82.50±0.70	72.80±0.60	85.20±0.50	67.30±1.40	39.50±1.00
ARGNN	86.65±1.00	**76.29±0.89**	90.15±0.25	65.58±2.05	38.82±0.92
CUSP	85.64±1.48	74.10±0.88	90.34±0.21	66.48±2.21	38.90±0.93
CUSP + CCSGW	**86.66±0.93**	74.24±1.23	**90.39±0.18**	**67.82±1.92**	**39.83±0.70**
Δ (CCSGW)	+1.02	+0.14	+0.06	+1.34	+0.93

$L = 32$) under the current curvature, which is kept fixed within this term. The resulting slices are standardized and averaged over directions and component pairs. This averaged discrepancy is added to the CurvGAD objective with the alignment weight $\lambda = 10$, shared across all datasets. Following the supervised protocol, we report test AUROC (mean ± 95% CI) over ten splits. We train the model with and without CCSGW in the training objective using all hyperparameters in Grover et al. (2025a). The results are summarized in Tab. 1, and the remaining baselines are adopted from Grover et al. (2025a), with further details provided in Sec. C.1.

Graph node classification. We integrate CCSGW into CUSP (Grover et al., 2025b), a state-of-the-art spectral graph neural network on mixed-curvature product manifolds, and evaluate the resulting model on three homophilic datasets, Cora, CiteSeer, and PubMed (Sen et al., 2008), and two heterophilic datasets, Chameleon (Pei et al., 2020) and Actor (Tang et al., 2009). As in CurvGAD, CCSGW aligns the constant-curvature components of CUSP's product-manifold node embeddings. Following CUSP (Grover et al., 2025b), we report the mean and 95% confidence interval of test weighted F1 at the best validation epoch over ten random 60/20/20 splits. We train the model with and without CCSGW in the training objective using all hyperparameters in Grover et al. (2025b). The results are summarized in Tab. 2, and the remaining baselines are adopted from Grover et al. (2025b), with further details provided in Sec. C.2.

EEG-based multimodal learning. We integrate CCSGW into EEG-MoCE (Zhou et al., 2026), a state-of-the-art multimodal learning framework based on hyperbolic mixture-of-curvature experts, and evaluate the resulting model on the emotion recognition task (Lee et al., 2024) and two cognitive tasks (Shin et al., 2018). For CCSGW alignment, we select one modality-specific hyperbolic manifold as the reference and align the embedding distribution of each remaining modality with that of the reference modality. Following Zhou et al. (2026), we perform subject-wise 10-fold cross-validation on EAV ($n = 42$ participants) and Cognitive ($n = 26$ participants), and report test balanced accuracy averaged over folds. We train the model with and without CCSGW in the training objective using all hyperparameters in Zhou et al. (2026). The results are summarized in Tab. 3, and the remaining baselines are adopted from Zhou et al. (2026), with details in Sec. C.3.

Table 3: Performance comparison on the emotion recognition and the Cognitive tasks. We report balanced accuracy. EAV uses cross-subject 10-fold cross-validation, while the Cognitive tasks use cross-subject 10-fold cross-validation. **First**, second, and third best are highlighted.

Emotion		**Cognitive**		
Method	Emotion (%)	Method	N-back	Word Generation
AGF-IB	54.59±12.45	CTMWA	49.86±11.72	59.87±8.15
CMERC	58.57±11.95	EFDFNet	49.15±12.85	58.85±7.63
CTMWA	52.86±5.63	EF-Net	54.41±14.03	64.53±11.08
GA2MIF	61.10±12.51	LMF	52.71±11.77	59.74±6.18
HEEGNet	61.74±8.33	MMML	45.32±9.37	56.35±8.22
Hyper-MML	60.76±12.15	STA-Net	46.63±11.67	62.50±8.72
LMF	57.29±12.37	ST2A	51.28±11.74	67.27±10.45
MMML	59.60±11.83	TSMMF	53.84±13.62	63.82±9.47
EEGMOCE	76.45±9.82	EEGMOCE	60.26±17.29	71.15±11.90
EEGMOCE + CCSGW	**77.14±9.72**	EEGMOCE + CCSGW	**61.68±17.24**	**72.76±13.48**
Δ **(CCSGW)**	+0.69	Δ **(CCSGW)**	+1.42	+1.61

Results. CCSGW consistently improves the mean performance of all three backbone models across the evaluated datasets and tasks. For graph anomaly detection (Tab. 1), it improves CurvGAD by up to 5.11 percentage points in AUROC on Elliptic. For graph learning (Tab. 2), CUSP improves weighted F1 by 1.02 and 1.34 percentage points on Cora and Chameleon, respectively. For EEG-based multimodal learning (Tab. 3), CCSGW increases balanced accuracy by 0.69, 1.42, and 1.61 percentage points on emotion recognition, N-back, and word generation, respectively. These consistent improvements support the effectiveness of CCSGW as a distribution-level alignment regularizer for heterogeneous cross-curvature representations. Moreover, the gains obtained while retaining the backbone architectures and manifold configurations suggest that explicitly aligning component distributions complements the representational flexibility of mixed-curvature models. Specifically, CCSGW encourages consistency in the relational structure of embedding distributions while allowing each component to retain its own curvature. However, the varying improvements across datasets also suggest that the benefit of such alignment may depend on the quality of the sampled geodesic projections, which may not fully preserve the geometry of the original high-dimensional manifolds. In the appendix, we further show that these gains are robust to the alignment weight, the number of projection directions and the number of sampled nodes (Sec. D.1). Additionally, using CUSP dataset as an example, we show that CCSGW outperforms SGW and matches GW in accuracy (Sec. D.2), and its cost is the same as that of SGW (Sec. D.3).

5 Discussion

In this work, we proposed CCSGW, a geometry-aware divergence for aligning distributions across heterogeneous constant-curvature spaces. We first introduce geodesic-based one-dimensional projections for spherical spaces and extending sliced GW to mixed-curvature settings. Our theoretical analysis shows that minimizing CCSGW controls the intrinsic GW discrepancy up to projection error, providing a principled connection between the sliced objective and distribution-level geometric consistency. Empirically, incorporating CCSGW into mixed-curvature models consistently improves performance across graph anomaly detection, node classification, and multimodal learning, suggesting that cross-curvature consistency can serve as a useful inductive bias across different architectures and tasks. The limitation of CCSGW is that its effectiveness depends on the quality of the sampled geodesic projections, which may not fully preserve the geometry of the original high-dimensional manifolds. Future work could therefore explore subspace alignment for more robust cross-curvature matching by preserving richer geometric information beyond one-dimensional projections. Extending CCSGW beyond constant-curvature product spaces to more general heterogeneous manifolds is another promising direction, potentially providing a broader framework for geometry-aware representation alignment.

AI USE STATEMENT

In this work, we used generative AI tools for polishing the grammar and writing, coding assistance, as well as assisting theoretical analysis. We have not used generative AI tools for drafting the paper or generating ideas. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

David Alvarez-Melis and Tommi Jaakkola. Gromov-wasserstein alignment of word embedding spaces. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 1881–1890, 2018.

Aniqa Arif, Yihe Wang, Rui Yin, Xiang Zhang, and Ahmed Helmy. EF-Net: Mental state recognition by analyzing multimodal EEG-fNIRS via CNN. *Sensors*, 24(6):1889, 2024. doi: 10.3390/s24061889.

Gregor Bachmann, Gary Bécigneul, and Octavian Ganea. Constant curvature graph convolutional networks. In *International conference on machine learning*, pp. 486–496. PMLR, 2020.

Deyu Bo, Xiao Wang, Chuan Shi, and Huawei Shen. Beyond low-frequency information in graph convolutional networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 3950–3957, 2021.

Clément Bonet, Paul Berg, Nicolas Courty, François Septier, Lucas Drumetz, and Minh-Tan Pham. Spherical sliced-wasserstein. In *International Conference on Learning Representations (ICLR)*, 2023a.

Clément Bonet, Laetitia Chapel, Lucas Drumetz, and Nicolas Courty. Hyperbolic sliced-wasserstein via geodesic and horospherical projections. In *Topological, Algebraic and Geometric Learning Workshops 2023*, pp. 334–370. PMLR, 2023b.

Clément Bonet, Lucas Drumetz, and Nicolas Courty. Sliced-wasserstein distances and flows on cartan-hadamard manifolds. *Journal of Machine Learning Research*, 26(32):1–76, 2025.

Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems*, 32, 2019.

Tianyu Chen, Xingcheng Fu, Yisen Gao, Haodong Qian, Yuecen Wei, Kun Yan, Haoyi Zhou, and Jianxin Li. Galaxy walker: Geometry-aware vlms for galaxy-scale understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 4112–4121, 2025.

Eli Chien, Jianhao Peng, Pan Li, and Olgica Milenkovic. Adaptive universal generalized pagerank graph neural network. *arXiv preprint arXiv:2006.07988*, 2020.

Philippe Chlenski, Quentin Chu, Raiyan R Khan, Kaizhu Du, Antonio Khalil Moretti, and Itsik Pe'er. Mixed-curvature decision trees and random forests. In *Forty-second International Conference on Machine Learning*, 2025a.

Philippe Chlenski, Quentin Chu, Raiyan R Khan, Kaizhu Du, Antonio Khalil Moretti, and Itsik Pe'Er. Mixed-curvature decision trees and random forests. In *International Conference on Machine Learning*, pp. 10469–10498. PMLR, 2025b.

Taco S Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical cnns. In *International Conference on Learning Representations*, 2018.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29, 2016.

Kaize Ding, Jundong Li, Rohit Bhanushali, and Huan Liu. Deep anomaly detection on attributed networks. In *Proceedings of the 2019 SIAM international conference on data mining*, pp. 594–602. SIAM, 2019.

Chanakya Ekbote, Ajinkya Deshpande, Arun Iyer, Sundararajan Sellamanickam, and Ramakrishna Bairi. Figure: Simple and efficient unsupervised node representations with filter augmentations. *Advances in Neural Information Processing Systems*, 36:35403–35425, 2023.

Haoyi Fan, Fengbin Zhang, and Zuoyong Li. Anomalydae: Dual autoencoder for anomaly detection on attributed networks. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5685–5689. IEEE, 2020.

Yuxiao Gao, Fuwei Zhang, Zhao Zhang, Xiaoshuang Min, and Fuzhen Zhuang. Mixed-curvature multi-modal knowledge graph completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 11699–11707, 2025.

Karish Grover, Geoffrey J Gordon, and Christos Faloutsos. Curvgad: Leveraging curvature for enhanced graph anomaly detection. In *International Conference on Machine Learning*, pp. 20429–20447. PMLR, 2025a.

Karish Grover, Haiyang Yu, Xiang Song, Qi Zhu, Han Xie, Vassilis N. Ioannidis, and Christos Faloutsos. Spectro-riemannian graph neural networks, 2025b. URL https://arxiv.org/abs/2502.00401.

Karish Grover, Haiyang Yu, Qi Zhu, Han Xie, Vassilis N Ioannidis, Christos Faloutsos, et al. Spectro-riemannian graph neural networks. In *The Thirteenth International Conference on Learning Representations*, 2025c.

Albert Gu, Frederic Sala, Beliz Gunel, and Christopher Ré. Learning mixed-curvature representations in product spaces. In *International conference on learning representations*, 2018.

Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

Junwei He, Qianqian Xu, Yangbangyan Jiang, Zitai Wang, and Qingming Huang. Ada-gad: Anomaly-denoised autoencoders for graph anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 8481–8489, 2024.

Mingguo He, Zhewei Wei, Hongteng Xu, et al. Bernnet: Learning arbitrary graph spectral filters via bernstein approximation. *Advances in neural information processing systems*, 34:14239–14251, 2021.

Neil He, Rishabh Anand, Hiren Madhu, Ali Maatouk, Smita Krishnaswamy, Leandros Tassiulas, Menglin Yang, and Rex Ying. Helm: Hyperbolic large language models via mixture-of-curvature experts. *arXiv preprint arXiv:2505.24722*, 2025.

Zijian Kang, Yueyang Li, Shengyu Gong, Weiming Zeng, Hongjie Yan, Lingbin Bian, Zhiguo Zhang, Wai Ting Siok, and Nizhuan Wang. Hypergraph multi-modal learning for EEG-based emotion recognition in conversation. *arXiv preprint arXiv:2502.21154*, 2025.

Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic image embeddings. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6418–6428, 2020.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Srijan Kumar, Xikun Zhang, and Jure Leskovec. Predicting dynamic embedding trajectory in temporal interaction networks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 1269–1278, 2019.

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

Min Ho Lee, Adai Shomanov, Balgyn Begim, Zhuldyz Kabidenova, Aruna Nyssanbay, Adnan Yazici, and Seong-Whan Lee. EAV: EEG-audio-video dataset for emotion recognition in conversational contexts. *Scientific Data*, 11(1):1026, 2024.

Jiang Li, Xiaoping Wang, Guoqing Lv, and Zhigang Zeng. GA2MIF: Graph and attention based two-stage multi-source information fusion for conversational emotion detection. *IEEE Transactions on Affective Computing*, 15(1):130–143, 2024. doi: 10.1109/TAFFC.2023.3261279.

Muyang Li, Yucheng Liu, Jianbo Ma, Elliot Osborne, Bo Han, and Tongliang Liu. Rethinking model selection in VLM through the lens of Gromov–Wasserstein distance, 2026a. URL https://arxiv.org/abs/2605.01325.

Shanglin Li, Shiwen Chu, Okan Koç, Yi Ding, Qibin Zhao, Motoaki Kawanabe, and Ziheng Chen. Heegnet: Hyperbolic embeddings for eeg. In *International Conference on Learning Representations*, 2026b.

Kay Liu, Yingtong Dou, Yue Zhao, Xueying Ding, Xiyang Hu, Ruitong Zhang, Kaize Ding, Canyu Chen, Hao Peng, Kai Shu, et al. Bond: Benchmarking unsupervised outlier node detection on static attributed graphs. *Advances in neural information processing systems*, 35:27021–27035, 2022.

Mutian Liu, Banghua Yang, Lin Meng, Yonghuai Zhang, Shouwei Gao, Peng Zan, and Xinxing Xia. STA-Net: Spatial–temporal alignment network for hybrid EEG-fNIRS decoding. *Information Fusion*, 119:103023, 2025. doi: 10.1016/j.inffus.2025.103023.

Yang Liu, Xiang Ao, Zidi Qin, Jianfeng Chi, Jinghua Feng, Hao Yang, and Qing He. Pick and choose: a gnn-based imbalanced learning approach for fraud detection. In *Proceedings of the web conference 2021*, pp. 3168–3177, 2021.

Yang Liu, Chuan Zhou, Shirui Pan, Jia Wu, Zhao Li, Hongyang Chen, and Peng Zhang. Curvdrop: A ricci curvature based approach to prevent graph neural networks from over-smoothing and over-squashing. In *Proceedings of the ACM web conference 2023*, pp. 221–230, 2023.

Zhun Liu, Ying Shen, Varun Bharadwaj Lakshminarasimhan, Paul Pu Liang, Amir Zadeh, and Louis-Philippe Morency. Efficient low-rank multimodal fusion with modality-specific factors. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pp. 2247–2256, 2018.

Julian John McAuley and Jure Leskovec. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In *Proceedings of the 22nd international conference on World Wide Web*, pp. 897–908, 2013.

Facundo Mémoli. Gromov–wasserstein distances and the metric approach to object matching. *Foundations of Computational Mathematics*, 11(4):417–487, 2011.

Yu Meng, Jiaxin Huang, Guangyuan Wang, Chao Zhang, Honglei Zhuang, Lance Kaplan, and Jiawei Han. Spherical text embedding. *Advances in neural information processing systems*, 32, 2019.

Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. *Advances in neural information processing systems*, 30, 2017.

Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3967–3976, 2019.

Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, and Bo Yang. Geom-gcn: Geometric graph convolutional networks. *arXiv preprint arXiv:2002.05287*, 2020.

Wei Peng, Tuomas Varanka, Abdelrahman Mostafa, Henglin Shi, and Guoying Zhao. Hyperbolic deep neural networks: A survey. *IEEE Transactions on pattern analysis and machine intelligence*, 44(12):10023–10044, 2021.

Oleg Platonov, Denis Kuznedelev, Michael Diskin, Artem Babenko, and Liudmila Prokhorenkova. A critical look at the evaluation of gnns under heterophily: Are we really making progress? *arXiv preprint arXiv:2302.11640*, 2023.

John G Ratcliffe. *Foundations of hyperbolic manifolds*. Springer, 2006.

Amit Roy, Juan Shu, Jia Li, Carl Yang, Olivier Elshocht, Jeroen Smeets, and Pan Li. Gad-nr: Graph anomaly detection via neighborhood reconstruction. In *Proceedings of the 17th ACM international conference on web search and data mining*, pp. 576–585, 2024.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Gallagher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93–106, 2008.

Xingbin Shi, Haiyan Wang, Baojiang Li, Yuxin Qin, Cheng Peng, and Yifan Lu. Fusion analysis of EEG-fNIRS multimodal brain signals: A multitask classification algorithm incorporating spatial-temporal convolution and dual attention mechanisms. *IEEE Transactions on Instrumentation and Measurement*, 74:2506312, 2025. doi: 10.1109/TIM.2025.3538086.

Jaeyoung Shin, Alexander von Lühmann, Do-Won Kim, Jan Mehnert, Han-Jeong Hwang, and Klaus-Robert Müller. Simultaneous acquisition of EEG and NIRS during cognitive tasks for an open access dataset. *Scientific Data*, 5(1):180003, 2018.

Yuntao Shou, Tao Meng, Wei Ai, Fuchen Zhang, Nan Yin, and Keqin Li. Adversarial alignment and graph fusion via information bottleneck for multimodal emotion recognition in conversations. *Information Fusion*, 112:102590, 2024.

Xiaopeng Si, Shuai Zhang, Zhuobin Yang, Jiayue Yu, and Dong Ming. A bidirectional cross-modal transformer representation learning model for EEG-fNIRS multimodal affective BCI. *Expert Systems with Applications*, 266:126081, 2025. doi: 10.1016/j.eswa.2024.126081.

Ondrej Skopek, Octavian-Eugen Ganea, and Gary Bécigneul. Mixed-curvature variational autoencoders. In *8th international conference on learning representations (ICLR 2020)(virtual)*. International Conference on Learning Representations, 2020.

Li Sun, Zhongbao Zhang, Junda Ye, Hao Peng, Jiawei Zhang, Sen Su, and Philip S Yu. A self-supervised mixed-curvature graph neural network. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 4146–4155, 2022.

Jianheng Tang, Jiajin Li, Ziqi Gao, and Jia Li. Rethinking graph neural networks for anomaly detection. In *International conference on machine learning*, pp. 21076–21089. PMLR, 2022.

Jianheng Tang, Fengrui Hua, Ziqi Gao, Peilin Zhao, and Jia Li. Gadbench: Revisiting and benchmarking supervised graph anomaly detection. *Advances in Neural Information Processing Systems*, 36:29628–29653, 2023.

Jie Tang, Jimeng Sun, Chi Wang, and Zi Yang. Social influence analysis in large-scale networks. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 807–816, 2009.

Geng Tu, Feng Xiong, Bin Liang, Hui Wang, Xi Zeng, and Ruifeng Xu. Multimodal emotion recognition calibration in conversations. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024.

Titouan Vayer, Rémi Flamary, Romain Tavenard, Laetitia Chapel, and Nicolas Courty. Sliced gromov-wasserstein. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

Xudong Wang, Chris Ding, Tongxin Li, and Jicong Fan. Adaptive riemannian graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 26606–26614, 2026.

Yanling Wang, Jing Zhang, Shasha Guo, Hongzhi Yin, Cuiping Li, and Hong Chen. Decoupling representation learning and classification for gnn-based anomaly detection. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pp. 1239–1248, 2021.

Mark Weber, Giacomo Domeniconi, Jie Chen, Daniel Karl I Weidele, Claudio Bellei, Tom Robinson, and Charles E Leiserson. Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics. *arXiv preprint arXiv:1908.02591*, 2019.

Zehui Wu, Ziwei Gong, Jaywon Koo, and Julia Hirschberg. Multimodal multi-loss fusion network for sentiment analysis. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3588–3602, Mexico City, Mexico, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.197. URL https://aclanthology.org/2024.naacl-long.197/.

Bo Xiong, Shichao Zhu, Nico Potyka, Shirui Pan, Chuan Zhou, and Steffen Staab. Pseudo-riemannian graph convolutional networks. *Advances in neural information processing systems*, 35:3488–3501, 2022.

Haiqi Xu, Qingshan She, Ming Meng, Yunyuan Gao, and Yingchun Zhang. EFDFNet: A multimodal deep fusion network based on feature disentanglement for attention state classification. *Biomedical Signal Processing and Control*, 109:108042, 2025. doi: 10.1016/j.bspc.2025.108042.

Hongteng Xu, Dixin Luo, Hongyuan Zha, and Lawrence Carin Duke. Gromov-wasserstein learning for graph matching and node embedding. In *International conference on machine learning*, pp. 6932–6941. PMLR, 2019.

Baozheng Zhang, Ziqi Yuan, Hua Xu, and Kai Gao. Crossmodal translation based meta weight adaption for robust image-text sentiment analysis. *IEEE Transactions on Multimedia*, 26:9949–9961, 2024. doi: 10.1109/TMM.2024.3405662.

Yiding Zhang, Xiao Wang, Chuan Shi, Xunqiang Jiang, and Yanfang Ye. Hyperbolic graph attention network. *IEEE Transactions on Big Data*, 8(6):1690–1701, 2021.

Tong Zhao, Chuchen Deng, Kaifeng Yu, Tianwen Jiang, Daheng Wang, and Meng Jiang. Error-bounded graph anomaly loss for gnns. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pp. 1873–1882, 2020.

Runhe Zhou, Shanglin Li, Guanxiang Huang, Xinliang Zhou, Qibin Zhao, Motoaki Kawanabe, Yi Ding, and Cuntai Guan. Eeg-based multimodal learning via hyperbolic mixture-of-curvature experts. *arXiv preprint arXiv:2604.12579*, 2026.

Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. Beyond homophily in graph neural networks: Current limitations and effective designs. *Advances in neural information processing systems*, 33:7793–7804, 2020.

A SPHERE GEODESIC PROJECTIONS

Geodesic Projection onto a Fixed Geodesic on the Sphere (Radius Model)

1. Geometric Setting

Let $K > 0$ and consider the sphere (radius model)

$$S_K^n = \{x \in \mathbb{R}^{n+1} : \|x\|^2 = 1/K\}.$$

Fix: - a reference point $x_0 \in S_K^n$, - a unit tangent vector $v \in T_{x_0} S_K^n$ with $\|v\| = 1$.

The geodesic starting from x_0 in direction v is given by the exponential map

$$\gamma(t) = \mathrm{Exp}_{x_0}(tv) = \cos(\sqrt{K}\, t)\, x_0 + \frac{\sin(\sqrt{K}\, t)}{\sqrt{K}}\, v, \qquad t \in \mathbb{R}.$$

For any $x \in S_K^n$, define the geodesic projection parameter

$$t^\star(x) \in \arg\min_{t \in \mathbb{R}} d(x, \gamma(t)).$$

where the geodesic distance on S_K^n is

$$d(x, y) = \frac{1}{\sqrt{K}} \arccos\big(K\langle x, y\rangle\big).$$

2. Theorem (Closed-form Geodesic Projection Parameter)

Let $x \in S_K^n$. The optimal projection parameter $t^\star(x)$ satisfies the following case-wise characterization.

Case 1: Generic case $\langle x, x_0\rangle \neq 0$

If $\langle x, x_0\rangle \neq 0$, then any optimal parameter $t^\star(x)$ satisfies

$$\tan(\sqrt{K}\, t^\star(x)) = \frac{\langle x, v\rangle}{\sqrt{K}\, \langle x, x_0\rangle}.$$

A consistent choice of branch is

$$t^\star(x) = \frac{1}{\sqrt{K}} \operatorname{atan2}\big(\langle x, v\rangle,\ \sqrt{K}\, \langle x, x_0\rangle\big).$$

—

Case 2: Equatorial case $\langle x, x_0\rangle = 0$ and $\langle x, v\rangle \neq 0$

If $\langle x, x_0\rangle = 0$ and $\langle x, v\rangle \neq 0$, then an optimal projection parameter is

$$t^\star(x) = \frac{1}{\sqrt{K}} \arcsin\big(\mathrm{sign}(\langle x, v\rangle)\big).$$

$$t^\star(x) = \frac{1}{\sqrt{K}} \begin{cases} \frac{\pi}{2}, & \langle x, v\rangle > 0, \\ -\frac{\pi}{2}, & \langle x, v\rangle < 0. \end{cases}$$

Case 3: Degenerate case $\langle x, x_0\rangle = 0$ and $\langle x, v\rangle = 0$

If $\langle x, x_0\rangle = 0$ and $\langle x, v\rangle = 0$, then

$$\langle x, \gamma(t)\rangle = 0, \qquad \text{for all } t \in \mathbb{R}.$$

and the projection is not unique: every $t \in \mathbb{R}$ is optimal.

3.Proof

Step 1: Rewrite the objective using the explicit distance

By definition,

$$t^\star(x) \in \arg\min_{t\in\mathbb{R}} d(x,\gamma(t)).$$

Using

$$d(x,y) = \frac{1}{\sqrt{K}}\,\arccos\big(K\langle x,y\rangle\big),$$

we have

$$d(x,\gamma(t)) = \frac{1}{\sqrt{K}}\,\arccos\big(K\langle x,\gamma(t)\rangle\big).$$

Since the factor $1/\sqrt{K}$ is positive and does not depend on t, we may drop it inside $\arg\min$:

$$t^\star(x) \in \arg\min_{t\in\mathbb{R}} \arccos\big(K\langle x,\gamma(t)\rangle\big).$$

Step 2: Use monotonicity of $\arccos$

On $[-1,1]$, $\arccos(\cdot)$ is strictly decreasing. Therefore, minimizing $\arccos(\cdot)$ is equivalent to maximizing its argument. Hence,

$$t^\star(x) \in \arg\max_{t\in\mathbb{R}} K\langle x,\gamma(t)\rangle.$$

Since $K>0$ is constant, maximizing $K\langle x,\gamma(t)\rangle$ is equivalent to maximizing $\langle x,\gamma(t)\rangle$:

$$t^\star(x) \in \arg\max_{t\in\mathbb{R}} \langle x,\gamma(t)\rangle.$$

Step 3: Expand the objective $\langle x,\gamma(t)\rangle$

Recall

$$\gamma(t) = \cos(\sqrt{K}\,t)\,x_0 + \frac{\sin(\sqrt{K}\,t)}{\sqrt{K}}\,v.$$

Then, by bilinearity of the Euclidean inner product,

$$\langle x,\gamma(t)\rangle = \left\langle x, \cos(\sqrt{K}\,t)\,x_0 + \frac{\sin(\sqrt{K}\,t)}{\sqrt{K}}\,v \right\rangle.$$

Pulling out scalar coefficients,

$$\langle x,\gamma(t)\rangle = \cos(\sqrt{K}\,t)\,\langle x,x_0\rangle + \frac{\sin(\sqrt{K}\,t)}{\sqrt{K}}\,\langle x,v\rangle.$$

Step 4: Differentiate with respect to t

Differentiate term by term. Using

$$\frac{d}{dt}\cos(\sqrt{K}\,t) = -\sqrt{K}\,\sin(\sqrt{K}\,t),$$

and

$$\frac{d}{dt}\sin(\sqrt{K}\,t) = \sqrt{K}\,\cos(\sqrt{K}\,t),$$

we obtain

$$\frac{d}{dt}\langle x,\gamma(t)\rangle = -\sqrt{K}\,\sin(\sqrt{K}\,t)\,\langle x,x_0\rangle + \cos(\sqrt{K}\,t)\,\langle x,v\rangle.$$

At any interior maximizer $t^\star(x)$ where the derivative exists, we must have

$$-\sqrt{K}\,\sin(\sqrt{K}\,t^\star(x))\,\langle x,x_0\rangle + \cos(\sqrt{K}\,t^\star(x))\,\langle x,v\rangle = 0.$$

Equivalently,

$$\cos(\sqrt{K}\,t^\star(x))\,\langle x,v\rangle = \sqrt{K}\,\sin(\sqrt{K}\,t^\star(x))\,\langle x,x_0\rangle.$$

—

Step 5: Solve the optimality condition (case-wise)

Case 1: $\langle x, x_0\rangle \neq 0$

Assume $\langle x, x_0\rangle \neq 0$. If $\cos(\sqrt{K}\, t^\star(x)) \neq 0$, divide both sides by $\sqrt{K}\,\langle x, x_0\rangle\,\cos(\sqrt{K}\, t^\star(x))$ to get

$$\tan(\sqrt{K}\, t^\star(x)) = \frac{\langle x, v\rangle}{\sqrt{K}\,\langle x, x_0\rangle}.$$

A consistent branch choice that also handles the correct quadrant is

$$t^\star(x) = \frac{1}{\sqrt{K}}\,\mathrm{atan2}\big(\langle x, v\rangle,\ \sqrt{K}\,\langle x, x_0\rangle\big).$$

Case 2: $\langle x, x_0\rangle = 0$ and $\langle x, v\rangle \neq 0$

Assume $\langle x, x_0\rangle = 0$ and $\langle x, v\rangle \neq 0$. The optimality condition becomes

$$\cos(\sqrt{K}\, t^\star(x))\,\langle x, v\rangle = 0,$$

hence

$$\cos(\sqrt{K}\, t^\star(x)) = 0.$$

Therefore $\sqrt{K}\, t^\star(x) = \frac{\pi}{2} + m\pi$ for some $m \in \mathbb{Z}$. To choose the maximizer, note that when $\langle x, x_0\rangle = 0$ the objective reduces to

$$\langle x, \gamma(t)\rangle = \frac{\sin(\sqrt{K}\, t)}{\sqrt{K}}\,\langle x, v\rangle.$$

This is maximized by choosing $\sin(\sqrt{K}\, t^\star(x)) = \mathrm{sign}(\langle x, v\rangle)$, hence one convenient representative is

$$t^\star(x) = \frac{1}{\sqrt{K}}\,\arcsin\big(\mathrm{sign}(\langle x, v\rangle)\big).$$

Case 3: $\langle x, x_0\rangle = 0$ and $\langle x, v\rangle = 0$

Assume $\langle x, x_0\rangle = 0$ and $\langle x, v\rangle = 0$. Then for all t,

$$\langle x, \gamma(t)\rangle = 0,$$

so every $t \in \mathbb{R}$ is optimal (the projection is not unique).

This completes the proof.

4. Remarks The generic case holds for almost every $x \in S_K^n$ except a measure-zero set.

B THEORETICAL ANALYSIS PROOFS

B.1 SETTING

Assumption B.1. Each distribution is supported in the closed ball $\bar{B}(o, R)$ around the origin o of its component, with $R < \infty$. For a spherical component with curvature $K > 0$, additionally

$$R < \frac{\pi}{2\sqrt{K}}. \tag{12}$$

We write d_X, d_Y for the geodesic distances of the ambient constant-curvature spaces. For $\pi \in \Pi(\mu, \nu)$, the distortion is defined as:

$$\mathrm{dis}(\pi) := \big\| d_X(x, x') - d_Y(y, y') \big\|_{L_2(\pi \otimes \pi)}, \tag{13}$$

so that $\mathrm{GW}_2(\mathcal{X}, \mathcal{Y}) = \inf_{\pi \in \Pi(\mu,\nu)} \mathrm{dis}(\pi)$, attained by $\pi^\star$ on compact supports (Mémoli, 2011).

For a unit tangent direction v at o_X, let γ_v be the unit-speed geodesic with $\gamma_v(0) = o_X$ and $\dot{\gamma}_v(0) = v$; $P_v^X(x)$ is the point of γ_v closest to x, and $p_v^X(x)$ is defined by $P_v^X(x) = \gamma_v(p_v^X(x))$. P_w^Y, p_w^Y follows similar definition analogously. With $\mu_v := (p_v^X)_{\#}\mu$ and $\nu_w := (p_w^Y)_{\#}\nu$, let $Q(v, w) := \mathrm{GW}_2^2(\mu_v, \nu_w)$, so that $S = \mathbb{E}_{v,w} Q(v, w)$ with $v \sim \lambda_X$, $w \sim \lambda_Y$ independent. Finally, define:

$$d_v^X(x, x') := \big| p_v^X(x) - p_v^X(x') \big|, \qquad h_v^X(x) := d_X\big(x, P_v^X(x)\big), \tag{14}$$

$$\delta_X(v) := \big\| d_X - d_v^X \big\|_{L_2(\mu \otimes \mu)}, \qquad \Delta_X := \big(\mathbb{E}_v\, \delta_X(v)^2\big)^{1/2}, \tag{15}$$

with $H_X = (\mathbb{E}_v \mathbb{E}_{x \sim \mu} h_v^X(x)^2)^{1/2}$ as in Sec. 3.3, and $d_w^Y, h_w^Y, \delta_Y, \Delta_Y$ analogously.

B.2 PROOF OF THM. 3.1

Lemma B.2. *Under Thm. B.1, for every direction v and all $x, x' \in \bar{B}(o, R)$, the projection $P_v(x)$ is unique, $|p_v(x)| \le d_K(x, o) \le R$, and:*

$$\big| p_v(x) - p_v(x') \big| = d_K\big(P_v(x), P_v(x')\big). \tag{16}$$

Proof. *Case $K \le 0$.* Euclidean and hyperbolic spaces are Cartan–Hadamard manifolds, for which the projection onto a geodesic is unique, the signed coordinate is an isometry onto $\mathbb{R}$, which is Eq. (16), and p_v is 1-Lipschitz (Bonet et al., 2025, Sec. 3.2). Since $p_v(o) = 0$, this gives $|p_v(x)| \le d_K(x, o) \le R$.

Case $K > 0$. Let $k = \sqrt{K}$ and $E = \mathrm{span}\{o, v\}$. By the great-circle projection formula of Bonet et al. (2023, Lemma 1), rescaled to radius $1/k$,

$$P_v(x) = \frac{\mathrm{proj}_E\, x}{k \| \mathrm{proj}_E\, x \|}.$$

Since $d_K(o, x) \le R < \pi/(2k)$, we have $\langle o, x \rangle > 0$, so this projection is well defined and unique. Moreover, because $\| \mathrm{proj}_E\, x \| \le \|x\| = 1/k$,

$$\cos\big(k d_K(o, P_v(x))\big) = \frac{k \langle o, x \rangle}{\| \mathrm{proj}_E\, x \|} \ge k^2 \langle o, x \rangle = \cos\big(k d_K(o, x)\big).$$

Thus $|p_v(x)| = d_K(o, P_v(x)) \le d_K(o, x) \le R$. Consequently, any two projected coordinates differ by at most $2R < \pi/k$, and hence their absolute difference equals the shortest great-circle distance between the projected points, proving Eq. (16). □

Lemma B.3. *Under Thm. B.1, $\Delta_X \le 2H_X$ and $\Delta_Y \le 2H_Y$.*

Proof. Fix v, x, x' and write $P = P_v^X(x)$, $P' = P_v^X(x')$. By Thm. B.2, $d_v^X(x, x') = d_X(P, P')$. The triangle inequality along $x \to P \to P' \to x'$ gives

$$d_X(x, x') \le d_X(x, P) + d_X(P, P') + d_X(P', x') = d_v^X(x, x') + h_v^X(x) + h_v^X(x'),$$

and along $P \to x \to x' \to P'$ gives

$$d_v^X(x,x') = d_X(P,P') \le d_X(P,x) + d_X(x,x') + d_X(x',P') = d_X(x,x') + h_v^X(x) + h_v^X(x').$$

Together,

$$\left|d_X(x,x') - d_v^X(x,x')\right| \le h_v^X(x) + h_v^X(x'). \tag{17}$$

Taking the $L_2(\mu\otimes\mu)$ norm and applying Minkowski's inequality, since x and x' each have law μ, $\delta_X(v) \le 2\|h_v^X\|_{L_2(\mu)}$. Squaring and averaging over v gives $\Delta_X^2 \le 4H_X^2$. The same argument applies to $\mathcal{Y}$. □

Lemma B.4. *Under Thm. B.1, for every v, w,*

$$\mathrm{GW}_2(\mathcal{X},\mathcal{Y}) \le \delta_X(v) + \sqrt{Q(v,w)} + \delta_Y(w). \tag{18}$$

Proof. Define the projected metric-measure spaces

$$\mathcal{X}_v := \big(\operatorname{supp}\mu_v, |\cdot - \cdot|, \mu_v\big), \qquad \mathcal{Y}_w := \big(\operatorname{supp}\nu_w, |\cdot - \cdot|, \nu_w\big).$$

Since $\mu_v = (p_v^X)_\#\mu$, the measure $\eta_X = (\mathrm{id}, p_v^X)_\#\mu$ is a coupling of μ and μ_v. Evaluating its distortion yields

$$\mathrm{GW}_2(\mathcal{X},\mathcal{X}_v) \le \left\|d_X - d_v^X\right\|_{L_2(\mu\otimes\mu)} = \delta_X(v).$$

Similarly, the coupling $\eta_Y = (\mathrm{id}, p_w^Y)_\#\nu$ gives

$$\mathrm{GW}_2(\mathcal{Y}_w,\mathcal{Y}) \le \delta_Y(w).$$

By the triangle inequality for GW (Mémoli, 2011, Theorem 5.1(a)),

$$\begin{aligned}\mathrm{GW}_2(\mathcal{X},\mathcal{Y}) &\le \mathrm{GW}_2(\mathcal{X},\mathcal{X}_v) + \mathrm{GW}_2(\mathcal{X}_v,\mathcal{Y}_w) + \mathrm{GW}_2(\mathcal{Y}_w,\mathcal{Y}) \\ &\le \delta_X(v) + \sqrt{Q(v,w)} + \delta_Y(w),\end{aligned}$$

where $\mathrm{GW}_2(\mathcal{X}_v,\mathcal{Y}_w) = \sqrt{Q(v,w)}$ by definition. □

Proof of Thm. 3.1. The left side of Eq. (18) does not depend on (v,w). Averaging the right side over $v\sim\lambda_X$, $w\sim\lambda_Y$ and using $\mathbb{E}Z \le (\mathbb{E}Z^2)^{1/2}$ for nonnegative Z,

$$\begin{aligned}\mathrm{GW}_2(\mathcal{X},\mathcal{Y}) &\le \mathbb{E}_v\delta_X(v) + \mathbb{E}_{v,w}\sqrt{Q(v,w)} + \mathbb{E}_w\delta_Y(w) \\ &\le \Delta_X + \sqrt{S} + \Delta_Y \\ &\le 2H_X + \sqrt{S} + 2H_Y,\end{aligned}$$

where the last step follows from Thm. B.3. □

B.3 Proof of Thm. 3.2

Lemma B.5. *For every $\pi\in\Pi(\mu,\nu)$ and every measurable $g: X\to Y$,*

$$\left\|d_Y\big(g(x),g(x')\big) - d_X(x,x')\right\|_{L_2(\pi\otimes\pi)} \le \mathrm{dis}(\pi) + 2\rho_\pi(g). \tag{19}$$

Proof. Fix $(x,y),(x',y')$. The triangle inequality in Y along $g(x)\to y\to y'\to g(x')$ gives

$$d_Y(g(x),g(x')) - d_X(x,x') \le \big[d_Y(y,y') - d_X(x,x')\big] + d_Y(g(x),y) + d_Y(g(x'),y'),$$

and along $y\to g(x)\to g(x')\to y'$ gives

$$d_X(x,x') - d_Y(g(x),g(x')) \le \big[d_X(x,x') - d_Y(y,y')\big] + d_Y(g(x),y) + d_Y(g(x'),y').$$

Together,

$$\left|d_Y(g(x),g(x')) - d_X(x,x')\right| \le \left|d_Y(y,y') - d_X(x,x')\right| + d_Y(g(x),y) + d_Y(g(x'),y'). \tag{20}$$

Taking the $L_2(\pi\otimes\pi)$ norm and applying Minkowski's inequality, the first term has norm $\mathrm{dis}(\pi)$, and each of the other two has norm $\rho_\pi(g)$ since (x,y) and (x',y') each have law π. □

Proof of Thm. 3.2. For $\pi^\star \otimes \pi^\star$-almost every pair with $x \neq x'$, we have $x, x' \in S_X$ and hence $d_X(x, x') \geq r$, so

$$\left| \frac{d_Y(g(x), g(x'))}{d_X(x, x')} - 1 \right| = \frac{\left| d_Y(g(x), g(x')) - d_X(x, x') \right|}{d_X(x, x')} \leq \frac{\left| d_Y(g(x), g(x')) - d_X(x, x') \right|}{r}.$$

Squaring and integrating over $\{x \neq x'\}$, the left side of the first claim is at most

$$r^{-1} \left\| d_Y(g(x), g(x')) - d_X(x, x') \right\|_{L_2(\pi^\star \otimes \pi^\star)}.$$

By Thm. B.5 with $\pi = \pi^\star$, $\mathrm{dis}(\pi^\star) = \mathrm{GW}_2(\mathcal{X}, \mathcal{Y})$, and Thm. 3.1, this norm is at most B, proving the first claim. For the second, let Z be the dilation deviation on $\{x \neq x'\}$ and $t := B/(r\sqrt{\epsilon})$; by Chebyshev's inequality,

$$(\pi^\star \otimes \pi^\star)\big(\{x \neq x',\, |Z| > t\}\big) \leq (B/r)^2/t^2 = \epsilon.$$

□

C EXPERIMENT DETAILS

C.1 GRAPH ANOMALY DETECTION

We evaluate on five GADBench datasets (Tang et al., 2023): Reddit (Kumar et al., 2019; Liu et al., 2022), Weibo (Zhao et al., 2020), Amazon (McAuley & Leskovec, 2013), Elliptic (Weber et al., 2019) and Questions (Platonov et al., 2023). Following the fully-supervised protocol, we report the mean and 95% confidence interval of test AUROC at the best validation epoch over ten splits. CurvGAD (Grover et al., 2025a) retains its original hyperparameters and manifold signatures, and all of its results are reproduced under this protocol. At every epoch, CCSGW draws 256 random nodes and, for each constant-curvature component, projects their embeddings along random geodesic directions (number of directions $L = 32$) under the current curvature, which is kept fixed within this term. The resulting slices are standardized, and their closed-form one-dimensional Gromov–Wasserstein discrepancy is averaged over directions and component pairs. This averaged discrepancy is added to the CurvGAD objective with the alignment weight $\lambda = 10$, shared across all datasets.

The baselines we consider for graph anomaly detection tasks are: GCN (Kipf & Welling, 2016), GAT (Veličković et al., 2017), SAGE (Hamilton et al., 2017), BernNet (He et al., 2021), ChebyNet (Defferrard et al., 2016), HGCN (Chami et al., 2019), HGAT (Zhang et al., 2021), KGCN (Bachmann et al., 2020), QGCN (Xiong et al., 2022), DCI (Wang et al., 2021), PCGNN (Liu et al., 2021), BWGNN (Tang et al., 2022), DOMINANT (Ding et al., 2019), AnomalyDAE (Fan et al., 2020), GADNR (Roy et al., 2024), ADAGAD (He et al., 2024).

C.2 GRAPH NODE CLASSIFICATION

We evaluate on three homophilic datasets, Cora, CiteSeer and PubMed (Sen et al., 2008), and two heterophilic datasets, Chameleon (Pei et al., 2020) and Actor (Tang et al., 2009). Following CUSP (Grover et al., 2025b), we report the mean and 95% confidence interval of test weighted F1 at the best validation epoch over ten random 60/20/20 splits. CUSP strictly follows the configuration of the original paper: each dataset uses the reported manifold signature, i.e., the type and dimension of every constant-curvature component, namely $\mathbb{H}^{16} \times \mathbb{S}^{16} \times \mathbb{E}^{16}$ for Cora and CiteSeer, $\mathbb{H}^{16} \times \mathbb{H}^{16} \times \mathbb{E}^{16}$ for PubMed and Actor, and $\mathbb{H}^{16} \times \mathbb{S}^{16} \times \mathbb{S}^{16}$ for Chameleon, with learnable curvatures initialized as in the original paper. All remaining hyperparameters, including the number of training epochs (300), are taken from the original search space, and all CUSP results are reproduced under this protocol with early stopping on validation F1. At every epoch, CCSGW draws 256 random nodes for homophilic and 512 for heterophilic datasets and, for each constant-curvature component, projects their embeddings along random geodesic directions (number of directions $L = 32$) under the current curvature, which also receives gradients from this term. The resulting slices are standardized, and their closed-form one-dimensional Gromov–Wasserstein discrepancy is averaged over directions and component pairs. This averaged discrepancy is added to the CUSP objective with the alignment weight $\lambda = 1$, shared across all datasets and linearly warmed up over the first 50 epochs.

The baselines we consider for graph node classification tasks are: GCN (Kipf & Welling, 2016), GAT (Veličković et al., 2017), SAGE (Hamilton et al., 2017), HGCN (Chami et al., 2019), HGAT (Zhang et al., 2021), KGCN (Bachmann et al., 2020), QGCN (Xiong et al., 2022), SelfMGNN (Sun et al., 2022), ChebyNet (Defferrard et al., 2016), BernNet (He et al., 2021), GPRGNN (Chien et al., 2020), FiGURe (Ekbote et al., 2023), H2GCN (Zhu et al., 2020), FAGCN (Bo et al., 2021), CurvDrop (Liu et al., 2023), ARGNN (Wang et al., 2026).

C.3 EEG-BASED MULTIMODAL LEARNING

EEG-based multimodal learning combines brain signals with complementary modalities to support tasks such as emotion recognition, sleep staging, and cognitive assessment. We further integrate CCSGW into EEG-MoCE (Zhou et al., 2026), a state-of-the-art multimodal learning framework based on hyperbolic mixture-of-curvature experts, and evaluate the resulting model on the emotion recognition tasks and two cognitive tasks. Following (Zhou et al., 2026), we perform subject-wise 10-fold cross-validation on EAV ($n = 42$ participants) and Cognitive ($n = 26$ participants), and report test balanced accuracy (BAcc) averaged over folds. Both EEG-MoCE and its CCSGW-augmented

variant follow the original preprocessing pipeline and model configuration, including the modality-specific encoders, hyperbolic experts with learnable curvatures, and curvature-aware fusion module. All optimization settings follow the original paper. For CCSGW alignment, we select one modality-specific hyperbolic manifold as the reference and align the embedding distribution of each remaining modality with that of the reference modality. The modality-specific embeddings are projected along random geodesic directions under their respective curvatures, and the resulting one-dimensional slices are standardized. The closed-form one-dimensional Gromov–Wasserstein discrepancies are then averaged over projection directions and pairs formed by the reference modality and each remaining modality. This averaged discrepancy is added to the original EEG-MoCE objective as a weighted alignment regularizer.

The baselines we consider for emotion recognition tasks are: AGF-IB Shou et al. (2024), CMERC Tu et al. (2024), CTMWA Zhang et al. (2024), GA2MIF Li et al. (2024), HEEGNet Li et al. (2026b), Hyper-MML Kang et al. (2025),LMF Liu et al. (2018), MMML Wu et al. (2024); for the cognitive tasks are: CTMWA Zhang et al. (2024), EFDFNet Xu et al. (2025), EF-Net Arif et al. (2024), LMF Liu et al. (2018), MMML Wu et al. (2024), STA-Net Liu et al. (2025), ST2A Shi et al. (2025), TSMMF Si et al. (2025) .

D Additional Experiments

D.1 Hyperparameter sensitivity

We study the sensitivity of CCSGW to its hyperparameters on Cora with CUSP and on Reddit with CurvGAD. In both backbones, node embeddings lie in a mixed-curvature space with m constant-curvature components, and CCSGW aligns every pair of components. Let μ_j denote the distribution of node embeddings in the j-th component, whose curvature is K_j. The training objective is

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda \binom{m}{2}^{-1} \sum_{1 \le j < k \le m} \frac{1}{L} \sum_{\ell=1}^{L} \mathrm{GW}_2^2\Big(d_{\mathbb{R}}^2,\ (p_{v_j^\ell}^{K_j})_{\#}\hat{\mu}_j,\ (p_{v_k^\ell}^{K_k})_{\#}\hat{\mu}_k\Big)\,, \tag{21}$$

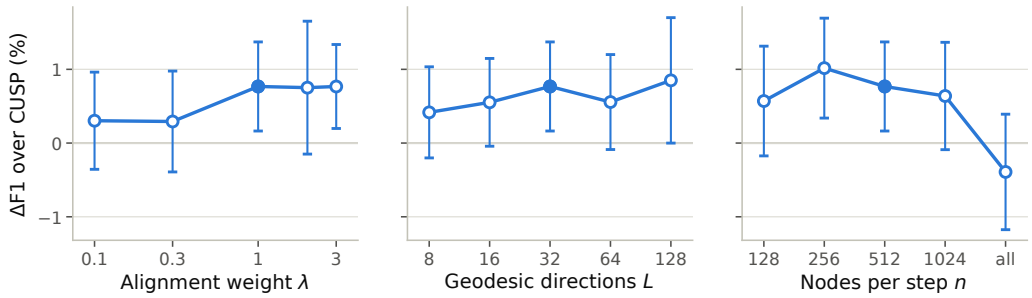
where $\hat{\mu}_j$ is the empirical measure of μ_j on n nodes sampled uniformly at random (the same nodes for all components), and $v_j^1, \dots, v_j^L$ are i.i.d. uniform unit tangent directions at the origin of the j-th component. Nodes and directions are redrawn at every epoch, and each projected slice is standardized as described in Sec. C.1. As in SGW (Eq. (5)), the integral over directions is replaced by an average over L random directions. CCSGW therefore has three hyperparameters: the alignment weight λ, the number of geodesic directions L, which controls the accuracy of this approximation, and the number of sampled nodes n, which sets the support size of the empirical measures. Each component pair costs $O(L\, n \log n)$.

We vary one hyperparameter at a time around a reference configuration (λ, L, n), set to $(1, 32, 512)$ on Cora and $(10, 32, 256)$ on Reddit, over the values on the horizontal axes of Fig. 3. All other settings, including the ten splits, follow Sec. C.2 for Cora and Sec. C.1 for Reddit. For each configuration, we report the gain Δ of the backbone trained with CCSGW over the same backbone trained without it ($\lambda = 0$) on the same split and seed, measured in test weighted F1 on Cora and test AUROC on Reddit and averaged over the ten splits.

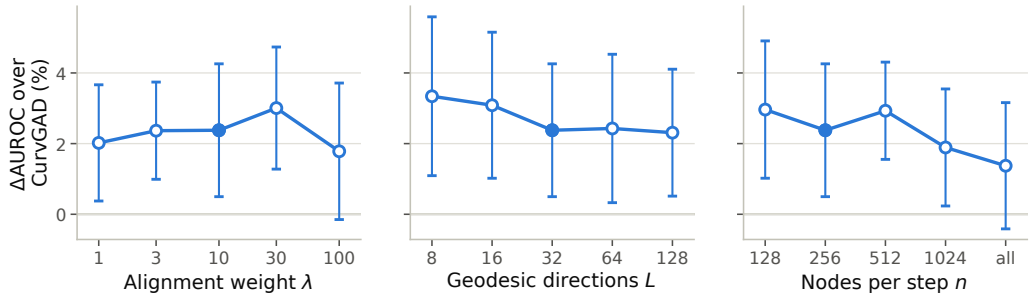
CCSGW is robust to all three hyperparameters. On both datasets, every configuration except one stays within the confidence interval of the reference configuration, and the gain remains positive across two orders of magnitude of λ and a sixteenfold range of L. The insensitivity to L follows from redrawing the directions at every epoch, which averages the approximation error of a small L over training. Subsamples of 128 to 1024 nodes perform comparably, whereas using all nodes gives the smallest gain on both datasets and the only negative one, on Cora. Subsampling therefore reduces the cost of CCSGW without sacrificing accuracy.

D.2 Comparison with SGW and GW

CCSGW differs from SGW in slicing along geodesics rather than Euclidean directions, and from GW in slicing at all. To isolate the effect of each choice, we replace CCSGW in the CUSP objective by either loss and keep everything else fixed. SGW shares the implementation of CCSGW, including



(a) Node classification: CUSP + CCSGW on Cora.



(b) Graph anomaly detection: CurvGAD + CCSGW on Reddit.

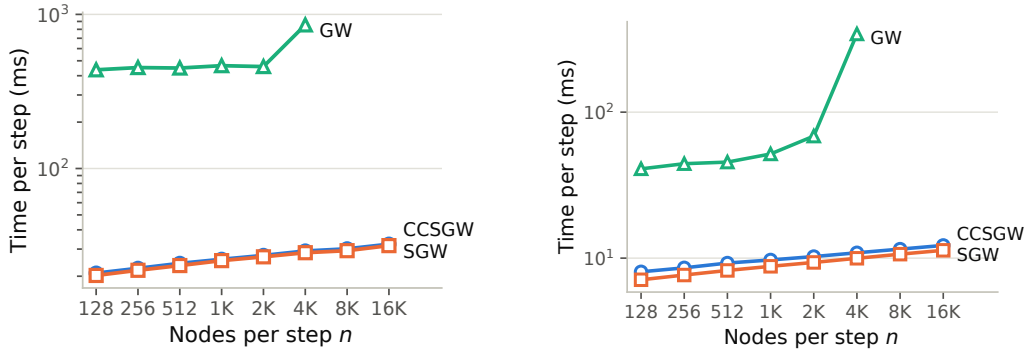
Figure 3: Hyperparameter sensitivity of CCSGW. Each panel varies one hyperparameter, namely the alignment weight λ, the number of geodesic directions L or the number of sampled nodes n ("all" uses every node), while the other two are fixed at the reference configuration (filled markers). The vertical axis shows the gain Δ of the backbone trained with CCSGW over the same backbone trained without it (gray line at zero). Points and error bars are the mean and 95% confidence interval over ten splits.

the independent directions of every component, with the geodesic projections replaced by linear projections in the ambient coordinates, which corresponds to setting every curvature to zero. GW is the entropic Gromov–Wasserstein discrepancy between the geodesic distance matrices of the n sampled nodes, normalized by their mean distance and solved with ten projected-gradient steps of fifty Sinkhorn iterations each ($\varepsilon = 0.05$). All three losses use the weight $\lambda = 1$, $L = 32$ directions where applicable, the node subsample of the main experiments ($n = 256$ for the homophilic and 512 for the heterophilic datasets), and the CUSP configuration and ten splits of Sec. C.2.

Table 4: Comparison of alignment losses on top of CUSP (test weighted F1 %(↑) ± 95% confidence interval over ten splits). The last column is the mean gain over CUSP across the five datasets. Best result per column in bold.

Method	Cora	CiteSeer	PubMed	Chameleon	Actor	**Av.** Δ
CUSP	85.64±1.48	74.10±0.88	90.34±0.21	66.48±2.21	38.90±0.93	–
CUSP + CCSGW	**86.66±0.93**	**74.24±1.23**	**90.39±0.18**	67.82±1.92	39.83±0.70	+0.70
CUSP + SGW	86.19±1.00	73.62±1.22	90.32±0.14	67.49±1.23	39.23±0.84	+0.28
CUSP + GW	86.18±1.16	73.37±1.03	90.31±0.25	**68.48±1.10**	**39.98±0.91**	+0.57

Tab. 4 shows that CCSGW improves over SGW on all five datasets, with a mean gain over CUSP of $+0.70$ against $+0.28$. Since the two losses differ only in the direction of the slices, most of the benefit comes from slicing along geodesics, which costs the same as slicing along Euclidean directions (Sec. D.3). CCSGW is also on par with GW: it is better on the three homophilic datasets and slightly worse on Chameleon and Actor. Replacing the full Gromov–Wasserstein discrepancy



(a) CUSP: three components, three aligned pairs.

(b) CurvGAD: two components, one aligned pair.

Figure 4: Time of one alignment-loss evaluation (forward and backward) as a function of the number of sampled nodes n, on a single A100 GPU (median of seven runs, log–log axes). GW runs out of memory beyond $n = 4096$.

by its sliced version therefore loses no accuracy, while it reduces the cost from quadratic to log-linear in the number of sampled nodes (Sec. D.3). CCSGW thus combines the accuracy of GW with the cost of SGW.

D.3 Computational cost

CCSGW and SGW cost $O(L\, n \log n)$ per component pair, whereas entropic GW on geodesic distances requires $O(n^2)$ memory and time per Sinkhorn iteration. We measure the wall-clock time of one evaluation of each alignment loss, including the backward pass, as a function of the number of sampled nodes n, on a single A100 GPU (median of seven repetitions after one warm-up call). Embeddings are drawn at random in the mixed-curvature space of each backbone, and every loss aligns the same n nodes with $L = 32$ directions and standardized slices. Since CCSGW aligns every pair of components, the number of aligned pairs is three for the CUSP signature $\mathbb{H}^{16} \times \mathbb{S}^{16} \times \mathbb{E}^{16}$ and one for the CurvGAD signature $\mathbb{H}^{24} \times \mathbb{S}^{24}$.

Fig. 4 shows that the geodesic projections come essentially for free: CCSGW and SGW coincide at every n, and both remain in the millisecond range over the whole tested range. GW is one to two orders of magnitude slower from a few thousand nodes on, where its n^2 cost matrices dominate, and exhausts the 40 GB of memory at $n = 8192$. The higher cost on CUSP reflects the larger number of aligned pairs. For the subsample sizes used in our experiments ($n \leq 512$), CCSGW adds only a small fraction of the cost of one training epoch of either backbone.