LIGHT ENTROPIC OPTIMAL TRANSPORT ON RIEMANNIAN MANIFOLDS

Anonymous authors
Paper under double-blind review

ABSTRACT

Entropic Optimal Transport (EOT) has become a practical framework for learning stochastic couplings between complex distributions, with applications in generative modeling and domain adaptation. However, most EOT solvers are designed for Euclidean spaces, while manifold extensions remain limited and often rely on costly iterative methods, simulated dynamics, or generic neural models that do not fully exploit the underlying geometry. We introduce ManifoldLightOT, a light approach for learning kernel-induced EOT couplings directly on common manifolds. Using the kernel form of the EOT solution, we construct geometry-specific Gibbs kernels together with compatible potential parameterizations for spheres, tori, $\mathrm{SO}(3)$, and $\mathrm{SE}(3)$. These choices yield closed-form normalization and directly sampleable conditional distributions. Our formulation naturally extends to products of manifolds, making it applicable to more complex geometries. The parameters of the potentials are optimized directly from samples using Monte Carlo estimates of the learning objective. Through synthetic and real-world experiments, we show that ManifoldLightOT often outperforms existing manifold OT methods while retaining direct sampling.

1 INTRODUCTION

In recent years, developing computationally efficient solvers for Optimal Transport (OT) and Entropic Optimal Transport (EOT) problems (Cuturi, 2013; Villani, 2008) has attracted considerable interest. OT seeks a coupling between prescribed source and target distributions that minimizes an expected transportation cost, while EOT augments this objective with an entropic regularization term. A well-established class of solvers operates on **discrete** probability distributions, representing the coupling as a matrix whose dimensions depend on the numbers of source and target samples; see Peyré et al. (2019) for a survey. In this work, we focus on **continuous** solvers (Gushchin et al., 2023b; Korotin et al., 2021), which learn conditional distributions or transport maps (plans), typically parameterized by neural networks, from samples of the source and target distributions. The EOT formulation is equivalent to the static Schrödinger bridge (SB) problem (Schrödinger, 1931; 1932; Léonard, 2013) when the transport cost is induced by the reference measure of the SB.

Several neural approaches have been proposed in recent years to solve the EOT or SB problems (Mokrov et al., 2024; Bortoli et al., 2023; Tong et al., 2024; Gushchin et al., 2023a; Korotin et al., 2024; Persiianov et al., 2026). These solvers support out-of-sample estimation, enabling applications such as image generation, image-to-image translation, domain adaptation, and biological tasks including modeling cell dynamics and predicting cellular responses to perturbations. However, these methods are formulated in Euclidean spaces, where source and target samples are elements of $\mathbb{R}^D$. However, these methods may not be directly adjusted when data are supported on curved manifolds.

Switching on manifold support has proved effective across several generative modeling frameworks, including flow matching (Chen & Lipman, 2024), continuous normalizing flows (Mathieu & Nickel, 2020), and score-based diffusion models (Bortoli et al., 2022). Manifold-aware formulations have also been developed for neural OT and EOT, notably Riemannian Neural OT (RNOT) (Micheli et al., 2026a) and Entropic Riemannian Neural OT (ERNOT) (Micheli et al., 2026b), respectively. In practice, RNOT approximates the c-transform through an additional inner optimization, while ERNOT relies on minibatch or finite-support approximations of the soft c-transform and the induced Gibbs conditionals. These approximations can introduce additional optimization error, sampling bias, and

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

sensitivity to the chosen target support in high-dimensional settings. Motivated by these observations, we propose ManifoldLightOT, an EOT solver with geometry-specific potential parameterizations for several classes of Riemannian manifolds. Following the terminology of LightSB (Korotin et al., 2024), we use the word "light" to denote parameterizations that yield closed-form expressions for the normalizing term and conditional distribution of the EOT coupling.

Our **main contribution** is ManifoldLightOT, a light solver for kernel-induced EOT problems on spheres, tori, $\mathrm{SO}(3)$, and $\mathrm{SE}(3)$. The method combines geometry-specific Gibbs kernels with compatible potential families, yielding closed-form normalization functions and directly sampleable conditional distributions. We further show that these tractability properties are preserved under Cartesian products, extending the construction to heterogeneous product manifolds. The resulting solver is trained using Monte Carlo estimates of the learning objective from unpaired source and target samples, without additional inner optimization problems.

We evaluate ManifoldLightOT on synthetic transport problems across the considered manifolds and on real-world manifold-valued data, comparing its performance with other existing OT methods.

Notation. Let $\mathcal{M}$ denote the underlying manifold, m its Riemannian volume measure, which reduces to the standard Lebesgue measure when $\mathcal{M} = \mathbb{R}^d$, and $\mathcal{P}(\mathcal{M})$ the set of probability measures on $\mathcal{M}$. We denote the source and target measures by $\mu, \nu \in \mathcal{P}(\mathcal{M})$, assumed to be absolutely continuous with respect to m, and their corresponding points by $x, y \in \mathcal{M}$. We use $\mu(x)$ and $\nu(y)$ to denote their densities with respect to m. For a coupling $q \in \Pi(\mu, \nu)$, we denote its density with respect to $m \otimes m$ by $q(x, y)$, and its conditional density by $q(y|x)$. Accordingly, $q(x, y) = \mu(x)q(y|x)$, the marginal constraints require $\int_{\mathcal{M}} q(x, y)\, dm(y) = \mu(x)$ and $\int_{\mathcal{M}} q(x, y)\, dm(x) = \nu(y)$.

2 Background

In this section, we recall the main concepts underlying our approach. We first review the formulation of entropic optimal transport on manifolds and the kernel characterization of its optimal coupling. We then recall its connection to the static Schrödinger bridge problem. Finally, we review the LightSB solver (Korotin et al., 2024) as a representative light approach for parameterizing the optimal coupling, while noting that its construction is restricted to Euclidean spaces.

Entropic Optimal Transport on Riemannian Manifolds. We consider the Entropic Optimal Transport (EOT) problem directly on a Riemannian manifold $\mathcal{M}$. For simplicity, throughout the paper we refer to this manifold formulation simply as EOT. Let $\mu, \nu \in \mathcal{P}(\mathcal{M})$ be the source and target probability measures, respectively, and let $\Pi(\mu, \nu)$ denote the set of couplings on $\mathcal{M} \times \mathcal{M}$ with marginals μ and ν. Given a transport cost $c : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ and a regularization parameter $\varepsilon > 0$, the EOT problem (Cuturi, 2013; Genevay, 2019) is defined as

$$q^* \in \arg\min_{q \in \Pi(\mu,\nu)} \left\{ \mathbb{E}_{(x,y)\sim q} \left[c(x, y) \right] - \varepsilon \mathrm{H}(q) \right\}. \tag{1}$$

Here, $\mathrm{H}(q)$ denotes the entropy of the coupling density $q(x, y)$ with respect to the product Riemannian volume measure $m \otimes m$. We assume the usual well-posedness conditions for continuous EOT, including absolute continuity, finite entropy, and finite expected cost. For $\mathrm{SE}(3)$, we also assume finite translational second moments. While many manifold OT methods use the squared geodesic cost $c(x, y) = \frac{1}{2} d^2_{\mathcal{M}}(x, y)$, our formulation uses geometry-specific kernel-induced costs.

Kernel characterization of the EOT solution. The solution q^* of the EOT problem in equation 1 admits a Gibbs kernel representation (Genevay, 2019; Peyré et al., 2019). In particular, for a given cost c and regularization parameter ε, the associated Gibbs kernel is defined as $K_\varepsilon(x, y) = \exp\left(-c(x, y)/\varepsilon\right)$. With this kernel, the density of the optimal EOT coupling with respect to the product Riemannian volume measure $m \otimes m$ admits the following form

$$q^*(x, y) = u^*(x) K_\varepsilon(x, y) v^*(y), \tag{2}$$

where u^* and v^* are positive scaling functions chosen such that q^* satisfies the marginal constraints μ and ν. Using the source marginal constraint, the optimal coupling in equation 2 can equivalently be expressed through its conditional distribution as

$$q^*(y|x) = \frac{1}{r^*(x)} K_\varepsilon(x, y) v^*(y), \tag{3}$$

where $r^*(x) = \int_{\mathcal{M}} K_\varepsilon(x, y)v^*(y)\, dm(y)$ is the corresponding normalization function. The EOT problem in equation 1 is equivalent to the static Schrödinger bridge problem (Schrödinger, 1931; 1932; Léonard, 2013) when the transport cost is induced by the reference measure from the SB.

Light Parameterizations of the EOT Solution. Under the equivalence described above, EOT and static SB share the same optimal coupling q^*. Following Light Schrödinger Bridge (LightSB) (Korotin et al., 2024), a parametric coupling $q_\theta \approx q^*$ is learned by minimizing the Kullback-Leibler (KL) divergence $\mathrm{KL}(q^* \,\|\, q_\theta)$, which is equivalent up to constants to minimizing the surrogate loss $\mathbb{E}_{x\sim\mu}[\log r_\theta(x)] - \mathbb{E}_{y\sim\nu}[\log v_\theta(y)]$. Therefore, learning the conditional EOT solution in equation 3 can be approached by choosing a suitable parametric scaling function v_θ. In the Euclidean setting, LightSB uses a light parameterization for v_θ as a learnable weighted mixture of Gaussian densities,

$$v_\theta(y) = \sum_{k=1}^{K} \alpha_k \, \mathcal{N}(y|\mu_k, \varepsilon S_k).$$

Here, the optimizable parameters are $\theta = \{\alpha_k, \mu_k, S_k\}_{k=1}^K$, where $\alpha_k > 0$ are mixture weights, $\mu_k \in \mathbb{R}^d$ are component means, and $S_k \in \mathbb{R}^{d\times d}$ are positive-definite covariance parameters. This parameterization leads to closed-form expressions for the normalizing function $r_\theta(x)$ and the conditional coupling $q_\theta(y|x)$. Together with the surrogate loss above, these closed forms enable training from independent samples of μ and ν, as well as direct sampling from $q_\theta(y|x)$. However, this construction is based on Gaussian distributions and Euclidean identities, so it is not directly applicable to manifold-supported distributions.

2.1 Continuous EOT Learning Setup

Although the EOT problem in equation 1 is formulated in terms of the source and target measures μ and ν, in practical applications these distributions are typically not available explicitly but only through empirical samples. Below we formalize the continuous EOT setup considered in our work.

> We assume that the learner is given empirical datasets $\{x^{(i)}\}_{i=1}^N$ and $\{y^{(j)}\}_{j=1}^M$, with $x^{(i)}, y^{(j)} \in \mathcal{M}$, consisting of i.i.d. samples from unknown source and target measures $\mu, \nu \in \mathcal{P}(\mathcal{M})$, respectively. The two datasets are assumed to be independent and unpaired. The task is to use these samples to learn an approximation of the EOT solution $q^* \in \Pi(\mu, \nu)$ in equation 1, and in particular its conditional distribution $q^*(y|x)$.

This setup corresponds to continuous EOT learning from empirical samples and should be distinguished from discrete EOT. Discrete EOT solvers, such as Sinkhorn (Cuturi, 2013) and gradient-based methods (Dvurechensky et al., 2018), treat the samples as empirical measures, $\widehat{\mu}_N = \frac{1}{N}\sum_{i=1}^N \delta_{x^{(i)}}$ and $\widehat{\nu}_M = \frac{1}{M}\sum_{j=1}^M \delta_{y^{(j)}}$, and return an $N \times M$ coupling matrix. Such a solution does not define transport for unseen source points. In contrast, we use the samples as training data to approximate the EOT coupling between measures μ and ν, yielding an out-of-sample model from which y^{new} can be generated according to an approximation of $q^*(y|x^{\text{new}})$ for $x^{\text{new}} \sim \mu$.

Several works extend empirical OT or EOT solutions to unseen samples through map estimation, barycentric projection, or smoothing (Hütter & Rigollet, 2021; Pooladian & Niles-Weed, 2021; Deb et al., 2021; Manole et al., 2024; Makkuva et al., 2020). However, they are specialized to quadratic Euclidean costs and rely on structures such as Brenier maps, Euclidean barycenters, or linear smoothers, which do not directly extend to our general kernel-induced costs on manifolds. Although SB provides an equivalent formulation of EOT under a suitable reference process (Léonard, 2013), but existing continuous SB solvers are typically tied to Euclidean reference processes and kernels (Bortoli et al., 2023; Gushchin et al., 2023a; Tong et al., 2024; Korotin et al., 2024). Adapting them to our geometry-specific K_ε would require additional manifold-specific modifications, so we do not include them as direct baselines. We review general-cost continuous EOT solvers in §4.

3 Manifold Entropic Optimal Transport Solver

The design of ManifoldLightOT is motivated by the observation that tractability (parameterized closed-form solution, direct sampling) in continuous EOT depends on the compatibility between the Gibbs kernel and its associated probability family. In the Euclidean setting, LightSB (Korotin et al., 2024) exploits this compatibility for the quadratic cost through Gaussian kernels and Gaussian

mixtures. We adopt this insight as a design criterion for manifold-valued transport. Our method specifies the EOT problem through geometry-aware Gibbs kernels. Each kernel K_ε induces the corresponding cost $c_\varepsilon(x, y) = -\varepsilon \log K_\varepsilon(x, y)$, up to additive constants. The goal is therefore not to approximate squared-geodesic EOT on every manifold, but to construct tractable kernel-induced EOT problems whose associated probability families admit closed-form normalization and conditional sampling. We apply this principle for spheres, tori, $\mathrm{SO}(3)$, $\mathrm{SE}(3)$, and their Cartesian products. We derive the learning objective in §3.1, introduce the manifold-specific constructions in §3.2, and describe training and sampling in §3.3.

3.1 Learning Objective

We first describe how the parametric coupling q_θ is induced. Given a positive scaling function v_θ, with finite-dimensional parameters θ, we define $r_\theta(x) = \int_{\mathcal{M}} K_\varepsilon(x, y)v_\theta(y)\, dm(y)$ and $q_\theta(y|x) = K_\varepsilon(x, y)v_\theta(y)/r_\theta(x)$. Together with the source marginal, this conditional defines the joint coupling $q_\theta(x, y) = \mu(x)q_\theta(y|x)$. Thus, q_θ is not parameterized as an unrestricted neural coupling; it is determined by the choice of v_θ and the fixed Gibbs kernel K_ε.

Following the same LightSB principle, we learn θ by minimizing the forward KL divergence from the optimal EOT coupling q^* to the induced coupling q_θ:

$$\min_\theta \mathrm{KL}\left(q^* \,\|\, q_\theta\right), \tag{4}$$

where optimal coupling q^* has form equation 2. Although the optimal coupling q^* is unknown, the terms in equation 4 that depend on θ can be expressed using only its source and target marginals. In particular, Korotin et al. (2024, Proposition 3.1) show that equation 4 has the same minimizers as

$$\min_\theta \mathcal{L}(\theta), \qquad \mathcal{L}(\theta) \stackrel{\text{def}}{=} \mathbb{E}_{x\sim\mu}\left[\log r_\theta(x)\right] - \mathbb{E}_{y\sim\nu}\left[\log v_\theta(y)\right]. \tag{5}$$

Therefore, the objective can be estimated from independent, unpaired samples from μ and ν, without requiring access to or samples from the unknown optimal coupling q^*. Importantly, this objective remains valid on manifolds, since its derivation relies only on the marginal constraints of the optimal coupling and not on the particular geometry of the underlying state space.

3.2 Tractable Manifold-aware Parameterizations

In this subsection, we introduce geometry-specific parameterizations v_θ of the optimal scaling function v^* that yield closed-form expressions for the corresponding normalization function r_θ and conditional distribution $q_\theta(y|x)$. We consider spheres ($\mathbb{S}^{d-1}$), tori ($\mathbb{T}^d$), the rotation group ($\mathrm{SO}(3)$), the special Euclidean group ($\mathrm{SE}(3)$), and Cartesian products of these manifolds. We first state a general construction underlying all the proposed parameterizations.

Proposition 1 (Tractable light parameterizations)**.** *Let K_ε be a positive kernel on $\mathcal{M} \times \mathcal{M}$, and let $v_{\theta,k} : \mathcal{M} \to \mathbb{R}_+$ be positive potential components such that, for each k, the product $K_\varepsilon(x, y)v_{\theta,k}(y)$ admits a closed-form normalization $r_{\theta,k}(x) = \int_{\mathcal{M}} K_\varepsilon(x, y)v_{\theta,k}(y)\, dm(y)$ and the normalized conditional $q_{\theta,k}(y|x) = K_\varepsilon(x, y)v_{\theta,k}(y)/r_{\theta,k}(x)$ can be sampled directly. For the mixture parameterization $v_\theta(y) = \sum_{k=1}^K \alpha_k v_{\theta,k}(y)$, with $\alpha_k > 0$, it follows that*

$$r_\theta(x) = \sum_{k=1}^K \alpha_k r_{\theta,k}(x), \qquad q_\theta(y|x) = \sum_{k=1}^K w_{\theta,k}(x)q_{\theta,k}(y|x), \tag{6}$$

where $w_{\theta,k}(x) = \alpha_k r_{\theta,k}(x)/r_\theta(x)$. Therefore, the compatibility of K_ε and $v_{\theta,k}$ yields closed-form normalization and direct conditional sampling for q_θ.

It is important to note that the construction in Proposition 1 extends the compatibility principle behind the Gaussian-mixture adjusted-potential parameterization used in LightSB. In the Euclidean quadratic-cost setting, this tractable construction can be written using the adjusted inner-product Gibbs kernel with Gaussian-mixture adjusted potentials, i.e., $\exp(-\|x - y\|^2/(2\varepsilon)) = \exp(-\|x\|^2/(2\varepsilon)) \exp(\langle x, y\rangle/\varepsilon) \exp(-\|y\|^2/(2\varepsilon))$, so the usual Gaussian Gibbs kernel differs from the inner-product kernel only by separate factors depending on x and y. We next apply this idea to several common non-Euclidean manifolds by constructing compatible choices of K_ε, $v_{\theta,k}$, and induced costs $c(x, y)$ for which the normalization and conditional sampling conditions of Proposition 1 are satisfied in closed form. All proofs appear in Appendix A.1.

Sphere. Let $\mathcal{M} = \mathbb{S}^{d-1}$, with $x, y \in \mathbb{S}^{d-1}$. For $g \in \mathbb{R}^d$, define the von Mises-Fisher normalizing function $Z_d(g) := \int_{\mathbb{S}^{d-1}} \exp(g^\top x)\, dm(x)$, and let $\mathrm{vMF}(g)$ denote the corresponding von Mises-Fisher distribution with natural parameter g.

$$K_\varepsilon(x,y) = \exp\left(\frac{x^\top y}{\varepsilon}\right), \qquad v_{\theta,k}(y) = \exp(f_k^\top y). \tag{7}$$

$$r_{\theta,k}(x) = Z_d\Big(f_k + \frac{x}{\varepsilon}\Big), \qquad q_{\theta,k}(\cdot|x) = \mathrm{vMF}\Big(f_k + \frac{x}{\varepsilon}\Big). \tag{8}$$

The product $K_\varepsilon(x,y)v_{\theta,k}(y)$ remains in the von Mises-Fisher family, yielding the closed-form normalization. The learnable parameters are $\theta = \{\alpha_k, f_k\}_{k=1}^K$, with $\alpha_k > 0$ and $f_k \in \mathbb{R}^d$. From $K_\varepsilon(x,y) = \exp[-c(x,y)/\varepsilon]$, this kernel induces the cost $c(x,y) = -x^\top y$. Conditional samples are drawn directly from $\mathrm{vMF}(f_k + x/\varepsilon)$.

Torus. Let $\mathcal{M} = \mathbb{T}^d$, with $x, y \in [0, 2\pi)^d$ denoting their angular coordinates and x^j, y^j their j-th coordinates. Define $h_{kj}(x) := (a_{kj} + \varepsilon^{-1}\cos x^j,\ b_{kj} + \varepsilon^{-1}\sin x^j)^\top$ and $\rho_{kj}(x) := \|h_{kj}(x)\|_2$. Let I_0 be the modified Bessel function of the first kind and $\mathrm{vM}(h)$ the circular von Mises distribution with natural parameter $h \in \mathbb{R}^2$.

$$K_\varepsilon(x,y) = \prod_{j=1}^d \exp\left[\frac{\cos(x^j - y^j)}{\varepsilon}\right], \quad v_{\theta,k}(y) = \prod_{j=1}^d \exp\big[a_{kj}\cos y^j + b_{kj}\sin y^j\big]. \tag{9}$$

$$r_{\theta,k}(x) = \prod_{j=1}^d 2\pi I_0(\rho_{kj}(x)), \qquad q_{\theta,k}(\cdot|x) = \prod_{j=1}^d \mathrm{vM}(h_{kj}(x)). \tag{10}$$

The learnable parameters are $\theta = \{\alpha_k, a_k, b_k\}_{k=1}^K$, with $\alpha_k > 0$ and $a_k, b_k \in \mathbb{R}^d$. The induced cost is $c(x,y) = -\sum_{j=1}^d \cos(x^j - y^j)$, which is equivalent to $c(x,y) = \sum_{j=1}^d [1 - \cos(x^j - y^j)]$ up to an additive constant. Sampling is done independently for each coordinate from $\mathrm{vM}(h_{kj}(x))$.

Rotation group. Let $\mathcal{M} = \mathrm{SO}(3)$, with x, y represented by unit quaternions in $\mathbb{S}^3$, where x and $-x$ represent the same rotation. To preserve this antipodal symmetry, we use hyperbolic-cosine kernel and potential components. Let $\lambda_R > 0$ denote a fixed rotational kernel scale, and define $\pi_{k,s}(x) := Z_4(f_k + s\lambda_R x/\varepsilon)/[Z_4(f_k + \lambda_R x/\varepsilon) + Z_4(f_k - \lambda_R x/\varepsilon)]$ for $s \in \{-1, +1\}$.

$$K_\varepsilon(x,y) = \cosh\left(\frac{\lambda_R}{\varepsilon}\langle x, y\rangle\right), \qquad v_{\theta,k}(y) = \cosh(\langle f_k, y\rangle). \tag{11}$$

$$r_{\theta,k}(x) = \frac{1}{2}\left[Z_4\left(f_k + \frac{\lambda_R}{\varepsilon}x\right) + Z_4\left(f_k - \frac{\lambda_R}{\varepsilon}x\right)\right], \quad q_{\theta,k}(\cdot|x) = \sum_{s\in\{-1,+1\}} \pi_{k,s}(x)\,\mathrm{vMF}\left(f_k + s\frac{\lambda_R}{\varepsilon}x\right). \tag{12}$$

Using the product identity for two hyperbolic cosines yields the closed-form normalizer above. The learnable parameters are $\theta = \{\alpha_k, f_k\}_{k=1}^K$, with $\alpha_k > 0$ and $f_k \in \mathbb{R}^4$. The kernel induces the cost $c(x,y) = -\varepsilon \log\cosh(\lambda_R\langle x, y\rangle/\varepsilon)$. Conditional sampling first draws s according to $\pi_{k,s}(x)$, then samples from the corresponding von Mises-Fisher distribution and maps the quaternion to $\mathrm{SO}(3)$.

Special Euclidean group. Let $\mathcal{M} = \mathrm{SE}(3)$, and write $x = (x^t, x^R)$, where $x^t \in \mathbb{R}^3$ is the translational component and $x^R \in \mathbb{S}^3$ is the unit-quaternion representation of the rotational component. We combine an unnormalized Gaussian construction for translations with the $\mathrm{SO}(3)$ construction above. Let $K_\varepsilon^t(x^t, y^t) := \exp(-\|x^t - y^t\|_2^2/(2\sigma_\varepsilon^2))$, and let K_ε^R, $r_{\theta,k}^R$, and $q_{\theta,k}^R$ denote the rotational kernel, normalizer, and conditional from the $\mathrm{SO}(3)$ construction.

$$\begin{aligned} K_\varepsilon(x,y) &= K_\varepsilon^t(x^t, y^t) K_\varepsilon^R(x^R, y^R), \\ v_{\theta,k}(y) &= \exp\left[-\frac{1}{2}(y^t - \mu_k)^\top \Sigma_k^{-1}(y^t - \mu_k)\right]\cosh(\langle f_k, y^R\rangle). \end{aligned} \tag{13}$$

$$r_{\theta,k}(x) = r_{\theta,k}^t(x^t) r_{\theta,k}^R(x^R), \qquad q_{\theta,k}(y|x) = q_{\theta,k}^t(y^t|x^t) q_{\theta,k}^R(y^R|x^R). \tag{14}$$

The translational normalizer is $r_{\theta,k}^t(x^t) = (2\pi)^{3/2}\sigma_\varepsilon^3 \det(\Sigma_k)^{1/2} \det(\Sigma_k + \sigma_\varepsilon^2 I_3)^{-1/2} \exp[-\frac{1}{2}(x^t - \mu_k)^\top(\Sigma_k + \sigma_\varepsilon^2 I_3)^{-1}(x^t - \mu_k)]$. The translational conditional is Gaussian, $q_{\theta,k}^t(\cdot|x^t) = \mathcal{N}(\cdot|\bar{\mu}_k, A_k)$,

with $A_k = (\sigma_\varepsilon^{-2} I_3 + \Sigma_k^{-1})^{-1}$ and $\bar{\mu}_k = A_k(\sigma_\varepsilon^{-2} x^t + \Sigma_k^{-1}\mu_k)$. The learnable parameters are $\theta = \{\alpha_k, \mu_k, \Sigma_k, f_k\}_{k=1}^K$, with $\alpha_k > 0$, $\mu_k \in \mathbb{R}^3$, positive-definite $\Sigma_k \in \mathbb{R}^{3\times 3}$, and $f_k \in \mathbb{R}^4$. The product kernel induces the transport cost $c(x,y) = \varepsilon\|x^t - y^t\|_2^2/(2\sigma_\varepsilon^2) - \varepsilon \log\cosh(\lambda_R\langle x^R, y^R\rangle/\varepsilon)$. Sampling draws the translational and rotational components independently from $q^t_{\theta,k}$ and $q^R_{\theta,k}$.

Proposition 2 (Closure under Cartesian products). *Let $\mathcal{M} = \prod_{j=1}^J \mathcal{M}_j$, with product measure $m = \bigotimes_{j=1}^J m_j$. Suppose that, for each $\mathcal{M}_j$, the kernel $K_{\varepsilon,j}$ and potential component $v_{\theta,k,j}$ satisfy the tractability conditions of Proposition 1, with closed-form normalizer $r_{\theta,k,j}$ and directly sampleable conditional $q_{\theta,k,j}$. Then, the product parameterization*

$$K_\varepsilon(x,y) = \prod_{j=1}^J K_{\varepsilon,j}(x^j, y^j), \qquad v_{\theta,k}(y) = \prod_{j=1}^J v_{\theta,k,j}(y^j) \tag{15}$$

also satisfies the conditions of Proposition 1, with

$$r_{\theta,k}(x) = \prod_{j=1}^J r_{\theta,k,j}(x^j), \qquad q_{\theta,k}(y|x) = \prod_{j=1}^J q_{\theta,k,j}(y^j|x^j). \tag{16}$$

Therefore, closed-form normalization and direct conditional sampling are preserved.

Proposition 2 allows the parameterizations introduced above to be combined without additional normalization or sampling procedures. In particular, it directly yields tractable parameterizations on product spaces such as $\mathrm{SO}(3)^J$, $\mathrm{SE}(3)^J$, and heterogeneous products of the considered manifolds.

We summarize the proposed kernels, potential components, closed-form normalizers, conditional distributions, and induced transport costs in Table 1.

Table 1: Summary of the proposed manifold-aware parameterizations and induced transport costs. For $\mathrm{SO}(3)$, $x, y \in \mathbb{S}^3$ denote unit-quaternion representatives, with antipodal points identified.

$\mathcal{M}$	K_ε $v_{\theta,k}$	$r_{\theta,k}$ $q_{\theta,k}$	Induced cost $c(x,y)$
$\mathbb{S}^{d-1}$	$\exp\left(\frac{x^\top y}{\varepsilon}\right)$ $\exp(f_k^\top y)$	$Z_d\left(f_k + \frac{x}{\varepsilon}\right)$ $\mathrm{vMF}\left(f_k + \frac{x}{\varepsilon}\right)$	$-x^\top y$
$\mathbb{T}^d$	$\prod_{j=1}^d \exp\left[\frac{\cos(x^j - y^j)}{\varepsilon}\right]$ $\prod_{j=1}^d \exp[a_{kj}\cos y^j + b_{kj}\sin y^j]$	$\prod_{j=1}^d 2\pi I_0(\rho_{kj})$ $\prod_{j=1}^d \mathrm{vM}(h_{kj})$	$-\sum_{j=1}^d \cos(x^j - y^j)$
$\mathrm{SO}(3)$	$\cosh\left(\frac{\lambda_R}{\varepsilon}\langle x, y\rangle\right)$ $\cosh(\langle f_k, y\rangle)$	$\frac{Z_4\left(f_k + \frac{\lambda_R}{\varepsilon}x\right) + Z_4\left(f_k - \frac{\lambda_R}{\varepsilon}x\right)}{2}$ $\sum_{s=\pm 1} \pi_{k,s}(x)\,\mathrm{vMF}\left(f_k + s\frac{\lambda_R}{\varepsilon}x\right)$	$-\varepsilon \log\cosh\left(\frac{\lambda_R}{\varepsilon}\langle x, y\rangle\right)$
$\mathrm{SE}(3)$	$K_\varepsilon^t K_\varepsilon^R$ $\exp\left[-\frac{1}{2}(y^t - \mu_k)^\top \Sigma_k^{-1}(y^t - \mu_k)\right]$ $\times \cosh(\langle f_k, y^R\rangle)$	$r^t_{\theta,k}(x^t) r^R_{\theta,k}(x^R)$ $q^t_{\theta,k}(y^t|x^t) q^R_{\theta,k}(y^R|x^R)$	$\frac{\varepsilon}{2\sigma_\varepsilon^2}\|x^t - y^t\|_2^2$ $-\varepsilon \log\cosh\left(\frac{\lambda_R}{\varepsilon}\langle x^R, y^R\rangle\right)$

3.3 Training and Sampling

Training. Training the proposed parameterizations requires only optimizing their learnable parameters θ using the objective in equation 5. Since both $v_\theta(y)$ and $r_\theta(x)$ are available in closed form for all the constructions introduced above, the loss can be estimated from independent minibatches $\{x^{(i)}\}_{i=1}^{B_0} \sim \mu$ and $\{y^{(j)}\}_{j=1}^{B_1} \sim \nu$ as

$$\widehat{\mathcal{L}}(\theta) = -\frac{1}{B_1}\sum_{j=1}^{B_1} \log v_\theta\left(y^{(j)}\right) + \frac{1}{B_0}\sum_{i=1}^{B_0} \log r_\theta\left(x^{(i)}\right). \tag{17}$$

This estimate can be optimized using standard gradient-based methods and requires neither paired samples, intermediate trajectory simulation, nor additional inner optimization problems.

Sampling. Given a source point x and trained parameters θ, samples from $q_\theta(\cdot|x)$ are generated using the geometry-specific samplers described in the corresponding constructions of §3.2. Algorithm 1 summarizes the general sampling procedure.

Algorithm 1 Sampling from the learned EOT coupling

Require: Source point x, trained parameters θ
for $k = 1, \dots, K$ **do**
  Compute $r_{\theta,k}(x)$
  $\widetilde{w}_k \leftarrow \alpha_k r_{\theta,k}(x)$
end for
$w_k \leftarrow \widetilde{w}_k / \sum_{j=1}^{K} \widetilde{w}_j$, for $k = 1, \dots, K$
Sample $k \sim \mathrm{Categorical}(w_1, \dots, w_K)$
Sample $y \sim q_{\theta,k}(\cdot|x)$ using the corresponding manifold-specific sampler
return y

4 Related Works

In this section, we review two categories of continuous optimal transport methods most relevant to our setting. We first discuss general continuous OT and EOT solvers in §4.1, which learn transport maps or coupling representations from samples and support out-of-sample inference. We then consider Riemannian-aware OT solvers in §4.2, which explicitly incorporate the geometry of the underlying manifold into the formulation.

4.1 Continous OT Solvers

Continuous neural OT methods provide a flexible framework for learning transport directly from samples. In Seguy et al. (2018) develop a stochastic optimization approach for large-scale regularized OT, while Daniels et al. (2025) use score-based models to sample from regularized transport couplings. Neural Optimal Transport (NeuralOT) (Korotin et al., 2023) introduces a minimax formulation for learning transport maps and plans under strong and weak transport costs. This framework is further generalized by NeuralOT with General Cost Functionals (Asadulaev et al., 2024), which accommodates a broader class of transport objectives beyond standard pointwise costs. For entropy-regularized OT, Energy-guided Entropic NOT (Mokrov et al., 2024) parameterizes conditional transport plans using energy-based models and supports general continuous ground costs. For comparison, we consider NeuralOT as a representative general continuous OT baseline.

4.2 Continous Riemmanian OT Solvers

Riemannian Neural Optimal Transport (RNOT) (Micheli et al., 2026a) learns continuous optimal transport maps directly on Riemannian manifolds for the squared geodesic cost. It parameterizes a continuous potential whose c-transform enforces c-concavity and constructs the transport map $T(x)$, allowing the learned map to transport previously unseen source samples without discretizing the manifold. Entropic Riemannian Neural Optimal Transport (ERNOT) (Micheli et al., 2026b) solves entropically regularized OT on Riemannian manifolds by learning a neural Schrödinger potential that defines the transport coupling. For each source point, this coupling gives a conditional distribution over target points. When a single transported point is required, ERNOT summarizes this distribution using manifold-aware constructions, such as its Riemannian barycenter or the mode of a heat-smoothed version of the conditional distribution.

5 Experimental Results

In this section, we evaluate our method against the manifold-aware solvers introduced in §4. We consider synthetic transport problems on $\mathbb{S}^{d-1}$, $\mathbb{T}^d$, $\mathrm{SO}(3)$, and $\mathrm{SE}(3)$, together with real-world applications including continental drift on $\mathbb{S}^2$ and crystallographic texture modeling on $\mathrm{SO}(3)$. Across all experiments, performance is measured using manifold-aware metrics comparing transported source samples with samples from the target distribution. Further details on dataset construction and metric computation are provided in Appendices A.2 and A.3, respectively. All RNOT and ERNOT

experiments use the squared geodesic cost $c(x, y) = d_{\mathcal{M}}(x, y)^2$, whereas NeuralOT uses the corresponding manifold-specific induced cost; the full cost expressions are given in Table 1. This protocol tests how each approach performs as a practical continuous transport solver on the same source and target distributions, while making the cost choice of each method explicit.

Synthetic Experiments. For the synthetic experiments, source and target samples are independently drawn from known distributions defined directly on each manifold. We first consider high-dimensional spheres and tori. On $\mathbb{S}^{d-1}$, the source is uniform on the sphere and the target is obtained by mapping Gaussian tangent-space samples around a fixed reference point through the exponential map. On $\mathbb{T}^d$, the source is uniform in angular coordinates, while the target is obtained from a Gaussian distribution in angular coordinates with periodic wrapping to $[0, 2\pi)^d$. We additionally consider setups on $\mathrm{SO}(3)$ and $\mathrm{SE}(3)$. On $\mathrm{SO}(3)$, source and target distributions are generated by mapping Gaussian samples from the Lie algebra through the exponential map, with different center rotations. On $\mathrm{SE}(3)$, the source combines Haar-uniform rotations with uniformly distributed translations, while the target combines concentrated rotations around a reference orientation with Gaussian translations. We report the squared Maximum Mean Discrepancy (MMD) Table 2. Additional results for high dimensional spheres and tori can be found in Table 4 in Appendix A.5.

Table 2: MMD^2 results across different manifolds, reported in units of 10^{-2}. We report mean $\pm$ standard deviation over five seeds. Lower is better.

Method	$\mathbb{S}^{60}$	$\mathbb{S}^{100}$	$\mathbb{T}^{60}$	$\mathbb{T}^{100}$	$\mathrm{SO}(3)$	$\mathrm{SE}(3)$
ManifoldLightOT (ours)	$\mathbf{0.07 \pm 0.00}$	0.45 ± 0.05	$\mathbf{0.08 \pm 0.00}$	$\mathbf{0.08 \pm 0.00}$	$\mathbf{0.03 \pm 0.01}$	$\mathbf{0.18 \pm 0.01}$
RNOT	0.18 ± 0.04	1.47 ± 0.21	2.15 ± 0.20	23.15 ± 0.57	0.61 ± 0.27	0.23 ± 0.04
ERNOT	0.22 ± 0.01	$\mathbf{0.32 \pm 0.09}$	0.27 ± 0.02	0.27 ± 0.03	0.47 ± 0.09	5.54 ± 0.72
NeuralOT	0.27 ± 0.06	4.38 ± 8.24	61.59 ± 0.06	61.61 ± 0.07	1.27 ± 2.58	0.27 ± 0.05

Continental Drift ($\mathbb{S}^2$). Following the continental-drift benchmark introduced by Cohen et al. (2021) and later considered by RNOT (Micheli et al., 2026a), we construct source and target distributions from continental boundaries at 150 million years ago and the present day, respectively, using paleogeographic reconstructions from GPlates (Müller et al., 2018). Points are sampled uniformly with respect to spherical area on $\mathbb{S}^2$ and retained when their longitude-latitude coordinates lie within the corresponding continental polygons, yielding empirical unpaired source and target distributions. The plot of predicted points for ManifoldLightOT appears in Figure 1.

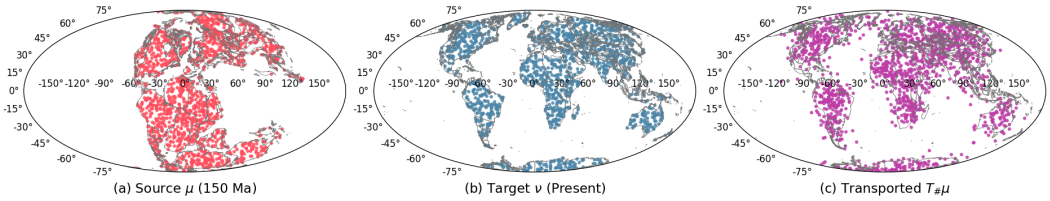

Figure 1: Prediction on continental drift from 150 Ma to the present day for ManifoldLightOT.

Crystallographic Texture Transfer. Crystallographic texture describes the distribution of crystal orientations within a polycrystalline material and plays an important role in determining its macroscopic mechanical properties. We consider the Ti-6Al-4V electron backscatter diffraction (EBSD) dataset of (Zeng et al., 2025), which characterizes how crystallographic texture evolves under different hot-rolling conditions. Hot rolling deforms a material by passing it through rollers at elevated temperature, and the reduction percentage denotes the decrease in material thickness during this process. Specifically, we formulate an unpaired transport task from the orientations observed in the starting material to those measured after rolling at 825°C with 75% reduction. We consider measurements from both the rolling-direction (RD) and transverse-direction (TD) sections, corresponding respectively to sections aligned with and perpendicular to the rolling direction. We convert the EBSD orientation measurements to unit quaternions; since each orientation represents a three-dimensional rotation, the resulting distributions are naturally supported on $\mathrm{SO}(3)$. In addition to the MMD metric used in our previous experiments, we evaluate the transported orientations using three crystallography-specific measures. The *normalized ODF difference* compares the predicted and target orientation distribution functions (ODFs), directly measuring agreement between their crystallographic textures, while the *M-index error* compares their texture strengths through

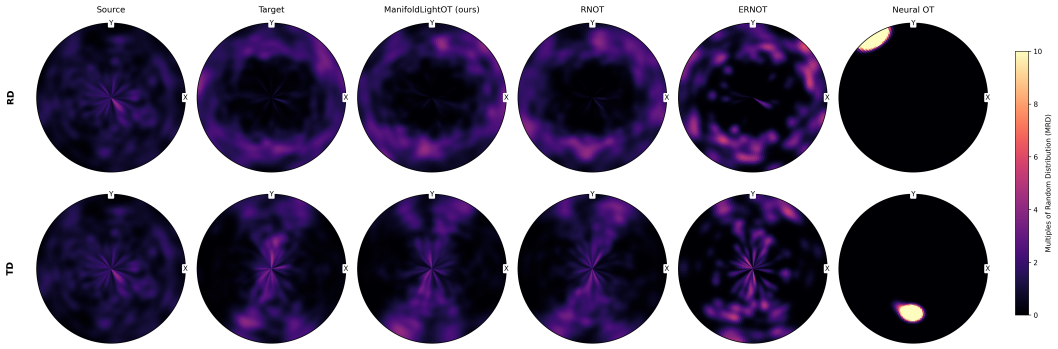

Figure 2: Pole-density plots for the $\{0001\}$ orientation family in the EBSD texture-transfer task, shown for the RD and TD conditions. For each orientation quaternion q, we take the fixed direction $h = (0, 0, 1)$, rotate it into the sample frame, project the resulting unit vector onto the two-dimensional disk using stereographic projection, and estimate the density of the projected points. Colors indicate orientation concentration in multiples of random distribution (MRD).

the symmetry-aware distribution of crystallographic misorientations. All metrics are computed separately for the RD and TD conditions and they are shown in Table 2. We also show the dataset preprocessing and complete metric definitions are provided in Appendix A.2 and A.3, respectively.

Table 3: Results on the crystallographic texture transfer task for the rolling-direction (RD) and transverse-direction (TD) conditions. Lower is better for all metrics.

Method	Direction	$\text{MMD}_{\text{SO(3)}}$	Texture Index Error	Normalized ODF Difference	M-Index Error
ManifoldLightOT (ours)	RD	**0.000592**	**0.01168**	**0.00270**	**0.00425**
	TD	0.000830	**0.00112**	**0.00265**	**0.00883**
RNOT	RD	0.000635	0.09022	0.01298	0.01304
	TD	**0.000680**	0.00441	0.01936	0.02033
ERNOT	RD	0.000901	0.11089	0.12242	0.01650
	TD	0.001056	0.13406	0.18600	0.01554
Neural OT	RD	0.482989	37.00789	36.40364	0.90857
	TD	0.997347	43.39924	45.39681	0.88277

6 Discussion

ManifoldLightOT combines strong empirical performance with a simpler training and sampling procedure than existing manifold-aware EOT solvers. The experiments show that the proposed kernel-induced EOT formulations adapt well across the considered geometries, including spheres, tori, $\text{SO}(3)$, and $\text{SE}(3)$. By using manifold-specific kernel-potential pairs, the method avoids inner optimization procedures and can be trained from Monte Carlo estimates of the learning objective.

It is important to note that the proposed pipeline, learning objective, and sampling procedure are not tied to a particular manifold. Extending ManifoldLightOT to a new geometry therefore amounts to deriving an appropriate kernel-potential pair that preserves tractable normalization and direct conditional sampling. This derivation is the main manifold-specific challenge of our formulation and can require additional analytical work. At the same time, the induced costs are not specific to ManifoldLightOT: they define valid geometry-aware EOT objectives and could, in principle, be incorporated into other continuous OT/EOT solvers that support general costs. Thus, the proposed kernels provide both a practical light solver and a set of cost functions for manifold-valued transport.

AI use statement

In this work, generative AI tools were used to assist with polishing the writing, debugging code, and formatting tables and mathematical expressions. The mathematical formulation, derivations, methodological contributions, and experimental results are original to the authors. All AI-assisted

content was reviewed and verified by the authors, who take responsibility for the final content of the work.

Ethics Statement

This work focuses on methodological developments for entropic optimal transport on manifolds and does not involve human subjects, personally identifiable information, or sensitive data. The experiments use synthetic benchmarks and publicly available scientific datasets. We are not aware of any specific ethical risks beyond those associated with the general use of machine learning methods, and we have aimed to report the methodology, experimental setup, and results transparently.

Reproducibility Statement

To ensure reproducibility, we provide the full implementation of ManifoldLightOT together with the implementations of all baseline methods used in our experiments, as well as the corresponding configuration files for every reported setting. The appendix documents the experimental setup, dataset construction and preprocessing, evaluation metrics, and additional implementation details. Together, these materials are intended to enable reproduction of the reported results and comparisons.

References

Arip Asadulaev, Alexander Korotin, Vage Egiazarian, Petr Mokrov, and Evgeny Burnaev. Neural optimal transport with general cost functionals, 2024. URL https://arxiv.org/abs/2205.15403.

Valentin De Bortoli, Emile Mathieu, Michael Hutchinson, James Thornton, Yee Whye Teh, and Arnaud Doucet. Riemannian score-based generative modelling, 2022. URL https://arxiv.org/abs/2202.02763.

Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion schrödinger bridge with applications to score-based generative modeling, 2023. URL https://arxiv.org/abs/2106.01357.

Ricky T. Q. Chen and Yaron Lipman. Flow matching on general geometries, 2024. URL https://arxiv.org/abs/2302.03660.

Samuel Cohen, Brandon Amos, and Yaron Lipman. Riemannian convex potential maps, 2021. URL https://arxiv.org/abs/2106.10272.

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

Mara Daniels, Tyler Maunu, and Paul Hand. Score-based generative neural networks for large-scale optimal transport, 2025. URL https://arxiv.org/abs/2110.03237.

Nabarun Deb, Promit Ghosal, and Bodhisattva Sen. Rates of estimation of optimal transport maps using plug-in estimators via barycentric projections. *Advances in Neural Information Processing Systems*, 34:29736–29753, 2021.

Pavel Dvurechensky, Alexander Gasnikov, and Alexey Kroshnin. Computational optimal transport: Complexity by accelerated gradient descent is better than by Sinkhorn's algorithm. In *International conference on machine learning*, pp. 1367–1376. PMLR, 2018.

Aude Genevay. *Entropy-regularized optimal transport for machine learning*. PhD thesis, Paris Sciences et Lettres (ComUE), 2019.

Nikita Gushchin, Alexander Kolesov, Alexander Korotin, Dmitry P. Vetrov, and Evgeny Burnaev. Entropic neural optimal transport via diffusion processes. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023a. URL https://openreview.net/forum?id=fHyLsfMDIs.

Nikita Gushchin, Alexander Kolesov, Petr Mokrov, Polina Karpikova, Andrey Spiridonov, Evgeny Burnaev, and Alexander Korotin. Building the bridge of schrödinger: A continuous entropic optimal transport benchmark, 2023b. URL https://arxiv.org/abs/2306.10161.

Jan-Christian Hütter and Philippe Rigollet. Minimax estimation of smooth optimal transport maps. 2021.

Alexander Korotin, Lingxiao Li, Aude Genevay, Justin Solomon, Alexander Filippov, and Evgeny Burnaev. Do neural optimal transport solvers work? a continuous wasserstein-2 benchmark, 2021. URL https://arxiv.org/abs/2106.01954.

Alexander Korotin, Daniil Selikhanovych, and Evgeny Burnaev. Neural optimal transport. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=d8CBRlWNkqH.

Alexander Korotin, Nikita Gushchin, and Evgeny Burnaev. Light Schrödinger bridge. In *The Twelfth International Conference on Learning Representations*, 2024.

Christian Léonard. A survey of the Schrödinger problem and some of its connections with optimal transport. *arXiv preprint arXiv:1308.0215*, 2013.

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.

Tudor Manole, Sivaraman Balakrishnan, Jonathan Niles-Weed, and Larry Wasserman. Plugin estimation of smooth optimal transport maps, 2024. URL https://arxiv.org/abs/2107.12364.

Emile Mathieu and Maximilian Nickel. Riemannian continuous normalizing flows, 2020. URL https://arxiv.org/abs/2006.10605.

Alessandro Micheli, Yueqi Cao, Anthea Monod, and Samir Bhatt. Riemannian neural optimal transport, 2026a. URL https://arxiv.org/abs/2602.03566.

Alessandro Micheli, Silvia Sapora, Anthea Monod, and Samir Bhatt. Entropic riemannian neural optimal transport, 2026b. URL https://arxiv.org/abs/2605.04255.

Petr Mokrov, Alexander Korotin, Alexander Kolesov, Nikita Gushchin, and Evgeny Burnaev. Energy-guided entropic neural optimal transport. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=d6tUsZeVs7.

R. Dietmar Müller, John Cannon, Xiaodong Qin, Robin J. Watson, Michael Gurnis, Simon Williams, Tobias Pfaffelmoser, Maria Seton, Samuel H. J. Russell, and Sabin Zahirovic. Gplates: Building a virtual earth through deep time. *Geochemistry, Geophysics, Geosystems*, 19(7):2243–2261, 2018. doi: https://doi.org/10.1029/2018GC007584. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2018GC007584.

Jason N. Ott, Cailey B. Condit, Vera Schulte-Pelkum, Rachel Bernard, and Matej Pec. Seismic Anisotropy of Mafic Blueschists: EBSD-Based Constraints From the Exhumed Rock Record. *Journal of Geophysical Research (Solid Earth)*, 129(2):e2023JB027679, February 2024. doi: 10.1029/2023JB027679.

Mikhail Persiianov, Jiawei Chen, Petr Mokrov, Alexander Tyurin, Evgeny Burnaev, and Alexander Korotin. Learning of population dynamics: Inverse optimization meets jko scheme, 2026. URL https://arxiv.org/abs/2506.01502.

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

Aram-Alexandre Pooladian and Jonathan Niles-Weed. Entropic estimation of optimal transport maps. *arXiv preprint arXiv:2109.12004*, 2021.

Erwin Schrödinger. *Über die Umkehrung der Naturgesetze*. Verlag der Akademie der Wissenschaften in Kommission bei Walter De Gruyter u. Company, 1931.

Erwin Schrödinger. Sur la théorie relativiste de l'électron et l'interprétation de la mécanique quantique. In *Annales de l'institut Henri Poincaré*, volume 2, pp. 269–310, 1932.

Vivien Seguy, Bharath Bhushan Damodaran, Rémi Flamary, Nicolas Courty, Antoine Rolet, and Mathieu Blondel. Large-scale optimal transport and mapping estimation, 2018. URL https://arxiv.org/abs/1711.02283.

Philip Skemer, Ikuo Katayama, Zhenting Jiang, and Shun ichiro Karato. The misorientation index: Development of a new method for calculating the strength of lattice-preferred orientation. *Tectonophysics*, 411(1):157–167, 2005. ISSN 0040-1951. doi: https://doi.org/10.1016/j.tecto.2005.08.023. URL https://www.sciencedirect.com/science/article/pii/S0040195105004439.

Alexander Tong, Nikolay Malkin, Kilian Fatras, Lazar Atanackovic, Yanlei Zhang, Guillaume Huguet, Guy Wolf, and Yoshua Bengio. Simulation-free schrödinger bridges via score and flow matching, 2024. URL https://arxiv.org/abs/2307.03672.

Paul Van Houtte, Saiyi Li, Marc Seefeldt, and Laurent Delannay. Deformation texture prediction: from the taylor model to the advanced lamel model. *International Journal of Plasticity*, 21(3):589–624, 2005. ISSN 0749-6419. doi: https://doi.org/10.1016/j.ijplas.2004.04.011. URL https://www.sciencedirect.com/science/article/pii/S0749641904000610.

Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

Xiaohan Zeng, Christopher S. Daniel, William Asheshov, Stefan Michalik, Philip Prangnell, and João Quinta da Fonseca. Effect of temperature, strain and microstructure on texture evolution of ti-6al-4v during hot rolling. *Metallurgical and Materials Transactions A*, 56(10):4620–4643, 2025. doi: 10.1007/s11661-025-07929-8.

A Appendix

A.1 Proofs

Proof of Proposition 1

Proof. By definition of the mixture potential,

$$v_\theta(y) = \sum_{k=1}^{K} \alpha_k v_{\theta,k}(y).$$

Therefore, using linearity of integration, its normalization function is

$$\begin{aligned} r_\theta(x) &= \int_{\mathcal{M}} K_\varepsilon(x,y) v_\theta(y)\, dm(y) \\ &= \int_{\mathcal{M}} K_\varepsilon(x,y) \sum_{k=1}^{K} \alpha_k v_{\theta,k}(y)\, dm(y) \\ &= \sum_{k=1}^{K} \alpha_k \int_{\mathcal{M}} K_\varepsilon(x,y) v_{\theta,k}(y)\, dm(y) \\ &= \sum_{k=1}^{K} \alpha_k r_{\theta,k}(x). \end{aligned}$$

The corresponding conditional distribution is

$$\begin{aligned} q_\theta(y|x) &= \frac{K_\varepsilon(x,y) v_\theta(y)}{r_\theta(x)} \\ &= \sum_{k=1}^{K} \frac{\alpha_k r_{\theta,k}(x)}{r_\theta(x)} \frac{K_\varepsilon(x,y) v_{\theta,k}(y)}{r_{\theta,k}(x)} \\ &= \sum_{k=1}^{K} w_{\theta,k}(x)\, q_{\theta,k}(y|x), \end{aligned}$$

where

$$w_{\theta,k}(x) = \frac{\alpha_k r_{\theta,k}(x)}{r_\theta(x)}.$$

Since all α_k and $r_{\theta,k}(x)$ are positive, $w_{\theta,k}(x) > 0$, and

$$\sum_{k=1}^{K} w_{\theta,k}(x) = 1.$$

Hence $q_\theta(\cdot|x)$ is a finite mixture of the component conditionals $q_{\theta,k}(\cdot|x)$. Since each component can be sampled directly, one can sample from $q_\theta(\cdot|x)$ by first drawing $k \sim \mathrm{Categorical}(w_{\theta,1}(x), \ldots, w_{\theta,K}(x))$ and then sampling from $q_{\theta,k}(\cdot|x)$. Thus, closed-form normalization and direct conditional sampling are preserved by the mixture. □

Proof of Proposition 2

Proof. Using the product structure of both the kernel and potential component,

$$K_\varepsilon(x,y) v_{\theta,k}(y) = \prod_{j=1}^{J} K_{\varepsilon,j}(x^j, y^j) v_{\theta,k,j}(y^j).$$

Since the reference measure also factorizes as $m = \bigotimes_{j=1}^{J} m_j$, the corresponding normalization function satisfies

$$\begin{aligned} r_{\theta,k}(x) &= \int_{\mathcal{M}} K_\varepsilon(x,y) v_{\theta,k}(y)\, dm(y) \\ &= \int_{\prod_{j=1}^{J} \mathcal{M}_j} \prod_{j=1}^{J} K_{\varepsilon,j}(x^j, y^j) v_{\theta,k,j}(y^j) \prod_{j=1}^{J} dm_j(y^j) \\ &= \prod_{j=1}^{J} \int_{\mathcal{M}_j} K_{\varepsilon,j}(x^j, y^j) v_{\theta,k,j}(y^j)\, dm_j(y^j) \\ &= \prod_{j=1}^{J} r_{\theta,k,j}(x^j). \end{aligned}$$

Therefore, the normalized conditional distribution becomes

$$\begin{aligned} q_{\theta,k}(y|x) &= \frac{K_\varepsilon(x,y) v_{\theta,k}(y)}{r_{\theta,k}(x)} \\ &= \prod_{j=1}^{J} \frac{K_{\varepsilon,j}(x^j, y^j) v_{\theta,k,j}(y^j)}{r_{\theta,k,j}(x^j)} \\ &= \prod_{j=1}^{J} q_{\theta,k,j}(y^j|x^j). \end{aligned}$$

Hence the conditional factorizes across the component manifolds. Since each $q_{\theta,k,j}(\cdot|x^j)$ can be sampled directly by assumption, a sample from $q_{\theta,k}(\cdot|x)$ is obtained by independently sampling $y^j \sim q_{\theta,k,j}(\cdot|x^j)$ for $j = 1, \ldots, J$ and setting $y = (y^1, \ldots, y^J)$. Thus, closed-form normalization and direct conditional sampling are preserved under Cartesian products. □

Proofs of the manifold-specific parameterizations

Distributional notation. For $g \in \mathbb{R}^d$, we define

$$Z_d(g) = \int_{\mathbb{S}^{d-1}} \exp(g^\top y)\, dm(y),$$

and denote by $\mathrm{vMF}(g)$ the von Mises-Fisher distribution on $\mathbb{S}^{d-1}$ with density $\exp(g^\top y)/Z_d(g)$. Since $Z_d(g)$ depends only on $\|g\|_2$, we have $Z_d(-g) = Z_d(g)$. For the torus construction, $\mathrm{vM}(h)$, with $h \in \mathbb{R}^2$, denotes the circular von Mises distribution whose density is proportional to $\exp(h^\top(\cos y, \sin y)^\top)$; its normalizing constant is $2\pi I_0(\|h\|_2)$, where I_0 is the modified Bessel function of the first kind of order zero, defined by

$$I_0(t) = \frac{1}{2\pi} \int_0^{2\pi} \exp(t \cos\phi)\, d\phi.$$

Sphere. Consider one potential component $v_{\theta,k}$. Multiplying it by the kernel in equation 7 gives

$$K_\varepsilon(x,y) v_{\theta,k}(y) = \exp\left(\frac{x^\top y}{\varepsilon}\right) \exp(f_k^\top y) = \exp\left(\left(f_k + \frac{x}{\varepsilon}\right)^\top y\right).$$

Thus, the kernel-potential product is itself an unnormalized von Mises-Fisher density with natural parameter $f_k + x/\varepsilon$. Integrating over the sphere immediately gives $r_{\theta,k}(x) = Z_d(f_k + x/\varepsilon)$, and dividing by this normalizer gives $q_{\theta,k}(\cdot|x) = \mathrm{vMF}(f_k + x/\varepsilon)$. Hence each component satisfies the assumptions of Proposition 1, which yields closed-form normalization and direct sampling for the complete mixture q_θ.

Torus. For a fixed coordinate j, the addition identity $\cos(x^j - y^j) = \cos x^j \cos y^j + \sin x^j \sin y^j$ gives

$$\begin{aligned}
&\frac{1}{\varepsilon}\cos(x^j - y^j) + a_{kj}\cos y^j + b_{kj}\sin y^j \\
&\quad = \left(a_{kj} + \frac{1}{\varepsilon}\cos x^j\right)\cos y^j + \left(b_{kj} + \frac{1}{\varepsilon}\sin x^j\right)\sin y^j \\
&\quad = h_{kj}(x)^\top \begin{pmatrix}\cos y^j \\ \sin y^j\end{pmatrix}.
\end{aligned}$$

Therefore,

$$K_\varepsilon(x,y)v_{\theta,k}(y) = \prod_{j=1}^{d} \exp\left[h_{kj}(x)^\top \begin{pmatrix}\cos y^j \\ \sin y^j\end{pmatrix}\right].$$

The product factorizes across coordinates. Since the normalizer of each factor is $2\pi I_0(\|h_{kj}(x)\|_2) = 2\pi I_0(\rho_{kj}(x))$, integration gives

$$r_{\theta,k}(x) = \prod_{j=1}^{d} 2\pi I_0(\rho_{kj}(x)), \qquad q_{\theta,k}(\cdot|x) = \prod_{j=1}^{d} \mathrm{vM}(h_{kj}(x)).$$

Thus the coordinates can be sampled independently, and Proposition 1 gives the corresponding mixture construction.

Rotation Group. Let

$$g_{k,s}(x) = f_k + s\frac{\lambda_R}{\varepsilon}x, \qquad s \in \{-1, +1\}.$$

Using $\cosh(a)\cosh(b) = \frac{1}{2}[\cosh(a+b) + \cosh(a-b)]$, we obtain

$$K_\varepsilon(x,y)v_{\theta,k}(y) = \frac{1}{2}\sum_{s\in\{-1,+1\}} \cosh\big(g_{k,s}(x)^\top y\big).$$

For any $g \in \mathbb{R}^4$,

$$\int_{\mathbb{S}^3} \cosh(g^\top y)\, dm(y) = \frac{Z_4(g) + Z_4(-g)}{2} = Z_4(g).$$

Consequently,

$$r_{\theta,k}(x) = \frac{1}{2}\left[Z_4\left(f_k + \frac{\lambda_R}{\varepsilon}x\right) + Z_4\left(f_k - \frac{\lambda_R}{\varepsilon}x\right)\right].$$

After normalization, the two terms are selected with probabilities

$$\pi_{k,s}(x) = \frac{Z_4\big(f_k + s\frac{\lambda_R}{\varepsilon}x\big)}{Z_4\big(f_k + \frac{\lambda_R}{\varepsilon}x\big) + Z_4\big(f_k - \frac{\lambda_R}{\varepsilon}x\big)}.$$

Recall that the unit quaternions y and $-y$ represent the same rotation. Accordingly, a density proportional to $\cosh(g^\top y)$ on $\mathrm{SO}(3)$ can be sampled by drawing a quaternion from $\mathrm{vMF}(g)$ on $\mathbb{S}^3$ and then mapping it to its corresponding rotation. Hence

$$q_{\theta,k}(\cdot|x) = \sum_{s\in\{-1,+1\}} \pi_{k,s}(x)\ \mathrm{vMF}\left(f_k + s\frac{\lambda_R}{\varepsilon}x\right),$$

where the sampled quaternion is understood to be mapped to $\mathrm{SO}(3)$. Each component is therefore directly sampleable, and Proposition 1 gives the complete mixture.

Special Euclidean Group. We use the product representation $\mathrm{SE}(3) \simeq \mathbb{R}^3 \times \mathrm{SO}(3)$. The rotational factor has already been shown above to satisfy the tractability conditions, so it remains to verify the translational factor. For K_ε^t and the translational part of $v_{\theta,k}$, define

$$A_k = \left(\sigma_\varepsilon^{-2} I_3 + \Sigma_k^{-1}\right)^{-1}, \qquad \bar{\mu}_k = A_k \left(\sigma_\varepsilon^{-2} x^t + \Sigma_k^{-1} \mu_k\right).$$

Completing the square in y^t gives

$$\begin{aligned}&\frac{\|x^t - y^t\|_2^2}{\sigma_\varepsilon^2} + (y^t - \mu_k)^\top \Sigma_k^{-1} (y^t - \mu_k) \\ &\quad = (y^t - \bar{\mu}_k)^\top A_k^{-1} (y^t - \bar{\mu}_k) + (x^t - \mu_k)^\top (\Sigma_k + \sigma_\varepsilon^2 I_3)^{-1} (x^t - \mu_k).\end{aligned}$$

The dependence on y^t is therefore Gaussian, so the normalized conditional is $q_{\theta,k}^t(\cdot|x^t) = \mathcal{N}(\cdot|\bar{\mu}_k, A_k)$. Integrating the Gaussian term yields

$$\begin{aligned}r_{\theta,k}^t(x^t) = (2\pi)^{3/2} \sigma_\varepsilon^3 \det(\Sigma_k)^{1/2} \det(\Sigma_k + \sigma_\varepsilon^2 I_3)^{-1/2} \\ \times \exp\left[-\frac{1}{2}(x^t - \mu_k)^\top (\Sigma_k + \sigma_\varepsilon^2 I_3)^{-1} (x^t - \mu_k)\right].\end{aligned}$$

Hence both the translational and rotational factors satisfy the assumptions of Proposition 1. Applying Proposition 2 gives $r_{\theta,k}(x) = r_{\theta,k}^t(x^t) r_{\theta,k}^R(x^R)$ and $q_{\theta,k}(y|x) = q_{\theta,k}^t(y^t|x^t) q_{\theta,k}^R(y^R|x^R)$. Finally, Proposition 1 gives the full K-component mixture.

A.2 Datasets

A.2.1 Synthetic Datasets

For all synthetic experiments, we independently generate 20000 training samples and 5120 test samples for each of the source and target marginals. Training uses only the unpaired source and target samples, while the held-out test sets are used for evaluation.

Synthetic Sphere. For the experiments on $\mathbb{S}^D \subset \mathbb{R}^{D+1}$, the source distribution is uniform on the sphere. A source sample is generated by drawing $z \sim \mathcal{N}(0, I_{D+1})$ and normalizing it as $x = z/\|z\|_2$. The target is a wrapped normal distribution centered at the fixed point $\ell = (-1, 0, \ldots, 0) \in \mathbb{S}^D$. We first draw $\xi \sim \mathcal{N}(0, \sigma^2 I_D)$ and embed it in $\mathbb{R}^{D+1}$ as $\widetilde{\xi} = (0, \xi)$. Since $\ell^\top \widetilde{\xi} = 0$, this vector is tangent to the sphere at ℓ, i.e., it points along the local D-dimensional plane orthogonal to ℓ. The sample is then mapped onto the sphere using the spherical exponential map,

$$y = \cos(\|\widetilde{\xi}\|_2)\ell + \frac{\sin(\|\widetilde{\xi}\|_2)}{\|\widetilde{\xi}\|_2}\widetilde{\xi},$$

where $y = \ell$ when $\|\widetilde{\xi}\|_2 = 0$. We use $\sigma = 0.3$ for all dimensions.

Synthetic Torus. A point on $\mathbb{T}^D$ is represented by D angular coordinates $\theta = (\theta^1, \ldots, \theta^D)$, with each $\theta^j \in [0, 2\pi)$. Source samples are generated independently as $\theta^j \sim \mathrm{Unif}[0, 2\pi)$. For the target, we use a wrapped Gaussian centered at $(\pi, \ldots, \pi)$. Specifically, we draw $z \sim \mathcal{N}(0, \sigma^2 I_D)$ and set $\theta = (\pi \mathbf{1} + z) \bmod 2\pi$, where the modulo operation is applied coordinate-wise and $\mathbf{1}$ denotes the D-dimensional vector of ones. We use $\sigma = 0.3$ in all experiments. When an ambient Euclidean representation is required, each angle is mapped to the corresponding point on the unit circle as $\theta^j \mapsto (\cos\theta^j, \sin\theta^j)$.

Synthetic $\mathrm{SO}(3)$. Rotations are represented by unit quaternions $q = (q_0, q_1, q_2, q_3) \in \mathbb{S}^3$, with q and $-q$ representing the same three-dimensional rotation. Both marginals are wrapped normal distributions with different center rotations. The source is centered at the identity rotation, represented by $q_{\mathrm{s}} = (1, 0, 0, 0)$. The target center is obtained by rotating through an angle $\phi = 2.5$ radians around the axis $a = (0, 0, 1)$, giving the unit quaternion $q_{\mathrm{t}} = (\cos(\phi/2), \sin(\phi/2)a)$.

To sample around either center q_c, we draw a three-dimensional perturbation $v \sim \mathcal{N}(0, \sigma^2 I_3)$, with $\sigma = 0.8$. Writing $r = \|v\|_2$, this perturbation is converted into the rotation quaternion

$$\delta(v) = \left(\cos\frac{r}{2}, \frac{\sin(r/2)}{r} v\right),$$

with $\delta(0) = (1, 0, 0, 0)$. The final sample is $q = q_c \otimes \delta(v)$, where $\otimes$ denotes quaternion multiplication. Thus, Gaussian perturbations in three-dimensional rotation-vector coordinates are wrapped around the corresponding center rotation. Source and target train and test samples are generated independently.

Synthetic $\mathrm{SE}(3)$**.** A point in $\mathrm{SE}(3)$ consists of a three-dimensional rotation and a three-dimensional translation, and is represented in our implementation by a unit quaternion $q \in \mathbb{S}^3$ together with a translation vector $t \in \mathbb{R}^3$. For the source distribution, rotations are sampled uniformly over all three-dimensional rotations, i.e., from the Haar-uniform distribution on $\mathrm{SO}(3)$, while translations are sampled uniformly from the cube $[-4, 4]^3$.

For the target rotation, we use a wrapped normal distribution centered at a rotation of 60° around the z-axis. As in the $\mathrm{SO}(3)$ experiment, we draw a rotation vector $v \sim \mathcal{N}(0, 0.3^2 I_3)$, convert it into a rotation quaternion $\delta(v)$, and compose it with the target center rotation. The target translations follow a Gaussian distribution with mean $\mu_t = (1, 0.5, -0.5)$ and isotropic standard deviation 0.5. To keep the translational support identical to that of the source, samples outside $[-4, 4]^3$ are rejected and resampled. Rotation and translation are sampled independently within each marginal. Each resulting sample is stored as $(q_0, q_1, q_2, q_3, t_1, t_2, t_3)$.

A.2.2 Continental Drift

Continental Drift. We follow the continental-drift benchmark of Cohen et al. (2021) using paleogeographic reconstructions from GPlates (Müller et al., 2018). The source distribution corresponds to continental regions at 150 million years ago, while the target corresponds to present-day continental regions. The reconstructions are provided as GeoJSON files containing polygonal continental boundaries in geographic coordinates, with each polygon represented by a sequence of longitude-latitude vertices. Thus, the raw data do not consist of points on $\mathbb{S}^2$; instead, these polygons define the continental regions from which spherical samples are obtained.

Points are sampled uniformly with respect to surface area on $\mathbb{S}^2$ and restricted to the corresponding continental regions by rejection sampling. Specifically, we draw a longitude $\lambda \sim \mathrm{Unif}[-\pi, \pi)$ and a scalar $z \sim \mathrm{Unif}[-1, 1]$. The latitude is then $\varphi = \arcsin z$, and the corresponding point on the unit sphere is

$$x = (\cos\varphi \cos\lambda,\ \cos\varphi \sin\lambda,\ \sin\varphi).$$

This construction is uniform with respect to spherical surface area. The longitude-latitude coordinates of each candidate point are tested against the continental polygons for the corresponding time period, and candidates outside all polygons are rejected. Accepted points are represented by their three-dimensional unit vectors $x \in \mathbb{S}^2$ for training and evaluation. The procedure is applied independently to the 150 Ma and present-day polygons, yielding unpaired empirical source and target distributions. We sample $50{,}000$ training points from each distribution.

A.2.3 Crystallographic Texture Dataset

We use the Ti-6Al-4V electron backscatter diffraction (EBSD) dataset of Zeng et al. (2025), which contains crystallographic orientation maps for the starting material and for samples subjected to different hot-rolling conditions. The raw measurements are provided as EBSD maps in the DefDAP format, with each spatial measurement associated with a crystallographic orientation and a phase label. In our experiments, we use the *starting material* as the source state and the material rolled at $825^\circ\mathrm{C}$ with a 75% thickness reduction as the target state. We construct two transport tasks corresponding to the rolling-direction (RD) and transverse-direction (TD) sections: for each section, the source distribution is formed by the crystallographic orientations measured in the starting material, while the target distribution is formed by the orientations measured after the $825^\circ\mathrm{C}$, 75%-reduction treatment. These data are naturally unpaired because the source and target orientations are obtained from separate EBSD maps acquired at different material states; consequently, an orientation measured at a given spatial location in the starting material has no one-to-one correspondence with an orientation in the rolled sample. The learning problem is therefore to transport between the two empirical orientation distributions rather than between matched orientation pairs.

We preprocess each EBSD map using `DefDAP`. Given a raw `.cpr` file, we load the corresponding EBSD map and flatten its orientation and phase arrays so that each measured point is represented independently. Each crystallographic orientation is converted to its quaternion coefficients $q =$

(q_0, q_1, q_2, q_3), after which the quaternion is normalized to unit norm. Since q and $-q$ represent the same element of $\mathrm{SO}(3)$, we use a canonical representative by changing the sign whenever $q_0 < 0$, so that all stored quaternions satisfy $q_0 \geq 0$. The resulting unit quaternions, together with their corresponding phase labels, are saved as PyTorch tensors for subsequent training and evaluation.

A.3 METRICS

Maximum Mean Discrepancy. For all manifold experiments, we compare the transported distribution with the target distribution using the squared Maximum Mean Discrepancy (MMD) equipped with a kernel adapted to the corresponding manifold. Let P and Q denote two probability distributions on $\mathcal{M}$, and let $k : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ be a positive-definite kernel. The squared MMD is defined as

$$\mathrm{MMD}^2(P, Q; k) = \mathbb{E}_{x,x' \sim P}[k(x, x')] + \mathbb{E}_{y,y' \sim Q}[k(y, y')] - 2\mathbb{E}_{x \sim P, y \sim Q}[k(x, y)], \tag{18}$$

where x, x' and y, y' are independent samples from their respective distributions. Given target samples $\{y_i\}_{i=1}^N$ and transported samples $\{\widehat{y}_j\}_{j=1}^M$, we use the empirical estimator

$$\widehat{\mathrm{MMD}}^2 = \frac{1}{N^2} \sum_{i,i'=1}^{N} k(y_i, y_{i'}) + \frac{1}{M^2} \sum_{j,j'=1}^{M} k(\widehat{y}_j, \widehat{y}_{j'}) - \frac{2}{NM} \sum_{i=1}^{N} \sum_{j=1}^{M} k(y_i, \widehat{y}_j). \tag{19}$$

This is the biased empirical estimator, with the diagonal terms included, and corresponds directly to the quantity computed in our implementation. Lower values indicate closer agreement between the transported and target distributions. We use different kernels for each manifold in order to respect its geometry.

Sphere. For $x, y \in \mathbb{S}^{d-1}$, we use the von Mises-Fisher kernel

$$k_{\mathbb{S}}(x, y) = \exp\left[\kappa \left(x^\top y - 1\right)\right], \tag{20}$$

where $\kappa > 0$ controls the concentration of the kernel. Since x and y are unit vectors, $x^\top y$ depends only on their relative position on the sphere, and the subtraction of 1 normalizes the kernel such that $k_{\mathbb{S}}(x, x) = 1$.

Torus. For $x, y \in \mathbb{T}^d$, represented by angular coordinates $x = (x^1, \ldots, x^d)$ and $y = (y^1, \ldots, y^d)$, we use the product von Mises kernel

$$k_{\mathbb{T}}(x, y) = \exp\left[\kappa \sum_{j=1}^{d} \left(\cos(x^j - y^j) - 1\right)\right]. \tag{21}$$

Equivalently, this kernel is the product of one-dimensional circular von Mises kernels over the d coordinates. It is invariant to shifts of any angular coordinate by integer multiples of 2π and therefore respects the periodic structure of $\mathbb{T}^d$.

Rotation group. For $\mathrm{SO}(3)$, rotations are represented by unit quaternions $q, q' \in \mathbb{S}^3$, with q and $-q$ corresponding to the same rotation. We first compute their rotation angle

$$\theta(q, q') = 2 \arccos\left(\left|q^\top q'\right|\right) \in [0, \pi]. \tag{22}$$

The absolute inner product makes the distance invariant to the quaternion sign ambiguity. We then use a normalized heat kernel on $\mathrm{SO}(3)$,

$$k_{\mathrm{SO}(3)}(q, q') = \frac{\sum_{\ell=0}^{\infty} (2\ell + 1) e^{-\tau \ell(\ell+1)} \chi_\ell(\theta(q, q'))}{\sum_{\ell=0}^{\infty} (2\ell + 1)^2 e^{-\tau \ell(\ell+1)}}, \tag{23}$$

where $\tau > 0$ is the heat time and

$$\chi_\ell(\theta) = 1 + 2 \sum_{m=1}^{\ell} \cos(m\theta) = \frac{\sin\left((\ell + \frac{1}{2})\theta\right)}{\sin(\theta/2)} \tag{24}$$

is the character of the $(2\ell+1)$-dimensional irreducible representation of $\mathrm{SO}(3)$. The denominator corresponds to the value of the unnormalized kernel at the identity and therefore ensures $k_{\mathrm{SO}(3)}(q,q)=1$. In practice, the spectral expansion is truncated once the remaining tail is below a numerical tolerance.

Special Euclidean group. For $x=(q,t)$ and $y=(q',t')$ in $\mathrm{SE}(3)$, where $q,q'\in\mathrm{SO}(3)$ denote rotations and $t,t'\in\mathbb{R}^3$ translations, we use a product kernel combining the rotational heat kernel above with a Gaussian kernel on the translation component:

$$k_{\mathrm{SE}(3)}(x,y)=k_{\mathrm{SO}(3)}^{\tau/\alpha^2}(q,q')\exp\left(-\frac{\|t-t'\|_2^2}{4\tau}\right). \tag{25}$$

Here, $k_{\mathrm{SO}(3)}^{\tau/\alpha^2}$ denotes the $\mathrm{SO}(3)$ heat kernel in equation 23 evaluated with heat time τ/α^2, while $\alpha>0$ controls the relative scale between rotational and translational differences. This product construction respects the $\mathbb{R}^3\times\mathrm{SO}(3)$ representation used for $\mathrm{SE}(3)$.

Kernel parameters. For the sphere and torus experiments, the kernel concentration is selected separately for each dimension using the scale of the target distribution. Specifically, we compute the median positive pairwise manifold distance s among the first 200 target samples and set $\kappa=1/s^2$. For the sphere, the distance used for this bandwidth selection is the spherical angle $d_{\mathbb{S}}(x,y)=\arccos(x^\top y)$, while for the torus we use the wrapped distance $d_{\mathbb{T}}(x,y)=\left(\sum_{j=1}^{D}\mathrm{wrap}(x^j-y^j)^2\right)^{1/2}$, where $\mathrm{wrap}(\delta)=\mathrm{atan2}(\sin\delta,\cos\delta)$.

For $D\in\{2,10,20,40,60,80,100\}$, the resulting sphere concentrations are $\kappa_{\mathbb{S}}\in\{4.071,0.700,0.438,0.458,0.808,2.145,10.157\}$, respectively. For the same dimensions, the torus concentrations are $\kappa_{\mathbb{T}}\in\{4.010,0.5947,0.2873,0.1412,0.09363,0.07003,0.05593\}$, respectively.

For $\mathrm{SO}(3)$, we use the heat kernel with a fixed heat time $\tau=0.5$ for all experiments. For $\mathrm{SE}(3)$, we likewise use $\tau=0.5$, with the rotational factor evaluated at heat time τ/α^2 and $\alpha=2$, as in the kernel definition above.

Crystallographic texture metrics. For the crystallographic texture experiment, let $\{y^{(i)}\}_{i=1}^N$ and $\{\widehat{y}^{(j)}\}_{j=1}^M$ denote the target and transported orientations, respectively. We estimate their orientation distribution functions (ODFs) using the same $\mathrm{SO}(3)$ heat kernel introduced above. Specifically, we use the heat-kernel density

$$\widetilde{k}_{\mathrm{SO}(3)}^{\tau}(q,q')=\sum_{\ell=0}^{\infty}(2\ell+1)e^{-\tau\ell(\ell+1)}\chi_\ell(\theta(q,q'))\,, \tag{26}$$

which corresponds to the numerator of equation 23. Unlike the MMD kernel, no normalization by its value at the identity is required here. The target and predicted ODFs are estimated as

$$f_{\mathrm{tgt}}(g)=\frac{1}{N}\sum_{i=1}^{N}\widetilde{k}_{\mathrm{SO}(3)}^{\tau_{\mathrm{tex}}}\left(g,y^{(i)}\right),\qquad f_{\mathrm{pred}}(g)=\frac{1}{M}\sum_{j=1}^{M}\widetilde{k}_{\mathrm{SO}(3)}^{\tau_{\mathrm{tex}}}\left(g,\widehat{y}^{(j)}\right), \tag{27}$$

where $g\in\mathrm{SO}(3)$ and we set $\tau_{\mathrm{tex}}=0.01$.

The *texture index* measures the concentration of an ODF and is defined as

$$\mathrm{TI}(f)=\int_{\mathrm{SO}(3)}f(g)^2\,dg. \tag{28}$$

We report the relative texture-index error

$$E_{\mathrm{TI}}=\frac{|\mathrm{TI}(f_{\mathrm{pred}})-\mathrm{TI}(f_{\mathrm{tgt}})|}{\mathrm{TI}(f_{\mathrm{tgt}})}. \tag{29}$$

In practice, these integrals are evaluated directly from the orientation samples using the heat-kernel identity, resulting in pairwise evaluations of $\widetilde{k}_{\mathrm{SO}(3)}^{2\tau_{\mathrm{tex}}}$; diagonal self-pairs are excluded when estimating each texture index.

The *normalized ODF difference* measures the discrepancy between the complete predicted and target orientation distributions:

$$E_{\mathrm{ODF}} = \frac{\int_{\mathrm{SO}(3)} \left(f_{\mathrm{pred}}(g) - f_{\mathrm{tgt}}(g)\right)^2 \, dg}{\int_{\mathrm{SO}(3)} f_{\mathrm{tgt}}(g)^2 \, dg}. \tag{30}$$

Finally, the *M-index* characterizes texture through the distribution of crystallographic misorientation angles. We compute HCP symmetry-aware disorientation angles from pairs of orientations and construct normalized histograms p_{tgt} and p_{pred}. Given the corresponding histogram p_{rand} for uniformly random orientations, the M-index is

$$M(p) = \frac{1}{2} \sum_{b=1}^{B} \left| p^{(b)} - p^{(b)}_{\mathrm{rand}} \right|, \tag{31}$$

and we report

$$E_{\mathrm{M}} = \left| M(p_{\mathrm{pred}}) - M(p_{\mathrm{tgt}}) \right|. \tag{32}$$

We use $B = 180$ bins, up to 4096 orientation pairs for each distribution, and $100{,}000$ random orientation pairs to construct p_{rand}. Lower values indicate better agreement with the target for all three metrics.

Quantitative comparison of crystallographic textures is commonly based on descriptors of the orientation distribution and crystallographic misorientations. The texture index, also known as the J-index, is widely used to quantify the strength or concentration of an orientation distribution, while the M-index was introduced by Skemer et al. (2005) as a complementary measure based on the distribution of crystallographic misorientation angles. Both measures have subsequently been used in EBSD-based texture analysis (Ott et al., 2024). To directly compare two orientation distributions, normalized squared differences between their ODFs have also been used in deformation-texture and crystal-plasticity studies to quantify agreement between predicted and measured textures (Van Houtte et al., 2005). Following these approaches, we use the texture index error, normalized ODF difference, and M-index error to evaluate complementary aspects of the agreement between the transported and target crystallographic textures.

A.4 Implementation Details

For each experiment, all methods use the same training and test distributions and are evaluated on the same test samples. Unless stated otherwise, we use minibatches of size 256. ManifoldLightOT, ERNOT, and NeuralOT are implemented in PyTorch, while RNOT follows its original JAX/Flax implementation. All experiments are run on a single GPU. The hyperparameters described below are those used for the reported experiments.

ManifoldLightOT. ManifoldLightOT directly optimizes the parameters of the geometry-specific mixture potentials introduced in §3.2, without using an additional neural network. The mixture components are initialized from randomly selected target training samples, with initially uniform mixture weights, and all parameters are optimized with Adam using $\beta_1 = 0.9$, $\beta_2 = 0.999$, and no weight decay.

Synthetic experiments. For the high-dimensional sphere and torus experiments, we use $K = 30$ components in dimension 2 and $K = 10D$ components for $D \in \{10, 20, 40, 60, 80, 100\}$, with batch size 256. We train for 50000 steps for $D \leq 20$, 100000 steps for $D \in \{40, 60, 80\}$, and 200000 steps for $D = 100$. The learning rate is 10^{-3} for the low- and medium-dimensional configurations and 10^{-2} for the higher-dimensional configurations. We use $\varepsilon = 0.1$ up to dimension 40 and $\varepsilon = 0.01$ for dimensions 60–100, with $\mathbb{T}^2$ also using $\varepsilon = 0.01$. For the synthetic $\mathrm{SO}(3)$ and $\mathrm{SE}(3)$ experiments, we use $K = 100$ components, batch size 256, and train for $50{,}000$ steps with learning rate 10^{-2}, using $\varepsilon = 0.05$ and 0.2, respectively. The $\mathrm{SO}(3)$ rotational kernel scale is $\lambda_R = 1$, while for $\mathrm{SE}(3)$ we use rotational scale 1 and translational scale 0.16.

Continental Drift. We use $K = 1000$ components, $\varepsilon = 0.1$, batch size 64, and train for 10^6 steps with learning rate 10^{-2}.

Crystallographic texture. We use $K = 1000$ components, $\varepsilon = 0.01$, rotational kernel scale $\lambda_R = 2$, batch size 256, and train for 200000 steps with learning rate 10^{-2}.

RNOT. Our RNOT implementation is based on the official implementation provided by the authors (`https://openreview.net/forum?id=ez4oLq7PR3`). The potential is parameterized using the Gromov distance embedding of the input with respect to a set of manifold landmarks, followed by an MLP with two hidden layers of 128 units, SiLU activations, and LayerNorm. The inner optimization required to evaluate the c-transform is performed with Adam using $\beta_1 = 0.9$, $\beta_2 = 0.999$, and tolerance 10^{-6}.

Synthetic experiments. We use 128 randomly selected landmarks, batch size 256, learning rate 10^{-3}, and no learning-rate decay. The inner optimization uses 2500 iterations with learning rate 5×10^{-2}. Most experiments are trained for 1000 outer steps, increased to 1500 for $\mathbb{T}^{40}$ and 2000 for $\mathbb{T}^{80}$, $\mathbb{T}^{100}$, and $\mathrm{SE}(3)$.

Continental Drift. We use 1024 landmarks obtained by farthest-point sampling from 4096 candidates, batch size 1024, and train for 1000 outer steps with learning rate 10^{-3} and no learning-rate decay. The inner optimization uses 1000 iterations with learning rate 5×10^{-2}.

Crystallographic texture. We use 128 randomly selected landmarks, batch size 256, and train for 10,000 outer steps with learning rate 10^{-3} and no learning-rate decay. The inner optimization uses 2500 iterations with learning rate 5×10^{-2}.

ERNOT. We parameterize the ERNOT Schrödinger potential using distances to manifold landmarks, followed by a two-layer MLP with 256 hidden units per layer, SiLU activations, and LayerNorm on the landmark embedding. The output layer is initialized with standard deviation 0.01. The regularization parameter ε is estimated separately for each dataset from 256 training samples using the scale factor 0.05. The potential is optimized with Adam with initial learning rate 10^{-3}, together with cosine annealing to zero.

Synthetic experiments. We use 256 landmarks, batch size 256, and train for 3000 steps with initial learning rate 10^{-3}.

Continental Drift. We use 256 landmarks, batch size 64, and train for 3000 steps with initial learning rate 10^{-3}.

Crystallographic texture. We use 256 landmarks, batch size 256, and train for 100,000 steps with initial learning rate 10^{-3}. At inference time, we use a target support of 200 samples.

NeuralOT. Our NeuralOT implementation is based on the official implementation (`https://github.com/iamalexkorotin/NeuralOptimalTransport`), adapted to the manifold representations and transport costs considered in this work. Both the transport map T and the potential f are represented by MLPs with three hidden layers and ReLU activations. The transport network maps between the corresponding manifold coordinate representations and its output is projected back to the manifold, while the potential network outputs a scalar. We use hidden widths of 256 or 512, depending on the experimental configuration. Training follows the alternating minimax procedure of NeuralOT: several transport-map updates are performed with the potential frozen, followed by one potential update with the transport map frozen. Both networks are optimized independently with Adam using $\beta_1 = 0.9$, $\beta_2 = 0.999$, and no weight decay. NeuralOT uses the manifold-specific costs corresponding to the kernels in Table 1.

Synthetic experiments. Both networks are optimized with learning rate 10^{-4}. We use 10 transport-map updates per potential update for all configurations except $\mathbb{T}^2$, for which we use 5. The batch size is 256, except for $\mathbb{T}^{40}$, where we use 512. On the sphere, we train for 10,000 steps for $D \in \{2, 10\}$, 20,000 steps for $D \in \{20, 40\}$, 30,000 steps for $D \in \{60, 80\}$, and 50,000 steps for $D = 100$. On the torus, we use 10,000 steps for $D \in \{2, 10, 20\}$ and 20,000 steps for $D \in \{40, 60, 80, 100\}$. The $\mathrm{SO}(3)$ and $\mathrm{SE}(3)$ experiments are trained for 20,000 steps.

Continental Drift. NeuralOT is not included in the Continental Drift experiments.

Crystallographic texture. We use hidden width 512, batch size 256, and 5 transport-map updates per potential update. Both networks are optimized with learning rate 10^{-4}, and training is performed for 20,000 steps.

A.5 Additional Results

In this section we present additional results for our synthetic and crystallization experimental setups.

Additional High Dimensional Spheres and Tori. In Table 4 we show additional dimensionalities which complements the ones appearing in Table 2. It is important to note that ManifoldLightOT remains showing a strong performance against other methods.

Table 4: MMD^2 results on high-dimensional spheres and tori, reported in units of 10^{-2}. We report mean $\pm$ standard deviation over five seeds. Lower is better.

Sphere							
Method	$\mathbb{S}^2$	$\mathbb{S}^{10}$	$\mathbb{S}^{20}$	$\mathbb{S}^{40}$	$\mathbb{S}^{60}$	$\mathbb{S}^{80}$	$\mathbb{S}^{100}$
ManifoldLightOT	0.08 ± 0.01	0.07 ± 0.01	0.08 ± 0.01	0.06 ± 0.00	0.07 ± 0.00	0.13 ± 0.01	0.45 ± 0.05
RNOT	0.13 ± 0.06	0.10 ± 0.02	0.10 ± 0.01	0.11 ± 0.02	0.18 ± 0.04	0.43 ± 0.09	1.47 ± 0.21
ERNOT	0.12 ± 0.03	0.15 ± 0.01	0.20 ± 0.03	0.20 ± 0.04	0.22 ± 0.01	0.25 ± 0.02	0.32 ± 0.09
NeuralOT	0.19 ± 0.07	0.12 ± 0.04	0.17 ± 0.02	0.17 ± 0.02	0.27 ± 0.06	0.29 ± 0.04	4.38 ± 8.24
Torus							
Method	$\mathbb{T}^2$	$\mathbb{T}^{10}$	$\mathbb{T}^{20}$	$\mathbb{T}^{40}$	$\mathbb{T}^{60}$	$\mathbb{T}^{80}$	$\mathbb{T}^{100}$
ManifoldLightOT	0.27 ± 0.06	0.08 ± 0.02	0.08 ± 0.01	0.07 ± 0.00	0.08 ± 0.00	0.08 ± 0.01	0.08 ± 0.00
RNOT	0.18 ± 0.06	0.20 ± 0.08	0.20 ± 0.02	0.39 ± 0.04	2.15 ± 0.20	7.49 ± 0.38	23.15 ± 0.57
ERNOT	0.14 ± 0.02	0.22 ± 0.06	0.27 ± 0.08	0.27 ± 0.04	0.27 ± 0.02	0.26 ± 0.04	0.27 ± 0.03
NeuralOT	33.03 ± 29.80	48.60 ± 27.08	61.31 ± 0.20	61.53 ± 0.09	61.59 ± 0.06	61.63 ± 0.05	61.61 ± 0.07