Robust Tangent Space Estimation via Laplacian Eigenvector Gradient Orthogonalization

Dhruv Kohli*
Department of Mathematics
UC San Diego
La Jolla, CA 92093
dhkohli@ucsd.edu

Sawyer J. Robertson*
Department of Mathematics
UC San Diego
La Jolla, CA 92093
s5robert@ucsd.edu

Gal Mishne
Halicioğlu Data Science Institute
UC San Diego
La Jolla, CA 92093
gmishne@ucsd.edu

Alexander Cloninger
Department of Mathematics and
Halicioğlu Data Science Institute
UC San Diego
La Jolla, CA 92093
acloninger@ucsd.edu

Abstract

Estimating the tangent spaces of a data manifold is a fundamental problem in data analysis. The standard approach, Local Principal Component Analysis (LPCA), struggles in high-noise settings due to a critical trade-off in choosing the neighborhood size. Selecting an optimal size requires prior knowledge of the geometric and noise characteristics of the data that are often unavailable. In this paper, we propose a spectral method, Laplacian Eigenvector Gradient Orthogonalization (LEGO), that utilizes the global structure of the data to guide local tangent space estimation. Instead of relying solely on local neighborhoods, LEGO estimates the tangent space at each data point by orthogonalizing the gradients of low-frequency eigenvectors of the graph Laplacian. We provide theoretical motivation for LEGO with a differential geometric analysis on a tubular neighborhood of a manifold. We show that gradients of low-frequency Laplacian eigenfunctions align closely with the tangent bundle, while an eigenfunction with high gradient in directions orthogonal to the manifold lie deeper in the spectrum. We demonstrate that LEGO yields tangent space estimates that are significantly more robust than those from LPCA, resulting in marked improvements in downstream tasks such as manifold learning, boundary detection, and local intrinsic dimension estimation.

1 Introduction

Tangent space estimation is a fundamental geometric task with broad applications across numerous domains, including manifold learning [1–5], data denoising [6], multi-manifold structure learning [7–10], local intrinsic dimension estimation [11], connection Laplacian approximation [12–14], and regression on manifolds [15], among others. The most commonly adopted approach for tangent space estimation is LPCA [1–3, 5, 6, 13, 15], which constructs the local covariance matrix using the k_{nn}-nearest neighbors of a data point and extracts the leading d eigenvectors, d being the local intrinsic dimension, as an estimate of the tangent basis at that point.

*equal contribution

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: New Perspectives in Advancing Graph Machine Learning.

The local formulation of LPCA makes it an efficient approach for tangent space estimation. However, as demonstrated in Figures 1a and 1b, it also leads to significant degradation of the quality of the estimates in the presence of noise. In fact, there is a well-known trade-off in the choice of neighborhood size for LPCA: small neighborhoods are prone to noise corruption, while larger ones introduce bias due to the underlying curvature and reach of the manifold [16–20]. One approach to improve robustness of LPCA involves selecting an adaptive neighborhood size [17] that balances these competing effects. However, implementation of such adaptive schemes is hindered by the fact that the geometric quantities—curvature, reach, and the noise level, are typically unknown.

In contrast, taking cues from the global structure of the data may offer an alternative route to robust tangent space estimation, avoiding the complexities of adapting neighborhood sizes while allowing them to remain small. This idea builds on [21] which shows that, for each point x on a d-dimensional Riemannian manifold [22], there exist d eigenfunctions of the manifold Laplacian which yield a bilipschitz local parameterization of a sufficiently small neighborhood of x into $\mathbb{R}^d$. This approach is practically realized in [4] where local parameterizations of the data manifold are constructed using global eigenvectors of the graph Laplacian. These parameterizations typically have low distortion, ensuring their Jacobians are full rank and span the d-dimensional tangent spaces. Together, these works motivate an approach to estimate tangent spaces using the gradients of Laplacian eigenvectors.

In the noisy setting, however, these graph Laplacian eigenvectors may still exhibit non-zero gradients orthogonal to the manifold, thereby picking up components in the noise directions [23–27]. Empirically, and as demonstrated in Figure 1c, it is often observed that the eigenvectors corresponding to small eigenvalues are robust to noise unlike the ones lying deeper into the spectrum [23–27]. Building upon this insight, we propose an algorithm to estimate tangent spaces using the gradients of *low-frequency* global eigenvectors of the graph Laplacian.

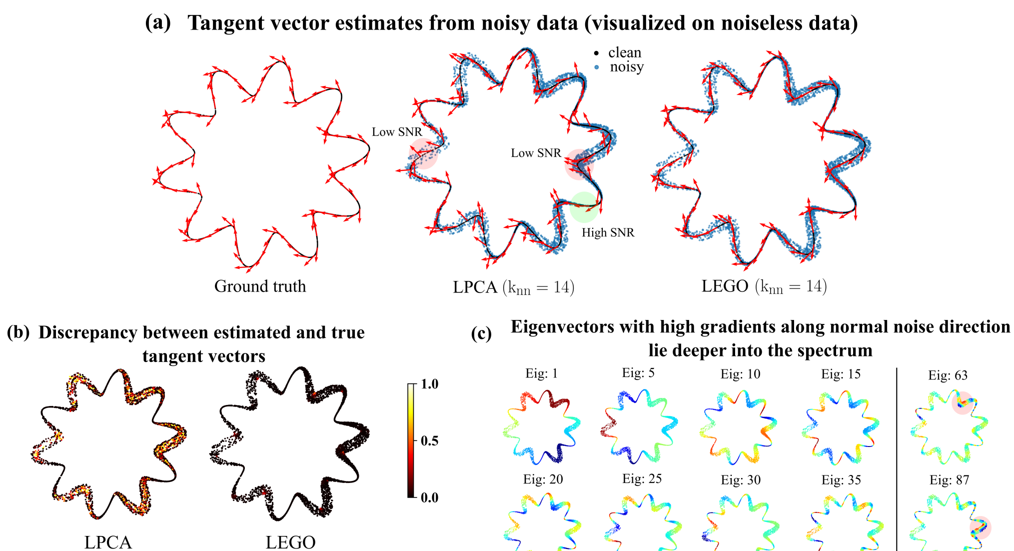

Figure 1: Tangent space estimation using LPCA and LEGO on a noisy point cloud generated by non-uniform sampling of a closed curve—wave on a circle—with heteroskedastic noise added in the normal direction. (a) Clean data points with ground truth tangent vectors, along with tangent vectors estimated from the noisy data using LPCA ($k_{nn} = 14$ and $d = 1$) and LEGO ($k_{nn} = 14$, $m_0 = 20$, $m = 100$ and $d = 1$). (b) Cosine dissimilarity between the true and the estimated tangent vectors. (c) Eigenvectors of the graph Laplacian constructed from noisy data [28], highlighting that those exhibiting high gradient in the noise direction lie deeper into the spectrum.

Our contributions are as follows. In Section 2, we present a spectral algorithm, LEGO (Laplacian Eigenvectors' Gradients Orthogonalization), for estimating tangent spaces at each data point by orthogonalizing the gradients of low-frequency *global* eigenvectors of the graph Laplacian derived from a noisy point cloud. A geometric justification of our method based on a tubular neighborhood noise model is provided in Section 3. In Section 4, we demonstrate that LEGO yields tangent space estimates that are significantly more robust to noise than those obtained via LPCA, resulting in significant improvements across multiple downstream tasks, including bottom-up manifold learning [1, 2, 29, 3, 5, 30], boundary detection [31–33], and local intrinsic dimension estimation [11].

2 Tangent space estimate via low-frequency Laplacian eigenvectors gradients

Here, we introduce our algorithm, LEGO, for estimating tangent spaces at noise-perturbed data points sampled from a tubular neighborhood of a smooth embedded submanifold. Our approach estimates orthonormal bases of the tangent spaces at the clean points by orthogonalizing the gradients of low-frequency global eigenvectors of the graph Laplacian constructed from the noisy data.

To begin with, let $Y = [Y_1, \ldots, Y_n] \in \mathbb{R}^{p\times n}$ be a point cloud sampled from a smooth compact d-dimensional submanifold B embedded in $\mathbb{R}^p$. Let $X = [X_1, \ldots, X_n] \in \mathbb{R}^{p\times n}$ be the noisy point cloud such that X_i is obtained by adding noise to Y_i in the directions orthogonal to the tangent space $T_{Y_i}B$. Let $\mathcal{N}_j = \{j_1, \ldots, j_{\mathrm{k_{nn}}}\}$ be a set containing the indices of the $\mathrm{k_{nn}}$-nearest neighbors of X_j obtained using the Euclidean metric in $\mathbb{R}^p$. Let $\mathcal{L} \in \mathcal{R}^{n\times n}$ be the graph Laplacian constructed from X using either the random walk [28, 34, 35], the self-tuned [36] or the doubly stochastic [37] kernel. These strategies ensure that, under appropriate scaling of the kernel bandwidth and sampling density, $\mathcal{L}$ converges with high probability to the manifold Laplacian $-\Delta_{\delta_p}$ on a tubular neighborhood of B. Recent results [36, 38] also establish the convergence of the spectrum of $\mathcal{L}$ to that of $-\Delta_{\delta_p}$.

Let $\phi_i \in \mathbb{R}^n$ be the i-th eigenvector of $\mathcal{L}$ corresponding to the i-th smallest eigenvalue, and $\nabla\phi_i \in \mathbb{R}^{p\times n}$ be a matrix whose j-th column, $\nabla\phi_i(X_j) \in \mathbb{R}^p$, represents the gradient of ϕ_i at X_j. The p components of the $\nabla\phi_i$ are modeled as a smooth function on X i.e. a vector in the span of the eigenvectors of $\mathcal{L}$. Since the eigenvectors associated with higher eigenvalues are more susceptible to noise (see, e.g., [39, 27] and the next section), we estimate $\nabla\phi_i$ using only the first $m_0 \ll n$ eigenvectors. To ensure local fidelity, we require $\nabla\phi_i$ to approximate ϕ_i, up to first order, on the neighborhood $\{X_{j_s} : j_s \in \mathcal{N}_j\}$ of each point X_j. Precisely, we define

$$\overline{X}_j = [X_{j_1} - X_j, \ldots, X_{j_{\mathrm{k_{nn}}}} - X_j]^T \text{ and } \overline{\phi}_i(X_j) = [\phi_i(X_{j_1}) - \phi_i(X_j), \ldots, \phi_i(X_{j_{\mathrm{k_{nn}}}}) - \phi_i(X_j)]^T$$

respectively. Then, the estimate $\widehat{\nabla}\phi_i \in \mathbb{R}^{p\times n}$ of the gradients $\nabla\phi_i$ is given by,

$$\widehat{\nabla}\phi_i = \widehat{C}_i {U_\Phi}^T \text{ where } U_\Phi \text{ is an orthonormal basis of the range of } \Phi = [\phi_1, \phi_2, \ldots, \phi_{m_0}] \text{ and} \quad (1)$$

$$\widehat{C}_i = \underset{C_i \in \mathbb{R}^{p\times m_0}}{\operatorname{argmin}} \frac{1}{n}\sum_{j=1}^{n} \left\| \overline{X}_j \nabla\phi_i(X_j) - \overline{\phi}_i(X_j) \right\|_2^2 \text{ such that } \nabla\phi_i = C_i {U_\Phi}^T. \quad (2)$$

Using the fact that ${U_\Phi}^T U_\Phi = I_{m_0}$, the least squares solution of the above optimization problem is,

$$\widehat{C}_i = \left[\overline{X}_1^\dagger \overline{\phi}_i(X_1) \quad \ldots \quad \overline{X}_n^\dagger \overline{\phi}_i(X_n)\right] U_\Phi. \quad (3)$$

Having obtained the gradient estimates of the eigenvectors $\{\phi_1, \ldots, \phi_m\}$ at X_j given by,

$$\widehat{\nabla}\phi(X_j) = \left[\widehat{\nabla}\phi_1(X_j) \quad \ldots \quad \widehat{\nabla}\phi_m(X_j)\right] \in \mathbb{R}^{p\times m}, \quad (4)$$

we obtain an estimate of the orthonormal basis $Q_j \in \mathbb{R}^{p\times d}$ of the d-dimensional tangent space at the jth point by orthogonalizing $\widehat{\nabla}\phi(X_j)$ i.e. using its top d left singular vectors. If the local intrinsic dimension d is not known then it can be estimated by selecting the smallest number of top eigenvalues of $\widehat{\nabla}\phi(X_j)$ whose normalized cumulative sum exceeds a user defined threshold. Finally, unlike LPCA which takes linear time in n, $\mathrm{k_{nn}}$, p and d, LEGO takes quadratic time in $\min\{\mathrm{k_{nn}}, p\}$ and linear in all other hyperparameters.

3 Laplacian eigenfunctions with high gradient along the cross sections of a tubular neighborhood lie deeper in the spectrum

Noisy data is often modeled as a sample drawn from a tubular neighborhood of an underlying smooth submanifold [40–42, 19, 20, 43]. The graph Laplacian constructed from such data [28, 36, 37] converges to the continuous Laplacian of the tubular neighborhood. This motivates the study of the Laplacian eigenfunctions on the tube to better understand the behavior of the graph Laplacian eigenvectors derived from noisy data. Here, we show that eigenfunctions exhibiting high gradients across the cross sections of the tube necessarily correspond to higher eigenvalues. Consequently, eigenfunctions associated with low eigenvalues exhibit minimal gradient in directions normal to the submanifold. The practical implication is that the gradients of low-frequency Laplacian eigenvectors have small components in the noise directions, making them suitable for tangent space estimation.

We start by describing the necessary constructs from [44] that are needed for our results. Let $B \subset \mathbb{R}^{d+k}$ be a smooth embedded compact d-dimensional submanifold with or without boundary, equipped with the metric $g_B = \delta_{d+k}|_B$ induced by the Euclidean metric δ_{d+k}. Let NB be the normal bundle of B equipped with the metric $g_B^{\perp} = \delta_{d+k}|_{NB}$. Assume that there exist a tubular neighborhood $\mathcal{T}^r$ of B such that $B \subset \mathcal{T}^r \subset \mathbb{R}^{d+k}$, where r is any finite number bounded by the global reach [40]. Define a map $\Psi : NB \to \mathbb{R}^{d+k}, (x,\nu) = x + \nu$ which, when restricted to $NB^r = \{(x,\nu) \in NB : \|\nu\|_{\delta_{d+k}} < r\} \subset NB$, is diffeomorphic to its image $\mathcal{T}^r$. Let $\pi : NB^r \to B$ be the canonical projection $\pi(x,\nu) = x$ onto B. By equipping NB^r with the pullback metric $g = \Psi^*\delta_{d+k}$, the tubular neighborhood $\mathcal{T}^r$ is isometric to NB^r. This also holds for ε-tubular neighborhood $\mathcal{T}^{\varepsilon r}$ of B and the normal bundle $NB^{\varepsilon r}$ for $\varepsilon \leq 1$. To keep the dependence on ε explicit, it is convenient to work with NB^r with the pullback metric $g^{\varepsilon} = \mathcal{D}_{\varepsilon}^* g$ where the map $\mathcal{D}_{\varepsilon} : NB^r \to NB^{\varepsilon r}$ is given by $\mathcal{D}_{\varepsilon}(x,\nu) = (x, \varepsilon\nu)$. In fact, NB^r equipped with g^{ε} is isometric to $\mathcal{T}^{\varepsilon r}$ equipped with Euclidean metric δ_{d+k}. Due to this construction, the Laplacian $-\Delta_{g^{\varepsilon}}$ on NB^r is unitarily equivalent to the Euclidean Laplacian $\Delta_{\delta_{d+k}}$ on $\mathcal{T}^{\varepsilon r}$. Moreover, the metric $g_s^{\varepsilon} = \pi^* g_B + \varepsilon^2 g_B^{\perp}$ turns π^* into a Riemannian submersion i.e. an isometry from $\pi^*(TB)$ to TB. Let $g_s = g_s^{\varepsilon=1}$ be the ε-independent unscaled version of the submersion metric.

Using g_s we define the horizontal energy $\mathcal{E}_B(\phi)$ that captures the net gradient of ϕ along the submanifold B, and vertical energy $\mathcal{E}_B^{\perp}(\phi)$ that measures the net gradient normal to B, across tube's cross sections. To this end, let $x^1, \ldots, x^d$ be the local coordinates on B and $\{e_\alpha\}_1^k$ be a locally orthonormal frame of NB^r with respect to $g_B^{\perp}$ such that every normal vector $\nu(x) \in N_xB$ can be written as $\nu(x) = n^{\alpha} e_{\alpha}(x)$. For $f \in C_0^{\infty}(NB^r)$, define its canonical gradients as $\nabla_x f = [\partial_{x^1} f, \ldots, \partial_{x^d} f]^T \in \mathbb{R}^d$ and $\nabla_n f = [\partial_{n^1} f, \ldots, \partial_{n^k} f]^T \in \mathbb{R}^k$. Then, for $\phi \in C_0^{\infty}(\mathcal{T}^{\varepsilon r})$, the horizontal energy $\mathcal{E}_B(\phi)$ and the vertical energy $\mathcal{E}_B^{\perp}(\phi)$ of ϕ are given by (here $\widehat{\phi} = \widehat{\mathcal{D}}_{\varepsilon}^{-1}\widehat{\Psi}\phi$ is the unitary lift of ϕ onto NB^r)

$$\mathcal{E}_B(\phi) = \frac{1}{\lambda_{B_2}} \frac{\int_{NB^r} \nabla_x \widehat{\phi}^T g_B^{-1} \nabla_x \widehat{\phi} dV_{g_s}}{\int_{NB^r} \widehat{\phi}^2 dV_{g_s}} = \frac{\int_{NB^r} \nabla_x \widehat{\phi}^T g_B^{-1} \nabla_x \widehat{\phi} \det(g_B) dx^1 \ldots dx^d dn^1 \ldots dn^k}{\int_{NB^r} \widehat{\phi}^2 \det(g_B) dx^1 \ldots dx^d dn^1 \ldots dn^k} \tag{5}$$

$$\mathcal{E}_B^{\perp}(\phi) = \frac{r^2}{C_k} \frac{\int_{NB^r} \nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi} dV_{g_s}}{\int_{NB^r} \widehat{\phi}^2 dV_{g_s}} = \frac{\int_{NB^r} \nabla_n \widehat{\phi}^T \delta_k^{-1} \nabla_n \widehat{\phi} \det(g_B) dx^1 \ldots dx^d dn^1 \ldots dn^k}{\int_{NB^r} \widehat{\phi}^2 \det(g_B) dx^1 \ldots dx^d dn^1 \ldots dn^k}. \tag{6}$$

Here, λ_{B_2} and C_k/r^2 are the first non-zero eigenvalues of $-\Delta_{g_B}$ on B and $-\Delta_{\delta_k}$ on a ball of radius r in $\mathbb{R}^k$, respectively. Here C_k is a constant that depends on the dimension k and the choice of the boundary conditions. The normalizations with these eigenvalues ensure that the horizontal and vertical energies are on similar scale with respect to r and ε.

Finally, we define a symmetric matrix $H_\alpha(x) \in \mathbb{R}^{d \times d}$ and a rectangular matrix $\Gamma_\beta(x) \in \mathbb{R}^{d \times k}$ that capture the coefficients of the scalar second fundamental form Π on B and the Christoffel symbols of the normal connection $\nabla^{\perp}$ with respect to $\{e_\alpha\}_1^k$, respectively. Specifically,

$$\text{For each } \alpha \in [1,k], (H_\alpha)_{i,j} = h_{\alpha i}^j = h_{\alpha j}^i = g_B^{\perp}(e_\alpha, \Pi(\partial_{x^i}, \partial_{x^j})), \text{ where } i,j \in [1,d], \text{ and} \tag{7}$$

$$\text{For each } \beta \in [1,k], (\Gamma_\beta)_{i,\alpha} = \gamma_{i\alpha}^{\beta} = g_B^{\perp}(\nabla_{\partial_{x^i}}^{\perp} e_\alpha, e_\beta), \text{ where } i \in [1,d], \alpha \in [1,k]. \tag{8}$$

Now we state our main result, the proof of which is provided in Appendix B.

Theorem 1. *If (λ, ϕ) is a Neumann or Dirichlet eigenpair of the Laplacian $\Delta_{\delta_{d+k}}$ on $\mathcal{T}^{\varepsilon r}$ then*

$$\lambda \geq \frac{C_k}{(\varepsilon r)^2} \frac{(1 - \varepsilon r \kappa^*)^d}{(1 + \varepsilon r \kappa^*)^d} \mathcal{E}_B^{\perp}(\phi), \tag{9}$$

$$\lambda \geq \frac{(1-\varepsilon r\kappa^*)^d}{(1+\varepsilon r\kappa^*)^d}\left(\frac{\lambda_{B_2}\mathcal{E}_B(\phi) - 2\kappa^{\perp *}\sqrt{\lambda_{B_2}C_k\mathcal{E}_B(\phi)\mathcal{E}_B^{\perp}(\phi)}}{(1+\varepsilon r\kappa^*)^2} + \frac{C_k\mathcal{E}_B^{\perp}(\phi)}{(\varepsilon r)^2}\right) \tag{10}$$

and

$$\lambda \leq \frac{(1+\varepsilon r\kappa^*)^d}{(1-\varepsilon r\kappa^*)^d}\left(\frac{\lambda_{B_2}\mathcal{E}_B(\phi)}{(1-\varepsilon r\kappa^*)^2} + \left(\left(\frac{\kappa^{\perp *}}{1-\varepsilon r\kappa^*}\right)^2 + \frac{1}{(\varepsilon r)^2}\right)C_k\mathcal{E}_B^{\perp}(\phi)\right), \tag{11}$$

where $\kappa^* = \max_{x\in B}\kappa(x) \in \mathbb{R}_{\geq 0}$ and $\kappa^{\perp *} = \max_{x\in B}\left\|\kappa^{\perp}(x)\right\|_2 \in \mathbb{R}_{\geq 0}$, and where $\kappa(x)$ is the absolute maximum principal curvature at $x \in B$ and $\kappa^{\perp}(x) \in \mathbb{R}^k_{\geq 0}$ quantifies the maximum rate of twisting of e_β, $\beta \in [1, k]$, in any direction in the tangent space T_xB (see Appendix B).

The above result shows that the eigenvalue λ scales as $\Omega\left(\frac{1}{(\varepsilon r)^2}\frac{(1-\varepsilon r\kappa^*)^d}{(1+\varepsilon r\kappa^*)^d}\right)$ per unit increase in $\mathcal{E}_B^{\perp}(\phi)$ versus $\mathcal{O}\left(\lambda_{B_2}\frac{(1+\varepsilon r\kappa^*)^d}{(1-\varepsilon r\kappa^*)^{d+2}}\right)$ per unit increase in $\mathcal{E}_B(\phi)$. Due to the fact $r\kappa^* < 1$, this simplifies to:

Corollary 1.1. *If $r \leq \sqrt{C_k/\lambda_{B_2}}$ and $(1-\varepsilon)^{d+1} \geq (1+\varepsilon)^d\varepsilon^{1-t}$ for some $t \in (0,1)$ then*

$$\lambda = \Omega\left(\varepsilon^{-2t}\mathcal{E}_B^{\perp}(\phi)\right) \text{ and } \lambda = \mathcal{O}\left(\mathcal{E}_B(\phi)\right). \tag{12}$$

Consequently, when ε is sufficiently small such that t is close to 1, any eigenvalue λ of $\Delta_{\delta_{d+k}}$ that is not too large—say, of order $\mathcal{O}(\varepsilon^{-2\eta})$ for some $\eta \in (0, 1/2)$—has a corresponding eigenfunction ϕ whose vertical energy $\mathcal{E}_B^{\perp}(\phi)$ is small, of order $\mathcal{O}(\varepsilon^{2t-2\eta})$. *The gradient of such an eigenfunction has small component in the normal directions to B, making it a suitable candidate for tangent space estimation.*

4 Experiments

Here, we estimate tangent spaces on noisy datasets using LPCA and LEGO, compare the estimated tangent spaces against the ground truth, and assess their utility in downstream tasks including manifold learning, boundary detection, and local intrinsic dimension estimation (see Appendix C).

First, we sample $n = 10700$ uniformly distributed points from a high-aspect ratio Swiss roll embedded in $\mathbb{R}^3$, yielding the clean dataset Y as shown in Figure 2a. Next, we perturb each point in the direction normal to the underlying tangent space. Specifically, the noisy data points are given by $X_j = Y_j + \eta_j\nu_j$, where ν_j is outward normal to the tangent space at Y_j and the coefficient η_j is uniformly distributed in $(-\varepsilon, \varepsilon)$ where $\varepsilon = 0.0175$. The resulting noisy dataset X is shown in Figure 2a.

We then estimate an orthonormal basis Q_j of the 2-dimensional tangent space at each point by applying LPCA and LEGO on the noisy data. The discrepancy $\mathcal{D}_j = \sum_{i=1}^d (1 - \cos(\theta_{j,i}))$, where $\{\theta_{j,i}\}_{i=1}^d$ are the principal angles between the estimate Q_j and the ground-truth Q_j^* obtained from clean data, is plotted in Figure 2b. These results show that LEGO produces significantly more accurate estimates across the noisy data while LPCA estimates are highly sensitive to noise. The same holds for noisy variants of a truncated curved torus data too. Ablation studies in Appendix D further reveal that LEGO estimates remain stable across different noise levels and hyperparameter values.

To assess how the tangent space estimates affect downstream tasks, we use them to compute a 2d embedding of the noisy data and to detect boundary points. As shown in Figures 2c and 2d, the LPCA-based embeddings and the detected boundary points are severely degraded by noise, while those based on LEGO closely match the results obtained from true tangent spaces. This is not surprising as the accuracy of the tangent space estimation is critical to the performance of several algorithms [1, 29, 3, 5, 32, 31] designed for these tasks. Finally, we evaluate the functional variance explained by each of the three principal directions (see Appendix C.3). As shown in Figure 2e, LEGO concentrates functional variance in the first two directions, aligning with the true intrinsic structure. In contrast, LPCA attributes significant variance to the third direction as well, indicating that the local intrinsic dimension estimates due to LPCA are sensitive to noise.

In the next experiment, we use an image dataset from [45], consisting of $n = 8100$ snapshots of a platform with two objects—Yoda and a bulldog—each rotating about its vertical axis at distinct

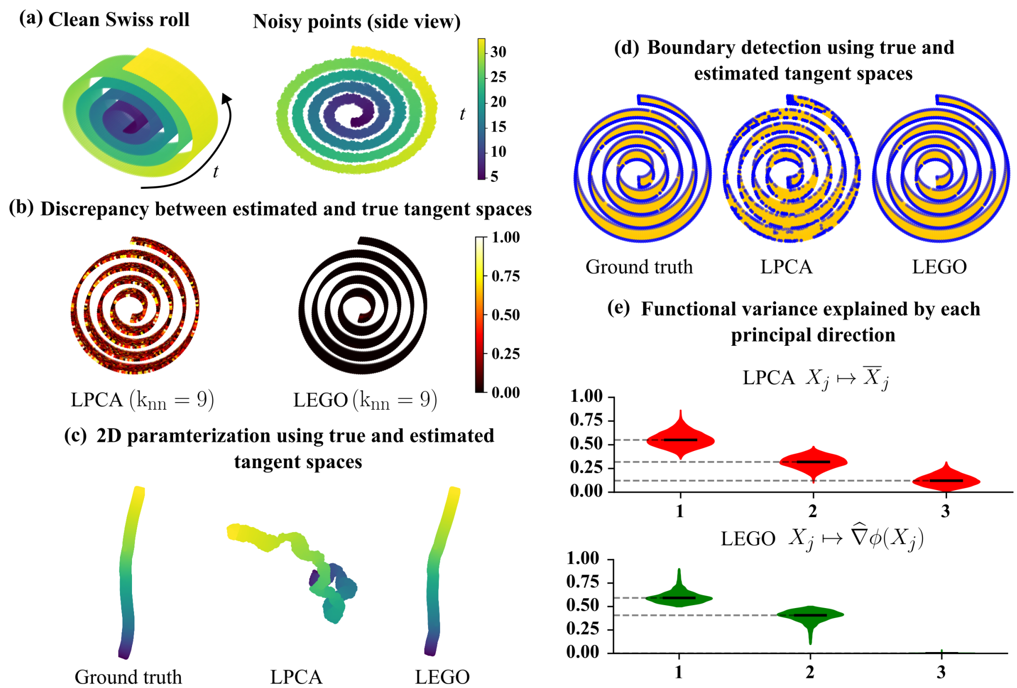

Figure 2: (a) Clean and noisy high-aspect ratio Swiss roll in $\mathbb{R}^3$ colored by the "roll" parameter. (b) Discrepancy between true and estimated tangent spaces due to LPCA ($k_{nn} = 9$) and LEGO ($k_{nn} = 9$, $m = 40$ and $m_0 = 100$). Note that $k_{nn} > 9$ creates erroneous "shortcuts" through the noisy data. (c, d) 2d parameterization and the boundary points detected from noisy data using the estimated and the true tangent spaces. (e) Functional variance explained across noisy points by each of the 3 principal directions obtained from LPCA and LEGO, reflecting local intrinsic dimension estimates.

frequencies. Therefore, the intrinsic geometry of the dataset is a 2-dimensional flat torus. The original images of size $320 \times 240 \times 3$ are first normalized to the range $[0, 1]$, followed by addition of uniformly distributed noise in $(-1, 1)$ to each pixel channel (Figure 3a). Due to the quadratic time complexity of LEGO with respect to the ambient dimension, we apply PCA to reduce the dimensionality of the noisy images to $p = 10$ dimensions where the explained variance ratio saturates (Figure 3b).

We then estimate the 2-dimensional tangent spaces using both LPCA and LEGO. These estimates are then used to compute a 2-dimensional "torn" embedding of noisy data following the bottom-up manifold learning framework in [4, 5]. As shown in Figure 3d, the LPCA-based embedding is non-interpretable, whereas LEGO produces a clear rectangular embedding. When visualized with "gluing instructions" [4, 5]—which identifies the same-colored points along the embedding tear—it becomes evident that opposite edges of the rectangle should be glued, revealing toroidal topology (Figure 3e). Moreover, examining the clean images at the points along the opposite edges reveals that only one of the two puppets undergoes rotation, further supporting the toroidal structure. Finally, Figure 3f shows that LEGO concentrates the variance explained in the first two directions, faithfully capturing the underlying 2d structure, while LPCA distributes the variance across multiple dimensions, highlighting its inability to accurately recover the intrinsic geometry in the noisy setting.

A Pseudocode and time complexity of LEGO

Assuming that the local intrinsic dimension d is known apriori, the cost of estimating tangent space at each point i.e., computing the top d principal directions from the local neighborhood using LPCA is $\mathcal{O}(k_{nn}pd)$. Therefore, the total time complexity of applying LPCA to all n points is $\mathcal{O}(nk_{nn}pd)$. In contrast, the time complexity of each stage of LEGO is as follows: (i) Computing eigenvectors $\phi_1, \ldots, \phi_{m_0}$ of the graph Laplacian $\mathcal{L}$ (Line 1 in Algorithm 1) using an iterative eigensolver requires $\mathcal{O}(nk_{nn}m_0T)$ time where T is the number of iterations required for convergence [46]. (ii) Estimating the gradients $\widehat{\nabla}\phi_i$ for all $i \in [1, m]$ (Line 2 in Algorithm 1) requires $\mathcal{O}(nk_{nn}p(\min\{k_{nn}, p\} + m)) +$

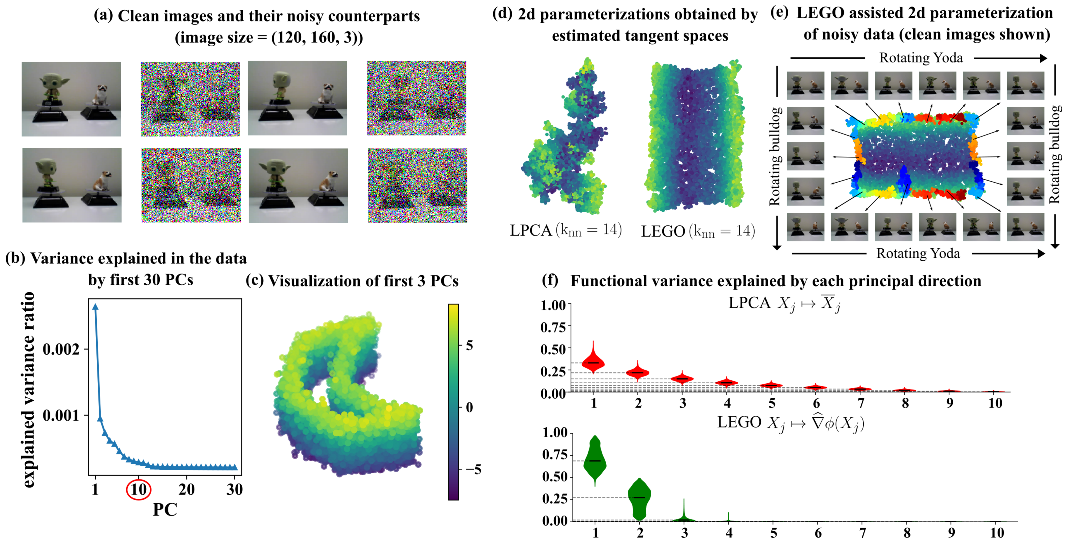

Figure 3: (a) Sample images [45] and their noise-perturbed versions (pixels clipped in $[0, 1]$). (b, c) The explained variance ratio saturating at the 10th dimension, and the visualization of the first three principal components of the noisy data, colored by the third PC. (d) Torn 2d embeddings [4, 5] of the noisy data using the estimated tangent spaces by LPCA ($\mathrm{k_{nn}} = 14$) and LEGO ($\mathrm{k_{nn}} = 14$, $m = 20$ and $m_0 = 100$). (e) The LEGO-based embedding equipped with the gluing instructions, reflecting toroidal topology. (f) Functional variance explained across noisy points by each of the 10 principal directions obtained from LPCA and LEGO, capturing the local intrinsic dimension estimates.

Algorithm 1 Tangent space estimation via LEGO.

Require: $X \in \mathbb{R}^{n \times p}$, $\mathcal{L}$, $\mathrm{k_{nn}}$, m and m_0 where $m \leq m_0$, either d or $\mathrm{f_{var}} \in (0, 1)$.
$\phi_1, \ldots, \phi_{m_0} \leftarrow$ eigenvectors of $\mathcal{L}$ corresponding to m_0 smallest eigenvalues.
Estimate $\widehat{\nabla}\phi_i$ using eqs. (1) and (3) for $i \in [1, m]$.
for $j \in [1, n]$ **do**
  Set $\widehat{\nabla}\phi(X_j)$ as in eq. (4).
  $U \in \mathbb{R}^{p \times p}, \sigma_1 \geq \ldots \geq \sigma_p \leftarrow$ Left singular vectors and values of $\widehat{\nabla}\phi(X_j)$.
  if d is provided **then**
    $Q_j \leftarrow$ first d columns of U.
  else
    $d_j \leftarrow \min\{s \in [1, p] : \sum_{i=1}^{s} \sigma_i^2 / \sum_{i=1}^{p} \sigma_i^2 \geq \mathrm{f_{var}}\}$.
    $Q_j \leftarrow$ first d_j columns of U.
  end if
end for
return $\{Q_j\}_1^n$.

$\mathcal{O}(nm_0mp)$ where the first term is exclusively corresponds to the estimation of $\widehat{C}_i$ in eq. (3) and the second term corresponds to multiplication by Φ on the right side in eq. (3) and by Φ^T in eq. (1). (iii) Computing the orthonormal basis Q_j for all $j \in [1, n]$ using the gradients of eigenvectors requires $\mathcal{O}(nmpd)$. Overall, the total time complexity of LEGO exceeds that of LPCA. In our experiments, the computation of the eigenvectors $\phi_1, \ldots, \phi_{m_0}$ seem to dominate the computational cost of LEGO.

B Proofs from Section 3

B.1 Preliminaries

As described in Section 3, $(x^1, \ldots, x^d, n^1, \ldots, n^k)$ form local coordinates of NB^r. The associated local coordinate vector fields are given by

$$\partial_i|_{(x,n)} = \partial_{x^i},\ \partial_{d+\alpha}|_{(x,n)} = \partial_{n^\alpha},\ \ i \in [1, d], \alpha \in [1, k]. \tag{13}$$

The span of $\{\partial_i|_{(x,n)}\}_1^d$ is not necessarily orthogonal to $\{\partial_{d+\alpha}|_{(x,n)}\}_1^k$. To aid the analysis, we obtain a new basis $\{\partial_i^H|_{(x,n)}\}_1^d$ by projecting each $\partial_i|_{(x,n)}$ orthogonal to the span of $\{\partial_{d+\alpha}|_{(x,n)}\}_1^k$ via Gram-Schmidt orthogonalization. Through basic manipulations we then obtain the following lemma.

Lemma 2. *The pullback metric $g^\varepsilon = \mathcal{D}_\varepsilon^* g = \mathcal{D}_\varepsilon^* \Psi^* \delta_{d+k}$ with respect to the coordinate vector fields $\{\partial_1^H|_{(x,n)}, \ldots, \partial_d^H|_{(x,n)}, \partial_{d+1}|_{(x,n)}, \ldots, \partial_{d+k}|_{(x,n)}\}$ on NB^r, is given by*

$$g^\varepsilon(x,n) = \begin{bmatrix} g_B^{1/2}(I_d - \varepsilon n^\alpha g_B^{-1/2} H_\alpha g_B^{-1/2})^2 g_B^{1/2} & \\ & \varepsilon^2 I_k \end{bmatrix} \tag{14}$$

Consequently, for $\widehat{\phi} \in C_0^\infty(NB^r)$ the Riemannian gradient of $\widehat{\phi}$ with respect to g^ε is given by,

$$\mathrm{grad}\widehat{\phi}(x,n) = \begin{bmatrix} g_B^{-1/2}(I_d - \varepsilon n^\alpha g_B^{-1/2} H_\alpha g_B^{-1/2})^{-2} g_B^{-1/2}\left(\nabla_x \widehat{\phi}(x,n) - n^\beta \Gamma_\beta \nabla_n \widehat{\phi}(x,n)\right) \\ \varepsilon^{-2}\nabla_n \widehat{\phi}(x,n) \end{bmatrix}. \tag{15}$$

Proof. The following form of the pullback metric $g = \Psi^*\delta_{d+k}$ with respect to the coordinate vector fields defined in eq. (13), and as derived in [44, Lemma 4.1], is given by

$$\begin{aligned} g_{i,j}(x,n) &= g_B(\partial_{x^i}, \partial_{x^j}) - 2\Pi_\nu(\partial_{x^i}, \partial_{x^j}) + g_B(W_\nu(\partial_{x^i}), W_\nu(\partial_{x^j})) + g_B^\perp(\nabla^\perp_{\partial_{x^i}}\nu, \nabla^\perp_{\partial_{x^j}}\nu) \\ g_{i,d+\alpha}(x,n) &= g_B^\perp(\nabla^\perp_{\partial_{x^i}}\nu, e_\alpha) \\ g_{d+\alpha,d+\beta}(x,n) &= g_B^\perp(e_\alpha, e_\beta) = \delta_{\alpha\beta}, \end{aligned}$$

for $i,j \in [1,d]$, $\alpha,\beta \in [1,k]$ and where (i) $W_{e_\alpha} : \mathfrak{X}(B) \to \mathfrak{X}(B)$ is the Weingarten map that captures the projection of $\overline{\nabla}_{\partial_{x^i}} e_\alpha(x)$ on TB i.e. $W_{e_\alpha}(\partial_{x^i}) = -(\nabla^{\mathbb{R}^{d+k}}_{\partial_{x^i}} e_\alpha(x))^\top$, and (ii) $\nabla^\perp : \mathfrak{X}(B) \times \mathfrak{X}(NB) \to \mathfrak{X}(NB)$ is the normal connection that captures the projection of $\nabla^{\mathbb{R}^{d+k}}_{\partial_{x^i}} e_\alpha(x)$ on NB, and is given by $\nabla^\perp_{\partial_{x^i}} e_\alpha(x) = (\nabla^{\mathbb{R}^{d+k}}_{\partial_{x^i}} e_\alpha(x))^\perp$. Since $\Pi_\nu(\partial_{x^i}, \partial_{x^j}) = g_B^\perp(\nu, \Pi(\partial_{x^i}, \partial_{x^j})) = g_B(\partial_{x^i}, W_\nu(\partial_{x^j})) = g_B(\partial_{x^j}, W_\nu(\partial_{x^i}))$, therefore, using the definitions of $h^j_{\alpha i}$ and $\gamma^\beta_{i\alpha}$ in Equations (7) and (8),

$$\Pi_\nu(\partial_{x^i}, \partial_{x^j}) = n^\alpha h^j_{\alpha i} = n^\alpha h^i_{\alpha j}. \tag{16}$$

$$W_\nu(\partial_{x^i}) = n^\alpha g_B^{kk'} h^i_{\alpha k'} \partial_{x^k} \tag{17}$$

$$\nabla^\perp_{\partial_{x^i}}\nu = n^\alpha \gamma^\beta_{i\alpha} e_\beta. \tag{18}$$

Therefore,

$$\begin{aligned} g_{i,j}(x,n) &= (g_B)_{ij} - 2n^\alpha h^j_{\alpha i} + n^\alpha n^\beta (g_B)_{kl} g_B^{kk'} h^i_{\alpha k'} g_B^{ll'} h^j_{\beta l'} + n^\alpha n^\beta \gamma^\omega_{i\alpha}\gamma^{\omega'}_{j\beta}\delta_{\omega\omega'} && (19) \\ &= (g_B)_{ij} - 2n^\alpha h^j_{\alpha i} + n^\alpha n^\beta g_B^{k'l'} h^i_{\alpha k'} h^j_{\beta l'} + n^\alpha n^\beta \gamma^\omega_{i\alpha}\gamma^{\omega'}_{j\beta}\delta_{\omega\omega'} && (20) \\ g_{i,d+\alpha}(x,n) &= n^\beta \gamma^\alpha_{i\beta} && (21) \\ g_{d+\alpha,d+\beta}(x,n) &= \delta_{\alpha\beta}. && (22) \end{aligned}$$

Consequently, the scaled pullback metric $g^\varepsilon = \mathcal{D}_\varepsilon^*\Psi^*\delta_{d+k}$ with respect to the coordinate vector fields in Equation (13) is given by,

$$\begin{aligned} {g^\varepsilon}_{i,j}(x,n) &= (g_B)_{ij} - 2\varepsilon n^\alpha h^j_{\alpha i} + \varepsilon^2 n^\alpha n^\beta g_B^{k'l'} h^i_{\alpha k'} h^j_{\beta l'} + \varepsilon^2 n^\alpha n^\beta \gamma^\omega_{i\alpha}\gamma^{\omega'}_{j\beta}\delta_{\omega\omega'} && (23) \\ {g^\varepsilon}_{i,d+\alpha}(x,n) &= \varepsilon^2 n^\beta \gamma^\alpha_{i\beta} && (24) \\ {g^\varepsilon}_{d+\alpha,d+\beta}(x,n) &= \varepsilon^2 \delta_{\alpha\beta} && (25) \end{aligned}$$

Now, the new basis vector $\partial_i^H|_{(x,n)}$ obtained by projecting $\partial_i|_{(x,n)}$ orthogonal to the span of $\{\partial_{d+\alpha}|_{(x,n)}\}_1^k$ is given by,

$$\partial_i^H|_{(x,n)} = \partial_i|_{(x,n)} - \sum_{\alpha=1}^k g_B^\perp(\nabla^\perp_{\partial_{x^i}}\nu, e_\alpha)\partial_{d+\alpha}|_{(x,n)} = \partial_i|_{(x,n)} - n^\beta\gamma^\alpha_{i\beta}\partial_{d+\alpha}|_{(x,n)}. \tag{26}$$

Consequently, the pullback metric g^ε in the new local coordinate fields $\{\partial_i^H|_{(x,n)}\}_1^d$ and $\{\partial_{d+\alpha}|_{(x,n)}\}_1^k$, is given by

$$
\begin{aligned}
g^\varepsilon{}_{i,j}(x,n) &= g_B(\partial_{x^i},\partial_{x^j}) - 2\varepsilon\Pi_\nu(\partial_{x^i},\partial_{x^j}) + \varepsilon^2 g_B(W_\nu(\partial_{x^i}), W_\nu(\partial_{x^j})) && (27)\\
&= (g_B)_{ij} - 2\varepsilon n^\alpha h^j_{\alpha i} + \varepsilon^2 n^\alpha n^\beta g_B^{k'l'} h^i_{\alpha k'} h^j_{\beta l'} && (28)\\
g^\varepsilon{}_{i,d+\alpha}(x,n) &= 0 && (29)\\
g^\varepsilon{}_{d+\alpha,d+\beta}(x,n) &= g_B^\perp(e_\alpha,e_\beta) = \varepsilon^2\delta_{\alpha\beta}. && (30)
\end{aligned}
$$

Using the definition of H_α in Equation (7),

$$
\begin{aligned}
g^\varepsilon(x,n) &= \begin{bmatrix} g_B - 2\varepsilon n^\alpha H_\alpha + \varepsilon^2 n^\alpha n^\beta H_\alpha g_B^{-1} H_\beta & \\ & \varepsilon^2 I_k \end{bmatrix} && (31)\\
&= \begin{bmatrix} (g_B^{1/2} - \varepsilon n^\alpha H_\alpha g_B^{-1/2})(g_B^{1/2} - \varepsilon g_B^{-1/2} n^\beta H_\beta) & \\ & \varepsilon^2 I_k \end{bmatrix} && (32)\\
&= \begin{bmatrix} g_B^{1/2}(I_d - \varepsilon n^\alpha g_B^{-1/2} H_\alpha g_B^{-1/2})^2 g_B^{1/2} & \\ & \varepsilon^2 I_k. \end{bmatrix} && (33)
\end{aligned}
$$

Finally, we decompose $\mathrm{grad}\widehat{\phi}$ into a component $(\mathrm{grad}\widehat{\phi})^H$ on $\pi^*(TB)$ and a component $(\mathrm{grad}\widehat{\phi})^V$ on $\ker(\pi_*)$. Specifically, $\mathrm{grad}\widehat{\phi} = (\mathrm{grad}\widehat{\phi})^H + (\mathrm{grad}\widehat{\phi})^V$ where

$$
(\mathrm{grad}\widehat{\phi})^H = g^{\varepsilon^{ij}} \partial_j^H \widehat{\phi} \partial_i^H = g^{\varepsilon^{ij}} \left(\frac{\partial \widehat{\phi}}{\partial x^j} - n^\beta \gamma^\alpha_{j\beta} \frac{\partial \widehat{\phi}}{\partial n^\alpha} \right) \partial_i^H \text{ and} \tag{34}
$$

$$
(\mathrm{grad}\widehat{\phi})^V = g^{\varepsilon^{d+\alpha,d+\beta}} \frac{\partial \widehat{\phi}}{\partial n^\beta} \partial_{d+\alpha} = \varepsilon^{-2} \frac{\partial \widehat{\phi}}{\partial n^\alpha} \partial_{d+\alpha}. \tag{35}
$$

Using the definition of Γ_β in Equation (8),

$$
\mathrm{grad}\widehat{\phi}|_{(x,n)} = \begin{bmatrix} g_B^{-1/2}(I_d - \varepsilon n^\alpha g_B^{-1/2} H_\alpha g_B^{-1/2})^{-2} g_B^{-1/2} \left(\nabla_x \widehat{\phi}(x,n) - n^\beta \Gamma_\beta \nabla_n \widehat{\phi}(x,n) \right) \\ \varepsilon^{-2} \nabla_n \widehat{\phi}(x,n) \end{bmatrix}. \tag{36}
$$

□

Note that g^ε is guaranteed to be positive semidefinite. However, for large ε it can become singular for certain values of $(n^1,\ldots,n^k)$. The following lemma provides a sufficient and necessary condition on ε that ensures the positivity of g^ε throughout NB^r.

Lemma 3. *Let $\kappa(x) \in \mathbb{R}_{\geq 0}$ and $\kappa^* \in \mathbb{R}_{\geq 0}$ be the absolute maximum principal curvature at $x \in B$ and across B, respectively, given by*

$$
\kappa^* = \max_{x\in B} \kappa(x) \ \ \textit{where} \ \ \kappa(x) = \max_{\|v\|_2=1} \left(\sum_{\alpha=1}^k \left(v^T g_B(x)^{-1/2} H_\alpha(x) g_B(x)^{-1/2} v \right)^2 \right)^{1/2}.
$$

Then, g^ε is positive definite on NB^r for all $\varepsilon < 1$. Moreover,

$$
\varepsilon^{-2k} \det(g_B) \left(1 - \varepsilon r \kappa^*\right)^{2d} \leq \det(g^\varepsilon) \leq \varepsilon^{-2k} \det(g_B) \left(1 + \varepsilon r \kappa^*\right)^{2d}. \tag{37}
$$

Proof. Using the expression of g^ε in Lemma 2, we obtain

$$
\det(g^\varepsilon) = \varepsilon^{2k} \det(g_B) \det\left(I_d - \varepsilon n^\alpha g_B^{-1/2} H_\alpha g_B^{-1/2} \right)^2. \tag{38}
$$

Using Cauchy-Schwarz inequality, we obtain

$$
\left\| n^\alpha g_B^{-1/2} H_\alpha g_B^{-1/2} \right\|_{\delta_k} = \sup_{\|v\|_2=1} v^T n^\alpha g_B^{-1/2} H_\alpha g_B^{-1/2} v \leq r\kappa(x) \leq r\kappa^*. \tag{39}
$$

Since, for each $x \in B$ the maximum value of $r\kappa(x)$ can be realized for some v dependent on x, therefore $\det(g^\varepsilon) > 0$ if and only if $\varepsilon r \kappa(x) < 1$. Since r is bounded by the global reach of B which in turn is bounded by $1/\kappa^*$ therefore $\det(g^\varepsilon) > 0$ for all $\varepsilon < 1$. Under this constraint, it follows that

$$
\varepsilon^{2k} \det(g_B) \left(1 - \varepsilon r \kappa^*\right)^{2d} \leq \det(g^\varepsilon) \leq \varepsilon^{2k} \det(g_B) \left(1 + \varepsilon r \kappa^*\right)^{2d}. \tag{40}
$$

□

Finally, as mentioned in the statement of Theorem 1, $\kappa^{\perp *}$ and $\kappa^{\perp}(x) \in \mathbb{R}^k_{\geq 0}$ are given by,

$$\kappa^{\perp *} = \max_{x \in B} \left\|\kappa^{\perp}(x)\right\|_2 \quad \text{where} \quad \kappa^{\perp}_{\beta}(x) = \left\|g_B(x)^{-1/2}\Gamma_\beta(x)\right\|_2, \ \beta \in [1, k]. \tag{41}$$

B.2 Proof of Theorem 1

Proof. First note that for $f \in C_0^\infty(\mathcal{T}^{\varepsilon r})$,

$$\int_{\mathcal{T}^{\varepsilon r}} f dV_{\delta_{d+k}} = \int_{NB^{\varepsilon r}} (\widehat{\Psi} f) dV_{\Psi^* \delta_{d+k}} = \int_{NB^r} (\widehat{\mathcal{D}}_\varepsilon^{-1}\widehat{\Psi} f) dV_{\mathcal{D}_\varepsilon^* \Psi^* \delta_{d+k}} = \int_{NB^r} (\widehat{\mathcal{D}}_\varepsilon^{-1}\widehat{\Psi} f) dV_{g^\varepsilon}. \tag{42}$$

Therefore, if (λ, ϕ) is an eigenpair of $\Delta_{\delta_{d+k}}$ on $\mathcal{T}^{\varepsilon r}$ with Neumann or Dirichlet boundary conditions then it follows that $\widehat{\phi} = \widehat{\mathcal{D}}_\varepsilon^{-1}\widehat{\Psi}\phi$ is an eigenfunction of Δ_{g^ε} with the same eigenvalue. Specifically,

$$\lambda = \frac{-\int_{\mathcal{T}^{\varepsilon r}} \phi \Delta_{\delta_{d+k}} \phi dV_{\delta_{d+k}}}{\int_{\mathcal{T}^{\varepsilon r}} \phi^2 dV_{\delta_{d+k}}} = \frac{-\int_{NB^r} \widehat{\phi}\Delta_{g^\varepsilon}\widehat{\phi} dV_{g^\varepsilon}}{\int_{NB^r} \widehat{\phi}^2 dV_{g^\varepsilon}} = \frac{\int_{NB^r} \langle \text{grad}\widehat{\phi}, \text{grad}\widehat{\phi}\rangle_{g^\varepsilon} dV_{g^\varepsilon}}{\int_{NB^r} \widehat{\phi}^2 dV_{g^\varepsilon}}. \tag{43}$$

Using Lemma 2, Equation (41) and Cauchy-Schwarz inequality, we obtain

$$\langle \text{grad}\widehat{\phi}, \text{grad}\widehat{\phi}\rangle_{g^\varepsilon} = \text{grad}\widehat{\phi}^T \ g^\varepsilon \ \text{grad}\widehat{\phi} \geq \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2}$$

and

$$\begin{aligned}
&\langle \text{grad}\widehat{\phi}, \text{grad}\widehat{\phi}\rangle_{g^\varepsilon} \\
&= \text{grad}\widehat{\phi}^T \ g^\varepsilon \ \text{grad}\widehat{\phi} \\
&= \left(\nabla_x \widehat{\phi} - n^\beta \Gamma_\beta \nabla_n \widehat{\phi}\right)^T g_B^{-1/2}(I_d - \varepsilon n^\alpha g_B^{-1/2} H_\alpha g_B^{-1/2})^{-2} g_B^{-1/2}\left(\nabla_x \widehat{\phi} - n^\beta \Gamma_\beta \nabla_n \widehat{\phi}\right) + \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2} \\
&\geq \frac{1}{(1+\varepsilon r \kappa^*)^2}\left(\nabla_x \widehat{\phi} - n^\beta \Gamma_\beta \nabla_n \widehat{\phi}\right)^T g_B^{-1}\left(\nabla_x \widehat{\phi} - n^\beta \Gamma_\beta \nabla_n \widehat{\phi}\right) + \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2} \\
&= \frac{1}{(1+\varepsilon r \kappa^*)^2}\left(\nabla_x \widehat{\phi}^T g_B^{-1}\nabla_x \widehat{\phi} + \left\|n^\beta g_B^{-1/2}\Gamma_\beta \nabla_n \widehat{\phi}\right\|_{\delta_k}^2 - 2n^\beta \nabla_x \widehat{\phi}^T g_B^{-1}\Gamma_\beta \nabla_n \widehat{\phi}\right) + \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2} \\
&\geq \frac{1}{(1+\varepsilon r \kappa^*)^2}\left(\nabla_x \widehat{\phi}^T g_B^{-1}\nabla_x \widehat{\phi} - 2n^\beta \nabla_x \widehat{\phi}^T g_B^{-1}\Gamma_\beta \nabla_n \widehat{\phi}\right) + \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2} \\
&\geq \frac{1}{(1+\varepsilon r \kappa^*)^2}\left(\nabla_x \widehat{\phi}^T g_B^{-1}\nabla_x \widehat{\phi} - 2\left(\nabla_x \widehat{\phi}^T g_B^{-1}\nabla_x \widehat{\phi}\right)^{1/2}\left\|n^\beta g_B^{-1/2}\Gamma_\beta \nabla_n \widehat{\phi}\right\|_{\delta_k}\right) + \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2} \\
&\geq \frac{1}{(1+\varepsilon r \kappa^*)^2}\left(\nabla_x \widehat{\phi}^T g_B^{-1}\nabla_x \widehat{\phi} - 2\left(\nabla_x \widehat{\phi}^T g_B^{-1}\nabla_x \widehat{\phi}\right)^{1/2}\left|n^\beta \kappa_\beta^\perp\right|\left\|\nabla_n \widehat{\phi}\right\|_{\delta_k}\right) + \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2} \\
&\geq \frac{1}{(1+\varepsilon r \kappa^*)^2}\left(\nabla_x \widehat{\phi}^T g_B^{-1}\nabla_x \widehat{\phi} - 2r\kappa^{\perp *}\left(\nabla_x \widehat{\phi}^T g_B^{-1}\nabla_x \widehat{\phi}\right)^{1/2}\left\|\nabla_n \widehat{\phi}\right\|_{\delta_k}\right) + \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2}.
\end{aligned}$$

In the last two equations we used $\|n\|_{\delta_k} \leq r$ and the definitions of $\kappa_\beta^\perp$ and $\kappa^{\perp *}$ provided in Equation (41). Combining the above with the bounds on $\det(g^\varepsilon)$ in Lemma 3, we obtain (the proof of Equation (9) follows similarly)

$$\begin{aligned}
\lambda &= \frac{\int_{NB^r} \langle \text{grad}\widehat{\phi}, \text{grad}\widehat{\phi}\rangle_{g^\varepsilon} dV_{g^\varepsilon}}{\int_{NB^r} \widehat{\phi}^2 dV_{g^\varepsilon}} \\
&= \frac{\int_{NB^r} \langle \text{grad}\widehat{\phi}, \text{grad}\widehat{\phi}\rangle_{g^\varepsilon} \sqrt{\det(g^\varepsilon)} dx^1 \ldots dx^d dn^1 \ldots dn^k}{\int_{NB^r} \widehat{\phi}^2 \sqrt{\det(g^\varepsilon)} dx^1 \ldots dx^d dn^1 \ldots dn^k} \\
&\geq \frac{(1-\varepsilon r \kappa^*)^d}{(1+\varepsilon r \kappa^*)^d} \frac{\int_{NB^r} \langle \text{grad}\widehat{\phi}, \text{grad}\widehat{\phi}\rangle_{g^\varepsilon} \sqrt{\det(g_B)} dx^1 \ldots dx^d dn^1 \ldots dn^k}{\int_{NB^r} \widehat{\phi}^2 \sqrt{\det(g_B)} dx^1 \ldots dx^d dn^1 \ldots dn^k}
\end{aligned}$$

$$
\begin{aligned}
&= \frac{(1-\varepsilon r\kappa^*)^d}{(1+\varepsilon r\kappa^*)^d} \frac{\int_{NB^r} \langle \mathrm{grad}\widehat{\phi}, \mathrm{grad}\widehat{\phi}\rangle_{g^\varepsilon} dV_{g_s}}{\int_{NB^r} \widehat{\phi}^2 dV_{g_s}} \\
&\geq \frac{(1-\varepsilon r\kappa^*)^d}{(1+\varepsilon r\kappa^*)^d} \left(\frac{\lambda_{B_2}\mathcal{E}_B(\phi) - 2r\kappa^{\perp *} \frac{\int_{NB^r} \left(\nabla_x \widehat{\phi}^T g_B^{-1} \nabla_x \widehat{\phi}\right)^{1/2} \left\|\nabla_n \widehat{\phi}\right\|_{\delta_k} dV_{g_s}}{\int_{NB^r} \widehat{\phi}^2 dV_{g_s}}}{(1+\varepsilon r\kappa^*)^2} + \frac{C_k \mathcal{E}_B^{\perp}(\phi)}{(\varepsilon r)^2} \right) \\
&\geq \frac{(1-\varepsilon r\kappa^*)^d}{(1+\varepsilon r\kappa^*)^d} \left(\frac{E_B(\phi) - 2r\kappa^{\perp *}\sqrt{\lambda_{B_2}\mathcal{E}_B(\phi) E_B^{\perp}(\phi)}}{(1+\varepsilon r\kappa^*)^2} + \frac{C_k \mathcal{E}_B^{\perp}(\phi)}{(\varepsilon r)^2} \right).
\end{aligned}
$$

The result follows from the definitions of horizontal and vertical energies in Equations (5) and (6). Similarly,

$$
\begin{aligned}
&\langle \mathrm{grad}\widehat{\phi}, \mathrm{grad}\widehat{\phi}\rangle_{g^\varepsilon} \\
&= \mathrm{grad}\widehat{\phi}^T \ g^\varepsilon \ \mathrm{grad}\widehat{\phi} \\
&= \left(\nabla_x \widehat{\phi} - n^\beta \Gamma_\beta \nabla_n \widehat{\phi}\right)^T g_B^{-1/2} (I_d - \varepsilon n^\alpha g_B^{-1/2} H_\alpha g_B^{-1/2})^{-2} g_B^{-1/2} \left(\nabla_x \widehat{\phi} - n^\beta \Gamma_\beta \nabla_n \widehat{\phi}\right) + \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2} \\
&\leq \frac{\left(\nabla_x \widehat{\phi} - n^\beta \Gamma_\beta \nabla_n \widehat{\phi}\right)^T g_B^{-1} \left(\nabla_x \widehat{\phi} - n^\beta \Gamma_\beta \nabla_n \widehat{\phi}\right)}{(1-\varepsilon r\kappa^*)^2} + \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2} \\
&= \frac{\left(g_B^{-1/2}\nabla_x \widehat{\phi} - n^\beta g_B^{-1/2} \Gamma_\beta \nabla_n \widehat{\phi}\right)^T \left(g_B^{-1/2}\nabla_x \widehat{\phi} - n^\beta g_B^{-1/2}\Gamma_\beta \nabla_n \widehat{\phi}\right)}{(1-\varepsilon r\kappa^*)^2} + \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2} \\
&\leq \frac{\left(\nabla_x \widehat{\phi}^T g_B^{-1} \nabla_x \widehat{\phi} + \left\|n^\beta g_B^{-1/2} \Gamma_\beta \nabla_n \widehat{\phi}\right\|_{\delta_k}^2\right)}{(1-\varepsilon r\kappa^*)^2} + \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2} \\
&\leq \frac{\left(\nabla_x \widehat{\phi}^T g_B^{-1} \nabla_x \widehat{\phi} + |n^\beta \kappa_\beta^{\perp}|^2 \nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}\right)}{(1-\varepsilon r\kappa^*)^2} + \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2} \\
&\leq \frac{\left(\nabla_x \widehat{\phi}^T g_B^{-1} \nabla_x \widehat{\phi} + (r\kappa^{\perp *})^2 \nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}\right)}{(1-\varepsilon r\kappa^*)^2} + \frac{\nabla_n \widehat{\phi}^T \nabla_n \widehat{\phi}}{\varepsilon^2}
\end{aligned}
$$

Combining the above with the bounds on $\det(g^\varepsilon)$ in Lemma 3, we obtain

$$
\begin{aligned}
\lambda &= \frac{\int_{NB^r} \langle \mathrm{grad}\widehat{\phi}, \mathrm{grad}\widehat{\phi}\rangle_{g^\varepsilon} dV_{g^\varepsilon}}{\int_{NB^r} \widehat{\phi}^2 dV_{g^\varepsilon}} = \frac{\int_{NB^r} \langle \mathrm{grad}\widehat{\phi}, \mathrm{grad}\widehat{\phi}\rangle_{g^\varepsilon} \sqrt{\det(g^\varepsilon)} dx^1 \dots dx^d dn^1 \dots dn^k}{\int_{NB^r} \widehat{\phi}^2 \sqrt{\det(g^\varepsilon)} dx^1 \dots dx^d dn^1 \dots dn^k} \\
&\leq \frac{(1+\varepsilon r\kappa^*)^d}{(1-\varepsilon r\kappa^*)^d} \frac{\int_{NB^r} \langle \mathrm{grad}\widehat{\phi}, \mathrm{grad}\widehat{\phi}\rangle_{g^\varepsilon} \sqrt{\det g_B} dx^1 \dots dx^d dn^1 \dots dn^k}{\int_{NB^r} \widehat{\phi}^2 \sqrt{\det g_B} dx^1 \dots dx^d dn^1 \dots dn^k} \\
&= \frac{(1+\varepsilon r\kappa^*)^d}{(1-\varepsilon r\kappa^*)^d} \frac{\int_{NB^r} \langle \mathrm{grad}\widehat{\phi}, \mathrm{grad}\widehat{\phi}\rangle_{g^\varepsilon} dV_{g_s}}{\int_{NB^r} \widehat{\phi}^2 dV_{g_s}} \\
&\leq \frac{(1+\varepsilon r\kappa^*)^d}{(1-\varepsilon r\kappa^*)^d} \left(\frac{\lambda_{B_2}\mathcal{E}_B(\phi)}{(1-\varepsilon r\kappa^*)^2} + \left(\left(\frac{\kappa^{\perp *}}{1-\varepsilon r\kappa^*}\right)^2 + (\varepsilon r)^{-2} \right) C_k \mathcal{E}_B^{\perp}(\phi) \right).
\end{aligned}
$$

The result follows from the definitions of normalized horizontal and vertical energies in Equations (5) and (6). □

C Overview of downstream tasks involving tangent space estimation

C.1 Bottom-up Manifold Learning

Given data points $X = [X_1, \dots, X_n] \in \mathbb{R}^{p \times n}$ sampled from a d-dimensional data manifold where $p \geq d$, bottom-up manifold learning algorithms [1–5] aim to recover a d-dimensional parameterization of the data by first constructing local embeddings and then aligning them on the overlaps to obtain a global embedding. Specifically, to construct local embeddings, an orthonormal basis $Q_j \in \mathbb{R}^{p\times d}$ is first estimated for the tangent space at each data point X_j. Using this basis, points in the neighborhood $\mathcal{N}_j$ of X_j are projected onto the estimated tangent space to obtain d-dimensional local coordinates. Precisely, the local coordinate of a neighbor X_{j_s}, where $j_s \in \mathcal{N}_j$, is given by:

$$\theta_{j_s,j} = Q_j^T(X_j - \mu_j) \text{ where } \mu_j = \frac{1}{\mathrm{k_{nn}}}\sum_{s=1}^{\mathrm{k_{nn}}} X_{j_s}. \tag{44}$$

In our experiments we estimate Q_j either using LPCA or our proposed method LEGO.

Once the local views (i.e., the low-dimensional embeddings for each neighborhood) are computed, they are aligned on the overlaps into a global embedding. This alignment can be performed using either rigid or affine transformations. While affine alignment is more flexible, it requires additional constraints—such as global uniformity [1, 29, 3]—to ensure the optimization problem is well-posed. However, such constraints often lead to severely distorted embeddings as shown in [5]. To preserve the intrinsic geometry of the manifold, we choose to align local views via rigid transformations [30, 47]. This leads to an optimization problem where, for each local view, we estimate an orthogonal matrix $S_j^* \in \mathbb{O}(d)$ and a translation vector $t_j^* \in \mathbb{R}^d$ that minimize the alignment error:

$$(S_j^*, t_j^*)_{j=1}^n = \mathrm{argmin}_{(S_j,t_j)_{j=1}^n} \sum_{k=1}^n \sum_{k\in\mathcal{N}_i\cap\mathcal{N}_j} \left\|(S_i^T\theta_{k,i} + t_i) - (S_j^T\theta_{k,j} + t_j)\right\|_2^2. \tag{45}$$

Several techniques have been proposed to solve this problem, including spectral and semidefinite relaxations [48, 49, 13, 50, 30, 51], generalized power method [52, 53], generalized Procrustes analysis [54, 55, 4, 56], and Riemannian gradient descent (RGD)[57, 47, 5]. In our experiments, we follow the procedure in[47, 5] where we first initialize the rigid transformations using Procrustes analysis and then refine them using RGD. Once the optimal rigid transformations are obtained, the global embedding is computed by averaging the transformed local coordinates:

$$\Theta_k = \frac{\sum_{k\in\mathcal{N}_j} S_j^{*T}\theta_{k,j} + t_j^*}{|\{j : k \in \mathcal{N}_j\}|}. \tag{46}$$

Note that when the data lies on a closed manifold—as in the case of the Yoda-Bulldog dataset—and the goal is to recover an embedding in the intrinsic dimension, the above alignment strategy leads to a collapsed non-injective embedding [5]. To address this, a tear-enabled rigid alignment framework was introduced in [5], which tears the data manifold to avoid collapse. We adopt this framework to compute a torn embedding in two dimensions of the Yoda-Bulldog dataset, along with gluing instructions at the tear (see Figure 3e). For further details regarding the tear-enabled alignment framework, we refer the reader to [5].

C.2 Boundary detection

Berry and Sauer in [32, 58] proposed a method to estimate the boundary points using a local statistic that approximates the normal direction near the boundary and yields a zero vector in the interior. A robust variant of this approach was recently developed in [31] which combines the doubly stochastic kernel [59, 60] along with tangent space estimates to detect the boundary points. The method starts by approximating the normal direction at X_j as,

$$\nu_j := \frac{1}{n-1}\sum_{i=1}^n W_{ij} Q_j^T(X_i - X_j). \tag{47}$$

Here W is the doubly stochastic kernel derived from the Gaussian kernel on the data via Sinkhorn iterations (see [60, 59] for details), and Q_j represents an estimate of the orthonormal basis of the tangent space at X_j. After computing ν_j for each $j \in [1, n]$, boundary points are identified by thresholding the norm $\|\nu_j\|_2$. In our experiments, we select a fixed percentile of these values as the threshold, labeling X_j as a boundary point if $\|\nu_j\|_2$ exceeds this threshold. The same percentile is used consistently across all methods for a given dataset.

C.3 Local intrinsic dimension

When the local intrinsic dimension d_j at the data point X_j is not known apriori, a typical procedure to estimate it—as used in LPCA—is to compute the eigenvalues $\lambda_1 \geq \ldots \geq \lambda_p$ of the local covariance matrix

$$C_j = \sum_{s=1}^{\mathrm{k_{nn}}} (X_{j_s} - \mu_j)(X_{j_s} - \mu_j)^T \text{ where } \mu_j = \frac{1}{\mathrm{k_{nn}}} \sum_{s=1}^{\mathrm{k_{nn}}} X_{j_s}, \tag{48}$$

and then assess the variance explained by the ith principal direction (the i-th eigenvector of C_j) at X_j via $\lambda_i / \sum_{k=1}^{p} \lambda_k$. The local intrinsic dimension at X_j is then selected to be the smallest index at which the cumulative variance explained exceeds a user defined threshold.

This strategy can be readily adapted to our method, with the key difference being that the eigenvalues $\lambda_1, \ldots, \lambda_p$ are now derived from the surrogate covariance matrix $\widehat{\nabla}\phi(X_j)\widehat{\nabla}\phi(X_j)^T$, constructed from the gradients of the low-frequency eigenvectors in LEGO. As demonstrated in our experiments, this provides a robust estimate of the local intrinsic dimension in the presence of noise and varying sampling density.

D Noise and hyperparmeter ablation

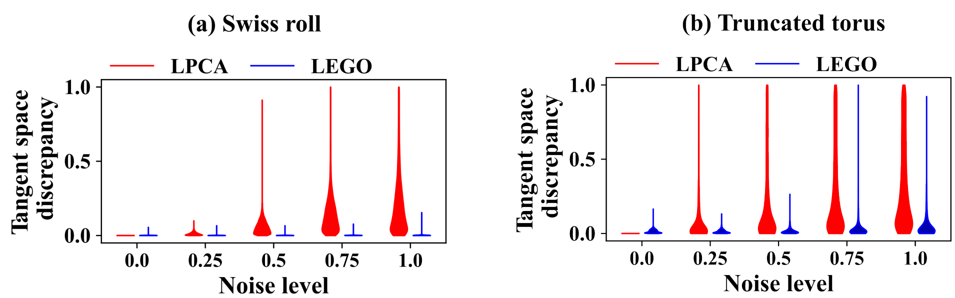

Figure 4: The discrepancy between true and the estimated tangent spaces from the noisy data $X_j = Y_j + \sigma \eta_j \nu_j$, $j \in [1, n]$, as the noise level σ varies between 0 and 1. In our experiments in Section 4, we used the maximum noise level i.e. $\sigma = 1$.

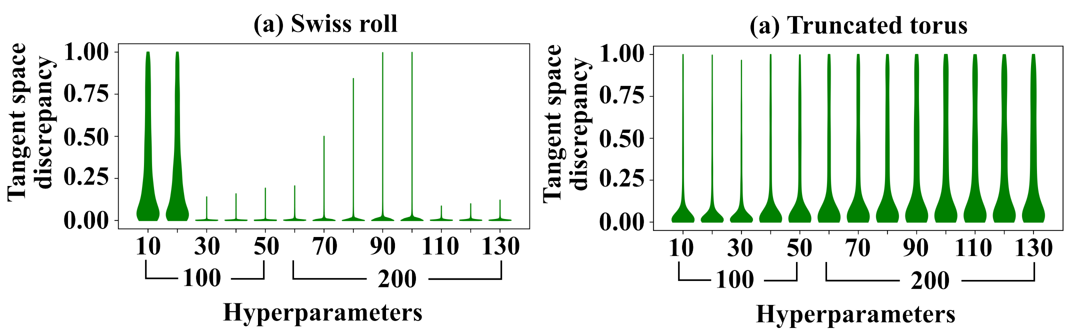

Figure 5: The discrepancy between the true and the estimated tangent spaces on the noisy datasets described in Section 4, against several different values of the hyperparameters m and m_0 in LEGO.

References

[1] Zhenyue Zhang and Hongyuan Zha. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM journal on scientific computing*, 26(1):313–338, 2004.

[2] Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326, 2000.

[3] David L Donoho and Carrie Grimes. Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Sciences*, 100(10):5591–5596, 2003.

[4] Dhruv Kohli, Alexander Cloninger, and Gal Mishne. LDLE: Low Distortion Local Eigenmaps. *Journal of Machine Learning Research*, 22(282):1–64, 2021.

[5] Dhruv Kohli, Johannes S Nieuwenhuis, Alexander Cloninger, Gal Mishne, and Devika Narain. RATS: Unsupervised manifold learning using low-distortion alignment of tangent spaces. *bioRxiv*, 2024.

[6] Dian Gong, Fei Sha, and Gérard Medioni. Locally linear denoising on image manifolds. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 265–272. JMLR Workshop and Conference Proceedings, 2010.

[7] Nicolas Garcia Trillos, Pengfei He, and Chenghui Li. Large sample spectral analysis of graph-based multi-manifold clustering. *Journal of Machine Learning Research*, 24(143):1–71, 2023.

[8] Yong Wang, Yuan Jiang, Yi Wu, and Zhi-Hua Zhou. Spectral clustering on multiple manifolds. *IEEE Transactions on Neural Networks*, 22(7):1149–1161, 2011.

[9] Dian Gong, Xuemei Zhao, and Gérard Medioni. Robust multiple manifolds structure learning. *arXiv preprint arXiv:1206.4624*, 2012.

[10] Ery Arias-Castro, Guangliang Chen, and Gilad Lerman. Spectral clustering based on local linear approximations. *Electronic Journal of Statistics*, 5(none):1537 – 1587, 2011.

[11] Elizaveta Levina and Peter Bickel. Maximum likelihood estimation of intrinsic dimension. *Advances in neural information processing systems*, 17, 2004.

[12] Amit Singer and Hau-Tieng Wu. Spectral convergence of the connection laplacian from random samples. *Information and Inference: A Journal of the IMA*, 6(1):58–123, 2017.

[13] Amit Singer and H-T Wu. Vector diffusion maps and the connection laplacian. *Communications on pure and applied mathematics*, 65(8):1067–1144, 2012.

[14] Sawyer Robertson, Dhruv Kohli, Gal Mishne, and Alexander Cloninger. On a generalization of Wasserstein distance and the Beckmann problem to connection graphs. *arXiv:2312.10295*, 2023.

[15] Ming-Yen Cheng and Hau-tieng Wu. Local linear regression on manifolds and its geometric interpretation. *Journal of the American Statistical Association*, 108(504):1421–1434, 2013.

[16] Eddie Aamari and Clément Levrard. Nonasymptotic rates for manifold, tangent space and curvature estimation. 2019.

[17] Daniel N Kaslovsky and François G Meyer. Non-asymptotic analysis of tangent space perturbation. *Information and Inference: a Journal of the IMA*, 3(2):134–187, 2014.

[18] Hemant Tyagi, Elıf Vural, and Pascal Frossard. Tangent space estimation for smooth embeddings of riemannian manifolds®. *Information and Inference: A Journal of the IMA*, 2(1):69–114, 2013.

[19] Anna V Little. *Estimating the intrinsic dimension of high-dimensional data sets: a multiscale, geometric approach.* PhD thesis, Duke University, 2011.

[20] Anna V Little, Mauro Maggioni, and Lorenzo Rosasco. Multiscale geometric methods for data sets i: Multiscale svd, noise and curvature. *Applied and Computational Harmonic Analysis*, 43(3):504–567, 2017.

[21] Peter W Jones, Mauro Maggioni, and Raanan Schul. Universal local parametrizations via heat kernels and eigenfunctions of the Laplacian. *arXiv preprint arXiv:0709.1975*, 2007.

[22] John M Lee. *Introduction to Riemannian manifolds*, volume 2. Springer, 2018.

[23] François G Meyer and Xilin Shen. Perturbation of the eigenvectors of the graph laplacian: Application to image denoising. *Applied and Computational Harmonic Analysis*, 36(2):326–334, 2014.

[24] Xiucai Ding and Hau-Tieng Wu. Impact of signal-to-noise ratio and bandwidth on graph laplacian spectrum from high-dimensional noisy point cloud. *IEEE Transactions on Information Theory*, 69(3):1899–1931, 2022.

[25] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE signal processing magazine*, 30(3):83–98, 2013.

[26] David K Hammond, Pierre Vandergheynst, and Rémi Gribonval. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 30(2):129–150, 2011.

[27] Xiuyuan Cheng and Gal Mishne. Spectral embedding norm: Looking deep into the spectrum of the graph laplacian. *SIAM journal on imaging sciences*, 13(2):1015–1048, 2020.

[28] Ronald R Coifman and Stéphane Lafon. Diffusion maps. *Applied and computational harmonic analysis*, 21(1):5–30, 2006.

[29] Zhenyue Zhang and Jing Wang. Mlle: Modified locally linear embedding using multiple weights. *Advances in neural information processing systems*, 19, 2006.

[30] Kunal N Chaudhury, Yuehaw Khoo, and Amit Singer. Global registration of multiple point clouds using semidefinite programming. *SIAM Journal on Optimization*, 25(1):468–501, 2015.

[31] Dhruv Kohli, Jesse He, Chester Holtz, Gal Mishne, and Alexander Cloninger. Robust estimation of boundary using doubly stochastic scaling of Gaussian kernel. *arXiv:2411.18942*, 2024.

[32] Tyrus Berry and Timothy Sauer. Density estimation on manifolds with boundary. *Computational Statistics & Data Analysis*, 107:1–17, 2017.

[33] Ryan Vaughn, Tyrus Berry, and Harbir Antil. Diffusion maps for embedded manifolds with boundary with applications to PDEs. *Applied and Computational Harmonic Analysis*, 68:101593, 2024.

[34] S.S. Lafon. Diffusion maps and geometric harmonics. *PhD Thesis*, page 45, 2004.

[35] Matthias Hein, Jean-Yves Audibert, and Ulrike von Luxburg. Graph laplacians and their convergence on random neighborhood graphs. *Journal of Machine Learning Research*, 8(6), 2007.

[36] Xiuyuan Cheng and Hau-Tieng Wu. Convergence of graph laplacian with knn self-tuned kernels. *Information and Inference: A Journal of the IMA*, 11(3):889–957, 2022.

[37] Xiuyuan Cheng and Boris Landa. Bi-stochastically normalized graph laplacian: convergence to manifold laplacian and robustness to outlier noise. *Information and Inference: A Journal of the IMA*, 13(4):iaae026, 2024.

[38] Nicolás García Trillos, Moritz Gerlach, Matthias Hein, and Dejan Slepčev. Error estimates for spectral convergence of the graph laplacian on random geometric graphs toward the laplace–beltrami operator. *Foundations of Computational Mathematics*, 20(4):827–887, 2020.

[39] Alex Kokot, Octavian-Vlad Murad, and Marina Meila. The noisy laplacian: a threshold phenomenon for non-linear dimension reduction. In *Forty-second International Conference on Machine Learning*.

[40] Partha Niyogi, Stephen Smale, and Shmuel Weinberger. Finding the homology of submanifolds with high confidence from random samples. *Discrete & Computational Geometry*, 39:419–441, 2008.

[41] Yariv Aizenbud and Barak Sober. Non-parametric estimation of manifolds from noisy data. *arXiv preprint arXiv:2105.04754*, 2021.

[42] Yariv Aizenbud and Barak Sober. Estimation of local geometric structure on manifolds from noisy data. *arXiv preprint arXiv:2503.07220*, 2025.

[43] Christopher R Genovese, Marco Perone-Pacifico, Isabella Verdinelli, and Larry Wasserman. Minimax manifold estimation. *The Journal of Machine Learning Research*, 13(1):1263–1291, 2012.

[44] Stefan Haag, Jonas Lampart, and Stefan Teufel. Generalised quantum waveguides. In *Annales Henri Poincaré*, volume 16, pages 2535–2568. Springer, 2015.

[45] Roy R Lederman and Ronen Talmon. Learning the geometry of common latent variables using alternating-diffusion. *Applied and Computational Harmonic Analysis*, 44(3):509–536, 2018.

[46] Richard B Lehoucq, Danny C Sorensen, and Chao Yang. *ARPACK users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods*. SIAM, 1998.

[47] Dhruv Kohli, Gal Mishne, and Alexander Cloninger. Non-degenerate rigid alignment in a patch framework. *arXiv:2303.11620*, 2023.

[48] Amit Singer. Angular synchronization by eigenvectors and semidefinite programming. *Applied and computational harmonic analysis*, 30(1):20–36, 2011.

[49] Mihai Cucuringu, Amit Singer, and David Cowburn. Eigenvector synchronization, graph rigidity and the molecule problem. *Information and Inference: A Journal of the IMA*, 1(1):21–67, 2012.

[50] Afonso S Bandeira, Moses Charikar, Amit Singer, and Andy Zhu. Multireference alignment using semidefinite programming. In *Proceedings of the 5th conference on Innovations in theoretical computer science*, pages 459–470, 2014.

[51] Afonso S Bandeira, Christopher Kennedy, and Amit Singer. Approximating the little Grothendieck problem over the orthogonal and unitary groups. *Mathematical programming*, 160(1):433–475, 2016.

[52] Shuyang Ling. Generalized power method for generalized orthogonal Procrustes problem: global convergence and optimization landscape analysis. *arXiv preprint arXiv:2106.15493*, 2021.

[53] Shuyang Ling. Near-optimal bounds for generalized orthogonal procrustes problem via generalized power method. *arXiv preprint arXiv:2112.13725*, 2021.

[54] John C Gower. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, 1975.

[55] Jos MF Ten Berge. Orthogonal Procrustes rotation for two or more matrices. *Psychometrika*, 42(2):267–276, 1977.

[56] Fabio Crosilla and Alberto Beinat. Use of generalised Procrustes analysis for the photogrammetric block adjustment by independent models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 56:195–209, 04 2002.

[57] Shankar Krishnan, Pei Yean Lee, John B Moore, Suresh Venkatasubramanian, et al. Global registration of multiple 3d point sets via optimization-on-a-manifold. In *Symposium on Geometry Processing*, pages 187–196, 2005.

[58] Ryan Vaughn, Tyrus Berry, and Harbir Antil. Diffusion maps for embedded manifolds with boundary with applications to pdes. *arXiv preprint arXiv:1912.01391v2*, 2019.

[59] Boris Landa, Ronald R Coifman, and Yuval Kluger. Doubly stochastic normalization of the gaussian kernel is robust to heteroskedastic noise. *SIAM journal on mathematics of data science*, 3(1):388–413, 2021.

[60] Boris Landa and Xiuyuan Cheng. Robust inference of manifold density and geometry by doubly stochastic scaling. *SIAM Journal on Mathematics of Data Science*, 5(3):589–614, 2023.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We have claimed that the proposed algorithm LEGO provide tangent space estimates robust to noise. The algorithm is described in Section 2 followed by theoretical justification in Section 3 and experimental analysis in Section 4.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Although LEGO results in significant improvement in tangent space estimates than the ones produced by LPCA, it has higher time complexity as mentioned at the end of Section 2.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: All the necessary constructs required for derivation of our results are provided in Section 3.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The data descriptions are provided in Section 4. Our algorithm LEGO is described in Section 2. A pseudocode is also provided in Appendix A. A brief explanation of the downstream tasks are provided in Appendix C.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: As noted in our previous response, we have included all necessary details in the paper to enable faithful reproduction of the experimental results.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The values of the hyperparameters are provided in the caption of the figures. Hyperparameter ablation analysis is provided in Appendix D.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [NA]

   Justification: The hyperparameter and noise ablation analysis are provided in Appendix D.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [NA]

Justification: All experiments are carried out on CPU workers, without relying on GPUs or cloud providers. The tangent space estimates in each experiment are obtained within just a few seconds.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The source of the puppets data in the experiments is cited.

    Guidelines:

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Description of tangent space estimation via LEGO is provided in Section 2 and a pseudocode is provided in Appendix A.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: LLM is only used for writing, editing, or formatting purposes.