Neural Projection Fusion for Sliced Wasserstein on the Hypersphere

Hongliang Zhang[1], **Jianjun Qian**[1], **Lei Luo**[1,*] and **Jian Yang**[1,2,*]
[1]PCA Lab, School of Computer Science and Engineering, Nanjing University of Science and Technology
[2]PCA Lab, School of Intelligence Science and Technology, Nanjing University
zhang1hongliang@njust.edu.cn, csjqian@njust.edu.cn, luoleipitt@gmail.com, csjyang@njust.edu.cn

Abstract

The Spherical Sliced Wasserstein (SSW) distance has emerged as a fundamental tool for comparing probability measures on the hypersphere, with broad applications spanning geology, medical imaging, computer vision, and representation learning. However, its reliance on uniformly sampling a large number of projection directions from the unit hypersphere can lead to suboptimal performance, as many directions are weakly informative and fail to capture salient differences between distributions. We address this limitation through Fusion Stereographic Spherical Sliced Wasserstein (FS3W), a data-adaptive divergence that incorporates distribution-dependent information into projection selection. FS3W uses a neural slicing fusion mechanism to refine prior directions, steering them toward projections aligned with discriminative geometric structures in the data. This adaptive approach produces more expressive and accurate characterizations of distributional discrepancies on the hypersphere. We evaluate FS3W across five diverse tasks, including gradient flows, Earth density estimation, and self-supervised representation learning. Empirical results consistently demonstrate that FS3W outperforms existing spherical optimal transport–based methods.

1 Introduction

In many real-world applications, data often resides on hypersphere, highlighting the critical role of spherical geometry across a wide range of tasks. Typical examples include geophysical and meteorological modeling [Di Marzio *et al.*, 2014; Besombes *et al.*, 2021], magnetic brain imaging in medical analysis [Vrba and Robinson, 2001; Hu *et al.*, 2023; Hu *et al.*, 2025], and texture mapping in computer graphics [Dominitz and Tannenbaum, 2009]. In such settings, latent representations are usually constrained to bounded domains that are well-approximated by unit spheres [Wang and Isola, 2020; Chen *et al.*, 2020]. These settings call for learning methods designed specifically for spherical manifolds.

[1]*Corresponding Author.

Traditional analysis on the hypersphere falls within spherical statistics [Jammalamadaka and Sengupta, 2001], which studies directional data such as orientations and rotations. Recently, Optimal Transport (OT) theory has gained attention for comparing distributions on the hypersphere [Cui *et al.*, 2019], due to its strong statistical and geometric properties [Peyré *et al.*, 2019]. However, OT often suffer from high computational complexity and the curse of dimensionality [Peyré and Cuturi, 2019], which has motivated the development of scalable solvers [Cuturi, 2013] and computationally efficient alternative distance metrics [Kolouri *et al.*, 2019a]. Among them, the Sliced-Wasserstein (SW) [Nguyen *et al.*, 2021; Ohana *et al.*, 2023] distance preserves the topological properties of the Wasserstein distance [Nadjahi *et al.*, 2020], while being more efficient and immune to the curse of dimensionality [Nietert *et al.*, 2022].

The SW distance has recently been extended to spherical domains for distribution comparison, which has led to several spherical sliced OT approaches [Bonet *et al.*, 2023; Quellmalz *et al.*, 2023; Tran *et al.*, 2024]. A key challenge is to generalize the classical Radon transform to the sphere. [Quellmalz *et al.*, 2023] introduced two such extensions, the vertical slice transform [Rubin, 2018] and the normalized semicircle transform [Groemer, 1998]. Bonet et al. [Bonet *et al.*, 2023] proposed a new spherical Radon transform and used the closed-form 1D Wasserstein distance on the circle to define Spherical Sliced-Wasserstein (SSW). Although both works [Bonet *et al.*, 2023; Quellmalz *et al.*, 2023] project spherical distributions onto circular marginals, computing OT on the circle remains relatively expensive. To reduce the computation cost, Tran et al. [Tran *et al.*, 2024] mapped the hypersphere to a hyperplane via stereographic projection and then applied the classical Radon transform to define Stereographic Sliced-Wasserstein (S3W). Tran et al. [Tran *et al.*, 2025] further proposed the Spherical Tree-Sliced Wasserstein (STSW) distance, which improves scalability by combining the spherical Radon transform with tree-based metrics.

Most existing spherical sliced Wasserstein methods sample projection directions uniformly, implicitly treating all directions as equally informative. However, in high-dimensional spherical spaces, discriminative information is often concentrated in only a few directions, making many random projections weakly informative. Recent adaptive methods such as EBSW [Nguyen and Ho, 2024] and DSSW [Zhang

et al., 2025] reweight projections according to their one-dimensional discrepancies, but still rely on fixed sampled directions and cannot correct poorly aligned projections. Max-SW [Deshpande *et al.*, 2019] selects the most discriminative projection, but sacrifices the averaging structure of sliced distances and may lead to unstable optimization.

To address these limitations, we propose Fusion Stereographic Sliced Wasserstein (FS3W). Rather than reweighting or selecting fixed projections, FS3W learns to refine projection directions themselves. Starting from uniformly sampled-directions, a neural slicing fusion mechanism generates data-dependent directions jointly from the two measures, better capturing their geometric structure while preserving the integral form of sliced Wasserstein distances. Thus, FS3W retains the stability and interpretability of averaging-based formulations while improving expressiveness through principled data adaptivity. Our contributions are as follows:

- We propose a neural slicing approach that adaptively refines projection directions for more informative and discriminative hyperspherical comparisons.
- We provide comprehensive theoretical analysis to guarantee the topological and statistical properties of FS3W.
- We validate the effectiveness of FS3W on several classical machine learning tasks, showing that it consistently achieves superior accuracy compared to SSW, S3W, STSW, SW, and the Wasserstein distance.

2 Background

This section reviews the Wasserstein distance, sliced Wasserstein (SW), and stereographic spherical sliced Wasserstein (S3W), which form the foundation of our proposed FS3W on the hypersphere $\mathbb{S}^d = \{s \in \mathbb{R}^{d+1} \mid \|s\|_2 = 1\}$.

Wasserstein Distance. Let M be a Riemannian manifold with distance $c(\cdot,\cdot)$. For $p \geq 1$, denote by $\mathcal{P}_p(M)$ the set of probability measures with finite p-th moment. The p-Wasserstein distance between $\mu, \nu \in \mathcal{P}_p(M)$ is defined as [Peyré and Cuturi, 2019]

$$W_p^p(\mu,\nu) = \inf_{\gamma \in \Pi(\mu,\nu)} \int_{M\times M} \big(c(x,y)\big)^p \,\mathrm{d}\gamma(x,y), \qquad (1)$$

where $\Pi(\mu,\nu)$ denotes the set of couplings between μ and ν. While Wasserstein distance has strong theoretical properties, its computation is costly for empirical measures, which motivates more efficient alternatives.

Sliced Wasserstein Distance. For one dimensional measures $\mu, \nu \in \mathcal{P}_p(\mathbb{R}^d)$, the Wasserstein distance between μ and ν admits the closed form as:

$$W_p^p(\mu,\nu) = \int_0^1 \left|F_\mu^{-1}(t) - F_\nu^{-1}(t)\right|^p \mathrm{d}t, \qquad (2)$$

where F_μ^{-1} and F_ν^{-1} are quantile functions. This leads to the sliced Wasserstein (SW) distance [Bonneel *et al.*, 2015],

$$SW_p^p(\mu,\nu) = \int_{\mathbb{S}^{d-1}} \mathrm{W}_p^p(\mathcal{R}f_\mu(\cdot,\theta), \mathcal{R}f_\nu(\cdot,\theta))\,\mathrm{d}\sigma_d(\theta), \quad (3)$$

where σ_d is the uniform distribution on the unit sphere $\mathbb{S}^{d-1}$. The Radon Transform [Helgason and others, 2011] $\mathcal{R} : L^1\left(\mathbb{R}^d\right) \to L^1\left(\mathbb{R}\times\mathbb{S}^{d-1}\right)$ is defined as: $\mathcal{R}f(t,\theta) = \int_{\mathbb{R}^d} f(x)\cdot\delta\left(t-\langle x,\theta\rangle\right)dx$, where L^1 is the set of absolutely integrable functions on $\mathbb{R}^d$ and δ denotes the Dirac delta function. Since the expectation in Eq. (3) is intractable, it is approximated via Monte Carlo estimation with computational complexity $\mathcal{O}\left(Ln\left(d+\log n\right)\right)$, where n and d denote the sample size and data dimensionality, respectively:

$$\widehat{SW}_p^p(\mu,\nu) = \frac{1}{L}\sum_{\ell=1}^{L} \mathrm{W}_p^p\left(\mathcal{R}f_\mu(\cdot,\theta_\ell), \mathcal{R}f_\nu(\cdot,\theta_\ell)\right), \quad (4)$$

where the directions $\{\theta_\ell\}_{\ell=1}^L$ are sampled independently and uniformly from $\mathcal{U}(\mathbb{S}^{d-1})$, and L is the number of projections.

Stereographic Projection. Stereographic projection is an explicit map that transforms points on a sphere onto a plane. The stereographic projection $\phi : \mathbb{S}^d \setminus \{s_n\} \to \mathbb{R}^d$ maps a point $s \in \mathbb{S}^d$ (excluding the "North Pole," $s_n = [0,\ldots,0,1]$) to a point $x \in \mathbb{R}^d$ as [Tran *et al.*, 2024]:

$$[x]_i = \frac{2[s]_i}{1-[s]_{d+1}}, \quad \text{for } i = 1,2,\ldots,d. \qquad (5)$$

This projection is smooth and bijective between $\mathbb{S}^d\setminus\{s_n\}$ and $\mathbb{R}^d$, corresponding to the hyperplane tangent to the sphere at the south pole ($s_{d+1} = -1$) and commonly used to visualize spherical geometry in Euclidean space.

Stereographic Spherical Radon Transform. Let $\mu \in \mathcal{M}(\mathbb{S}^d)$ be a Radon measure defined on $\mathbb{S}^d$ that does not assign mass to the North Pole, *i.e.*, $\mu(\{s_n\}) = 0$. Consider the stereographic projection $\phi : \mathbb{S}^d \setminus \{s_n\} \to \mathbb{R}^d$, which is bijection, so that $\phi_{\#}\mu$ defines a Radon measure on $\mathbb{R}^d$. Based on this mapping, the stereographic spherical Radon transform [Tran *et al.*, 2024] is defined as:

$$\mathcal{S}_\mathcal{R}(\mu) = \mathcal{R}(\phi_{\#}\mu) \in \mathcal{M}(\mathbb{R}\times\mathbb{S}^{d-1}), \qquad (6)$$

where $\mathcal{R}(\phi_{\#}\mu)_\theta := (\pi_\theta)_{\#}(\phi_{\#}\mu)$ with $\pi_\theta(x) = \langle x,\theta\rangle$.

Stereographic Spherical Sliced Wasserstein. Let $\mu,\nu \in \mathcal{P}_p(\mathbb{S}^d)$ denote two probability measures on the unit sphere in $\mathbb{R}^{d+1}$. Based on the stereographic spherical Radon transform, the stereographic spherical sliced Wasserstein (S3W) distance [Tran *et al.*, 2024] is defined as:

$$S3W_{\mathcal{R},p}^p(\mu,\nu) = \int_{\mathbb{S}^{d-1}} W_p^p(\mathcal{S}_\mathcal{R}(\mu)_\theta, \mathcal{S}_\mathcal{R}(\nu)_\theta)\,\mathrm{d}\sigma_d(\theta). \quad (7)$$

To maintain the maintain rotational symmetry of spherical OT, Tran et al. [Tran *et al.*, 2024] introduce two rotationally invariant extensions of S3W, termed RI-S3W and ARI-S3W. RI-S3W attains invariance by integrating over the sphere's rotation group, while ARI-S3W offers a more efficient approximation via sampling a finite set of rotations.

3 Method

The Spherical Sliced Wasserstein distance compares hyperspherical distributions by averaging one-dimensional Wasserstein distances over uniformly sampled projection directions. However, uniformly sampled directions may differ in their discriminative power, especially in high-dimensional spaces.

To address this, we propose a neural slicing fusion mechanism that injects data-driven information into projection selection. Rather than using fixed directions, our method adapts projections to the input distributions and aligns them with discriminative geometric structures. This yields FS3W, which preserves the integral form of Sliced Wasserstein distances while offering a more expressive and adaptive metric for hyperspherical probability measures.

Definition 1 (**Neural Slicing Fusion Mechanism**). *Let $\mu, \nu \in \mathcal{P}_p(\mathbb{S}^d)$ denote two probability measures on the unit sphere in $\mathbb{R}^{d+1}$, given a neural network $h_\psi : \mathbb{R}^{(2n+1)*d} \to \mathbb{R}^{1*d}$, a neural slicing fusion mechanism $a_\psi : \mathbb{R}^{n*d} \times \mathbb{R}^{n*d} \times \mathbb{R}^{1*d} \to \mathbb{R}^{1*d}$ is defined as:*

$$a_\psi(\mu, \nu, \theta_\ell) := \frac{1}{2}\left(h_\psi([\mu, \nu, \theta_\ell]) + h_\psi([\nu, \mu, \theta_\ell])\right). \quad (8)$$

The neural network h_ψ can be implemented with a standard nonlinear architecture or a linear self-attention mechanism with learnable parameters ψ. This mechanism generates data-dependent projection directions, which are used to compare the distributions. Further details on the architecture of h_ψ are provided in Supplementary Material Section B.1.

Definition 2 (**FS3W**). *For $p \geq 1$, let $\mu, \nu \in \mathcal{P}_p(\mathbb{S}^d)$ denote two probability measures on the unit sphere in $\mathbb{R}^{d+1}$, and let $a_\psi(\mu, \nu, \theta)$ be the neural slicing fusion mechanism. We define the FS3W distance as:*

$$FS3W^p_{\mathcal{R},p}(\mu, \nu; a_\psi) := \int_{\mathbb{S}^{d-1}} W_p^p\left(\mathcal{S}_\mathcal{R}(\mu)_{a_\psi(\mu,\nu,\theta)}, \mathcal{S}_\mathcal{R}(\nu)_{a_\psi(\mu,\nu,\theta)}\right) \mathrm{d}\sigma_d(\theta), \quad (9)$$

where σ_d is the uniform distribution on $\mathbb{S}^{d-1}$, W_p^p is the p-Wasserstein distance, and $\mathcal{S}_\mathcal{R}(\cdot)_\theta$ denotes the stereographic spherical Radon transform. For simplicity, we use $FS3W_2$ to denote $FS3W_{\mathcal{R},2}$ in the subsequent experiments.

Monte Carlo Estimation. Similar to the S3W distance, we adopt Monte-Carlo estimation to approximate FS3W. We randomly sample projection directions $\theta_1, \ldots, \theta_L \overset{i.i.d}{\sim} \sigma_d \in \mathcal{U}(\mathbb{S}^{d-1})$. The estimator of FS3W is then given by:

$$\widehat{FS3W}^p_{\mathcal{R},p}(\mu, \nu; a_\psi) = \frac{1}{L}\sum_{\ell=1}^{L} W_p^p\left(\mathcal{S}_\mathcal{R}(\mu)_{a_\psi(\mu,\nu,\theta_\ell)}, \mathcal{S}_\mathcal{R}(\nu)_{a_\psi(\mu,\nu,\theta_\ell)}\right). \quad (10)$$

To ensure projections capture the most discriminative geometric structures, we maximize the discrepancy between projected measures. Specifically, the training objective (Max-FS3W) for mechanism a_ψ is:

$$\mathcal{L}(\psi) = \max_\psi \left\{\sum_{\ell=1}^{L} W_p^p\left(\mathcal{S}_\mathcal{R}(\mu)_{a_\psi(\mu,\nu,\theta_\ell)}, \mathcal{S}_\mathcal{R}(\nu)_{a_\psi(\mu,\nu,\theta_\ell)}\right)\right\}^{\frac{1}{p}}, \quad (11)$$

where L is the number of projection directions.

We employ AdamW with a learning rate of 1.0e-3 and 50 iterations for all tasks. The implementation pseudocode of our FS3W variant, RI-FS3W variant, and ARI-FS3W variant is presented in Algorithms 1, 2, and 3 (Supplementary Material Section B.2), respectively. During training, h_ψ learns to integrate data-specific distribution information into prior projection directions, improving the ability of FS3W to measure distributional discrepancies more accurately.

To implement the fusion mechanism a_ψ, we employ a nonlinear neural network and linear self-attention [Nguyen *et al.*, 2023], yielding two FS3W variants: $\mathcal{N}$FS3W and $\mathcal{L}$AFS3W. Extending RI-S3W and ARI-S3W with these architectures produces four rotationally invariant variants: RI-$\mathcal{N}$FS3W, RI-$\mathcal{L}$AFS3W, ARI-$\mathcal{N}$FS3W, and ARI-$\mathcal{L}$AFS3W.

Proposition 1 (**Topological properties**). *Let $p \geq 1$ and let $a_\psi(\mu, \nu, \theta)$ be a neural slicing fusion mechanism satisfying $a_\psi(\mu, \nu, \theta) = a_\psi(\nu, \mu, \theta)$. Then the functional $FS3W_{\mathcal{R},p}$ is non-negative and symmetric.*

Proposition 1 shows that FS3W preserves the basic topological properties: non-negativity and symmetry. These properties come directly from the corresponding properties of the underlying S3W distance and the symmetry of the fusion mechanism. All proofs for the propositions and theories in this paper are detailed in Supplementary Material Section A.

Proposition 2 (**Convergence property**). *For any $p \geq 1$ and the neural slicing fusion mechanism $a_\psi(\mu, \nu, \theta)$, assume that the stereographic projection is well-defined on the support of μ and that a_ψ is continuous with respect to (μ, ν, θ). Let $\mu_k, \mu \in \mathcal{P}_p(\mathbb{S}^d)$. If $\mu_k \Rightarrow \mu$ and $\sup_k \int_{\mathbb{S}^d} \|s\|^p \, d\mu_k(s) < \infty$, then $\lim_{k\to\infty} FS3W^p_{\mathcal{R},p}(\mu_k, \mu; a_\psi) = 0$.*

Proposition 2 establishes the convergence of FS3W under weak convergence of probability measures with bounded moments. This result ensures that FS3W is consistent with standard notions of convergence used in optimal transport and defines a valid discrepancy between probability measures.

Proposition 3 (**Sample complexity**). *For any $p \geq 1$ and the neural slicing fusion mechanism $a_\psi(\mu, \nu, \theta)$, assumed to be Lipschitz continuous in θ, let $x_1, \ldots, x_n \overset{i.i.d.}{\sim} \mu$ and $y_1, \ldots, y_n \overset{i.i.d.}{\sim} \nu$ be observed samples, where $\mu, \nu \in \mathcal{P}(\mathbb{S}^d)$ are supported on a compact set. Let $\hat{\mu}_n$ and $\hat{\nu}_n$ be the corresponding empirical measures.Let R be the diameter of the joint support. Then there exists a constant $C_{p,R} > 0$ depending only on (p, R) such that*

$$\mathbb{E}\left[\left|FS3W^p_{\mathcal{R},p}(\hat{\mu}_n, \hat{\nu}_n; a_\psi) - FS3W^p_{\mathcal{R},p}(\mu, \nu; a_\psi)\right|\right] \leq 2C_{p,R}\left(\sqrt{\frac{(d+1)\log(n+1)}{n}}\right). \quad (12)$$

Proposition 3 characterizes the statistical behavior of FS3W estimated from finite samples. The bound reveals that the estimation error decreases with sample size and depends on the problem dimension, consistent with classical sliced distances. This indicates that the neural fusion mechanism does not increase the sample complexity.

Proposition 4 (**Projection Complexity**). *Let $\mu, \nu \in \mathcal{P}_p(\mathbb{S}^d)$, and let $a_\psi(\mu, \nu, \theta)$ be the neural slicing fusion mechanism, assumed to be L_ψ-Lipschitz with respect to $\theta \in \mathbb{S}^{d-1}$. Assume that the stereographic Wasserstein functional $W_p^p(\cdot, \cdot)$*

is L_W-Lipschitz with respect to its projection parameter. Let $\theta_1, \ldots, \theta_L \overset{i.i.d.}{\sim} \sigma_{d-1}$ be sampled uniformly from the unit sphere $\mathbb{S}^{d-1}$. Define the Monte Carlo estimator

$$\widehat{FS3W}^p_{\mathcal{R},p,L}(\mu,\nu;a_\psi) := \frac{1}{L}\sum_{\ell=1}^{L} W_p^p\left(\mathcal{S}_{\mathcal{R}}(\mu)_{a_\psi(\mu,\nu,\theta_\ell)}, \mathcal{S}_{\mathcal{R}}(\nu)_{a_\psi(\mu,\nu,\theta_\ell)}\right). \tag{13}$$

Then there exists a constant $C_d > 0$, depending only on the geometry of $\mathbb{S}^{d-1}$, such that

$$\mathbb{E}\left[\left|\widehat{FS3W}^p_{\mathcal{R},p,L}(\mu,\nu;a_\psi) - FS3W^p_{\mathcal{R},p}(\mu,\nu;a_\psi)\right|\right] \leq \frac{L_W L_\psi C_d}{\sqrt{Ld}}, \tag{14}$$

Proposition 4 analyzes the error of Monte Carlo approximation for projection directions. The bound shows the convergence rate depends on the number of projections, problem dimension, and the Lipschitz regularity of the fusion network. This matches existing results for sliced-based methods, confirming FS3W retains their favorable projection complexity.

Proposition 5 (Neural Fusion Deviation Error). *Let a_ψ be a trained neural slicing fusion mechanism and a^* denote an ideal mapping that produces optimal discriminative projection directions. We define the network-induced error as:*

$$\epsilon_{\text{net}} := \left|FS3W(\mu,\nu;a_\psi) - FS3W(\mu,\nu;a^*)\right|. \tag{15}$$

Under mild regularity conditions on the fusion network and stereographic Wasserstein, we have the bound:

$$\epsilon_{\text{net}} \leq L_W L_\psi \|\psi - \psi^*\|. \tag{16}$$

Proposition 5 relates the accuracy of FS3W to the quality of the learned fusion mechanism. The bound shows that errors in the learned projection mapping translate linearly into errors in the resulting distance. This result highlights the importance of effective training and provides theoretical support for learning data-adaptive projection directions.

4 Implementation Details

In this section, we present the implementation details and a computational complexity analysis of FS3W, together with a runtime comparison against existing distance metrics.

We follow the standard SW procedure and use two empirical spherical distributions with the same sample size, $\mu = \frac{1}{n}\sum_{i=1}^n \delta_{x_i}$ and $\nu = \frac{1}{n}\sum_{i=1}^n \delta_{y_i}$, where each consists of n samples. First, we apply stereographic projection operator ϕ to map the points into the hyperplane, as defined in Eq. (5), then we randomly sample L projections $\{\theta_\ell\}_{\ell=1}^L$ from the uniform distribution σ on the unit sphere $\mathbb{S}^{d-1}$, thereafter, we adopt the neural slicing fusion mechanism $a_\psi(\mu,\nu,\theta)$ to obtain the improved projection directions as in Eq. (10). Next, we project the transformed samples onto each projection direction to form the slicing process. Together, the stereographic projection and the slicing process give the stereographic spherical Radon transform in Eq. (6). Finally, we approximate FS3W by Monte Carlo estimation as in Eq. (10).

Computation Complexity. As discussed in S3W [Tran *et al.*, 2024], the stereographic projection serves as a crucial preprocessing step to map samples from the unit hypersphere $\mathbb{S}^d$ to the Euclidean space $\mathbb{R}^d$. Performing the stereographic projection for all samples requires a complexity of $\mathcal{O}(nd)$, where n is the total number of samples from the source and target distribution and $(d+1)$ is the dimensionality of the data. The slicing process, which projects the transformed samples onto L projection directions, incurs a cost of $\mathcal{O}(Lnd)$. Sorting the data projections takes $\mathcal{O}(Ln\log n)$, and the final distance calculation requires $\mathcal{O}(Ln)$. In our FS3W formulation, the neural slicing fusion mechanism introduces an additional cost of $\mathcal{O}(LnT_{max})$, where T_{max} is the maximum iterations for training the nonlinear neural network or linear self-attention mechanism used to generate the enhanced projection directions. Therefore, the overall time complexity of our FS3W is $\mathcal{O}(Ln(d+\log n+T_{max}))$. For the RI-FS3W variant, which incorporates N_R rotations, the total complexity becomes $\mathcal{O}(N_R Ln(d+\log n+T_{max}))$. In addition, generating N_R random rotation matrices requires $\mathcal{O}(N_R d^3)$, and applying these rotations to the data incurs a cost of $\mathcal{O}(N_R nd^2)$. Consequently, the overall time complexity of RI-FS3W is $\mathcal{O}\left(N_R\left(d^3 + nd^2 + Ln(d+\log n+T_{max})\right)\right)$.

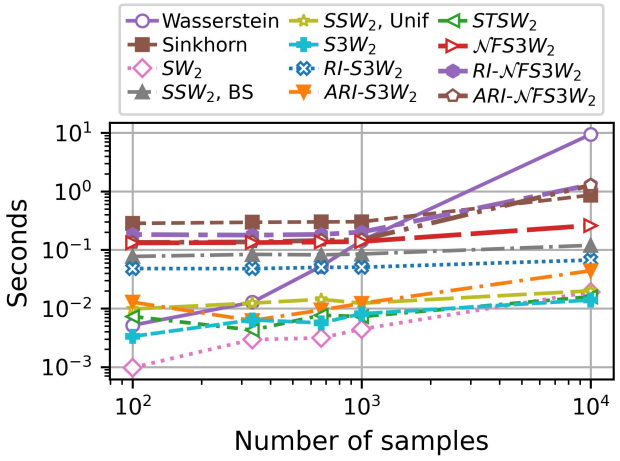


Figure 1: Runtime comparison for different distance metrics, averaged over 50 runs. $\mathcal{N}FSW_2$, RI-$\mathcal{N}FS3W_2$, and ARI-$\mathcal{N}FS3W_2$ denote our FS3W variants.

Runtime Comparison. Following the S3W protocol [Tran *et al.*, 2024], we compare the runtime of different distance metrics by computing distances between a uniform distribution and a von Mises–Fisher distribution on $\mathbb{S}^{100}$. Results in Fig. 1 are averaged over 50 runs across sample sizes. For all sliced-based methods, we use $L = 200$ projections. RI-S3W and ARI-S3W use $N_R = 10$ rotations, with an additional pool size of 100 for ARI-S3W. As shown in Fig. 1, FS3W variants are slower than their S3W counterparts due to the extra training cost of neural slicing fusion, yet remain much faster than Wasserstein and comparable to Sinkhorn. Moreover, their runtime grows smoothly with sample size, showing that fusion learning adds only modest overhead while pre-

serving the scalability of sliced-based methods. We further study the effects of dimension, projections, and rotations in Section. C.1, with detailed runtime analysis in Section. C.2.

5 Experiments

In this section, we present experimental results on four core tasks: Gradient Flow, Self-Supervised Learning, Earth Density Estimation, and Sliced-Wasserstein Auto-Encoder. For each task, we report quantitative metrics, qualitative visualizations, and comparisons with a range of state-of-the-art baseline methods. Together, these results show the effectiveness and versatility of our FS3W across different machine learning scenarios. Additional details and results are provided in Supplementary Material Section C.3-C.6. All experiments are implemented in PyTorch [Paszke *et al.*, 2019] on Ubuntu 20.04 using a single NVIDIA A100 GPU with 40GB RAM.

5.1 Gradient Flows on The Sphere

Similarly to S3W [Tran *et al.*, 2024] and STSW [Tran *et al.*, 2025], we use FS3W to learn a target distribution ν from a source distribution μ, initialized as a uniform distribution on the hypersphere, by minimizing $FS3W_2(\mu, \nu)$. We also employ the Projected Gradient Descent (PGD) algorithm [Madry *et al.*, 2018] to solve this optimization problem. We compare our approach with SW [Bonneel *et al.*, 2015], SSW [Bonet *et al.*, 2023], S3W variants [Tran *et al.*, 2024], DSSW [Zhang *et al.*, 2025] and STSW [Tran *et al.*, 2025].

Following S3W and STSW, we set the target distribution ν as a mixture of 12 vMFs with $\kappa = 50$, using 2400 samples evenly allocated across components.

The projected gradient descent [Bonet *et al.*, 2023] can be formulated as follows:

$$\begin{cases} x^{(k+1)} = x^{(k)} - \gamma \nabla_{x^{(k)}} \mathrm{FS3W}\left(\hat{\mu}_k, \nu\right), \\ x^{(k+1)} = \frac{x^{(k+1)}}{\left\|x^{(k+1)}\right\|_2}. \end{cases} \tag{17}$$

Setup. We set $L = 200$ trees and $k = 5$ lines for STSW, and use $L = 1000$ projections for the other methods. ARI-S3W (30) uses 30 rotations from a pool of 1000, while RI-S3W (1) and RI-S3W (5) use 1 and 5 rotations, respectively. Batch sizes are 200 and 2400 for mini- and full-batch settings, respectively. We train FS3W variants and other baselines with Adam using $lr = 0.01$ for full-batch and $lr = 0.001$ for mini-batch over 500 epochs and 10 runs. Following S3W and STSW, we use log 2-Wasserstein distance and negative log-likelihood (NLL) as evaluation metrics.

Results. The mini-batch results for all methods are reported in Table 1, showing that our FS3W variants outperform the baselines across all metrics. Furthermore, the Molleweide projections of the mini-batch are demonstrated in Fig. 2. We also report the full-batch results in Table A1 and provide the visualization of the Mollweide projections for full-batch projected gradient descent in Fig. C8 (Supplementary Material Section C.3). The full-batch results validate the same conclusion that our FS3W variants achieve better performance in approximating the target distribution than other distances.

The additional sliced-Wasserstein variational inference experiment shows that our FS3W variants consistently outperform or match other state-of-the-art baselines, with full results provided in Supplementary Material Section C.6.

Distance	NLL ↓	$\log W_2$ ↓
SW	-282.48 ± 17.42	-2.77 ± 0.10
SSW	-287.11 ± 6.22	-2.78 ± 0.08
S3W	-181.38 ± 8.72	-2.61 ± 0.07
RI-S3W (1)	-213.63 ± 19.24	-2.68 ± 0.09
RI-S3W (5)	-256.23 ± 10.72	-2.77 ± 0.13
RI-S3W (10)	-285.20 ± 13.22	-2.77 ± 0.11
ARI-S3W (30)	-291.38 ± 16.48	-2.82 ± 0.12
DSSW (nonlinear)	-319.96 ± 6.65	<u>-2.97 ± 0.15</u>
DSSW (attention)	-320.16 ± 5.58	-2.93 ± 0.10
$\mathcal{N}$FS3W	-242.04 ± 13.11	-2.62 ± 0.28
$\mathcal{LA}$FS3W	-252.06 ± 16.64	-2.70 ± 0.10
RI-$\mathcal{N}$FS3W (1)	-286.53 ± 11.74	-2.87 ± 0.07
RI-$\mathcal{LA}$FS3W (1)	-272.65 ± 12.69	-2.84 ± 0.15
ARI-$\mathcal{N}$FS3W (30)	<u>-351.82 ± 5.23</u>	**-3.04 ± 0.17**
ARI-$\mathcal{LA}$FS3W (30)	**-353.66 ± 5.93**	-2.96 ± 0.21

Table 1: Mini-batch results of learning a mixture of 12 vMFs. We use 1, 5, 10, and 30 rotations for RI-S3W (1), RI-S3W (5), RI-S3W (10), and ARI-S3W (30) with the extra pool size of 1000.

5.2 Earth Density Estimation

We apply FS3W to the density estimation on $\mathbb{S}^2$ using an exponential map normalizing flow model [Mathieu and Nickel, 2020]. This model defines an invertible transformation T trained by minimizing $\min_T FS3W_2(T_{\#}\mu, p_Z)$, where μ is the empirical distribution, and p_z is a prior distribution on $\mathbb{S}^2$. In this task, we use a uniform prior, following the setup of the earth dataset [Mathieu and Nickel, 2020], which contains Earthquake [NOAA, 2022], Flood [Brakenridge, 2017] and Fire [EOSDIS, 2020]. The density at any point $x \in \mathbb{S}^2$ is estimated asc $f_\mu(x) = p_Z(T(x))\left|\det J_T(x)\right|$, where $J_T(x)$ is the Jacobian matrix of T at x.

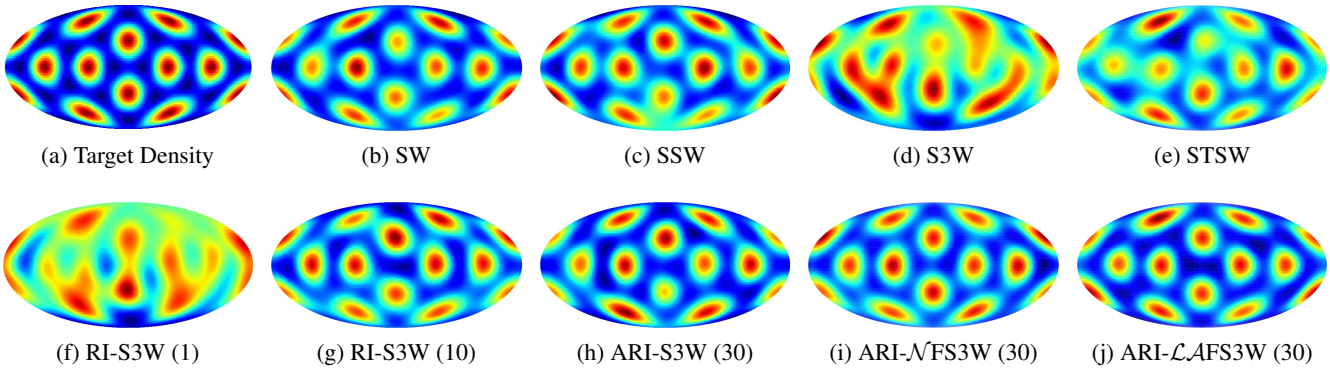
Following prior work [Bonet *et al.*, 2023; Tran *et al.*, 2024; Tran *et al.*, 2025], we adopt an exponential mapping normalizing flows model with 48 radial blocks and 100 components each, totaling 24000 parameters. The model is then trained with full batch gradient descent using the Adam optimizer.

Setup. We fix $L = 1000$ projection directions for SW, SSW, S3W, RI-S3W and ARI-S3W, while our FS3W variants only use $L = 100$ projection directions. We set the learning rate to $lr = 0.05$ for STSW, S3W, RI-S3W, ARI-S3W and our methods and $lr = 0.1$ for SW and SSW. We train other sliced dsitances for 20000 epochs, while our methods and STSW are only trained for 10000 epochs.

Results. We compare our method with SW, SSW, S3W variants, DSSW, and STSW, using NLL as the metric. Table 2 shows that ARI-FS3W achieves the best performance, indicating that FS3W models spherical data more effectively than the baselines. Our FS3W variants also require only 100 projection directions, which is fewer than the other methods. Density visualizations for FS3W on test data are provided in Figs. C9–C15 (Supplementary Section C.4) to give a clear view of the model behavior across datasets.

5.3 Sliced-Wasserstein Autoencoder

Similar to S3W [Tran *et al.*, 2024] and STSW [Tran *et al.*, 2025], we integrate FS3W into the Sliced-Wasserstein Auto-



(a) Target Density (b) SW (c) SSW (d) S3W (e) STSW

(f) RI-S3W (1) (g) RI-S3W (10) (h) ARI-S3W (30) (i) ARI-$\mathcal{N}$FS3W (30) (j) ARI-$\mathcal{LA}$FS3W (30)

Figure 2: The Mollweide projections for mini-batch projected gradient descent. We use 1, and 30 rotations for RI-S3W (1), and RI-S3W (10), respectively. We also use 10 rotations with a pool size of 1000 for ARI-S3W (30).

Method	Quake ↓	Flood ↓	Fire ↓
Stereo	2.04 ± 0.19	1.85 ± 0.03	1.34 ± 0.11
SW	1.12 ± 0.07	1.58 ± 0.02	0.55 ± 0.18
SSW	0.84 ± 0.05	1.26 ± 0.03	0.24 ± 0.18
S3W	0.88 ± 0.09	1.33 ± 0.05	0.36 ± 0.04
RI-S3W (1)	0.79 ± 0.07	1.25 ± 0.02	0.15 ± 0.06
ARI-S3W (50)	0.78 ± 0.06	1.24 ± 0.04	0.10 ± 0.04
DSSW (exp)	0.70 ± 0.09	1.22 ± 0.04	0.05 ± 0.08
DSSW (nonlinear)	0.71 ± 0.06	1.20 ± 0.03	0.05 ± 0.05
STSW	0.68 ± 0.04	1.23 ± 0.03	-0.07 ± 0.05
$\mathcal{N}$FS3W	0.73 ± 0.02	1.27 ± 0.03	0.37 ± 0.22
$\mathcal{LA}$FS3W	0.71 ± 0.03	1.28 ± 0.03	0.34 ± 0.20
RI-$\mathcal{N}$FS3W (1)	0.75 ± 0.02	1.24 ± 0.02	0.23 ± 0.07
RI-$\mathcal{LA}$FS3W (1)	0.76 ± 0.05	1.22 ± 0.03	0.25 ± 0.12
ARI-$\mathcal{N}$FS3W (50)	**0.58 ± 0.04**	**1.11 ± 0.02**	**-0.08 ± 0.01**
ARI-$\mathcal{LA}$FS3W (50)	0.60 ± 0.04	1.12 ± 0.02	**-0.08 ± 0.01**

Table 2: NLL on test data averaged over 5 runs with random data split. We use 1 rotation for RI-S3W (1). We also use 50 rotations with a pool size of 1000 for ARI-S3W (50).

Method	λ	$\log W_2$ ↓	NLL ↓	BCE ↓
Supervised AE	1	-0.5132	0.0060	0.6319
SW	0.001	-3.3229	-0.0007	0.6348
SSW	10	-2.1949	0.0052	0.6323
S3W	0.001	-3.3381	0.0025	0.6318
RI-S3W (5)	0.001	-3.1424	-0.0043	0.6376
ARI-S3W (5)	0.001	-3.3853	0.0028	0.6341
STSW	10	-3.4191	-0.0051	0.6341
DSSW (identity)	10	-3.4203	-0.0011	0.6318
DSSW (attention)	10	-3.4242	-0.0002	0.6324
$\mathcal{N}$FS3W	0.01	-3.3453	-0.0014	0.6312
$\mathcal{LA}$FS3W	0.01	-3.3874	-0.0015	0.6312
RI-$\mathcal{N}$FS3W (5)	0.01	-3.4146	**-0.0070**	0.6306
RI-$\mathcal{LA}$FS3W (5)	0.01	-3.3658	-0.0046	**0.6303**
ARI-$\mathcal{N}$FS3W (5)	0.01	**-3.4342**	-0.0064	0.6304
ARI-$\mathcal{LA}$FS3W (5)	0.01	-3.4189	0.0019	0.6307

Table 3: SWAE results on CIFAR-10 test data with a mixture of 10 vMFs as the prior on $\mathbb{S}^2$. RI-S3W (5) uses 5 rotations, and ARI-S3W (5) uses 5 rotations with a pool size of 1000.

Encoder (SWAE) [Kolouri *et al.*, 2019b] framework to evaluate the performance of generative modeling, which forces the latent space distribution to follow a prior distribution ν. let $E : \mathcal{X} \to \mathbb{S}^d$ and $D : \mathbb{S}^d \to \mathcal{X}$ be the parametric encoder and decoder. The training objective of the revised SWAE is

$$\min_{E,D} \mathbb{E}_{x\sim\mu}\left[C(x, D(E(x)))\right] + \lambda \cdot FS3W_2(E_{\#}\mu, \nu)\,, \quad (18)$$

where μ is the source distribution, $C(\cdot,\cdot)$ is the reconstruction loss implemented as the standard Binary Cross Entropy (BCE) loss, λ means the regularization coefficient.

Setup. Following prior works [Bonet *et al.*, 2023; Tran *et al.*, 2025; Zhang *et al.*, 2025], We train with Adam optimizer with a learning rate of 10^{-3} and a batch size of 500 over 100 epochs using BCE loss as our reconstruction loss. We fix $L = 200$ trees and $\kappa = 10$ lines for STSW [Tran *et al.*, 2025], we set $L = 100$ projection directions for all other methods. We use $N_R = 5$ rotations for RI-S3W and ARI-S3W, with ARI-S3W using a pool size of 100 random rotations. For CIFAR-10, use a mixture of 10 vMFs as the prior, setting $\lambda = 1$ for STSW, $\lambda = 10$ for SSW, $\lambda = 0.01$ for our FS3W variants, and $\lambda = 0.001$ for SW and S3W variants. In addition, we use the same model architecture from [Tran *et al.*, 2024].

Results. We evaluate FS3W against SW, SSW, S3W variants, DSSW, and STSW on MNIST [LeCun *et al.*, 1998] and CIFAR-10 [Krizhevsky and Hinton, 2009] over $\mathbb{S}^2$. For CIFAR-10, we use a mixture of 10 vMFs as the prior and report log 2-Wasserstein distance, NLL, and BCE loss. As shown in Table 3, FS3W variants achieve the best results across all metrics. Detailed configurations and MNIST results, including FID scores in Table B3 and generated images in Fig. C16, are provided in Supplementary Material Sec. C.5, further confirming the performance gains of FS3W.

5.4 Self-Supervised Representation Learning

Normalizing feature vectors onto the unit hypersphere improves representation quality and helps prevent feature col-

lapse [Wang and Isola, 2020]. As shown in [Wang and Isola, 2020], contrastive learning decomposes into alignment, which pulls positive pairs together, and uniformity, which spreads features across the hypersphere. Following S3W and STSW, we replace the Gaussian kernel uniformity loss with FS3W, resulting in the objective

$$\frac{1}{n}\sum_{i=1}^{n}\underbrace{\|z_i^A - z_i^B\|_2^2}_{\text{Alignment loss}} + \frac{\lambda}{2}\underbrace{\left(FS3W_2(z^A,\nu)+FS3W_2(z^B,\nu)\right)}_{\text{Uniformity loss}}, \quad (19)$$

where $z^A, z^B \in \mathbb{R}^{n\times(d+1)}$ are embeddings of two augmented views of the same input, projected onto the unit hypersphere $\mathbb{S}^d$. The factor $\lambda > 0$ balances alignment and uniformity, and $\nu = \mathcal{U}\left(\mathbb{S}^d\right)$ denotes the uniform hyperspherical prior.

Method	Acc. E(%) ↑	Acc. P(%) ↑
Supervised	92.38	91.77
hypersphere	79.76	74.57
SimCLR	79.69	72.78
SW-SSL (λ=1, L=200)	74.45	68.35
SSW-SSL (λ=20, L=200)	70.46	64.52
S3W-SSL (λ=0.5, L=200)	78.54	73.84
RI-S3W(5)-SSL (λ=0.5, L=200)	79.97	74.27
ARI-S3W(5)-SSL (λ=0.5, L=200)	79.92	75.07
DSSW-SSL (exp, λ=100, L=200)	80.10	76.30
DSSW-SSL (linear, λ=100, L=200)	80.15	76.87
STSW-SSL (λ=10.0, L=200, $k = 20$)	80.53	76.78
$\mathcal{N}$FS3W (λ=0.2, L=200)	80.41	**78.37**
$\mathcal{LA}$FS3W (λ=0.4, L=200)	79.67	76.60
RI-$\mathcal{N}$FS3W (5) (λ=0.3, L=200)	80.50	77.03
RI-$\mathcal{LA}$FS3W (5) (λ=0.4, L=200)	80.27	77.01
ARI-$\mathcal{N}$FS3W (5) (λ=0.3, L=200)	<u>80.81</u>	<u>77.37</u>
ARI-$\mathcal{LA}$FS3W (5) (λ=0.4, L=200)	**81.00**	76.77

Table 4: CIFAR-10 linear evaluation precision for encoded features (E) and projected (P) features on $\mathbb{S}^9$. RI-S3W (5) uses 5 rotations, and ARI-S3W (5) uses 5 rotations with a pool size of 1000.

Setup. Similar to [Tran *et al.*, 2025], we train a ResNet18 on CIFAR-10 for 200 epochs with batch size 512, using SGD with $lr = 0.05$, momentum 0.9, and weight decay 10^{-3}. Positive pairs are generated by standard augmentations, including resizing/cropping, flipping, color jittering, and grayscale conversion [Zhang *et al.*, 2025]. We then train a linear classifier on the learned features for 100 epochs using Adam with $lr = 10^{-3}$ and decay 0.2 at epochs 60 and 80, following SSW. For STSW, we set $L = 200$ trees and $k = 20$ lines; other sliced distances use $L = 200$ projections. Following [Tran *et al.*, 2024], RI-S3W and ARI-S3W use $N_R = 5$ rotations from a pool of 100 random rotations.

Results. Table 4 shows that ARI-$\mathcal{LA}$FS3W achieves the best linear evaluation accuracy on encoded features, while $\mathcal{N}$FS3W reaches the highest accuracy on projected features on $\mathbb{S}^9$. Compared with the original S3W variants, FS3W performs better on both encoded and projected features on CIFAR-10, demonstrating clear gains across settings. The visualizations in Fig. 3 shows that FS3W produces uniform and well-clustered embeddings on $\mathbb{S}^2$, highlighting its strength in spherical representation learning.

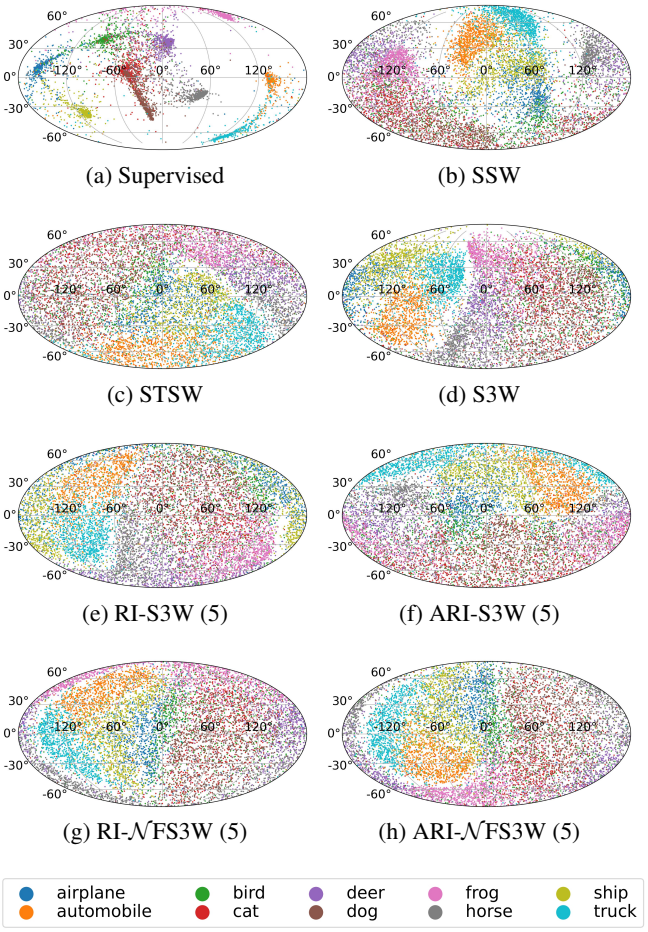


Figure 3: Projected features on $\mathbb{S}^2$ for CIFAR10

6 Conclusion

This work introduces FS3W, a new approach for comparing probability distributions on the hypersphere. FS3W adapts a neural slicing fusion mechanism to refine directions adaptively using data-dependent information. By aligning directions with the discriminative geometric structures in the data distributions, FS3W provides a more expressive and reliable measure of distributional discrepancy, demonstrating strong performance in tasks like gradient flow, Earth density estimation, and self-supervised learning. Future work will focus on reducing the additional computational cost introduced by neural training and improving the efficiency of the method. The fusion mechanism is both general and modular, and it can be extended to the sliced Wasserstein distance in Euclidean spaces and other non-Euclidean settings, including hyperbolic spaces and manifolds. These extensions promise broader applications of data-adaptive sliced optimal transport and offer potential for further advancements in the field.

Acknowledgements

This work was supported by the NSFC under Grant Nos. 62276135, 62176124, U24A20330 and 62361166670, as well as by the Basic Research Program of Jiangsu under Grant No. BK20253028.

References

[Besombes *et al.*, 2021] Camille Besombes, Olivier Pannekoucke, Corentin Lapeyre, Benjamin Sanderson, and Olivier Thual. Producing realistic climate data with generative adversarial networks. *Nonlinear Processes in Geophysics*, 28(3):347–370, 2021.

[Bonet *et al.*, 2023] Clément Bonet, Paul Berg, Nicolas Courty, François Septier, Lucas Drumetz, and Minh Tan Pham. Spherical sliced-wasserstein. In *The Eleventh International Conference on Learning Representations*, 2023.

[Bonneel *et al.*, 2015] Nicolas Bonneel, Julien Rabin, Gabriel Peyré, and Hanspeter Pfister. Sliced and radon wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51:22–45, 2015.

[Brakenridge, 2017] G Brakenridge. Global active archive of large flood events, 2017.

[Chen *et al.*, 2020] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 2020.

[Cui *et al.*, 2019] Li Cui, Xin Qi, Chengfeng Wen, Na Lei, Xinyuan Li, Min Zhang, and Xianfeng Gu. Spherical optimal transportation. *Computer-Aided Design*, 115:181–193, 2019.

[Cuturi, 2013] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 2292–2300, 2013.

[Deshpande *et al.*, 2019] Ishan Deshpande, Yuan-Ting Hu, Ruoyu Sun, Ayis Pyrros, Nasir Siddiqui, Sanmi Koyejo, Zhizhen Zhao, David A Forsyth, and Alexander G Schwing. Max-sliced wasserstein distance and its use for gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10640–10648, 2019.

[Di Marzio *et al.*, 2014] Marco Di Marzio, Agnese Panzera, and Charles C Taylor. Nonparametric regression for spherical data. *Journal of the American Statistical Association*, 109(506):748–763, 2014.

[Dominitz and Tannenbaum, 2009] Ayelet Dominitz and Allen Tannenbaum. Texture mapping via optimal mass transport. *IEEE transactions on visualization and computer graphics*, 16(3):419–433, 2009.

[EOSDIS, 2020] EOSDIS. Land, atmosphere near real-time capability for eos (lance) system operated by nasa's earth science data and information system (esdis), 2020.

[Groemer, 1998] H Groemer. On a spherical integral transformation and sections of star bodies. *Monatshefte für Mathematik*, 126(2):117–124, 1998.

[Helgason and others, 2011] Sigurdur Helgason et al. *Integral geometry and Radon transforms*, volume 1. Springer, 2011.

[Hu *et al.*, 2023] Xiantao Hu, Bineng Zhong, Qihua Liang, Shengping Zhang, Ning Li, Xianxian Li, and Rongrong Ji. Transformer tracking via frequency fusion. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(2):1020–1031, 2023.

[Hu *et al.*, 2025] Xiantao Hu, Ying Tai, Xu Zhao, Chen Zhao, Zhenyu Zhang, Jun Li, Bineng Zhong, and Jian Yang. Exploiting multimodal spatial-temporal patterns for video object tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 3581–3589, 2025.

[Jammalamadaka and Sengupta, 2001] S Rao Jammalamadaka and Ambar Sengupta. *Topics in circular statistics*, volume 5. world scientific, 2001.

[Kolouri *et al.*, 2019a] Soheil Kolouri, Kimia Nadjahi, Umut Simsekli, Roland Badeau, and Gustavo K. Rohde. Generalized sliced wasserstein distances. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 261–272, 2019.

[Kolouri *et al.*, 2019b] Soheil Kolouri, Phillip E. Pope, Charles E. Martin, and Gustavo K. Rohde. Sliced wasserstein auto-encoders. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[Krizhevsky and Hinton, 2009] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, Toronto, Ontario, 2009.

[LeCun *et al.*, 1998] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998.

[Madry *et al.*, 2018] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[Mathieu and Nickel, 2020] Emile Mathieu and Maximilian Nickel. Riemannian continuous normalizing flows. *Advances in Neural Information Processing Systems*, 33:2503–2515, 2020.

[Nadjahi *et al.*, 2020] Kimia Nadjahi, Alain Durmus, Lénaïc Chizat, Soheil Kolouri, Shahin Shahrampour, and Umut Simsekli. Statistical and topological properties of sliced probability divergences. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 20802–20812. Curran Associates, Inc., 2020.

[Nguyen and Ho, 2024] Khai Nguyen and Nhat Ho. Energy-based sliced wasserstein distance. *Advances in Neural Information Processing Systems*, 36, 2024.

[Nguyen *et al.*, 2021] Khai Nguyen, Nhat Ho, Tung Pham, and Hung Bui. Distributional sliced-wasserstein and applications to generative modeling. In *International Conference on Learning Representations*, 2021.

[Nguyen *et al.*, 2023] Khai Nguyen, Dang Nguyen, and Nhat Ho. Self-attention amortized distributional projection optimization for sliced wasserstein point-cloud reconstruction. In *International Conference on Machine Learning*, pages 26008–26030. PMLR, 2023.

[Nietert *et al.*, 2022] Sloan Nietert, Ziv Goldfeld, Ritwik Sadhu, and Kengo Kato. Statistical, robustness, and computational guarantees for sliced wasserstein distances. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 28179–28193. Curran Associates, Inc., 2022.

[NOAA, 2022] NOAA. Ncei/wds global significant earthquake database, 2022.

[Ohana *et al.*, 2023] Ruben Ohana, Kimia Nadjahi, Alain Rakotomamonjy, and Liva Ralaivola. Shedding a PAC-Bayesian light on adaptive sliced-Wasserstein distances. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pages 26451–26473, 2023.

[Paszke *et al.*, 2019] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035, 2019.

[Peyré *et al.*, 2019] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

[Peyré and Cuturi, 2019] Gabriel Peyré and Marco Cuturi. Computational optimal transport: With applications to data science. *Foundations and TrendsÂ® in Machine Learning*, 11(5-6):355–607, 2019.

[Quellmalz *et al.*, 2023] Michael Quellmalz, Robert Beinert, and Gabriele Steidl. Sliced optimal transport on the sphere. *Inverse Problems*, 39(10):105005, 2023.

[Rubin, 2018] Boris Rubin. The vertical slice transform in spherical tomography, 2018.

[Tran *et al.*, 2024] Huy Tran, Yikun Bai, Abihith Kothapalli, Ashkan Shahbazi, Xinran Liu, Rocio Diaz Martin, and Soheil Kolouri. Stereographic spherical sliced wasserstein distances. In *International Conference on Machine Learning*, 2024.

[Tran *et al.*, 2025] Hoang V. Tran, Thanh Chu, Minh-Khoi Nguyen-Nhat, Huyen Trang Pham, Tam Le, and Tan Minh Nguyen. Spherical tree-sliced wasserstein distance. In *The Thirteenth International Conference on Learning Representations*, 2025.

[Vrba and Robinson, 2001] Jiri Vrba and Stephen E Robinson. Signal processing in magnetoencephalography. *Methods*, 25(2):249–271, 2001.

[Wang and Isola, 2020] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 9929–9939. PMLR, 2020.

[Zhang *et al.*, 2025] Hongliang Zhang, Shuo Chen, Lei Luo, and Jian Yang. Towards better spherical sliced-wasserstein distance learning with data-adaptive discriminative projection direction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 22425–22433, 2025.