RIGID INVARIANT SLICED WASSERSTEIN VIA INDEPENDENT EMBEDDINGS

Anonymous authors
Paper under double-blind review

ABSTRACT

Comparing probability measures when their supports are related by an unknown rigid transformation is an important challenge in geometric data analysis, arising in shape matching and machine learning. Classical optimal transport (OT) distances, including Wasserstein and sliced Wasserstein, are sensitive to rotations and reflections, while Gromov-Wasserstein (GW) is invariant to isometries but computationally prohibitive for large datasets. We introduce *Rigid-Invariant Sliced Wasserstein via Independent Embeddings* (RISWIE), a scalable pseudometric that combines the invariance of NP-hard approaches with the efficiency of projection-based OT. RISWIE utilizes data-adaptive bases and matches optimal signed permutations along axes according to distributional similarity to achieve rigid invariance with near-linear complexity in the sample size. We prove bounds relating RISWIE to GW in special cases and empirically demonstrate dimension-independent statistical stability. Our experiments on cellular imaging and 3D human meshes demonstrate that RISWIE outperforms GW in clustering tasks and discriminative capability while significantly reducing runtime.

1 INTRODUCTION

Optimal transport (OT) distances have recently gained popularity in data analysis due to their usefulness for comparing probability measures. In applications where the geometry of the underlying space is important (Peyré & Cuturi, 2019; Santambrogio, 2015) (e.g. geometric data analysis), this role is complicated by the fact that many datasets are embedded in coordinate systems that are not canonically aligned (Besl & McKay, 1992); a rigid transformation of the ambient space may leave the original object unchanged while altering the numerical representation substantially. While rigid transformations preserve pairwise distances, finding an optimal rigid correspondence between two point clouds is computationally intractable (NP-hard), as it requires a search over all possible point permutations (Cela, 2013).

Addressing invariance to rigid transformations has been a challenge shared across shape analysis, graph matching, and manifold learning. Existing methods such as isometry-invariant embeddings (Bronstein et al., 2006) and Gromov–Wasserstein distances (Mémoli, 2011) achieve rigid invariance by ignoring the underlying coordinate system, but this comes at the cost of complex, high-order optimization schemes that limits scalability. On the other hand, projection-based methods lower computational costs by reducing higher dimensional OT to many one-dimensional OT problems, but they lack rigid invariance due to the shared coordinate system in the one-dimensional problem.

Our proposed method retains the efficiency of projection-based OT while separating the invariance problem from the transport computation entirely. This requires computing a geometry-aware coordinate system for each dataset, aligning these coordinates across datasets, and quantifying their agreement. Our method preserves the geometric sensitivity of OT, achieves rigid invariance, and scales efficiently to large sample sizes.

Contributions. Our main contributions are:

(i) We introduce **RISWIE**, a sliced transport distance that combines data-dependent embeddings with optimal signed-permutation alignment to compare measures up to rigid transformations at near-linear cost in the size of the empirical measures.

(ii) We establish theoretical guarantees, including rigid invariance, the pseudometric property, closed-form expressions for Gaussian measures, and explicit bounds relating RISWIE to Gromov–Wasserstein.

(iii) We demonstrate empirical dimension-independent finite-sample convergence for bias and variance.

(iv) We show that RISWIE achieves state-of-the-art runtime with essentially no accuracy tradeoffs in shape partitioning, clustering, and alignment benchmarks.

The remainder of the paper is organized as follows. Section 2 reviews optimal transport and existing techniques. Section 3.1 formalizes the problem setting and introduces our proposed distance function. Section 3.2 establishes its invariance and pseudometric properties, derives closed-form expressions in special cases, and bounds its relationship to Gromov–Wasserstein. Section 3.3 discusses RISWIE's statistical behavior in relation to other optimal transport distances. Section 4 presents synthetic and real-data experiments illustrating the utility of the method, and Section 5 concludes with a discussion of limitations, extensions, and open questions.

2 Preliminaries

We use $\|\cdot\|$ to denote the ℓ_2 norm on $\mathbb{R}^d$, $\mathcal{P}(\mathbb{R}^d)$ the set of Borel probability measures on $\mathbb{R}^d$, and $\mathcal{P}_2(\mathbb{R}^d)$ the subset with finite second moments. Given $\mu, \nu \in \mathcal{P}_2(\mathbb{R}^d)$, the 2-Wasserstein distance is

$$W_2^2(\mu, \nu) = \inf_{\pi \in \Pi(\mu,\nu)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x - y\|^2 \, d\pi(x, y), \tag{1}$$

where $\Pi(\mu, \nu)$ is the set of couplings with marginals μ, ν (Villani, 2008; Santambrogio, 2015). In practice, the above measures are approximated by the empirical sample-based measures

$$\mu_s = \tfrac{1}{s} \sum_{i=1}^{s} \delta_{x_i}, \quad \nu_t = \tfrac{1}{t} \sum_{j=1}^{t} \delta_{y_j},$$

which can be shown to converge weakly as $s, t \to \infty$ by a theorem of Varadarajan (Varadarajan, 1958). For n samples, the computation of this distance scales as $O(n^3 \log n)$, and entropic regularization reduces this to $O(n^2)$ per iteration using Sinkhorn updates (Peyré & Cuturi, 2019). Despite these improvements, Wasserstein remains expensive in high dimensions and sensitive to rigid transformations. While there has been work done to search over all point permutations and orthogonal transformations to make Wasserstein rigid-invariant, this formulation is NP-Hard (Grave et al., 2018).

In one dimension, W_2 admits the closed form

$$W_2^2(\mu, \nu) = \int_0^1 \big(F_\mu^{-1}(t) - F_\nu^{-1}(t)\big)^2 dt,$$

which can be evaluated in $O(n \log n)$ (Villani, 2008). The sliced Wasserstein (SW) distance extends this to higher dimensions by projecting onto directions $\theta \in S^{d-1}$ and averaging:

$$\mathrm{SW}_2^2(\mu, \nu) = \int_{S^{d-1}} W_2^2\big(P_\theta \# \mu, P_\theta \# \nu\big) \, d\theta,$$

where $P_\theta(x) = \langle x, \theta \rangle$ (Rabin et al., 2012; Kolouri et al., 2019). Approximating with L random projections yields $O(Ln \log n)$ scaling (Nietert et al., 2022), but SW is not invariant to rigid transformations since both measures are projected along the same directions.

The Gromov–Wasserstein (GW) distance compares measures without requiring a shared ambient space by aligning their internal distance structures (Mémoli, 2011):

$$\mathrm{GW}_2^2(\mu, \nu) = \inf_{\pi \in \Pi(\mu,\nu)} \iint \big|d_X(x, x') - d_Y(y, y')\big|^2 \, d\pi(x, y) \, d\pi(x', y').$$

While GW is invariant to rigid transformations, it also requires solving an NP-Hard quadratic assignment problem (Cela, 2013; Kravtsova, 2025). Even approximate solvers scale as $O(n^4)$ per iteration, making GW computations scale poorly with sample size (Kerdoncuff et al., 2021).

While existing distances have a trade off between rigid invariance and computational efficiency, we aim to define a distance that preserves intrinsic geometry with straightforward computational scalability. Thus, in what follows, we demonstrate the efficacy of a new distance that preserves the invariance property of GW while maintaining the computational efficiency of projection-based OT.

3 Methodology

We now define a new distance, which we denote as the Rigid-Invariant Sliced Wasserstein via Independent Embeddings (RISWIE) distance. The construction has three components: (i) data-dependent embeddings that map each distribution into a low-dimensional coordinate system derived from its own geometry, (ii) an alignment step that pairs axes across embeddings using signed permutations, and (iii) an aggregation of one-dimensional Wasserstein costs over the matched axes. This design separates invariance from transport problem itself, reducing rigid alignment to a discrete assignment problem while retaining the efficiency of sliced OT. In what follows, we give the precise formulation, prove its invariance and pseudometric properties, and analyze its statistical behavior.

3.1 Problem Formulation

Let $\mu, \nu \in \mathcal{P}_2(\mathbb{R}^d)$ be probability measures. We first need to define an object that formalizes the idea of a rigid transformation.

Definition 1 (Signed Permutation Group)**.** *The* signed permutation group *on k elements is*

$$\mathcal{O}_k^{\pm} := \{R \in \mathbb{R}^{k\times k} : R^\top R = I_k,\ R_{ij} \in \{0, \pm 1\},\ \textit{one nonzero per row/column}\}. \quad (|\mathcal{O}_k^{\pm}| = 2^k\, k!)$$

Equivalently, $\mathcal{O}_k^{\pm} = \{D_\varepsilon P_\pi : \pi \in S_k,\ D_\varepsilon = \mathrm{diag}(\varepsilon_1, \dots, \varepsilon_k),\ \varepsilon_j \in \{\pm 1\}\}$.

In particular, our objective is to construct an invariant distance $D(\mu, \nu)$ such that $D(\mu, \nu) = D((R_1)_\# \mu, (R_2)_\# \nu)$ for any $R_1, R_2 \in \mathcal{O}_d^{\pm}$ where $(f)_\# \mu$ denotes the pushforward of the measure μ by f. In addition, the computation of $D(\mu, \nu)$ should scale in polynomial time with respect to sample size and dimension. Under rigid invariance, $D(\mu, \nu) = 0$ whenever ν is the pushforward of μ by some $R \in \mathcal{O}_k^{\pm}$.

The RISWIE distance defined below can be seen as the minimum cost axis and relative sign pairing across all $2^k k!$ pairings, where the cost is defined as the Wasserstein distance between the distributions embedded on those axes.

Definition 2 (RISWIE Distance)**.** *Let μ, ν be centered probability measures on $\mathbb{R}^{d_1}$ and $\mathbb{R}^{d_2}$, respectively. Let $\phi := (\phi_1, \dots, \phi_k) : \mathbb{R}^{d_1} \to \mathbb{R}^k$ and $\psi := (\psi_1, \dots, \psi_k) : \mathbb{R}^{d_2} \to \mathbb{R}^k$ be fixed embedding functions. Let $\mathcal{O}_k^{\pm}$ denote the group of signed permutation matrices of size $k \times k$. For $R \in \mathcal{O}_k^{\pm}$, define $(R\psi)_j := \varepsilon_j \psi_{\pi(j)}$, where R corresponds to a signed permutation (π, ε).*

The Rigid-Invariant Sliced Wasserstein via Independent Embeddings (RISWIE) distance is defined as

$$D^2(\mu, \nu) := \min_{R \in \mathcal{O}_k^{\pm}} \frac{1}{k} \sum_{j=1}^{k} W_2^2\left((\phi_j)_\# \mu,\ ((R\psi)_j)_\# \nu\right),$$

where W_2 denotes the 2-Wasserstein distance on $\mathbb{R}$ and $(\phi_j)_\# \mu$ is the pushforward of μ under ϕ_j.

For the rest of the paper, we denote the RISWIE distance by D unless stated otherwise. This definition only requires considering the relative sign difference between any two axes that are compared because W_2 is invariant under simultaneous reflection in one dimension. Thus, the minimization is equivalent to evaluating all possible axis pairings together with all possible sign assignments for each pairing. We require the distributions to be centered at 0 (by subtracting off the mean).

The embeddings ϕ_j and ψ_j are user-friendly and may be obtained via linear (e.g. PCA) or nonlinear (e.g. diffusion maps) dimensionality reduction techniques (Coifman & Lafon, 2006), or other data-dependent procedures. This formulation avoids requiring a common projection basis, since alignment is performed directly between the one-dimensional pushforwards of μ and ν.

The group $\mathcal{O}_k^{\pm}$ captures the necessary permutations and sign changes of embedding coordinates, corresponding to orthogonal transformations that preserve the independence of axes. Furthermore, minimization over $\mathcal{O}_k^{\pm}$ is a finite assignment problem solvable in $O(k^3)$ via the Hungarian algorithm, assuming pairwise costs have already been computed (Munkres, 1957).

Algorithm 1: RISWIE Empirical Computation

```
Input: Empirical measures $X = \{x_1, \dots, x_{n_1}\} \subset \mathbb{R}^{d_1}$, $Y = \{y_1, \dots, y_{n_2}\} \subset \mathbb{R}^{d_2}$;
       embeddings $\Phi = (\phi_1, \dots, \phi_k)$, $\Psi = (\psi_1, \dots, \psi_k)$.
Output: $D(X, Y) = D(X, Y)$.
$X \leftarrow \{x_i - \mathrm{mean}(X)\}_{i=1}^{n_1}$;   $Y \leftarrow \{y_i - \mathrm{mean}(Y)\}_{i=1}^{n_2}$
for $\ell = 1, \dots, k$ do
    $A_\ell \leftarrow \big(\phi_\ell(x_1), \dots, \phi_\ell(x_{n_1})\big)$ ;                 // embed $X$ onto axis $\ell$
    $B_\ell \leftarrow \big(\psi_\ell(y_1), \dots, \psi_\ell(y_{n_2})\big)$ ;                 // embed $Y$ onto axis $\ell$
    $\widetilde{A}_\ell \leftarrow \mathrm{sort}(A_\ell)$;  $\widetilde{B}_\ell \leftarrow \mathrm{sort}(B_\ell)$ ;     // sort in ascending order before
for $\ell = 1, \dots, k$ do
    for $m = 1, \dots, k$ do
        $c_{\ell m}^{+} \leftarrow \mathsf{W2sorted}^2\big(\widetilde{A}_\ell,\, \widetilde{B}_m\big)$;
        $c_{\ell m}^{-} \leftarrow \mathsf{W2sorted}^2\big(\widetilde{A}_\ell,\, \mathrm{reverse}(-\,\widetilde{B}_m)\big)$ ;        // reflect and reverse
        $C_{\ell m} \leftarrow \min\{c_{\ell m}^{+},\, c_{\ell m}^{-}\}$ ;                  // best sign for pair $(\ell, m)$
$\pi^\star \leftarrow \arg\min_{\pi \in S_k} \sum_{\ell=1}^{k} C_{\ell,\pi(\ell)}$ ;                  // solved by Hungarian
$Z \leftarrow \sum_{\ell=1}^{k} C_{\ell,\pi^\star(\ell)}$;
return $D(X, Y) \leftarrow \sqrt{Z/k}$;
```

Note: $\mathsf{W2sorted}^2$ assumes its two input vectors are already sorted (ascending). For equal weights, it returns $\frac{1}{N}\sum_{i=1}^{N}(u_i - v_i)^2$ when the two lists are length-N; for unequal lengths/weights, it runs the standard two-pointer monotone coupling in $O(n_1 + n_2)$ time. Pre-sorting each projected list once (above) avoids re-sorting inside every 1D OT call, saving a factor of k. Negating reflects the distribution across 0; reversing ensures the reflected list remains sorted in ascending order.

To analyze time complexity, we take $d := \max\{d_1, d_2\}$ and $n := \max\{n_1, n_2\}$. We also assume that $k \leq d$ and $n \geq d$, as is common in practice.

For PCA embeddings,

$$O\big(\underbrace{nd^2}_{\text{covariances}} + \underbrace{kd^2}_{\text{top-}k\text{ eigens}} + \underbrace{knd}_{\text{projection}} + \underbrace{kn\log n}_{\text{sort once}} + \underbrace{k^2 n}_{k^2\text{ sorted }W_2^2\text{ calls}} + \underbrace{k^3}_{\text{Hungarian}}\big) = O\big(nd^2 + dn\log n\big).$$

For Diffusion Map embeddings,

$$O\big(\underbrace{n^2 d}_{\text{kernel build}} + \underbrace{kn^2}_{\text{top-}k\text{ eigens}} + \underbrace{kn\log n}_{\text{sort once}} + \underbrace{k^2 n}_{k^2\text{ sorted }W_2^2\text{ calls}} + \underbrace{k^3}_{\text{Hungarian}}\big) = O\big(n^2 d\big).$$

Both of the above embedding choices are computationally efficient when used with the proposed scheme, with PCA-RISWIE being nearly linear in the number of samples. With $n \geq d$, these approaches are faster than standard Optimal Transport, Gromov-Wasserstein, and equivalent asymptotically to Sliced Wasserstein with d projection axes. However, because random sampling can perform poorly in higher dimensions, one might instead choose a superlinear number of axes (such as $d \log d$), in which case RISWIE becomes asymptotically faster.

3.2 Theoretical Properties

We verify that RISWIE meets the criteria specified in the preceding sections. The first result establishes rigid invariance under mild conditions on the embedding procedure. We then show that

RISWIE is a pseudometric on $\mathcal{P}_2(\mathbb{R}^d)$. Additionally, we give a closed-form expression for Gaussian measures with PCA embeddings, compare it to the Gromov–Wasserstein distance, and present explicit bounds.

Theorem 1 (Rigid-Invariance)**.** *Let $\mu, \nu \in \mathcal{P}_2(\mathbb{R}^d)$, and $T(x) = Rx + t$ an affine transformation for $R \in O(d)$, $t \in \mathbb{R}^d$. Suppose either:*

- *(i) (PCA) All nonzero eigenvalues of the centered covariance of μ are unique (so μ has finite second moments); or*
- *(ii) (Diffusion map) The embedding returns the same set of eigenvectors (up to sign) for a given matrix (i.e., deterministic eigensolver for fixed input).*

Then

$$D(\mu, \nu) = D(T_{\#}\mu, \nu).$$

In particular, $D(\mu, T_{\#}\mu) = 0$.

While the above theorem characterizes RISWIE as translation invariant for two popular choices of embeddings, RISWIE is not scale-invariant by default. For instance, under PCA embeddings, scaling the input distribution by a factor will scale the marginal distributions induced on each principal axis. However, it is trivial to allow scale-invariance, for example by appropriately choosing the bandwidth of the diffusion maps kernel to be based on the median pairwise distance.

Theorem 2 (Pseudometric)**.** *For any $X, Y, Z \in \mathcal{P}_2(\mathbb{R}^d)$ and for any embedding procedure, the RISWIE distance is a pseudometric.*

Symmetry and non-negativity follow directly from Eq. 1. For the triangle inequality, we define an upper bound on RISWIE by composing the optimal axis matchings and applying the triangle inequality for W_2 with Minkowski's inequality.

Determining whether two sets of points differ by a rigid transformatio is computationally intractable (requiring a search over $n!$ point permutations in the worst case) (Chaudhury et al., 2015). As such, it is unreasonable to expect this property in a computable distance. However, one can show the rigid equivalence property in special cases, such as for Gaussian distributions, as a corollary of the next result, and leave a counterexample to the general property in the appendix.

Theorem 3 (RISWIE Distance for Gaussians under PCA Embeddings)**.** *Let $A \sim \mathcal{N}(\omega_A, \Sigma_A)$ and $B \sim \mathcal{N}(\omega_B, \Sigma_B)$ be Gaussian probability measures on $\mathbb{R}^d$ with finite second moments so that they admit eigendecompositions $\Sigma_A = U_A \Lambda_A U_A^\top$ and $\Sigma_B = U_B \Lambda_B U_B^\top$, where $\Lambda_A = \operatorname{diag}(\lambda_1^A, \ldots, \lambda_d^A)$ and $\Lambda_B = \operatorname{diag}(\lambda_1^B, \ldots, \lambda_d^B)$ with $\lambda_1^A > \cdots > \lambda_d^A \geq 0$ and $\lambda_1^B > \cdots > \lambda_d^B \geq 0$. Denote*

$$\mathbf{a} := \big(\sqrt{\lambda_1^A}, \ldots, \sqrt{\lambda_d^A}\big), \quad \mathbf{b} := \big(\sqrt{\lambda_1^B}, \ldots, \sqrt{\lambda_d^B}\big).$$

Then, the RISWIE distance (using all d PCA axes) admits the closed-form:

$$D^2(A, B) = \frac{1}{d}\|\mathbf{a} - \mathbf{b}\|_2^2.$$

The square roots of the eigenvalues are standard deviations along a principal axis. This result is intuitive given that projecting a Gaussian distribution onto any vector yields another Gaussian.

Theorem 4 (RISWIE–GW Comparison for Gaussians)**.** *Let A and B satisfy the same assumptions as in Theorem 3 and additionally be full rank. Define $\alpha := \min_i(a_i + b_i)$. Then the RISWIE distance under PCA embeddings satisfies:*

(i)

$$D^2(A, B) \;\leq\; \frac{GW_2^2(A, B)}{8d\,\alpha^2} \;+\; \frac{\|\Sigma_A\|_F\,\|\Sigma_B\|_F}{d\,\alpha^2}\left(1 - \frac{1}{\sqrt{d}}\right)$$

(ii)

$$\begin{aligned} D^2(A, B) \;&\leq\; \frac{1}{2\sqrt{d}}\sqrt{GW_2^2(\mu, \nu) - 4\left(\operatorname{tr}(\Lambda_0) - \operatorname{tr}(\Lambda_1)\right)^2 - 4\left(\|\Lambda_0\|_F - \|\Lambda_1\|_F\right)^2} \\ &\leq\; \frac{GW_2(A, B)}{2\sqrt{d}} \end{aligned}$$

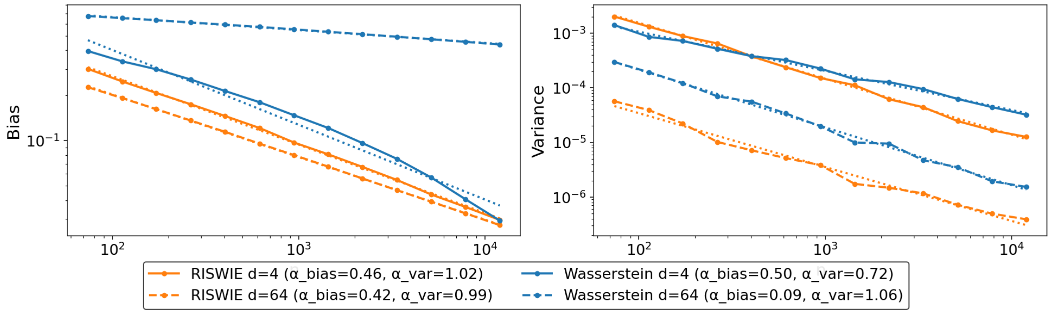


Figure 1: RISWIE-PCA vs. OT: bias (left) and variance (right). RISWIE bias and variance do not become worse in higher dimensions. Ground-truth population distances are calculated with the Gaussian closed form that exists for both distances, and the empirical distances are calculated repeatedly and averaged across sampled distributions. The exponent α corresponds to the empirical decay rate in the log–log plot: we fit a power law of the form $An^{-\alpha}$ to each curve (separately for bias and variance), which estimates the convergence rate.

Gromov-Wasserstein for Gaussians has no closed form, but there have been proven lower and upper bounds for it in the Gaussian case (Salmona et al., 2022). Interestingly, we were able to relate RISWIE^2 to both GW_2 and GW_2^2. The α normalization resolves the difference in units.

3.3 Statistical Properties

As one may expect, $D(\hat{\mu}_n, \hat{\nu}_n) \xrightarrow{\text{a.s.}} D(\mu, \nu)$ as $n \to \infty$ where $\hat{\mu}_n, \hat{\nu}_n$ denote empirical measures of size n drawn i.i.d. from μ, ν (see Theorem 7 in the Appendix). However, a finite sample will always include bias. Consider $D(\mu, \mu) = 0$, yet $\mathbb{E}\big[D(\hat{\mu}_n, \hat{\mu}'_n)\big] > 0$ where $\hat{\mu}'_n$ is an another independent empirical measure of size n drawn i.i.d. from μ. Thus, it is important to consider the bias and variance of $D(\hat{\mu}_n, \hat{\nu}_n)$.

Figure 1 empirically investigates the finite-sample convergence guarantees of the RISWIE-PCA and Wasserstein-2 distances relative to the population distance, which are made possible by the Gaussian closed-form that each distance has. We sample n points from two Gaussian distributions repeatedly, recording the empirical distances between the resulting point clouds and comparing their average to the true population value (bias), as well as their sample variance across trials.

RISWIE exhibits strong empirical statistical behavior–for both low and high-dimensional settings, the bias scales as $O(n^{-1/2})$ and variance as $O(n^{-1})$. In contrast, W_2 converges with a rate of $O(n^{-1/d})$, meaning exponentially many samples are needed to get the same error as in lower dimensions (Weed & Bach, 2017). This is problematic given the computational cost associated with more samples for W_2 and similar distances.

4 Experiments

We evaluated RISWIE with PCA embeddings in classification tasks, using the MPI-FAUST dataset of human meshes (Bogo et al., 2014) and spatially resolved tissue data from the HuBMAP consortium (Hickey et al., 2023). The below numerical results quantify computational efficiency and assess discriminative, clustering, and classification performance relative to existing distances.

We use the Python Optimal Transport (POT) library's implementations of Gromov–Wasserstein (via an approximate solver) and Wasserstein (standard OT) in our comparisons (Flamary et al., 2021; 2024). For FAUST and HuBMAP experiments, we sample 64 axes to ensure robustness against variability in sampling from the unit sphere.

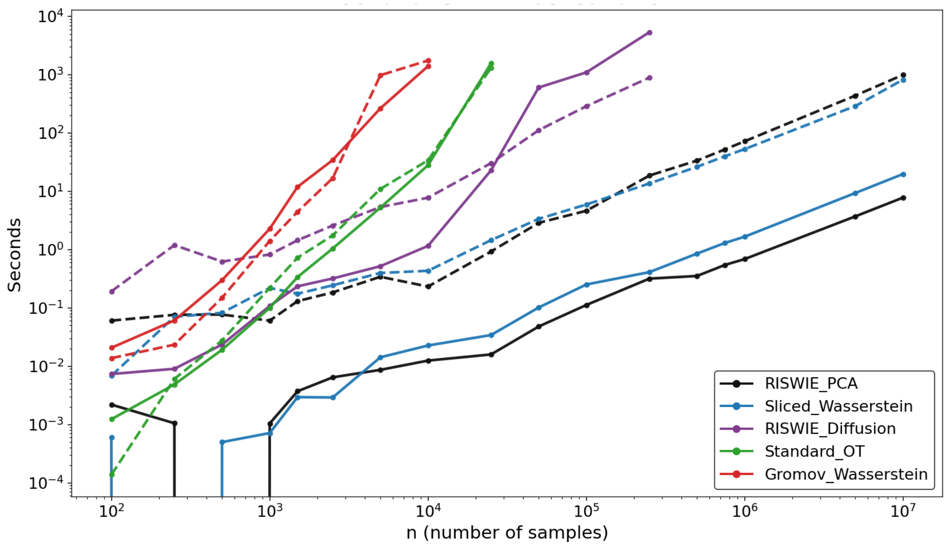


Figure 2: Runtime scaling with sample size n for different distance metrics in $d = 3$ (solid) and $d = 64$ (dashed).

4.1 Computational Efficiency

To evaluate efficiency, we measure wall-clock runtime as a function of the number of sampled points n under two settings: low-dimensional ($d = 3$) and high-dimensional ($d = 64$).

Figure 2 shows that RISWIE-PCA achieves near-linear computational growth in both regimes, much preferred to Wasserstein and Gromov–Wasserstein (GW), while matching the efficiency of Sliced Wasserstein (SW). Wasserstein and Gromov-Wasserstein are computed using the Python Optimal Transport (POT) library. Notably, the computation of GW becomes intractable beyond $\sim 10^4$ samples, and OT beyond $\sim 2.5 \times 10^4$ samples. In contrast, both RISWIE-PCA and RISWIE-Diffusion are significantly more computationally efficient, allowing them to be run with up to 100,000 points per point cloud without any issues, even in high-dimensional settings. A complementary real-data runtime comparison is provided in Table 2.

4.2 Human Pose Alignment and Discrimination

On MPI-FAUST, we treat each registered mesh as a point cloud and compare pairs from the same subject under distinct pose and orientations. As shown in Figure 3, RISWIE aligns the target to the anchor by matching principal axes up to permutation and sign. After alignment, the point clouds overlay closely and their 1D marginals along the first three principal components nearly coincide, indicating robustness to rigid motions.

We further evaluate unsupervised pose clustering on MPI-FAUST (10 subjects × 10 poses). For each method, we compute a 100×100 pairwise distance matrix and embed each mesh as a row. For consistency, all distances are calculated with 1000 subsampled vertices per mesh. This is done for the computability of Wasserstein and Gromov-Wasserstein. However, RISWIE could use all 6890 vertices at negligible extra cost, which is detailed in the appendix.

We evaluate K-Means, Spectral, Agglomerative, and t-SNE–based clustering on mesh embeddings (distance matrix rows), measuring performance with V-measure, ARI, and accuracy. Table 1 reports V-measure: RISWIE matches or outperforms GW and other baselines across clustering strategies. Over our grid of settings, RISWIE surpasses GW in V-measure and NMI in 90.9% of cases and in ARI and accuracy in 100% of cases, while computing the full distance matrix in $\sim$10 seconds versus $\sim$5 hours for GW. Thus, regardless of the clustering method used in unsupervised learning, RISWIE provides consistently strong and efficient performance.

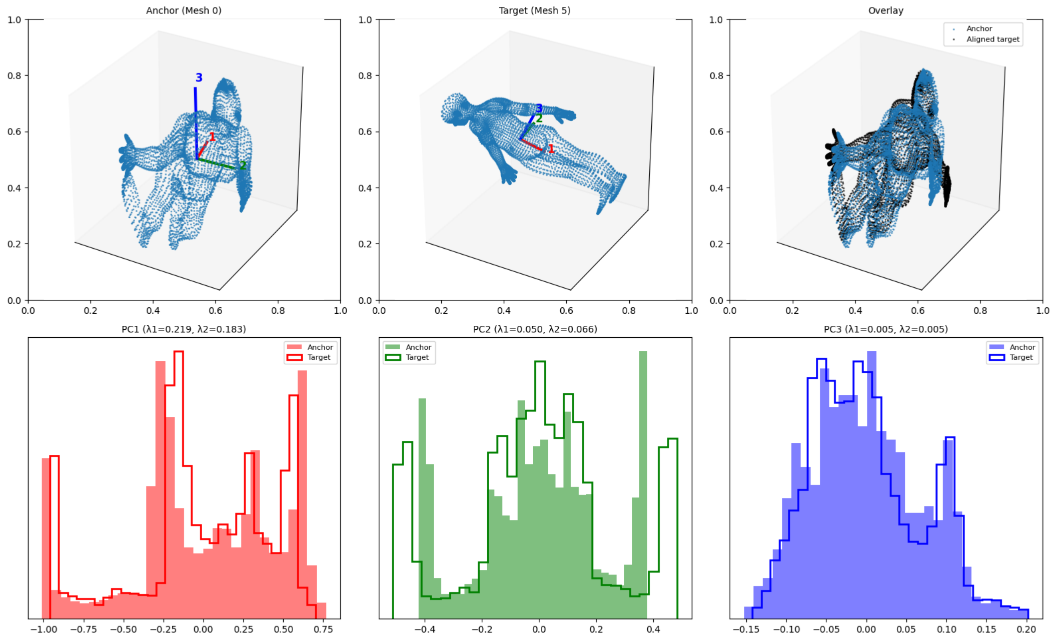


Figure 3: 3D Example of RISWIE alignment. We illustrate how RISWIE aligns two point clouds by matching their marginal distributions along embedded axes. This method naturally extends to higher dimensions. For each axis of the anchor shape, we evaluate all possible pairings with axes of the target, including sign flips (reflections) to minimize the 1D Wasserstein cost. The second row shows the optimal axis matching determined by this process, and we show the poses overlaid with this alignment procedure.

Table 1: V-measure (mean only) by method and distance function on MPI-FAUST pose clustering. See the appendix for standard deviations

Distance Pipeline	Euclidean	Gromov	Wasserstein	RISWIE	Sliced
Agglomerative (avg, precomp)	0.2214	0.6568	0.6715	**0.8094**	0.5478
KMeans (dist rows)	0.3778	0.5930	0.5967	**0.7839**	0.4331
Spectral (RBF of dist)	0.3721	0.5630	0.5757	**0.8138**	0.6291
t-SNE-2D + KMeans	0.4066	0.6649	0.6480	**0.8612**	0.6329
t-SNE-2D + Spectral	0.3907	0.6481	0.6136	**0.8196**	0.6173
AUC-ROC (same-vs-different)	0.6099	0.8929	0.8603	**0.9404**	0.7843

4.3 Tissue Clustering

We evaluate RISWIE on two-dimensional tissue slices of the human small intestine, where each slice is represented as a point cloud of cell coordinates (Hickey et al., 2023), orientated arbitrarily. Ground-truth labels group slices by intestine identity.

Table 2 reports runtime and stack assignment accuracy across distances. For clustering/assignment, we apply a farthest-point seeding strategy with greedy assignment based on intra-cluster distances, with more information available in the appendix. RISWIE achieves sub-second computation and the highest accuracy (95.8%), while Gromov–Wasserstein is slower by over four orders of magnitude. Sliced Wasserstein and classical Wasserstein are faster than GW but substantially less accurate.

Beyond assignment, RISWIE provides stronger discriminative power. Using pairwise distances to score same-intestine versus different-intestine pairs, RISWIE achieves an AUC-ROC of 0.943

Subsample Size	**Distance**	**Time (s)**	**Accuracy**
1000 points	RISWIE	**1**	**95.83**%
	Gromov–Wasserstein	10352	85.42%
	Sliced Wasserstein	2	52.08%
	Wasserstein	111	54.17%
2000 points	RISWIE	**1**	**95.83**%
	Gromov–Wasserstein	56614	**95.83**%
	Sliced Wasserstein	6	47.92%
	Wasserstein	746	47.92%

Table 2: Cells dataset: runtime and stack assignment accuracy for different point subsampling levels.

compared to 0.921 for Gromov–Wasserstein under identical sampling. Since RISWIE scales nearly linearly with sample size, it can exploit larger point sets with little additional cost, which would further improve discriminatory power. However, we again subsample the same number of points for consistency.

5 Discussion

Our empirical results demonstrate that the effective benefits of RISWIE do not degrade accuracy. On tissue slices, RISWIE recovers intestine identity more reliably than GW and achieves the highest stack assignment accuracy while running several orders of magnitude faster. On 3D human meshes, it consistently surpasses GW across clustering methods and evaluation metrics, with distance matrices that can be computed in seconds rather than hours. These results confirm that RISWIE preserves the geometric sensitivity of OT while enforcing rigid invariance, and moreover that it can be deployed on domains where GW is computationally intractable.

RISWIE also recovers a signed axis permutation that aligns axes, which when using PCA can be interpreted as a rigid transformation between eigenspaces. This determines an explicit rotation/reflection aligning two shapes. As a result, we can define boosted variants of any distance function–apply RISWIE's alignment step and then evaluate the distance. These variants inherit rigid invariance without modifying the underlying metric. This makes RISWIE useful both as a standalone distance measure and as a preprocessing step for downstream geometric data analysis.

Two limitations should be noted. First, our method relies on discrete axis matchings. This provides invariance but introduces non-differentiability, limiting direct integration into some deep learning frameworks (Alvarez-Melis & Jaakkola, 2018). We introduce a soft variant in the appendix that replaces hard assignments with probabilistic matchings; however, its empirical performance remains to be fully evaluated. Second, performance depends on the stability of the embedding procedure. When eigengaps are small in PCA or diffusion maps, axis orderings may fluctuate, reducing alignment quality. One possible extension is to treat nearly degenerate eigenspaces as blocks and compare them jointly, though consistent block matching is nontrivial.

By optimizing over a large finite group of signed permutations, RISWIE achieves the robustness of Gromov–Wasserstein while maintaining the scalability of sliced OT. We established its theoretical properties, including pseudometric guarantees, and closed forms for Gaussian measures. Empirically, RISWIE consistently matches or exceeds the accuracy of Gromov–Wasserstein across clustering and alignment tasks, while reducing runtime by several orders of magnitude. These results position RISWIE as a practical distance for large-scale geometric data analysis and a foundation for future work on invariant transport methods.

6 Ethics and Reproducibility Statement

This work relies solely on public, de-identified datasets released under their original licenses. No new data were collected, and no human subjects or sensitive information are involved. To the best of our knowledge, the research complies fully with the ICLR Code of Ethics and raises no ethical concerns.

We describe the method in full detail in the main text, provide complete proofs of theoretical results in the appendix, and use only publicly available datasets. An anonymized code release is included in the supplementary material to reproduce the main experiments.

References

David Alvarez-Melis and Tommi S. Jaakkola. Gromov-wasserstein alignment of word embedding spaces, 2018. URL `https://arxiv.org/abs/1809.00013`.

P.J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992. doi: 10.1109/34.121791.

Federica Bogo, Javier Romero, Matthew Loper, and Michael J. Black. FAUST: Dataset and evaluation for 3D mesh registration. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Piscataway, NJ, USA, June 2014. IEEE.

Alexander M. Bronstein, Michael M. Bronstein, and Ron Kimmel. Efficient computation of isometry-invariant distances between surfaces. *SIAM Journal on Scientific Computing*, 28 (5):1812–1836, 2006. doi: 10.1137/050639296. URL `https://doi.org/10.1137/050639296`.

E. Cela. *The Quadratic Assignment Problem: Theory and Algorithms*. Combinatorial Optimization. Springer US, 2013. ISBN 9781475727883. URL `https://books.google.hu/books?id=cpMCswEACAAJ`.

{K. N.} Chaudhury, Y. Khoo, and A. Singer. Global registration of multiple point clouds using semidefinite programming. *SIAM Journal on Optimization*, 25(1):468–501, 2015. ISSN 1052-6234. doi: 10.1137/130935458. Publisher Copyright: © 2015 Society for Industrial and Applied Mathematics.

Ronald R. Coifman and Stéphane Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, 2006. doi: 10.1016/j.acha.2006.04.006. Special Issue: Diffusion Maps and Wavelets.

Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Zeghal Alaya, Arnaud 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`.

Rémi Flamary, Cédric Vincent-Cuaz, Nicolas Courty, Alexandre Gramfort, Oleksii Kachaiev, Huy Quang Tran, Laurène David, Clément Bonet, Nathan Cassereau, Théo Gnassounou, Eloi Tanguy, Julie Delon, Antoine Collas, Sonia Mazelet, Laetitia Chapel, Tanguy Kerdoncuff, Xizheng Yu, Matthew Feickert, Paul Krzakala, Tianlin Liu, and Eduardo Fernandes Montesuma. Pot python optimal transport (version 0.9.5), 2024. URL `https://github.com/PythonOT/POT`.

Edouard Grave, Armand Joulin, and Quentin Berthet. Unsupervised alignment of embeddings with wasserstein procrustes, 2018. URL `https://arxiv.org/abs/1805.11222`.

J. Hickey, C. Caraccio, G. Nolan, and HuBMAP Consortium. Organization of the human intestine at single cell resolution. HuBMAP Consortium, 2023.

Tanguy Kerdoncuff, Rémi Emonet, and Marc Sebban. Sampled Gromov Wasserstein. *Machine Learning*, 2021. doi: 10.1007/s10994-021-06035-1. URL `https://hal.science/hal-03232509`.

Soheil Kolouri, Kimia Nadjahi, Umut Simsekli, Roland Badeau, and Gustavo Rohde. Generalized sliced wasserstein distances. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Natalia Kravtsova. The np-hardness of the gromov-wasserstein distance, 2025. URL `https://arxiv.org/abs/2408.06525`.

Facundo Mémoli. Gromov-wasserstein distances and the metric approach to object matching. *Foundations of Computational Mathematics*, 11(4):417–487, 2011. doi: 10.1007/s10208-011-9093-5.

James Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1):32–38, 1957. doi: 10.1137/0105003.

Sloan Nietert, Ziv Goldfeld, Ritwik Sadhu, and Kengo Kato. Statistical, robustness, and computational guarantees for sliced wasserstein distances. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 28179–28193. Curran Associates, Inc., 2022. URL `https://proceedings.neurips.cc/paper_files/paper/2022/file/b4bc180bf09d513c34ecf66e53101595-Paper-Conference.pdf`.

Gabriel Peyré and Marco Cuturi. Computational optimal transport: With applications to data science. *Foundations and Trends in Machine Learning*, 11(5-6):355–607, 2019. doi: 10.1561/2200000073.

Julien Rabin, Gabriel Peyré, Julie Delon, and Marc Bernot. Wasserstein barycenter and its application to texture mixing. In Alfred M. Bruckstein, Bart M. ter Haar Romeny, Alexander M. Bronstein, and Michael M. Bronstein (eds.), *Scale Space and Variational Methods in Computer Vision (SSVM)*, pp. 435–446, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. doi: 10.1007/978-3-642-24785-9_37.

Antoine Salmona, Julie Delon, and Agnès Desolneux. Gromov-Wasserstein Distances between Gaussian Distributions. *Journal of Applied Probability*, 59(4), December 2022. URL `https://hal.science/hal-03197398`.

Filippo Santambrogio. *Optimal Transport for Applied Mathematicians*. Birkhäuser, 2015.

Maksym Shamrai. Perturbation analysis of singular values in concatenated matrices, 2025. URL `https://arxiv.org/abs/2505.01427`.

V. S. Varadarajan. On the convergence of sample probability distributions. *Sankhyā: The Indian Journal of Statistics (1933-1960)*, 19(1/2):23–26, 1958. ISSN 00364452. URL `http://www.jstor.org/stable/25048365`.

Titouan Vayer, Laetitia Chapel, Rémi Flamary, Romain Tavenard, and Nicolas Courty. Optimal transport for structured data with application on graphs, 2019. URL `https://arxiv.org/abs/1805.09114`.

C. Villani. *Optimal Transport: Old and New*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, 2008. ISBN 9783540710509. URL `https://books.google.com/books?id=hV8o5R7_5tkC`.

Jonathan Weed and Francis Bach. Sharp asymptotic and finite-sample rates of convergence of empirical measures in wasserstein distance, 2017. URL `https://arxiv.org/abs/1707.00087`.

A Appendix

A.1 RISWIE Variants

To facilitate differentiable optimization, we define a soft relaxation of RISWIE, denoted SRISWIE, which replaces hard axis matching with entropic transport over a soft cost matrix. This provides a continuous approximation that is always rigid invariant and converges to RISWIE in the limit as $\beta \to \infty$ and $\varepsilon \to 0$.

Definition 3 (Soft RISWIE (SRISWIE) Distance)**.** *Let μ, ν be centered probability measures in $\mathcal{P}_2(\mathbb{R}^d)$, and again let $\varphi = (\varphi_1, \ldots, \varphi_k)$, $\psi = (\psi_1, \ldots, \psi_k)$ be fixed embedding functions.*

For each $(j, m) \in \{1, \ldots, k\}^2$, define

$$C_{jm}^{+} := W_2^2((\varphi_j)_{\#}\mu,\ (\psi_m)_{\#}\nu), \quad C_{jm}^{-} := W_2^2((\varphi_j)_{\#}\mu,\ (-\psi_m)_{\#}\nu)$$

and set the cost of a pairing as:

$$\tilde{C}_{jm} := w_{jm} C_{jm}^{+} + (1 - w_{jm}) C_{jm}^{-}, \quad \textit{where} \quad w_{jm} := \frac{1}{1 + \exp\big(\beta(C_{jm}^{+} - C_{jm}^{-})\big)}.$$

Let $\tilde{C} \in \mathbb{R}^{k \times k}$ be the resulting soft cost matrix. Define the SRISWIE distance as:

$$\mathrm{SRISWIE}^2(\mu, \nu; \varepsilon, \beta) = \min_{\mathbf{P} \in \mathcal{U}_k} \left\{ \frac{1}{k} \sum_{j=1}^{k} \sum_{m=1}^{k} \mathbf{P}_{jm} \tilde{C}_{jm} + \varepsilon \sum_{j=1}^{k} \sum_{m=1}^{k} \mathbf{P}_{jm} \log \mathbf{P}_{jm} \right\}$$

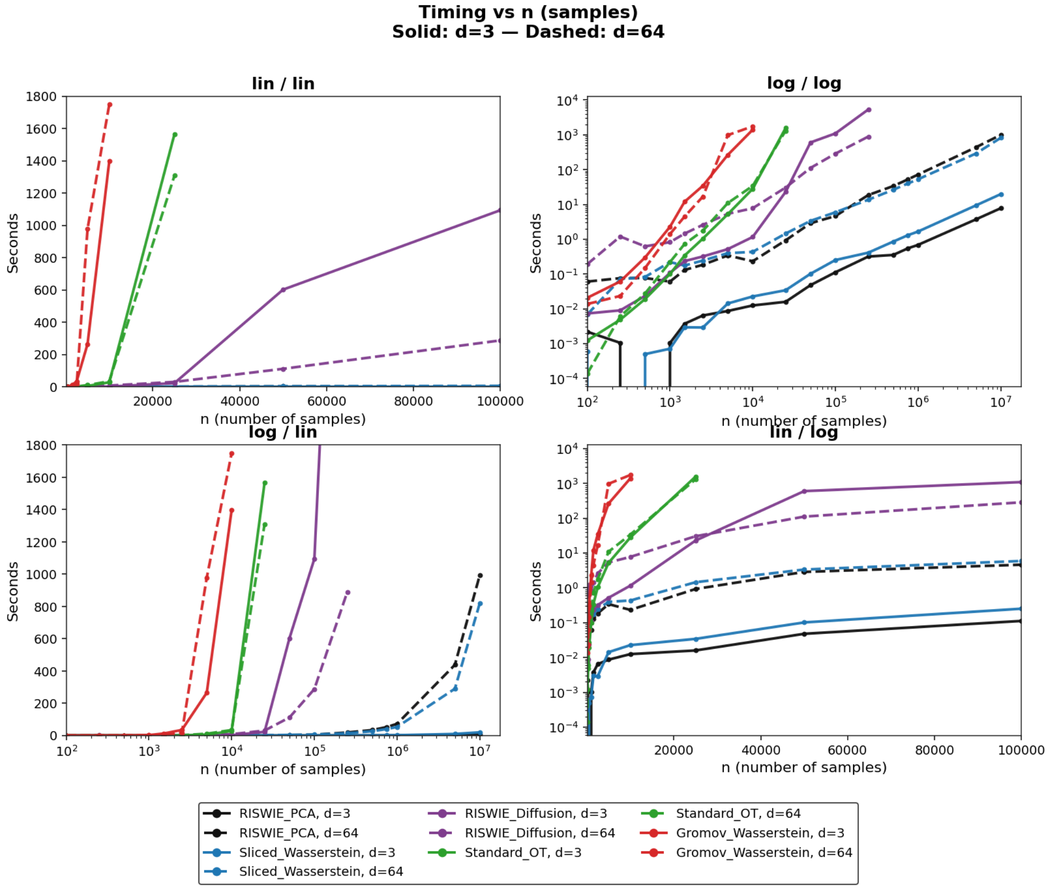
where $\mathcal{U}_k$ denotes the set of $k \times k$ doubly stochastic matrices.

This variant replaces the hard signed-permutation matching over $O_k^{\pm}$ with an entropic optimal transport problem and handles axis reflections with a smooth soft-min.

Performance of SRISWIE on more sophisticated deep learning tasks is still to be evaluated. On the FAUST dataset clustering task, SRISWIE was able to compute a 100×100 distance matrix between meshes with the full 6890 points in 34 seconds. Downstream spectral clustering on these meshes embedded as a row/column of the distance matrix yielded a V-measure of 0.8541.

We also extract the optimal axis pairing and optimal relative sign for each axis pairing to align shapes before computing other distances such as Wasserstein or Sliced Wasserstein. We call these distances Boosted Optimal Transport and Boosted Sliced Wasserstein, respectively. See Section A.4 for comparisons of how these boosted distances perform in solving the balanced partitioning problem.

A.2 Timing Results



For our timing experiments, we set the number of projection axes for Sliced Wasserstein to $\max(10, d \log d)$ and the number of embedding functions of RISWIE-PCA to d. The former is done to make Sliced Wasserstein robust to bad sampling directions as they are not data dependent. For diffusion-based RISWIE, we implement diffusion maps by building a sparse neighborhood graph with $k = \lceil d \log n \rceil$ neighbors, then apply heat-kernel affinities and symmetric normalization before computing the top d eigenvectors.

A.3 FAUST Full Experiment

Table **??** reports performance across clustering pipelines, where abbreviations like "avg, precomp", "dist rows", and "RBF of dist" refer to specific clustering setups described in the table caption and glossary.

Table 3: Description of clustering pipelines used in the experiments.

Pipeline Label	**Description**
KMeans (dist rows)	KMeans on rows of the pairwise distance matrix as Euclidean vectors.
KMedoids (precomputed dist)	KMedoids using the full precomputed pairwise distance matrix.
Agglomerative (avg, precomp)	Average-linkage agglomerative clustering on the precomputed distance matrix.
Spectral (RBF of dist)	Spectral clustering using an RBF kernel of the distance matrix: $A_{ij} = \exp\left(-D_{ij}^2/(2\sigma^2)\right)$ with $\sigma = \mathrm{median}(D[D > 0])$.
MDS-2D + KMeans	2D MDS embedding of distances followed by KMeans.
MDS-3D + KMeans	3D MDS embedding of distances followed by KMeans.
MDS-2D + Spectral	2D MDS embedding, RBF kernel on embedded points, then Spectral clustering.
t-SNE-2D + KMeans	2D t-SNE on precomputed distances (perplexity 10), then KMeans.
t-SNE-3D + KMeans	3D t-SNE on precomputed distances, then KMeans.
t-SNE-2D + Spectral	2D t-SNE followed by RBF kernel and Spectral clustering.
t-SNE-3D + Spectral	3D t-SNE followed by RBF kernel and Spectral clustering.

Table 4: V-measure (mean ± std) by method and distance function on MPI-FAUST pose clustering.

Distance	Euclidean	Gromov	OT	RISWIE	Sliced
Pipeline					
Agglomerative (avg, precomp)	0.2214 ± 0.0252	0.6568 ± 0.0586	0.6715 ± 0.0164	**0.8094 ± 0.0268**	0.5478 ± 0.0346
KMeans (dist rows)	0.3778 ± 0.0257	0.5930 ± 0.0478	0.5967 ± 0.0259	**0.7839 ± 0.0192**	0.4331 ± 0.0292
Spectral (RBF of dist)	0.3721 ± 0.0248	0.5630 ± 0.0412	0.5757 ± 0.0225	**0.8138 ± 0.0190**	0.6291 ± 0.0387
t-SNE-2D + KMeans	0.4066 ± 0.0274	0.6649 ± 0.0447	0.6480 ± 0.0264	**0.8612 ± 0.0270**	0.6329 ± 0.0351
t-SNE-2D + Spectral	0.3907 ± 0.0308	0.6481 ± 0.0482	0.6136 ± 0.0215	**0.8196 ± 0.0183**	0.6173 ± 0.0275

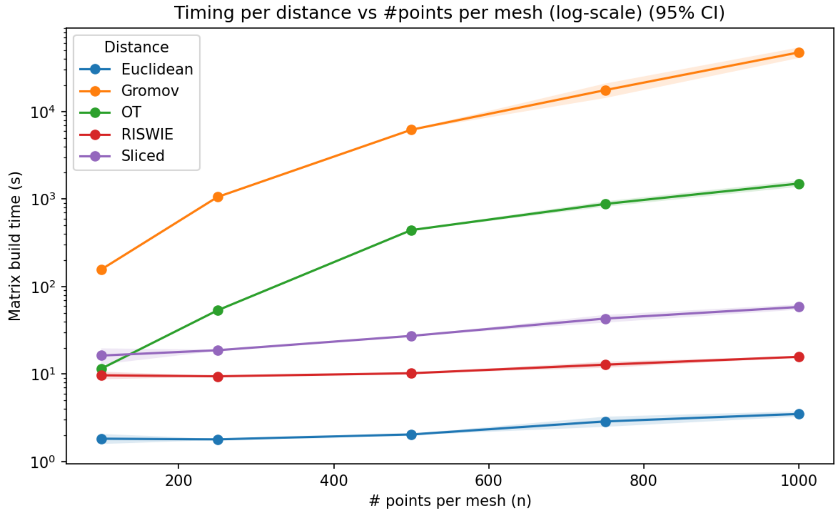


Figure 4: Matrix-build time versus number of points per mesh n (log-scale). Markers show means across repeats; shaded ribbons are 95% CIs. Euclidean is fastest; RISWIE grows gently with n and stays well below Sliced/OT, while Gromov–Wasserstein is the slowest by far.

Table 5: Accuracy by Method and Distance Function

Method	RISWIE	Gromov	OT	Euclidean	Sliced
KMeans (dist rows)	**0.7200**	0.5700	0.5600	0.3500	0.3600
Spectral (RBF of dist)	**0.7800**	0.7500	0.5300	0.3200	0.6000
Agglomerative (avg, precomp)	**0.7200**	0.5300	0.4600	0.1400	0.4500
MDS-2D + KMeans	**0.7300**	0.5800	0.5400	0.3100	0.4200
MDS-2D + Spectral	**0.5800**	0.4600	0.4300	0.3200	0.3300
MDS-3D + KMeans	**0.7800**	0.7000	0.5000	0.3200	0.4300
MDS-3D + Spectral	**0.7300**	0.6700	0.5200	0.3100	0.4200
t-SNE-2D + KMeans	**0.8700**	0.8200	0.6500	0.4100	0.6100
t-SNE-2D + Spectral	**0.7200**	0.6800	0.5600	0.4100	0.5300
t-SNE-3D + KMeans	**0.8000**	0.7500	0.5300	0.3500	0.5200
t-SNE-3D + Spectral	**0.7600**	0.6800	0.5700	0.3000	0.5000

Table 6: V-measure by Method and Distance Function

Method	RISWIE	Gromov	OT	Euclidean	Sliced
KMeans (dist rows)	**0.8058**	0.6802	0.5957	0.4007	0.4373
Spectral (RBF of dist)	0.8238	**0.8303**	0.5790	0.3220	0.6437
Agglomerative (avg, precomp)	**0.8082**	0.7420	0.6763	0.2137	0.6092
MDS-2D + KMeans	**0.7454**	0.6721	0.5506	0.2986	0.4386
MDS-2D + Spectral	**0.7065**	0.5958	0.4921	0.3161	0.3510
MDS-3D + KMeans	**0.8231**	0.7879	0.5818	0.2870	0.4892
MDS-3D + Spectral	**0.7789**	0.7422	0.5700	0.3162	0.4676
t-SNE-2D + KMeans	**0.8829**	0.8577	0.6779	0.4138	0.6246
t-SNE-2D + Spectral	**0.8291**	0.7896	0.6357	0.3954	0.6022
t-SNE-3D + KMeans	**0.7832**	0.7606	0.5847	0.3486	0.5281
t-SNE-3D + Spectral	**0.7754**	0.7039	0.5843	0.2856	0.4686

A.4 Cells Full Experiment

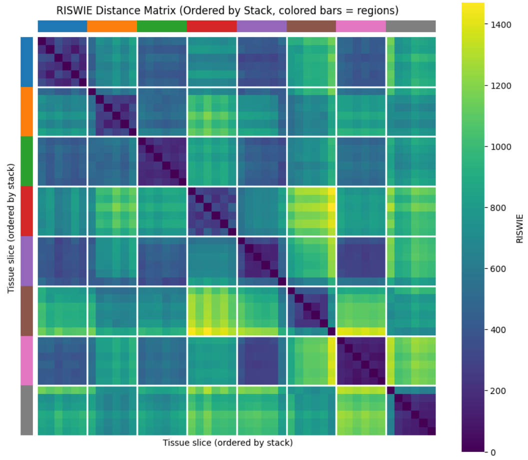


Figure 5: RISWIE Distance matrix for the HuBMAP tissue slices. Each block along the diagonal corresponds to slices from the same tissue stack. Within a block, RISWIE distances are consistently near zero, indicating strong invariance to small perturbations and local alignment of slices from the same sample. Across blocks, RISWIE captures larger geometric variation between tissues from different regions, producing higher inter-block distances.

Table 7: Adjusted Rand Index (ARI) by Method and Distance Function

Method	RISWIE	Gromov	OT	Euclidean	Sliced
KMeans (dist rows)	**0.5844**	0.3910	0.3673	0.1359	0.1618
Spectral (RBF of dist)	**0.6825**	0.6154	0.3312	0.0944	0.4277
Agglomerative (avg, precomp)	**0.5526**	0.4197	0.3796	0.0171	0.3498
MDS-2D + KMeans	**0.5454**	0.3906	0.3067	0.0486	0.1723
MDS-2D + Spectral	**0.4363**	0.2881	0.2318	0.0696	0.1078
MDS-3D + KMeans	**0.6531**	0.5645	0.3336	0.0499	0.2214
MDS-3D + Spectral	**0.5576**	0.5028	0.3427	0.0732	0.2026
t-SNE-2D + KMeans	**0.7965**	0.7416	0.4946	0.1765	0.4116
t-SNE-2D + Spectral	**0.6436**	0.5718	0.4102	0.1480	0.3569
t-SNE-3D + KMeans	**0.6529**	0.6085	0.3552	0.1013	0.3136
t-SNE-3D + Spectral	**0.6107**	0.4572	0.3301	0.0584	0.2254

Table 8: Normalized Mutual Information (NMI) by Method and Distance Function

Method	RISWIE	Gromov	OT	Euclidean	Sliced
KMeans (dist rows)	**0.8058**	0.6802	0.5957	0.4007	0.4373
Spectral (RBF of dist)	0.8238	**0.8303**	0.5790	0.3220	0.6437
Agglomerative (avg, precomp)	**0.8082**	0.7420	0.6763	0.2137	0.6092
MDS-2D + KMeans	**0.7454**	0.6721	0.5506	0.2986	0.4386
MDS-2D + Spectral	**0.7065**	0.5958	0.4921	0.3161	0.3510
MDS-3D + KMeans	**0.8231**	0.7879	0.5818	0.2870	0.4892
MDS-3D + Spectral	**0.7789**	0.7422	0.5700	0.3162	0.4676
t-SNE-2D + KMeans	**0.8829**	0.8577	0.6779	0.4138	0.6246
t-SNE-2D + Spectral	**0.8291**	0.7896	0.6357	0.3954	0.6022
t-SNE-3D + KMeans	**0.7832**	0.7606	0.5847	0.3486	0.5281
t-SNE-3D + Spectral	**0.7754**	0.7039	0.5843	0.2856	0.4686

We compute the all-pairs RISWIE distance matrix between point clouds from different tissue types and vertical slices. Each block in the matrix compares all slices of one tissue to all slices of another. Since each slice may be arbitrarily rotated or reflected, a rigid-invariant distance should yield low pairwise values within diagonal blocks (same tissue), despite variations in orientation or sampling. Figure 5 highlights RISWIE's robustness to such transformations, showing consistently low intra-tissue distances.

To evaluate RISWIE's effectiveness in recovering biologically meaningful groupings, we perform balanced partitioning of tissue slices into spatial stacks based on the computed pairwise distances between tissue slices. We use a farthest-point seeding strategy to encourage diversity among initial stack centers and apply a greedy assignment procedure to add tissue slices to a cluster that they are most similar to.

In other words, we are trying to minimize

$$\mathcal{L}(\mathcal{S}_1,\ldots,\mathcal{S}_K)=\sum_{k=1}^{K}\sum_{\substack{i,j\in\mathcal{S}_k\\ i<j}} D_{\text{Input Distance}}(X_i,X_j)$$

where $\mathcal{X}=\{X_1,X_2,\ldots,X_n\}$ is the set of tissue slices and we want to partition them into stacks $\mathcal{S}_1,\ldots,\mathcal{S}_K$, each of size n/K.

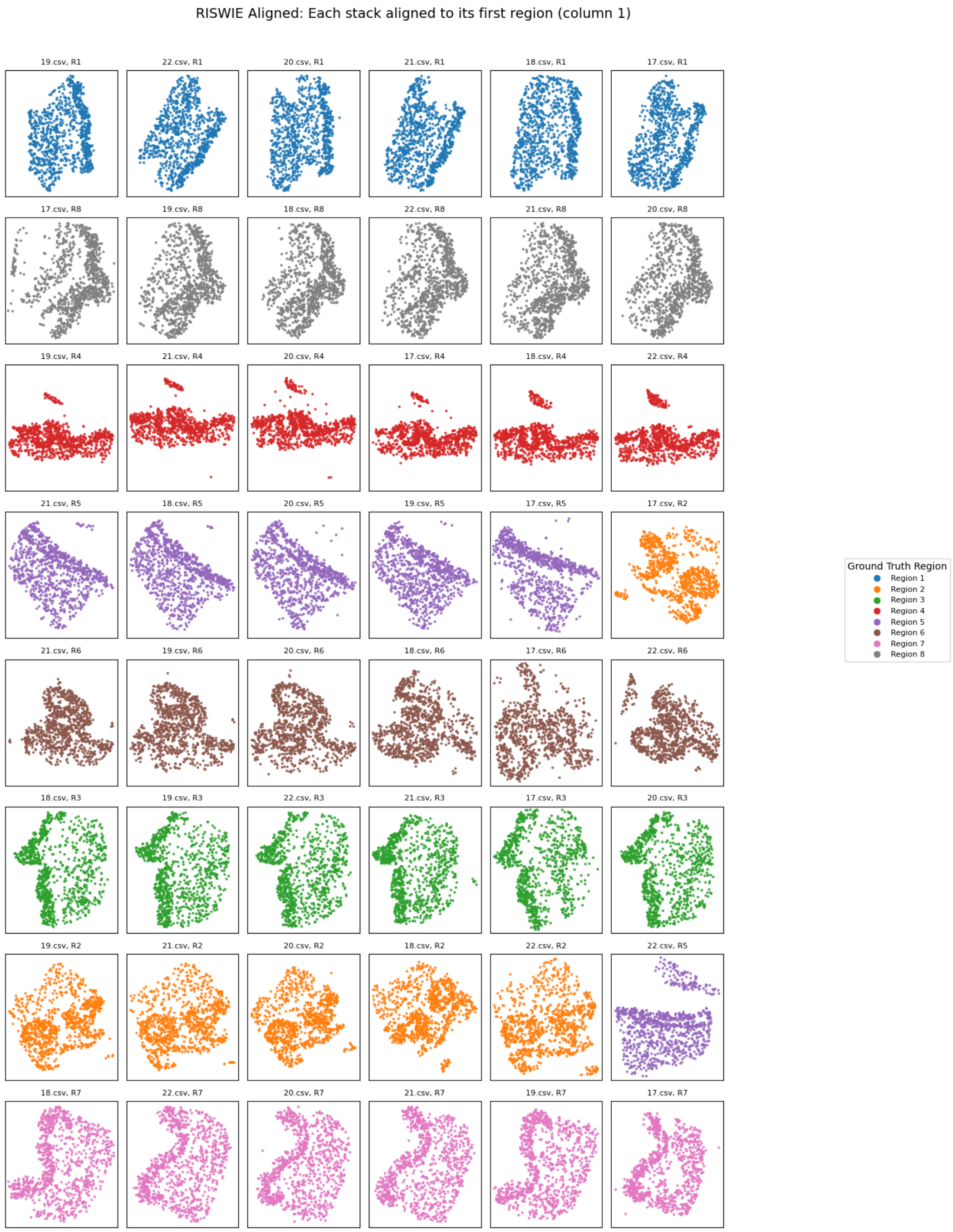
Table 9: Clustering performance using RISWIE with no subsampling. Accuracy, V-measure, ARI, and NMI are reported across clustering pipelines.

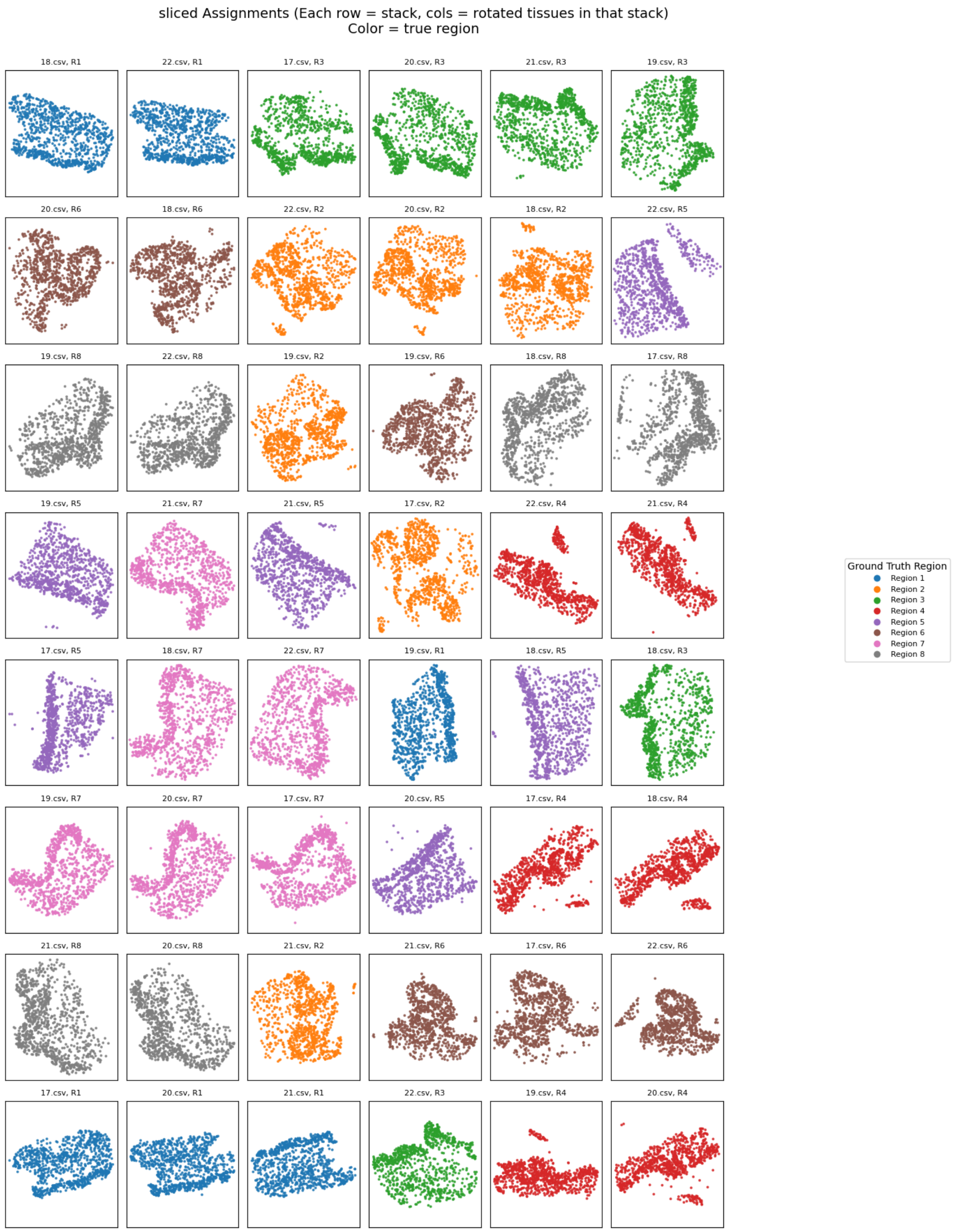
Method	Accuracy	V-measure	ARI	NMI
KMeans (dist rows)	0.7500	0.8469	0.6446	0.8469
KMedoids (precomputed dist)	0.8200	0.8296	0.6966	0.8296
Spectral (RBF of dist)	0.7900	0.8343	0.6921	0.8343
Agglomerative (avg, precomp)	0.7800	0.8549	0.6655	0.8549
MDS-2D + KMeans	0.7500	0.7756	0.5934	0.7756
MDS-2D + KMedoids	0.7500	0.7666	0.5878	0.7666
MDS-2D + Spectral	0.6600	0.7531	0.5121	0.7531
MDS-3D + KMeans	0.7300	0.7517	0.5608	0.7517
MDS-3D + KMedoids	0.7100	0.7541	0.5776	0.7541
MDS-3D + Spectral	0.7200	0.7843	0.5382	0.7843
t-SNE-2D + KMeans	0.8300	0.8498	0.7348	0.8498
t-SNE-2D + KMedoids	0.8300	0.8498	0.7348	0.8498
t-SNE-2D + Spectral	0.7000	0.8339	0.6081	0.8339
t-SNE-3D + KMeans	0.7600	0.7850	0.6276	0.7850
t-SNE-3D + KMedoids	0.7700	0.7633	0.6116	0.7633
t-SNE-3D + Spectral	0.6400	0.7145	0.4688	0.7145

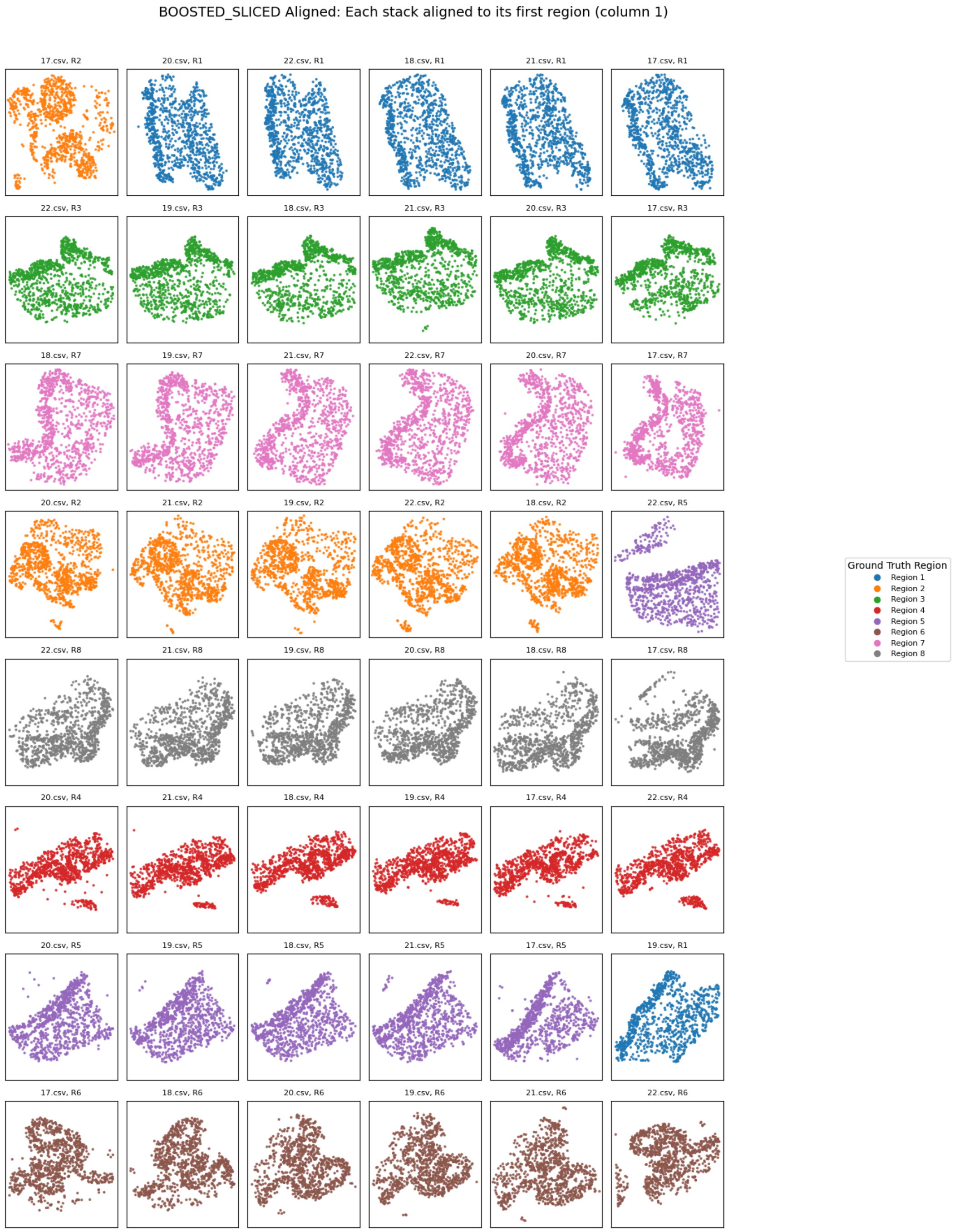
Algorithm 2: Stack Assignment via RISWIE, Farthest-Point Seeding, and Greedy Assignment

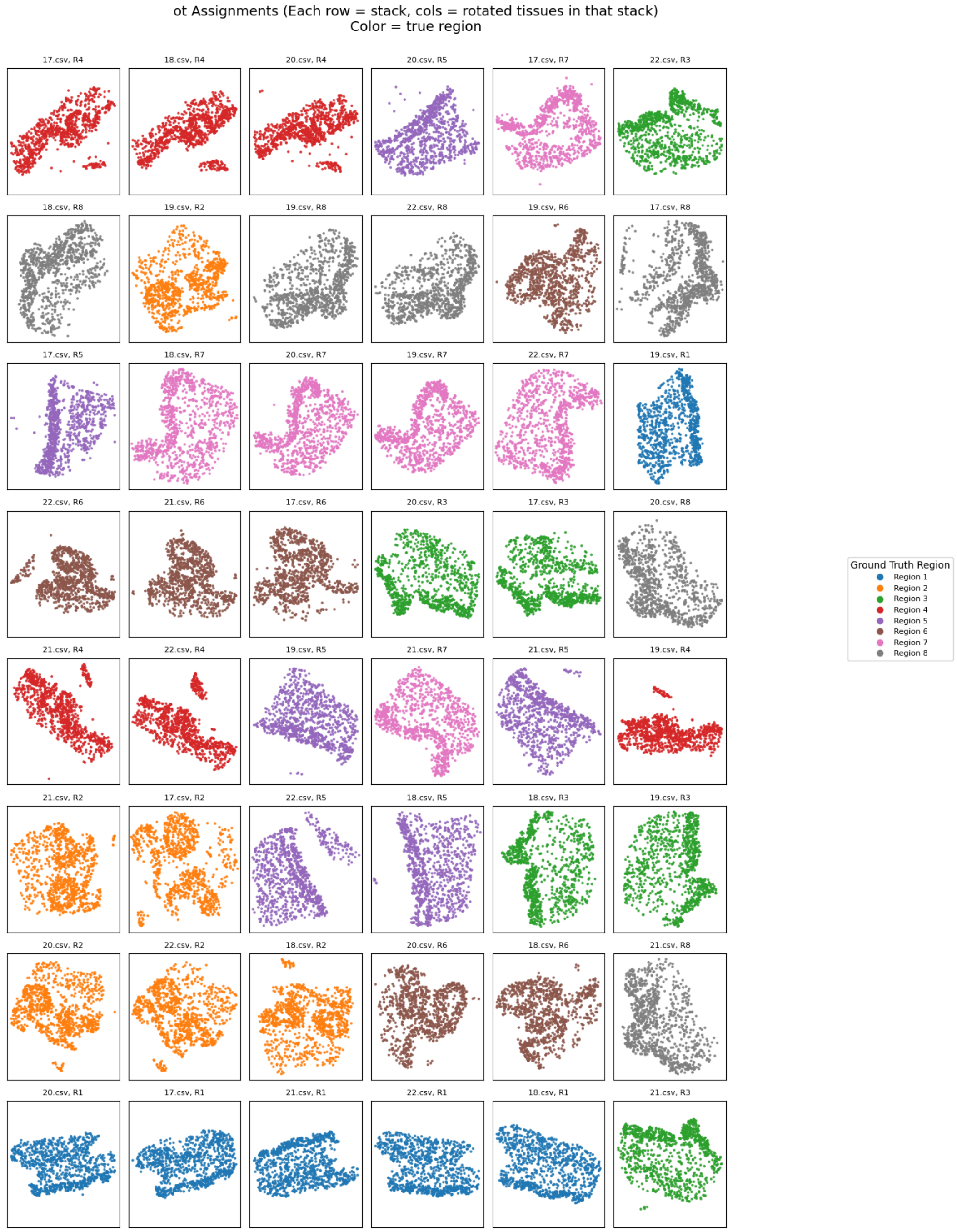
```
Input: Set of $n = 48$ regions (point clouds) $\{X_i\}$
Output: Optimal grouping of regions into $K$ balanced stacks
Step 1: Compute Distance Matrix
for $i = 1$ to $n$ do
    for $j = i + 1$ to $n$ do
        $D_{ij} \leftarrow$ RISWIE_distance$(X_i, X_j)$ ;
        $D_{ji} \leftarrow D_{ij}$ ;
Step 2: Farthest Point Seeding and Greedy Assignment
for $s = 1$ to $n$                    // Try each region as first seed do
    $S \leftarrow [s]$                                        // Seed indices
    while $|S| < K$ do
        Select $t = \arg\max_{t \notin S} \min_{u \in S} D_{tu}$ ;
        Append $t$ to $S$ ;
    Initialize $K$ stacks, each with one seed from $S$ ;
    while unassigned regions remain do
        for each unassigned region r, and each stack k not full do
            Compute cost $c_{r,k} = \sum_{b \in \text{stack}_k} D_{r,b}$ ;
        Assign $r^*$ to stack $k^*$ minimizing $c_{r,k}$, breaking ties arbitrarily ;
    Compute total within-stack sum $C_s = \sum_{k=1}^{K} \sum_{i,j \in S_k,\, i<j} D_{ij}$ ;
    Store stacks and $C_s$ ;
Select the stack assignment with lowest within-stack sum, summed across all stacks: $\sum_s C_s$ ;
Step 3 (Optional): Random Seeds
Optionally repeat the greedy assignment with some number random initializations of $K$ stacks
  and take the lowest cost stack assignment across all completed stacks.
```

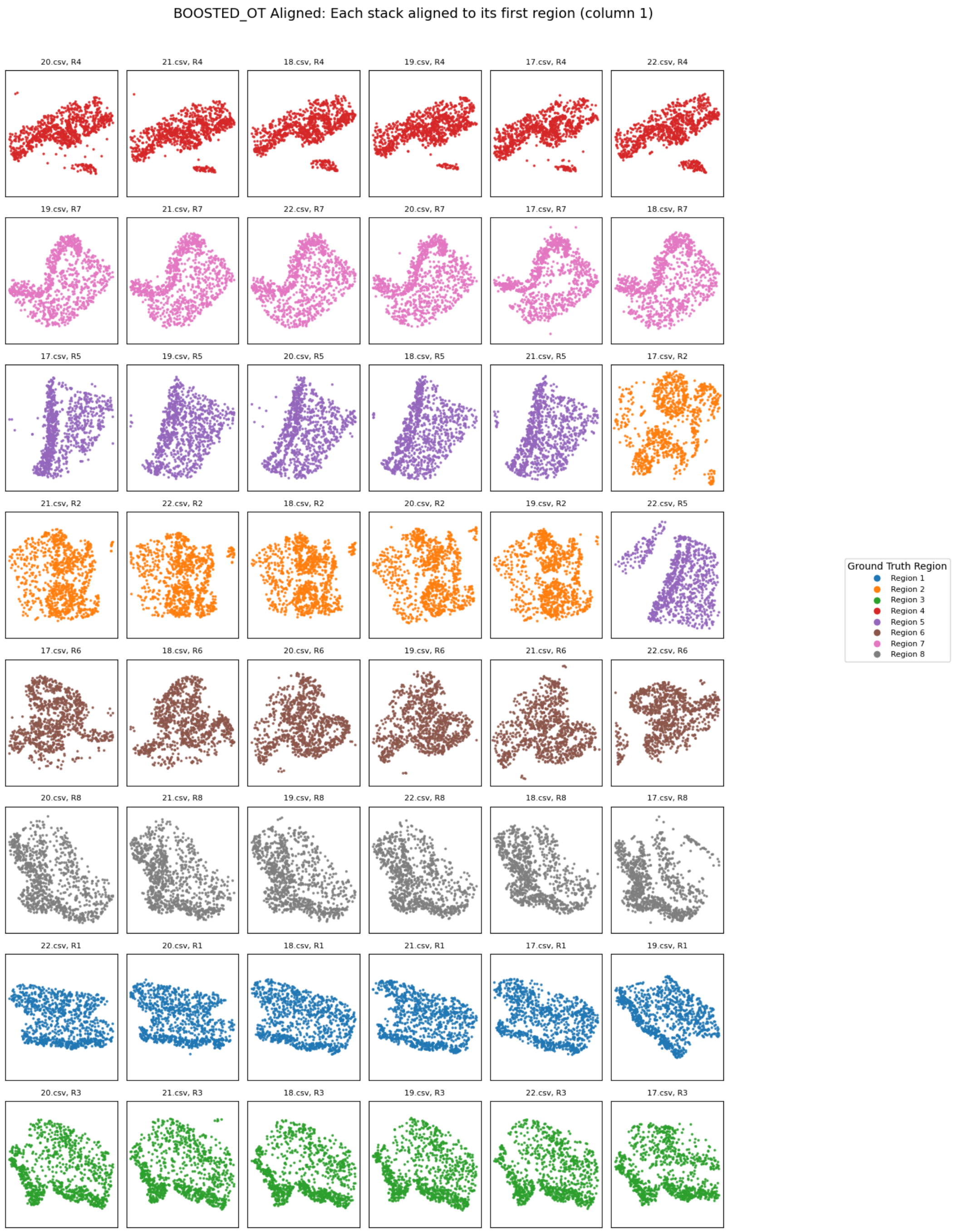
The assignment accuracy reported reflects the best label alignment between predicted and ground truth stacks, computed via Hungarian matching.

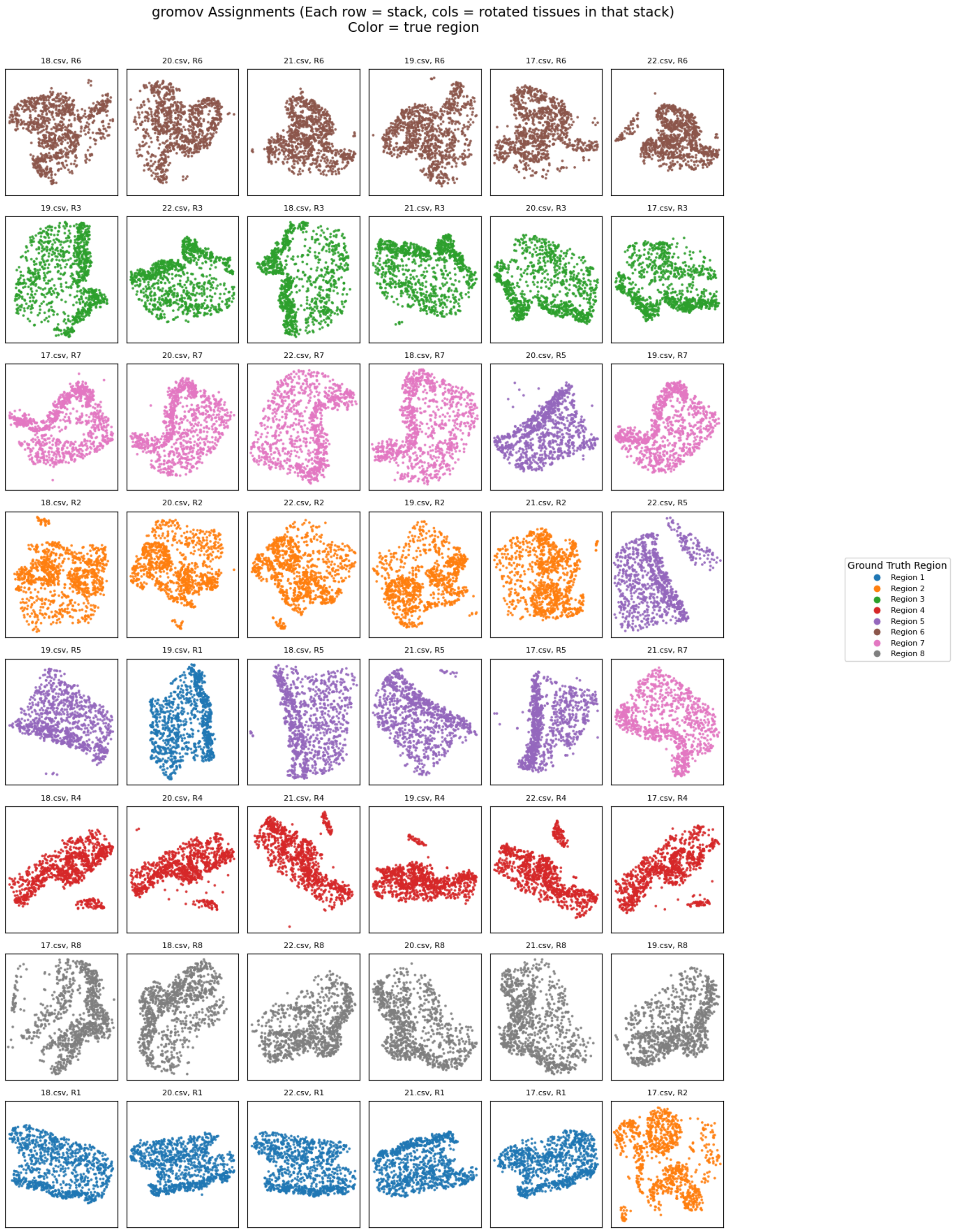

RISWIE Aligned: Each stack aligned to its first region (column 1)
19.csv, R1
22.csv, R1
20.csv, R1
21.csv, R1
18.csv, R1
17.csv, R1
17.csv, R8
19.csv, R8
18.csv, R8
22.csv, R8
21.csv, R8
20.csv, R8
19.csv, R4
21.csv, R4
20.csv, R4
17.csv, R4
18.csv, R4
22.csv, R4
21.csv, R5
18.csv, R5
20.csv, R5
19.csv, R5
17.csv, R5
17.csv, R2
21.csv, R6
19.csv, R6
20.csv, R6
18.csv, R6
17.csv, R6
22.csv, R6
18.csv, R3
19.csv, R3
22.csv, R3
21.csv, R3
17.csv, R3
20.csv, R3
19.csv, R2
21.csv, R2
20.csv, R2
18.csv, R2
22.csv, R2
22.csv, R5
18.csv, R7
22.csv, R7
20.csv, R7
21.csv, R7
19.csv, R7
17.csv, R7
Ground Truth Region
Region 1
Region 2
Region 3
Region 4
Region 5
Region 6
Region 7
Region 8


sliced Assignments (Each row = stack, cols = rotated tissues in that stack)
Color = true region
18.csv, R1
22.csv, R1
17.csv, R3
20.csv, R3
21.csv, R3
19.csv, R3
20.csv, R6
18.csv, R6
22.csv, R2
20.csv, R2
18.csv, R2
22.csv, R5
19.csv, R8
22.csv, R8
19.csv, R2
19.csv, R6
18.csv, R8
17.csv, R8
19.csv, R5
21.csv, R7
21.csv, R5
17.csv, R2
22.csv, R4
21.csv, R4
17.csv, R5
18.csv, R7
22.csv, R7
19.csv, R1
18.csv, R5
18.csv, R3
19.csv, R7
20.csv, R7
17.csv, R7
20.csv, R5
17.csv, R4
18.csv, R4
21.csv, R8
20.csv, R8
21.csv, R2
21.csv, R6
17.csv, R6
22.csv, R6
17.csv, R1
20.csv, R1
21.csv, R1
22.csv, R3
19.csv, R4
20.csv, R4
Ground Truth Region
Region 1
Region 2
Region 3
Region 4
Region 5
Region 6
Region 7
Region 8


BOOSTED_SLICED Aligned: Each stack aligned to its first region (column 1)
17.csv, R2
20.csv, R1
22.csv, R1
18.csv, R1
21.csv, R1
17.csv, R1
22.csv, R3
19.csv, R3
18.csv, R3
21.csv, R3
20.csv, R3
17.csv, R3
18.csv, R7
19.csv, R7
21.csv, R7
22.csv, R7
20.csv, R7
17.csv, R7
20.csv, R2
21.csv, R2
19.csv, R2
22.csv, R2
18.csv, R2
22.csv, R5
22.csv, R8
21.csv, R8
19.csv, R8
20.csv, R8
18.csv, R8
17.csv, R8
20.csv, R4
21.csv, R4
18.csv, R4
19.csv, R4
17.csv, R4
22.csv, R4
20.csv, R5
19.csv, R5
18.csv, R5
21.csv, R5
17.csv, R5
19.csv, R1
17.csv, R6
18.csv, R6
20.csv, R6
19.csv, R6
21.csv, R6
22.csv, R6
Ground Truth Region
Region 1
Region 2
Region 3
Region 4
Region 5
Region 6
Region 7
Region 8


ot Assignments (Each row = stack, cols = rotated tissues in that stack)
Color = true region
17.csv, R4
18.csv, R4
20.csv, R4
20.csv, R5
17.csv, R7
22.csv, R3
18.csv, R8
19.csv, R2
19.csv, R8
22.csv, R8
19.csv, R6
17.csv, R8
17.csv, R5
18.csv, R7
20.csv, R7
19.csv, R7
22.csv, R7
19.csv, R1
22.csv, R6
21.csv, R6
17.csv, R6
20.csv, R3
17.csv, R3
20.csv, R8
21.csv, R4
22.csv, R4
19.csv, R5
21.csv, R7
21.csv, R5
19.csv, R4
21.csv, R2
17.csv, R2
22.csv, R5
18.csv, R5
18.csv, R3
19.csv, R3
20.csv, R2
22.csv, R2
18.csv, R2
20.csv, R6
18.csv, R6
21.csv, R8
20.csv, R1
17.csv, R1
21.csv, R1
22.csv, R1
18.csv, R1
21.csv, R3
Ground Truth Region
Region 1
Region 2
Region 3
Region 4
Region 5
Region 6
Region 7
Region 8


BOOSTED_OT Aligned: Each stack aligned to its first region (column 1)
20.csv, R4
21.csv, R4
18.csv, R4
19.csv, R4
17.csv, R4
22.csv, R4
19.csv, R7
21.csv, R7
22.csv, R7
20.csv, R7
17.csv, R7
18.csv, R7
17.csv, R5
19.csv, R5
20.csv, R5
18.csv, R5
21.csv, R5
17.csv, R2
21.csv, R2
22.csv, R2
18.csv, R2
20.csv, R2
19.csv, R2
22.csv, R5
17.csv, R6
18.csv, R6
20.csv, R6
19.csv, R6
21.csv, R6
22.csv, R6
20.csv, R8
21.csv, R8
19.csv, R8
22.csv, R8
18.csv, R8
17.csv, R8
22.csv, R1
20.csv, R1
18.csv, R1
21.csv, R1
17.csv, R1
19.csv, R1
20.csv, R3
21.csv, R3
18.csv, R3
19.csv, R3
22.csv, R3
17.csv, R3
Ground Truth Region
Region 1
Region 2
Region 3
Region 4
Region 5
Region 6
Region 7
Region 8


gromov Assignments (Each row = stack, cols = rotated tissues in that stack)
Color = true region
18.csv, R6
20.csv, R6
21.csv, R6
19.csv, R6
17.csv, R6
22.csv, R6
19.csv, R3
22.csv, R3
18.csv, R3
21.csv, R3
20.csv, R3
17.csv, R3
17.csv, R7
20.csv, R7
22.csv, R7
18.csv, R7
20.csv, R5
19.csv, R7
18.csv, R2
20.csv, R2
22.csv, R2
19.csv, R2
21.csv, R2
22.csv, R5
19.csv, R5
19.csv, R1
18.csv, R5
21.csv, R5
17.csv, R5
21.csv, R7
18.csv, R4
20.csv, R4
21.csv, R4
19.csv, R4
22.csv, R4
17.csv, R4
17.csv, R8
18.csv, R8
22.csv, R8
20.csv, R8
21.csv, R8
19.csv, R8
18.csv, R1
20.csv, R1
22.csv, R1
21.csv, R1
17.csv, R1
17.csv, R2
Ground Truth Region
Region 1
Region 2
Region 3
Region 4
Region 5
Region 6
Region 7
Region 8

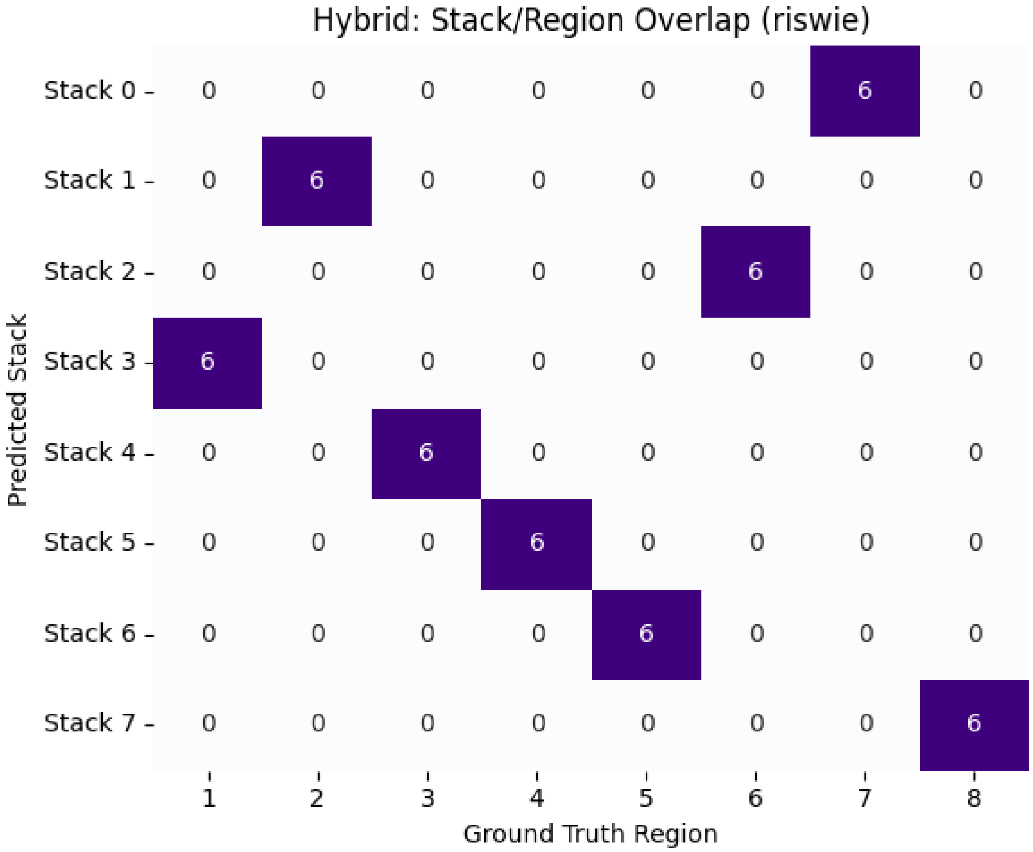


Figure 6: Hybrid Chosen Stack Assignment with RISWIE as the spatial distance and $\lambda = 0.5$

A.4.1 Hybrid Spatial–Marker Distance and Stack Assignment

To incorporate both spatial structure and marker expression in our region-level comparisons, and taking inspiration from Vayer et al. (2019), we define a hybrid distance matrix that interpolates between them.

For each pair of regions, we compute two quantities.

- A spatial distance using a selected geometric distance function (e.g., RISWIE, etc), applied to the cell coordinates within each region.
- A marker distance computed as the 2-Wasserstein distance between high-dimensional cell marker embeddings sampled from each region.

Let D_{ij}^{spatial} and D_{ij}^{marker} denote these pairwise dissimilarities, both scaled to [0, 1] via min-max normalization.

We then define

$$D_{ij}^{\text{hybrid}} = \lambda \cdot D_{ij}^{\text{spatial}} + (1 - \lambda) \cdot D_{ij}^{\text{marker}},$$

where $\lambda \in [0, 1]$ is tunable.

We then use this hybrid distance matrix to perform stack assignment as before. Interestingly, $\lambda = 0.5$ is able to recover perfect stack accuracy using RISWIE as the spatial distance, while $\lambda = 1.0$ and $\lambda = 0.0$ were unable to.

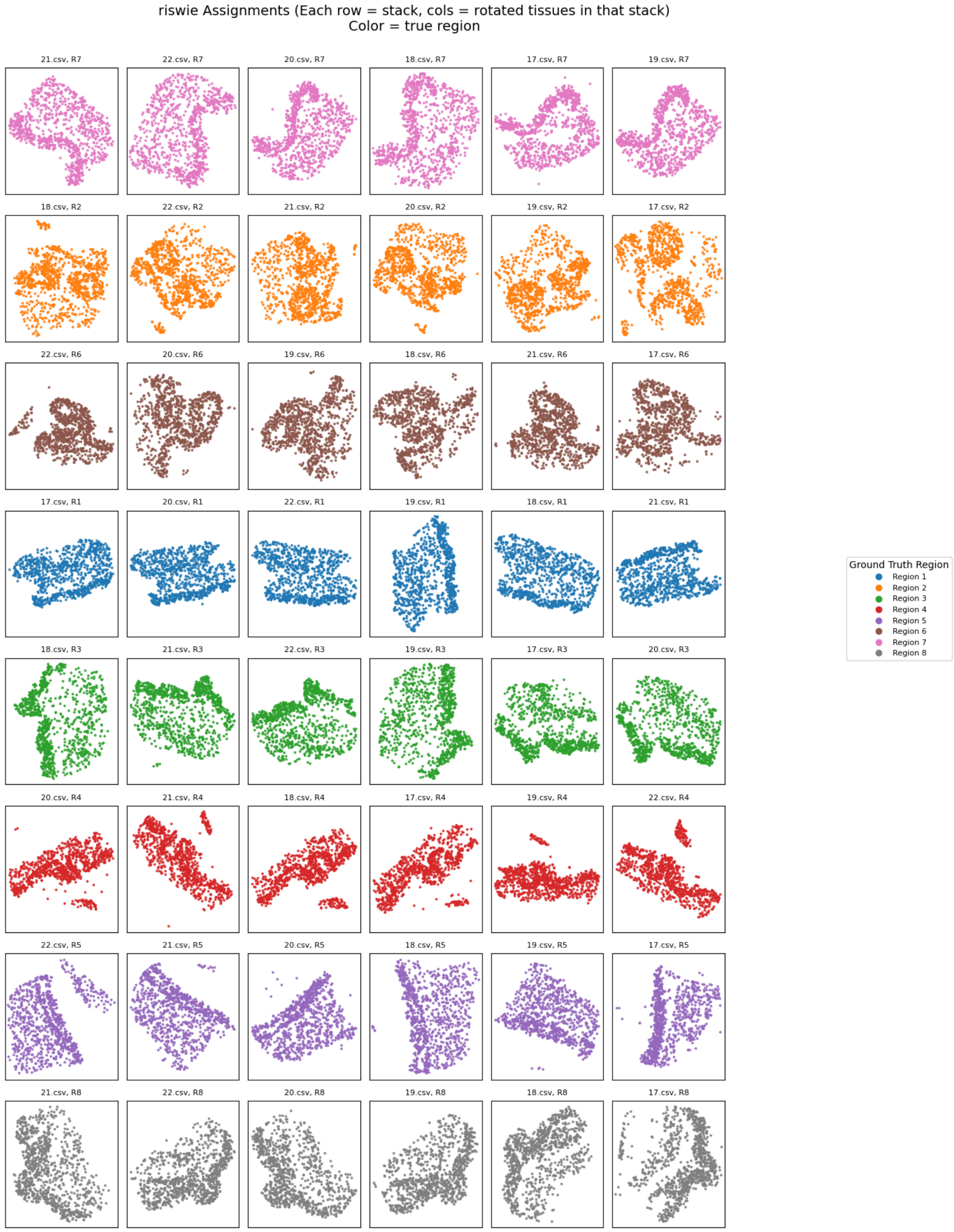


Figure 7: Unaligned Chosen Stack Assignment with RISWIE as the spatial distance and $\lambda = 0.5$

A.5 Ordering Agreement Between RISWIE and Gromov–Wasserstein

We also investigate how often the ordering induced by Gromov–Wasserstein aligns with that induced by RISWIE. Specifically, for the cell dataset, we compute the proportion of consistent orderings:

$$\frac{\sum \mathbb{I}\left[\operatorname{sign}(\mathrm{GW}(a,b)-\mathrm{GW}(c,d))=\operatorname{sign}(D(a,b)-D(c,d))\right]}{\sum 1}$$

where the sum ranges over all unique pairs of upper-triangular (off-diagonal) entries in the pairwise distance matrix.

Gromov–Wasserstein and RISWIE agreed on the ordering of 87.4% of all 635,628 region pair comparisons. The mean (median) absolute percentile difference between the two metrics was 0.091 (0.064).

When restricting to region pairs separated by at least one Gromov–Wasserstein standard deviation, the ordering agreement increased to 99.4% (302,853 out of 304,720 pairs).

Note that we approximate Gromov–Wasserstein using the solver provided in the POT library (Flamary et al., 2021; 2024). This does not guarantee exact agreement with the theoretical (NP-hard) Gromov–Wasserstein value.

A.6 Proofs

Proof of Theorem 1. RISWIE is defined on centered embeddings (the means are subtracted), so translation t has no effect on the pushforwards; we may assume $t = 0$ w.l.o.g.

PCA: Let $\Sigma_\mu = U\Lambda U^\top$ be the eigendecomposition of the covariance where $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_d)$ and the eigenvalues are ordered $\lambda_1 > \cdots > \lambda_r > 0 = \lambda_{r+1} = \cdots = \lambda_d$

Applying $T(x) = Rx + t$, the covariance of $T_\# \mu$ is

$$\Sigma_{T_\#\mu} = R\Sigma_\mu R^\top = (RU)\Lambda(RU)^\top$$

Seen on an individual eigenvector level,

$$\Sigma_\mu u = \lambda u \implies \Sigma_{T_\#\mu}(Ru) = R\Sigma_\mu R^\top(Ru) = R(\Sigma_\mu u) = \lambda(Ru),$$

Thus, the eigenvalues of $\Sigma_{T_\#\mu}$ are equal to those of Σ_μ and its eigenvectors are interpreted as orthogonally transformed versions of those of μ. For the eigenvectors corresponding to the non-zero eigenvalues, the transformation is unique up to sign. The two covariance matrices have the same distribution of eigenvalues (unique non-zero eigenvalues, some number of zero eigenvalues), so the only ambiguity in finding a non-zero eigenvalue eigenvector is the sign. For the zero-eigenvalue eigenvectors, which may have multiplicity, there is more to say.

For the zero-eigenvalue eigenspace, any orthonormal basis spans the kernel. Projections of μ onto any direction in this subspace yield Dirac masses at zero. Although there is some ambiguity in choosing them, we only use these eigenvectors to induce distributions on the real line, so the end effect is the same. Also, the sign ambiguity doesn't matter either (reflection of a Dirac mass at zero is still a Dirac mass at 0).

For the non-zero eigenvalue eigenvectors, the projection of rotated data onto rotated eigenvectors induces the same distribution. That is,

$$\text{for all } x \in \mathbb{R}^d: \quad \langle Rx,\, Ru\rangle = \langle x,\, u\rangle, \ \text{ so for any sample } \{x_i\},\ \{\langle Rx_i,\, Ru\rangle\}_i = \{\langle x_i,\, u\rangle\}_i$$

This assumes that we chose the optimal relative sign difference, because otherwise one of these multisets is reflected across 0. The element in the cost matrix for this pairing removes the ambiguity regarding the sign and recovers the correct relative sign between them. That is, for projections onto non-zero eigenvalue eigenvectors, we knew the induced distributions were unique up to sign, and s handles the relative difference in sign.

$$c(\pm u, \pm Ru) = \min_{s\in\{\pm 1\}} W_2^2\left(\{\langle x_i, u\rangle\}_{i=1}^n\,,\ \{s\langle Rx_i, Ru\rangle\}_{i=1}^n\right)$$

Notationally, what we are illustrating is that there is sign ambiguity in how each axis is obtained from PCA (up to sign), but regardless of that, the cost matrix entry will be the same.

W_2 is a metric, so W_2^2 is 0 if and only if the two multisets are equal. Thus, for one of these two terms in the minimization, W_2^2 will be 0. This is because Wasserstein is invariant under simultaneous reflection, so we only need to consider two cases instead of four.

As stated earlier, the zero eigenvalues all yield Dirac masses at 0, and the cost matrix entry between them will be 0.

Thus, if $\pi(i)$ is defined to pair axes with the same eigenvalue to axes of the same eigenvalue, each $c_{i,\pi(i)}$ will be 0. This is feasible because they have the same eigenvalue distribution. This can be done uniquely for the top r eigenvectors, and in any such way for the remaining indices $r+1, ...d$. The end result is that identical (up to sign) multisets are paired together, and scored as 0 cost, and any Diracs are paired together for 0 cost.

$$c_{i,\pi(i)} = \min_{s\in\{\pm 1\}} W_2^2\Big(\{\langle x_j, u_i\rangle\}_j,\ \{s\,\langle Rx_j, v_{\pi(i)}\rangle\}_j\Big) = 0,$$

Thus, $\mathrm{D}^2(\mu, T_\#\mu) = 0 \implies \mathrm{D}(\mu, T_\#\mu) = 0$ as

$$\mathrm{D}^2 \le \frac{1}{k}\sum_{j=1}^{k} c\big(u_j,\ v_{\pi(j)}\big) = 0$$

as we constructed one such signed permutation that is minimized over and RISWIE is non-negative.

Note that we can take only the top k eigenvectors (truncated SVD) and still obtain rigid-invariance by defining the same bijection π but truncating the two sets of eigenvectors, keeping only the top k by eigenvalue in each. This will also result in a RISWIE distance of 0.

We have directly shown the special case that when two distributions differ by a rigid transformation that their distance is 0. It is a simple generalization to show that arbitrary rigid transformations applied to one of two different distributions do not change the RISWIE distance.

That is, for two measures μ, ν (still making simple non-zero covariance eigenvalue assumptions), any for any rigid maps $T(x) = Rx,\ S(y) = Qy$,

$$D(\mu,\nu)\ =\ D(T_\#\mu,\nu)\ =\ D(\mu,S_\#\nu)\ =\ D(T_\#\mu,S_\#\nu)$$

This is because the RISWIE distance is just a function of the 1D marginals. The 1D marginals are actually the same up to sign for the same distribution before and after a rigid transformation. Thus, when we do axis-pairing, it doesn't matter whether a distribution was rigidly transformed or not. RISWIE will optimize over signs and remove that ambiguity.

Diffusion Maps: Define the kernel

$$K_{ij}\ =\ k\left(\frac{\|x_i - x_j\|^2}{\varepsilon}\right) \qquad \text{(e.g. } k(s) = e^{-s}\text{)}$$

Rigid transformations preserve pairwise distances

$$\|T(x_i) - T(x_j)\| = \|Rx_i + t - (Rx_j + t)\| = \|R(x_i - x_j)\| = \|x_i - x_j\|$$

Consequently, the construction of the kernel matrix itself is rigid-invariant. If we called the kernel matrix K' (build from $\{T(x_i)\}$), then $K' = K$.

As such, given that the entire diffusion procedure (writing the degree matrix E, Laplacian L, EVD, etc) is entirely derived from the kernel matrix, the embedded distributions should be exactly the same.

$$E' = \operatorname{diag}(K\mathbf{1}) = E, \qquad L'_{\text{rw}} = E^{-1}K = L_{\text{rw}}, \quad L'_{\text{sym}} = I - E^{-1/2}KE^{-1/2} = L_{\text{sym}}.$$

Let $L_{sym}\Phi = \Phi\Lambda$ be an be an eigendecomposition.

Point i is embedded with diffusion coordinates

$$\Psi_t(i) = \left(\lambda_1^t\,\phi_1(i), \ldots, \lambda_k^t\,\phi_k(i)\right)^\top$$

for some fixed time t.

Given that the construction of L_{sym} is rigid-invariant, the eigenvectors returned by an eigensolver for L_{sym} and L'_{sym} should be the same. Whether this is true in practice depends on the implementation of numerical eigensolvers. It would suffice to assume a simple spectrum, which would ensure that the eigenvectors are unique up to sign, but it is not necessary. As such, we only assume that the eigensolver used is deterministic.

Thus, following the same argument as for PCA, if the k 1D distributions are the same whether or not a rigid transformation is applied to the distribution, then the RISWIE distance between any two shapes does not depend on arbitrary rigid transformations applied to them. So $D(\mu, \nu) = D(T_\# \mu, S_\# \nu)$ where diffusion map embeddings in D are implicitly used as well.

□

Proof of Theorem 2. Let $\mathcal{E}$ be any deterministic k-dimensional embedding procedure. Then for any $X, Y, Z \in \mathcal{P}_2(\mathbb{R}^d)$, the RISWIE distance satisfies:

(i) Non-negativity: $D(X, Y) \geq 0$,

(ii) Symmetry: $D(X, Y) = D(Y, X)$,

(iii) Triangle inequality: $D(X, Z) \leq D(X, Y) + D(Y, Z)$,

The square root of the average of W_2^2 distances is non-negative and symmetric.

$$\text{Let } R_{XY} \;=\; argmin_{R \in \mathcal{O}_k^{\pm}} \frac{1}{k} \sum_{j=1}^{k} W_2^2\big(\alpha_j, \beta_{Rj}\big), \quad R_{YZ} \;=\; argmin_{R \in \mathcal{O}_k^{\pm}} \frac{1}{k} \sum_{j=1}^{k} W_2^2\big(\beta_j, \gamma_{Rj}\big).$$

Define the composite signed permutation $R_{XZ} \;=\; R_{YZ}\,R_{XY} \;\in\; \mathcal{O}_k^{\pm}$. For each j, let

$$u_j = W_2\big(\alpha_j, \beta_{R_{XY}j}\big), \quad v_j = W_2\big(\beta_{R_{XY}j}, \gamma_{R_{XZ}j}\big), \quad w_j = W_2\big(\alpha_j, \gamma_{R_{XZ}j}\big).$$

By the one-dimensional triangle inequality,

$$w_j \;=\; W_2\big(\alpha_j, \gamma_{R_{XZ}j}\big) \;\leq\; W_2\big(\alpha_j, \beta_{R_{XY}j}\big) \;+\; W_2\big(\beta_{R_{XY}j}, \gamma_{R_{XZ}j}\big) \;=\; u_j + v_j.$$

Hence componentwise $w \leq u + v$, so

$$\|w\|_2 \;\leq\; \|u + v\|_2 \;\leq\; \|u\|_2 + \|v\|_2,$$

and dividing by $\sqrt{k}$ gives

$$\sqrt{\frac{1}{k}\sum_{j=1}^{k} w_j^2} \;\leq\; \sqrt{\frac{1}{k}\sum_{j=1}^{k} u_j^2} \;+\; \sqrt{\frac{1}{k}\sum_{j=1}^{k} v_j^2}.$$

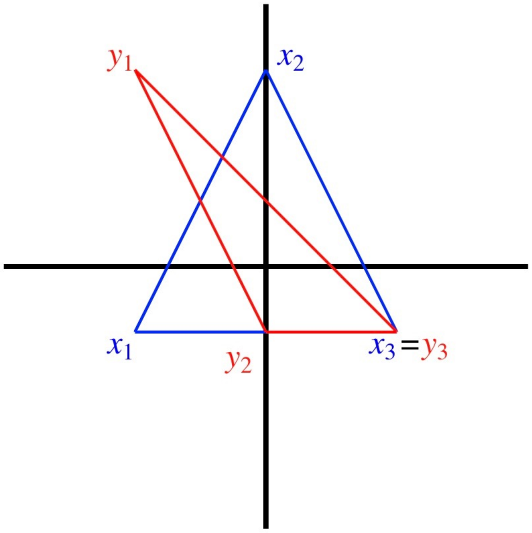


Figure 8: Using embeddings defined as the projection onto the standard basis vectors, these two point clouds of three points have RISWIE distance 0.

Since R_{XZ} is only a candidate for the minimization defining $D(X, Z)$,

$$\mathrm{D}(X,Z) \;=\; \min_{R\in\mathcal{O}_k^{\pm}} \sqrt{\frac{1}{k}\sum_{j=1}^{k} W_2^2(\alpha_j, \gamma_{Rj})} \;\le\; \sqrt{\frac{1}{k}\sum_{j=1}^{k} w_j^2} \;\le\; \mathrm{D}(X,Y) \;+\; \mathrm{D}(Y,Z).$$

□

Remark 1. While RISWIE is designed to be invariant to rigid transformations, a RISWIE distance of zero does not necessarily imply that two point clouds are related by a rigid transformation. Heuristically, this is essentially always the case with data-dependent embeddings, but it is a theoretical limitation. We show a counterexample to this property for RISWIE using a poor choice of embeddings (coordinate extraction, i.e., projecting onto e_1 and e_2). Thus, it remains true that an embedding must be appropriately and reasonably chosen to yield meaningful RISWIE distances.

Proof of Theorem 3. Without loss of generality, consider the centered versions $A \sim \mathcal{N}(0, \Sigma_A)$ and $B \sim \mathcal{N}(0, \Sigma_B)$, as RISWIE is translation-invariant.

Projecting $A \sim \mathcal{N}(0, \Sigma_A)$ onto its ith PCA axis u_i yields a one-dimensional Gaussian, since $u_i^\top x \sim \mathcal{N}(0, \lambda_i^A)$. Similarly, projecting $B \sim \mathcal{N}(0, \Sigma_B)$ onto its jth PCA axis v_j yields, with $v_j^\top y \sim \mathcal{N}(0, \lambda_j^B)$. Take

$$a_i := \sqrt{\lambda_i^A}$$

and $b_j := \sqrt{\lambda_j^B}$. It is known that the squared Wasserstein-2 distance between $\mathcal{N}(0, \lambda_i^A)$ and $\mathcal{N}(0, \lambda_j^B)$ is $(a_i - b_j)^2$.

Thus, the RISWIE cost for a permutation $\pi \in S_d$ is

$$C(\pi) := \frac{1}{d}\sum_{i=1}^{d}(a_i - b_{\pi(i)})^2.$$

We claim this is minimized when both vectors are sorted in increasing order (i.e., $\pi^* = \mathrm{id}$). Note that $a_1 \le \cdots \le a_d$ (the a_i are sorted).

Indeed, consider swapping two positions, say $i < j$, and compare the change in costs between the two permutations:

$$\begin{aligned}\Delta &:= \Big[(a_i - b_j)^2 + (a_j - b_i)^2\Big] - \Big[(a_i - b_i)^2 + (a_j - b_j)^2\Big] \\ &= \big[a_i^2 - 2a_ib_j + b_j^2 + a_j^2 - 2a_jb_i + b_i^2\big] - \big[a_i^2 - 2a_ib_i + b_i^2 + a_j^2 - 2a_jb_j + b_j^2\big] \\ &= \big[-2a_ib_j + b_j^2 - 2a_jb_i + b_i^2\big] - \big[-2a_ib_i + b_i^2 - 2a_jb_j + b_j^2\big] \\ &= -2a_ib_j + b_j^2 - 2a_jb_i + b_i^2 + 2a_ib_i - b_i^2 + 2a_jb_j - b_j^2 \\ &= 2a_i(b_i - b_j) + 2a_j(b_j - b_i) \\ &= 2(a_j - a_i)(b_j - b_i).\end{aligned}$$

If $b_j < b_i$ (an inversion relative to the $a-$order, then $b_j - b_i < 0$ and hence $\Delta \leq 0$. So swapping b_i, b_j for the increasing sorted order does not increase the cost, and strictly decreases it unless $a_i = a_j$.

Thus, given any permutation, it can be improved by swapping inverted adjacent pairs. The only time we can't improve a solution is there are no inversions, i.e. when

$$b_{\pi(1)} \leq b_{\pi(2)} \leq \cdots \leq b_{\pi(d)}$$

Since any permutation can be reduced to the identity via a sequence of such swaps, and each swap never increases the cost, the minimal cost is achieved by the identity permutation:

$$C(\mathrm{id}) = \frac{1}{d}\sum_{i=1}^{d}(a_i - b_i)^2.$$

Therefore,

$$\mathrm{D}_G^2(A, B) = \frac{1}{d}\|\mathbf{a} - \mathbf{b}\|_2^2,$$

as claimed. Here, we denote D_G to be the Gaussian closed form. □

Proof of Theorem 4. We use the bounds from (Salmona et al., 2022):

$$LGW_2^2(A, B) = 4(\mathrm{tr}(\Lambda_A) - \mathrm{tr}(\Lambda_B))^2 + 4(\|\Lambda_A\|_F - \|\Lambda_B\|_F)^2 + 4\|\Lambda_A - \Lambda_B\|_F^2,$$

$$GGW_2^2(A, B) = 4(\mathrm{tr}(\Lambda_A) - \mathrm{tr}(\Lambda_B))^2 + 8\|\Lambda_A - \Lambda_B\|_F^2 + 8(\|\Lambda_A\|_F^2 - \|\Lambda_A^{(n)}\|_F^2).$$

Here, LGW and GGW are lower and upper bounds for GW_2^2. The results from Salmona et al. (2022) are general and apply to Gaussian measures defined on Euclidean spaces of differing dimensions. For clarity and interpretability, however, we focus on the case where both distributions lie in the same ambient space. As such, we have already dropped an additional term from the original formulation, which accounted for the difference in Frobenius norm between the full covariance eigenvalue matrix and its truncation to the lower-dimensional space. This term vanishes in our setting since both distributions lie in the same ambient space, and no truncation is required.

Let $a_i = \sqrt{\lambda_i^A}$, $b_i = \sqrt{\lambda_i^B}$, and $\alpha = \min_i(a_i + b_i)$. Note that $(\lambda_i^A - \lambda_i^B)^2 = (a_i + b_i)^2(a_i - b_i)^2 \geq \alpha^2(a_i - b_i)^2$ for all i.

Therefore,

$$\|\Lambda_A - \Lambda_B\|_F^2 = \sum_{i=1}^{d}(\lambda_i^A - \lambda_i^B)^2 \geq \alpha^2\sum_{i=1}^{d}(a_i - b_i)^2 = d\alpha^2 D_G^2(A, B).$$

Since all other terms in LGW_2^2 are nonnegative,

$$LGW_2^2(A, B) \geq 4\|\Lambda_A - \Lambda_B\|_F^2 \geq 4d\alpha^2 D_G^2(A, B).$$

Similarly,

$$GGW_2^2(A, B) \geq 8\|\Lambda_A - \Lambda_B\|_F^2 \geq 8d\alpha^2 D_G^2(A, B).$$

Hence,

$$D_G^2(A,B) \leq \frac{GGW_2^2(A,B)}{8d\alpha^2}.$$

Additionally, Salmona et al. (2022) shows a bound on the difference between the upper and lower bounds:

$$GGW_2^2(A,B) - LGW_2^2(A,B) \leq 8\|\Sigma_A\|_F\|\Sigma_B\|_F \left(1 - \frac{1}{\sqrt{d}}\right).$$

Because $GW_2^2(A,B) \leq GGW_2^2(A,B)$, and $LGW_2^2(A,B) \leq GW_2^2(A,B)$, we may write

$$\begin{aligned} GGW_2^2(A,B) &= GW_2^2(A,B) + (GGW_2^2(A,B) - GW_2^2(A,B)) \\ &\leq GW_2^2(A,B) + (GGW_2^2(A,B) - LGW_2^2(A,B)). \end{aligned}$$

Plugging this into the previous bound,

$$\begin{aligned} \mathrm{D}_G^2(A,B) &\leq \frac{GW_2^2(A,B)}{8d\alpha^2} + \frac{GGW_2^2(A,B) - LGW_2^2(A,B)}{8d\alpha^2} \\ &\leq \frac{GW_2^2(A,B)}{8d\alpha^2} + \frac{\|\Sigma_A\|_F\|\Sigma_B\|_F}{d\alpha^2}\left(1 - \frac{1}{\sqrt{d}}\right). \end{aligned}$$

For the second bound, note that for all i,

$$(a_i - b_i)^2 = \left(\sqrt{\lambda_i^A} - \sqrt{\lambda_i^B}\right)^2 \leq \left|\lambda_i^A - \lambda_i^B\right|,$$

since by the factorization $a_i^2 - b_i^2 = (a_i - b_i)(a_i + b_i)$ and the triangle inequality,

$$|a_i - b_i| \leq |a_i + b_i| \implies (a_i - b_i)^2 \leq |a_i^2 - b_i^2| = |\lambda_i^A - \lambda_i^B|.$$

Thus,

$$\mathrm{D}_G^2(A,B) = \frac{1}{d}\sum_{i=1}^d (a_i - b_i)^2 \leq \frac{1}{d}\sum_{i=1}^d |\lambda_i^A - \lambda_i^B|.$$

By Cauchy–Schwarz,

$$\sum_{i=1}^d |\lambda_i^A - \lambda_i^B| \leq \sqrt{d}\left(\sum_{i=1}^d (\lambda_i^A - \lambda_i^B)^2\right)^{1/2} = \sqrt{d}\|\Lambda_A - \Lambda_B\|_F.$$

Thus,

$$\mathrm{D}_G^2(A,B) \leq \frac{1}{\sqrt{d}}\|\Lambda_A - \Lambda_B\|_F.$$

But $GW_2^2(A,B) \geq 4\|\Lambda_A - \Lambda_B\|_F^2 + 4(\mathrm{tr}(\Lambda_A) - \mathrm{tr}(\Lambda_B))^2 + 4(\|\Lambda_A\|_F - \|\Lambda_B\|_F)^2$, so

$$\|\Lambda_A - \Lambda_B\|_F^2 \leq \frac{1}{4}\left(GW_2^2(A,B) - 4(\mathrm{tr}(\Lambda_A) - \mathrm{tr}(\Lambda_B))^2 - 4(\|\Lambda_A\|_F - \|\Lambda_B\|_F)^2\right).$$

Therefore,

$$\|\Lambda_A - \Lambda_B\|_F \leq \frac{1}{2}\sqrt{GW_2^2(A,B) - 4(\mathrm{tr}(\Lambda_A) - \mathrm{tr}(\Lambda_B))^2 - 4(\|\Lambda_A\|_F - \|\Lambda_B\|_F)^2}.$$

Putting this together,

$$\mathrm{D}_G^2(A,B) \leq \frac{1}{2\sqrt{d}}\sqrt{GW_2^2(A,B) - 4\left(\mathrm{tr}(\Lambda_A) - \mathrm{tr}(\Lambda_B)\right)^2 - 4\left(\|\Lambda_A\|_F - \|\Lambda_B\|_F\right)^2}.$$

□

Corollary 5 (Identity of Indiscernibles for Gaussians). *Under the same setting as above, $D_G(A,B) = 0$ if and only if there exists an orthogonal matrix R and translation t such that B is the distribution of $RX + t$ for $X \sim A$.*

Proof. $D_G(A, B) = 0$ if and only if there exists $R \in \mathcal{O}_d^{\pm}$ such that

$$\sqrt{\lambda_j^A} = \sqrt{\lambda_{Rj}^B}, \quad \forall\, j = 1, \ldots, d,$$

or equivalently, $\lambda_j^A = \lambda_{Rj}^B$ for all j.

This means there exists a signed permutation R such that $\Lambda_A = R^\top \Lambda_B R$, i.e., the eigenvalues of Σ_A and Σ_B match up (possibly up to permutation and sign flip of axes). Without loss of generality, assuming A and B are centered Gaussians, it follows that their covariance matrices satisfy

$$\Sigma_B = R \Sigma_A R^\top.$$

Therefore, B is the law of RX for $X \sim A$, and more generally, the law of $TX + t$ for some orthogonal T and translation t.

Conversely, if B is the distribution of $TX + t$ for some orthogonal T and $t \in \mathbb{R}^d$, then A and B have matching covariance eigenvalues, so $D_G(A, B) = 0$.

□

Theorem 6 (Stability of RISWIE under Gaussian Covariance Perturbations). *If $\Sigma' = \Sigma_X + E$ with $E = E^\top$ and all eigenvalues of Σ_X, Σ' are $\geq \lambda_{\min} > 0$, then*

$$D_G(X, X') \leq \frac{\|E\|_2}{2\sqrt{\lambda_{\min}}}.$$

Proof. By Weyl's theorem for symmetric matrices (discussed by Shamrai (2025)), for each $i = 1, \ldots, d$,

$$|\lambda_i(\Sigma') - \lambda_i(\Sigma_X)| \leq \|\Sigma' - \Sigma_X\|_2 = \|E\|_2 \leq \eta,$$

where we set $\eta := \|E\|_2$.

Consider the function $f(x) = \sqrt{x}$ for $x \geq 0$. By the mean value theorem, for each i, there exists ξ_i between $\lambda_i(\Sigma_X)$ and $\lambda_i(\Sigma')$ such that

$$\left| \sqrt{\lambda_i(\Sigma')} - \sqrt{\lambda_i(\Sigma_X)} \right| = f'(\xi_i) \cdot |\lambda_i(\Sigma') - \lambda_i(\Sigma_X)|\,.$$

Since $f'(x) = \frac{1}{2\sqrt{x}}$ and all eigenvalues of Σ_X and Σ' are at least $\lambda_{\min}$, we have $\xi_i \geq \lambda_{\min}$, and $f'(\xi_i)$ is decreasing, so

$$f'(\xi_i) = \frac{1}{2\sqrt{\xi_i}} \leq \frac{1}{2\sqrt{\lambda_{\min}}}.$$

Therefore,

$$\left| \sqrt{\lambda_i(\Sigma')} - \sqrt{\lambda_i(\Sigma_X)} \right| \leq \frac{1}{2\sqrt{\lambda_{\min}}} \cdot \eta.$$

Let $\sigma_i := \sqrt{\lambda_i(\Sigma_X)}$, $\sigma_i' := \sqrt{\lambda_i(\Sigma')}$, and collect them as vectors $\sigma = (\sigma_1, \ldots, \sigma_d)$, $\sigma' = (\sigma_1', \ldots, \sigma_d')$.

Then,

$$\|\sigma' - \sigma\|_2 \leq \sqrt{\sum_{i=1}^{d} \left(\frac{\eta}{2\sqrt{\lambda_{\min}}} \right)^2} = \frac{\eta}{2\sqrt{\lambda_{\min}}} \sqrt{d},$$

so

$$D_G(X, X') \leq \frac{1}{\sqrt{d}} \cdot \frac{\eta}{2\sqrt{\lambda_{\min}}} \sqrt{d} = \frac{\eta}{2\sqrt{\lambda_{\min}}}.$$

More generally, if the lower bound for each eigenvalue is $\min(\lambda_i(\Sigma_X), \lambda_i(\Sigma'))$, then by the same reasoning,

$$D_G(X, X') \leq \frac{\eta}{2} \sqrt{\sum_{i=1}^{d} \frac{1}{\min(\lambda_i(\Sigma_X), \lambda_i(\Sigma'))}}.$$

□

Theorem 7 (Consistency of empirical RISWIE). *Let $\hat{\mu}_n, \hat{\nu}_n$ denote empirical measures of size n drawn i.i.d. from $\mu, \nu \in \mathcal{P}_2(\mathbb{R}^d)$, respectively. Then*

$$D(\hat{\mu}_n, \hat{\nu}_n) \xrightarrow{a.s.} D(\mu, \nu) \quad \textit{as } n \to \infty.$$

Proof. Fix $R \in \mathcal{O}_k^{\pm}$. Since the projections ϕ_j and ψ_j are measurable and bounded, the pushforward measures $(\phi_j)_{\#}\widehat{\mu}_n$ converge weakly almost surely to $(\phi_j)_{\#}\mu$ for each j, by the strong law of large numbers. Similarly, $(\psi_j)_{\#}\widehat{\nu}_n$ converge weakly almost surely to $(\psi_j)_{\#}\nu$.

In one dimension, the Wasserstein-2 distance W_2 is continuous with respect to weak convergence plus convergence of second moments. Since the measures are supported on a bounded interval and have finite second moments by construction, we conclude that

$$W_2\big((\phi_j)_{\#}\widehat{\mu}_n, (\psi_{Rj})_{\#}\widehat{\nu}_n\big) \xrightarrow{\text{a.s.}} W_2\big((\phi_j)_{\#}\mu, (\psi_{Rj})_{\#}\nu\big) \quad \text{as } n \to \infty.$$

Averaging over $j = 1, \ldots, k$ preserves almost sure convergence, and since the minimum of a finite collection of continuous functions is continuous, the minimum over $R \in \mathcal{O}_k^{\pm}$ also converges almost surely to its limit. Therefore,

$$D(\widehat{\mu}_n, \widehat{\nu}_n) \xrightarrow{\text{a.s.}} D(\mu, \nu) \quad \text{as } n \to \infty.$$

□

Remark 2 (Bias of the empirical RISWIE estimator). Let μ be Borel probability measure with finite second moments. Then, $D(\mu, \mu) = 0$, but

$$\mathbb{E}\big[D(\hat{\mu}_n, \hat{\mu}'_n)\big] > 0,$$

where $\hat{\mu}'_n$ is another independent sample of μ.

Proof. We have $D(\mu, \mu) = 0$, since projecting and optimally matching each direction trivially yields zero cost. However, the independent empirical marginals $\hat{\alpha}_j$ and $\hat{\alpha}'_j$ almost surely differ, and thus $W_2^2(\hat{\alpha}_j, \hat{\alpha}'_j) > 0$ almost surely for each j. Therefore, averaging and minimizing still yields strictly positive expectation:

$$\mathbb{E}\big[D(\hat{\mu}_n, \hat{\mu}'_n)\big] > 0.$$

□

A.7 LLM Usage Acknowledgement

We made regular use of a large language model to assist with coding tasks, including writing boilerplate functions, debugging implementation issues, and streamlining parts of the experimental pipeline. All generated code was reviewed, modified, and validated by the authors before inclusion in the project. LLMs were not used to develop theoretical results, to draft or edit the writing of the paper, or to identify or verify prior work. All proofs, algorithms, and textual content are entirely authored by the listed authors, and all references were manually checked against their original sources. The use of LLMs was confined to programming support, and the authors take full responsibility for the final implementation and results.