Advances in Data Analysis and Classification
https://doi.org/10.1007/s11634-020-00426-3

REGULAR ARTICLE

A Riemannian geometric framework for manifold learning of non-Euclidean data

Cheongjae Jang[1] · Yung-Kyun Noh[2] · Frank Chongwoo Park[1]

Received: 5 May 2020 / Revised: 17 September 2020 / Accepted: 4 November 2020

Abstract

A growing number of problems in data analysis and classification involve data that are non-Euclidean. For such problems, a naive application of vector space analysis algorithms will produce results that depend on the choice of local coordinates used to parametrize the data. At the same time, many data analysis and classification problems eventually reduce to an optimization, in which the criteria being minimized can be interpreted as the distortion associated with a mapping between two curved spaces. Exploiting this distortion minimizing perspective, we first show that manifold learning problems involving non-Euclidean data can be naturally framed as seeking a mapping between two Riemannian manifolds that is closest to being an isometry. A family of coordinate-invariant first-order distortion measures is then proposed that measure the proximity of the mapping to an isometry, and applied to manifold learning for non-Euclidean data sets. Case studies ranging from synthetic data to human mass-shape data demonstrate the many performance advantages of our Riemannian distortion minimization framework.

Keywords Manifold learning · Non-Euclidean data · Riemannian geometry · Distortion · Harmonic map

Mathematics Subject Classification 53A35 · 53B21 · 58C35 · 58E20

Cheongjae Jang and Frank Chongwoo Park were supported in part by the NAVER LABS' AMBIDEX Project, MSIT-IITP (2019-0-01367, BabyMind), SNU-IAMD, SNU BK21+ Program in Mechanical Engineering, SNU Institute for Engineering Research, the National Research Foundation of Korea (NRF-2016R1A5A1938472), the Technology Innovation Program (ATC+, 20008547) funded by the Ministry of Trade, Industry, and Energy (MOTIE, Korea), and SNU BMRR Grant DAPAUD190018ID. Yung-Kyun Noh was supported by Samsung Research Funding & Incubation Center of Samsung Electronics under Project Number SRFC-IT1901-13 and by Hanyang University (HY-2019). (Corresponding author: Frank Chongwoo Park.).

Extended author information available on the last page of the article

Published online: 27 November 2020

1 Introduction

1.1 Motivation and contribution

A growing number of problems in data analysis and classification involve data that are non-Euclidean in nature. Examples include human shape and motion data (Vinué et al. 2016; Barahona et al. 2018), data collected from sensor networks, social networks in computational social sciences (Bronstein et al. 2017), and MRI imaging data (Fletcher and Joshi 2007). In all these cases it is a priori known that the collected data are drawn from a space that is not a vector space, but a curved space possessing additional geometric structure.

Given that most data analysis algorithms are formulated in a vector space setting, common practice when encountering non-Euclidean data is to ignore the fact that the data are drawn from a curved space. Instead, standard data analysis algorithms are applied to appropriate "vectorized" versions of the data in the hope that the algorithm will "learn" as needed any underlying geometric structure. More often than not, however, the way in which the data are "vectorized" and measured—more technically, the choice of local coordinates used to parametrize the data, and the choice of metric used to measure distances and angles between data points—has a profound effect on the results.

A more desirable solution is to formulate the analysis problem in a way that is independent of the choice of local coordinates, in the same way that, e.g., the choice of basis for a vector space should not affect the solution to a well-defined vector space analysis problem, or the notion of shortest path on a sphere should not depend on which coordinates are used to parametrize the sphere. Rather, the only choice to be made should be an intrinsic metric that reflects any physical or other intuitive attributes of the problem, similar to choosing a quadratic form that defines an inner product and norm for Euclidean space.

Many problems in data analysis and classification also reduce to an optimization problem. Although not recognized as such, the objective function being minimized is often a measure of distortion incurred when mapping one curved space into another. A representative example is manifold learning, which is a widely used technique for the more general problem of nonlinear dimensionality reduction. In classical manifold learning one is given a set of points in a higher-dimensional space, with the points assumed to lie on some lower-dimensional manifold embedded in this space; the objective then is to find a Euclidean parametrization for this manifold that best preserves distances and angles.[1] Existing manifold learning algorithms can in fact be distinguished by the choice of Euclidean parametrization and distortion measure, which are mostly ad hoc and in some cases not even coordinate-independent.

[1] A useful analogy is the problem of making two-dimensional Cartesian maps of the earth: given a set of data points sampled from the earth's surface, a two-dimensional surface—in this case a sphere—is first fitted to these points, and a Cartesian map of the sphere that best preserves distances and angles is then sought.

The main contribution of this paper is a Riemannian geometric framework for formulating and solving data analysis problems involving non-Euclidean data, all in a coordinate-invariant way. Our focus will be on manifold learning problems although many of the concepts and techniques are also applicable to other data analysis and classification problems. We first introduce a family of coordinate-invariant first-order measures that capture the distortion of a mapping between two Riemannian manifolds. A corresponding set of iterative gradient descent algorithms are also derived for this family of measures.

Using this geometric framework, we then formulate manifold learning as a Riemannian distortion minimization problem, while involving the case of non-Euclidean data. Several new criteria and algorithms are introduced that are not only valid for both non-Euclidean and Euclidean data, but even outperform existing manifold learning methods for Euclidean problems. We also show that many of the existing manifold learning methods can be reformulated in terms of our Riemannian distortion perspective.

The paper is organized as follows. Section 2 presents our Riemannian distortion framework. Section 3 frames manifold learning for non-Euclidean data as a distortion minimization problem, with case studies presented in Sect. 4.

1.2 Relation to existing works

Differential geometry is the natural language in which to study non-Euclidean data, and in Bronstein et al. (2017) a compelling case is made for the need to account for the geometry of non-Euclidean data, together with an introduction of the most relevant geometric concepts like Riemannian metrics, minimal geodesics, and the general importance of coordinate-invariance. One of the first works to frame a class of supervised learning problems as that of regression between Riemannian manifolds is Steinke et al. (2010). In their work a specific Hessian-based measure for capturing the distortion of a mapping between Riemannian manifolds is proposed, and applied to a point correspondence matching problem between two surfaces. Because the Hessian-based measure involves second derivatives of the mapping, however, second-order quantities must be estimated from the data, which is computationally involved and often highly sensitive to noise.

While there is a growing body of literature on how to generalize, e.g., density estimation (Pelletier 2005), principal component analysis (Fletcher and Joshi 2007), convolution (Bronstein et al. 2017), and other common data analysis and classification notions to non-Euclidean domains, almost all of the previous work on manifold learning and dimensionality reduction focus on problems in which the data are given in vector form with no a priori manifold structure. All of them also try to capture, at least implicitly, the general intuitive qualities of distortion: the locally linear embedding (Roweis and Saul 2000), Isomap (Tenenbaum et al. 2000), Laplacian eigenmap (Belkin and Niyogi 2003), diffusion map (Coifman and Lafon 2006), local tangent space alignment (Zhang and Zha 2004), local coordinates alignment (Zhang et al. 2008), and Hessian eigenmap (Donoho and Grimes 2003) propose different optimization criteria for constructing a lower-dimensional embedded manifold from a set of data points in a high-dimensional Euclidean space, and of obtaining a Euclidean parametrization of this manifold. Lin et al. (2015) further attempt to classify manifold learning algorithms into Laplacian-based, Hessian-based, and parallel field-based approaches.

By and large, classical manifold learning algorithms return embeddings that are far from ideal, with important information such as relative distances and directions among the data points often distorted (the situation is even worse for non-Euclidean data). Differential geometrically speaking, the obtained mappings deviate significantly from the ideal case of an **isometry**, i.e., a mapping between two spaces that preserves distances and angles everywhere.

In this paper, by quantifying and minimizing the deviation of a mapping from an isometry using our Riemannian distortion framework, we find optimal embeddings that better preserve the original input data structure. One of the most relevant works in this regard is the Riemannian relaxation method of McQueen et al. (2016); this work proposes a multidimensional integral criterion for the more restricted case of a mapping from a lower-dimensional input Riemannian manifold—a manifold structure is extracted with a Riemannian metric estimated from a set of high-dimensional data points using the projection method described in Perrault-Joncas and Meila (2013)—into a higher-dimensional Euclidean output space. Our formulation is in fact a more general one that not only subsumes McQueen et al. (2016) and other related manifold learning algorithms, but also leads to novel algorithms possessing several desirable properties. A second important feature of our approach is that it seamlessly extends our manifold learning methods to non-Euclidean data domains.

We conclude this section with some applications of Riemannian distortion that have been developed in other applications contexts. Park and Brockett (1994) have suggested a set of dexterity measures for robotic mechanisms based on Riemannian distortion and applied these to the optimal kinematic design problem. In computer graphics, different distortion minimization approaches have been proposed to find two-dimensional mesh parametrizations of discrete three-dimensional surfaces, which can be thought of as a low-dimensional manifold learning problem (Desbrun et al. 2002; Gu et al. 2004; Mullen et al. 2008). In Belkin et al. (2006) a Laplacian-based regularization criterion for semi-supervised learning has been proposed for mappings from a Euclidean-embedded manifold into Euclidean space, producing mappings in which neighboring points in the input space tend to be adjacent to each other in the output space.

2 Riemannian distortion

In this section, we show how to formulate global geometric distortion measures for a smooth mapping between two Riemannian manifolds. We first construct general coordinate-invariant first-order functionals for a mapping between two Riemannian manifolds, i.e., the functional is invariant with respect to the choice of local coordinates for the source and target Riemannian manifolds. We then examine those functionals that measure how close the mapping is to being an isometry. We conclude this section with a discussion of how manifold learning problems can be framed within this Rie-

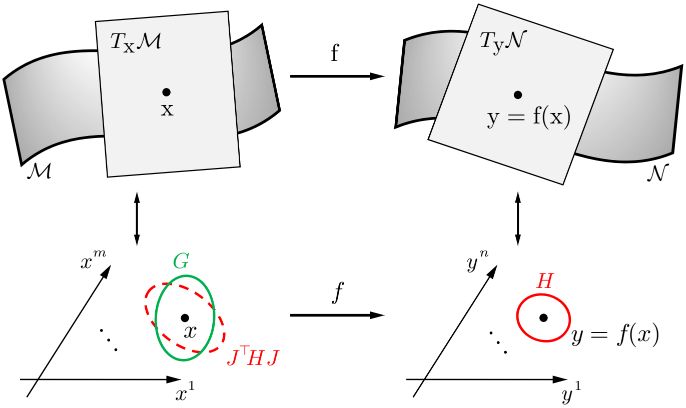

Fig. 1 Local coordinates, tangent spaces, and Riemannian metrics for the mapping $\mathrm{f} : \mathcal{M} \to \mathcal{N}$. Local coordinates are denoted in italics

mannian distortion framework, emphasizing the similarities and differences with the extant mathematical literature on minimum distortion mappings between Riemannian manifolds.

2.1 Constructing coordinate-invariant functionals on Riemannian manifolds

In this section we show how to construct coordinate-invariant functionals for a smooth mapping between two Riemannian manifolds. For further mathematical details on differentiable manifolds and Riemannian geometry, we refer the reader to Boothby (1986) and Dubrovin et al. (1992). Referring to Fig. 1, let $\mathcal{M}$ be an m-dimensional manifold with local coordinates $x = (x^1, \ldots, x^m)$ and Riemannian metric

$$ds^2 = \sum_{i=1}^{m}\sum_{j=1}^{m} g_{ij}(x)\, dx^i dx^j. \tag{1}$$

Let $\mathcal{N}$ be an n-dimensional manifold with local coordinates $y = (y^1, \ldots, y^n)$ and Riemannian metric

$$dr^2 = \sum_{\alpha=1}^{n}\sum_{\beta=1}^{n} h_{\alpha\beta}(y)\, dy^\alpha dy^\beta. \tag{2}$$

Throughout we use italics to represent local coordinates, e.g., a point $\mathrm{x} \in \mathcal{M}$ has local coordinates $x \in \mathbb{R}^m$, the mapping $\mathrm{f} : \mathcal{M} \to \mathcal{N}$ is represented in local coordinates as $y = f(x)$. The two metrics will also be denoted in matrix form as $G(x) = \big(g_{ij}(x)\big)$ and $H(y) = \big(h_{\alpha\beta}(y)\big)$, respectively.

Given a smooth mapping $\mathrm{f} : \mathcal{M} \to \mathcal{N}$, $\mathrm{x} \mapsto \mathrm{y} = \mathrm{f}(\mathrm{x})$, its differential $\mathrm{df}_{\mathrm{x}} : T_{\mathrm{x}}\mathcal{M} \to T_{\mathrm{y}}\mathcal{N}$ is denoted in local coordinates by the matrix

$$J(x) = \left(\frac{\partial f^i}{\partial x^j}(x)\right) \in \mathbb{R}^{n\times m}. \tag{3}$$

At a point x $\in \mathcal{M}$, find the characteristic values of the pullback metric $J(x)^\top H(f(x)) J(x)$ relative to the metric $G(x)$ of $\mathcal{M}$, i.e., the m real roots of the characteristic polynomial $p(\lambda) = \det(J^\top HJ - G\lambda) = 0$, which are the same as the eigenvalues of $J^\top HJG^{-1}$; apart from their order, these characteristic values are intrinsically associated with $J^\top HJ$ and G. (To see why, observe that under a pair of local coordinate transformations $x \mapsto x' = \phi(x)$ and $y \mapsto y' = \psi(y)$, G, H, and J transform according to the following rules: (i) $G \mapsto G' = \Phi^{-\top} G \Phi^{-1}$, where $\Phi = \frac{\partial \phi}{\partial x}$; (ii) $H \mapsto H' = \Psi^{-\top} H \Psi^{-1}$, where $\Psi = \frac{\partial \psi}{\partial y}$; (iii) $J \mapsto J' = \Psi J \Phi^{-1}$, from which it can be verified that the characteristic values of $J'^\top H' J'$ relative to G' remain the same.)

Let $\sigma(\lambda_1, \ldots, \lambda_m)$ be any symmetric function (i.e., a function whose value is invariant with respect to permutations of its arguments) of the m roots of the characteristic polynomial $p(\lambda)$. Then the integral

$$\int_{\mathcal{M}} \sigma(\lambda_1, \ldots, \lambda_m) \sqrt{\det G}\, dx^1 \cdots dx^m \tag{4}$$

is an intrinsic quantity, i.e., coordinate-invariant.

2.2 Riemannian distortion and isometry

Based on the above, we now examine choices for σ that capture the intrinsic distortion of the mapping f. The ideal case of no distortion is achieved when f is an **isometry**, i.e., angles and distances are preserved everywhere. Mathematically the equality $J^\top HJ = G$ must hold at every x $\in \mathcal{M}$. If $\dim(\mathcal{M}) \leq \dim(\mathcal{N})$, then for an isometry between $\mathcal{M}$ and f$(\mathcal{M}) \subseteq \mathcal{N}$ $J^\top HJG^{-1}$ is the identity, or equivalently, the eigenvalues $\lambda_1, \ldots, \lambda_m$ of $J^\top HJG^{-1}$ must all identically be one. The most straightforward choice for $\sigma(\lambda)$ is therefore

$$\sigma(\lambda) = \frac{1}{2} \sum_{i=1}^{m} (\lambda_i - 1)^2 = \frac{m}{2} + \sum_{i=1}^{m} \frac{\lambda_i^2}{2} - \lambda_i. \tag{5}$$

Ignoring the constant term and noting that the sum of a matrix's eigenvalues is its trace, while the sum of the squares of a matrix's eigenvalues is the trace of the squared matrix, the corresponding global distortion measure can be written

$$\min \int_{\mathcal{M}} \mathrm{Tr}\left((J^\top HJG^{-1})^2 - 2J^\top HJG^{-1} \right) \sqrt{\det G}\, dx^1 \cdots dx^m. \tag{6}$$

Closely related to the above is the alternative criterion

$$\min \int_{\mathcal{M}} \max_i \ (\lambda_i - 1)^2 \sqrt{\det G}\, dx^1 \cdots dx^m, \tag{7}$$

i.e., minimize the spectral norm of $J^{\top}HJG^{-1}-I$.[2] A third alternative is to minimize

$$\min \int_{\mathcal{M}} \sum_{i=1}^{m} (\log \lambda_i)^2 \sqrt{\det G}\, dx^1 \cdots dx^m, \tag{8}$$

which happens to correspond to the following widely used formula for measuring the distance between two symmetric positive-definite matrices A and B (Fletcher and Joshi 2007):

$$\operatorname{dist}(A, B)^2 = \sum_{i=1}^{m} (\log \lambda_i(A, B))^2\,, \tag{9}$$

where $\lambda_i(A, B)$ here denotes the roots of the polynomial $\det(B - A\lambda) = 0$.

There may also arise situations where the isometry requirement can be relaxed, e.g., preserving either shapes or volumes (but not both) is enough. If $\sigma(\lambda) = (\log(\lambda_1\lambda_2\cdots\lambda_m))^2 = (\log(\det(J^{\top}HJG^{-1})))^2$, then the corresponding global distortion functional measures the extent to which volumes are preserved (a volume-preserving map locally preserves volumes, but not distances and angles). The choice $\sigma(\lambda) = \lambda_{\max}/\lambda_{\min}$ corresponds to the condition number of $J^{\top}HJG^{-1}$; when the condition number attains the minimum possible value of 1, shapes (but not necessarily their volumes) are preserved.

Since the integrands in all of the above functionals locally measure the deviation of the mapping f from an isometry, there is no question that their integrals also serve as global measures of distortion. Even the existence of solutions to these multidimensional calculus of variations problem is difficult to prove, however. In Eells and Sampson (1964) it is shown that

$$\int_{\mathcal{M}} \operatorname{Tr}(J^{\top}HJG^{-1})\sqrt{\det G}\, dx^1 \cdots dx^m, \tag{10}$$

which corresponds to the choice $\sigma(\lambda) = \lambda_1 + \ldots + \lambda_m$, is in fact a valid global measure of distortion. More specifically, provided the boundary conditions for f are well-specified (i.e., how the boundary $\partial\mathcal{M}$ of the region of integration in $\mathcal{M}$ maps to a boundary $\partial\mathcal{N}$ in $\mathcal{N}$), solutions are extrema of a global measure of distortion. A useful physical analogy (Eells and Sampson 1964) is to imagine wrapping a curved object made of marble ($\mathcal{N}$) by an elastic sheet ($\mathcal{M}$); **harmonic maps**, which are extrema of (10), can be viewed as solutions corresponding to elastic equilibria. As a familiar example, minimal geodesics on $\mathcal{N}$ are special cases of harmonic maps in which $\mathcal{M}$ is some interval $[a, b] \in \mathbb{R}$ with endpoint boundary conditions given.

Although $\operatorname{Tr}(J^{\top}HJG^{-1})$ by itself cannot be interpreted as a local measure of deviation from an isometry, by imposing appropriate boundary conditions (needed to avoid trivial solutions $J = 0$, or equivalently f = constant), the solutions do in fact admit an interpretation as minimum distortion maps. We refer the reader to the

[2] Recall the spectral norm of a square matrix A is the positive square root of the maximum eigenvalue of $A^{\top}A$. It can also be verified that if λ_i is an eigenvalue of $J^{\top}HJG^{-1}$, then $\lambda_i - 1$ is an eigenvalue of $J^{\top}HJG^{-1} - I$.

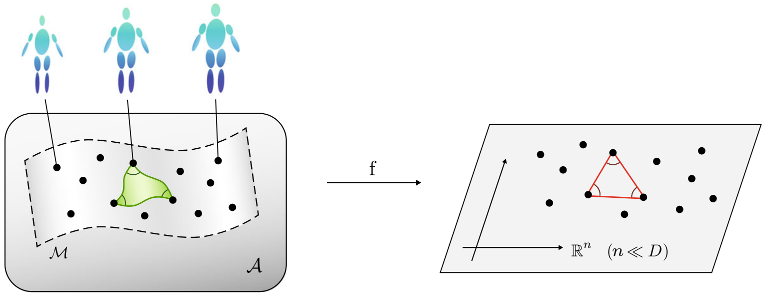

Fig. 2 Manifold learning for non-Euclidean data (e.g., human mass-inertia data composed of symmetric positive-definite matrices)

extensive literature on the theory and applications of harmonic maps, e.g., Eells and Lemaire (1978, 1988), Park and Brockett (1994), and Gu et al. (2004).

Finally, if $\dim(\mathcal{M}) > \dim(\mathcal{N})$, the pullback metric $J^\top HJ$ as well as $J^\top HJG^{-1}$ will always be singular. In this case the above distortion measures are still applicable by using only the nonzero eigenvalues ($m - n$ of the eigenvalues will always be zero). Isometries are unachievable in this case but the distortion measure is still valid and physically meaningful.

2.3 Riemannian distortion and manifold learning

In most of the mathematics literature the source and target Riemannian manifolds $(\mathcal{M}, G)$ and $(\mathcal{N}, H)$ are specified, but typical problems in data analysis and classification are much less well-defined. In manifold learning, for example, only a collection of points $\mathcal{S}$ in some higher-dimensional space $\mathcal{V}$ is given; a Riemannian distortion formulation of manifold learning entails determining some combination of the manifolds $\mathcal{M}$ and $\mathcal{N}$, their respective metrics G and H, as well as the mapping $\mathrm{f} : \mathcal{M} \to \mathcal{N}$. Often boundary conditions on f, or other constraints are required to avoid trivial solutions (e.g., the mapping f becoming constant when minimizing (10)). The integral distortion functional on $\mathcal{M}$ is also replaced by a weighted summation over a set of discrete points in $\mathcal{M}$.

Given the numerous tasks involved, existing manifold learning methods for Euclidean data adopt a divide-and-conquer approach, making explicit a priori choices for, e.g., the manifolds and metrics for the source and target spaces, assuming certain parametric forms for the mapping between the source and target manifolds, or using k-nearest neighbor graphs and geodesic distances to fit a low-dimensional manifold to the data. Reformulating existing algorithms from our Riemannian distortion perspective clearly provides a better qualitative understanding of the distinguishing features of each algorithm, as well as a means for more rigorous quantitative analysis. Our Riemannian distortion framework also leads to several novel manifold learning algorithms for non-Euclidean data. These issues are further discussed in Sect. 3, with case studies provided in Sect. 4.

3 Manifold learning for non-Euclidean data

We now formalize manifold learning for non-Euclidean data. Let $\mathcal{A}$ be a geodesically complete manifold (i.e., a connected manifold in which the existence of minimal geodesics is guaranteed (Boothby 1986)) of dimension D from which a set of data points $\mathcal{S}$ is drawn. Referring to Fig. 2, the data are assumed to lie on or near an m-dimensional submanifold $\mathcal{M} \subseteq \mathcal{A}$ to be determined, with $m \ll D$. A metric G on $\mathcal{M}$ must first be specified that reflects any features and properties of the data points $\mathcal{S}$. In the next section we propose a sample-based algorithm that directly estimates $JG^{-1}J^{\top}$ from $\mathcal{S}$. The target manifold $\mathcal{N}$ is taken to be $\mathbb{R}^n$ equipped with the Euclidean metric $H = I$, where $n \geq m$ (if $n \geq 2m$, then from the strong version of Whitney's embedding theorem, $\mathrm{f} : \mathcal{M} \rightarrow \mathcal{N}$ can be an isometric immersion). A gradient-based algorithm for minimizing the general Riemannian distortion measure (4) as well as the more specific harmonic map-based measure (10) is then presented.

3.1 Estimation of $JG^{-1}J^{\top}$ for non-Euclidean data

In this section we generalize the Laplace–Beltrami operator-based method of Perrault-Joncas and Meila (2013) for estimating $JG^{-1}J^{\top}$ from a finite set of Euclidean data points to the case of non-Euclidean data. In Perrault-Joncas and Meila (2013) the Laplace–Beltrami operator of the submanifold embedded in Euclidean space is approximated from the Euclidean data points using the normalized graph Laplacian $L \in \mathbb{R}^{N \times N}$ (Coifman and Lafon 2006). The inverse metric $JG^{-1}J^{\top}$ is then estimated at each data point x_i using L and the embeddings $y_i = f(x_i) \in \mathbb{R}^n, i = 1, \ldots, N$ (we elaborate further on this below). This method of estimating $JG^{-1}J^{\top}$ is also applied to Euclidean data manifold learning in McQueen et al. (2016).

To generalize this procedure for estimating $JG^{-1}J^{\top}$ to non-Euclidean data, the normalized graph Laplacian in Coifman and Lafon (2006) is expanded to approximate the Laplace–Beltrami operator of a submanifold embedded in Riemannian ambient space. Let $\mathcal{S} = \{\mathrm{u}_1, \ldots, \mathrm{u}_N\}$ for $\mathrm{u}_i \in \mathcal{M} \subseteq \mathcal{A}, i = 1, \ldots, N$, ($x_i \in \mathbb{R}^m$ in local coordinates) be the finite set of data points. The kernel function in Coifman and Lafon (2006) is replaced by a kernel function $k_h : \mathcal{A} \times \mathcal{A} \rightarrow \mathbb{R}$ of the form

$$k_h(\mathrm{u}, \mathrm{v}) = k\left(\frac{\mathrm{dist}_{\mathcal{A}}(\mathrm{u}, \mathrm{v})^2}{h}\right), \tag{11}$$

where $k(\cdot)$ is an exponentially decaying function (e.g., $k(t) = \exp(-t)$ for $t > 0$), h is a bandwidth parameter, and $\mathrm{dist}_{\mathcal{A}}(\mathrm{u}, \mathrm{v})$ denotes the length of the minimal geodesic in $\mathcal{A}$ between two points $\mathrm{u}, \mathrm{v} \in \mathcal{A}$.[3] Algorithm 1 summarizes our method for obtaining the normalized graph Laplacian. We note here that for the choice of exponential kernel function $k(t) = e^{-t}$, we have $c = \frac{1}{4}$ (Perrault-Joncas and Meila 2013). We refer the

[3] The kernel function defined on Riemannian manifolds as in (11) is known to not be positive-definite in general (Jayasumana et al. 2015; Feragen et al. 2015). However, for our manifold learning purposes that mainly target to capture only the submanifold on which the data points lie, we do not require the positive-definiteness of the kernel.

Algorithm 1 The normalized graph Laplacian L

```
Given: Data points $\mathrm{u}_i \in \mathcal{M}$, $i = 1, \ldots, N$, choice of kernel function $k : \mathbb{R} \to \mathbb{R}$, kernel bandwidth parameter $h$, pairwise distance function $\mathrm{dist}_{\mathcal{A}} : \mathcal{A} \times \mathcal{A} \to \mathbb{R}$.
$K_{ij} = k_h(\mathrm{u}_i, \mathrm{u}_j) = k\left(\frac{\mathrm{dist}_{\mathcal{A}}(\mathrm{u}_i,\mathrm{u}_j)^2}{h}\right), \quad i, j = 1, \ldots, N.$
$K = (K_{ij}), \;\; d_i = \sum_j K_{ij}, \;\; D = \mathrm{diag}(d_i).$
$\tilde{K} = D^{-1} K D^{-1}, \;\; \tilde{d}_i = \sum_j \tilde{K}_{ij}, \;\; \tilde{D} = \mathrm{diag}(\tilde{d}_i).$
$L = \frac{\tilde{D}^{-1}\tilde{K} - I}{c\,h}.$
Output: $L$, ($\tilde{D}$, $\tilde{K}$ if required).
```

reader to Appendix A in Jang (2019) for a justification of the choice of kernel function (11).

We now provide a brief review of the method to estimate the inverse metric $JG^{-1}J^{\top}$ proposed in Perrault-Joncas and Meila (2013). For a point $p \in \mathbb{R}^m$ (in local coordinates) on the manifold $\mathcal{M}$, define a function $q^{\alpha\beta}(x; p) = \frac{1}{2}(f^{\alpha}(x) - f^{\alpha}(p))(f^{\beta}(x) - f^{\beta}(p))$, where $f : \mathbb{R}^m \to \mathbb{R}^n$ is the embedding function in local coordinates satisfying $y_i = f(x_i)$ for $i = 1, \ldots, N$ and $f^{\alpha} : \mathbb{R}^m \to \mathbb{R}$ is a map that extracts the α-th coordinate of f. Applying the Laplace–Beltrami operator to $q^{\alpha\beta}(x; p)$ at $x = p$ results in

$$\Delta q^{\alpha\beta}(p; p) = \sum_{j=1}^{m}\sum_{k=1}^{m} \frac{\partial f^{\alpha}}{\partial x^j}\frac{\partial f^{\beta}}{\partial x^k} g^{jk}\bigg|_{x=p}, \tag{12}$$

which corresponds to the (α, β) entry of $JG^{-1}J^{\top}$ at $x = p$.

Given data points $\mathrm{u}_i \in \mathcal{M}$ ($x_i \in \mathbb{R}^m$ in local coordinates) for $i = 1, \ldots, N$, the (α, β) entry of $JG^{-1}J^{\top}$ at $x = x_i$ can be approximated using the normalized graph Laplacian $L \in \mathbb{R}^{N \times N}$ (obtained from Algorithm 1) as

$$(JG^{-1}J^{\top})_{\alpha\beta}(x_i) = \Delta q^{\alpha\beta}(x_i; x_i) = L_i Q_i^{\alpha\beta}, \tag{13}$$

where $L_i \in \mathbb{R}^N$ is the i-th row of L and $Q_i^{\alpha\beta} = (q^{\alpha\beta}(x_1; x_i), \ldots, q^{\alpha\beta}(x_N; x_i)) \in \mathbb{R}^N$.

Using the matrix representation of the embeddings $Y = \left[y_1, \ldots, y_N\right] \in \mathbb{R}^{n \times N}$ to express $Q_i^{\alpha\beta}$ and gathering (13) for $\alpha, \beta = 1, \ldots, n$, the inverse metric $JG^{-1}J^{\top}$ at $x = x_i$ can then be written

$$JG^{-1}J^{\top}(x_i) = \frac{1}{2}Y(\mathrm{diag}(L_i) - e_i e_i^{\top} L - L^{\top} e_i e_i^{\top})Y^{\top}, \tag{14}$$

where $e_i = (0, \ldots, 1, \ldots, 0) \in \mathbb{R}^N$ is a standard basis vector whose i-th component is one. The inverse metric $JG^{-1}J^{\top}(x_i)$ in (14) is guaranteed to be positive semi-definite:

Proposition 1 *For the normalized graph Laplacian $L \in \mathbb{R}^{N \times N}$ obtained from Algorithm 1 and any embeddings $Y = \left[y_1, \ldots, y_N\right] \in \mathbb{R}^{n \times N}$, the inverse metric $JG^{-1}J^{\top}$ at $x = x_i$ in (14) is positive semi-definite for all $i = 1, \ldots, N$.*

Algorithm 2 Estimating $JG^{-1}J^\top$ (Perrault-Joncas and Meila 2013)

```
Given: The normalized graph Laplacian $L \in \mathbb{R}^{N\times N}$ obtained from Algorithm 1.
Input: Current embedding $Y \in \mathbb{R}^{n\times N}$.
Iteration:
  for $\alpha, \beta = 1, \ldots, n$ do
      $(JG^{-1}J^\top)_{\alpha\beta} = \frac{1}{2}(L(Y_\alpha \cdot Y_\beta) - Y_\alpha \cdot (LY_\beta) - Y_\beta \cdot (LY_\alpha)) \in \mathbb{R}^N,$
    where $(JG^{-1}J^\top)_{\alpha\beta}$ denotes an $N$-dimensional vector comprised of the $(\alpha, \beta)$ entry
    of $JG^{-1}J^\top(x_i)$ for $i = 1, \ldots, N$, $Y_\alpha \in \mathbb{R}^N$ denotes the $\alpha$-th row of $Y$, and $\cdot$ denotes
    the element-wise product.
    for $j = 1, \ldots, n, k = 1, \ldots, N$ do
        $\frac{\partial (JG^{-1}J^\top)_{\alpha\beta}}{\partial Y_{jk}} = \frac{1}{2}\big(\delta_{\alpha j}\big(L(e_k \cdot Y_\beta) - e_k \cdot (LY_\beta) - Y_\beta \cdot (Le_k)\big)$
            $+ \delta_{\beta j}(L(Y_\alpha \cdot e_k) - Y_\alpha \cdot (Le_k) - e_k \cdot (LY_\alpha))\big),$
      where $Y_{jk}$ denotes the $(j, k)$ entry of $Y$, and $\delta_{\alpha j} = 1$ if $\alpha = j$ and 0 otherwise.
    end for
  end for
Output: $JG^{-1}J^\top$, $\frac{\partial (JG^{-1}J^\top)_{\alpha\beta}}{\partial Y_{jk}}$ for all points $\mathrm{u}_i$.
```

We refer the reader to Appendix A.1 for the proof of Proposition 1. The algorithm for estimating $JG^{-1}J^\top$ and its gradient is summarized in Algorithm 2.

3.2 Gradient-based optimization

Minimizing (4) is a multidimensional calculus of variations problem. Here we consider direct numerical gradient-based methods for their optimization. The integral in the objective function is approximated by a finite weighted sum of the integrand over a set of discrete points $\mathrm{u}_i \in \mathcal{M}$ (or $x_i \in \mathbb{R}^m$ in local coordinates), with weights $\tilde{d}_i$ obtained from Algorithm 1, $i = 1, \ldots, N$.[4] The mapping f can be parametrically approximated using the actual embedding function values $y_i = f(x_i) \in \mathbb{R}^n$ for the purpose of manifold learning; in this case the $JG^{-1}J^\top(x_i)$ also need to be evaluated together with the y_i. The optimization parameter θ is set to $\theta = (y_1, \ldots, y_N) \in \mathbb{R}^p \simeq \mathbb{R}^n \times \cdots \times \mathbb{R}^n$ (an N-fold product of $\mathbb{R}^n$, i.e., $p = nN$).

[4] Such a choice for weights is based on the approximation $\tilde{d}_i \approx c' \frac{\sqrt{\det G}}{\rho}(x_i)$ for a constant $c' > 0$, where $\rho : \mathcal{M} \to \mathbb{R}$ is the underlying probability density generating data x_i, satisfying $\rho(x) \geq 0$ for all $x \in \mathbb{R}^m$ and $\int_{\mathcal{M}} \rho(x)\, dx = 1$. We refer the reader to equation (A.1.27) in Appendix A.1 of Jang (2019) for this approximation.

Algorithm 3 Gradient-Based Minimization Algorithm

```
Given: Data points u_i $\in \mathcal{M} \subseteq \mathcal{A}$ (in local coordinates, $x_i \in \mathbb{R}^m$), corresponding weights $\tilde{d}_i \in \mathbb{R}$ obtained from Algorithm 1, $i = 1, \ldots, N$, the parameter vector $\theta = (y_1, \ldots, y_N) \in \mathbb{R}^p$ ($y_i = f(x_i) \in \mathbb{R}^n$), evaluation methods for $JG^{-1}J^\top(x_i)$ and its gradient with respect to $\theta$ (Algorithm 2), output manifold Riemannian metric $H = I \in \mathbb{R}^{n \times n}$, objective function $\sigma(\lambda_1, \ldots, \lambda_m)$.
Input: Initial parameter value (or initial embeddings) $\theta_0 = (y_{1,0}, \ldots, y_{N,0}) \in \mathbb{R}^p$.
Initialize: Set $\theta = \theta_0$.
Iteration:
  while not converged do
    $(y_1, \ldots, y_N) \leftarrow \theta$
    for $i = 1, \ldots, N$ do
      $(JG^{-1}J^\top)_i = JG^{-1}J^\top(x_i)$ using Algorithm 2.
      $\frac{\partial (JG^{-1}J^\top)_i}{\partial \theta_j} = \frac{\partial}{\partial \theta_j}\left(JG^{-1}J^\top(x_i)\right)$, $j = 1, \ldots, p$, using Algorithm 2.
    end for
    $\frac{\partial \mathcal{D}(\theta)}{\partial \theta} = \sum_{i=1}^N \frac{\partial \sigma(\lambda)_i}{\partial \theta} \tilde{d}_i$ according to (16).
    Determine update direction $V \in \mathbb{R}^p$ based on the gradient $\frac{\partial \mathcal{D}(\theta)}{\partial \theta}$
    (e.g., from steepest descent, $V \leftarrow -\frac{\partial \mathcal{D}(\theta)}{\partial \theta}$).
    Perform line search along the update direction to obtain stepsize $\epsilon \geq 0$.
    Update along the update direction with the stepsize $\epsilon$, i.e., $\theta \leftarrow \theta + \Delta\theta$, where $\Delta\theta = \epsilon V$.
  end while
Output: Optimized parameter value $\theta_{\text{opt}} = \theta$.
```

The minimization of (4) is now approximated by minimizing, over $\theta \in \mathbb{R}^p$, an objective function of the form

$$\mathcal{D}(\theta) = \sum_{i=1}^{N} \sigma(\lambda_{1,i}(\theta), \ldots, \lambda_{m,i}(\theta))\, \tilde{d}_i, \tag{15}$$

where $\lambda_{k,i}(\theta)$ denotes the k-th eigenvalue of $J^\top HJG^{-1}$ (or equivalently $JG^{-1}J^\top H$) at x_i. Note that some choices of $\sigma(\lambda)$, e.g., (7), (8), require explicit calculation of the eigenvalues, while others, e.g., (6), (10), do not. In the latter case, the gradient of the objective function is simply

$$\frac{\partial \mathcal{D}}{\partial \theta} = \sum_{i=1}^{N} \frac{\partial \sigma(\lambda)_i}{\partial \theta} \tilde{d}_i, \tag{16}$$

where $\sigma(\lambda)_i$ denotes the value of $\sigma(\lambda)$ at x_i. We refer the reader to Jang (2019) for cases where gradients of the eigenvalues of $J^\top HJG^{-1}$ are required, e.g., $\sigma(\lambda)$ in (7), (8).

The gradient-based iterative numerical algorithm is summarized in Algorithm 3. The algorithm for the choice $\sigma(\lambda) = \sum_i (\lambda_i - 1)^2$ in (6) is referred to as the least-squares spectral distortion algorithm. For the choice $\sigma(\lambda) = \sum_i \lambda_i$ corresponding to the harmonic mapping distortion (10), additional boundary conditions that are imposed on the mapping f require a modified gradient-based algorithm as explained in the following section.

3.3 Harmonic mapping distortion

For the special case of the harmonic mapping distortion measure (10), the boundary condition $\mathrm{f}(\partial\mathcal{M}) = \partial\mathcal{N}$ is additionally imposed on the mapping f, requiring a modification of Algorithm 3. Given data points $\mathrm{u}_i \in \mathcal{M} \subseteq \mathcal{A}$ with embeddings $y_i \in \mathbb{R}^n$, $i = 1, \ldots, N$, suppose N_b of the u_i are boundary points, denoted $\mathrm{u}_{b(j)}$ for $j = 1, \ldots, N_b$, with $b(j) \in \{1, \ldots, N\}$ all distinct. Let $y_{b(j)}$ be the n-dimensional embeddings for $\mathrm{u}_{b(j)}$, $Y_b \in \mathbb{R}^{n\times N_b}$ be the matrix with $y_{b(1)}, \ldots, y_{b(N_b)}$ as its columns, $Y_r \in \mathbb{R}^{n\times(N-N_b)}$ be the matrix with the y_i corresponding to the interior (non-boundary) points as its columns, and $Y = \begin{bmatrix} Y_b & Y_r \end{bmatrix} \in \mathbb{R}^{n\times N}$. The discretized version of the harmonic mapping distortion minimization (10) then admits a closed-form solution:

Proposition 2 *For $H = I$, the discrete formulation of the harmonic mapping distortion minimization reduces to the following optimization with respect to Y_r:*

$$\min_{Y_r} \mathrm{Tr}(Y(\tilde{D} - \tilde{K})Y^\top) = \mathrm{Tr}(Y_b(\tilde{D}_{bb} - \tilde{K}_{bb})Y_b^\top - 2Y_b\tilde{K}_{br}Y_r^\top + Y_r(\tilde{D}_{rr} - \tilde{K}_{rr})Y_r^\top), \tag{17}$$

where $\tilde{D}$ and $\tilde{K}$ (obtained from Algorithm 1) are rearranged and partitioned as

$$\tilde{D} = \begin{bmatrix} \tilde{D}_{bb} & 0 \\ 0 & \tilde{D}_{rr} \end{bmatrix} \quad \text{and} \quad \tilde{K} = \begin{bmatrix} \tilde{K}_{bb} & \tilde{K}_{br} \\ \tilde{K}_{br}^\top & \tilde{K}_{rr} \end{bmatrix},$$

and Y_b is a constant matrix specified by the boundary condition. A closed-form solution exists for Y_r, given by $Y_r = Y_b W$ with $W = \tilde{K}_{br}(\tilde{D}_{rr} - \tilde{K}_{rr})^{-1} \in \mathbb{R}^{N_b\times(N-N_b)}$. Y is then of the form

$$Y = Y_b \begin{bmatrix} I & W \end{bmatrix}. \tag{18}$$

Furthermore, assume that $\tilde{K}_{ij} = \tilde{K}_{ji} \geq 0$ for all $i, j = 1, \ldots, N$, a graph with $\tilde{K}_{rr}$ as its adjacency matrix is connected, and $\tilde{K}_{br}$ is not a zero matrix. Then every entry of W is non-negative, with the entries of each column of W summing to one, i.e., $\sum_{i=1}^{N_b} W_{ij} = 1$ for $j = 1, \ldots, (N - N_b)$.

From (18), the optimal embeddings for the interior (non-boundary) points can be interpreted as a weighted average of the boundary point embeddings. We refer the reader to Appendix A.3 for the proof of Proposition 2.

When the boundary $\partial\mathcal{N}$ is unspecified, i.e., Y_b is not given, trivial solutions must be excluded in order for the harmonic mapping functional to be meaningful. Alternatively, $\partial\mathcal{N}$ can be determined by minimizing, e.g., (6), (7), (8), or any of the previous distortion measures in which the integrand is a valid local measure of deviation from an isometry. For example, the objective function (6) can be minimized via Algorithm 3 with $\theta = (y_{b(1)}, \ldots, y_{b(N_b)})$. Note from (18) that the derivatives with respect to the boundary points can be calculated via simple chain rule:

$$\frac{\partial s}{\partial Y_{b,ab}} = \sum_{i=1}^{n}\sum_{j=1}^{N} \frac{\partial s}{\partial Y_{ij}} \frac{\partial Y_{ij}}{\partial Y_{b,ab}} = \sum_{j=1}^{N} \frac{\partial s}{\partial Y_{aj}} \begin{bmatrix} I & W \end{bmatrix}_{bj}, \tag{19}$$

where s is any scalar function on Y, $a = 1, \ldots, n$, and $b = 1, \ldots, N_b$.

3.4 Reformulation of some existing algorithms via Riemannian distortion

In addition to leading to novel manifold learning algorithms, our Riemannian distortion framework can also be used to reformulate existing algorithms as the minimization of a corresponding distortion measure. Rearranging the objective functions of existing algorithms according to the distortion formulation (4), fundamental geometric properties about the corresponding distortion measure can be revealed, and subtle differences among the various methods explained in a geometrically revealing way.

For example, the Riemannian relaxation (RR) method of McQueen et al. (2016) corresponds to minimizing the distortion measure $\sigma(\lambda) = \max_i (\lambda_i - 1)^2$ in (7), according to the reformulation (using our Riemannian distortion framework) provided in Appendix A.2. A crucial difference between the RR algorithm and our method is that in the RR method, the input manifold $\mathcal{M}$ is confined to be a submanifold embedded in the Euclidean ambient space $\mathbb{R}^D$, i.e., only Euclidean data are considered. Our method in contrast allows for non-Euclidean data sampled from an ambient Riemannian manifold $\mathcal{A}$. Moreover, their choice of the objective function (7) is computationally more involved than the least-squares spectral distortion of (6), since it requires performing eigenvalue decomposition on the metrics estimated at each data point to compute the objective function and gradient.

In the case of some well-known locality-preserving manifold learning algorithms, e.g., the locally linear embedding (Roweis and Saul 2000), Laplacian eigenmap (Belkin and Niyogi 2003), and diffusion map methods (Coifman and Lafon 2006), the algorithms share the same form as the harmonic mapping distortion (10), i.e., $\sigma(\lambda) = \lambda_1 + \cdots + \lambda_m$; the primary differences among these methods can be traced to the construction of the (pseudo-)metric G, the choice of volume element, and the constraints imposed on the map $\mathrm{f} : \mathcal{M} \to \mathcal{N}$. More explicit details of the differences among these methods are provided in Jang (2019).

4 Manifold learning case studies

This section presents detailed case studies of our manifold learning algorithms for non-Euclidean data sets. We begin however with a well-known Euclidean data set, the Swiss roll with hole, to demonstrate that our Riemannian framework outperforms existing state-of-the-art methods for even Euclidean data. We then consider synthetic data on the manifold P(2) of 2 × 2 symmetric positive-definite matrices, and human mass-inertia data.

4.1 Example: Swiss roll

Using N non-uniformly sampled points with $N = 1500$ for the Swiss roll with hole shown in Fig. 3a, we obtain embeddings for the least-squares spectral distortion (LS) and harmonic mapping distortion (HM) via Algorithm 3. For the harmonic

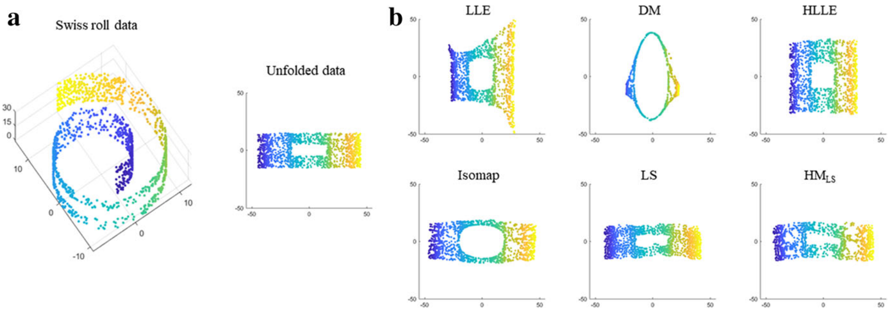

Fig. 3 Manifold learning results obtained for Swiss roll with hole. **a** Swiss roll with hole in $\mathbb{R}^3$ and its unfolded version in $\mathbb{R}^2$; **b** Embeddings obtained from LLE, DM, HLLE, Isomap, LS, and $\mathrm{HM_{LS}}$

mapping distortion (HM), the boundary $\partial\mathcal{N}$ is assumed unspecified; the boundary points are determined to minimize the least-squares spectral distortion over $\theta = (y_{b(1)}, \dots, y_{b(N_b)})$, with $b(j)$ given for $j = 1, \dots, N_b$; this method is denoted $\mathrm{HM_{LS}}$. We refer the reader to Appendix B.1 for further experimental details. Results of the manifold learning are shown in Fig. 3b, together with results obtained for the locally linear embedding (LLE; Roweis and Saul 2000), diffusion map (DM; Coifman and Lafon 2006), Hessian eigenmap (HLLE; Donoho and Grimes 2003), and Isomap (Tenenbaum et al. 2000). The embeddings obtained from the LLE, DM, and HLLE methods fail to capture the original unfolded data manifold of Fig. 3a, as the data manifold does not fit well the algorithm constraints, which usually force the embeddings to be isotropic (Goldberg et al. 2008). Isomap embeddings also result in a distorted shape, by preserving the pairwise graph distances that are overestimated due to the hole. In contrast, LS and $\mathrm{HM_{LS}}$ clearly produce less distorted versions of the original unfolded data of Fig. 3a.

4.2 Synthetic P(2) data

We now consider manifold learning for a set of data points drawn from P(2), the three-dimensional manifold of 2 × 2 symmetric positive-definite matrices. The affine-invariant Riemannian metric on P(n) leading to the distance metric (9) is chosen, i.e., the distance between two P(n) matrices A and B is given by $\mathrm{dist}(A, B)^2 = \sum_{i=1}^{n}(\log \lambda_i)^2$, where λ_i, $i = 1, \dots, n$, are the eigenvalues of $A^{-1}B$. For the data set, 1,000 points are sampled from the two-dimensional submanifold of P(2) shown in Fig. 4a. The submanifold is depicted in (p, q, θ)-coordinates defined in Appendix B.2.1.

Choosing $\mathcal{N} = \mathbb{R}^2$ with Euclidean metric $H = I$, Algorithm 3 is used to find the embeddings for the least-squares spectral distortion (LS), and also the harmonic mapping distortion with boundary $\partial\mathcal{N}$ chosen to minimize LS ($\mathrm{HM_{LS}}$). The metric G projected onto the submanifold is represented by black ellipses along the backbone curve shown in Fig. 4a. The analytic form of the projected metric G as well as further details about the submanifold are provided in Appendix B.2.1.

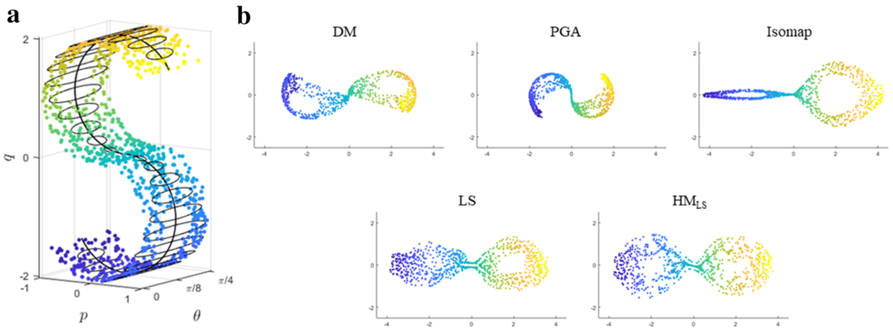

Fig. 4 Manifold learning results obtained for P(2) data. **a** A two-dimensional submanifold of P(2) data represented in (p, q, θ)-coordinates; **b** Embeddings obtained from DM, PGA, Isomap, LS, and HM_{LS}

Table 1 Manifold learning results obtained for P(2) data

Embedding	DM	PGA	Isomap	LS	HM_{LS}
Distance error ($q < 0$)	0.296	0.704	0.256	0.161	**0.0658**
Distance error ($q \geq 0$)	0.352	0.771	**0.0353**	0.113	0.0822
Angle error	0.290	0.498	0.380	0.326	**0.142**

Bold values indicate the best performing algorithms and their values for the given criterion

For HM_{LS}, we assume the data set boundary (i.e., $b(j)$, $j = 1, \ldots, N_b$ described in Sect. 3.3) is specified, with the optimization parameters for the boundary points on $\partial\mathcal{N}$ set to $\theta = (y_{b(1)}, \ldots, y_{b(N_b)})$ and $N_b = 300$. The embedding obtained from Isomap is used as the initial guess for the optimization.

The obtained embeddings are shown in Fig. 4b. For comparison we also plot the results obtained from the diffusion map (DM) with kernel function (11), principal geodesic analysis (PGA; Fletcher and Joshi 2007), and Isomap. As evident from the figures, both the LS and HM_{LS} embeddings better capture the geometry of the submanifold, more accurately reflecting variations along the backbone curve of Fig. 4a with respect to the horizontal axis, and variations in the projected metrics along the vertical axis. In contrast, the DM embedding fails to separate such variations in an orthogonal manner, while the PGA embedding also is not successful in unfolding the nonlinear submanifold onto a plane. The Isomap embedding results in a partially collapsed structure as a side-effect of preserving pairwise graph distances globally. Table 1 lists the pairwise distance errors among points inside the regions corresponding to $q < 0$ and $q \geq 0$, and also the tangent vector angle errors among 30 nearest neighbors (as described in Appendix B.2.2). The embedding for HM_{LS} more accurately preserves the local structure of the submanifold.

4.3 Human mass-inertia data

As shown in Wensing et al. (2018) and Lee and Park (2018), the mass-inertial parameters of a N-link multibody system can be described by a set of N elements of P(4).

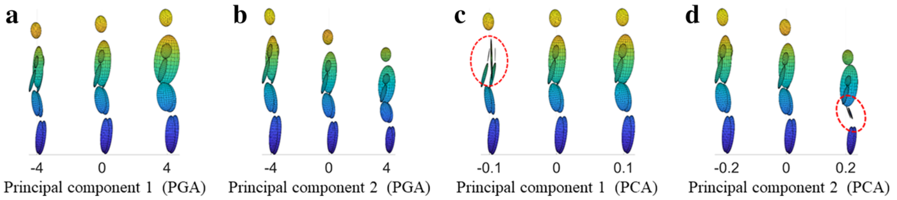
a
b
c
d
-4
0
4
Principal component 1 (PGA)
-4
0
4
Principal component 2 (PGA)
-0.1
0
0.1
Principal component 1 (PCA)
-0.2
0
0.2
Principal component 2 (PCA)

Fig. 5 Principal components (PCs) of the human mass-inertia data obtained from PGA and vector space PCA. **a**–**b** respectively depict the first two PCs obtained from PGA for the range of ±4 standard deviations from the mean; **c**–**d** respectively depict the first two PCs obtained from PCA for different range of standard deviations from the mean

Following these ideas, we define the manifold of human inertial parameters $\mathcal{I}$ as an N_l-fold product of P(4), i.e., $\mathcal{I} \simeq \mathrm{P}(4)^{N_l}$, where N_l denotes the number of links comprising the human body. Since P(4) is a ten-dimensional manifold, the total dimensionality of human mass-inertia data is $10N_l$. The Riemannian metric on $\mathcal{I}$ is straightforwardly defined as the direct product of the affine-invariant metrics assigned for each link.

For this case study, human-mass inertia are extracted from the shape data of 1000 human subjects reported in Yang et al. (2014) (see Appendix B.3.1 for the details). We first qualitatively identify the principal variations in body shapes by performing a principal geodesic analysis (PGA) of the data set with respect to the Riemannian metric on $\mathcal{I}$. These results are then compared to a principal component analysis (PCA) of the vector space representation of the data, i.e., an N_l-fold product of ten-dimensional inertial parameter vectors.

The variations corresponding to the first two principal components of PGA are shown in Fig. 5a–b. Inertial parameters for each link are plotted as ellipsoids. It can be observed that the first principal component captures the overall thickness of the body, while the second principal component captures a combination of height and upper body thickness.

For comparison, the variations corresponding to the first two principal components of vector space PCA are shown in Fig. 5c–d. The variations near the mean are qualitatively similar to those obtained for PGA. However, it can be observed that the positive-definiteness requirement is violated even for data points just 0.2 standard deviations away from the mean for both the first two principal components, i.e., the inertial parameters displayed in the dashed red ellipses of Fig. 5c–d, in which the ellipsoids for the infeasible inertial parameters collapse. We refer the reader to Appendix B.3.2 for the higher order principal components.

The variances along the principal components for both PGA and PCA are shown in Fig. 6; the values are normalized with respect to the variances along the first principal components for each method. Note that the relative values obtained for PCA are extremely small; while this may seem to imply that only variations corresponding to the first principal component are dominant, meaningful variations can still be observed along the second principal component in Fig. 5d. This shows another pitfall of using vector space representations that fails to take into account the underlying geometry of the space of mass-inertia parameters.

We now perform manifold learning for the human mass-inertia data set taking into account the Riemannian structure of $\mathcal{I}$. Embeddings of the human mass-inertia data

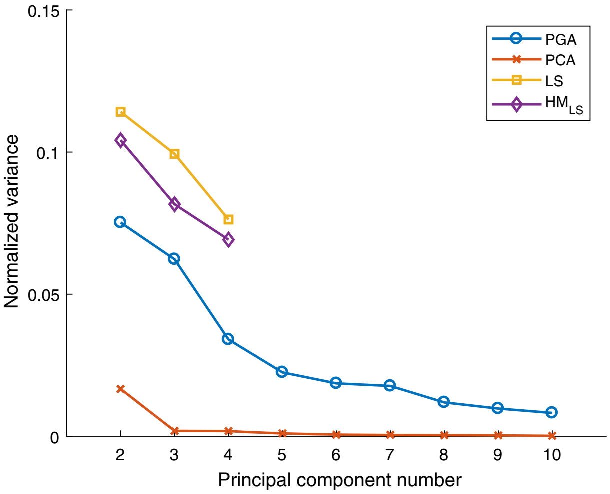

Fig. 6 The normalized variances along the principal components of embeddings obtained from PGA, vector space PCA, LS, and $\mathrm{HM}_{\mathrm{LS}}$

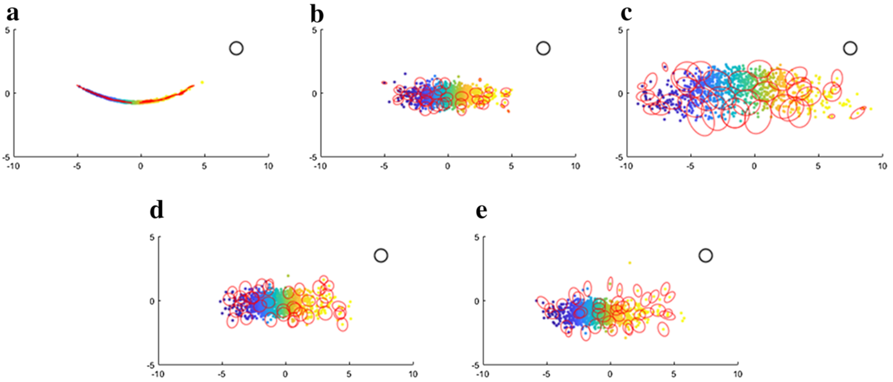

Fig. 7 Manifold learning results obtained for human inertia data. **a** Embedding from DM (used as the initial guess); **b** PGA; **c** Isomap; **d** LS; **e** $\mathrm{HM}_{\mathrm{LS}}$. The red ellipses represent the estimated metric $(JG^{-1}J^{\top})$

set are obtained that minimize the least-squares spectral distortion (LS) and also the harmonic mapping distortion with boundary $\partial\mathcal{N}$ chosen to minimize LS ($\mathrm{HM}_{\mathrm{LS}}$). Algorithm 3 is applied with G chosen to be the metric on $\mathcal{I}$ projected onto $\mathcal{M}$, i.e., the submanifold embedded in $\mathcal{I}$ formed by the data set. We set the dimension of the target manifold to four, i.e., $\mathcal{N} = \mathbb{R}^4$, with $H = I$. For $\mathrm{HM}_{\mathrm{LS}}$, the data set boundary (i.e., the $b(j)$, $j = 1, \ldots, N_b$ described in Sect. 3.3) is specified by choosing increasingly larger values of $\tilde{d}_i$ with $N_b = 500$. The embedding from the diffusion map (DM) is used as the initial guess for the optimization.

Figure 7 plots the first two axes of the embeddings of the human mass-inertia data obtained from the diffusion map, PGA, Isomap, LS, and $\mathrm{HM}_{\mathrm{LS}}$ (the variances along each axis of the embeddings are plotted in Fig. 6). The curved shape of the

Table 2 Manifold learning results obtained for human inertia data

Embedding	DM	PGA	Isomap	LS	$\mathrm{HM}_{\mathrm{LS}}$
Distance error ($k = 50$)	0.483	0.337	0.283	**0.198**	0.246
Angle error ($k = 30$)	0.817	0.226	0.283	**0.209**	0.242

Bold values indicate the best performing algorithms and their values for the given criterion

initial diffusion map embedding is flattened and more evenly distributed along the vertical axis for both the LS and $\mathrm{HM}_{\mathrm{LS}}$ embeddings. A qualitative comparison of the distortion for each embedding can also be obtained by comparing the estimated metric (14). The red ellipses in Fig. 7 represent the estimated $JG^{-1}J^{\top}$ projected to two-dimensional space. Compared to the identity metric (represented by the black circle), the metrics estimated for the diffusion map embedding show the highest eccentricity, while those of PGA and Isomap vary in size from point to point. The LS and $\mathrm{HM}_{\mathrm{LS}}$ embeddings show more uniform values of $JG^{-1}J^{\top}$ over the range of data points, with less eccentricity and similar areas of the ellipses.

Pairwise distance errors and tangent vector angle errors are compared in Table 2 for each of the embeddings, where the errors are computed in a similar manner as described in Appendix B.2.2 with appropriate modifications to reflect the metric on $\mathcal{I}$. Since ground truth values for these errors are not available, we perform only a local comparison, i.e., the pairwise distance values and the angle between tangent vectors among k-nearest neighbors in the ambient space of $\mathcal{I}$ are set as the reference values. The diffusion map embedding has the largest errors, while the embedding obtained from PGA also shows substantial pairwise distance errors (the projection to a low-dimensional subspace causes distances to be underestimated). The Isomap algorithm tends to overestimate the pairwise graph distances in this case, inducing a more distant embedding and higher pairwise distance errors. The embeddings from both LS and $\mathrm{HM}_{\mathrm{LS}}$ show reduced errors compared to the initial diffusion map embedding. The embedding obtained from LS achieves the best performance in preserving local pairwise distances and angles between tangent vectors of the ambient space, demonstrating the performance advantages of our framework.

5 Conclusion

This paper has presented a unified geometric framework for capturing the distortion of mappings between Riemannian manifolds in a coordinate-invariant way, and shown how manifold learning problems can be formulated within this framework. We have shown how to construct general coordinate-invariant functionals of mappings between Riemannian manifolds, and proposed a family of functionals that measures the proximity of the mapping to an isometry. Based on this framework, we formulate Riemannian distortion measures for manifold learning for non-Euclidean data, together with gradient-based optimization algorithms.

Drawing upon non-Euclidean data examples from human mass-inertia parameter identification as well as a range of synthetic data case studies, we show that our

Riemannian distortion framework is applicable to any manifold learning problem in which the data are known to be drawn from a Riemannian manifold. We emphasize that the results do not depend on the choice of local coordinates used to parametrize the data, and in many cases offer better physical insight into the underlying features that are captured by the embedding.

How to choose meaningful Riemannian metrics appropriate to the problem at hand, and to efficiently calculate and optimize the corresponding Riemannian distortion measures, remain topics of further investigation. Moreover, expanding the set of applications of our Riemannian distortion framework to involve other data analysis and classification tasks is another intriguing area of future research.

Appendix

A: Further mathematical details of manifold learning algorithms

A.1 Proof of Proposition 1

Proof The inverse metric $JG^{-1}J^\top$ at $x = x_i$ is obtained in (14) as

$$JG^{-1}J^\top(x_i) = \frac{1}{2}Y(\mathrm{diag}(L_i) - e_i e_i^\top L - L^\top e_i e_i^\top)Y^\top,$$

where $Y = \left[y_1, \ldots, y_N\right] \in \mathbb{R}^{n\times N}$ is the matrix representation of the embeddings, $L = \frac{1}{ch}(\tilde{D}^{-1}\tilde{K} - I) \in \mathbb{R}^{N\times N}$ is the normalized graph Laplacian ($\tilde{D}$, $\tilde{K} \in \mathbb{R}^{N\times N}$ are obtained from Algorithm 1 and both $c, h > 0$), $L_i \in \mathbb{R}^N$ is the i-th row of L, and $e_i = (0, \ldots, 1, \ldots, 0) \in \mathbb{R}^N$ is a standard basis vector whose i-th component is one.

To see if $JG^{-1}J^\top(x_i)$ is positive semi-definite, it suffices to see if

$$M_i \equiv c\,h\left(\mathrm{diag}(L_i) - e_i e_i^\top L - L^\top e_i e_i^\top\right) \in \mathbb{R}^{N\times N} \tag{20}$$

is positive semi-definite. For any $v = (v_1, \ldots, v_N) \in \mathbb{R}^N$,

$$v^\top M_i v = c\,h\left(\sum_{j,k=1}^{N}\left(\mathrm{diag}(L_i) - e_i e_i^\top L - L^\top e_i e_i^\top\right)_{jk} v_j v_k\right) \tag{21}$$

$$= c\,h\left(\sum_{j=1}^{N} L_{ij}v_j^2 - 2L_{ij}v_i v_j\right) \tag{22}$$

$$= v_i^2 + \sum_{j=1}^{N}(\tilde{D}_{ii})^{-1}\tilde{K}_{ij}(v_j^2 - 2v_i v_j) \tag{23}$$

$$= \sum_{j=1}^{N}(\tilde{D}_{ii})^{-1}\tilde{K}_{ij}(v_i - v_j)^2 \geq 0, \tag{24}$$

where L_{ij} denotes the (i, j) entry of L in (22). In deriving (23)-(24), we use the equalities $L_{ij} = \frac{1}{ch}((\tilde{D}_{ii})^{-1}\tilde{K}_{ij} - \delta_{ij})$ ($\delta_{ij} = 1$ if $i = j$ and 0 otherwise) and $\sum_{j=1}^{N}(\tilde{D}_{ii})^{-1}\tilde{K}_{ij} = 1$, and also the inequality $\tilde{K}_{ij} \geq 0$ for $i, j = 1, \ldots, N$. Since the inequality $v^\top M_i v \geq 0$ holds for any $v \in \mathbb{R}^N$, M_i is positive semi-definite; then $JG^{-1}J^\top(x_i) = \frac{1}{2ch}YM_iY^\top$ also becomes positive semi-definite for all $i = 1, \ldots, N$. □

A.2 Riemannian relaxation

In the Riemannian relaxation method of McQueen et al. (2016), $\mathcal{M}$ is chosen to be an m-dimensional submanifold of Euclidean ambient space $\mathbb{R}^D$, with Riemannian metric G corresponding to the Euclidean metric on $\mathbb{R}^D$ projected to $\mathcal{M}$. The target manifold $\mathcal{N}$ is set to be $\mathbb{R}^n$ for some a priori chosen dimension $n \geq \dim(\mathcal{M})$; the Riemannian metric on $\mathcal{N}$ is set to $H = I$.

Given Euclidean data points $u_i \in \mathbb{R}^D, i = 1, \ldots, N$ ($x_i \in \mathbb{R}^m$ in local coordinates), denote their n-dimensional embeddings by $y_i \in \mathbb{R}^n$. The embedding is then obtained as the solution to the following optimization:

$$\min_{y_i} \sum_{i=1}^{N} \|JG^{-1}J^\top(u_i) - I\|^2\alpha_i, \tag{25}$$

where $JG^{-1}J^\top(u_i)$ denotes the $JG^{-1}J^\top$ estimated on u_i using the method presented in Perrault-Joncas and Meila (2013), $\|\cdot\|$ denotes the matrix spectral norm, and α_i are weights. If $n > \dim(\mathcal{M})$, I in (25) is replaced by $R_m R_m^\top$, where $R_m = [r_1, \ldots, r_m] \in \mathbb{R}^{n\times m}$ with $r_i \in \mathbb{R}^n$ the i-th singular vector of $JG^{-1}J^\top$.

From the perspective of our Riemannian distortion framework, assuming the rank of $JG^{-1}J^\top$ is m and the weights α_i in (25) are set to $\tilde{d}_i$ (obtained from Algorithm 1), the objective function in (25) can be expressed as

$$\min_f \mathcal{D}(f) = \int_{\mathcal{M}} \max_i (\lambda_i - 1)^2 \sqrt{\det G}\, dx^1 \cdots dx^m, \tag{26}$$

where the λ_i are the m nonzero eigenvalues of $JG^{-1}J^\top$, which are identical to those of $J^\top JG^{-1}$. Since in practice the numerical estimation of $JG^{-1}J^\top \in \mathbb{R}^{n\times n}$ may yield a rank higher than m when $n > \dim(\mathcal{M})$, one solution is to impose a soft constraint on the rank of $JG^{-1}J^\top$, e.g., in McQueen et al. (2016) the optimization is formulated as

$$\min_f \mathcal{D}(f) = \int_{\mathcal{M}} \max\left(\max_{i\in\mathcal{I}_m}(\lambda_i - 1)^2,\ \max_{i\notin\mathcal{I}_m}\left(\frac{\lambda_i}{\epsilon}\right)^2\right)\sqrt{\det G}\, dx^1 \cdots dx^m, \tag{27}$$

where λ_i are the eigenvalues of $JG^{-1}J^\top$, $\mathcal{I}_m$ denotes the set of indices of the m largest eigenvalues, and $\epsilon > 0$ is a scalar parameter intended to suppress the smaller $(n - m)$ eigenvalues.

A.3 Proof of Proposition 2

Proof For $H = I$, the discretized formulation of the harmonic mapping distortion in the form of (15) is obtained as follows:

$$\mathcal{D}(Y) = \sum_{i=1}^{N} \mathrm{Tr}(JG^{-1}J^{\top}(x_i))\, \tilde{d}_i \tag{28}$$

$$= \frac{1}{2}\sum_{i=1}^{N} \mathrm{Tr}\left(Y(\mathrm{diag}(L_i) - e_i e_i^{\top} L - L^{\top} e_i e_i^{\top})Y^{\top}\right) \tilde{d}_i \tag{29}$$

$$= \frac{1}{2}\mathrm{Tr}\left(Y(\mathrm{diag}(\mathbb{1}_N^{\top}\tilde{D}L) - \tilde{D}L - L^{\top}\tilde{D})Y^{\top}\right) \tag{30}$$

$$= \frac{1}{c\,h}\mathrm{Tr}(Y(\tilde{D} - \tilde{K})Y^{\top}), \tag{31}$$

where $\tilde{K}, \tilde{d}_i, \tilde{D}$ are obtained from Algorithm 1, $L = \frac{1}{c\,h}(\tilde{D}^{-1}\tilde{K} - I) \in \mathbb{R}^{N\times N}$ is the graph Laplacian from Algorithm 1, L_i is the i-th row of L, $e_i = (0, \ldots, 1, \ldots, 0) \in \mathbb{R}^N$ is a standard basis vector whose i-th component is one, and $\mathbb{1}_N \in \mathbb{R}^N$ denotes an N-dimensional vector whose components are all one. In deriving (28)–(31), the estimate of $JG^{-1}J^{\top}$ at x_i in (14), and the equalities $\mathrm{Tr}(J^{\top}HJG^{-1}) = \mathrm{Tr}(JG^{-1}J^{\top}H)$ and $\mathbb{1}_N^{\top}\tilde{D}L = 0$ are used.

Given a constant matrix Y_b specified by the boundary condition, minimizing (31) for Y_r reduces to

$$\min_{Y_r} \mathrm{Tr}(Y(\tilde{D} - \tilde{K})Y^{\top}) = \mathrm{Tr}(Y_b(\tilde{D}_{bb} - \tilde{K}_{bb})Y_b^{\top} - 2Y_b\tilde{K}_{br}Y_r^{\top} + Y_r(\tilde{D}_{rr} - \tilde{K}_{rr})Y_r^{\top}). \tag{32}$$

A closed-form solution for Y_r is obtained as

$$Y_r = Y_b\tilde{K}_{br}(\tilde{D}_{rr} - \tilde{K}_{rr})^{-1} = Y_b W, \tag{33}$$

where $W = \tilde{K}_{br}(\tilde{D}_{rr} - \tilde{K}_{rr})^{-1} \in \mathbb{R}^{N_b \times N_r}$.

Assume that $\tilde{K}_{ij} = \tilde{K}_{ji} \geq 0$ for all $i, j = 1, \ldots, N$, a graph with $\tilde{K}_{rr}$ as its adjacency matrix is connected, and $\tilde{K}_{br}$ is not a zero matrix. Then the matrix $(\tilde{D}_{rr} - \tilde{K}_{rr})$ becomes positive-definite, so that W always exists. The positive-definiteness of $(\tilde{D}_{rr} - \tilde{K}_{rr})$ can be shown from the following inequality: for any $v = (v_1, \ldots, v_{N_r}) \neq 0 \in \mathbb{R}^{N_r}$,

$$v^{\top}(\tilde{D}_{rr} - \tilde{K}_{rr})v = \sum_{i,j=1}^{N_r} (\tilde{D}_{rr} - \tilde{K}_{rr})_{ij} v_i v_j \tag{34}$$

$$= \sum_{i} = 1 =^{N_r} (\tilde{D}_{rr})_{ii} v_i^2 - \sum_{i,j=1}^{N_r} (\tilde{K}_{rr})_{ij} v_i v_j \tag{35}$$

$$= \sum_{i=1}^{N_r} \left(\sum_{k=1}^{N_b} (\tilde{K}_{br})_{ki} \right) v_i^2 + \frac{1}{2} \sum_{i,j=1}^{N_r} (\tilde{K}_{rr})_{ij} (v_i - v_j)^2 > 0, \quad (36)$$

where we use the fact that $(\tilde{D}_{rr})_{ii} = \sum_{k=1}^{N_b} (\tilde{K}_{br})_{ki} + \sum_{k=1}^{N_r} (\tilde{K}_{rr})_{ki}$ in deriving (36). From the direct application of Cramer's rule, it can be shown that each entry of $(\tilde{D}_{rr} - \tilde{K}_{rr})^{-1}$ is non-negative. Since every entry of $\tilde{K}_{br}$ is non-negative, all the entries of W are also non-negative. Furthermore, W satisfies the equation $\mathbb{1}_{N_r}^\top = \mathbb{1}_{N_b}^\top W$ from the equality $\tilde{D}_{rr} \mathbb{1}_{N_r} = \tilde{K}_{rr} \mathbb{1}_{N_r} + \tilde{K}_{br}^\top \mathbb{1}_{N_b}$; hence the entries of each column of W sum to one. □

B: Experimental details for Section 4

B.1 Swiss roll

Here we explain further experimental details for the case study performed in Sect. 4.1. The data points are non-uniformly sampled; referring to the unfolded manifold in Fig. 3a, the density is set to oscillate along the horizontal axis, while uniform along the vertical axis. When choosing the initial parameter value θ_0 for Algorithm 2, locality-preserving embeddings are preferable. As a choice for such an initial parameter value, we use two-dimensional embedding obtained from the Isomap (Tenenbaum et al. 2000). Any other embeddings that preserve locality can also be used as an initial guess, e.g., those from locally linear embedding (LLE; Roweis and Saul 2000), Laplacian eigenmap (LE; Belkin and Niyogi 2003), diffusion map (DM; Coifman and Lafon 2006), Hessian eigenmap (HLLE; Donoho and Grimes 2003), or local tangent space alignment (LTSA; Zhang and Zha 2004).

For the embedding obtained from the Isomap method, we test its five different scalings as the initial parameter value θ_0 for Algorithm 3; we then choose the output embeddings that show the best match to the pairwise distances between ten nearest neighbors in the ambient space. Also note that the kernel bandwidth parameter h for the approximation of the graph Laplacian in Algorithm 2 is chosen to have the same order to the averaged nearest neighbor distance from each of the data points according to Lafon (2004).

B.2 Synthetic P(2) data

B.2.1 Details for the submanifold considered in Section 4.2

The tangent space of P(n) at any $P \in$ P(n) can be identified with S(n), the space of $n \times n$ symmetric matrices. Given $X, Y \in$ S(n), the affine-invariant Riemannian metric at P is defined by the inner product

$$\langle X, Y \rangle_P = \mathrm{Tr}(P^{-1} X P^{-1} Y). \quad (37)$$

Consider the following orthogonal decomposition of $P \in \mathrm{P}(2)$:

$$P = RSR^\top, \tag{38}$$

where $R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \in \mathrm{SO}(2)$ with $\theta \in [0, \frac{\pi}{2})$, and $S = \mathrm{diag}(e^p, e^q)$ for scalar p, q. A local coordinate chart can be defined in terms of (p, q, θ) on an open subset $U = \{P \in \mathrm{P}(2) \mid P \neq cI \text{ for } c > 0\}$. The affine-invariant Riemannian metric in (37) is then represented in (p, q, θ)-coordinates (at $p \neq q$) as

$$ds^2 = \mathrm{Tr}((P^{-1}dP)^2) = dp^2 + dq^2 + 2\left(e^{p-q} + e^{q-p} - 2\right)d\theta^2. \tag{39}$$

For the case study in Sect. 4.2, the data set shown in Fig. 4a is generated by joining two cylinders (with a hole) $\mathcal{C}_1$ and $\mathcal{C}_2$ in (p, q, θ)-coordinates, where $\mathcal{C}_1 = \{(p, q, \theta) \mid p = \sin\theta_S,\ q = -1 + \cos\theta_S,\ \theta_S \in \left[0, \frac{4}{3}\pi\right],\ \theta \in \left[0, \frac{\pi}{4}\right]\}$ and $\mathcal{C}_2 = \{(p, q, \theta) \mid p = \sin\theta_S,\ q = 1 - \cos\theta_S,\ \theta_S \in \left[-\frac{4}{3}\pi, 0\right],\ \theta \in \left[0, \frac{\pi}{4}\right]\}$ (see Fig. 4a; the backbone curve in the figure corresponds to the direction along which θ_S varies). The affine-invariant Riemannian metric on this submanifold (at $\theta_S \neq 0$) is obtained in terms of coordinates (θ, θ_S), $\theta_S \neq 0$, as

$$ds^2 = d\theta_S^2 + 2\left(e^{1+\sqrt{2}\sin(|\theta_S|-\frac{\pi}{4})} + e^{-1-\sqrt{2}\sin(|\theta_S|-\frac{\pi}{4})} - 2\right)d\theta^2. \tag{40}$$

Because of the nonzero Riemannian curvature of this submanifold, isometric embeddings in two-dimensional Euclidean space do not exist.

B.2.2 Evaluation of the pairwise distance and tangent vector angle errors

For data points $x_i \in \mathrm{P}(2)$ and corresponding embeddings $y_i \in \mathbb{R}^2$, $i = 1, \ldots, N$, the pairwise distance error for k nearest points is defined as

$$\text{Pairwise distance error} = \frac{1}{kN}\sum_{i=1}^{N}\sum_{j\in\{\mathrm{NN}_k(i)\}}\left(\|y_i - y_j\| - \mathrm{dist}(x_i, x_j)\right)^2, \tag{41}$$

where y_i denotes the optimal embedding of x_i, $\mathrm{NN}_k(i)$ denotes the set of indices of k nearest neighbor points to x_i, and $\mathrm{dist}(x_i, x_j)$ denotes the ground truth distance between x_i and x_j, i.e., the geodesic distance measured on the submanifold. To measure angles, the tangent vectors are approximated by the difference between nearest neighbors. The tangent vector angle error is defined as

$$\text{Tangent vector angle error} \\ = \frac{2}{k(k-1)N}\sum_{l=1}^{N}\sum_{i,j\in\{\mathrm{NN}_k(l)\}}\left(\mathrm{acos}(\langle v_i, v_j\rangle) - \mathrm{acos}(\langle V_i, V_j\rangle)\right)^2, \tag{42}$$

where v_i, V_i respectively denote the tangent vector from the l-th data point to the i-th data point in the optimal embeddings and the original data points, and $\langle \cdot, \cdot \rangle$ denotes the inner product.

When reporting the final manifold learning results in Table 1, for the reference values to evaluate the pairwise distance error, we numerically obtain the minimal geodesic distances on the submanifolds. Also, the inner product in (37) is used to calculate the reference values for the angles between tangent vectors.

B.3 Human mass-inertia data

B.3.1 Synthesizing human mass-inertia data

Since mass-inertial parameter data for humans are not readily available, we use human shape data from Yang et al. (2014) to synthesize this data set; specifically, assuming uniform mass density, we integrate the volumes of the human body shapes to construct mass-inertial parameter data for the corresponding $N_l = 10$ links.

B.3.2 Further principal components of human mass-inertia data

As a supplement of Fig. 5 in Sect. 4.3, here we provide the third and fourth principal components of the human mass-inertia data obtained from both principal geodesic analysis (PGA) and vector space principal component analysis (PCA). The variations corresponding to the third and fourth principal components of PGA are shown in Fig. 8a–b. Principal component 3 captures variations in the height and torso thickness, and principal component 4 captures variations mainly in the height.

In the case of vector space PCA shown in Fig. 8c–d, the variations near the mean are qualitatively similar to those obtained for PGA. However, the positive-definiteness requirement is violated even for data points just 0.5 standard deviations away from the mean. The ellipsoids for those inertial parameters collapse and can be observed in the dashed red ellipses of Fig. 8c–d.

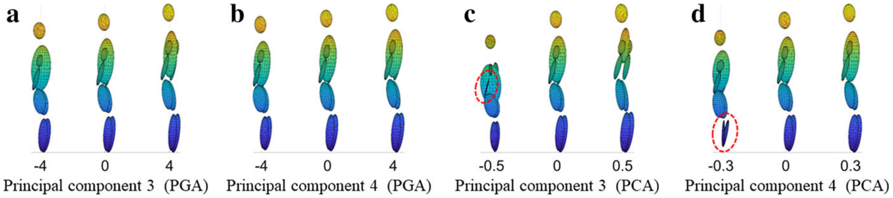
a
-4
0
4
Principal component 3 (PGA)
b
-4
0
4
Principal component 4 (PGA)
c
-0.5
0
0.5
Principal component 3 (PCA)
d
-0.3
0
0.3
Principal component 4 (PCA)

Fig. 8 Principal components (PCs) of the human mass-inertia data obtained from PGA and vector space PCA. **a**–**b** respectively depict the third and fourth PCs obtained from PGA for the range of ±4 standard deviations from the mean; **c**–**d** respectively depict the third and fourth PCs obtained from PCA for different range of standard deviations from the mean

References

Barahona S, Gual-Arnau X, Ibáñez MV, Simó A (2018) Unsupervised classification of children's bodies using currents. Adv Data Anal Classif 12(2):365–397

Belkin M, Niyogi P (2003) Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput 15(6):1373–1396

Belkin M, Niyogi P, Sindhwani V (2006) Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. J Mach Learn Res 7:2399–2434

Boothby WM (1986) An introduction to differentiable manifolds and Riemannian geometry, vol 120. Academic press, Cambridge

Bronstein MM, Bruna J, LeCun Y, Szlam A, Vandergheynst P (2017) Geometric deep learning: going beyond euclidean data. IEEE Signal Process Magazine 34(4):18–42

Coifman RR, Lafon S (2006) Diffusion maps. Appl Comput Harmonic Anal 21(1):5–30

Desbrun M, Meyer M, Alliez P (2002) Intrinsic parameterizations of surface meshes. Comput Graph Forum Wiley Online Libr 21:209–218

Donoho DL, Grimes C (2003) Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. Proc Natl Acad Sci 100(10):5591–5596

Dubrovin BA, Fomenko AT, Novikov SP (1992) Modern geometry-methods and applications Part I. The geometry of surfaces, transformation groups, and fields. Springer, Berlin

Eells J, Lemaire L (1978) A report on harmonic maps. Bull London Math Soc 10(1):1–68

Eells J, Lemaire L (1988) Another report on harmonic maps. Bull London Math Soc 20(5):385–524

Eells J, Sampson JH (1964) Harmonic mappings of Riemannian manifolds. Am J Math 86(1):109–160

Feragen A, Lauze F, Hauberg S (2015) Geodesic exponential kernels: When curvature and linearity conflict. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 3032–3042

Fletcher PT, Joshi S (2007) Riemannian geometry for the statistical analysis of diffusion tensor data. Signal Process 87(2):250–262

Goldberg Y, Zakai A, Kushnir D, Ritov Y (2008) Manifold learning: the price of normalization. J Mach Learn Res 9:1909–1939

Gu X, Wang Y, Chan TF, Thompson PM, Yau ST (2004) Genus zero surface conformal mapping and its application to brain surface mapping. IEEE Trans Med Imag 23(8):949–958

Jang C (2019) Riemannian distortion measures for non-euclidean data. Ph.D. thesis, Seoul National University

Jayasumana S, Hartley R, Salzmann M, Li H, Harandi M (2015) Kernel methods on riemannian manifolds with gaussian rbf kernels. IEEE Trans Pattern Anal Mach Intell 37(12):2464–2477

Lafon SS (2004) Diffusion maps and geometric harmonics. PhD thesis, Yale University Ph.D dissertation

Lee T, Park FC (2018) A geometric algorithm for robust multibody inertial parameter identification. IEEE Robot Autom Lett 3(3):2455–2462

Lin B, He X, Ye J (2015) A geometric viewpoint of manifold learning. Appl Inform 2:3. https://doi.org/10.1186/s40535-015-0006-6

McQueen J, Meila M, Perrault-Joncas D (2016) Nearly isometric embedding by relaxation. In: Lee D, Sugiyama M, Luxburg U, Guyon I, Garnett R (eds) Advances in Neural Information Processing Systems, pp 2631–2639

Mullen P, Tong Y, Alliez P, Desbrun M (2008) Spectral conformal parameterization. Comput Graph Forum Wiley Online Libr 27:1487–1494

Park FC, Brockett RW (1994) Kinematic dexterity of robotic mechanisms. Int J Robot Res 13(1):1–15

Pelletier B (2005) Kernel density estimation on riemannian manifolds. Stat Probab Lett 73(3):297–304

Perrault-Joncas D, Meila M (2013) Non-linear dimensionality reduction: Riemannian metric estimation and the problem of geometric discovery. arXiv preprint arXiv:1305.7255

Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500):2323–2326

Steinke F, Hein M, Schölkopf B (2010) Nonparametric regression between general riemannian manifolds. SIAM J Imag Sci 3(3):527–563

Tenenbaum JB, De Silva V, Langford JC (2000) A global geometric framework for nonlinear dimensionality reduction. Science 290(5500):2319–2323

Vinué G, Simó A, Alemany S (2016) The k-means algorithm for 3d shapes with an application to apparel design. Adv Data Anal Classif 10(1):103–132

Wensing PM, Kim S, Slotine JJE (2018) Linear matrix inequalities for physically consistent inertial parameter identification: a statistical perspective on the mass distribution. IEEE Robot Autom Lett 3(1):60–67

Yang Y, Yu Y, Zhou Y, Du S, Davis J, Yang R (2014) Semantic parametric reshaping of human body models. In: 3D Vision (3DV), 2014 2nd International Conference on, IEEE, vol 2, pp 41–48

Zhang T, Li X, Tao D, Yang J (2008) Local coordinates alignment (lca): a novel manifold learning approach. Int J Pattern Recogn Artif Intell 22(04):667–690

Zhang Z, Zha H (2004) Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. SIAM J Sci Comput 26(1):313–338

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Cheongjae Jang[1] · **Yung-Kyun Noh**[2] · **Frank Chongwoo Park**[1]

✉ Frank Chongwoo Park
fcp@snu.ac.kr

Cheongjae Jang
jchastro@gmail.com

Yung-Kyun Noh
nohyung@hanyang.ac.kr

[1] Department of Mechanical Engineering, Seoul National University, Seoul, Korea

[2] Department of Computer Science, Hanyang University, Seoul, Korea