LINEAR SPHERICAL SLICED OPTIMAL TRANSPORT: A FAST METRIC FOR COMPARING SPHERICAL DATA

Anonymous authors
Paper under double-blind review

ABSTRACT

Efficient comparison of spherical probability distributions becomes important in fields such as computer vision, geosciences, and medicine. Sliced optimal transport distances, such as spherical and stereographic spherical sliced Wasserstein distances, have recently been developed to address this need. These methods reduce the computational burden of optimal transport by slicing hyperspheres into one-dimensional projections, i.e., lines or circles. Concurrently, linear optimal transport has been proposed to embed distributions into L^2 spaces, where the L^2 distance approximates the optimal transport distance, thereby simplifying comparisons across multiple distributions. In this work, we introduce the Linear Spherical Sliced Optimal Transport (LSSOT) framework, which utilizes slicing to embed spherical distributions into L^2 spaces while preserving their intrinsic geometry, offering a computationally efficient metric for spherical probability measures. We establish the metricity of LSSOT and demonstrate its superior computational efficiency in applications such as cortical surface registration, 3D point cloud interpolation via gradient flow, and shape embedding. Our results demonstrate the significant computational benefits and high accuracy of LSSOT in these applications.

1 INTRODUCTION

Spherical signal analysis has received increasing attention in many domains such as computer vision, astrophysics (Jarosik et al., 2011), climatology (Garrett et al., 2024) and neuroscience (Elad & Kimmel, 2002; Zhao et al., 2019; 2023). In practice, certain types of data are intrinsically defined on a sphere, namely omnidirectional images, ambisonics, and earth data. In addition, some forms of data are studied canonically in hyperspheres by spherical mappings (Gotsman et al., 2003; Gu et al., 2004; Shen & Makedon, 2006; Chung & Chen, 2022) from the original data spaces. Among them, a notable category of data is the genus-zero surfaces, which can be conformally mapped to unit spheres according to the Poincare uniformization theorem (Abikoff, 1981). Cortical surface-based analysis, for example, can be facilitated by the inherent spherical topology (0-genus) of the cerebral cortex. Compared to conventional 3D volumetric analysis, applying spherical representations of cortical surfaces offers a more streamlined and accurate geometric framework to examine the brain. This approach proves particularly advantageous in both cross-sectional and longitudinal studies of brain structure and function (Zhao et al., 2021; 2023).

Analyzing spherical data through machine learning techniques necessitates metrics that can effectively compare signals while maintaining data structure and geometry. In this context, optimal transport (OT) has emerged as a versatile and widely adopted tool in the machine learning community. OT provides a framework for comparing probability distributions in various domains (Kolouri et al., 2017; Peyré et al., 2019; Kolkin et al., 2019; Arjovsky et al., 2017; Seguy et al., 2017; **?**; Makkuva et al., 2020; Cang et al., 2023). OT has demonstrated remarkable effectiveness not only in Euclidean spaces, but its theoretical foundations have also been firmly established for various other manifolds (Fathi & Figalli, 2010; McCann & Topping, 2010), especially on spheres (Cui et al., 2019; Hamfeldt & Turnquist, 2022). However, the calculation of OT distances is prohibitively expensive with a cubic cost $\mathcal{O}(N^3)$. This computational burden poses a substantial challenge to applications where the input size N is large. In light of this, researchers have enthusiastically pursued and developed faster alternatives, including entropic optimal transport (Cuturi, 2013) and sliced optimal transport (Bonnotte, 2013; Bonneel et al., 2015; Kolouri et al., 2016; 2018; Deshpande et al., 2018; Kolouri et al., 2019).

Sliced Optimal Transport (SOT) accelerates computation by projecting high-dimensional data onto one-dimensional subspaces and applying efficient OT solvers for the resulting one-dimensional probability distributions. In this paper, we are interested in sliced optimal transport methods on hyperspheres. Related work includes the Spherical Sliced Wasserstein (SSW) discrepancy (Bonet et al., 2023), which proposes slicing the sphere by the spherical Radon transform and leveraging circular optimal transport; and the Stereographic Spherical Sliced Wasserstein (S3W) distance (Tran et al., 2024), which utilizes stereographic projection to solve the SOT problem in Euclidean space with minimal distortion.

Many machine learning applications, from manifold learning to k-nearest neighbor classification and regression, rely on pairwise distance calculations among members of a set, often requiring distance evaluations on the order of the square of the number of objects. To accelerate pairwise calculation of OT distances between probability measures, Linear Optimal Transport (LOT) (Wang et al., 2013) was developed as a framework that embeds distributions into L^2 spaces where pairwise distance computations can be performed with significantly higher efficiency. For example, Wasserstein Embedding and Sliced Wasserstein Embedding have been applied in Euclidean spaces to enable efficient image analysis Basu et al. (2014), faster graph learning (Kolouri et al., 2020), supervised task transferability prediction (Liu et al., 2022), point cloud retrieval (Lu et al., 2024), etc. A more recent work, Martin et al. (2023) focused on $\mathbb{S}^1$ and established a computationally efficient Linear Circular Optimal Transport (LCOT) framework to embed circular distributions into a L^2 space.

In this paper, we build on the work of Martin et al. (2023) and extend the linearized methodology from circles to hyperspheres. We develop a novel embedding technique for spherical probability distributions, which significantly enhances the efficiency of group analysis in spherical domains.

Contributions. In summary, our contributions are as follows:

1. We propose Linear Spherical Sliced Optimal Transport (LSSOT) to embed spherical distributions into L^2 space while preserving their intrinsic spherical geometry.
2. We prove that LSSOT defines a metric, and demonstrate the superior computation efficiency over other baseline metrics.
3. We conduct a comprehensive set of experiments to show the effectiveness and efficiency of LSSOT in diverse applications, from point cloud analysis to cortical surface registration.

2 Background

2.1 Circular Optimal Transport and Linear Circular Optimal Transport

Consider two circular probability measures $\mu, \nu \in \mathcal{P}(\mathbb{S}^1)$, where $\mathbb{S}^1$ denotes the unit circle in $\mathbb{R}^2$. Let us parametrize $\mathbb{S}^1$ with the angles in between 0 and 1 and consider the cost function $c(x, y) := h(|x - y|_{\mathbb{S}^1})$, where $h : \mathbb{R} \to \mathbb{R}_+$ is a convex increasing function and $|x - y|_{\mathbb{S}^1} := \min\{|x - y|, 1 - |x - y|\}$ for $x, y \in [0, 1)$. The Circular Optimal Transport (COT) problem between μ and ν is defined by the following two equivalent minimization problems:

$$COT_h(\mu, \nu) := \inf_{\gamma \in \Gamma(\mu,\nu)} \int_{\mathbb{S}^1 \times \mathbb{S}^1} c(x, y) d\gamma(x, y) = \inf_{\alpha \in \mathbb{R}} \int_0^1 h(|F_\mu^{-1}(x) - F_\nu^{-1}(x - \alpha)|) dx, \quad (1)$$

where in the first expression $\Gamma(\mu, \nu)$ is the set of all couplings between μ and ν, and in the second expression F_μ (respectively, F_ν) is the cumulative distribution function of μ on $\mathbb{S}^1$ (i.e., $F_\mu(y) := \mu([0, y)) = \int_0^y d\mu, \forall y \in [0, 1)$) extended on $\mathbb{R}$ by $F_\mu(y + n) = F_\mu(y) + n, \forall y \in [0, 1), n \in \mathbb{Z}$, and its inverse (or quantile function) is defined by $F_\mu^{-1}(y) = \inf\{x : F_\mu(x) > y\}$. When $h(x) = |x|^p$ for $1 \leq p < \infty$, we denote $COT_h(\cdot, \cdot)$ as $COT_p(\cdot, \cdot)$, and $COT_p(\cdot, \cdot)^{1/p}$ defines a metric on $\mathcal{P}(\mathbb{S}^1)$. Moreover, if $\mu = \text{Unif}(\mathbb{S}^1)$ and $h(x) = |x|^2$, the minimizer $\alpha_{\mu,\nu}$ of (1) is the antipodal of $\mathbb{E}(\nu)$, i.e., $\alpha_{\mu,\nu} = \mathbb{E}(\nu) - 1/2$. We refer the reader to Delon et al. (2010); Rabin et al. (2011); Bonet et al. (2023) for more details.

Recently, Martin et al. (2023) expanded the linear optimal transport framework (Wang et al., 2012) to encompass circular probability measures, introducing the Linear Circular Optimal Transport (LCOT) embedding and metric. For the reader's convenience, we summarize the key definitions from their work below.

The **LCOT embedding** corresponds to the optimal circular displacement that comes from the problem of transporting a *reference measure* to a *target measure* with respect to $COT_h(\cdot,\cdot)$. Precisely, given a reference measure $\mu \in \mathcal{P}(\mathbb{S}^1)$ that is absolutely continuous with respect to the Lebesgue measure on $\mathbb{S}^1$, the LCOT embedding of a target measure $\nu \in \mathcal{P}(\mathbb{S}^1)$ is defined by

$$\widehat{\nu}^{\mu,h}(x) := F_\nu^{-1}(F_\mu(x) - \alpha_{\mu,\nu}) - x, \quad \forall x \in [0,1), \tag{2}$$

where $\alpha_{\mu,\nu}$ is a minimizer of (1). When $h(x) = |x|^p$, for $1 \leq p < \infty$, this embedding gives rise to the **LCOT distance** between circular measures:

$$LCOT_{\mu,p}(\nu_1,\nu_2) := \|\widehat{\nu_1}^{\mu,h} - \widehat{\nu_2}^{\mu,h}\|_{L^p(\mathbb{S}^1,d\mu)}, \qquad \forall \nu_1,\nu_2 \in \mathcal{P}(\mathbb{S}^1), \tag{3}$$

where

$$L^p(\mathbb{S}^1,d\mu) := \{f : \mathbb{S}^1 \to \mathbb{R} : \|f\|^p_{L^p(\mathbb{S}^1,d\mu)} := \int_{\mathbb{S}^1} |f(t)|^p_{\mathbb{S}^1}\, d\mu(t) < \infty\}. \tag{4}$$

If $\mu = Unif(\mathbb{S}^1)$, we use the notation $L^p(\mathbb{S}^1) := L^p(\mathbb{S}^1, d\mu)$. If also $h(x) = |x|^2$, the LCOT embedding becomes

$$\widehat{\nu} := F_\nu^{-1}\left(x - \mathbb{E}(\nu) + \frac{1}{2}\right) - x, \qquad \forall x \in [0,1), \tag{5}$$

and we denote the LCOT distance simply as $LCOT_2(\cdot,\cdot)$.

2.2 Spherical Slicing

Bonet et al. (2023) recently presented the Spherical Sliced Wasserstein (SSW) discrepancy for probability measures supported on the sphere $\mathbb{S}^{d-1} \subset \mathbb{R}^d$, that projects $\forall x \in \mathbb{S}^{d-1}$ on great circles C to:

$$\arg\min_{y \in C}\{\arccos(\langle x,y\rangle)\}, \tag{6}$$

where $\arccos(\langle\cdot,\cdot\rangle)$ is the geodesic distance on $\mathbb{S}^{d-1}$. Since each great circle is obtained by intersecting the sphere with a 2-dimensional plane in $\mathbb{R}^d$, the authors proposed to average over all great circles by integrating over all planes in $\mathbb{R}^d$ (i.e., the Grassmann manifold (Bendokat et al., 2020)) and then project the distributions onto the intersection with $\mathbb{S}^{d-1}$. However, due to the impracticality of the Grassmann manifold, the set of rank 2 projectors is used instead:

$$V_2(\mathbb{R}^d) := \{U \in \mathbb{R}^{d\times 2} : U^T U = I_2\},$$

where $V_2(\mathbb{R}^d)$ is the Stiefel manifold (Bendokat et al., 2020). Hence, the integration process can be performed over $V_2(\mathbb{R}^d)$ according to the uniform distribution $\sigma \in \mathcal{P}(V_2(\mathbb{R}^d))$. The projection onto $U \in V_2(\mathbb{R}^d)$ can be computed as in (Bonet et al., 2023, Lemma 1) by

$$P^U(x) = \frac{U^T x}{\|U^T x\|_2}, \qquad \forall x \in \mathbb{S}^{d-1} \setminus \{x : U^T x = 0\} \tag{7}$$

Finally, given $\mu,\nu \in \mathcal{P}(\mathbb{S}^{d-1})$, the Spherical Sliced Wasserstein discrepancy between them, for $1 \leq p < \infty$, is defined in (Bonet et al., 2023) by

$$(SSW_p(\mu,\nu))^p := \int_{V_2(\mathbb{R}^d)} \left(COT_p(P^U_\#\mu, P^U_\#\nu)\right)^p \, d\sigma(U). \tag{8}$$

3 Method

In this section, we first introduce the Linear Spherical Sliced Optimal Transport (LSSOT) embedding and discrepancy, and then state the main theorem —- LSSOT indeed defines a metric in $\mathcal{P}(\mathbb{S}^{d-1})$.

3.1 Linear Spherical Sliced Optimal Transport (LSSOT)

Definition 3.1. *Consider spherical measures $\nu_1,\nu_2 \in \mathcal{P}(\mathbb{S}^{d-1})$. We define the **Linear Spherical Sliced Optimal Transport (LSSOT) embedding** of ν_i, $i = 1,2$, with respect to a slice $U \in V_2(\mathbb{R}^d)$ by*

$$\widehat{\nu_i}^S(x,U) := F^{-1}_{P^U_\#\nu_i}\left(P^U(x) - \mathbb{E}(P^U_\#\nu_i) + \frac{1}{2}\right), \quad x \in \mathbb{S}^{d-1},\, U \in V_2(\mathbb{R}^d), \tag{9}$$

where $P^U_\# \nu_i$ denotes the pushforward measure on $\mathbb{S}^1$ of ν_i by the map P^U. We also define their ***LSSOT discrepancy*** *by*

$$\left(LSSOT_2(\nu_1,\nu_2)\right)^2 := \int_{V_2(\mathbb{R}^d)} \left(LCOT_2(P^U_\#\nu_1, P^U_\#\nu_2)\right)^2 \, d\sigma(U) \tag{10}$$
$$= \int_{V_2(\mathbb{R}^d)} \|\widehat{\nu_1}^S(\cdot,U) - \widehat{\nu_2}^S(\cdot,U)\|^2_{L^2(\mathbb{S}^1)} \, d\sigma(U),$$

Remark 3.2 (LSSOT extends LCOT). *When $d = 2$, $P^U = I_2$ for every $U \in V_2(\mathbb{R}^2)$ since $V_2(\mathbb{R}^2)$ coincides with the set of orthonormal basis on $\mathbb{R}^2$. Thus, for any $\nu \in \mathcal{P}(\mathbb{S}^1)$ we have that $\widehat{\nu}^S(x,U) = \widehat{\nu}^S(x,V)$ for all $U, V \in V_2(\mathbb{R}^2)$, so the LSSOT embedding and the LSSOT discrepancy coincide with the LCOT embedding and with the LCOT distance respectively, as the integration is normalized to 1 on $V_2(\mathbb{R}^2)$ (which is isomorphic to the orthogonal group $O(2)$ of dimension 1).*

Theorem 3.3 (Metric property of LSSOT). *$LSSOT_2(\cdot,\cdot)$ defines a pseudo-metric in $\mathcal{P}(\mathbb{S}^{d-1})$, and a metric when restricting to probability measures with continuous density functions. We will refer to it as $LSSOT$ distance.*

We refer the reader to the Appendix for detailed proof. The symmetry of $LSSOT_2(\cdot,\cdot)$ is easy to deduce from its definition. The triangle inequality holds as an application of Minkowski inequality by using the fact that $LCOT_2(\cdot,\cdot)$ is a metric (see Proposition D.1 in Appendix D). For proving the identity of indiscernibles, the idea is to give an alternative expression of the $LSSOT_2(\cdot,\cdot)$ involving a Spherical Radon Transform $\mathcal{R}$: see (45) in Appendix C.1, which is inspired by Bonet et al. (2023). Such formula reads for probability density functions $f_1, f_2 \in L^1(\mathbb{S}^{d-1})$[1] as

$$\left(LSSOT_2(\nu_1,\nu_2)\right)^2 = \int_{V_2(\mathbb{R}^d)} \left(LCOT_2(\mathcal{R}f_1(U,\cdot), \mathcal{R}f_2(U,\cdot))\right)^2 \, d\sigma(U) \tag{11}$$

where $\mathcal{R} : L^1(\mathbb{S}^{d-1}) \to L^1(V_2(\mathbb{R}^d) \times \mathbb{S}^1)$ is the integral operator

$$\mathcal{R}f(U,z) = \int_{\{x \in \mathbb{S}^{d-1} \colon z = P^U(x)\}} f(x)dx,$$

where the surface/line integral is taken over a $(d-2)$-dimensional surface, which coincides with a (1-dimensional) *great semicircle* when $d = 3$. We will call this transformation as *Semicircle Transform* when $d = 3$, and $(d-2)$-*Hemispherical Transform on* $\mathbb{S}^{d-1}$ in general. We describe $\mathcal{R}$ acting on $L^1(\mathbb{S}^2)$ by using different and detailed parameterizations in Appendix A and generalize it for any dimension d in Appendix B. In Appendix C, $\mathcal{R}$ is extended to the set of measures on the sphere, and the disintegration theorem from measure theory is used to obtain the general version of (11). The **novelty** is that we will relate $\mathcal{R}$ with an integral transformation presented in Groemer (1998), instead of relating it with the Hemispherical Transform in Rubin (1999) as done in Bonet et al. (2023) (see Remark B.8 for a discussion in this regard). By doing so, we can prove the injectivity of $\mathcal{R}$ on continuous functions (Corollary B.7), which is the key property to show that if $LSSOT_2(\nu_1,\nu_2) = 0$ then $\nu_1 = \nu_2$. As a byproduct, we also obtain that $SSW_p(\cdot,\cdot)$ is not only a pseudo-metric as proven in Bonet et al. (2023), but also a metric when it is restricted to probability measures with continuous density functions (Theorem D.3 in Appendix D).

3.2 LSSOT Implementation

Following the Monte-Carlo approach for computing the classical Sliced Wasserstein distance, we consider samples from the uniform distribution $\sigma \in \mathcal{P}(V_2(\mathbb{R}^d))$ for generating the slices. To do so, a standard procedure is to first construct matrices in $\mathbb{R}^{d\times 2}$ whose components are drawn from the standard normal distribution $\mathcal{N}(0,1)$, then use the Gram-Schmidt algorithm to orthonormalize the column vectors of the matrix, or apply the QR-decomposition and finally

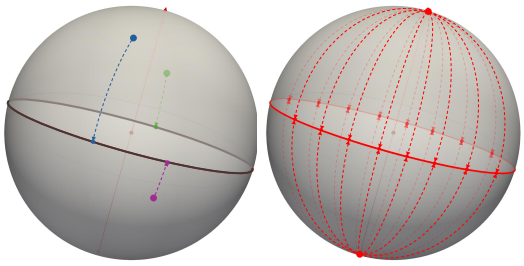

Figure 1: Semicircle Transform projects non-polar points to the closest location on the circle, and the north/south poles to everywhere on the circle.

[1](which is equivalent to considering two absolutely continuous probability measures on $\mathbb{S}^{d-1}$)

Algorithm 1 Linear Spherical Sliced Optimal Transport (LSSOT)

Require: Spherical distributions $\mu = \sum_{i=1}^{N_\mu} a_i \delta_{x_i}$ and $\nu = \sum_{j=1}^{N_\nu} b_j \delta_{y_j}$; number of slices L; reference size M, threshold ϵ

for $l = 1$ to L **do**

Construct a matrix $Z_l \in \mathbb{R}^{d\times 2}$ with entries randomly drawn from $\mathcal{N}(0,1)$

Apply QR decomposition on Z_l to get an orthogonal Q matrix U_l

Project $\{x_i\}_{i=1}^{N_\mu}$ and $\{y_j\}_{j=1}^{N_\nu}$ on $\mathbb{R}^2$ to get $\hat{x}_i^l = U_l^T x_i$ and $\hat{y}_j^l = U_l^T y_j, \forall i, j$

Find $I_l^* = \{i^*\}, J_l^* = \{j^*\}$ such that $\|\hat{x}_{i^*}^l\| \leq \epsilon$ or $\|\hat{y}_{j^*}^l\| \leq \epsilon$

$a_l^* = \sum_{i^*\in I^*} a_{i^*}$, $b_l^* = \sum_{j^*\in J^*} b_{j^*}$; $\tilde{N}_\mu = N_\mu - |I_l^*|$, $\tilde{N}_\nu = N_\nu - |J_l^*|$

} Apply ϵ-cap around poles

$\tilde{\mu}_l = \sum_{\tilde{i}=1}^{\tilde{N}_\mu}(a_{\tilde{i}} + \frac{a^*}{\tilde{N}_\mu})\delta_{\hat{x}_{\tilde{i}}^l}$, $\tilde{\nu}_l = \sum_{\tilde{j}=1}^{\tilde{N}_\nu}(b_{\tilde{j}} + \frac{b^*}{\tilde{N}_\nu})\delta_{\hat{y}_{\tilde{j}}^l}$

Project all $\hat{x}_{\tilde{i}}^l$ and $\hat{y}_{\tilde{j}}^l$ on $\mathbb{S}^1$: $P(\hat{x}_{\tilde{i}}^l) = \frac{\hat{x}_{\tilde{i}}^l}{\|\hat{x}_{\tilde{i}}^l\|}$, $P(\hat{y}_{\tilde{j}}^l) = \frac{\hat{y}_{\tilde{j}}^l}{\|\hat{y}_{\tilde{j}}^l\|}$

Calculate $LCOT_{\bar{\mu},2}(\tilde{\mu}_l^{proj}, \tilde{\mu}_l^{proj})$ by (3), where $\tilde{\mu}_l^{proj} = \sum_{\tilde{i}=1}^{\tilde{N}_\mu}(a_{\tilde{i}} + \frac{a^*}{\tilde{N}_\mu})\delta_{P(\hat{x}_{\tilde{i}}^l)}$, $\tilde{\nu}_l^{proj} = \sum_{\tilde{j}=1}^{\tilde{N}_\nu}(b_{\tilde{j}} + \frac{b^*}{\tilde{N}_\nu})\delta_{P(\hat{y}_{\tilde{j}}^l)}$, $\bar{\mu}$ is the discrete uniform reference measure of size M

end for

return $LSSOT_2(\mu,\nu) \approx \left(\frac{1}{L}\sum_{l=1}^{L} LCOT_2^2(\tilde{\mu}_l^{proj}, \tilde{\mu}_l^{proj})\right)^{\frac{1}{2}}$

normalize the orthogonal matrix Q (dividing each column vector of Q by its Euclidean norm). In contrast to the Spherical Sliced Wasserstein (SSW) numerical implementation in which projections of poles are not well defined (when $U^T x = 0$ in 7), LSSOT maps the polar points everywhere on the equator by definition. To achieve this mapping numerically, we put an ϵ-cap around the poles, and redistribute all the mass inside the ϵ-cap evenly onto the discrete points outside the cap. See Figure 1 for a visual depiction of the projection and Algorithm 1 for detailed implementation steps.

3.3 Time Complexity and Runtime Analysis

Let L denote the number of slices and d denote the dimension. For a discrete spherical distribution ν with N samples, the slicing and projection require $\mathcal{O}(LdN)$. For a single slice, the time complexity of calculating the LCOT embedding (Equation 5) of the corresponding circular distributions with respect to the uniform distribution is $\mathcal{O}(N)$ after sorting. Therefore the time complexity of calculating LSSOT embedding of ν is $\mathcal{O}(LN(d + \log N + 1))$, where $LN \log N$ comes from the sorting $\mathbb{S}^1$.

For a set of K distributions $\{\nu_k\}_{k=1}^K$, each with N samples, the pairwise LSSOT distances can be boiled down to K LSSOT embeddings and $\frac{LK(K-1)}{2}$ distances in $L^2(\mathbb{S}^1)$, thus cost $\mathcal{O}(KLN(d+\log N+1+\frac{K-1}{2}))$. However, other methods such as spherical Optimal Transport (OT), Sinkhorn divergence, Sliced Wasserstein Distance (SWD), SSW, and Stereographic Spherical Sliced Wasserstein distance (S3W) all require the time complexity $\mathcal{O}(K^2 D)$, where D is the complexity of each distance, e.g. $\mathcal{O}(K^2N^3 \log N)$ for spherical OT and $\mathcal{O}(K^2 LN \log N)$ for SSW, compared to $\mathcal{O}(K^2 LN)$ for our method. Figure 2 shows the pairwise distances computation time of LSSOT along with the methods mentioned above. All methods are run on the CPU for fair comparison. The distributions are randomly generated with sample sizes $N = 1000, 5000, 10000, 12500, 15000$. For each fixed sample size, we observe that LSSOT stands out as the fastest in all methods under the increase in the number of distributions. In the Appendix E, we also provide the runtime comparisons for slice-based methods with varying numbers of slices.

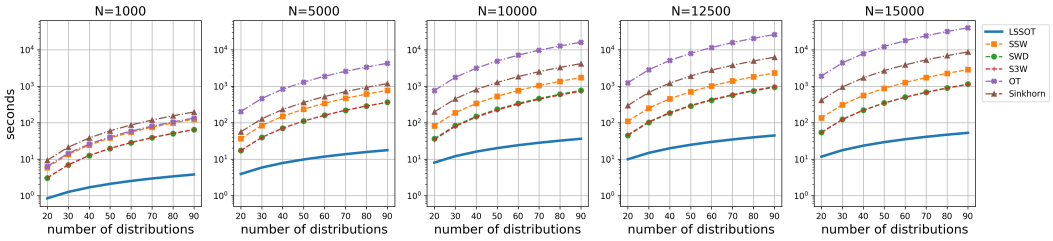

Figure 2: Pairwise distances runtime (log scale) comparison w.r.t the number of distributions. N denotes sample sizes in each distribution. The number of slices is 500 for all slice-based methods.

4 EXPERIMENTS

To demonstrate that the proposed LSSOT is an effective and efficient metric of spherical distributions while preserving the geometry, we design and implement the following experiments on toy examples, cortical surfaces, and point clouds.

4.1 ROTATION AND SCALING IN SPHERICAL GEOMETRY

We test the LSSOT metric on spherical geometric transformations such as rotation and scaling.

For rotations, we generate a set of 20 Von Mises-Fisher (VMF) distributions $\{p^i(x;\mu_i,\kappa) = \frac{\kappa}{2\pi(e^{\kappa}-e^{-\kappa})}\exp(\kappa\mu_i^T x)\}_{i=1}^{20}$ with the same concentration parameter κ, but the mean directions μ_i's are 20 distinct points on the Fibonacci Sphere, i.e. spread evenly on $\mathbb{S}^2$. Since the 20 VMFs are of the same shape with varying mean directions, each can be interpreted as a rotated version of another. We calculate the pairwise LSSOT distances among the 20 distributions and visualize the embeddings in 3-dimensional space using multi-dimensional scaling (MDS). We use the Scikit Learn (Pedregosa et al., 2011) sklearn.manifold.MDS function for MDS implementation. As shown in Figure 3 (left), the embeddings spread almost evenly on the sphere, implying the LSSOT captures the rotation geometry.

To study the scaling geometry, we generate 6 VMF distributions with fixed μ but varying κ, and then apply the same LSSOT pairwise distances calculation followed by the MDS embedding procedure. Figure 3 (right) illustrates that the LSSOT metric preserves the linear transitions in scaling geometry, with embeddings falling approximately onto a straight line.

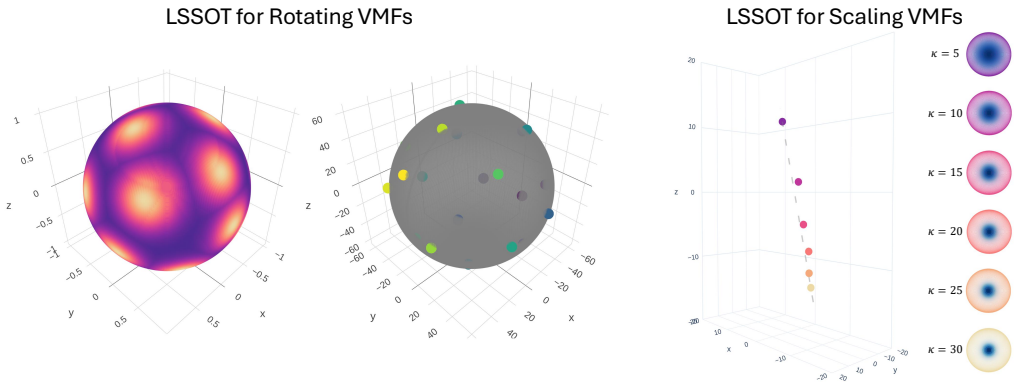

Figure 3: Generated VMF distributions and manifold learning results by LSSOT. Left: a group of 20 VMFs generated by rotating a source VMF, and the corresponding 3-dimensional visualizations by LSSOT and MDS. Right: 6 VMFs generated by scaling a source VMF with $\kappa = 30$, and the same 3-dimensional visualizations.

4.2 CORTICAL SURFACE REGISTRATION

We further verify the validity and efficiency of the proposed LSSOT method in comparing spherical cerebral cortex data for registration tasks. Cortical surface registration seeks to establish meaningful anatomical correspondences across subjects or time points. This process is essential for surface-based analysis in both cross-sectional and longitudinal neuroimaging studies. Cortical surface scans are often acquired from diverse spatial frameworks, necessitating a preliminary alignment process to situate them within a common space. This alignment is crucial for enabling meaningful group analyses and comparisons. The unit sphere $\mathbb{S}^2$ is usually chosen to be this common space, owing to the inherent genus-0 spherical topology of the cortical surfaces (Wang et al., 2004; Gu et al., 2004; Lui et al., 2007). Thus, such aligning involves mapping each hemisphere of the convoluted cerebral cortical surface onto the $\mathbb{S}^2$ space with minimal distortions. Then the goal is to register spherical cortical surfaces to a template, i.e. computing a smooth velocity field on the sphere which is used to deform the moving vertices into the fixed template. Through this deformation, correspondences are established across a group of cortical surfaces.

4.2.1 EXPERIMENTAL SETUP

We leverage the Superfast Spherical Surface Registration (S3Reg) (Zhao et al., 2021) neural network to perform atlas-based registration, which focuses on registering all surfaces to one atlas surface. S3Reg is an end-to-end unsupervised diffeomorphic registration network based on the Voxelmorph framework (Balakrishnan et al., 2018) and the Spherical Unet (Zhao et al., 2019) backbone, offering flexibilities in choosing the similarity measure for spherical surfaces in the training objectives. Following the S3Reg experimental setup with resolution level at the 6-th icosahedron subdivisions (i.e. 40962 vertices on each surface), we replace the original mean squared error (MSE) similarity loss with our LSSOT distance and other baselines, and evaluate the registration performance achieved by each similarity measure. We use the Freesurfer fsaverage (Fischl, 2012) surface as the fixed (atlas) surface. All models are implemented in PyTorch (Paszke et al., 2017) and run on a Linux server with NVIDIA RTX A6000 GPU. More experimental details are documented in Appendix F.

Dataset and Preprocessing. We perform registration on the NKI dataset (Nooner et al., 2012) and ADNI dataset (Jack Jr et al., 2008). NKI contains T1-weigted brain MRI images from 515 subjects, which are processed to obtain reconstructed cortical surfaces, spherical mappings, parcellations, and anatomical features using the FreeSurfer pipeline (Fischl, 2012). A subset of the ADNI dataset that contains 412 subjects is used and processed in the same FreeSurfer pipeline. We choose the sulcal depth as the single-channel feature for registration and normalize features into probability vectors. For parcellations, we choose the DKT Atlas (Klein & Tourville, 2012; Klein et al., 2017), which parcellates each hemisphere into 31 cortical regions of interest (ROIs). We split both datasets into 70% scans for training, 10% scans for validation, and 20% scans for testing.

Training Losses. The training loss has three components: the similarity loss for aligning the moving surfaces to the fixed atlas surface, the Dice loss to impose biological validity, and the regularization loss to enforce the smoothness of the deformation field. Specifically, we employ the parcellation maps generated by FreeSurfer and the torchmetrics.Dice function from the torchmetrics library (Nicki Skafte Detlefsen et al., 2022) for Dice loss. The training loss is a linear combination of the three components. In our experimental design, we keep the regularization and the Dice terms in all setups. Our primary focus is on investigating the impact of various similarity losses on the registration performance, including LSSOT distance and the following baselines.

Baselines. Since the original S3Reg framework used MSE as part of the similarity loss, we include MSE as one of the baselines. Besides, Spherical Sliced-Wasserstein (SSW) (Bonet et al., 2023) and Stereographic Spherical Sliced Wasserstein Distances (S3W) (Tran et al., 2024) are powerful spherical distances to test LSSOT against. We use the SSW implementation from the official repo [2] and the Amortized Rotationally Invariant Extension of S3W (AI-S3W) from the official implementation [3] with 10 rotations. We also add the Slice Wasserstein distance (SWD), which is defined in Euclidean space $\mathbb{R}^3$ instead of $\mathbb{S}^2$. The implementation of the SWD algorithm is from the Python Optimal Transport (POT) library (Flamary et al., 2021).

Metrics We quantitatively evaluate the similarities between the registered moving surfaces and the fixed atlas surface using mean absolute error (MAE) and Pearson correlation coefficient (CC) on features, as well as LSSOT, SWD on the spherical distributions (represented by vertex locations and feature probability vectors). We calculate the Dice score of the registered parcellations and the fixed parcellation. Moreover, we measure the area (resp. edge) distortions between the original moving surface meshes and the deformed surface meshes as relative changes in the areas of all faces (resp. the lengths of all edges) of meshes.

4.2.2 RESULTS

Table 1 summarizes the performance of registration evaluated by the metrics for four scenarios: left/right hemispheres of the NKI and ADNI datasets. As the two best methods overall, our proposed LSSOT metric and SSW are on par with each other, yet LSSOT costs less than 10% of the training time. We emphasize that LSSOT is the leading method in the alignment of the parcellations (i.e. Dice Score), indicating that LSSOT yields more biologically meaningful registrations. Besides, with the least area/edge distortions, LSSOT finds the simplest solution of the deformation field to align the

[2] https://github.com/clbonet/Spherical_Sliced-Wasserstein

[3] https://github.com/mint-vu/s3wd

moving surface to the fixed surface. Figure 4 shows the registered sulcal depth map and parcellation map of a random subject. More visualizations can be found in the Appendix G. It's worth noting that the LSSOT embedding $\hat{v}^S_{\text{fix}}$ of the fixed surface is calculated once and for all before training, and in each training iteration, only the LSSOT embedding of the deformed moving surface $\hat{v}^S_{\text{mov}}$ is calculated and then compared to $\hat{v}^S_{\text{fix}}$. Other competing methods, however, need to process both surfaces and calculate the distance during each iteration, which accounts for LSSOT taking less time than SWD. This makes LSSOT a perfect fit for atlas-based registrations, as the template is fixed.

		MSE	S3W	SWD	SSW	LSSOT (Ours)
NKI (Left Hemisphere)	LSSOT(↓)	0.2754±0.0611	0.2298±0.0346	0.2411± 0.0366	**0.2079±0.0369**	**0.1890±0.0361**
	SWD(↓)	**0.0051±0.0017**	0.0053±0.0010	**0.0027±0.0011**	0.0059±0.0011	0.0052±0.0011
	MAE(↓)	**0.1129±0.0471**	0.2278±0.0372	0.2658±0.0410	**0.2145±0.0266**	0.2516±0.0397
	CC(↑)	0.8269±0.0425	**0.9216±0.0174**	**0.8722±0.0302**	0.8671±0.0314	0.8649±0.0295
	Dice (↑)	0.7746±0.0861**	**0.8498±0.0670****	0.7984±0.0539**	0.8429±0.0692*	**0.8462±0.0548**
	Edge Dist.(↓)	0.3060±0.0404**	0.3922±0.0647**	0.3442±0.0346**	**0.2476±0.0348****	**0.2365±0.0343**
	Area Dist.(↓)	0.4305±0.0488**	0.4048±0.0752**	0.4073±0.0368**	**0.2733±0.0402****	**0.2897±0.0396**
	Time(seconds)(↓)	**73.07**	121.00	118.96	1350.96	**101.01**
NKI (Right Hemisphere)	LSSOT(↓)	0.2847±0.0502	0.2062±0.0510]	0.2454±0.0396	**0.1341±0.0209**	**0.1121±0.0253**
	SWD(↓)	0.0048±0.0015	0.0046±0.0009	**0.0019±0.0003**	**0.0031±0.0017**	0.0033±0.0015
	MAE(↓)	**0.1411±0.0215**	**0.2224±0.0324**	0.2379±0.0348	0.2268±0.0551	0.2489±0.0568
	CC(↑)	**0.9334±0.0111**	0.9145±0.0130	0.9191±0.0263	0.8979±0.0383	**0.9284±0.0599**
	Dice (↑)	0.7318±0.0943**	0.6961±0.0777**	0.6972±0.0820**	**0.7952±0.0751****	**0.7996±0.0706**
	Edge Dist.(↓)	0.2872±0.0421**	**0.1395±0.0709****	0.2160±0.0406**	0.1460±0.0417**	**0.1428±0.0162**
	Area Dist.(↓)	0.3453±0.0422**	0.3125±0.0876**	0.2924±0.0412**	**0.1709±0.0126****	**0.1910±0.0189**
	Time(seconds)(↓)	**72.14**	121.02	116.91	1362.16	**100.68**
ADNI (Left Hemisphere)	LSSOT(↓)	0.1476±0.0396	0.1340±0.0397	0.1412±0.0419	**0.1262±0.0261**	**0.1041±0.0277**
	SWD(↓)	0.0046±0.0011	**0.0041±0.0011**	**0.0027±0.0009**	0.0052±0.0014	0.0063±0.0016
	MAE(↓)	**0.2784±0.0367**	0.3295±0.0427	0.3627±0.0477	**0.3153±0.0485**	0.3332±0.0873
	CC(↑)	**0.9141±0.0269**	0.9077±0.0207	**0.9139±0.0406**	0.9032±0.0139	0.9036±0.1128
	Dice (↑)	0.7859±0.0949**	0.7733±0.0816**	0.7457±0.0683**	**0.7987±0.0775****	**0.8119±0.0940**
	Edge Dist.(↓)	0.3241±0.0516**	0.2836±0.0750**	0.2379±0.0459**	**0.1927±0.0582****	**0.0979±0.0129**
	Area Dist.(↓)	0.2990±0.0580**	0.1937±0.0849**	0.1930±0.0450**	**0.1008±0.0053****	**0.1303±0.0088**
	Time(seconds)(↓)	**59.34**	100.78	96.69	1094.06	**84.51**
ADNI (Right Hemisphere)	LSSOT(↓)	0.1807±0.0347	0.1367±0.0375	0.1497±0.0509	**0.1349±0.0541**	**0.1295±0.0281**
	SWD(↓)	**0.0034±0.0010**	0.0046±0.0011	**0.0020±0.0005**	0.0041±0.0016	0.0037±0.0020
	MAE(↓)	**0.2433±0.0273**	0.3229±0.0386	**0.2618±0.0450**	0.3236±0.0519	0.3493±0.0732
	CC(↑)	**0.9330±0.0163**	0.9139±0.0204	0.8985±0.0391	**0.9375±0.0437**	0.8013±0.0851
	Dice (↑)	0.8073±0.0983**	0.7617±0.0806**	0.8122±0.0648**	**0.8622±0.0915****	**0.8518±0.0879**
	Edge Dist.(↓)	0.3371±0.0627**	0.2868±0.0835**	0.3690±0.0513**	**0.1841±0.0576****	**0.1226±0.0213**
	Area Dist.(↓)	0.3015±0.0568**	0.3109±0.0987**	0.4123±0.0438**	**0.1939±0.0392***	**0.1948±0.0325**
	Time(seconds)(↓)	**59.69**	101.35	97.52	1114.83	**85.37**

Table 1: Evaluation metrics on the test datasets (mean ± standard deviation) for sulcal depth registration of both hemispheres from NKI dataset and ADNI dataset. The best two methods are bolded. The average training time for each epoch is also included in the bottom row of each scenario. We would like to note that as the number of samples is 40,962, using spherical optimal transport or even Sinkhorn divergence is computationally intractable in both time and space. We perform significance testing for the difference on Dice, Edge Dist. and Area Dist. against LSSOT. ** denotes $p < 0.001$ and * denotes $p < 0.01$.

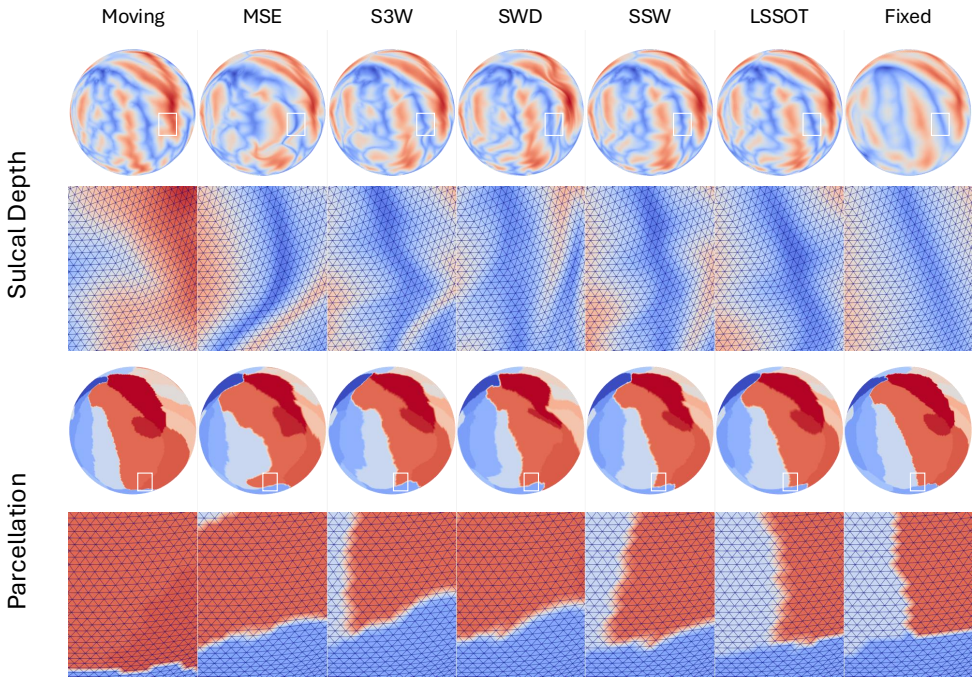

Figure 4: Qualitative registration results (middle columns) from a moving surface (left column) to the fixed surface (right column). Both sulcal depth and parcellations are visualized with global and close-up views. This moving surface is from Subject A00065749 in the NKI dataset. Visualizations for more subjects can be found in the Appendix G.

4.3 Point Cloud Interpolation

LSSOT can also be utilized in point cloud analysis, once each point cloud is endowed with a spherical representation. In this experiment, we explore the interpolations between point cloud pairs from the ModelNet dataset (Wu et al., 2015). Specifically, we train an autoencoder to project the original point clouds to a spherical latent space, to represent each of them as a spherical distribution. Then we apply gradient flow between the pairs of spherical distributions using the LSSOT metric along with SSW and Spherical OT. Finally, this transformation is reconstructed in the original space by the trained decoder, resulting in an interpolation between the original pairs. See Figure 5 (top) for an illustration of this process.

4.3.1 Mapping Point Clouds to Spherical Latent Space via LSSOT Autoencoder

As point clouds are set-structured data, we leverage the Set Transformer (Lee et al., 2019) architecture to build the autoencoder[4]. The encoder Φ_{enc} is composed of three consecutive Set Attention Blocks, followed by a linear layer and a normalization layer to output a set of points on $\mathbb{S}^2$. The decoder Ψ_{dec} consists of four Set Attention Blocks with a linear layer at the end. For a set of point clouds $\{X_i\}_{i=1}^I$ (each point cloud X_i is a set of points), the autoencoder is trained to reconstruct each X_i under a soft regularity constraint in the latent space. That is, the objective function is defined as

$$\mathcal{L} = \frac{1}{I}\sum_{i=1}^{I}\Big(\|\Psi_{\text{dec}}(\Phi_{\text{enc}}(X_i)) - X_i\|_2^2 + \lambda \cdot LSSOT_2(\Phi_{\text{enc}}(X_i), X_{\text{unif}})\Big) \tag{12}$$

where X_{unif} is a set of samples from the uniform distribution on $\mathbb{S}^2$.

After training, the encoder Φ_{enc} can serve as a spherical mapping from $\mathbb{R}^3$ to $\mathbb{S}^2$, and the decoder Ψ_{dec} acts as an inverse mapping. Enforcing the latent representation to closely approximate the uniform distribution offers significant benefits. It minimizes the occurrence of singularities in both the spherical mapping and its inverse function, resulting in fewer distortions or discontinuities, hence the structural and geometrical characteristics of the original point clouds are better preserved. In implementation, we select $\lambda = 1e-4$ through hyperparameter search. We set the number of slices to be $L = 1000$, and the reference size as $M = 1024$ for the $LSSOT_2$ function.

[4]https://github.com/juho-lee/set_transformer

4.3.2 GRADIENT FLOW AND INTERPOLATIONS OF SPHERICAL REPRESENTATIONS

Given two spherical distributions as the source and the target, we can attach gradients on the source distribution and drive it toward the target by minimizing the distance between them. For the distance, we use LSSOT, SSW, and Spherical OT for comparisons. Here the Riemannian Gradient flow algorithm (see Appendix I.3) is implemented on $\mathbb{S}^2$. With a slight abuse of notation to mimic the constant speed geodesic, we sample 5 time points denoted as $T = 0, 1/4, 2/4, 3/4, 1$, from the beginning ($T = 0$) to the loss converging ($T = 1$), such that the loss goes down by approximately the same amount in each interval. In Figure 5 (bottom), we visualize the interpolations for two example pairs of point clouds.

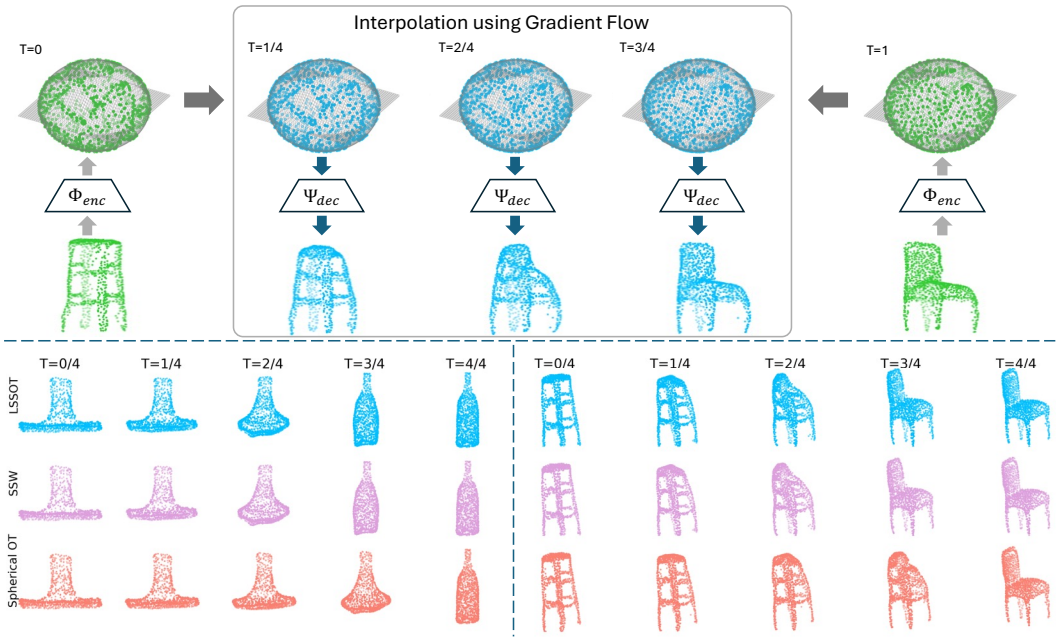

Figure 5: Top panel: the interpolation process between a pair of point clouds using gradient flow in the latent space $\mathbb{S}^2$. Bottom panel: gradient flow interpolations from a range hood to a bottle (left), and from a stool to a chair (right) using three metrics LSSOT, SSW and spherical OT. See Appendix I.4 for more interpolated pairs.

5 LIMITATIONS AND FUTURE WORK

To the best of our knowledge, this work is the first to adapt the Linear Optimal Transport methodology onto hyperspheres, enabling rapid comparisons among a group of spherical distributions. However, spherical data/signals may not come in the form of distributions in practice, and normalizing data into probability distributions may result in the loss of information, thus one limitation of our work is that LSSOT is defined only for probability measures. Immediate future work includes a partial extension of LSSOT for positive signals with unequal mass, and transport L^p Thorpe et al. (2017); Liu et al.; Crook et al. (2024) extension for non-positive spherical signals, both of which will be studied on $\mathbb{S}^1$ as the first step, and extended to $\mathbb{S}^2$ by slicing.

6 CONCLUSION

In this work, we propose Linear Spherical Sliced Optimal Transport (LSSOT) as a novel and fast metric for defining the similarities between spherical data. We provide rigorous mathematical proofs establishing LSSOT as a true metric. We also demonstrate the computation efficiency of the LSSOT, superior to other baselines in the setting of pairwise comparisons of spherical data. We verify that LSSOT captures the rotation and scaling geometry on spheres through experiments on the Von Mises-Fisher distributions. In the task of atlas-based cortical surface registration on various datasets, LSSOT delivers top registration performance across multiple evaluation metrics while reducing the training time significantly. Furthermore, LSSOT is shown to yield meaningful interpolations between point cloud pairs. The metric's effectiveness in both point cloud analysis and practical neuroimaging applications demonstrates its broad potential in 3D data analysis.

REFERENCES

William Abikoff. The uniformization theorem. *American Mathematical Monthly*, 88:574–592, 1981. URL https://api.semanticscholar.org/CorpusID:124716245.

P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2008.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223. PMLR, 2017.

Guha Balakrishnan, Amy Zhao, Mert Rory Sabuncu, John V. Guttag, and Adrian V. Dalca. Voxelmorph: A learning framework for deformable medical image registration. *IEEE Transactions on Medical Imaging*, 38:1788–1800, 2018. URL https://api.semanticscholar.org/CorpusID:52281312.

Saurav Basu, Soheil Kolouri, and Gustavo K Rohde. Detecting and visualizing cell phenotype differences from microscopy images using transport-based morphometry. *Proceedings of the National Academy of Sciences*, 111(9):3448–3453, 2014.

Thomas Bendokat, Ralf Zimmermann, and P-A Absil. A grassmann manifold handbook: Basic geometry and computational aspects. *arXiv preprint arXiv:2011.13699*, 2020.

Clément Bonet, Paul Berg, Nicolas Courty, François Septier, Lucas Drumetz, and Minh-Tan Pham. Spherical Sliced-Wasserstein. *ICLR*, 2023.

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.

Nicolas Bonnotte. *Unidimensional and evolution methods for optimal transportation*. PhD thesis, Université Paris Sud-Paris XI; Scuola normale superiore (Pise, Italie), 2013.

Nicolas Boumal, Pierre-Antoine Absil, and Coralia Cartis. Global rates of convergence for nonconvex optimization on manifolds. *IMA Journal of Numerical Analysis*, 39(1):1–33, 2019.

Zixuan Cang, Yanxiang Zhao, Axel A Almet, Adam Stabell, Raul Ramos, Maksim V Plikus, Scott X Atwood, and Qing Nie. Screening cell–cell communication in spatial transcriptomics via collective optimal transport. *Nature methods*, 20(2):218–228, 2023.

Moo K Chung and Zijian Chen. Embedding of functional human brain networks on a sphere. *arXiv preprint arXiv:2204.03653*, 2022.

Oliver M Crook, Mihai Cucuringu, Tim Hurst, Carola-Bibiane Schönlieb, Matthew Thorpe, and Konstantinos C Zygalakis. A linear transportation lp distance for pattern recognition. *Pattern Recognition*, 147:110080, 2024.

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.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

Julie Delon, Julien Salomon, and Andrei Sobolevski. Fast transport optimization for Monge costs on the circle. *SIAM Journal on Applied Mathematics*, 70(7):2239–2258, 2010.

Ishan Deshpande, Ziyu Zhang, and Alexander G Schwing. Generative modeling using the sliced wasserstein distance. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3483–3491, 2018.

A. Elad and R. Kimmel. *Spherical Flattening of the Cortex Surface*, pp. 77–89. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002.

Albert Fathi and Alessio Figalli. Optimal transportation on non-compact manifolds. *Israel Journal of Mathematics*, 175:1–59, 2010.

Bruce Fischl. Freesurfer. *Neuroimage*, 62(2):774–781, 2012.

Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z. Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, Léo Gautheron, Nathalie T.H. Gayraud, Hicham Janati, Alain Rakotomamonjy, Ievgen Redko, Antoine Rolet, Antony Schutz, Vivien Seguy, Danica J. Sutherland, Romain Tavenard, Alexander Tong, and Titouan Vayer. Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021. URL http://jmlr.org/papers/v22/20-451.html.

Robert C Garrett, Trevor Harris, Bo Li, and Zhuo Wang. Validating climate models with spherical convolutional wasserstein distance. *arXiv preprint arXiv:2401.14657*, 2024.

Daniela Giorgi, Silvia Biasotti, and Laura Paraboschi. Shape retrieval contest 2007: Watertight models track. 2007. URL https://api.semanticscholar.org/CorpusID:14645500.

Craig Gotsman, Xianfeng Gu, and Alla Sheffer. Fundamentals of spherical parameterization for 3d meshes. In *ACM SIGGRAPH 2003 Papers*, pp. 358–363. 2003.

Manuel Gräf and Daniel Potts. Sampling sets and quadrature formulae on the rotation group. *Numerical Functional Analysis and Optimization*, 30(7-8):665–688, 2009.

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

Xianfeng Gu, Yalin Wang, Tony F Chan, Paul M Thompson, and Shing-Tung Yau. Genus zero surface conformal mapping and its application to brain surface mapping. *IEEE transactions on medical imaging*, 23(8):949–958, 2004.

Brittany Froese Hamfeldt and Axel GR Turnquist. A convergence framework for optimal transport on the sphere. *Numerische Mathematik*, 151(3):627–657, 2022.

Ralf Hielscher, Daniel Potts, and Michael Quellmalz. An svd in spherical surface wave tomography. *New Trends in Parameter Identification for Mathematical Models*, pp. 121–144, 2018.

Clifford R Jack Jr, Matt A Bernstein, Nick C Fox, Paul Thompson, Gene Alexander, Danielle Harvey, Bret Borowski, Paula J Britson, Jennifer L. Whitwell, Chadwick Ward, et al. The alzheimer's disease neuroimaging initiative (adni): Mri methods. *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine*, 27(4):685–691, 2008.

N Jarosik, CL Bennett, J Dunkley, B Gold, MR Greason, M Halpern, RS Hill, G Hinshaw, A Kogut, Eiichiro Komatsu, et al. Seven-year wilkinson microwave anisotropy probe (wmap*) observations: Sky maps, systematic errors, and basic results. *The Astrophysical Journal Supplement Series*, 192 (2):14, 2011.

Arno Klein and Jason Tourville. 101 labeled brain images and a consistent human cortical labeling protocol. *Frontiers in neuroscience*, 6:171, 2012.

Arno Klein, Satrajit S Ghosh, Forrest S Bao, Joachim Giard, Yrjö Häme, Eliezer Stavsky, Noah Lee, Brian Rossa, Martin Reuter, Elias Chaibub Neto, et al. Mindboggling morphometry of human brains. *PLoS computational biology*, 13(2):e1005350, 2017.

Max Kochurov, Rasul Karimov, and Serge Kozlukov. Geoopt: Riemannian optimization in pytorch, 2020.

Nicholas Kolkin, Jason Salavon, and Gregory Shakhnarovich. Style transfer by relaxed optimal transport and self-similarity. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10051–10060, 2019.

Soheil Kolouri, Yang Zou, and Gustavo K Rohde. Sliced wasserstein kernels for probability distributions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5258–5267, 2016.

Soheil Kolouri, Se Rim Park, Matthew Thorpe, Dejan Slepcev, and Gustavo K Rohde. Optimal mass transport: Signal processing and machine-learning applications. *IEEE signal processing magazine*, 34(4):43–59, 2017.

Soheil Kolouri, Gustavo K Rohde, and Heiko Hoffmann. Sliced wasserstein distance for learning gaussian mixture models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3427–3436, 2018.

Soheil Kolouri, Kimia Nadjahi, Umut Simsekli, Roland Badeau, and Gustavo Rohde. Generalized sliced Wasserstein distances. *Advances in neural information processing systems*, 32, 2019.

Soheil Kolouri, Navid Naderializadeh, Gustavo K Rohde, and Heiko Hoffmann. Wasserstein embedding for graph learning. In *International Conference on Learning Representations*, 2020.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International conference on machine learning*, pp. 3744–3753. PMLR, 2019.

Xinran Liu, Yikun Bai, Huy Tran, Zhanqi Zhu, Matthew Thorpe, and Soheil Kolouri. Ptlp: Partial transport L^p distances. In *NeurIPS 2023 Workshop Optimal Transport and Machine Learning*.

Xinran Liu, Yikun Bai, Yuzhe Lu, Andrea Soltoggio, and Soheil Kolouri. Wasserstein task embedding for measuring task similarities. *arXiv preprint arXiv:2208.11726*, 2022.

Yuzhe Lu, Xinran Liu, Andrea Soltoggio, and Soheil Kolouri. Slosh: Set locality sensitive hashing via sliced-wasserstein embeddings. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2566–2576, 2024.

Lok Ming Lui, Yalin Wang, Tony F Chan, and Paul Thompson. Landmark constrained genus zero surface conformal mapping and its application to brain mapping research. *Applied Numerical Mathematics*, 57(5-7):847–858, 2007.

Ashok Makkuva, Amirhossein Taghvaei, Sewoong Oh, and Jason Lee. Optimal transport mapping via input convex neural networks. In *International Conference on Machine Learning*, pp. 6672–6681. PMLR, 2020.

Daniel S Marcus, Michael P Harms, Abraham Z Snyder, Mark Jenkinson, J Anthony Wilson, Matthew F Glasser, Deanna M Barch, Kevin A Archie, Gregory C Burgess, Mohana Ramaratnam, et al. Human connectome project informatics: quality control, database services, and data visualization. *Neuroimage*, 80:202–219, 2013.

Rocio Diaz Martin, Ivan Medri, Yikun Bai, Xinran Liu, Kangbai Yan, Gustavo K Rohde, and Soheil Kolouri. Lcot: Linear circular optimal transport. *arXiv preprint arXiv:2310.06002*, 2023.

Robert J McCann and Peter M Topping. Ricci flow, entropy and optimal transportation. *American Journal of Mathematics*, 132(3):711–730, 2010.

Nicki Skafte Detlefsen, Jiri Borovec, Justus Schock, Ananya Harsh, Teddy Koker, Luca Di Liello, Daniel Stancl, Changsheng Quan, Maxim Grechkin, and William Falcon. TorchMetrics - Measuring Reproducibility in PyTorch, February 2022. URL `https://github.com/Lightning-AI/torchmetrics`.

Kate Brody Nooner, Stanley J Colcombe, Russell H Tobe, Maarten Mennes, Melissa M Benedict, Alexis L Moreno, Laura J Panek, Shaquanna Brown, Stephen T Zavitz, Qingyang Li, et al. The nki-rockland sample: a model for accelerating the pace of discovery science in psychiatry. *Frontiers in neuroscience*, 6:152, 2012.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

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.

Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

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

Julien Rabin, Julie Delon, and Yann Gousseau. Transportation distances on the circle. *Journal of Mathematical Imaging and Vision*, 41:147—-167, 2011.

Boris Rubin. Inversion and characterization of the hemispherical transform. *Journal d'Analyse Mathématique*, 77:105–128, 1999.

Vivien Seguy, Bharath Bhushan Damodaran, Rémi Flamary, Nicolas Courty, Antoine Rolet, and Mathieu Blondel. Large-scale optimal transport and mapping estimation. *arXiv preprint arXiv:1711.02283*, 2017.

Li Shen and Fillia Makedon. Spherical mapping for processing of 3d closed surfaces. *Image and vision computing*, 24(7):743–761, 2006.

Matthew Thorpe, Serim Park, Soheil Kolouri, Gustavo K Rohde, and Dejan Slepčev. A transportation l^ p l p distance for signal analysis. *Journal of mathematical imaging and vision*, 59:187–210, 2017.

Huy Tran, Yikun Bai, Abihith Kothapalli, Ashkan Shahbazi, Xinran Liu, Rocio Diaz Martin, and Soheil Kolouri. Stereographic spherical sliced wasserstein distances. *arXiv preprint arXiv:2402.02345*, 2024.

Wei Wang, Dejan Slepčev, Saurav Basu, John A. Ozolek, and Gustavo Kunde Rohde. A linear optimal transportation framework for quantifying and visualizing variations in sets of images. *International Journal of Computer Vision*, 101:254–269, 2012. URL `https://api.semanticscholar.org/CorpusID:5982072`.

Wei Wang, Dejan Slepčev, Saurav Basu, John A Ozolek, and Gustavo K Rohde. A linear optimal transportation framework for quantifying and visualizing variations in sets of images. *International journal of computer vision*, 101:254–269, 2013.

Yalin Wang, Xianfeng Gu, Tony F Chan, Paul M Thompson, and Shing-Tung Yau. Intrinsic brain surface conformal mapping using a variational method. In *Medical Imaging 2004: Image Processing*, volume 5370, pp. 241–252. SPIE, 2004.

Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

Fenqiang Zhao, Shun ren Xia, Zhengwang Wu, Dingna Duan, Li Wang, Weili Lin, John H. Gilmore, Dinggang Shen, and Gang Li. Spherical u-net on cortical surfaces: Methods and applications. *Information processing in medical imaging : proceedings of the ... conference*, 11492:855–866, 2019. URL `https://api.semanticscholar.org/CorpusID:90262007`.

Fenqiang Zhao, Zhengwang Wu, Fan Wang, Weili Lin, Shunren Xia, Dinggang Shen, Li Wang, and Gang Li. S3reg: Superfast spherical surface registration based on deep learning. *IEEE Transactions on Medical Imaging*, 40(8):1964–1976, 2021. doi: 10.1109/TMI.2021.3069645.

Fenqiang Zhao, Zhengwang Wu, Li Wang, Weili Lin, and Gang Li. Fast spherical mapping of cortical surface meshes using deep unsupervised learning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 163–173. Springer, 2022.

Fenqiang Zhao, Zhengwang Wu, and Gang Li. Deep learning in cortical surface-based neuroimage analysis: a systematic review. *Intelligent Medicine*, 3(1):46–58, 2023. ISSN 2667-1026.

A SPHERICAL RADON TRANSFORM IN $\mathbb{S}^2$

In Bonet et al. (2023), a **Spherical Radon transform** is introduced and it is used to give an equivalent formula of the SSW discrepancy (8). In this section, we will discuss such Spherical Radon transform and its relation with the **Semicircle Transform**.

A.1 NOTATIONS ON SPHERE $\mathbb{S}^2$

Let $U \in V_2(\mathbb{R}^3)$, where

$$V_2(\mathbb{R}^3) = \{U \in \mathbb{R}^{3\times 2} : U^T U = I_2\}.$$

Each U determines a 2-dimensional plane spanned by the columns of U:

$$H(U) := \text{span}\{U[:,1], U[:,2]\} \subset \mathbb{R}^3.$$

The set $\{U[:,1], U[:,2]\}$ is an orthonormal basis for $H(U)$. The subspace $H(U)$ can be equivalently defined by a normal unit vector n_U, which is the uniquely determined by U (up to scalar ± 1):

$$n_U \in \mathbb{S}^2, \qquad U^T n_U = 0_2. \tag{13}$$

Thus, the plane $H(U)$ can be defined as

$$H(U) = n_U^{\perp} := \{x \in \mathbb{R}^3 : x^T n_U = 0\}.$$

Note that $\mathcal{S}(U) := H(U) \cap \mathbb{S}^2$ defines a great circle.

A.2 PROJECTION ONTO THE GREAT CIRCLE $\mathcal{S}(U)$

Given $U \in V_2(\mathbb{R}^3)$, the operator $UU^T : \mathbb{R}^3 \to \mathbb{R}^3$ defines a projection. Indeed, for each point $s \in \mathbb{S}^2 \setminus \{\pm n_U\}$, the mapping

$$s \mapsto \mathcal{P}^1_{\mathcal{S}(U)}(s) := \frac{UU^T s}{\|UU^T s\|} = \frac{UU^T s}{\|U^T s\|} = \frac{U[:,1]^T s}{\|U^T s\|} U[:,1] + \frac{U[:,2]^T s}{\|U^T s\|} U[:,2] \in \mathcal{S}(U) \subset \mathbb{R}^3. \tag{14}$$

defines a projection onto the great circle $\mathcal{S}(U)$, where in (14) we used that

$$\left\|UU^T s\right\| = s^T UU^T UU^T s = s^T UU^T s = \left\|U^T s\right\|$$

(see (Bonet et al., 2023, Lemma 1)). Note that in Bonet et al. (2023), the above projection is written as

$$s \mapsto P^U s := \frac{U^T s}{\|U^T s\|}, \tag{15}$$

which is a map from $\mathbb{S}^2 \setminus \{\pm n_U\}$ to $\mathbb{R}^2$ that saves the coordinates $[U[:,1]^T s, U[:,2]^T s]$. This function maps s into a 1-dimensional circle $\approx \mathbb{S}^1$, which can be regarded as a representation/parametrization of the great circle $\mathcal{S}(U)$.

An equivalent method that allows us to define such a projection is introduced by the following parameterization system Quellmalz et al. (2023):

Each $s \in \mathbb{S}^2$ can be written in spherical coordinates:

$$s = [\cos\alpha \sin\beta, \sin\alpha \sin\beta, \cos\beta]^T.$$

where $\alpha \in [0, 2\pi)$ is called the azimuth angle of s, and $\beta \in [0, \pi]$ the zenith angle of s. We will use the notation $\alpha = azi(s)$, $\beta = zen(s)$.

Remark A.1. *When $s = \pm[0,0,1]$, the above parametrization is not unique. As in Quellmalz et al. (2023), we set $\alpha = 0, \beta = 0$ for $s = [0,0,1]$ and $\alpha = 0, \beta = \pi$ for $s = [0,0,-1]$. Thus,*

$$s \mapsto (\alpha, \beta)$$

becomes a bijection (see (Quellmalz et al., 2023, page 6)).

Let us define the following mappings:

$$[0, 2\pi) \times [0, \pi] \ni (\alpha, \beta) \mapsto \Phi(\alpha, \beta) := [\cos\alpha \sin\beta, \sin\alpha \sin\beta, \cos\beta]^T =: s \in \mathbb{S}^2 \tag{16}$$

$$\mathbb{S}^2 \ni s \mapsto azi(s) =: \alpha \in [0, 2\pi) \tag{17}$$

$$\mathbb{S}^2 \ni s \mapsto zen(s) =: \beta \in [0, \pi] \tag{18}$$

In addition, we define the following matrix:

$$\Psi(\alpha, \beta, \gamma) := \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

By (Quellmalz et al., 2023, page 7) and Gräf & Potts (2009),

$$\mathbb{S}^2 \times [0, 2\pi) \ni (s, \gamma) \mapsto \Psi(azi(s), zen(s), \gamma) \in SO(3) \tag{19}$$

is a bijection, where $SO(3)$ is the special orthogonal group in $\mathbb{R}^3$:

$$SO(3) = \{G \in \mathbb{R}^{3\times 3} : GG^T = G^T G = I_3, \det(G) = 1\}.$$

In the special case $\gamma = 0$, we adopt the following notation:

$$\Psi(s) := \Psi(\alpha, \beta, 0) = \begin{bmatrix} \cos\alpha\cos\beta & -\sin\alpha & \cos\alpha\sin\beta \\ \sin\alpha\cos\beta & \cos\alpha & \sin\alpha\sin\beta \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$$

where, given $s \in \mathbb{S}^2$, α and β are as in (17) and (18) (or equivalently, given (α, β), $s \in \mathbb{S}^2$ is defined by (16)).

It is straightforward to verify

$$\Psi(s)^T s = \mathfrak{n} := [0, 0, 1]^T.$$

That is, we can regard $\Psi(s)^T$ as a rotation matrix that can rotate s back to the North Pole $\mathfrak{n} = [0, 0, 1]^T$. Equivalently, $\Psi(s)$ is a rotation matrix, which can rotate $[0, 0, 1]^T$ to s.

Define the Equator by

$$\mathcal{E} := \{[\cos\gamma, \sin\gamma, 0] : \gamma \in [0, 2\pi)\}.$$

Given $U \in V_2(\mathbb{R}^3)$, consider the following set

$$\{s \in \mathbb{S}^2 : \Psi(n_U)^T s \in \mathcal{E}\}$$

to obtain the following characterization of great circles for $\mathbb{S}^2 \subset \mathbb{R}^3$:

Lemma A.2. *Given $U \in V_2(\mathbb{R}^3)$, it defines a great circle $\mathcal{S}(U) = H(U) \cap \mathbb{S}^2$ which is characterized by the following identity:*

$$\mathcal{S}(U) = \{s \in \mathbb{S}^2 : \Psi(n_U)^T s \in \mathcal{E}\}. \tag{20}$$

Proof. First, since $\Psi(n_U)$ is a rotation matrix, $\Psi(n_U)^T s \in \mathbb{S}^2$ for all $s \in \mathbb{S}^2$, as $\|\Psi(n_U)^T s\| = \|s\| = 1$.

Now, write $n_U = [\cos\alpha^* \sin\beta^*, \sin\alpha^* \sin\beta^*, \cos\beta^*]^T$, where $\alpha^* := azi(n_U)$ and $\beta^* = zen(n_U)$.

For any $s \in \mathcal{S}(U)$, we have $(n_U)^T s = 0$.

We have to prove that the third coordinate of $\Psi(n_U)^T s \in \mathbb{S}^2$ is 0:

$$[\Psi(n_U)^T s][3] = [\cos\alpha^* \sin\beta^*, \sin\alpha^* \sin\beta^*, \cos\beta^*]s = (n_U)^T s = 0.$$

That is, for every $s \in \mathcal{S}(U)$, we have obtained that $s \in \{s \in \mathbb{S}^2 : \Psi(n_U)^T s \in \mathcal{E}\}$.

Similarly, if we consider $s \in \{s \in \mathbb{S}^2 : \Psi(n_U)^T s \in \mathcal{E}\}$, we have $0 = [\Psi(n_U)^T s][3] = (n_U)^T s$. Thus, $s \in \mathcal{S}(U)$.

Therefore, we have the equality in (20). □

Lemma A.3. *The mapping*

$$[0, 2\pi) \ni \gamma \mapsto \Phi(\gamma, \pi/2) = [\cos\gamma, \sin\gamma, 0]^T \in \mathcal{E}$$

and the mapping

$$\mathcal{E} \ni s \mapsto \Psi(n_U)s \in \mathcal{S}(U)$$

are bijections. Thus, the mapping

$$[0, 2\pi) \ni \gamma \mapsto s := \Psi(n_U)\Phi(\gamma, \pi/2) \in \mathcal{S}(U)$$

is also a bijection.

Proof. It is trivial to verify that the first mapping, $\gamma \mapsto [\cos\gamma, \sin\gamma, 0]^T$, is a bijection.

By the Lemma A.2, the mapping $s \mapsto \Psi(n_U)^T s$ is well-defined from $\mathcal{S}(U)$ to $\mathcal{E}$. Since $\Psi(n_U)^T$ is a rotation matrix, it is bijection $\mathbb{S}^2 \to \mathbb{S}^2$ whose inverse is $\Psi(n_U)$, and so the mapping $s \mapsto \Psi(n_U)s$ is a well-defined bijection from $\mathcal{E}$ to $\mathcal{S}(U)$: that is, if we restrict $\Psi(n_U)$ to $\mathcal{E}$, by Lemma A.2, the matrix $\Psi(n_U)$ defines a bijection from $\mathcal{E}$ to Range$(\Psi(n_U)\mathcal{E}) = \mathcal{S}(U)$. □

Now, we define the following composition map $\mathcal{P}^2_{\mathcal{S}(U)}(s)$:

$$\mathbb{S}^2 \setminus \{\pm n_U\} \ni s \mapsto \gamma := azi(\Psi(n_U)^T s) \mapsto \mathcal{P}^2_{\mathcal{S}(U)}(s) := \Psi(n_U)\Phi(\gamma, \pi/2) \in \mathcal{S}(U) \tag{21}$$

Proposition A.4. *The mappings $\mathcal{P}^1_{\mathcal{S}(U)}$ (defined by 14) and $\mathcal{P}^2_{\mathcal{S}(U)}$ (defined by 21) coincide on $\mathbb{S}^2 \setminus \{\pm n_U\}$.*

Proof. Case 1: Suppose $n_U = [0, 0, 1]^T$. Consider $s = [\cos\alpha \sin\beta, \sin\alpha \sin\beta, \cos\beta]^T \in \mathbb{S}^2 \setminus \{\pm n_U\}$, i.e., where $\alpha = azi(s)$ and $\beta = zen(s)$ with $\beta \in (0, \pi)$. In this case, from (13), $U[3, :] = 0_2$. Thus, $\{U[1 : 2, 1], U[1 : 2, 2]\}$ forms an orthonomal basis for $\mathbb{R}^2$. From (14), we have

$$UU^T s = \begin{bmatrix} I_2 & 0 \\ 0 & 0 \end{bmatrix} s = [\cos\alpha \sin\beta, \sin\alpha \sin\beta, 0]^T$$

Thus,

$$\mathcal{P}^1_{\mathcal{S}(U)}(s) = \frac{UU^T s}{\|U^T s\|} = [\cos\alpha, \sin\alpha, 0]^T.$$

In addition, in this case, $\Psi(n_U) = I_3$. Thus, $\gamma := azi(\Psi(n_U)^T s) = azi(s) = \alpha$. Therefore,

$$\mathcal{P}^2_{\mathcal{S}(U)}(s) = \Psi(n_U)\Phi(\gamma, \pi/2) = \Phi(\gamma, 0) = [\cos\alpha, \sin\alpha, \pi/2]^T.$$

Thus, 14 and 21 coincide in this case.

Case 2: Suppose $n_U \neq [0, 0, 1]$. Each point $s \in \mathbb{S}^2$ can be written as $s = \Psi(n_U)s'$ where $s' = \Psi(n_U)^T s$. In addition, when $s \neq \pm n_U$, we can parametrize s' via $s' = [\cos\alpha \sin\beta, \sin\alpha \sin\beta, \cos\beta]^T$, where $\beta \in (0, \pi)$. We also have that $\Psi(n_U)^T n_U = [0, 0, 1]^T$.

Let $U' = \Psi(n_U)^T U$. We have

$$(U')^T[0, 0, 1]^T = U^T\Psi(n_U)[0, 0, 1]^T = U^T n_U = 0_2.$$

Thus $n_{U'} = [0, 0, 1]^T$. Besides, we have that

$$UU^T s = \Psi(n_U)U'(U')^T\Psi(n_U)^T\Psi(n_U)s' = \Psi(n_U)U'(U')^T s'.$$

Thus,

$$\mathcal{P}^1_{\mathcal{S}(U)}(s) = \Psi(n_U)\mathcal{P}^1_{\mathcal{S}(U')}(s'), \qquad \forall s \neq \pm n_U.$$

In addition, by using the parametrization of s' in spherical coordinates, we have

$$\gamma := azi(\Psi(n_U)^T s) = azi(s') = \alpha.$$

Thus,

$$\mathcal{P}^2_{\mathcal{S}(U)}(s) = \Psi(n_U)[\cos\alpha, \sin\alpha, 0]^T = \Psi(n_U)\mathcal{P}^2_{\mathcal{S}(U')}(s').$$

From Case 1, we have

$$\mathcal{P}^1_{\mathcal{S}(U')}(s') = \mathcal{P}^2_{\mathcal{S}(U')}(s'),$$

and from the the fact that $s' \mapsto s = \Psi(n_U)s'$ is a bijection on $\mathbb{S}^2$, we get

$$\mathcal{P}^1_{\mathcal{S}(U)}(s) = \mathcal{P}^2_{\mathcal{S}(U)}(s).$$

□

Lemma A.5. *Given $z \in \mathbb{S}^1, U \in V_2(\mathbb{R}^3)$, there exists $\alpha^* \in [0, 2\pi), \beta^* \in [0, \pi], \gamma \in [0, 2\pi)$ such that*

$$n_U = \Phi(\alpha^*, \beta^*) = [\cos\alpha^* \sin\beta^*, \sin\alpha^* \sin\beta^*, \cos\beta^*]^T \tag{22}$$

$$Uz = \Psi(n_U)[\cos\gamma, \sin\gamma, 0]^T \tag{23}$$

Vice-versa: Given $\alpha^* \in [0, 2\pi), \beta^* \in [0, \pi], \gamma \in [0, 2\pi)$, there exist $z \in \mathbb{S}^2, U \in V_2(\mathbb{R}^3)$ for which formulas 22 and 23 hold.

Proof. Given $U \in V_2(\mathbb{R}^3)$, one can find $n_U \in \mathbb{S}^2$ via (13). (Note that if n_U satisfies (13), then $-n_U$ also satisfies (13), and so n_U is unique up to sign.) Then, once n_U is chosen, we can uniquely find (α^*, β^*) as (17), (18).

In addition, for $z \in \mathbb{S}^1 \subset \mathbb{R}^2$, since

$$\mathcal{P}^1_{\mathcal{S}(U)}(Uz) = \frac{UU^TUz}{\|U^TUz\|} = \frac{Uz}{\|z\|} = Uz,$$

we obtain that $Uz \in \mathcal{S}(U)$. By Lemma A.3, $\Psi(n_U)^TUz \in \mathcal{E}$. Then, we can uniquely determine γ (see Lemma A.3). Thus, (22), and (23) are satisfied.

For the other direction, consider angles $\alpha^*, \beta^*, \gamma$ and let us separate the analysis into the following cases:

Case 1: $\alpha^* = 0, \beta^* \in \{0, \pi\}$. In this case, we have $n_U = [0, 0, \pm 1]$ and thus we can set

$$U := \begin{bmatrix} I_2 \\ 0 \end{bmatrix}.$$

In addition, set $z := [\cos\gamma, \sin\gamma]^T \in \mathbb{S}^1$ and so the identities 22, and 23 are satisfied.

Case 2: If $\alpha^* \neq 0$, then we set n_U by 22. Choose an orthornormal basis $\{u_1, u_2\}$ for $n_U^\perp = \{x \in \mathbb{R}^3 : x^T n_U = 0\}$, and set $U := [u_1, u_2]$. Then (U, n_U) satisfies 13.

By Lemma A.3, we have $\Psi(n_U)[\cos\gamma, \sin\gamma, 0]^T \in \mathcal{S}(U)$. Then,

$$UU^T\Psi(n_U)[\cos\gamma, \sin\gamma, 0]^T \in \mathcal{S}(U),$$

and by setting

$$z := U^T\Psi(n_U)[\cos\gamma, \sin\gamma, 0]^T,$$

we have

$$\begin{aligned} Uz = UU^T\Psi(n_U)[\cos\gamma, \sin\gamma, 0]^T &= \frac{UU^T\Psi(n_U)[\cos\gamma, \sin\gamma, 0]^T}{\|U^T\Psi(n_U)[\cos\gamma, \sin\gamma, 0]^T\|} \\ &= \mathcal{P}^1_{\mathcal{S}(U)}(\Psi(n_U)[\cos\gamma, \sin\gamma, 0]^T) = \Psi(n_U)[\cos\gamma, \sin\gamma, 0]^T \end{aligned}$$

Thus, 23 is satisfied. □

Now, given $\gamma \in [0, 2\pi)$, and $n_U = \Phi(\alpha^*, \beta^*) \in \mathbb{S}^2$, we can define the following **semicircle**:

$$\mathcal{SC}^\gamma_{n_U} := \mathcal{SC}^\gamma_{\alpha^*, \beta^*} := \{s \in \mathbb{S}^2 : azi(\Psi(n_U)^T s) = \gamma\}. \tag{24}$$

Remark A.6. *When $\gamma \neq 0$, we can replace the above definition by*

$$\mathcal{SC}^\gamma_{n_U} := \mathcal{SC}^\gamma_{\alpha^*, \beta^*} := \{s \in \mathbb{S}^2 \setminus \{\pm n_U\} : azi(\Psi(n_U)^T s) = \gamma\}. \tag{25}$$

Proposition A.7. *Let $z \in \mathbb{S}^1, U \in V_2(\mathbb{R}^3)$. Consider $\alpha^* \in [0, 2\pi), \beta^* \in [0, \pi], \gamma \in [0, 2\pi)$ as in Lemma A.5 (or equivalently, first, fix $(\alpha^*, \beta^*) \in ((0, 2\pi) \times [0, \pi]) \cup \{0\} \times \{0, \pi\}), \gamma \in [0, 2\pi)$, and then, find z, U according to Lemma A.5). We have:*

$$\mathcal{SC}^\gamma_{n_U} \setminus \{\pm n_U\} = \mathcal{SC}^\gamma_{\alpha^*, \beta^*} \setminus \{\pm n_U\} = \{s \in \mathbb{S}^2 \setminus \{\pm n_U\} : P^U s = z\}.$$

Proof. We have that

$$\{s \in \mathbb{S}^2 \setminus \{\pm n_U\} : P^U s = z\} = \left\{s \in \mathbb{S}^2 \setminus \{\pm n_U\} : \frac{UU^T s}{\|U^T s\|} = Uz\right\}$$
$$= \left\{s \in \mathbb{S}^2 \setminus \{\pm n_U\} : \mathcal{P}^1_{\mathcal{S}(U)}(s) = Uz\right\}$$
$$= \left\{s \in \mathbb{S}^2 \setminus \{\pm n_U\} : \mathcal{P}^2_{\mathcal{S}(U)}(s) = \Psi(n_U)[\cos\gamma, \sin\gamma, 0]^T\right\} \quad \text{(by Proposition A.4 and (23))}$$
$$= \left\{s \in \mathbb{S}^2 \setminus \{\pm n_U\} : \Psi(n_U)\Phi(azi(\Psi(n_U)^T s), \pi/2) = \Psi(n_U)[\cos\gamma, \sin\gamma, 0]^T\right\} \quad \text{(by definition 21)}$$
$$= \left\{s \in \mathbb{S}^2 \setminus \{\pm n_U\} : \Phi(azi(\Psi(n_U)^T s), \pi/2) = [\cos\gamma, \sin\gamma, 0]^T\right\} \quad \text{(applying } \Psi^T(n_U) \text{ on both sides)}$$
$$= \left\{s \in \mathbb{S}^2 \setminus \{\pm n_U\} : azi(\Psi(n_U)^T s) = \gamma\right\} \qquad \text{(by Lemma A.3)}$$
$$= \mathcal{SC}^\gamma_{n_U} \setminus \{\pm n_U\} = \mathcal{SC}^\gamma_{\alpha^*,\beta^*} \setminus \{\pm n_U\}.$$

□

Remark A.8. *With the notation of the above proposition. For each fixed pair $(z, U) \in \mathbb{S}^2 \times V_2(\mathbb{R}^3)$, if we choose distinct $(\alpha^*, \beta^*, \gamma)$ and $(\tilde{\alpha}^*, \tilde{\beta}^*, \tilde{\gamma})$ such that the conditions in Lemma A.5 are satisfied, we obtain*

$$\mathcal{SC}^\gamma_{\alpha^*,\beta^*} = \mathcal{SC}^{\tilde{\gamma}}_{\tilde{\alpha}^*,\tilde{\beta}^*}.$$

A.3 Semicircle Transform and Spherical Radon transform

Let $f \in L^1(\mathbb{S}^2)$. The **Normalized Semicircle Transform** of f is a function $\mathbb{S}^2 \to \mathbb{R}$ defined as:

$$\mathcal{W}f(\alpha^*, \beta^*, \gamma) := \frac{1}{4\pi} \int_{\mathcal{SC}^\gamma_{n_U}} f(s) \sin(zen(s)) d\mathcal{SC}^\gamma_{n_U} \tag{26}$$

where $d\mathcal{SC}^\gamma_{n_U}$ is the curve differential on the semicircle $\mathcal{SC}^\gamma_{n_U}$.

Its **un-normalized version** is defined as

$$\mathcal{UW}f(\alpha^*, \beta^*, \gamma) := \frac{1}{4\pi} \int_{\mathcal{SC}^\gamma_{n_U}} f(s) d\mathcal{SC}^\gamma_{n_U} \tag{27}$$

The **Spherical Radon Transform** of $f \in L^1(\mathbb{S}^{d-1})$, for $d \geq 2$, defined in Bonet et al. (2023) is the mapping $\mathcal{R}f : V_2(\mathbb{R}^d) \times \mathbb{S}^1 \mapsto \mathbb{R}$ given by:

$$\mathcal{R}f(U, z) = \int_{\mathbb{S}^{d-1}} f(s) \delta(z = P^U(s)) ds \tag{28}$$

When considering $d = 3$, we obtain the following relation between the above-defined transforms.

Proposition A.9. *The Unbounded Semicircle Transform 27 and the Spherical Radon Transform 28 are equivalent. In particular, given $z \in \mathbb{S}^1, U \in V_2(\mathbb{R}^3)$, consider $\alpha^* \in [0, 2\pi), \beta^* \in [0, \pi], \gamma \in [0, 2\pi)$ defined as in Lemma A.5, then*

$$\mathcal{UW}f(\alpha^*, \beta^*, \gamma) = \frac{1}{4\pi} \mathcal{R}f(U, z).$$

Proof. From Proposition A.7, we have $\mathcal{SC}^\gamma_{\alpha^*,\beta^*} = \{s : P^U s = z\}$, thus

$$\mathcal{R}f(U, z) = \int_{\mathcal{SC}^\gamma_{n_U}} f(s) d\mathcal{SC}^\gamma_{n_U} = |4\pi| \mathcal{UW}(f)(\alpha^*, \beta^*, \gamma).$$

□

Remark A.10. *We notice that there are other versions of the Spherical Radon Transform. For example, in Groemer (1998) the spherical Radon transformation is defined by integrating over the full great circles $\mathcal{S}(U)$*

$$\widetilde{\mathcal{R}}f(n_U) = \int_{\mathcal{S}(U)} f \, d\mathcal{S}(U).$$

In this paper we are using the variant given in Bonet et al. (2023), which we have proven coincides with the integral transformation over semicircles. That is, this version coincides with the integral transformation denoted by $\mathcal{B}$ in Groemer (1998). It holds that

$$\widetilde{\mathcal{R}}(n_U) = \mathcal{R}f(U, z) + \mathcal{R}f(U, -z)$$

We will review this in Appendix B when generalizing from $d = 3$ to higher dimensions.

We end this section with a discussion about some alternative formulations for $\mathcal{W}f$ and $\mathcal{UW}f$.

For each $s \in \mathcal{SC}^{\gamma}_{\alpha^*,\beta^*} \setminus \{\pm[0, 0, 1]^T\}$, we have

$$azi(\Psi(\alpha^*, \beta^*, 0)^T s) = \gamma.$$

Thus, we can write s as

$$s = \Psi(\alpha^*, \beta^*, 0)\Phi(\gamma, \xi) = \Psi(\alpha^*, \beta^*, \gamma)\Phi(0, \xi),$$

for some $\xi \in (0, \pi)$ (recall that $\Phi(0, \xi) = [\sin(\xi), 0, \cos(\xi)]^T$, and see Groemer (1998) for more details).

Similar to Remark A.6, we can redefine the semicircle as

$$\mathcal{SC}^{\gamma}_{\alpha,\beta} = \{\Psi(\alpha, \beta, \gamma)\Phi(0, \xi) : \xi \in (0, \pi)\},$$

and $\Psi(\alpha^*, \beta^*, \gamma)$ is called **primal median** of $\mathcal{SC}^{\gamma}_{\alpha^*,\beta^*}$.

Combining this with the fact that $(\alpha^*, \beta^*, \gamma) \mapsto \Psi(\alpha^*, \beta^*, \gamma)$ is a bijection, we can rewrite (26), and (27) as

$$\mathcal{W}f(Q) = \frac{1}{4\pi} \int_0^{\pi} f(Q\Phi(0, \xi)) \sin(\xi) d\xi$$
$$\mathcal{UW}f(Q) = \frac{1}{4\pi} \int_0^{\pi} f(Q\Phi(0, \xi)) d\xi$$

where $Q \in SO(3)$ (by using the identity $s = Q\Phi(0, \xi)$, for an appropriate $Q \in SO(3)$).

In Hielscher et al. (2018), the **Un-normalized Semicircle Transform** is defined as

$$\widetilde{\mathcal{UW}}(f)(Q) := \int_{-\pi/2}^{\pi/2} f(Q^T\Phi(\xi, \pi/2)) d\xi, \qquad Q \in SO(3). \tag{29}$$

The equivalence relation between $\mathcal{UW}(f)$ and $\widetilde{\mathcal{UW}}(f)$ reads as follows:

Lemma A.11. *For any $Q \in SO(3)$, we have*

$$\mathcal{UW}f(Q) = \widetilde{\mathcal{UW}}f\left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} Q^T\right).$$

Proof. Given $Q \in SO(3), \xi \in (0, \pi)$, let $Q' = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} Q^T, \xi' = \xi - \beta$. Then,

$$Q'[1, :] = Q^T[1 :] = Q[:, 1]$$
$$Q'[2, :] = -Q^T[3, :] = -Q[:, 3]$$
$$Q'[3, :] = Q^T[2, :] = Q[:, 2]$$

and so

$$\begin{aligned} Q\Phi(0, \xi) &= Q[\sin\xi, 0, \cos\xi]^T \\ &= Q[:, 1]\sin\xi + Q[:, 3]\cos\xi \\ &= Q[:, 1]\cos(\xi') - Q[:, 3]\sin(\xi') \\ &= Q'[1, :]\cos\xi' + Q'[2, :]\sin\xi' \\ &= (Q')^T[\cos\xi', \sin\xi', 0]^T \\ &= (Q')^T\Phi(\xi', \pi/2) \end{aligned}$$

Thus, $\mathcal{UW}f(Q) = \widetilde{\mathcal{UW}}f(Q)$. □

Another formulation has been introduced in Groemer (1998). Since it allows us to define $\mathcal{UW}f$ in higher dimensions ($d \geq 3$), we will discuss it in the next section.

B SPHERICAL RADON TRANSFORM IN $\mathbb{S}^{d-1}$

B.1 GREAT CIRCLE AND SEMICIRCLE / $(d-2)$-DIMENSIONAL HEMISPHERE IN $\mathbb{S}^{d-1}$

We set $d \geq 3$ and consider

$$V_2(\mathbb{R}^d) := \{U \in \mathbb{R}^{d\times 2} : U^T U = I_2\}.$$

Similar to the previous section, we can define 2D **hyperplane**

$$H(U) := \text{span}\{U[:,1], U[:,2]\},$$

and $U^\perp := H(U)^\perp$.

In addition, we define the corresponding **great circle**: $\mathcal{S}(U) = H(U) \cap \mathbb{S}^{d-1}$. Note that the Equator is a specific great circle:

$$\mathcal{E} = \{s \in \mathbb{S}^{d-1} : s^T \mathfrak{n} = 0\},$$

where $\mathfrak{n} = [0, 0, \ldots, 0, 1]^T \in \mathbb{S}^{d-1} \subset \mathbb{R}^d$. Similar to the previous section, the corresponding projection onto the great circle is

$$\mathcal{P}^1_{\mathcal{S}(U)}(s) = \frac{UU^T s}{\|U^T s\|}, \qquad s \in \mathbb{S}^{d-1} \setminus U^T \tag{30}$$

Similar to Proposition A.7 and (24), we define the $(d-2)$-dimensional **hemisphere**:

$$\mathcal{HS}(U, z) := \{s : P^U s = z\} \cup (U^\perp \cap \mathbb{S}^{d-1}) \tag{31}$$

Remark B.1. *When $d = 3$, $U^\perp = \{\pm n_U\}$ and if we ignore these two points,* (31) *is consistent to* (24).

By introducing the hemisphere, the projection 30 can be regarded as the following: Let $s \in \mathbb{S}^d \setminus U^\perp$, we have: $s \in \mathcal{HS}(U, z)$ where $z = \frac{U^T s}{\|U^T s\|}$. Then the function $\mathcal{P}^1_{\mathcal{S}(U)}$ maps s to the intersection point $\mathcal{HS}(U, z) \cap \mathcal{SC}(U)$.

Proposition B.2. *The following identity holds for great circles:*

$$\mathcal{S}(U) = \mathcal{P}^1_{\mathcal{S}(U)}(\mathbb{S}^d \setminus U^\perp) = \{Uz : z \in \mathbb{S}^1\} \tag{32}$$

Proof. Since $\mathcal{P}^1_{\mathcal{S}(U)}$ is the projection, we have $\mathcal{P}^1_{\mathcal{S}(U)}(\mathbb{S}^d \setminus U^\perp) \subset \mathcal{S}(U)$.

For the other direction, choose $s \in \mathcal{S}(U)$ and we decompose s as

$$s = s_U + s_{U^\perp},$$

where $s_U \in H(U), s_{U^\perp} \in U^\perp$.

By definition of $\mathcal{S}(U)$, we have $s_{U^\perp} = 0$. Thus $s = s_U$. From the fact $H(U) \cap U^\perp = \{0_d\}$, we have $s \notin U^\perp$. Therefore, $s \in \mathbb{S}^{d-1} \setminus U^\perp$. Thus $s = \mathcal{P}^1_{\mathcal{S}(U)}(s)$. Therefore, we have

$$\mathcal{S}(U) \subset \mathcal{P}^1_{\mathcal{S}(U)}(\mathbb{S}^d \setminus U^\perp).$$

It remains to verify the second equality. Pick $z \in \mathbb{S}^1$, we have

$$\mathcal{P}^1_{\mathcal{S}(U)}(Uz) = \frac{UU^T Uz}{\|U^T Uz\|} = Uz$$

Thus $Uz \in \mathcal{S}(U)$. For the other direction, pick $s \in \mathcal{S}(U)$, thus

$$s = s_U = UU^T s$$

In addition, $1 = \|s\| = \|UU^T s\| = \|U^T s\|$. Thus, $s \in \{Uz : z \in \mathbb{S}^1\}$ and the second equality holds. □

B.2 GREAT CIRCLES AND HEMISPHERES IN GROEMER (1998)

In this subsection, we re-parametrize the great circles and semicircles by using the notation in Groemer (1998).

Let $\mathcal{B}^d = \{(u, v) : u, v \in \mathbb{S}^{d-1}, u^T v = 0\}$. We define, as in Groemer (1998), the hyperplane $H(u, v) = \text{span}\{u, v\}$ and the corresponding "great circle"

$$\mathcal{S}(u, v) = \mathbb{S}^{d-1} \cap H(u, v).$$

The $(d-2)$-dimensional hemisphere is defined by

$$\mathcal{HS}(u, v) = \{s \in \mathbb{S}^{d-1} : s^T u = 0, s^T v \geq 0\} \tag{33}$$

Lemma B.3. *Pick $U \in V_2(\mathbb{R}^d)$, $z \in \mathbb{S}^1$, there exists $(u, v) \in \mathcal{B}^d$ such that*

$$\textit{span}(U) = \textit{span}(\{u, v\}) \tag{34}$$

$$U^T v = z \tag{35}$$

and vice-versa.

In addition, if U, z, u, v satisfy (34)*, and* (35)*, we have:*

$$\mathcal{S}(U) = \mathcal{S}(u, v)$$
$$\mathcal{HS}(U, z) = \mathcal{HS}(u, v)$$

Proof. First, we pick U, z. We set : $v = Uz$. There exists a uniquely determined (up to scalar ± 1) $u \in H(U) \cap v^\perp$ with $\|u\| = 1$. Since $U^T v = U^T U z = z$, we have that (34), and (35) hold.

Now consider $(u, v) \in \mathcal{B}^d$. We set $U = [u, v]$. (Note that the choice of U is not unique, for example, we can also set $U = [-u, v]$ or $U = [u, -v]$.) Let $z = U^T v = [0, 1]^T$. We have that (34) and (35) hold.

Suppose U, z, u, v satisfy (34) and (35). We have $H(U, z) = H(u, v)$, and thus

$$\mathcal{S}(U) = \mathcal{S}(u, v).$$

Consider $s \in \mathcal{HS}(U, z)$, we decompose s as $s = s_U + s_{U^\perp}$, then (by Proposition B.2)

$$\mathcal{P}^1_{\mathcal{S}(U)}(s) = Uz = UU^T v = v.$$

The last equation holds from the fact $v \in H(U)$. Since $u, v \in H(U)$, we have

$$s^T v = s_U^T v = (UU^T s)^T v = \|U^T s\| v^T v \geq 0$$
$$s^T u = s_U^T u = (UU^T s)^T u = \|U^T s\| v^T u = 0$$

So, $s \in \mathcal{HS}(u, v)$ and thus

$$\mathcal{HS}(U, z) \subset \mathcal{HS}(u, v).$$

For the other direction, we pick $s \in \mathcal{HS}(u, v)$. We have $s^T u = 0$.

Since $u \in H(U)$, we obtain:

$$0 = s^T u = s_U^T u + s_{U^\perp}^T u = s_U^T u$$

By combining this with the fact that $s_U \in H(U) = \text{span}(\{u, v\})$, we have $s_U = \alpha v$ for some $\alpha \in \mathbb{R}$. In addition, from $s^T v \geq 0$, we have $\alpha \geq 0$.

Case 1: If $\alpha = 0$, we have $s \in U^\perp$, thus $s \in \mathcal{HS}(U, z)$.

Case 2: If $\alpha > 0$, we have:

$$P^U s = P^U s_U = \frac{U^T s}{\|U^T s\|} = \frac{U^T \alpha v}{\alpha} = U^T U z = z.$$

Thus, $s \in \mathcal{HS}(U, z)$. Therefore $\mathcal{HS}(u, v) \subset \mathcal{HS}(U, z)$. □

Remark B.4. *When $d = 3$, then $u = U[:, 1]$, $v = U[:, 2]$, and $n_U = \frac{1}{\|u \times v\|} u \times v$ (cross product, normalized).*

Moreover, these $(d-2)$-dimensional hemispheres $\mathcal{HS}$ generalize the semicircles $\mathcal{SC}$ (see (24)*) given for $d = 3$.*

B.3 Spherical Radon transform and a Hemispherical transform

We call the *integral transformation* defined in (Groemer, 1998, Section 2) as the "$(d-2)$-**Hemispherical Transform on** $\mathbb{S}^{d-1}$": it is the $d-$dimensional extension of the Un-normalized **Semicircle Transform** given by (27) for $d=3$. For $f \in L^1(\mathbb{S}^{d-1})$, we define

$$\mathcal{H}f(u,v) := \int_{\mathcal{HS}(u,v)} f(s) d\mathcal{HS}(u,v)$$

where $d\mathcal{HS}(u,v)$ is the surface area differential on $\mathcal{HS}(u,v)$. (We note that in the article Groemer (1998), the notation for the above transformation is $\mathcal{B}$.)

Remark B.5. *In the literature, the so-called* ***Hemispherical Transform*** *integrates over a full* $(d-1)$*-dimensional hemisphere of* $\mathbb{S}^{d-1}$*, that is,*

$$f(z) := \int_{\{x \in \mathbb{S}^{d-1} :\, x^T z \geq 0\}} f(x) dx \qquad (z \in \mathbb{S}^{d-1}). \tag{36}$$

See, e.g., Rubin (1999); Groemer (1998).

The **Spherical Radon Transform** in $\mathbb{S}^{d-1}$ is defined in Bonet et al. (2023) as the linear and bounded integral operator $\mathcal{R} : L^1(\mathbb{S}^{d-1}) \to L^1(V_2(\mathbb{R}^d) \times \mathbb{S}^1)$ with the formula:

$$\mathcal{R}f(U,z) := \int_{\mathbb{S}^{d-1}} \delta(P^U s = z) f(s) ds. \tag{37}$$

Thus, we can re-write it as

$$\mathcal{R}f(U,z) = \int_{\mathbb{S}^{d-1}} \delta(P^U s = z) f(s) ds = \int_{\mathcal{HS}(U,z)} f(s) d\mathcal{HS}(U,z) \tag{38}$$

We define the following *variant* of such Spherical Radon Transform:

$$\widetilde{\mathcal{H}}(f)(U,z) := \int_{\mathbb{S}^{d-1} \setminus U^\perp} \delta(P^U s = z) f(s) ds = \int_{\mathcal{HS}(U,z) \setminus U^\perp} f(s) d\mathcal{HS}(U,z). \tag{39}$$

Proposition B.6. *For* $U \in V_2(\mathbb{R}^d)$ *and* $z \in \mathbb{S}^1$*, consider* (u,v) *as in Lemma B.3. Then*

$$\mathcal{R}f(U,z) = \mathcal{H}f(u,v) = \widetilde{\mathcal{H}}f(U,z), \qquad \forall f \in L^1(\mathbb{S}^{d-1})$$

Proof. By Lemma B.3, we have $\mathcal{HS}(U,z) = \mathcal{HS}(u,v)$. Then, by (38),

$$\mathcal{R}f(U,z) = \mathcal{H}f(u,v).$$

Thus, it remains to show $\widetilde{\mathcal{H}}f = \mathcal{R}f$.

For each (U,z), $U^\perp$ is a $(d-2)$-dimensional subspace. Thus, $\mathbb{S}^{d-1} \cap U^\perp$ is a $(d-3)$-dimensional sub-sphere.

Since $\{s \in \mathbb{S}^{d-1} \setminus U^\perp\}, U^\perp \cap \mathbb{S}^{d-1}$ are disjoint, we have:

$$\begin{aligned} \mathcal{HS}(U,z) \cap U^\perp &= \left[\{s \in \mathbb{S}^{d-1} \setminus U^\perp : P^U s = z\} \cup (U^\perp \cap \mathbb{S}^{d-1})\right] \cap U^\perp \\ &= U^\perp \cap \mathbb{S}^{d-1} \end{aligned} \tag{40}$$

In addition, since $U^\perp \cap \mathbb{S}^{d-1}$ is a $(d-3)$-dimensional sub-sphere, while $\mathcal{HS}(U,z) = \mathcal{HS}(u,z)$ is a $(d-3)$-dimensional hemisphere, we have

$$\begin{aligned} \mathcal{R}f(U,z) &= \widetilde{\mathcal{H}}f(U,z) + \int_{U^\perp \cap \mathbb{S}^{d-1}} f(x) d\mathcal{HS}(U,z) \\ &= \widetilde{\mathcal{H}}f(U,z) + 0 = \widetilde{\mathcal{H}}f(U,z) \end{aligned}$$

and we have completed the proof. □

Corollary B.7. *Given continuous function* $f, g \in C(\mathbb{S}^{d-1})$*, if* $\mathcal{R}f = \mathcal{R}g$*, then* $f = g$*.*

Proof. For every (U, z) there exists $u, v \in \mathbb{S}^{d-1}$ such that, by Lemma B.6,

$$\mathcal{R}f(U, z) = \mathcal{H}f(u, v), \quad \mathcal{R}g(U, z) = \mathcal{H}g(u, v).$$

Thus, $\mathcal{H}f(u, v) = \mathcal{H}g(u, v)$. From (Groemer, 1998, Theorem 1), we have $f = g$. □

Remark B.8. *In the article Bonet et al. (2023) the authors relate the Spherical Radon Transform $\mathcal{R}$ with the Hemispherical Transform $\mathcal{F}$ given in* (36)*. Indeed,*

$$\mathcal{R}f(U, z) = \mathcal{F}(f_{|_{\mathcal{S}(U)}})(z),$$

where $f_{|_{\mathcal{S}(U)}}$ denotes the restriction of $f \in L^1(\mathbb{S}^{d-1})$ to the $(d-2)$-dimensional "great circle" $\mathcal{S}(U) \simeq \mathbb{S}^{d-2}$, and the above operator $\mathcal{F}$ is the Hemispherical Transform in $L^1(\mathcal{S}^{d-2})$. By doing so, the authors can use the characterization of $Ker(\mathcal{F})$ provided in Rubin (1999) to prove their characterization of $Ker(\mathcal{R})$ (see (Bonet et al., 2023, Proposition 4)). Also, in Bonet et al. (2023) the authors claim that they leave for future works checking whether the set $Ker(\mathcal{R})$ is null or not. In this paper we show that it is more natural to relate $\mathcal{R}$ with the integral transformation given in (Groemer, 1998, Section 2) that we have named as the "$(d-2)$-Hemispherical Transform on $\mathbb{S}^{d-1}$" and denoted by $\mathcal{H}$. Thus, (Groemer, 1998, Theorem 1) we can show the injectivity of $\mathcal{R}$ for absolutely continuous measures on the sphere with continuous density functions. (Since $\mathbb{S}^{d-1}$ is compact, the space of continuous functions $C(\mathbb{S}^{d-1})$ is dense in $L^1(\mathbb{S}^{d-1})$ under closure with respect to the L^1-norm. However, we still can not guarantee that the injectivity property of $\mathcal{R}$ extends from $C(\mathbb{R}^{d-1})$ to $L^1(\mathbb{S}^{d-1})$, which represents the densities of absolutely continuous measures).

C Spherical Radon transform for measures on $\mathbb{S}^{d-1}$

Similarly to the classical Radon Transform in Euclidean spaces, the dual-operator of $\mathcal{R}$, called the *back-projection* operator and denoted as $\mathcal{R}^* : C_0(V_2(\mathbb{R}^d) \times \mathbb{S}^1) \to C(\mathbb{S}^{d-1})$, is given in Bonet et al. (2023) by the following formula:

$$\mathcal{R}^*(\psi)(x) = \int_{V_2(\mathbb{R}^d)} \psi(U, P^U(x)) d\sigma(U), \qquad x \in \mathbb{S}^{d-1}, \quad \psi \in C_0(V_2(\mathbb{R}^d) \times \mathbb{S}^1). \tag{41}$$

That is, $\mathcal{R}^*$ is such that the following identity holds:

$$\int_{\mathbb{S}^1 \times V_2(\mathbb{R}^d)} \mathcal{R}f(U, z)\psi(U, z) d\sigma(U) dz = \int_{\mathbb{S}^{d-1}} f(s)\, \mathcal{R}^*\psi(s) ds, \tag{42}$$

for $f \in L^1(\mathbb{S}^{d-1}), \psi \in C_0(V_2(\mathbb{R}^d) \times \mathbb{S}^1)$ (see (Bonet et al., 2023, Proposition 2)). Then, the Spherical Radon transform $\mathcal{R}$ can be extended from $L^1(\mathbb{S}^{d-1})$ to the space of measures supported on $\mathbb{S}^{d-1}$ by defining $\mathcal{R}\mu$ as a new measure on $V_2(\mathbb{R}^d) \times \mathbb{S}^1$ by the "duality expression":

$$\int_{\mathbb{S}^1 \times V_2(\mathbb{R}^d)} \psi(U, z) d\mathcal{R}\mu(U, z) = \int_{\mathbb{S}^{d-1}} \mathcal{R}^*\psi(s) d\mu(s), \qquad \forall \psi \in C_0(V_2(\mathbb{R}^d) \times \mathbb{S}^1). \tag{43}$$

Remark C.1. *For each $s \in \mathbb{S}^{d-1}$, if $s \in U^\perp$, then $U[:, 1], U[:, 2] \in s^\perp$. Thus*

$$\begin{aligned} &\sigma(\{U \in V_2(\mathbb{R}^d) : s \in U^\perp\}) \\ &= \sigma(\{U \in V_2(\mathbb{R}^d) : U[:, 1], U[:, 2] \in s^\perp\}) \\ &= 0 \end{aligned}$$

In Bonet et al. (2023), the authors mention the above that definition is considered a.e.. Via the above property, we have that (41) *is well-defined for all $s \in \mathbb{S}^{d-1}$.*

Now, consider probability measures $\mu \in \mathcal{P}(\mathbb{S}^{d-1})$. In analogy to the classical Radon Transform in Euclidean spaces, by the **disintegration** theorem in classic measure theory, there exists a σ-almost everywhere uniquely determined family of 1-dimensional probability measures $\{(\mathcal{R}\mu)^U\}_{U \in V_2(\mathbb{R}^d)}$ on the unit circle $\mathbb{S}^1$ such that for any $\psi \in C_0(V_2(\mathbb{R}^d) \times \mathbb{S}^1)$ we have

$$\int_{V_2(\mathbb{R}^d) \times \mathbb{S}^1} \psi(U, z)\, d\mathcal{R}\mu(U, z) = \int_{V_2(\mathbb{R}^d)} \int_{\mathbb{S}^1} \psi(t, \theta)\, d(\mathcal{R}\mu)^U(z)\, d\sigma(U). \tag{44}$$

In (Bonet et al., 2023, Proposition 3) the authors prove that if μ_f is an absolutely continuous probability measure on the sphere $\mathbb{S}^{d-1}$, that is, it has a density $f \in L^1(\mathbb{S}^{d-1})$, then

$$(\mathcal{R}\mu_f)^U = P^U_\#(\mu_f)$$

The same computation can be extended to any measure μ supported on $\mathbb{S}^{d-1}$ to get $(\mathcal{R}\mu)^U = P^U_\#(\mu)$.

Remark C.2. *In the particular case of μ_f, that is, for functions $f \in L^1(\mathbb{S}^{d-1})$ we have that the measure $\mathcal{R}\mu_f$ has density $\mathcal{R}f$. This implies that the function $\mathcal{R}f(U,\cdot)$ (i.e., the function $z \mapsto \mathcal{R}f(U,z)$ defined for $z \in \mathbb{S}^1$) is the density of $(\mathcal{R}\mu_f)^U$, and therefore $\mathcal{R}f(U,\cdot)$ is also the density of $P^U_\#(\mu_f)$.*

C.1 Linear Spherical Sliced Optimal Transport in terms of the Spherical Radon Transform

By using the previous expressions, given $\nu^1, \nu^2 \in \mathcal{P}(\mathbb{S}^{d-1})$ we can re-write $LSSOT_2(\cdot,\cdot)$ as follows:

$$\begin{aligned}(LSSOT_2(\nu_1,\nu_2))^2 &:= \int_{V_2(\mathbb{R}^d)} \left(LCOT_2(P^U_\#\nu_1, P^U_\#\nu_2)\right)^2 \, d\sigma(U) \\ &= \int_{V_2(\mathbb{R}^d)} \left(LCOT_2((\mathcal{R}\nu_1)^U, (\mathcal{R}\nu_2)^U)\right)^2 \, d\sigma(U). \end{aligned} \tag{45}$$

D Metric property of Linear Spherical Sliced Optimal Transport

Proposition D.1. *$LSSOT_2(\cdot,\cdot)$ is a well-defined **pseudo-metric** in $\mathcal{P}(\mathbb{S}^{d-1})$, that is, $LSSOT_2(\nu,\nu) = 0$ for all $\nu \in \mathcal{P}(\mathbb{S}^1)$, it is non-negative, symmetric, and satisfies the triangle inequality.*

Proof. Let $\nu_1, \nu_2, \nu_3 \in \mathcal{P}(\mathbb{S}^{d-1})$.

It is straightforward to verify positivity (i.e., $LSSOT_2(\nu_1,\nu_2) \geq 0$), and symmetry (i.e., $LSSOT_2(\nu_1,\nu_2) = LSSOT_2(\nu_2,\nu_1)$).

If $\nu_1 = \nu_2$, for each $U \in V_2(\mathbb{R}^d)$, we have $P^U_\#\nu_1 = P^U_\#\nu_2$, thus $\hat{\nu}^S_1(\cdot,U) = \hat{\nu}^S_2(\cdot,U)$. It implies that $\|\hat{\nu}^S_1(\cdot,U) - \hat{\nu}^S_2(\cdot,U)\|^2_{L^2(\mathbb{S}^{d-1})} = 0$, and so $LSSOT_2(\nu_1,\nu_2) = 0$.

Finally, we verify the triangle inequality. Let $U \in V_2(\mathbb{R}^d)$. By (Martin et al., 2023, Theorem 3.6), we have $LCOT_2(\cdot,\cdot)$ defines a metric. Thus, the following triangle inequality holds:

$$LCOT_2(P^U_\#\nu_1, P^U_\#\nu_2) \leq LCOT_2(P^U_\#\nu_1, P^U_\#\nu_3) + LCOT_2(P^U_\#\nu_2, P^U_\#\nu_3).$$

Thus,

$$\begin{aligned}
&LSSOT_2(\nu_1,\nu_2) \\
&= \left(\int_{V_2(\mathbb{R}^d)} (LCOT_2(P^U_\#\nu_1, P^U_\#\nu_2))d\sigma(U)\right)^{1/2} \\
&\leq \left(\int_{V_2(\mathbb{R}^d)} (LCOT_2(P^U_\#\nu_1, P^U_\#\nu_3) + LCOT_2(P^U_\#\nu_2, P^U_\#\nu_3))d\sigma(U)\right)^{1/2} \\
&\leq \left(\int_{V_2(\mathbb{R}^d)} (LCOT_2(P^U_\#\nu_1, P^U_\#\nu_3))d\sigma(U)\right)^{1/2} + \left(\int_{V_2(\mathbb{R}^d)} (LCOT_2(P^U_\#\nu_2, P^U_\#\nu_3))d\sigma(U)\right)^{1/2} \qquad (46) \\
&= LSSOT_2(\nu_1,\nu_3) + LSSOT_2(\nu_2,\nu_3)
\end{aligned}$$

where (46) holds by Minkowski inequality in space $L^2(V_2(\mathbb{R}^d), d\sigma)$. □

Theorem D.2 (Theorem 3.3)**.** *$LSSOT_2(\cdot,\cdot)$ is a well-defined **metric** in the set of absolutely continuous measures in $\mathcal{P}(\mathbb{S}^{d-1})$ with continuous density function, that is, it is pseudo-metric satisfying also the identity of indiscernibles.*

Proof. From Proposition D.1, we only have to prove that if $LSSOT_2(\nu_1, \nu_2) = 0$, then $\nu_1 = \nu_2$. To do so, we will the expression of the $LSSOT_2(\cdot, \cdot)$ in terms of a Spherical Radon Transform given in given in (45). Let $\nu_1, \nu_2 \in \mathcal{P}(\mathbb{S}^{d-1})$ such that

$$0 = \left(LSSOT_2(\nu_1, \nu_2)\right)^2 = \int_{V_2(\mathbb{R}^d)} \left(LCOT_2((\mathcal{R}\nu_1)^U, (\mathcal{R}\nu_2)^U\right)^2 \, d\sigma(U).$$

Thus,

$$LCOT_2((\mathcal{R}\nu_1)^U, (\mathcal{R}\nu_2)^U) = 0 \qquad \sigma(U) - a.s.$$

Since $LCOT_2(\cdot, \cdot)$ defines a metric in $\mathcal{P}(\mathbb{S}^1)$ (see again (Martin et al., 2023, Theorem 3.6)), we have that

$$(\mathcal{R}\nu_1)^U = (\mathcal{R}\nu_2)^U \qquad \sigma(U) - a.s.$$

As the disintegration is a family σ-almost everywhere uniquely determined, we have that

$$\mathcal{R}\nu_1 = \mathcal{R}\nu_2.$$

Finally, if ν_1, ν_2 have continuous density functions, by injectivity of the operator $\mathcal{R}$ (Corollary B.7), we have that $\nu_1 = \nu_2$. □

As a byproduct of our deductions, we obtain that the SSW in Bonet et al. (2023) and defined in (8), is also a metric in $\mathcal{P}(\mathbb{S}^{d-1})$:

Theorem D.3. *The Spherical Sliced Wasserstein discrepancy $SSW_p(\cdot, \cdot)$, for $p \geq 1$, defines a metric in the set of absolutely continuous measures in $\mathcal{P}(\mathbb{S}^{d-1})$ with continuous density function.*

Proof. First, notice that since the sphere is compact, there is no need to ask that $\nu \in \mathcal{P}(\mathbb{S}^{d-1})$ has to have finite p-moment, as it automatically has such a property. In (Bonet et al., 2023, Proposition 5), the authors show that $SSW_p(\cdot, \cdot)$ is a pseudo-metric. Finally, by using the expression

$$\left(SSW_p(\mu, \nu)\right)^p := \int_{V_2(\mathbb{R}^d)} \left(COT_p(P^U_\# \mu, P^U_\# \nu)\right)^p \, d\sigma(U) \tag{47}$$

$$= \int_{V_2(\mathbb{R}^d)} \left(COT_p((\mathcal{R}\mu)^U, (\mathcal{R}\nu)^U)\right)^p \, d\sigma(U) \tag{48}$$

and repeating the arguments in the proof of Theorem 3.3 with the Wasserstein distance $COTp(\cdot, \cdot)$ in the unit circle in place of the LCOT distance, we get that $SSW_p(\cdot, \cdot)$ satisfies the identity of indiscernibles when restricted to probability measures with continuous densities. □

E Runtime Analysis with respect to Number of Slices

Here we provide further running time comparisons of the slice-based methods for pairwise distance calculations under different numbers of slices $L = 500, 1000, 5000$. All methods are run on NVIDIA RTX A6000 GPU.

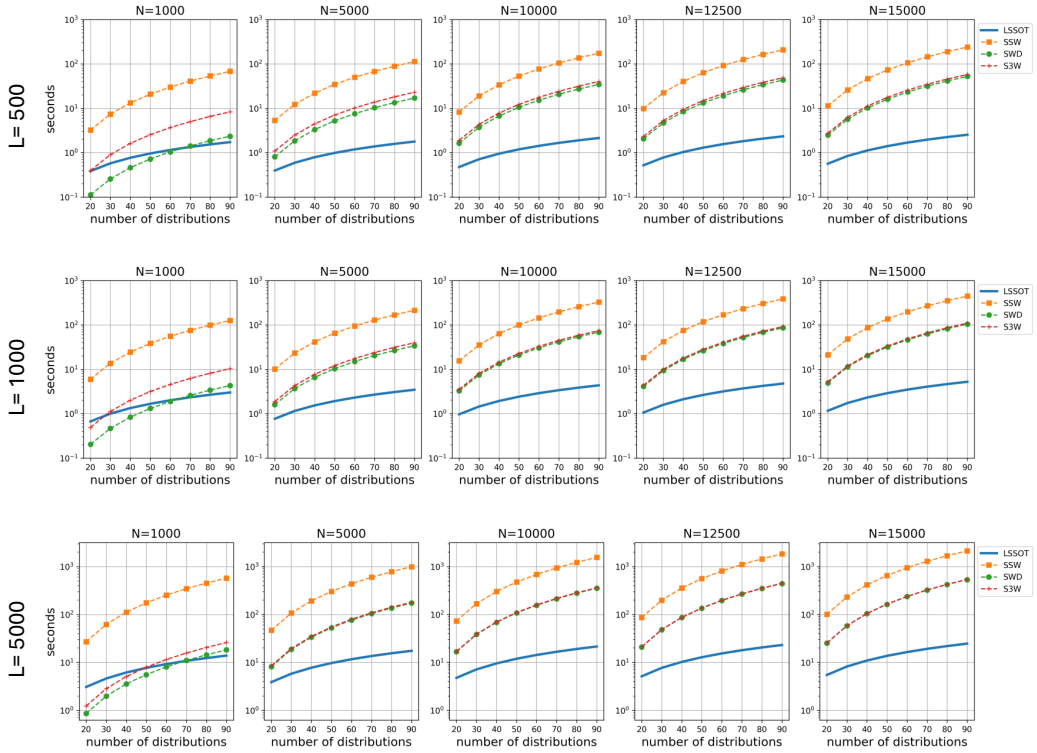

Figure 6: Running time comparison under varying number of slices. L denotes the number of slices used in all the slice-based methods in each row. N is the number of samples in each distribution for each column.

F Implementation Details of Cortical Surface Registration

F.1 Icospheres and Resampling

S3Reg relies on Spherical-Unet (Zhao et al., 2019) as the backbone, which capitalizes on the regular structure of icospheres to perform convolution and pooling on spheres. Spherical surfaces are resampled on icospheres in order to be input to the S3Reg network.

Icospheres are generated from a regular convex icosahedron with 12 vertices, then by recursively adding a new vertex to the center of each edge in each triangle. The number of vertices on the icospheres is increased from 12 to 42 (level 1), 162 (level 2), 642 (level 3), 2562 (level 4), 10,242 (level 5), 40,962 (level 6), 163,842 (level 7), etc. In this work, we focus on the level-6 icosphere with 40,962 vertices on the cortical surfaces.

For resampling in the preprocessing step, we employ the simple but fast Nearest Neighbor method and implement it using sklearn.neighbors.KDTree function from Scikit Learn (Pedregosa et al., 2011). The feature value on each output point is represented as a weighted sum of the feature values on the nearest three vertices.

F.2 S3Reg Framework and Loss Components

In a nutshell, the S3Reg framework receives a concatenation of fixed and moving spherical surface maps defined on icospheres as input, processes them through an encoder-decoder architecture, and finally outputs the velocity field represented by tangent vectors on $\mathbb{S}^2$. In particular, as the convolution kernel in Spherical U-Net could cause distortions around the poles (Zhao et al., 2019), 3 complementary orthogonal Spherical U-Nets are combined to predict the deformation field so that the polar areas are also regularly processed at least two times. As the 3 predicted velocity fields should

be orthogonal accordingly, a loss term of inconsistency is added to the total loss. We refer the readers to Zhao et al. (2021) (Section III-A) for details of the consistency loss.

As the similarity losses are thoroughly discussed in the main text, we will now describe other loss components in more detail. Let F denote the fixed image and M be the moving image. Suppose p_F, p_M are the corresponding parcellation maps of F and M, respectively. Let ϕ denote the velocity field predicted by the network.

Dice Score and Dice Loss Parcellation maps can be integrated into the training process to enforce biological validity. The Dice Score is a measure of similarity/overlap between two parcellations. Assume both p_F and p_M have K regions of interest (ROIs), then the Dice score is defined as

$$\text{Dice}(p_F, p_M \circ \phi) = \frac{1}{K} \sum_{k=1}^{K} 2 \frac{|p_F^k \cap (p_M^k \circ \phi)|}{|p_F^k| + |p_M^k \circ \phi|}, \tag{49}$$

where $p_M^k \circ \phi$ is the warped moving image by ϕ, the superscript k denotes the k-th ROI. Since the Dice Score is between 0 and 1, the Dice Loss can be defined as $L_{\text{Dice}}(p_F, p_M \circ \phi) = 1 - \text{Dice}(p_F, p_M \circ \phi)$.

Smoothness Loss The local spatial non-smoothness of the output velocity field ϕ is penalized by $L_{\text{smooth}}(\phi) = \sum_v |\nabla_{\mathbb{S}^2} C(\phi(v))|$, where the sum is over all the moving vertices v, C is the 1-ring convolution operator (Zhao et al., 2019), and $\nabla_{\mathbb{S}^2}$ denotes the spherical gradients.

G More visualizations of Cortical Surface Registration

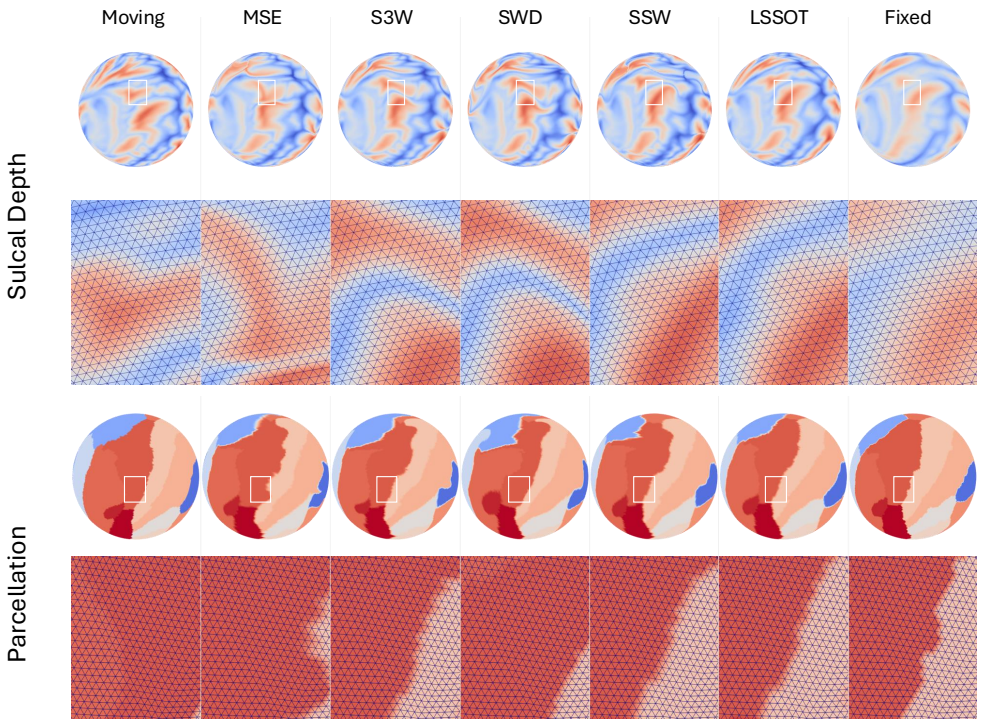

Figure 7: Qualitative registration results from Subject A00060516 in the NKI dataset.

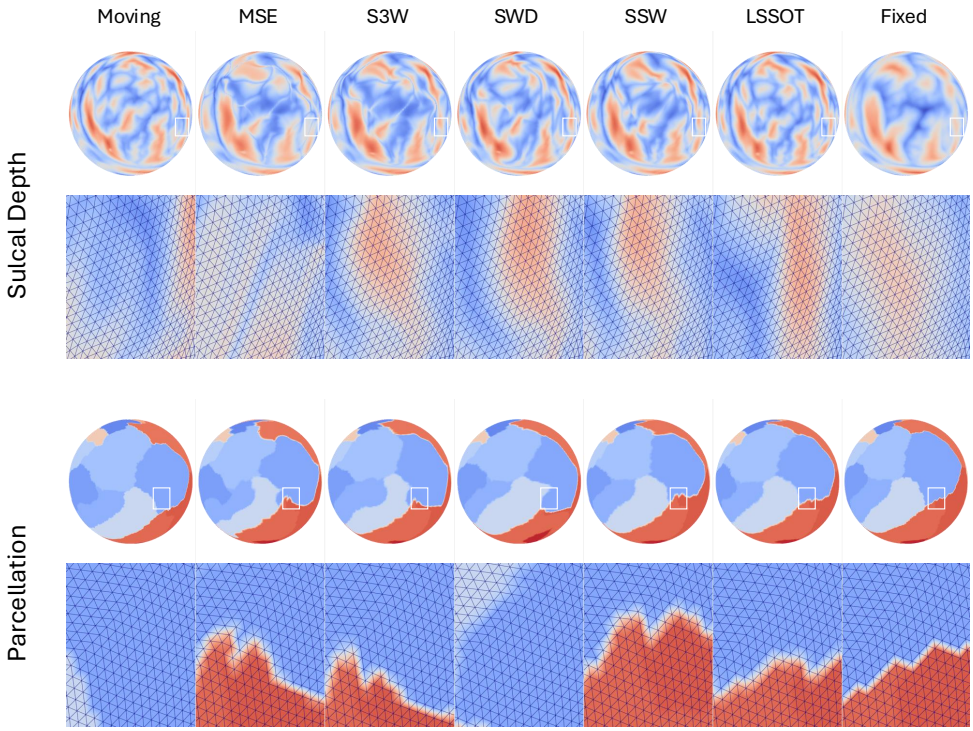

Figure 8: Qualitative registration results from Subject A00076798 in the NKI dataset.

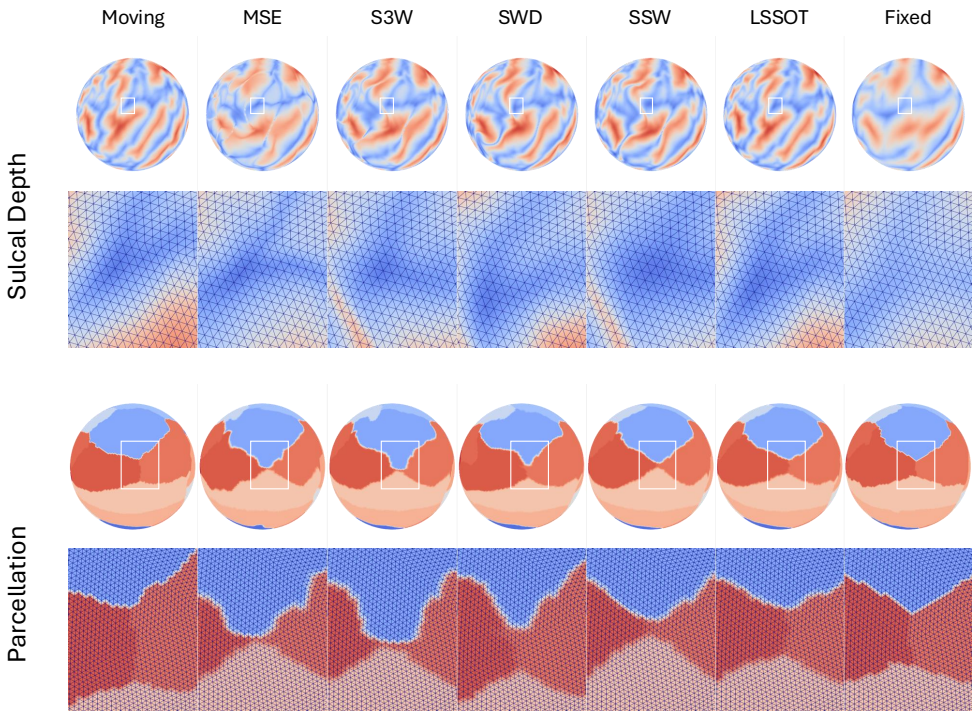

Figure 9: Qualitative registration results from Subject 0356 in the ADNI dataset.

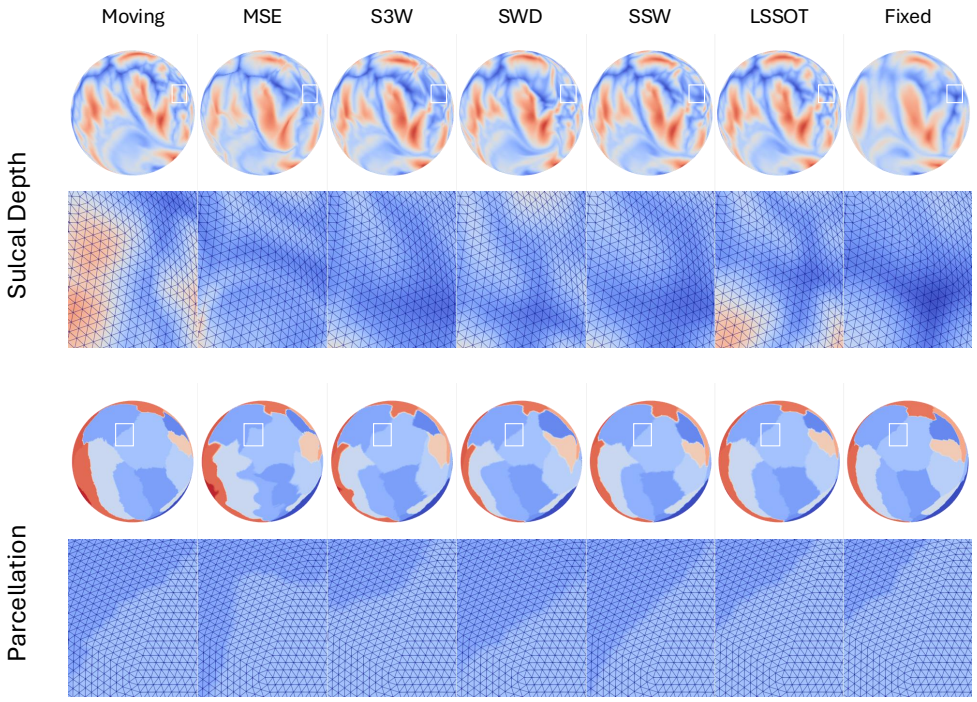

Figure 10: Qualitative registration from Subject 6650 in the ADNI dataset.

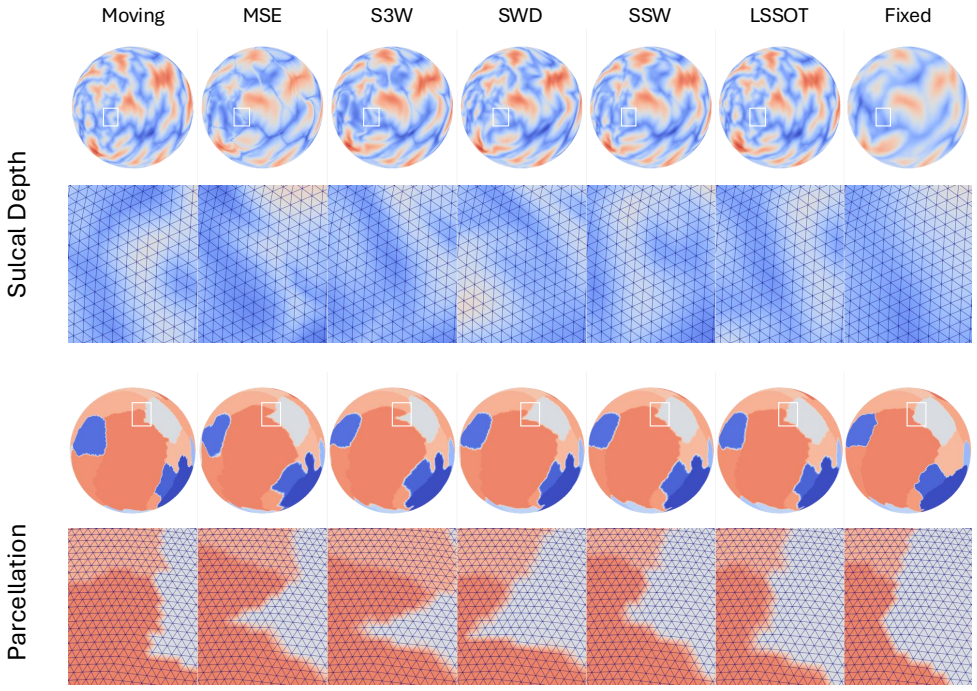

Figure 11: Qualitative registration results from Subject 7060 in the ADNI dataset.

H 3D SHAPE CLUSTERING AND RETRIEVAL

To further validate the effectiveness of LSSOT in evaluating the similarity of spherical distributions, we conduct additional experiments using a variety of genus-0 3D shapes. Our methodology involves a two-step process: 1) mapping the 3D shapes onto spherical surfaces; and 2) performing clustering and nearest neighbor retrieval tasks on these spherical representations. For the spherical mapping process, we employ two distinct techniques, Inflation and Projection (IAP) and Deep Unsupervised Learning using PointNet++ Qi et al. (2017).

H.1 DATASET

We conduct experiments on mesh models from the Watertight Track of SHREC 2007 dataset (Giorgi et al., 2007). This dataset contains surfaces belonging to 19 categories, among them, we select 4 genus-0 closed surface models, human, octopus, teddy bear, and Fish. Moreover, as explored in the previous experiment, LSSOT is sensitive to rotations, therefore we remove the meshes with different orientations than the rest.

H.2 SPHERICAL MAPPING

Inflation and Projection. A direct projection from a genus-0 closed surface to the sphere by fixing an inside point source and normalizing the vertices may cause undesirable topologically incorrect foldings. One workaround is iteratively smoothing and inflating the surface before the final projection. This inflation and projection is a common spherical mapping strategy used in neuroimaging data processing tools such as FreeSurfer (Fischl, 2012), Connectome Workbench (Marcus et al., 2013).

Deep Unsupervised Learning using PointNet++. To accelerate the forward pass of spherical mapping, we capitalize on deep unsupervised learning and train a PointNet++ network to map a cortical surface on a sphere with minimal distortion. The trained network then serves as a deep learning-based spherical mapping tool.

Loss function. Zhao et al. (2022) proposed an unsupervised approach of mapping cortical surfaces to a sphere using Spherical U-Net Zhao et al. (2019) to learn the spherical diffeomorphic deformation field for minimizing the metric (distance), area, or angle distortions. We follow their definitions of metric (J_d) and area (J_a) distortions as local relative differences, and conformal/angle distortion (J_c) after market share normalization, but adapt them to general meshes instead of restricting to icosahedron-reparameterized surfaces. For a pair of original input surface R and the predicted spherical mesh S, the loss function has three components:

$$J_d(R,S) = \min_{r\in\mathbb{R}^+} \frac{1}{N_e}\sum_{i=1}^{N_e} \frac{|rd_i^S - d_i^R|}{d_i^R} \tag{50}$$

$$J_a(R,S) = \min_{r\in\mathbb{R}^+} \frac{1}{N_f}\sum_{j=1}^{N_f} \frac{|ra_j^S - a_j^R|}{a_j^R} \tag{51}$$

$$J_c(R,S) = \frac{1}{3N_f}\sum_{k=1}^{N_f} \frac{1}{\pi}(|\alpha_k^R - \alpha_k^S| + |\beta_k^R - \beta_k^S| + |\theta_k^R - \theta_k^S|) \tag{52}$$

where N_e is the total number of edges, N_f is the total number of faces. d_i^R and d_i^S are the length of i-th edge on original surface and predicted sphere. Similarly, a_j^R and a_j^S are the area of j-th face on original surface and predicted sphere; $(\alpha_k^R, \beta_k^R, \theta_k^R)$ and $(\alpha_k^S, \beta_k^S, \theta_k^S)$ are the inner angles of the k-th face on original surface and predicted sphere.

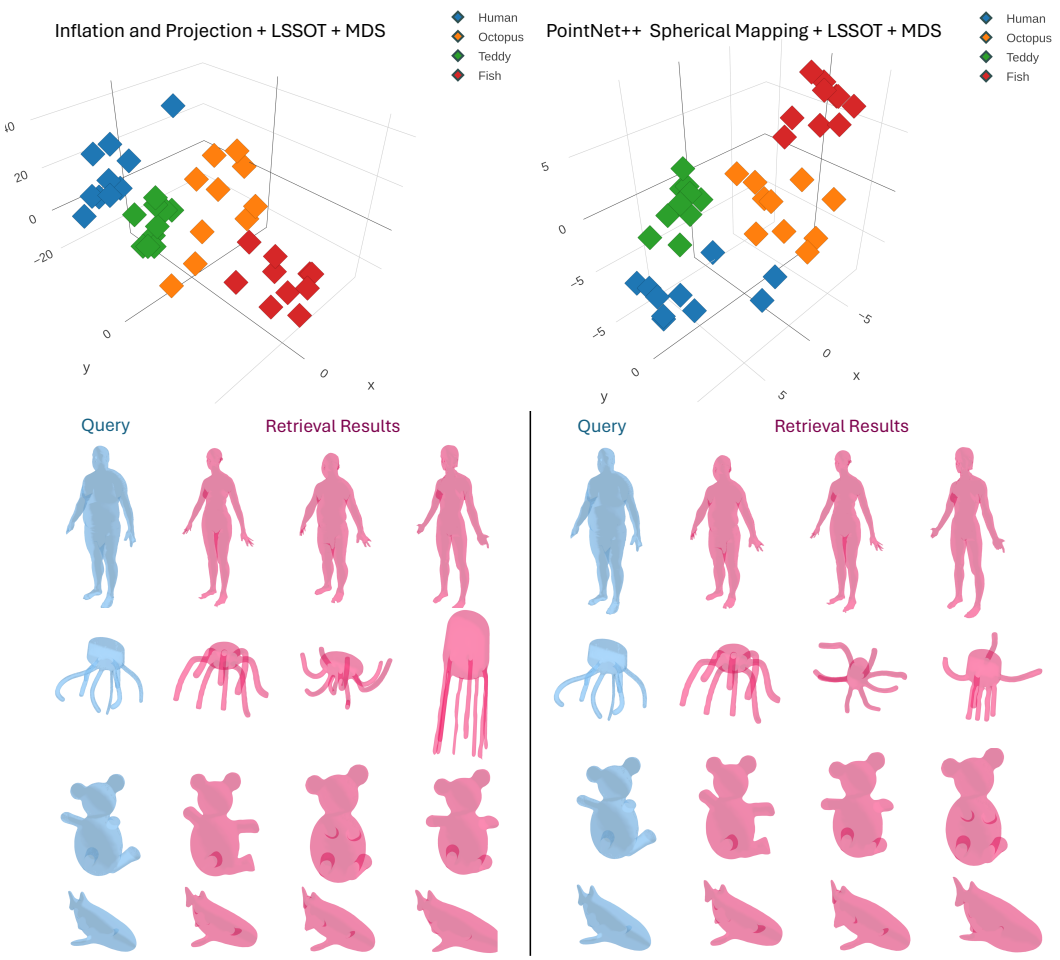

Figure 12: Manifold learning of clustering results (top) and nearest neighbor retrieval results (bottom) using Inflation and Projection (left) and PointNet++ (right) as spherical mappings.

H.3 Results

Figure 12 illustrates the effectiveness of the LSSOT metric under both spherical mapping techniques. The results demonstrate that LSSOT successfully identifies inherent similarities and dissimilarities among the spherical surface representations, regardless of the mapping method employed. This capability translates into good performance in both clustering and nearest neighbor retrieval tasks. The consistency of LSSOT's effectiveness under different mapping approaches underscores its robustness as a metric for analyzing spherical distributions and comparing surface geometries.

I Gradient flow with respect to LSSOT in the spherical space

I.1 Background: introduction to Stiefel manifold

In this section, we briefly introduce the gradient flow and related concepts in spherical space. First, we introduce the Riemannian structure of the Stiefel manifold, which is a general case of spherical space. Next, we specify the related concepts in the spherical space.

Given positive integers $1 \leq k \leq d$, the Stiefel manifold is given by

$$\mathbb{V}_{d,k} := \mathbb{V}_k(\mathbb{R}^d) = \{V \in \mathbb{R}^{d\times k} : V^\top V = I_k\}.$$

It is a $\left(dk - (k + \binom{k}{2})\right)$-dimensional manifold.

Given $X \in \mathbb{V}_{d,k}$, the tangent space is given by:

$$\mathcal{T}_X(\mathbb{V}_{d,k}) := \{Z \in \mathbb{R}^{d\times k} : ZX + XZ = 0\}. \tag{53}$$

Given matrix $G \in \mathbb{R}^{d\times k}$, the projection of G onto $\mathcal{T}_X(\mathbb{V}_{d,k})$ is given by,

$$P_{T_X(\mathbb{V}_{d,k})}(G) := G - X\text{sym}(G^\top X), \tag{54}$$

where $\text{sym}(G^\top X) = \frac{1}{2}(G^\top X + X^\top G)$ is the symmetric matrix obtained from $G^\top X$.

Given a differentiable function $F : V_{d,k} \to \mathbb{R}$, the following steps can describe the gradient of F in the manifold. In particular, choose $X \in \mathbb{V}_{d,k}$:

- Step 1: compute Euclidean gradient $\nabla_E(F(X))$.
- Step 2: the Stiefel manifold gradient is defined as:

$$\nabla_S F(X) := P_{\mathcal{T}_X(\mathbb{V}_{d,k})}(\nabla_E(F(X))). \tag{55}$$

We consider the gradient flow problem:

$$\partial_t X_t = -\nabla_S F(X_t), \forall X \in \mathbb{V}_{d,k}.$$

Let η denote the step size (i.e., the learning rate for the machine learning community), the discrete gradient descent step of the above problem becomes:

$$X \mapsto X - \eta\nabla_S F(X) = X - \eta(\nabla_E F(X) - X\text{sym}(\nabla_E F(X)^\perp X)). \tag{56}$$

I.2 Gradient flow in spherical space

In spherical space, we have $k = 1$ (in this case, we use convention $\binom{1}{2} = 0$). Thus, the tangent space (53) becomes:

$$\mathcal{T}_x(\mathbb{S}^{d-1}) := \text{span}(x)^\perp := \{z \in \mathbb{R}^d : z^\top x = 0\}, \tag{57}$$

where $\text{span}(x)$ is the 1D line spaned by x. The corresponding projection (54) becomes

$$\mathbb{R}^d \ni g \mapsto P_{\mathcal{T}_x(\mathbb{V}_{d,k})}(g) := g - x(g^\top x) \in \mathcal{T}_x(\mathbb{S}^{d-1}). \tag{58}$$

Note, the above projection is essentially the Euclidean projection of x into the hyperplane $x^\perp$.

Given $X = [x_1, \ldots, x_n] \in \mathbb{R}^{d\times n}, Y = [y_1, \ldots, y_m] \in \mathbb{R}^{d\times m}$, where Y is fixed. Let $\nu_1 = \sum_{i=1}^n p_i\delta_{x_i}, \nu_2 = \sum_{j=1}^m q_j\delta_{y_j}$. We consider the following function :

$$\begin{aligned} C(X) &:= LSSOT(\mu, \nu) \\ &\approx \frac{1}{T}\sum_{t=1}^{T} LCOT((\theta_t)_\#\mu, (\theta_t)_\#\nu). \end{aligned} \tag{59}$$

where each $\theta_t \in \mathbb{S}^{d-1}$, (59) is the Monte Carlo approximation.

We select one θ and consider the 1-dimensional LCOT problem

$$\begin{aligned} &LCOT((\theta)_\#\nu_1, (\theta)_\#\nu_2) \\ &= \int_0^1 h\left(\left|F^{-1}_{\theta_\#\nu_1}(s - \mathbb{E}[\theta_\#\nu_1] + \frac{1}{2}) - |F^{-1}_{\theta_\#\nu_2}(s - \mathbb{E}[\theta_\#\nu_2] + \frac{1}{2})\right|^2\right) ds \end{aligned}$$

where $h(r) = \min(r, 1 - r)$.

Note, the above quantity implicitly defines a transportation plan between μ and ν. In particular, suppose $\gamma_\theta \in \mathbb{R}^{n\times m}$, with

$$\gamma_\theta[i, j] = |\{s : F^{-1}_{\theta_\#\nu_1}(s - \mathbb{E}[\theta_\#\nu_1]) \equiv x_i, F^{-1}_{\theta_\#\nu_1}(s - \mathbb{E}[\theta_\#\nu_2]) \equiv y_j\}|, \tag{60}$$

where $|\cdot|$ is the uniform measure (Lebesgue measure) in $[0, 1)$, $a \equiv_1 b$ means $a \mod 1 = b$. It is straightforward to verify $\gamma_\theta \in \Gamma(\theta_\#\nu_1, \theta_\#\nu_2)$, and thus is a transportation plan between $\theta_\#\nu_1, \theta_\#\nu_2$.

Therefore, we can explicitly write (59) in term of $X = [x_1, \dots x_n]$:

$$(59) = \frac{1}{T}\sum_{t=1}^{T}\sum_{i,j} h(\theta_t^\top x_i - \theta_t^\top y_j)\gamma_{\theta_t}[i,j]. \tag{61}$$

where each γ_{θ_t} is defined via (60). For each x_i, the Euclidean gradient is given by

$$\nabla_E C(x_i) = \frac{1}{T}\sum_{t=1}^{T}\theta_t \left(\sum_{j:|\theta_t^\top x_i - \theta_t^\top y_j| \le 1/2} 2(\theta_t^\top x_i - \theta_t^\top y_j)\gamma_{\theta_t}[i,j] \right.$$
$$\left. - \sum_{j:|\theta_t^\top x_i - \theta_t^\top y_j| > 1/2} 2(\theta_t^\top x_i - \theta_t^\top y_j)\gamma_{\theta_t}[i,j] \right). \tag{62}$$

Based on this, we derive the gradient descent step for the following gradient flow problem.

$$-\nabla_S C(X) = -V(X)$$

That is,

$$X \mapsto X - \eta\nabla_S(C(X)) = X - (X - X(\nabla_E C(X)^\top X)) \tag{63}$$

where $\nabla_E(C(X))[:, i]$ is given by (62) for each i.

I.3 Implementation of Gradient Flow

Riemannian Gradient Descent. Let $\mu_{\mathcal{X}} = \frac{1}{N}\sum_{n=1}^{N}\delta_{x_n}$ be the source distribution with mass uniformly distributed at $\mathcal{X} = \{x_n\}_{n=1}^{N} \subset S^{d-1}$, and $\nu_{\mathcal{Y}} = \frac{1}{M}\sum_{m=1}^{M}\delta_{y_m}$ be the target distribution with $\mathcal{Y} = \{y_m\}_{m=1}^{M} \subset S^{d-1}$. For each $x \in \mathcal{X}$, let $x^{(k)}$ be the location of x at k-th iteration, then at $(k+1)$-th iteration, the update should be the following (Absil et al., 2008; Boumal et al., 2019):

$$x^{(k+1)} = \exp_{x^{(k)}}(-\gamma_k \nabla_{S^{d-1}}\mathcal{F}(x^{(k)})), \tag{64}$$

where γ_k is the learning rate at k-th iteration, the exponential map $\exp_x(v) = x\cos\|v\| + \frac{v}{\|v\|}\sin\|v\|$, and $\nabla_{S^{d-1}}\mathcal{F}(x^{(k)})$ is the projection of the Euclidean gradient $\nabla_{\mathbb{R}^d}\mathcal{F}(x^{(k)})$ onto the tangent space at $x^{(k)}$. We implement this gradient descent method using the exponential function and projection function in the Geoopt library (Kochurov et al., 2020) [5].

Spherical Coordinate Gradient Descent. A more straightforward gradient descent method is to work in the spherical coordinate system. By the following correspondences (in $\mathbb{S}^2$ for example) between $(x, y, z) \in \mathbb{S}^2$ and (θ, ϕ)

$$x = \cos\theta\sin\phi$$
$$y = \sin\theta\sin\phi$$
$$z = \cos\phi$$

and

$$\theta = \arctan(\frac{y}{x})$$
$$\phi = \arccos z$$

we can calculate the gradient of the loss function with respect to (θ, ϕ). Note here $\arctan$ must be suitably defined according to the quadrant.

[5] https://github.com/geoopt/geoopt

I.4 Additional Point Cloud Interpolation Visualizations

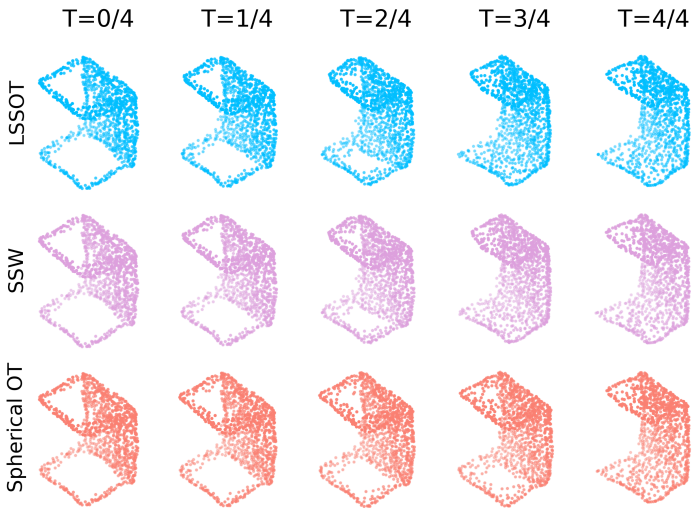

Figure 13: Gradient flow interpolations between two tables.

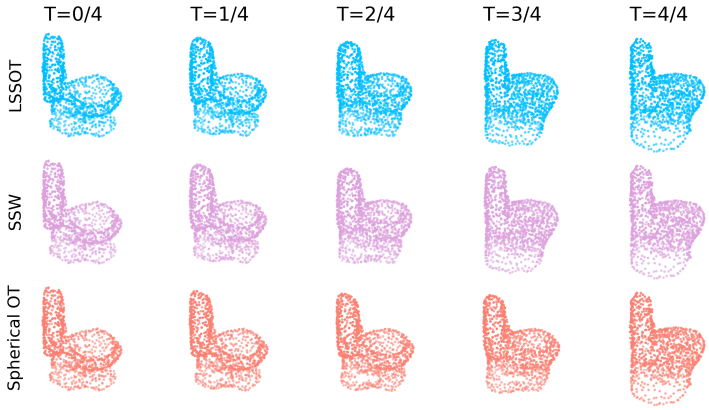

Figure 14: Gradient flow interpolations between two toilets.

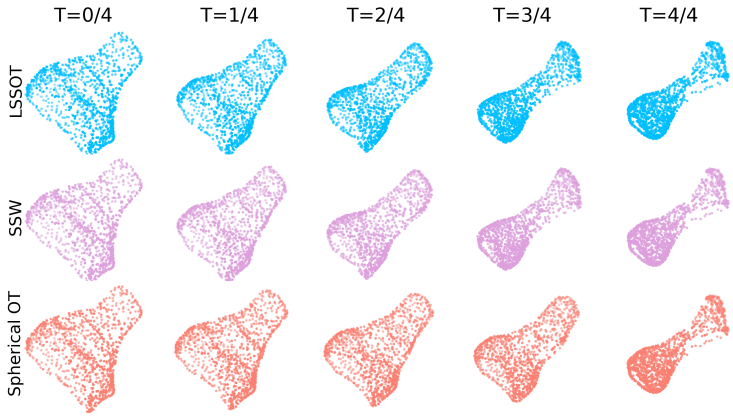

Figure 15: Gradient flow interpolations from a range hood to a plant.

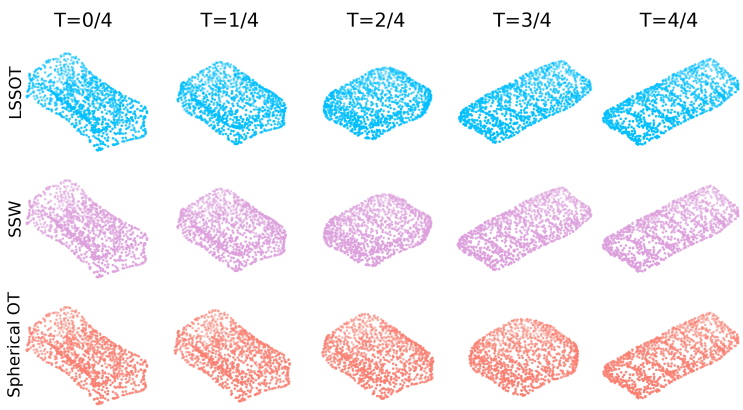

Figure 16: Gradient flow interpolations from a TV stand to a bookshelf.

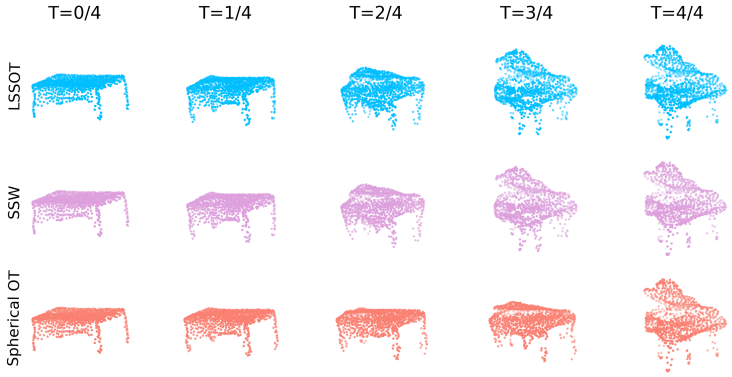

Figure 17: Gradient flow interpolations from a table to a piano.

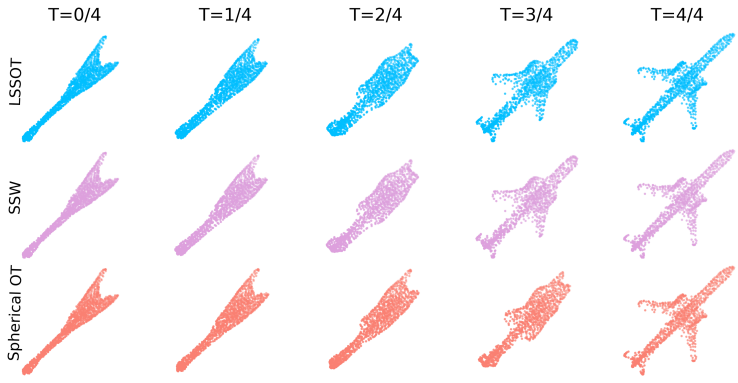

Figure 18: Gradient flow interpolations from a guitar to an airplane.

I.5 Gradient Flow Mollweide View Visualizations

We provide more visualizations for the gradient flow of spherical distributions with respect to LSSOT metric, SSW, and spherical optimal transport distance (spherical OT). At first, a source distribution and a target distribution are generated, both of which are discrete distributions with uniform mass at all points. Then we perform gradient descent on the mass locations of the source distribution with respect

to the above three objectives, driving the source toward the target distribution by minimizing these losses. We employ two methods of gradient descent on the sphere. See Figure 19 to Figure 30 for the Mollweide views of the gradient flows between two different pairs of source-target distributions.

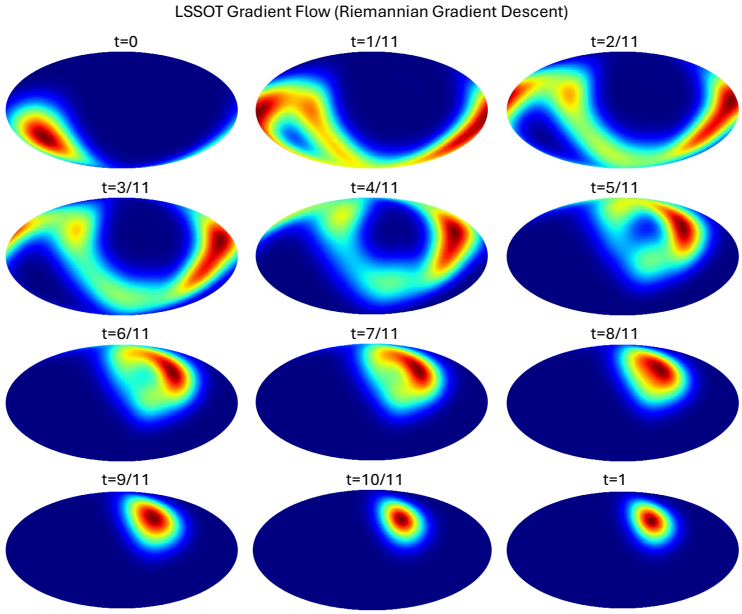

Figure 19: LSSOT gradient flow using Riemannian gradient descent.

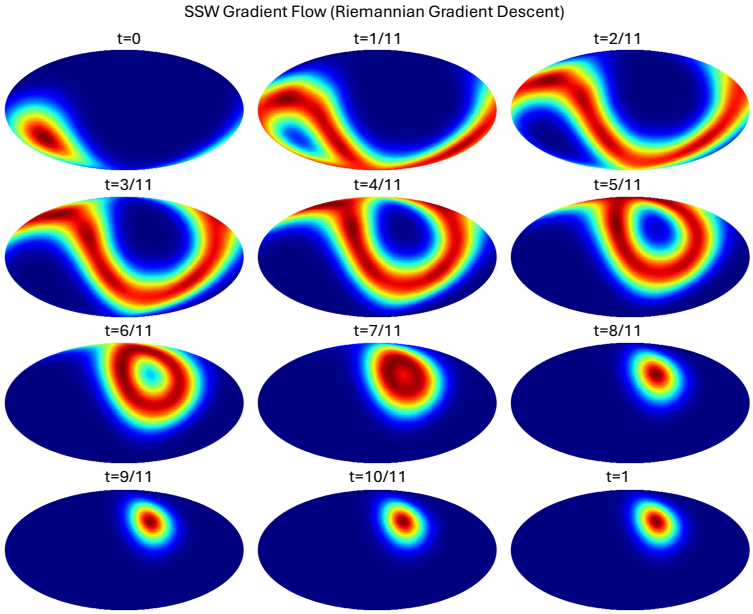

Figure 20: SSW gradient flow using Riemannian gradient descent.

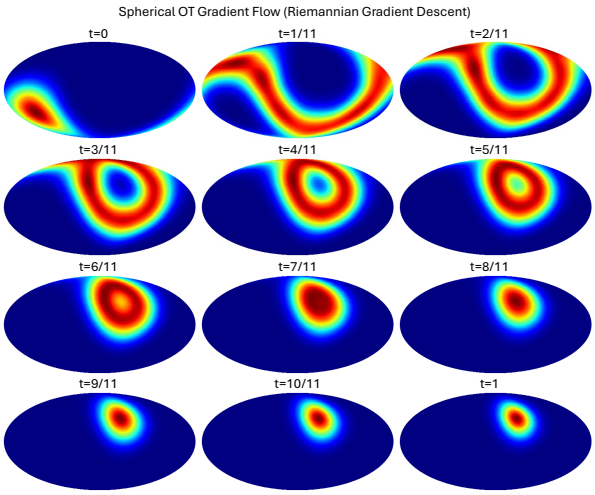

Figure 21: Spherical OT gradient flow using Riemannian gradient descent.

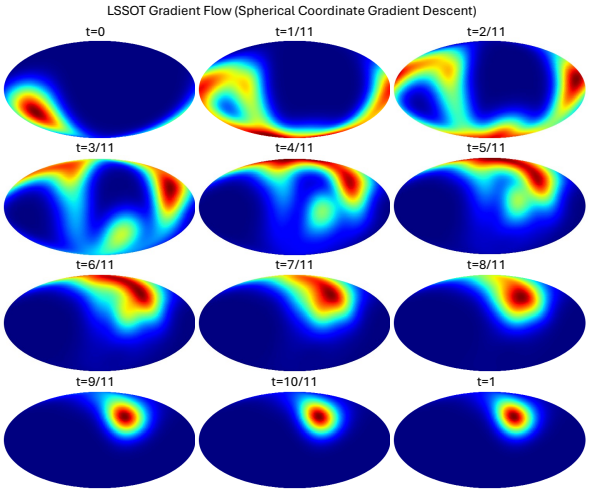

Figure 22: LSSOT gradient flow using spherical coordinate gradient descent.

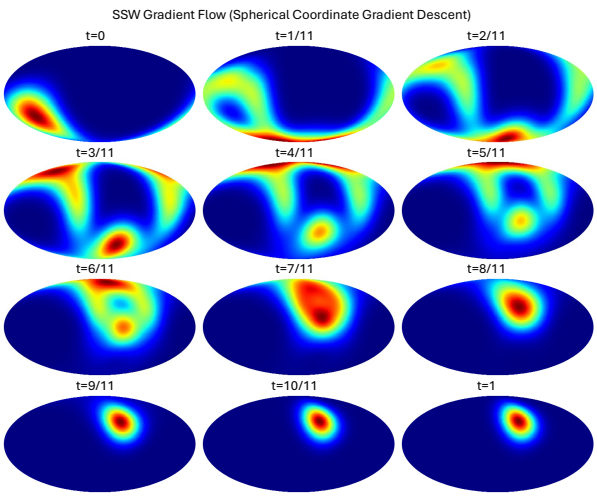

Figure 23: SSW gradient flow using spherical coordinate gradient descent.

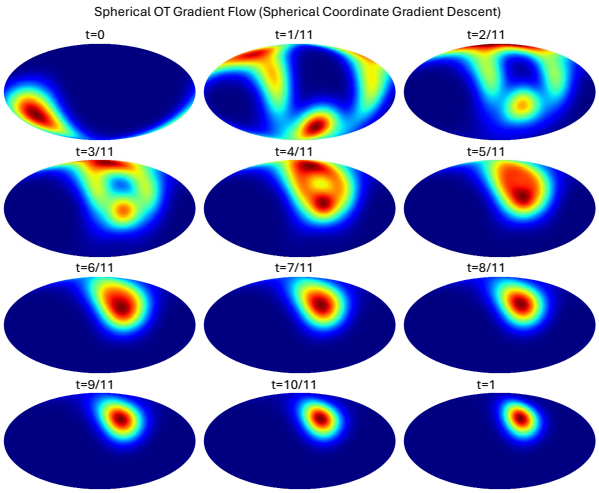

Figure 24: Spherical OT gradient flow using spherical coordinate gradient descent.

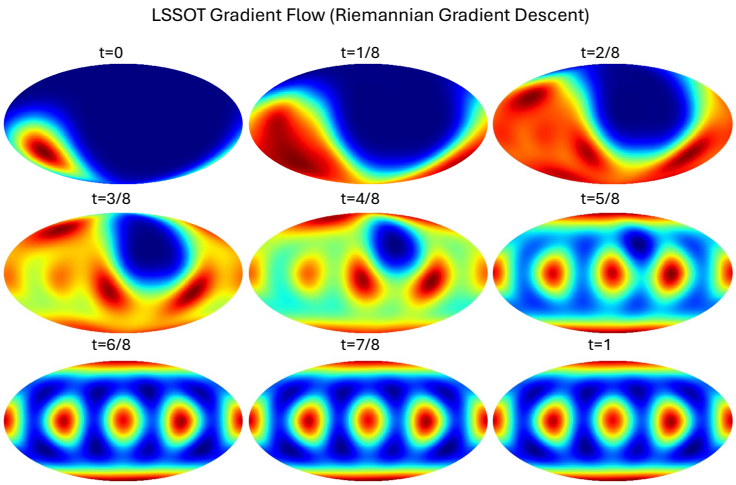

Figure 25: LSSOT gradient flow using Riemannian gradient descent.

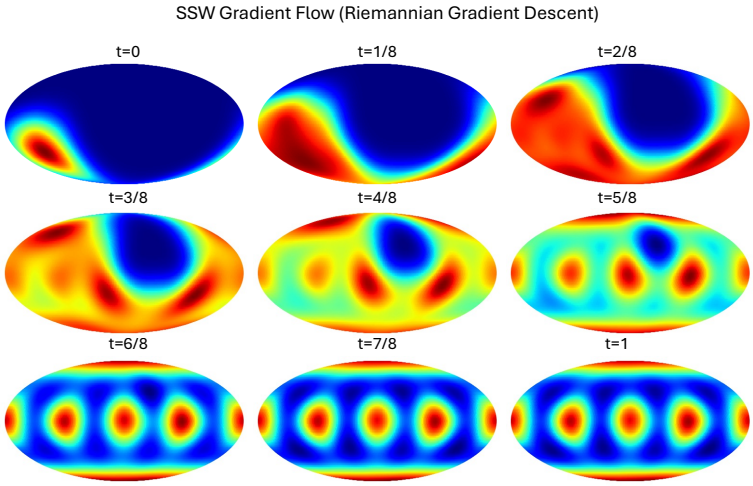

Figure 26: SSW gradient flow using Riemannian gradient descent.

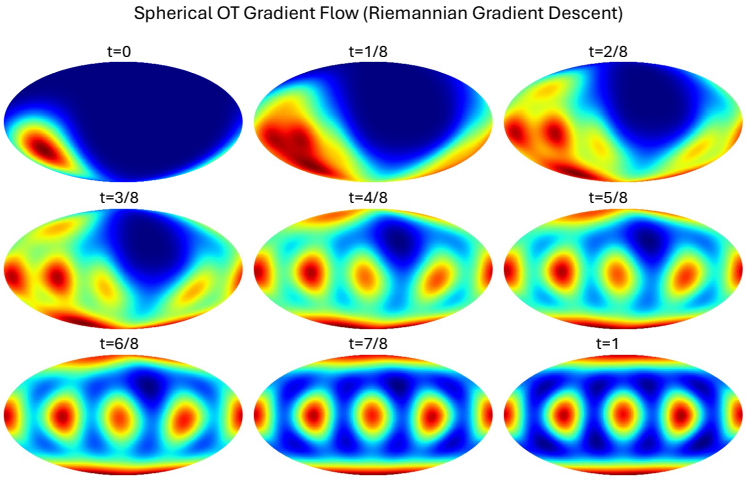

Figure 27: Spherical OT gradient flow using Riemannian gradient descent.

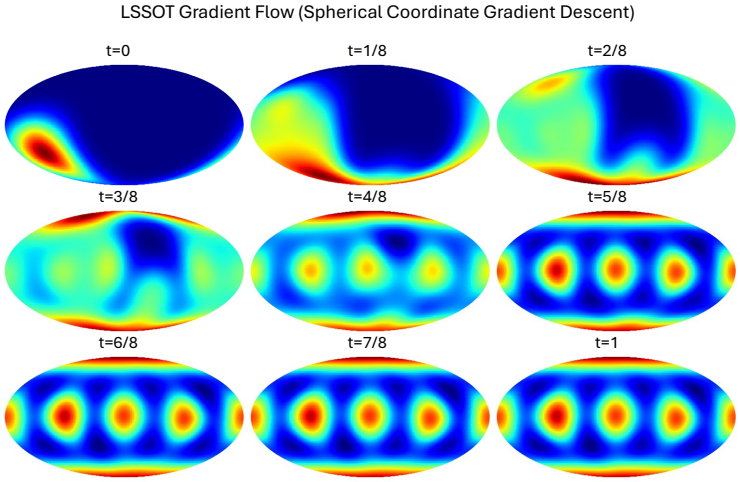

Figure 28: LSSOT gradient flow using spherical coordinate gradient descent.

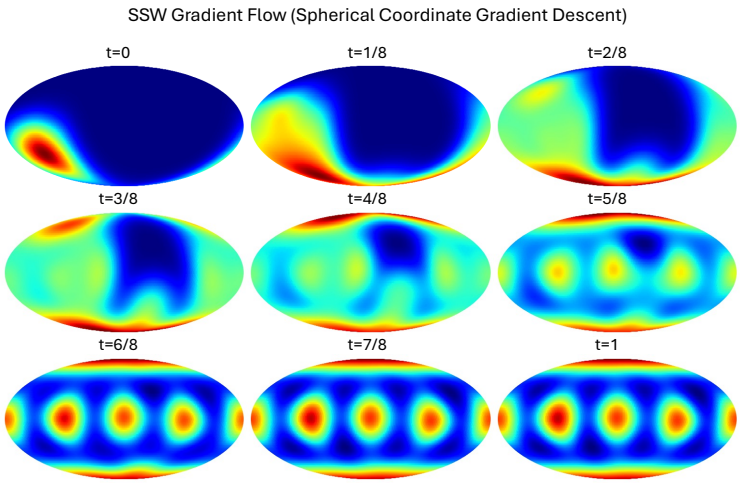

Figure 29: SSW gradient flow using spherical coordinate gradient descent.

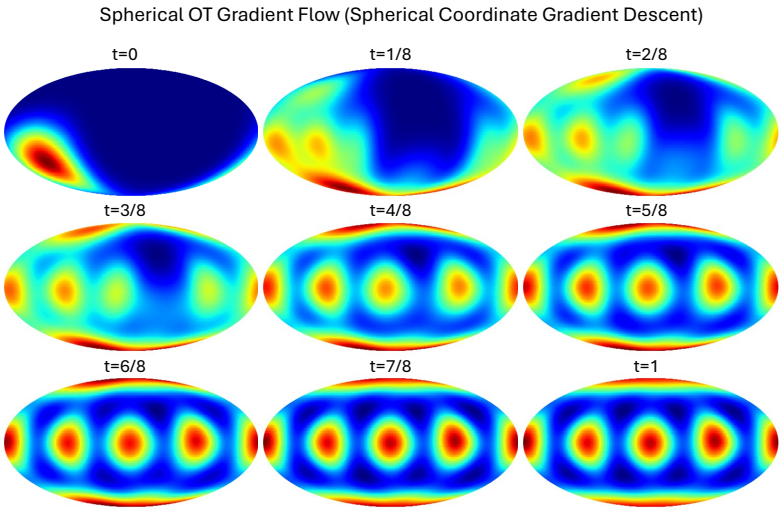

Figure 30: Spherical OT gradient flow using spherical coordinate gradient descent.

J Computation Complexity

We provide the complexities for d-dimensional discrete distributions with support of size N:

Method	Complexity	Pairwise Distance Calculation Complexity
LSSOT	$\mathcal{O}(LN(d+\log N+1))$	$\mathcal{O}(KLN(d+\log N+1+K))$
SSW	$\mathcal{O}(LN(d+\log N+\log(\frac{1}{\epsilon})))$	$\mathcal{O}(K^2LN(d+\log N+\log(\frac{1}{\epsilon})))$
OT	$\mathcal{O}(N^2(N\log N+d))$	$\mathcal{O}(K^2N^2(N\log N+d)$
Sinkhorn	$\mathcal{O}(N^2(\frac{\log N}{\epsilon^2}+d))$	$\mathcal{O}(K^2N^2(\frac{\log N}{\epsilon^2}+d))$
S3W	$\mathcal{O}(LN(d+\log N))$	$\mathcal{O}(K^2LN(d+\log N))$
SWD	$\mathcal{O}(LN(d+\log N))$	$\mathcal{O}(K^2LN(d+\log N))$

Table 2: Computation complexities for one pair of distributions (middle column) and for pairwise distance computation among a set of K distributions. The ϵ in the complexity of SSW arises from approximating the solution with precision ϵ in binary search. Similarly, the ϵ in the complexity of Sinkhorn refers to finding an ϵ-approximation of the OT distance by Sinkhorn's algorithm.

Of note, all methods except SWD are under the spherical geometry, whereas the underlying ground distance for SWD is the Euclidean distance. For one pair of distributions, SWD and S3W are marginally faster than LSSOT. However, when K is large enough in the pairwise comparison, the linear complexity with respect to K will overrule that slight advantage (as shown in Figure 2), making LSSOT the fastest of all the methods.

K Influence of Number of Projections and Reference Size

As the number of slices and reference size are the hyperparameters of the LSSOT method, we plot the evolution of LSSOT with respect to the number of slices in Figure 31 and the evolution with respect to reference size in Figure 32 for different dimensions d and VMF distributions with different κ. With more than 100 slices, LSSOT performs robustly as the projector varies. LSSOT also behaves consistently as reference size changes.

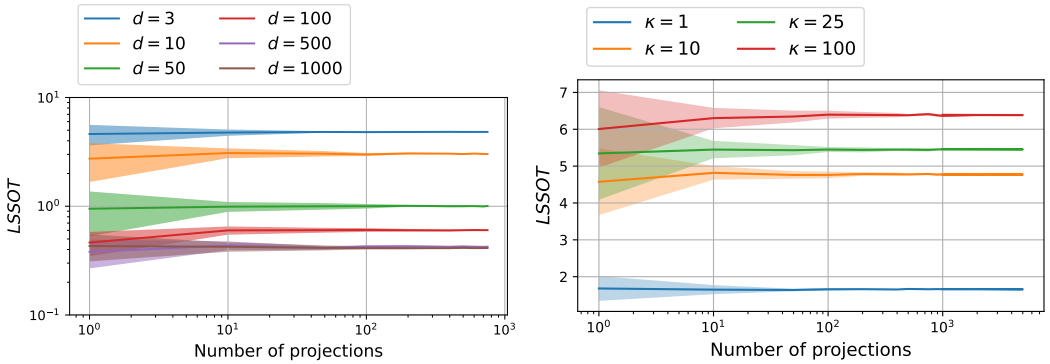

Figure 31: Evolution of LSSOT with respect to number of slices. The sample sizes are fixed to be 500.

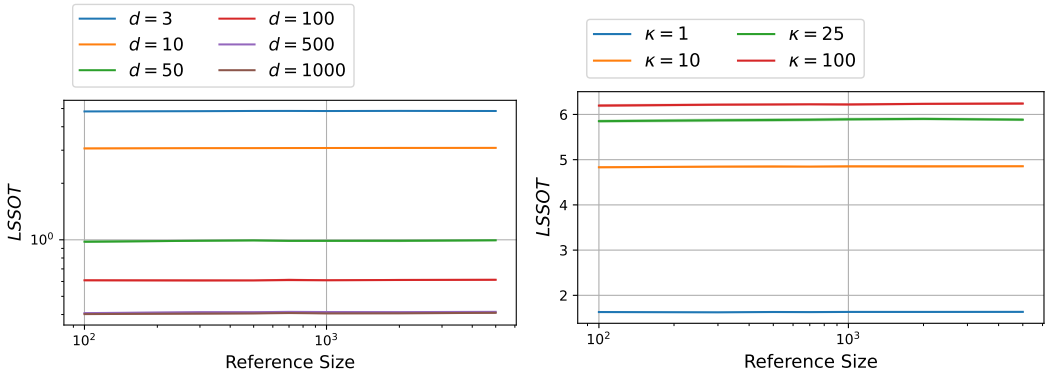

Figure 32: Evolution of LSSOT with respect to reference size. Reference is sampled from the uniform distribution on $[0, 1)$.

L Invariance to Rotations

By definition of LSSOT as an integration over the uniform distribution σ (Equation 10), it is invariant to rotations. To empirically test the invariance, we fix a pair of distributions and generate 20 random rotations that are applied on the distributions. The LSSOT distances for all 20 rotated distributions are plotted in Figure 33, with mean and standard deviation indicated. All 20 values fall closely with each other, with slight deviation due to the discrete approximation of the uniform distribution over the rank-2 projectors.

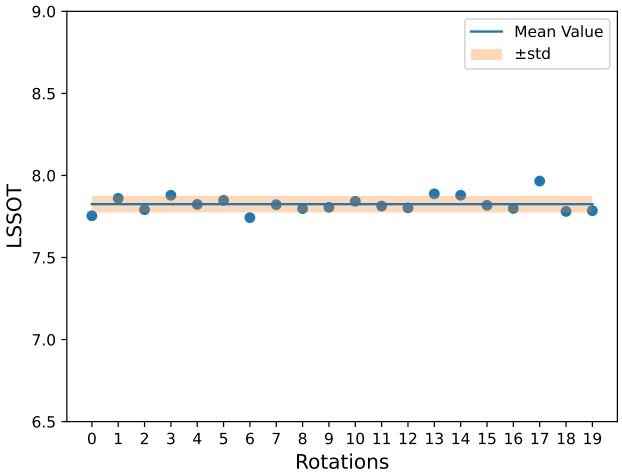

Figure 33: LSSOT distances for a pair of distributions and their rotated variants.

M IDENTIFYING INCORRECTLY LABELED SUBJECTS

With metric properties, LSSOT can be applied to identify the same distribution. One scenario in neuroimaging analysis is when the same subject is labeled as different subjects in the datasets. We design a simplified experiment to demonstrate the effectiveness of LSSOT in identifying the same subjects in the ADNI dataset. We randomly select 25 subjects from the ADNI dataset and duplicate one of them (the 8-th subject) as the test subject with an incorrect label. Figure 34 depicts that e^{LSSOT} stands out at subject 8 with value 1, as the LSSOT distance between them is arbitrarily close to 0. Such behavior makes LSSOT a very convenient tool in correcting the labels in the preprocessing of the datasets.

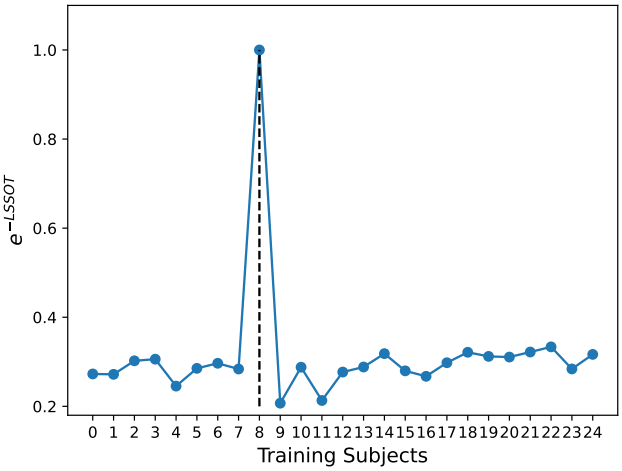

Figure 34: Exponential of the negative LSSOT distance from a test subject to a set of 25 training subjects.

N COMPARING WITH FREESURFER REGISTRATION

		Freesurfer	MSE	S3W	SWD	SSW	LSSOT (Ours)
NKI (Left Hemisphere)	LSSOT(↓)	0.2877±0.0392	0.2754±0.0611	0.2298±0.0346	0.2411± 0.0366	**0.2079±0.0369**	**0.1890±0.0361**
	SWD(↓)	0.0060±0.0020	**0.0051±0.0017**	0.0053±0.0010	**0.0027±0.0011**	0.0059±0.0011	0.0052±0.0011
	MAE(↓)	**0.1053±0.0214**	**0.1129±0.0471**	0.2278±0.0372	0.2658±0.0410	0.2145±0.0266	0.2516±0.0397
	CC(↑)	0.8190±0.0264	0.8269±0.0425	**0.9216±0.0174**	**0.8722±0.0302**	0.8671±0.0314	0.8649±0.0295
	Dice (↑)	0.7541±0.0651**	0.7746±0.0861**	**0.8498±0.0670****	0.7984±0.0539**	0.8429±0.0692*	**0.8462±0.0548**
	Edge Dist.(↓)	0.3207±0.0436**	0.3060±0.0404**	0.3922±0.0647**	0.3442±0.0346**	**0.2476±0.0348****	**0.2365±0.0343**
	Area Dist.(↓)	0.4652±0.0698**	0.4305±0.0488**	0.4048±0.0752**	0.4073±0.0368**	**0.2733±0.0402****	**0.2897±0.0396**
	Time(seconds)(↓)	–	**73.07**	121.00	118.96	1350.96	**101.01**

Table 3: Updated table for left hemispheres of NKI dataset after adding Freesurfer as a baseline. Significance test is performed for the difference on Dice, Edge Dist. and Area Dist. against LSSOT. ** denotes $p < 0.001$ and * denotes $p < 0.01$.