Journal of Mathematical Imaging and Vision (2021) 63:1204–1221
https://doi.org/10.1007/s10851-021-01047-x

Elastic Shape Analysis of Planar Objects Using Tensor Field Representations

Ruiyi Zhang[1] · **Anuj Srivastava**[1]

Received: 26 September 2020 / Accepted: 28 June 2021 / Published online: 16 August 2021

Abstract
Shape analysis of objects in images is a critical area of research, and several approaches, including those that utilize elastic Riemannian metrics, have been proposed. While elastic techniques for shape analysis of curves are pretty advanced, the corresponding results for higher-dimensional objects (surfaces and disks) are less developed. This paper studies shapes of solid planar objects that are embeddings of a compact domain—a unit square or a unit disk—in $\mathbb{R}^2$. Specifically, it introduces a mathematical representation of objects using tensor fields and uses a re-parametrization-invariant Riemannian metric on these tensor fields to analyze object shapes elastically. The essential contribution here is developing an efficient numerical technique to map tensor fields back to the object space, allowing one to approximate geodesic paths in these objects' shape spaces. Finally, the paper extends this framework to reach landmark-driven registration and improve geodesic computations. The paper illustrates this framework using several simulated and natural objects.

Keywords Elastic shape analysis · Invariant metrics · Landmark registration · Shape geodesics

1 Introduction

Characterization of objects in image or video data is an integral part of computer vision and image understanding, and shapes of objects play an essential role in this characterization. Since objects display a tremendous variability in their appearances, even within the same object class, the *statistical* analysis of shapes becomes critical. Statistical shape analysis of objects in images has a wide variety of applications, ranging from anatomy, biology, and chemistry on the one hand to surveillance, security, graphics, and gaming on the other. The shapes of objects exhibit huge variability, differing in geometries and topologies, complexity, scales, representations, and functionalities. Shape analysis is challenging because the shape is a property that is invariant to certain transformations—translation, rotation, scale, and parameterization—of objects, and this fact rules out the use of classical Euclidean calculus for analyzing shapes.

✉ Ruiyi Zhang
ruiyi.zhang@stat.fsu.edu

Anuj Srivastava
anuj@stat.fsu.edu

1 Departmentof Statistics, Florida State University, Tallahassee, FL, USA

To develop statistical models and testing procedures for shapes, one needs precise mathematical representations and invariant metrics that can efficiently capture shape variability within and across shape classes while being invariant to the transformations mentioned above. This necessitates narrowing the scope and focusing on a specific family of objects to pursue more detailed studies. Here, we focus on solid planar shapes extracted from 2D images, with an interest in objects' boundaries *and* interiors. Mathematically speaking, we are interested in objects specified as mappings of compact planar domains (the unit square or the unit disk) into $\mathbb{R}^2$. Figure 1 shows some examples of these planar objects considered here. The top part shows some natural objects—leaves, face images, and butterfly wings, while the bottom part shows some parameterized mathematical objects. Ignoring the textures and patterns on objects, we will restrict ourselves to: (1) A boundary curve with the parameterized interior and (2) If available, some discrete landmarks (points) in the interior of each object. For such objects, our goal here is to develop a framework that enables computations of geodesic paths between *shapes* of objects.

From a methodological perspective, we focus on Riemannian approaches because they provide comprehensive toolboxes for statistical shape analysis. Within that broad field, we are interested in elastic shape analysis (ESA), an

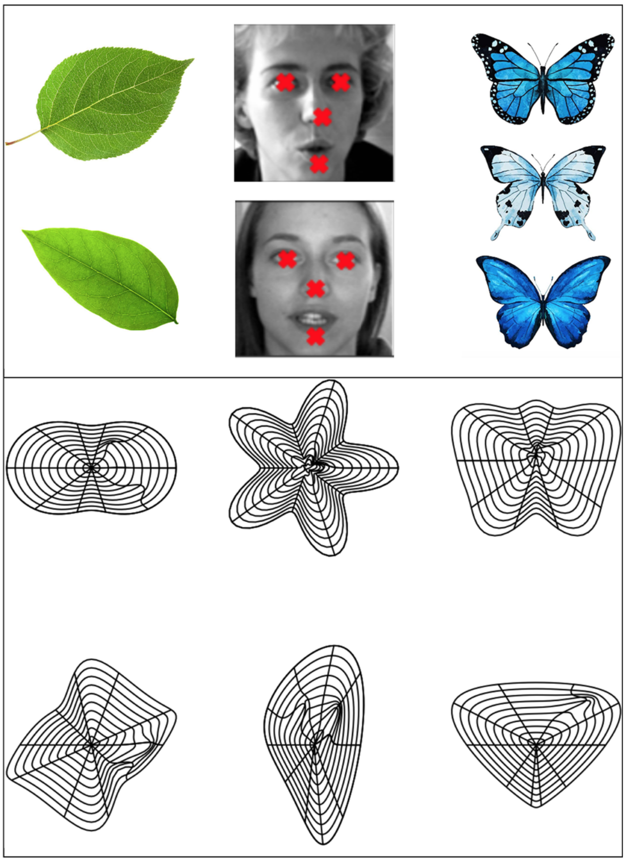

Fig. 1 Examples of planar objects, having boundaries and interiors, studied in this paper. The top two rows show images, while the bottom two show mathematical objects alongwith their coordinate systems

approach that integrates the registration problem (the problem of dense matching of points across objects), with the quantification of shape differences. A typical ESA framework first chooses a representation space for shapes of interest, endows it with an invariant Riemannian structure and derives solutions for items such as the exponential maps, inverse exponential maps, and geodesics. Finally, it induces a metric structure on the original object space by inverting the representation and preserving the registration and metric. Among other things, it provides a tool for computing geodesic paths (optimal deformations) and geodesic distances between *shapes* of objects.

1.1 Related Past Work on 2D Shapes

Before we lay down our approach and outline the main contributions, we briefly review some relevant past research in this area.

1.1.1 2D Curves

Previous work on analyzing 2D shapes using Riemannian methods has primarily focussed on analyzing their boundaries as simple closed curves. This field started with some foundational papers [15,18,19,22,23] and has developed into a very mature discipline with many follow up papers, including [1,2,6,20,21]. While these methods show great power in quantification, registration, and classification of shapes, they are primarily designed for studying the boundaries or the silhouettes of objects. In the case of objects containing information inside the boundaries, which is essential in registration and shape comparisons, it is not easy to extend curve-based representations to incorporate this additional information.

1.1.2 2D Deformable Images

Another approach for studying shapes of 2D objects is to embed them in planar domains and transform the domains using diffeomorphism groups' actions. A well-known approach of this type, named LDDMM [5], uses the diffeomorphisms of Euclidean domains to help register points across objects. While this is a prominent approach for registering images, especially in computational anatomy, it is not naturally suited for shape analysis of objects. Specifically, it does not always result in invariant metrics or geodesics, especially when the action of the diffeomorphism group is not transitive, that are essential for statistical shape analysis. We will, however, use LDDMM to provide registrations of landmarks on objects when such landmark information is available.

1.1.3 Other Approaches

In order to present some other relevant ideas and to distinguish our approach, we consider different mathematical representations of 2D objects, as listed in Table 1. Let $f : D \to \mathbb{R}^2$ represent a parameterized 2D object, where D is a two-dimensional domain (such as the unit disk or the unit square). As the table shows, one can either work with f directly or use some maps derived from f for making the representations invariant. These maps, or features, are typically derivative-based – for $s \in D$, we have the option of using $\nabla f(s) \in \mathbb{R}^{2\times 2}$, $(\nabla f(s))^T \cdot \nabla f(s) \in \mathbb{R}^{2\times 2}$, and so on. Each choice carries its advantages and disadvantages. One would like a representation that allows efficient computation of geodesics *and* invariance to shape-preserving transformations. If we use one of these feature maps, there is an additional requirement of inverting the representation back to f. This inversion is needed to derive shape analysis tools in the original object space rather than the feature space. Typically, one seeks to balance the desire for invariance to

Table 1 Candidate representations of embedding f

Representation	Surface	Gradient	Tensor field	Area density
Symbol	f	∇f	$g = \nabla f^T \cdot \nabla f$	$\mu = \det(\nabla f^T \cdot \nabla f)$
Invariance	None	Translation	Translation + Rotation	Translation + Rotation
Elastic Metric	Difficult	Simple	Simple	Simple
Extrinsic geodesic	Difficult	Complicated	Closed-form	Simple
Registration	Difficult	Complicated	Complicated	Simple
Inversion	N.A.	Trivial	Difficult	Ill-defined

Trans. Translation, *Rot.* Rotation

shape-preserving transformations against the computational cost of constructing geodesics and inverting representations.

A recent paper [3] studies shapes of 2D objects using their gradient vectors fields $\nabla f(s)$, as *one forms*. Such a representation comes with invariance to translation but the computation of geodesics (under an invariant metric) becomes relatively harder. Another paper [4] studies diffeomorphism-based density matching on Euclidean domains, and can be applied to the current problem in a limited way. If f is represented by its area element $\det(\nabla f(s)^T \cdot \nabla f(s)) \in \mathbb{R}$ (last column of Table 1), then some form of shape comparison can be performed in that feature space. While the matching problem here is straightforward, the inversion problem is ill-defined for this case. That is, in going from $f \mapsto \det(\nabla f^T \cdot \nabla f)$ one loses significant information and it seems difficult to reconstruct f from this scalar (area) function.

1.2 Proposed Approach

Considering different possibilities presented in Table 1, we see that representing an object f by its tensor field $g = \nabla f^T \cdot \nabla f$ enjoys some useful properties and remains relatively unexplored in the literature. The advantages of choosing this representation are several:

- The definition of the tensor field g ensures invariance to translation and rotation of objects. One removes the scale easily through area normalization.
- There is a convenient Riemannian metric, for comparing 2D tensor fields, that is invariant to reparametrizations of objects. The geodesic between two tensor fields g_1 and g_2 is also available with an explicit and straightforward expression.
- Optimization over the diffeomorphism group, for registration of points on objects, is relatively straightforward when we use a simpler ($\mathbb{L}^2$-norm based) surrogate objective function.

In this paper, we will use the tensor field g to represent and analyze the shape of f. The only challenge from this choice lies in mapping the representation g back to the object f. There are some open questions associated with the forward map $f \mapsto g$. While we know that this map is neither injective nor surjective, for some natural choices of domains and range spaces, the geometries of these spaces are not well understood. Bypassing these fundamental mathematical questions, we take a purely computational approach to this inversion. For a given tensor field g, we use a numerical approach to find an f whose forward map is as close to the target g as possible. Such a numerical approach has been used successfully in the literature before. For example, Laga et al. in [17] use a numerical inversion of the so-called *square-root normal fields* to study shapes of genus-0 surfaces. We will take a similar multiresolution approach to registration and inversion in this paper.

The main contributions of this paper are the following:

1. We develop a framework for elastic shape analysis of embeddings of the type $f : D \to \mathbb{R}^2$, using a novel tensor-field representation g. This framework involves the development of a numerical approach for inverting the map $f \mapsto g$.
2. We extend past results for computing geodesics between individual tensors, under a well-known invariant Riemannian metric, to a similar geodesic analysis between complete tensor fields on a domain D.
3. Given a small number of registered landmarks (in the interior of the domain D) for each object, we develop a framework for landmark-guided elastic shape analysis of f. We derive efficient numerical techniques for solving for optimal registration of objects while respecting the given landmarks.
4. We illustrate these ideas using simulated and real data examples from drosophila wings and tree leaf databases.

The rest of this paper is as follows. Section 2 summarizes the ESA framework for shape analysis of solid planar shapes. Section 3 develops tools for computing geodesics in the pre-shape space $\mathcal{F}$, using a numerical solution to invert back from g to f. Section 4 solves the optimization problem over the reparametrization group, resulting in an extrinsic geodesic in the shape space $\mathcal{S}$. Section 5 extends this framework to incorporate information from registered landmarks in reg-

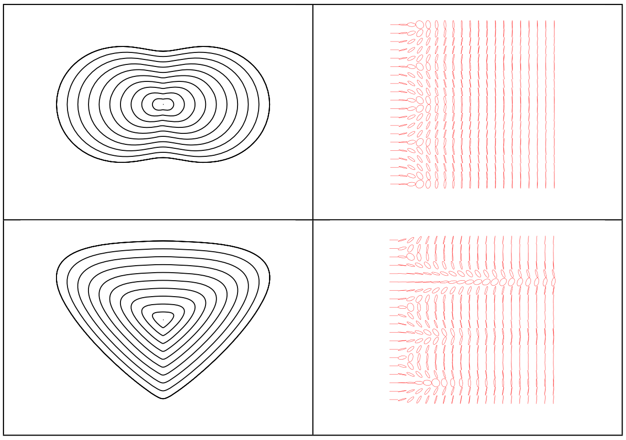

Fig. 2 Examples of objects f and their representations as tensor fields $g = \nabla f^T \nabla f$

istering objects and computing geodesics. It also illustrates these ideas using a wing dataset and a leaf dataset.

2 Mathematical Framework

2.1 Mathematical Representation of Embeddings

Let $\mathcal{F} = \{f : D \to \mathbb{R}^2 | f \text{ is an embedding}\}$ be the object space of solid planar shapes, where D is a compact set in $\mathbb{R}^2$. In this paper, D will either be D_s, the unit square, or D_k, the unit disk. We will utilize a tensor field g to represent and analyze shapes of f in an elastic Riemannian framework.

Definition 1 Given an embedding $f = (f_1(s), f_2(s)) \in \mathcal{F}$, define the tensor field g at a point $s = (x, y)$ to be:

$$g(s) = \nabla f^T \cdot \nabla f = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} \end{bmatrix}^T \begin{bmatrix} \frac{\partial f_1}{\partial x} & \frac{\partial f_1}{\partial y} \\ \frac{\partial f_2}{\partial x} & \frac{\partial f_2}{\partial y} \end{bmatrix}$$
$$= \begin{bmatrix} \langle f_x, f_x \rangle & \langle f_x, f_y \rangle \\ \langle f_x, f_y \rangle & \langle f_y, f_y \rangle \end{bmatrix},$$
$$f_x = \left[\frac{\partial f_1}{\partial x}, \frac{\partial f_2}{\partial x}\right], \quad f_y = \left[\frac{\partial f_1}{\partial y}, \frac{\partial f_2}{\partial y}\right].$$

Denote this mapping as $G : f \mapsto g$.

For each point $s = (x, y) \in D$, $g(s)$ is a 2×2 tensor or a symmetric, positive-definite matrix (SPDM). Thus, g is a tensor field on D. Figure 2 illustrates this representation pictorially for two objects, with the objects shown on the left and the tensor field shown on the right. The domain in this illustration is $D_s = [0, 1]^2$. A tensor or SPDM at each $s \in D_s$ is drawn using an ellipse in the right panels. We will denote the representation space of all tensor fields as $\mathcal{G} \triangleq \{g | g : D \to \text{Sym}^+(2)\}$, where $\text{Sym}^+(2)$ is the space of 2×2 SPDMs. Thus, $G : \mathcal{F} \to \mathcal{G}$.

To analyze the shape of f, we need representations and metrics invariant to rigid rotation, translation, global scaling, and re-parameterization of f. We check the invariance of our chosen representation g.

1. *Scale:* We can get rid of the scale variability through a pre-processing step—rescale f so that its total area is one: $f \mapsto \frac{f}{\mu_f}$, where

$$\mu_f = \int_D \sqrt{\det(\nabla f^T(s) \cdot \nabla f(s))} ds .$$

   If we multiply an object f by a positive scalar $a \in \mathbb{R}_+$, that is $f \mapsto af$, then its tensor field gets multiplied by a^2, $g \mapsto a^2 g$. Imposing a unit area constraint on f, imposes a corresponding constraint on its tensor field g, *i.e.* g should satisfy $\int_D \sqrt{\det(g(s))}\, ds = 1$. One can impose this constraint either in $\mathcal{F}$ or $\mathcal{G}$ space; we will impose it in $\mathcal{F}$ during a preprocessing step. In this paper, we will restrict to the rescaled objects and, with a slight abuse of notation, we will denote the space of these rescaled embeddings as $\mathcal{F}$.
2. *Translation:* The representation g is naturally invariant to the translation of f, because of the use of ∇f in the definition of g.
3. *Rotation:* The tensor field g is invariant to the rotation of f because for a $\tilde{f} = Of$, where $O \in SO(2)$, we have $\tilde{g} = \nabla \tilde{f}^T \cdot \nabla \tilde{f} = \nabla f^T O^T \cdot O \nabla f = \nabla f^T \cdot \nabla f = g$. In fact, the loss of rotation is much stronger than that desired. Even if we introduce a different rotation at each point $s \in D$, *i.e.* $\tilde{f}(s) = O(s) f(s)$, the resulting tensor field remains same. This loss of a full *rotation field* makes it more challenging to invert the map $f \mapsto g$, as discussed later.
4. *Re-Parameterization:* Let Γ be the group of all orientation-preserving diffeomorphisms of D. Elements of Γ play the role of reparameterizations (or domain warpings) for elements of $\mathcal{F}$. For any $\gamma \in \Gamma$ and $f \in \mathcal{F}$, the embeddings f and $f \circ \gamma$ has the same shape. This is illustrated pictorially using two examples in Fig. 3. Each row shows an embedding f, a diffeomorphism γ and their composition $f \circ \gamma$. Under γ, a boundary point of f goes to a boundary point of $f \circ \gamma$, and an interior point maps to an interior point.

If g is the tensor field of an embedding f, then tensor field of $f \circ \gamma$ is not g, *i.e.*, g is not invariant to the reparameterizations of f. Thus, the invariance to reparameterization is not automatic but is enforced algebraically by forming a quotient space corresponding to the action of Γ on $\mathcal{G}$. This is described next.

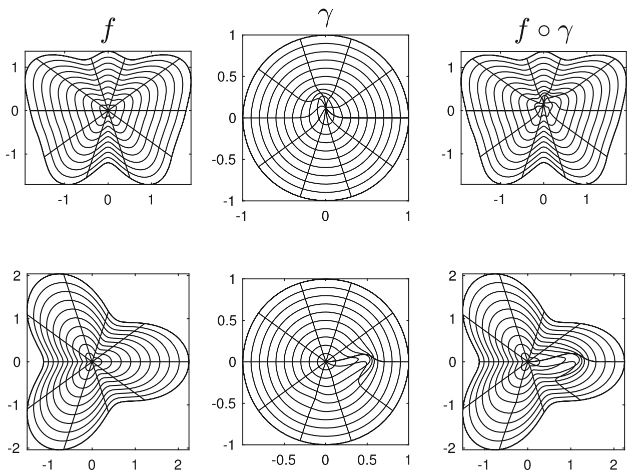

Fig. 3 f and $f \circ \gamma$ have the same shape

Since elements of $\mathcal{G}$ change under reparameterizations, this set is called the *preshape space*. The action of $\varGamma$ on the preshape space $\mathcal{G}$ is as follows.

Definition 2 Define the right group action of $\varGamma$ on $\mathcal{G}$ as $(g \star \gamma) = \nabla(f \circ \gamma)^T \nabla(f \circ \gamma) = J_\gamma^T (g \circ \gamma) J_\gamma$, where J_γ is the Jacobian matrix of γ:

$$J_\gamma = \begin{bmatrix} \frac{\partial \gamma_1}{\partial x} & \frac{\partial \gamma_1}{\partial y} \\ \frac{\partial \gamma_2}{\partial x} & \frac{\partial \gamma_2}{\partial y} \end{bmatrix} .$$

It is useful to note the fact that this group action is linear, i.e.

$$((c_1 g_1 + c_2 g_2) \star \gamma) = c_1 (g_1 \star \gamma) + c_2 (g_2 \star \gamma) ,$$

for any $c_1, c_2 \in \mathbb{R}$, $g_1, g_2 \in \mathcal{G}$, and $\gamma \in \varGamma$.

As shown in Fig. 4, the mapping $f \mapsto g$ and the group actions of $\varGamma$ on these spaces form a diagram that commutes. Whether we apply G to f first and then apply γ to the resulting tensor field g, or if we apply γ to f first then compute its tensor field, the result is identical, *i.e.* $G(f \circ \gamma) = (G(f) \star \gamma)$. (Keep in mind that the actions are different in the two spaces.) The action of $\varGamma$ on $\mathcal{G}$ leads to orbits and partitioning of $\mathcal{G}$ into equivalence classes. The orbit of a tensor field $g \in \mathcal{G}$ under $\varGamma$ is given by:

$$[g] = \text{closure}\left\{(g \star \gamma) | \gamma \in \varGamma\right\} ,$$

which represents all possible re-parameterizations of f: $[f] = \text{closure}\{f \circ \gamma | \gamma \in \varGamma\}$. Each orbit represents a shape of interest uniquely, and the set of all $[g]$ forms the quotient space $\mathcal{S} = \mathcal{G}/\varGamma$; we shall call it the *shape space*.

Now that we have a representation space, we impose a metric structure on it next.

$$\begin{tikzcd} f \arrow[r, "\text{re-parametrization}"] \arrow[d, "\text{representation}"] & f \circ \gamma \arrow[d, "\text{representation}"] \\ g = \nabla f^T \nabla f \arrow[r, "\text{re-parametrization}"] & (g \star \gamma) = J_\gamma^T (g \circ \gamma) J_\gamma \end{tikzcd}$$

Fig. 4 Representation and reparameterization of an embedding f using the tensor field g

2.2 Riemanian Structure on the Representation Space

To compare shapes of embeddings, we need a proper Riemannian metric defined on the representation space $\mathcal{G} = \left\{g | g : D \to \text{Sym}^+(2)\right\}$. Furthermore, we need a metric that is invariant to the action of $\varGamma$ on $\mathcal{G}$. Let $T_g(\mathcal{G})$ denotes the tangent space of $\mathcal{G}$ at the point $g \in \mathcal{G}$.

Definition 3 For any $g \in \mathcal{G}$ and $\delta g_1, \delta g_2 \in T_g(\mathcal{G})$, define an indexed family of Riemannian metrics, indexed by $\kappa \in \mathbb{R}_+$, according to:

$$\begin{aligned} \langle\langle \delta g_1, \delta g_2 \rangle\rangle_g = \int_D \Big[& \text{tr}\left(g^{-1} \delta g_1 g^{-1} \delta g_2\right) \\ & + \kappa \,\text{tr}\left(g^{-1} \delta g_1\right) \text{tr}\left(g^{-1} \delta g_2\right)\Big] \sqrt{\det(g)}\, ds . \end{aligned} \tag{1}$$

For a fixed γ, the differential of the mapping $g \mapsto (g \star \gamma)$ helps map the tangent vectors at g to the tangent vectors at $(g \star \gamma)$. Since the action of $\varGamma$ on $\mathcal{G}$ is linear, the mapping of the tangent vectors is identical. A vector tangent to $\mathcal{G}$ at $\tilde{g} = (g \star \gamma)$, say $\delta \tilde{g}$, is related to the tangent vector at g, say δg, according to:

$$\delta \tilde{g} = J_\gamma^T (\delta g \circ \gamma) J_\gamma = (\delta g \star \gamma) \in T_{g \circ \gamma}(\mathcal{G}) .$$

This mapping helps establish the invariance of the chosen Riemannian metric to the action of $\varGamma$ on $\mathcal{G}$.

Theorem 1 *The Riemannian metric on $\mathcal{G}$, given in* Eq. 1, *is invariant to the action of $\varGamma$ on $\mathcal{G}$. That is, for any $g \in \mathcal{G}$, $\delta g_1, \delta g_2 \in T_g(\mathcal{G})$, and $\gamma \in \varGamma$, we have*

$$\langle\langle \delta \tilde{g}_1, \delta \tilde{g}_2 \rangle\rangle_{\tilde{g}} = \langle\langle \delta g_1, \delta g_2 \rangle\rangle_g , \tag{2}$$

where $\tilde{g} = (g \star \gamma) = J_\gamma^T (g \circ \gamma) J_\gamma$ and $\delta \tilde{g}_i = J_\gamma^T (\delta g_i \circ \gamma) J_\gamma, i = 1, 2$.

Proof Start with the Riemannian metric at the new point $\tilde{g}$:

$$\begin{aligned} \langle\langle \delta \tilde{g}_1, \delta \tilde{g}_2 \rangle\rangle_{\tilde{g}} = \int_D \Big[& \text{tr}\left(\tilde{g}^{-1} \delta \tilde{g}_1 \tilde{g}^{-1} \delta \tilde{g}_2\right) \\ & + \kappa \,\text{tr}(\tilde{g}^{-1} \delta \tilde{g}_1) \text{tr}\left(\tilde{g}^{-1} \delta \tilde{g}_2\right)\Big] \sqrt{\det(\tilde{g})} ds. \end{aligned}$$

For the first term in the integrand, we substitute for $\tilde{g}$ and $\delta\tilde{g}$ and simplify to obtain,

$$\int_D \left[\mathrm{tr}\left(\tilde{g}^{-1}\delta\tilde{g}_1\tilde{g}^{-1}\delta\tilde{g}_2\right)\right]\sqrt{\det(\tilde{g})}ds$$
$$= \int_D \left[\mathrm{tr}\left((g\circ\gamma)^{-1}(\delta g_1\circ\gamma)(g\circ\gamma)^{-1}(\delta g_2\circ\gamma)\right)\right]$$
$$\sqrt{\det(g\circ\gamma)}\det(J_\gamma)ds\ .$$

Substituting by $\tilde{s} = \gamma(s)$, and $d\tilde{s} = \det(J_\gamma)ds$, this term becomes:

$$\int_D \left[\mathrm{tr}\left(g(\tilde{s})^{-1}\delta g_1(\tilde{s})g(\tilde{s})^{-1}\delta g_2(\tilde{s})\right)\right]\sqrt{\det(g(\tilde{s}))}d\tilde{s}\ .$$

Similarly, for the second term, we substitute and simplify to obtain

$$\int_D \left[\mathrm{tr}\left(\tilde{g}^{-1}\delta\tilde{g}_1\right)\mathrm{tr}(\tilde{g}^{-1}\delta\tilde{g}_2)\right]\sqrt{\det(\tilde{g})}ds$$
$$= \int_D \left[\mathrm{tr}\left(g^{-1}\delta g_1\right)\mathrm{tr}\left(g^{-1}\delta g_2\right)\right]\sqrt{\det(g)}d\tilde{s}\ .$$

Combining the two terms, we get the right side of Eqn. 2. □

Now we have a representation space and an invariant Riemannian metric defined on it. It is worth noting that the Riemannian structure on $\mathcal{G}$ imposed by Eq. 1, *i.e.,* the inner product on its tangent bundle, separates into a point-wise calculation. That is, given two representations $g_1, g_2 \in \mathcal{G}$, corresponding to two embeddings f_1, f_2, respectively, computing geodesics between them under the chosen metric boils down to computing individual geodesics between $g_1(s)$ and $g_2(s)$, for each $s \in D$. The problem of constructing geodesic paths under this Riemannian metric has been addressed in paper [12]. Some relevant items from that paper: the Riemannian structure of $\mathrm{Sym}^+(2)$, formulae for geodesics, exponential map, and inverse exponential maps—are presented in Appendix A.

2.3 Geodesics and Distances in Pre-Shape Space

For any two tensor fields $g_1, g_2 \in \mathcal{G}$, the geodesic between them is given by point-wise geodesic at each location. That is, for any $s \in D$, a spatial component α_s, of the full geodesic $\alpha : [0, 1] \to \mathcal{G}$ is as follows. Compute the geodesic between $g_1(s)$ and $g_2(s)$, given by the expression in Theorem 5 and parameterized by τ, and set that to be $\alpha_s(\tau)$, the s component of $\alpha(\tau)$. The full set, for all $s \in D$, constitutes the geodesic $\alpha(\tau) = \{\alpha_s(\tau)|\ s \in D\}$. The corresponding geodesic distance between g_1 and g_2 is given by:

$$\mathrm{d}_{\mathcal{G}}(g_1, g_2)^2 = \int_D \mathrm{d}(g_1(s), g_2(s))^2\mathrm{d}s\ , \qquad (3)$$

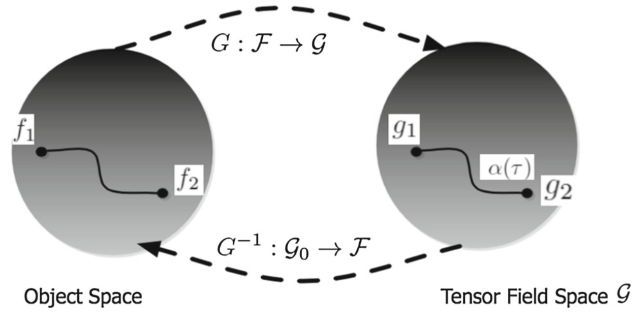

Fig. 5 Relationship between the object space $\mathcal{F}$ and the representation space $\mathcal{G}$

where the expression for $d(\cdot, \cdot)$ is given by Theorem 6.

Another way to state the invariance of chosen metric to the action of Γ on the representation space is by the isometry condition: $\mathrm{d}_{\mathcal{G}}(g_1, g_2) = \mathrm{d}_{\mathcal{G}}(g_1\star\gamma, g_2\star\gamma)$ for all $g_1, g_2 \in \mathcal{G}$ and $\gamma \in \Gamma$. This follows by construction from Theorem 1 or one can prove it directly using the expression for $d_{\mathcal{G}}$.

3 Extrinsic Geodesic Between Two Objects

Given any two objects $f_1, f_2 \in \mathcal{F}$, we can now compute an extrinsic geodesic between their representations $g_1 = \nabla f_1^T \cdot \nabla f_1$ and $g_2 = \nabla f_2^T \cdot \nabla f_2$ in $\mathcal{G}$. The expression for geodesic between g_1 and g_2 is referred to in the previous section. In order to map this geodesic back to $\mathcal{F}$, we face an inversion problem that we discuss next.

3.1 Inverting Tensor Field Representation

The inversion problem is as follows: Let $\alpha : [0, 1] \to \mathcal{G}$ be the geodesic starting from $\alpha(0) = g_1$ and ending at $\alpha(1) = g_2$. For any intermediate time point $\tau \in (0, 1)$, and the geodesic value $g_\tau \triangleq \alpha(\tau)$, find an $f_\tau \in \mathcal{F}$ such that $g_\tau = \nabla f_\tau^T \nabla f_\tau$. This problem is illustrated pictorially in Fig. 5.

This problem raises some fundamental questions: Is this mapping $G : f \mapsto g$ invertible? For the composition $f \mapsto \nabla f \mapsto (\nabla f)^T \cdot (\nabla f)$ to be bijective, we need both the maps ($f \mapsto \nabla f$ and $v \mapsto v^T \cdot v$) to be bijective in their respective spaces. Let $\mathcal{V}$ be the set of all smooth vector fields on D. The first map, $f \mapsto \nabla f$, viewed as a mapping $\mathcal{F} \to \mathcal{V}$ is neither injective nor surjective. The second map, from $\mathcal{V} \to \mathcal{G}$, is not injective but it is surjective because for any $g \in \mathcal{G}$ (with $g(s) \in \mathrm{Sym}^+(2)$), we can find a $v : D \to \mathbb{R}^2$ such that $g(s) = v(s)^T \cdot v(s)$ for each $s \in D$. So, how does one invert a map that is neither injective nor surjective? Our approach is to relax the problem and pose a minimization problem instead. Given a $g_\tau \in \mathcal{G}$, find an embedding f_τ whose forward map $G(f_\tau)$ is as close to g_τ as possible, under the Frobenius norm.

That is, we solve for: $f_\tau = \text{argmin}_{f\in\mathcal{F}} E(f; g_\tau)$, where

$$E(f; g_\tau) = \int_D \|G(f) - g_\tau\|_F^2 \mathrm{d}s \,, \tag{4}$$

where $\|\cdot\|_F$ denotes the Frobenius norm of a matrix. We will solve this minimization problem over $\mathcal{F}$ using numerical optimization, as described next. The resulting solution $\{f_\tau, \tau \in [0, 1]\}$ will be called an *extrinsic geodesic*. This is because the original geodesic α is in the full space $\mathcal{G}$ and not necessarily restricted to be in the image of G.

3.2 Numerical Optimization

To minimize $E(f; g_\tau)$, we are going to use a gradient-descent method and that requires specification of the gradient $\nabla_f E$.

For the functional $E(\cdot; g_\tau) : \mathcal{F} \to \mathbb{R}$, let $d_f E : \mathcal{F} \to \mathbb{R}$ denote its differential at an $f \in \mathcal{F}$. We can evaluate this differential in any direction $w \in \mathcal{F}$ using:

$$\mathrm{d}_f E(w) = \frac{\mathrm{d}}{\mathrm{d}t}|_{t=0} E(f + tw; g_\tau) \,.$$

This is also called the directional derivative of E in the direction of w. Using the $\mathbb{L}^2$ inner-product on $\mathcal{F}$, we can define the gradient $\nabla_f E$ to be the quantity that satisfies: $d_f E(w) = \langle \nabla_f E, w\rangle$, for all $w \in \mathcal{F}$. Let $\mathcal{B}_F$ be an orthonormal basis of $\mathcal{F}$ with respect to the $\mathbb{L}^2$ norm, then we can express the gradient using the directional derivatives according to:

$$\nabla_f E = \sum_{b\in\mathcal{B}_F} (d_f E(b))b \,.$$

Next, the directional derivative $d_f E(b)$ can be derived as follows:

$$\begin{aligned}
\mathrm{d}_f E(b) &= \frac{\mathrm{d}}{\mathrm{d}t}\bigg|_{t=0} E(f + tb; g_\tau)) \\
&= \frac{\mathrm{d}}{\mathrm{d}t}\bigg|_{t=0} \int_D \|g_\tau - \nabla(f + tb)^T \cdot \nabla(f + tb)\|_F^2 ds \\
&= 2\int_D \text{trace}(\Big(\nabla f^T \cdot \nabla f - g_\tau\Big)\Big(\nabla f^T \cdot \nabla b \\
&\quad + \nabla b^T \cdot \nabla f\Big))\mathrm{d}s.
\end{aligned}$$

The gradient update is given by $f \mapsto f - \epsilon\nabla_f E$, for a step size $\epsilon > 0$. To improve convergence speed, we use the accelerated gradient descent (AGD) algorithm. The complete procedure is summarized in Algorithm 1.

We demonstrate this algorithm with a few examples for the unit disk domain D_k. In these examples we use a Fourier basis for elements of $\mathcal{F}$; this basis is specified in the next subsection. Given a true f_{true}, $g_{\text{true}} = G(f_{\text{true}}) = \nabla f_{\text{true}}^T \cdot \nabla f_{\text{true}}$, we solve for $\hat{f} = \underset{f\in\mathcal{F}}{\arg\min}\, E(f; g_{\text{true}})$ using Algorithm 1. In this setting, the inverse exists and the infimum of $E(f; g_{\text{true}})$ is zero. As shown in the bottom two examples of Fig. 6, the algorithm actually recovers the original f_{true} perfectly, and E becomes negligible. However, since the search is gradient-based, it can fail to reach a global solution if the initial condition is far away from that global solution. The top example of Fig. 6 illustrates such a case where the initial f is far from f_{true} and the energy reduces to 9.2527, a small number but not zero. This example highlights the need for a good initial condition for the numerical inversion to be successful.

Algorithm 1 Accelerated gradient-descent for numerical inversion of tensor field

1. Choose a proper embedding f_0 as an initial guess and choose a scalar step size $\epsilon > 0$;
2. Compute the gradient $\nabla_f E(f_0)$, and update $f_1 = f_0 - \epsilon\nabla_f E(f_0)$. Set i=1;
3. Compute $y = f_i + \frac{i-1}{i+2}(f_i - f_{i-1})$;
4. Compute $\nabla_f E(f_i)$ and set $f_{i+1} = y - \epsilon\nabla_f E(f_i)$;
5. Stop if converged, otherwise set $i = i + 1$ and return to Step 3.

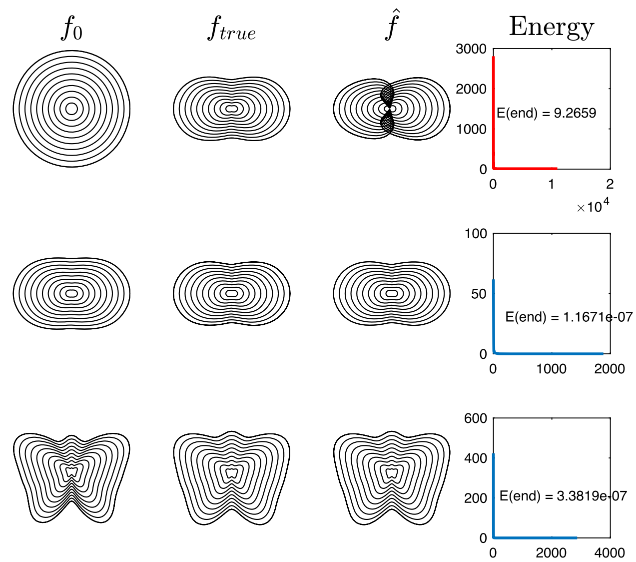

Fig. 6 Simulated examples of numerical inversion of tensor fields using Algorithm 1. Each row shows a different example. The first panel shows the initial guess for the gradient descent algorithm, the second panel shows the ground truth, the third panel shows the final estimated shape, and the last panel shows the change in the cost E during minimization

3.3 Fourier Basis $\mathcal{B}_F$ for $\mathcal{F}$ on D_k

To express the gradient $\nabla_f E = \sum_{b\in\mathcal{B}_F} (\nabla_f E(b))b$, we need an orthonormal basis $\mathcal{B}_F$ of the vector space $\mathcal{F}$, with respect

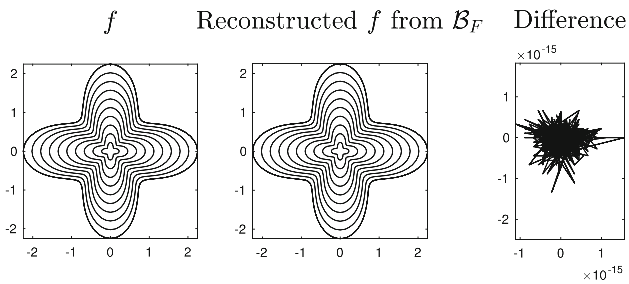

Fig. 7 Reconstruction using Fourier basis on D_k: The first panel shows $f = (9 + 3\cos(8\pi y))\,[x\cos(2\pi y), x\sin(2\pi y)]$; the second panel shows the reconstruction of f using $\mathcal{B}_{\mathrm{F}}$; the last panel shows that the reconstruction error is negligible

to the $\mathbb{L}^2$ norm. In this paper, we will use the Fourier basis

$$\begin{aligned}\mathcal{B}_F &= \{1, x\cos(2k\pi y), x\sin(2k\pi y) : k = 1, 2, \ldots, \}\\ &\quad \times \{1, x\cos(2k\pi y), x\sin(2k\pi y) : k = 1, 2, \ldots\}\,.\end{aligned}$$

Here $s = (x, y) \in D_k$. We then apply the Gram-Schmidt method to this set to make the elements orthonormal. Figure 7 shows an example of representing an object f via this Fourier basis $\mathcal{B}_F$. As shown in the figure, this reconstruction of f can be made arbitrarily close to the original f for smooth objects.

3.4 Extrinsic Geodesics Between Embeddings

Putting together different pieces developed so far, we reach a procedure for computing an *extrinsic* geodesic path between any two objects $f_1, f_2 \in \mathcal{F}$. First, we construct a geodesic between the tensor fields g_1 and g_2 in $\mathcal{G}$, and then invert selected points (or tensor fields) along that geodesic in using Algorithm 1. (These geodesics are termed extrinsic because the first step technically yields geodesics in the larger space $\mathcal{G}$ and not necessarily restricted to the range space of G.) A critical issue in a gradient-descent-based inversion is to choose a good initial condition. In the current context, there are at least two ideas for selecting an initial condition recursively. Firstly, one can go forward in time as follows. Starting from the source f_1, use it as the initial conditional for inverting the tensor field at the next discrete-time. Then, use that solution for the next time step and so on. Secondly, one can start at the target f_2 and go backward in time. Theoretically, the two results—forward and backward—should be identical, but they can differ in practice due to numerical errors and locality of solutions. We take advantage of both the directions and form a combined geodesic by choosing the one with smaller energy at each intermediate time point. These steps for computing a combined geodesic are summarized next.

Figure 8 shows examples of geodesics between some simulated objects. In each row, we show f_1 (first object) and f_2 (last object) and the intermediate objects are equidistant time

---

Algorithm 2 Combined geodesic between embeddings

Suppose we want $N + 1$ time points on the geodesic between f_1 and f_2. Set geo$\mathcal{F}^0 = f_1$, geo$\mathcal{F}^N = f_2$.

1. Represent f_i with the tensor field $g_i = \nabla f_i^T \cdot \nabla f_i, i = 1, 2$.
2. Find the geodesic between g_1 and g_2 using the expression given in Theorem 5, and denote it by $\alpha(\tau)$, $\tau \in \{j/N : j = 0, 1, \ldots, N\}$.
3. To find the forward geodesic, start from geo$\mathcal{F}^0$, and solve for
$$\mathrm{geo}\mathcal{F}^j = \arg\min E\left[G(f); \alpha(j/N)\right], \; j = 1, \ldots, N\,,$$
with geo$\mathcal{F}^{j-1}$ as the initial condition.
4. To find the backward geodesic, start from geo$\mathcal{F}^N$, and solve for
$$\mathrm{geo}\mathcal{F}^j = \arg\min E\left[G(f); \alpha(j/N)\right], \; j = N-1, N-2, \ldots, 0$$
with geo$\mathcal{F}^{j+1}$ as the initial condition.
5. For each j, choose the geo$\mathcal{F}^j$ with the smaller energy from the forward geodesic and the backward geodesic to get the combined geodesic.

---

points along the geodesics obtained using Algorithm 2. The bottom panel shows E values associated with the inversion minimization problem at each of the three intermediate times for each of the three examples.

4 Registration Between Embeddings

So far we have developed tools for computing geodesic paths, albeit extrinsic, between parameterized objects. Next, we consider the problem of registering these objects in order to remove the parameterization variability. The shape space $\mathcal{S}$ is defined to be the quotient space $\mathcal{G}/\Gamma$ and the geodesic in $\mathcal{G}/\Gamma$ is given by: $\alpha_\tau(g_1, (g_2 \star \gamma^*))$ where,

- $\gamma^* = \mathrm{argmin}_{\gamma \in \Gamma}\, d_{\mathcal{G}}(g_1, (g_2 \star \gamma))$,
- $d_{\mathcal{G}}$ is defined in Eqn. 3.

We perform optimization over Γ in two steps:

- First, we register the boundary of f_1 with the boundary of f_2. Then, we register the interior points while keeping the boundary registration fixed. If the first step provides multiple solutions, then we compute the eventual cost (after performing the second step for each of the first step solutions) of all these options and keep the overall minimum.
- For the internal registration, the objective function $d_{\mathcal{G}}(g_1, (g_2 \star \gamma))$ seems computationally expensive to optimize over, despite the availability of a closed-form expression for $d_{\mathcal{G}}$. Instead, we choose a simpler surrogate that is both easier to minimize and to evaluate. Because of that, the final deformation is not guaranteed to be a proper geodesic in the shape space. We will simply call it the extrinsic geodesic between registered objects.

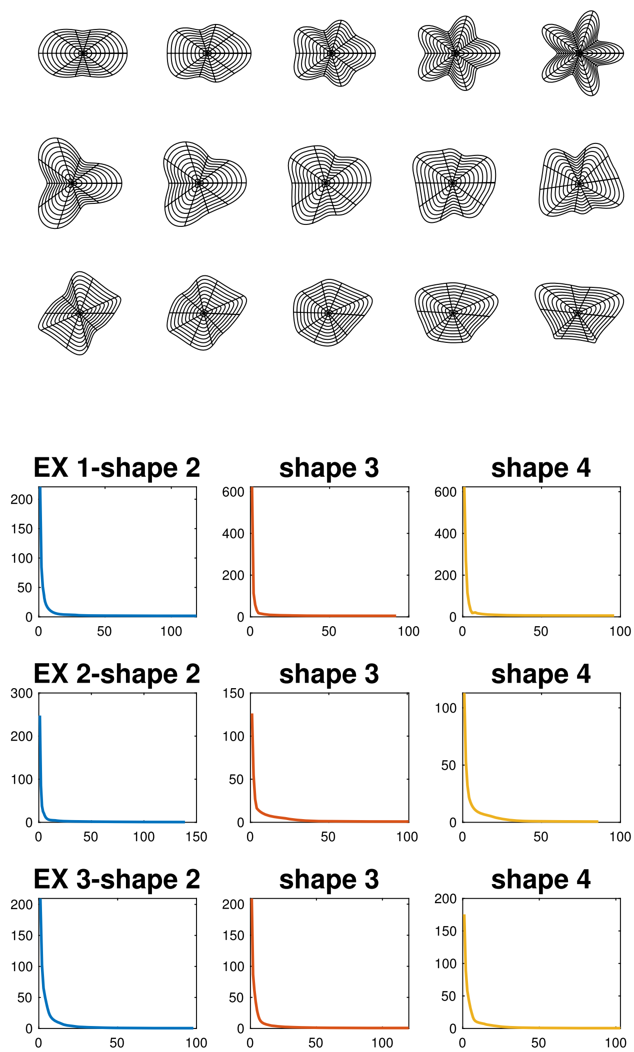

Fig. 8 Simulated examples of extrinsic geodesics in $\mathcal{G}$ using Algorithm 2. The bottom three rows show the energy function E during inversion for the intermediate objects on the geodesic

We describe these two steps in the next sections.

4.1 Registration of Boundaries

There are several possibilities for registering the boundaries of objects. Since boundaries form planar, closed curves, we can use techniques from elastic shape analysis of planar curves [20,22,23] for registration of boundary points. These techniques provide a diffeomorphism of $\mathbb{S}^1$, optimal under a chosen Riemannian distance, to register points along the boundaries of f_1 and f_2.

A much simpler alternative is linear registration that seeks the best circular shift that registers the two closed curves. As shown in Fig. 9, a planar shape with n points on the boundary has n possible linear shifts. We compute a geodesic distance $d_{\mathcal{G}}$ for each candidate shift and choose the one with

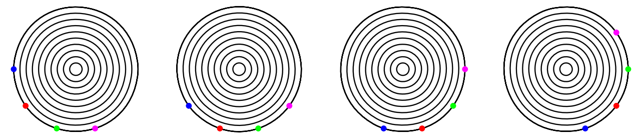

Fig. 9 Linear shifting of points along the boundary of D to register boundaries of two embeddings

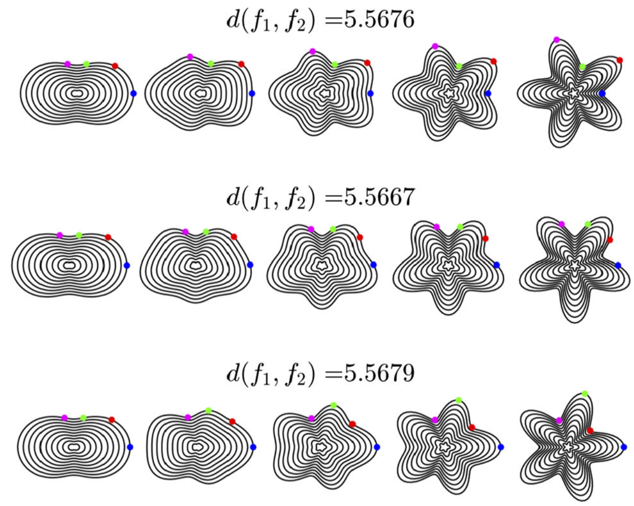

Fig. 10 Simulated examples of extrinsic geodesics in $\mathcal{G}$ inverted back to $\mathcal{F}$ under different linear registrations of boundary points

the smallest distance. This is shown in Fig. 10 where each row represents a geodesic path for a different linear shift of the boundary of the second object. The geodesic distance for each shift is calculated using Eqn. 14. We select the boundary registration that results in the minimum distance.

4.2 Registration of Interiors

As mentioned above, the complicated expression in Eq. 14 makes it difficult to optimize this quantity over Γ. Instead, we a surrogate objective function that is much simpler and intuitive. Secondly, we will perform this optimization over Γ incrementally. At each iteration, we will assume a current diffeomorphism, say γ_0, and we will seek an *incremental diffeomorphism* γ that best reduces the chosen cost. The advantage of using incremental solutions is that a small diffeomorphism can be expressed as an element of the tangent space $T_{\gamma_{id}}(\Gamma)$, which is a vector space. Thus, one can use an orthonormal basis of $T_{\gamma_{id}}(\Gamma)$ to help solve for these incremental diffeomorphisms, one at a time.

Define the energy function associated with a $\gamma \in \Gamma$ as:

$$L(\gamma; g_1, g_2) = \int_D \|g_1 - (\tilde{g}_2 \star \gamma)\|^2 ds, \qquad (5)$$

where $\tilde{g}_2 = G(\tilde{f}_2) = G(f_2 \circ \gamma_0)$ and γ_0 is the current cumulative re-parametrization from previous iterations. We will simplify by writing $L(\gamma)$ when the other two arguments are clear from the context. Let $d_{\gamma_{id}}L : T_{\gamma_{id}}(\Gamma) \to \mathbb{R}$ denote the differential of L at γ_{id} and, thus, $d_{\gamma_{id}}L(\nu)$ is the directional derivative of L at γ_{id} in the direction of $\nu \in T_{\gamma_{id}}(\Gamma)$. Let Ψ denote an orthonormal basis of $T_{\gamma_{id}}(\Gamma)$ with respect to the $\mathbb{L}^2$ norm. Then, the gradient of L with respect to γ can written in terms of the directional derivatives according to the sum: $\nabla_\gamma L = \sum_{\psi\in\Psi}(d_{\gamma_{id}}L(\psi))\psi$. This expression assumes the use of $\mathbb{L}^2$ metric for expressing the differential of L as a gradient.

4.2.1 Directional Derivative of L for D_s

Next we derive an expression for the directional derivative of the cost functional L, focusing on the square domain D_s first.

Theorem 2 *Let $\psi \in T_{\gamma_{id}}(\Gamma)$, $\tilde{f}_2 \in \mathcal{F}$, $\tilde{g}_2 = \nabla\tilde{f}_2^T \cdot \nabla\tilde{f}_2 \in \mathcal{G}$, and define $\phi(\gamma) \stackrel{\Delta}{=} (\tilde{g}_2\star\gamma) \in \mathcal{G}$. Then, the directional derivative of L with respect to γ at γ_{id}, in the direction of $\psi \in \Psi$, is given by*

$$\begin{aligned} d_{\gamma_{id}}L(\psi) &= \left.\frac{dL(\gamma_{id}+t\psi)}{dt}\right|_{t=0} \\ &= -2\int_D \operatorname{tr}\left((g_1-\tilde{g}_2)\left(H^T\cdot\nabla\tilde{f}_2+\nabla\tilde{f}_2^T\cdot H\right)\right)ds, \end{aligned} \tag{6}$$

where:

- *the quantity H is given by* $H = \begin{bmatrix} \nabla\tilde{f}_{2,1x}\cdot\psi & \nabla\tilde{f}_{2,1y}\cdot\psi \\ \nabla\tilde{f}_{2,2x}\cdot\psi & \nabla\tilde{f}_{2,2y}\cdot\psi \end{bmatrix} + \nabla f\cdot\nabla\psi$*; and,*
- *the term* $\nabla\tilde{f}_{2,1x}\cdot\psi = \frac{\partial\tilde{f}_{2,1x}}{\partial x}\psi_1 + \frac{\partial\tilde{f}_{2,1x}}{\partial y}\psi_2$, $\nabla\tilde{f}_{2,1y}\cdot\psi = \frac{\partial\tilde{f}_{2,1y}}{\partial x}\psi_1 + \frac{\partial\tilde{f}_{2,1y}}{\partial y}\psi_2$*, and so on. Here $\psi = (\psi_1,\psi_2)$ denote two components of the vector field ψ.*

Proof Since γ is an incremental diffeomorphism, we can express it as $\gamma = \gamma_{id} + t\psi$ which, in turn, implies that $\nabla\gamma = I_2 + t\nabla\psi$. Using chain rule, we evaluate the derivative:

$$\begin{aligned} \frac{d\left[\nabla(\tilde{f}_2\circ\gamma)\right]}{dt} &= \frac{d\left[(\nabla\tilde{f}_2\circ\gamma)\nabla\gamma\right]}{dt} \\ &= \frac{d\left[\nabla\tilde{f}_2\circ\gamma\right]}{dt}\nabla\gamma + (\nabla\tilde{f}_2\circ\gamma)\frac{d(\nabla\gamma)}{dt}. \end{aligned}$$

We will derive each term on the right one by one. Starting with a part of the first term:

$$\begin{aligned} \frac{d\left[\nabla\tilde{f}_2\circ\gamma\right]}{dt} &= \frac{d}{dt}\begin{bmatrix} (\tilde{f}_{2,1x}\circ\gamma) & (\tilde{f}_{2,1y}\circ\gamma) \\ (\tilde{f}_{2,2x}\circ\gamma) & (\tilde{f}_{2,2y}\circ\gamma) \end{bmatrix} \\ &= \begin{bmatrix} \left\langle(\nabla\tilde{f}_{2,1x}\circ\gamma), \frac{\partial\gamma}{\partial t}\right\rangle & \left\langle(\nabla\tilde{f}_{2,1y}\circ\gamma), \frac{\partial\gamma}{\partial t}\right\rangle \\ \left\langle(\nabla\tilde{f}^2_{2,2x}\circ\gamma), \frac{\partial\gamma}{\partial t}\right\rangle & \left\langle(\nabla\tilde{f}_{2,2y}\circ\gamma), \frac{\partial\gamma}{\partial t}\right\rangle \end{bmatrix} \end{aligned}$$

. Evaluating this expression for $t = 0$, we set $\gamma = \gamma_{id}$, $\nabla\gamma = I_2$, and the full first term becomes

$$\Rightarrow \left.\frac{d\left[\nabla\tilde{f}_2\circ\gamma\right]}{dt}\nabla\gamma\right|_{t=0} = \begin{bmatrix} \left\langle\nabla\tilde{f}_{2,1x},\psi\right\rangle & \left\langle\nabla\tilde{f}_{2,1y},\psi\right\rangle \\ \left\langle\nabla\tilde{f}_{2,2x},\psi\right\rangle & \left\langle\nabla\tilde{f}_{2,2y},\psi\right\rangle \end{bmatrix}.$$

The full directional derivative thus becomes:

$$\Rightarrow \left.\frac{d\left[\nabla(\tilde{f}_2\circ\gamma)\right]}{dt}\right|_{t=0} = \begin{bmatrix} \left\langle\nabla\tilde{f}_{2,1x},\psi\right\rangle & \left\langle\nabla\tilde{f}_{2,1y},\psi\right\rangle \\ \left\langle\nabla\tilde{f}_{2,2x},\psi\right\rangle & \left\langle\nabla\tilde{f}_{2,2y},\psi\right\rangle \end{bmatrix} + \nabla f\cdot\nabla\psi.$$

We define the term on the right of equality as H. Now, since $\phi(\gamma) = (\tilde{g}_2\star\gamma) = \left[\nabla(\tilde{f}_2\circ\gamma)\right]^T\cdot\left[\nabla(\tilde{f}_2\circ\gamma)\right]$, we have

$$\left.\frac{d\phi(\gamma)}{dt}\right|_{t=0} = H^T\cdot\nabla\tilde{f}_2 + \nabla\tilde{f}_2^T\cdot H$$

and the directional derivative of L at γ_{id} is given by

$$\begin{aligned} &\Rightarrow d_{\gamma_{id}}L(\psi) \\ &= \left.\frac{dL(\gamma_{id}+t\psi)}{dt}\right|_{t=0} \\ &= -2\int_D \operatorname{tr}\left((g_1-(\tilde{g}_2\star\gamma_{id}))\left(\left.\frac{d\phi(\gamma)}{dt}\right|_{t=0}\right)\right)ds \\ &= -2\int_D \operatorname{tr}\left((g_1-\tilde{g}_2)\left(H^T\cdot\nabla\tilde{f}_2+\nabla\tilde{f}_2^T\cdot H\right)\right)ds. \end{aligned}$$

□

4.2.2 Direction Derivative of L for D_k

For the disk domain D_k, the expression for the directional derivative of L is the same, except for a change of variable. We have $D_k = \left\{(p,q)|p^2+q^2\le 1\right\}$ and $D_s = \{(x,y)|0\le x,y\le 1\}$. The relationship between the two coordinate systems is $(p,q) = (x\cos(2\pi y), x\sin(2\pi y))$. Therefore, $\nabla f_{pq}(s) = \nabla f_{xy}(s)J(s)$, where

$$\begin{aligned} J(s) &= \frac{\partial(x,y)}{\partial(p,q)} = \begin{bmatrix} \frac{\partial x}{\partial p} & \frac{\partial x}{\partial q} \\ \frac{\partial y}{\partial p} & \frac{\partial y}{\partial q} \end{bmatrix} = \begin{bmatrix} \frac{\partial p}{\partial x} & \frac{\partial p}{\partial y} \\ \frac{\partial q}{\partial x} & \frac{\partial q}{\partial y} \end{bmatrix}^{-1} \\ &= \begin{bmatrix} \cos(2\pi y) & -2\pi x\sin(2\pi y) \\ \sin(2\pi y) & 2\pi x\cos(2\pi y) \end{bmatrix}^{-1} \\ &= \begin{bmatrix} \cos(2\pi y) & \sin(2\pi y) \\ -\frac{\sin(2\pi y)}{2\pi x} & \frac{\cos(2\pi y)}{2\pi x} \end{bmatrix} \stackrel{\Delta}{=} J_{xy} \end{aligned}$$

Thus, it is easy to prove the following result.

$$\nabla f_{pq} = \nabla f_{xy}J_{xy}, \quad g_{pq} = \nabla f_{pq}^T\nabla f_{pq} = J_{xy}^T g_{xy}J_{xy}.$$

For $L(g_{1pq}, \tilde{g}_{2pq}) = \int_{D_s}\left(g_{1pq}-\tilde{g}_{2pq}\right)^2 dp\,dq$, the gradient of L with respect to γ at γ_{id}, in the direction of ψ, is given by:

$$\mathrm{d}_{\gamma_{id}} L(\psi) = -2 \int_{D_s} \mathrm{tr}\left((g_{1pq} - \tilde{g}_{2pq})(H_{pq}^T \nabla \tilde{f}_{2pq} + \nabla \tilde{f}_{2pq}^T H_{pq}) \right) \mathrm{d}p \, \mathrm{d}q. \tag{7}$$

With change of variables, the volume element $dp\ dq = \det(\frac{\partial(p,q)}{\partial(x,y)})\mathrm{d}x\ \mathrm{d}y = \det(J_{xy}^{-1})\mathrm{d}x\ \mathrm{d}y = 2\pi x \mathrm{d}x\ \mathrm{d}y$.

4.3 The Tangent Space $T_{\gamma_{id}}(\Gamma)$

The next step is to specify a convenient basis for the tangent space of Γ at identity γ_{id}, $T_{\gamma_{id}}(\Gamma)$. It is the set of all ψ that are smooth vector fields in the interior of D and zero on the boundaries. Although one can form this basis analytically, using Fourier or other well-known bases, we will take an empirical approach instead. Our idea is to generate lots of small deformations using simple techniques and use them to learn the desired basis. We will use the well-known LDDMM algorithm, in conjunction with some uniformly-spaced landmarks, to generate some vector fields and then use these fields to form a basis $\mathcal{B}_T$.

The use of the LDDMM algorithm to match a set of landmarks is discussed later in Sect. 5.1. We start with a set of uniformly distributed points in the interior of the domain D. See the top row of Fig. 11. Then we select a pair of neighboring points and use the LDDMM algorithm to generate a vector field matching these two points. That is, if γ is the output of LDDMM for matching one point to the other, then set $b = \gamma - \gamma_{id}$. The bottom two rows of Fig. 11 show pictorial examples of this idea. Repeating this step several times generates a set of vector fields $\{b\}$. We apply the Gram–Schmidt algorithm to this set using the $\mathbb{L}^2$ norm to result in an orthonormal set $\mathcal{B}_T$. For example, using a spacing of $\delta = 0.08$ between the neighboring pairs and a grid size of 21×21, we end up with 1407 elements in $\mathcal{B}_T$.

4.4 Registration Algorithm and Examples

Here, we collect all the pieces needed for registering objects and list the full procedure (for optimization over boundary-preserving diffeomorphisms) in Algorithm 3.

We demonstrate this algorithm with a few simulated examples. In the first set of examples, we take an object f_1 and form f_2 by simply re-parameterizing it with an arbitrary $\gamma \in \Gamma$. That is, we construct $f_2 = f_1 \circ \gamma$. Then, we use Algorithm 3 to solve for the optimal registration γ^* between f_1 and f_2. Here, we should get $\min_{\gamma \in \Gamma} L(\gamma; g_1, g_2) = 0$ and $f_2 \circ \gamma^* = f_1$ and this is what we obtain in practice also, as shown in Fig. 12. Each row in this figure shows the original f_2, the registered f_2 (*i.e.,* $f_2 \circ \gamma^*$), the difference $f_1 - f_2^*$, and the evolution of the cost L during optimization. One can see that the algorithm is quite successful in recovering the original γ, or rather its inverse, that was used in creating f_2.

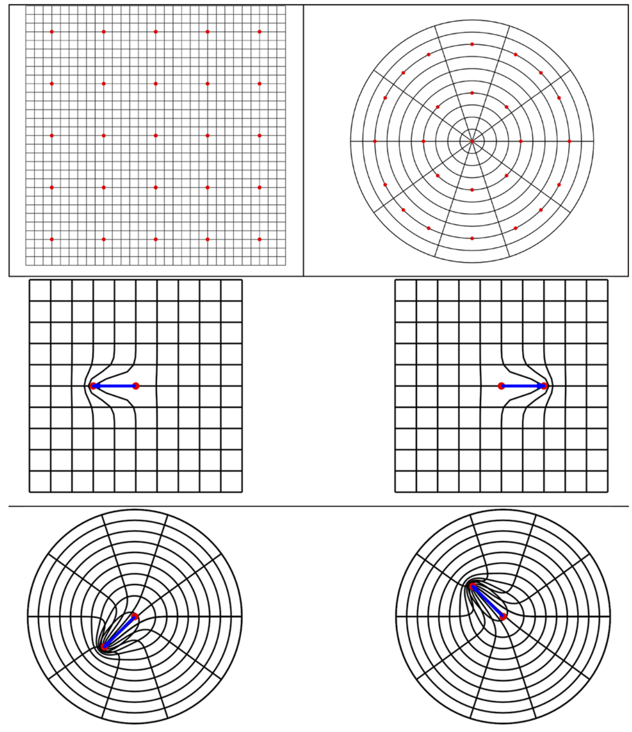

Fig. 11 Construction of orthonormal basis of $T_{\gamma_{id}}(\Gamma)$. The top panel shows a uniform placement of points in D and bottom two rows show deformations resulting from applying LDDMM to match some neighboring points

Algorithm 3 Gradient-descent method for optimization over Γ

Goal: To solve $\gamma^* = \arg\min_{\gamma \in \Gamma} L(\gamma; g_1, (g_2 \star \gamma))$,

1. Choose a proper step size ϵ and let $g_2^0 = g_2$.
2. $L(\gamma) = L(\gamma; g_1, (g_2^i \star \gamma))$, $\nabla L_{\gamma_{id}} \approx \sum_{i=1}^{N} (d_{\gamma_{id}} L(\psi_i))\psi_i$ where $d_{\gamma_{id}} L(\psi_i)$ is given in Eqns. 6, (7).
3. $\gamma^i = \gamma_{id} - \epsilon \nabla L_{\gamma_{id}}$, $g_2^{i+1} = (g_2^i \star \gamma^i)$.
4. If converged, then stop. Else, set $i = i + 1$ and return to Step 2. The final re-parameterization is $\gamma^* = \gamma^0 \circ \gamma^1 \circ \cdots \circ \gamma^{end}$ and the registered second tensor field is $g_2^* = g_2^{end} = (g_2 \star \gamma^*)$.

In the second experiment, we use the same setup, except this time we use f_1 and f_2 with noticeable differences in their shapes. In each row of Fig. 13, we show an experiment with the first three panels showing f_1, γ^* and $f_2 \circ \gamma^*$, respectively. The last panel in each row shows the evolution of L during optimization using Algorithm 3. As these results suggest, the algorithm is successful in reducing the objective function and providing a good registration between points across objects.

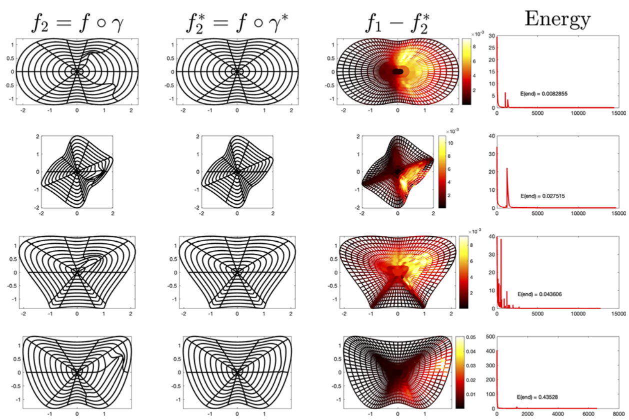

Fig. 12 Simulated examples of optimization over Γ. Given the ground truth f and some $\gamma_0 \in \Gamma$, let $f_1 = f$, $f_2 = f \circ \gamma_0$. Theoretically, the best γ^* that registers f_2 to f_1 should be γ_0^{-1}. The first panel shows the known object $f_2 = f \circ \gamma_0$; the second panel shows after registration $f_2^* = f_2 \circ \gamma^*$; the third panel shows that f_2^* is basically the same as f_1; the last panel shows the energy function in solving γ^* with Algorithm 3

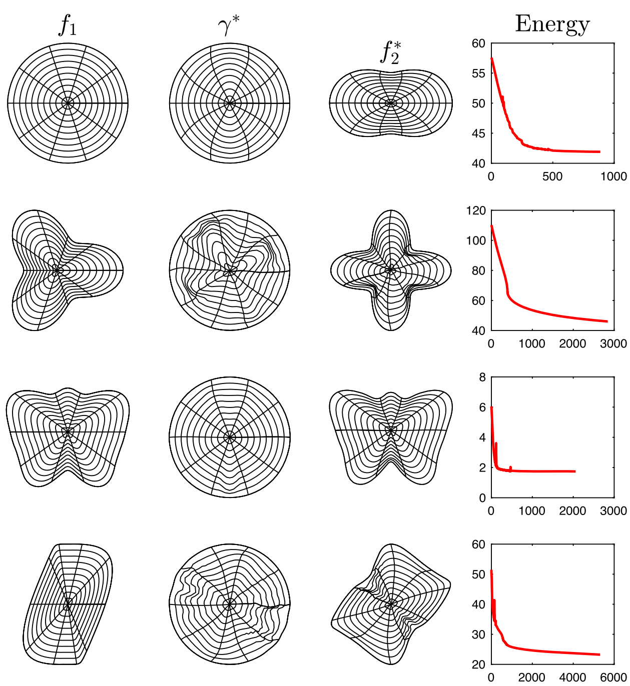

Fig. 13 Simulated examples of optimization over Γ. Given two objects $f_1, f_2 \in \mathcal{F}$, the first panel shows f_1; the second panel shows the best γ^* that registers f_2 to f_1; the third panel shows $f_2^* = f_2 \circ \gamma^*$; the last panel shows the energy function in solving γ^* with Algorithm 3

4.5 Extrinsic Geodesic Between Registered Objects

Once we find a γ that optimally registers $f_1, f_2 \in \mathcal{F}$, we can solve for the geodesic between f_1 and $f_2^* = f_2 \circ \gamma^*$ as laid out in Sect. 3.4. First, we calculate the geodesic between registered tensor fields $g_1 = \nabla f_1^T \cdot \nabla f_1$ and $g_2^* = \nabla f_2^{*T} \cdot \nabla f_2^*$. Then, we invert some sample points along this geodesic by minimizing the energy function: $f_\tau = \text{argmin}_{f \in \mathcal{F}} E(f; g_\tau)$, where E is defined in Eq. 4. In Sect. 3.4, we use the Fourier basis $\mathcal{B}_F$ for evaluating the gradient $\nabla_f E = \sum_{b \in \mathcal{B}_F} (\nabla_f E(b))b$. Theoretically, any basis for representing element $f \in \mathcal{F}$ should suffice for representing $(f \circ \gamma) \in \mathcal{F}$ also. However, we have found that re-parameterization sometimes introduces roughness in $f \circ \gamma$, and one needs a lot of Fourier elements to represent it well. Therefore, we suggest supplementing the basis as follows.

4.5.1 A Basis for Re-Parametrized Functions

Our goal is to form an orthonormal basis for representing functions of the type $f \circ \gamma$, for smooth f and γ. Recall that for small diffeomorphisms, we can use the approximation $\gamma = \gamma_{id} + b$, $b \in \mathcal{B}_T$, where $\mathcal{B}_T$ denotes a basis of $T_{\gamma_{id}}(\Gamma)$. So, one way to construct a basis for re-parameterized functions is by forming a collection $\{f \circ \gamma | f \in \mathcal{B}_F, \gamma = \gamma_{id} + b, b \in \mathcal{B}_T\}$ and orthonromalize it using Gram-Schmidt basis. However, this process results in a huge set, and that leads to inefficiency in representation. Instead, we use the Taylor expansion,

$$f \circ \gamma \approx f \circ \gamma_{id} + \nabla f(\gamma_{id}) \cdot (\gamma - \gamma_{id})$$

to motivate another idea. This expansion shows that a function $f \circ \gamma$ can be approximated using a linear combination of a basis of $f \equiv f \circ \gamma_{id}$ and a basis for $b = (\gamma - \gamma_{id})$. So, we can construct a new basis for $\mathcal{F}$ with $\mathcal{B} = \mathcal{B}_F \cup \mathcal{B}_T$, followed by the Gram–Schmidt procedure to orthonormalize it.

4.5.2 Numerically Inverting Registered Tensor Fields

The next step is to invert a tensor field, denoted as g_τ^*, that lies along the geodesic between g_1 and $g_2^* = \nabla f_2^{*T} \cdot \nabla f_2^*$. For this step, we slightly modify the objective function given in Eq. 4 by replacing f with $f_0 + w$, to get $E(f_0 + w; g_\tau^*)$. Here f_0 represents the initial guess for f and a good guess improves the search results. One possibility for f_0 is to use the inversion of g_τ, along the geodesic between g_1 and $g_2 = \nabla f_2^T \cdot \nabla f_2$, resulting from inversion with the Fourier basis $\mathcal{B}_F$. A better idea is to use a linear interpolation between γ_{id} and γ^* to further improve that initial guess. Algorithm 4 summarizes different steps used in inverting the geodesic between g_1 and g_2^*. We use $\mathcal{B}$ derived in the previous section to represent the unknown variable w.

Figure 14 presents an illustration of Algorithm 4 for calculating the geodesic in $\mathcal{S}$. First row shows the linear interpolation between γ_{id} and γ^*; the second row shows the geodesic between f_1 and f_2 according to Algorithm 2; the third row shows the point-wise composition of the geodesic between f_1 and f_2 and the linear interpolation between γ_{id} and γ^*. Each of these fs are chosen as the initial conditions

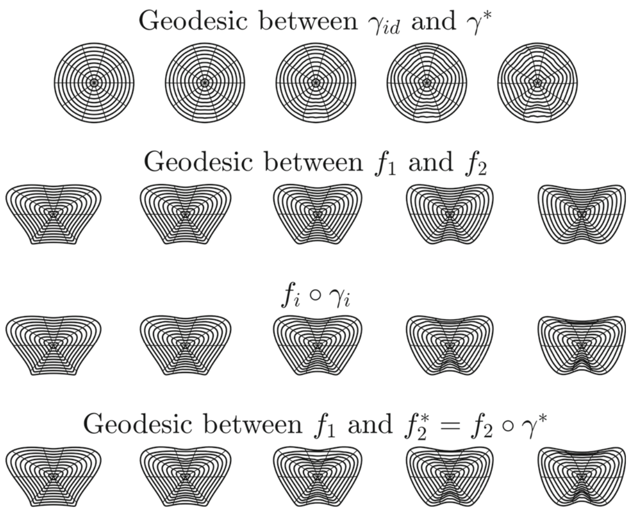

Fig. 14 An example of geodesic in $\mathcal{S}$ illustrating Algorithm 4

for inverting g_τ^* using the gradient descent method at the corresponding τ. The last row shows the final path between f_1 and f_2^* by inverting the geodesic between g_1 and g_2^*. Figure 15 shows the comparison of geodesic between f_1 and f_2 without and with optimization over Γ. Figure 16 shows more examples of geodesic paths between simulated shapes in $\mathcal{S}$.

Algorithm 4 Geodesic Between Registered Shapes

Given $f_1, f_2 \in \mathcal{F}$, solve $\gamma^* = \arg\min_{\gamma \in \Gamma} L(g_1, (g_2 \star \gamma))$. Suppose we want N shapes along the geodesic between f_1 and $f_2^* = f_2 \circ \gamma^*$, $\text{geo}\mathcal{S}^0 = f_1$, $\text{geo}\mathcal{S}^N = f_2^*$.

1. Calculate the geodesic between γ_{id} and γ^* : $\gamma_i = \gamma_{id} + \frac{i}{N}(\gamma^* - \gamma_{id})$, $i = 0, \cdots, N$;
2. Calculate the geodesic between f_1 and f_2 in $\mathcal{F}$: $\text{geo}\mathcal{F}$ with Algorithm 2;
3. Set $\text{geo}\mathcal{F}^i \circ \gamma_i$ as the initial condition for inverting by gradient-descent method and solving for $\text{geo}\mathcal{S}^i$;
4. Calculate the geodesic between g_1 and g_2^* in $\mathcal{G}$: $\text{geo}\mathcal{G}$;
5. To minimize $E(w; g_\tau) = E(G(f_0 + w); g_\tau)$, where g_τ is an intermediate point on $\text{geo}\mathcal{G}$, use iterative updates f with $f - \epsilon \nabla E_f$, $\nabla E = \sum_{b \in \mathcal{B}} \nabla E_f(b) * b$.

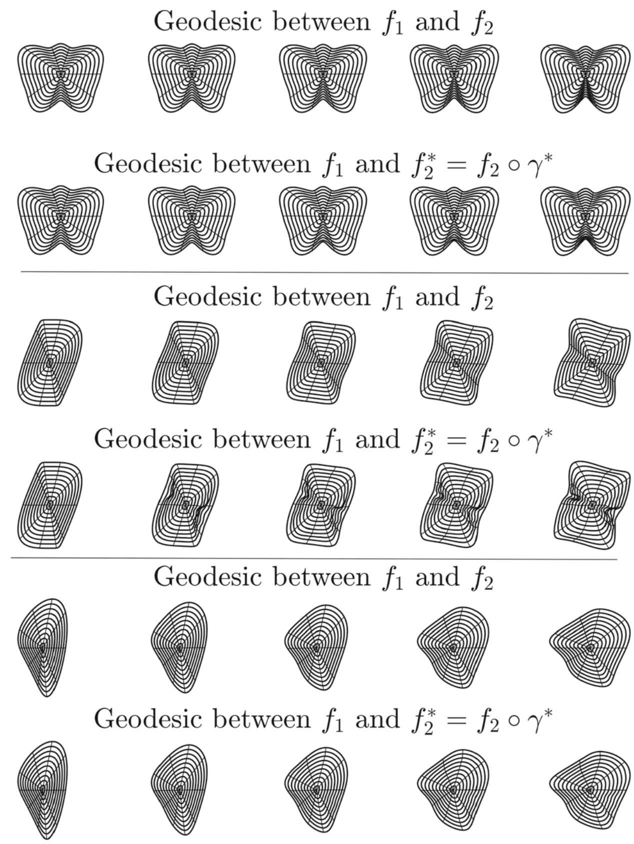

Fig. 15 Examples of geodesics before and after optimization over Γ

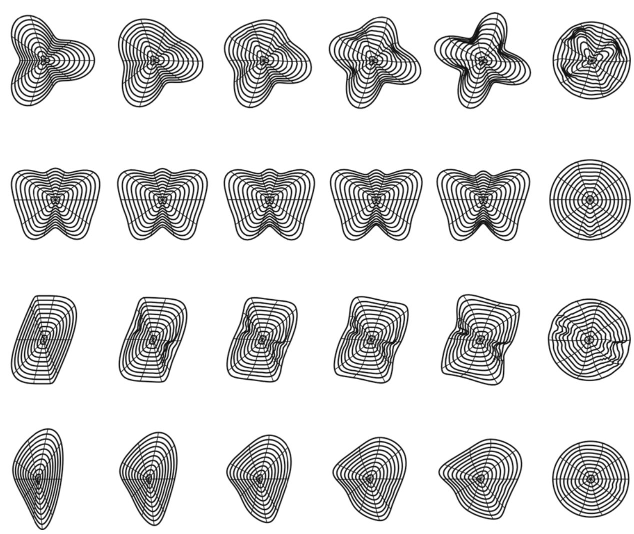

Fig. 16 Each row shows an example of geodesic in $\mathcal{S}$. The last panel in each row shows the optimal registration $\gamma^* = \arg\min_{\gamma \in \Gamma} E(f_1, f_2 \circ \gamma)$

5 Landmark-Guided Shape Analysis

So far, the registration of objects and computation of the geodesic paths were fully automated and did not utilize any additional information. In case we are given a finite set of registered landmarks (points in the domain D) for all objects, how can we incorporate that additional information in the registration and shape analysis of objects? We describe this process next.

We start with the problem of registering the whole objects. Similar to [16], this registration is performed in two steps. The first step is to find an initial registration that matches the given landmarks, irrespective of the rest of the shapes. In mathematical terms, let $f_1, f_2 \in \mathcal{F}$ be two objects and let

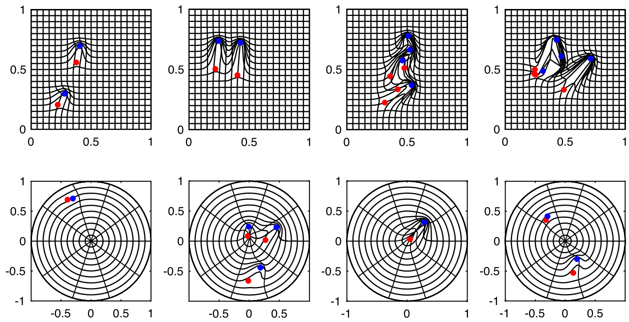

Fig. 17 Examples of using LDDMM to find a γ that registers red landmarks to blue landmarks for $D_{\rm s}$ (top) and $D_{\rm k}$ (bottom) (Color figure online)

$\{x_i \in D, i = 1, 2, \ldots, k\}$, $\{y_i \in D, i = 1, 2, \ldots, k\}$ denote their respective landmarks. For the initial registration, our goal is to find a $\gamma \in \Gamma$ such that $y_i = \gamma(x_i)$. The next step is to optimize L (Eqn. 5) over those elements of Γ that preserve this landmark registration.

5.1 Initial Registration Using LDDMM

We will use the well-known LDDMM framework [5] adapted to landmark problem [14] for this purpose. Given two sets of landmarks $\{x_i, y_i, i = 1, \ldots, j\} \in D$, the goal of landmark registration is to compute a boundary-preserving diffeomorphism $\gamma : D \leftrightarrow D$, such that $y_i = \gamma(x_i)$. The solution presented in [14] constructs a path $\phi_t \in \Gamma, t \in [0, 1]$ that starts at $\phi_0 = \gamma_{id}$ and ends at $\phi_1 = \gamma$. The path ϕ_t is controlled by a time-dependent vector field $d\phi_t/dt = v_t(\phi_t), t \in [0, 1]$. Thus, $\gamma = \phi_1 = \phi_0 + \int_0^1 v_t(\phi_t)dt$ registers the given landmarks.

The optimality of the velocity field is defined by the minimization problem:

$$\min_v \left(\int_0^1 \int_D \|\mathcal{O}v(s,t)\| \, ds dt + \frac{1}{\sigma^2} \sum_{n=1}^{N} \|\phi_1(x_n) - y_n\|^2 \right) \quad (8)$$

where $\phi_1(s) = s + \int_0^1 v_t(\phi_t(s))dt$ and $\mathcal{O}$ is a linear differential operator that controls the smoothness of the velocity field and ensures that $\phi_t, t \in [0, 1]$ are diffeomorphisms.

Figure 17 shows some examples of landmark registration using this solution. In each panel we show a diffeomorphism that matches landmarks shown in red dots—$\{x_n\}$—to the landmarks shown in blue dots – $\{y_n\}$. These diffeomorphisms provide an initial matching of domains.

5.2 Landmark-Preserving Registration

Next, we consider the rest of the information in objects f_1, f_2 and register them while preserving the given registration of landmarks. In other words, we are interested in solving the optimization problem stated in Eq. 5, except the search space for incremental γ is now:

$$\Gamma_0 = \{\gamma \in \Gamma | \gamma(y_i) = y_i, i = 1, 2, \ldots, n\} .$$

That is, the incremental γ's do not change the landmarks on f_2. The new optimization problem is given by

$$\hat{\gamma} = \mathop{\mathrm{argmin}}_{\gamma \in \Gamma_0} \int_D (g_1 - (\tilde{g}_2 \star \gamma))^2 \mathrm{d}s, \quad (9)$$

There are two parts to this matching step: (1 Matching of the boundary curves, and (2) Matching of the interior points while preserving the landmark registration.

5.2.1 Registration of Boundaries

Similar to Sect. 4.1, we find the best circular shift that registers two closed curves representing the two boundaries. Note that since LDDMM-based registration of landmarks does not affect the boundaries of f_1 and f_2, this step of matching the boundaries using circular shifts is independent of that step. In other words, the two optimization problems do not interact.

5.2.2 Registration of Interior

Here, we provide an altered basis for $T_{\gamma_{id}}(\Gamma_0)$ for use in this optimization. The rest of the procedure, as stated in Algorithm 3, remains the same. We need to construct a set $T_{\gamma_{id}}(\Gamma_0)$ whose elements are vector fields that vanish at $\{y_i\}$. Following [16], we begin by providing a construction for an orthonormal basis that vanishes at only one point, say $y_1 \in D$.

1. Generate the full basis for $T_{\gamma_{id}}(\Gamma)$, denoted by $\mathcal{B}_T$ with elements $b_1, b_2, \ldots$.
2. Among the basis elements in $\mathcal{B}_T$, choose two of them, say b_1 and b_2, such that $b_1(y_1), b_2(y_1)$ form a basis of $T_{y_1}(D)$.
3. For each basis element b_i other than b_1 and b_2, replace it by $b_i - (z_1 b_1 + z_2 b_2)$, where scalars z_1 and z_2 are chosen so that $b_i(y_1) = z_1 b_1(y_1) + z_2 b_2(y_1)$.
4. Remove b_1 and b_2 from the basis set.
5. The altered vector fields form a basis of the smooth vector fields that vanish at y_1.
6. Orthonormalize the remaining basis elements using the Gram–Schmidt procedure under the $\mathbb{L}^2$ metric.

This was the case for one landmark. For n landmarks, in Step 2, choose $2n$ basis elements with the property that together, their values at the n points form a basis for the direct sum of the tangent spaces at the n landmark points. The rest of the procedure remains the same.

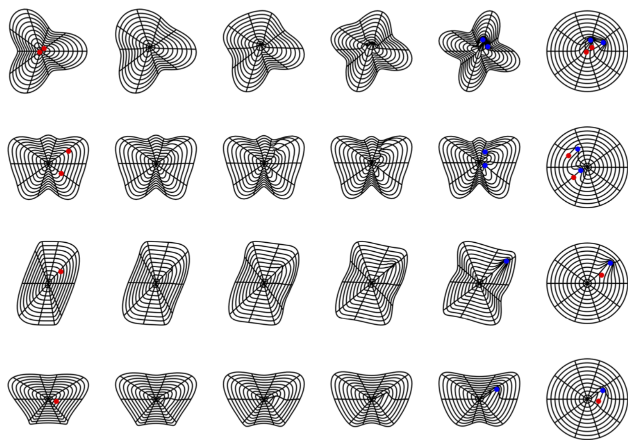

Fig. 18 Examples of geodesic between f_1 and $f_2 \circ \hat{\gamma}$. The last panel: $\hat{\gamma}$ and landmarks. The red points represent the set of landmarks on f_1; the blue points represent the set of landmarks on f_2(Color figure online)

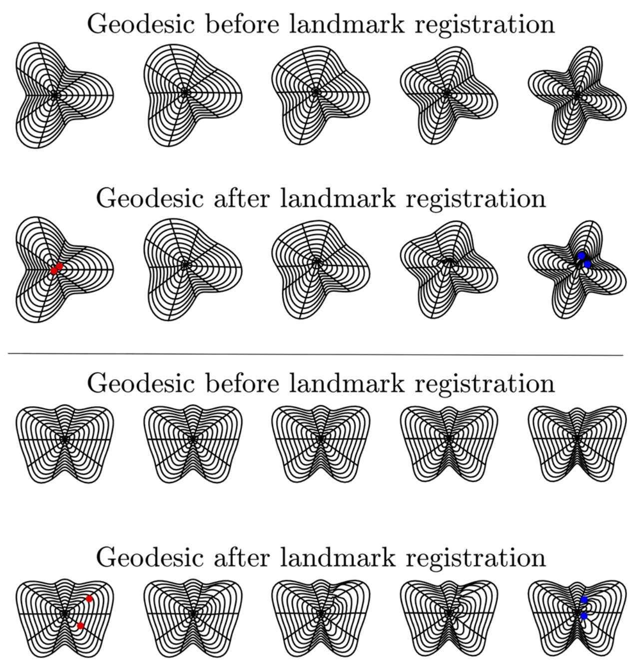

Fig. 19 Examples of geodesic with and without landmark registration

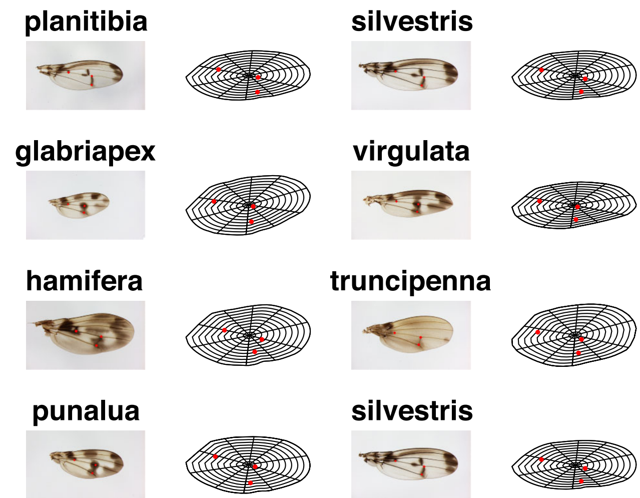

Fig. 20 Dataset: drosophila wing shapes

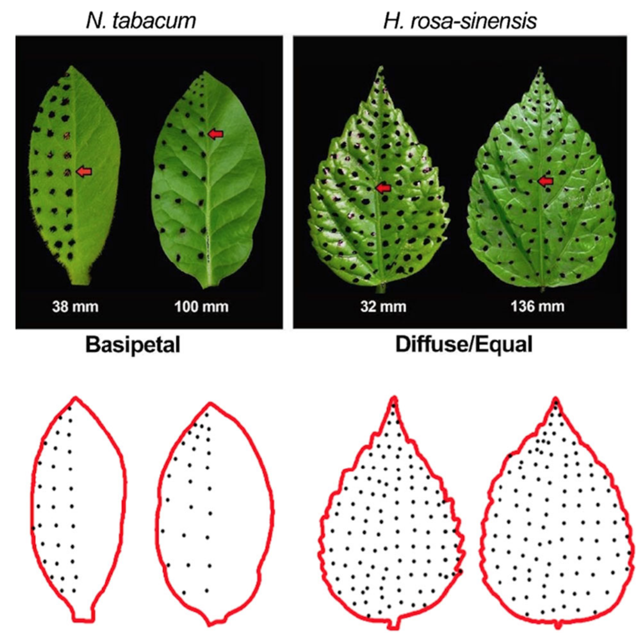

Fig. 21 Dataset: leaf shapes

Once we find the optimal landmark-constrained registration, the geodesic between registered objects (f_1 and $f_2 \circ \gamma$) can be obtained using Algorithm 4. Some examples using simulated objects are shown in Fig. 18. Each row in this figure shows the final geodesic (first five shapes) and the optimal $\hat{\gamma}$ in the last panel. Additionally, we provide some comparisons of geodesics with and without landmark registration in Fig. 19.

5.3 Applications on Real Data

Next, we present applications of landmark-constrained shape analysis on two datasets:

1. Drosophila wing shapes, shown in Fig. 20, are collected in pairs by Edwards et al. [9] for drosophila species living on the Hawaiian island chain, and the landmarks are chosen at the intersections of the veins;
2. Leaf shapes, shown in Fig. 21, are collected by Gupta and Nath [13], and the landmarks are ink spots marked to monitor the leaf's growth.

We start by extracting the boundary and the landmarks from the image and then parameterize the interior and the boundary to provide an initial coordinate system. In drosophila wing shapes, we take four pairs of shapes and compute elastic geodesics between them with registration.

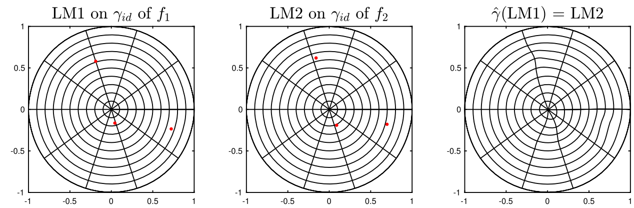

Fig. 22 $\hat{\gamma}$ that registers landmarks on planitibia and silvestris

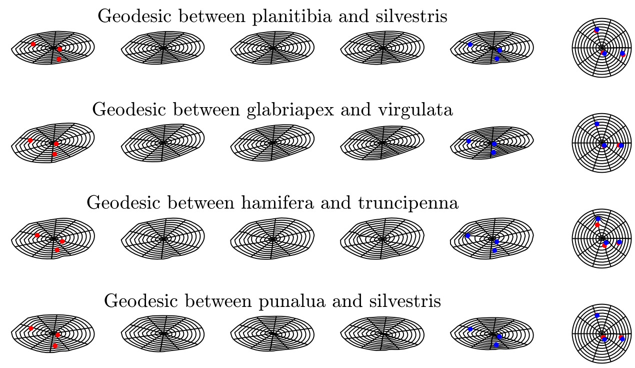

Fig. 23 Geodesic between f_1 and $f_2 \circ \hat{\gamma}$ for wing shapes. The last panel: $\hat{\gamma}$ and landmarks

We use LDDMM to register the landmarks for each pair and provide an initial registration for f_1 and f_2. The overall shapes are very similar in these examples, and additional landmark-constrained registration does not significantly change the result. In practice, for such similar shapes, one can ignore the full re-parametrizations and perhaps manage with only the (LDDMM-based) landmark registration step. An example $\hat{\gamma}$ that registers the landmarks on these shapes is shown in Fig. 22. Specifically, we show the example of registering *Silvestri* to *Planitibia*. The left two panels show individual landmarks and the right panel shows $\hat{\gamma}$. We can see that the deformation is relatively small as the shapes and the landmarks are relatively similar. $\hat{\gamma}$ for the other pairs are similar and hence not displayed. The geodesics for four of shapes (displayed in Fig. 20) are presented in Fig. 23.

In leaf shapes, there are many landmarks, perhaps too many, to make a difference. Therefore, we select a small subset of them to help perform registration. We take the two pairs of shapes shown in Fig. 21 and compute elastic geodesics between them. Figure 24 shows the resulting geodesics for registered leaf shapes.

6 Summary and Discussion

In this paper, we introduce a tensor field representation for analyzing the shapes of planar solid objects. While this representation provides several desired invariances to nuisance transformations, it presents a challenge in inverting the representations back to objects. We use a numerical approach and minimize a particular energy function to perform inversion successfully. This step, combined with dense registration of points across objects, results in elastic geodesics and geodesic distances between shapes of planar objects. The paper develops techniques for the registration of points with or without using discrete landmarks.

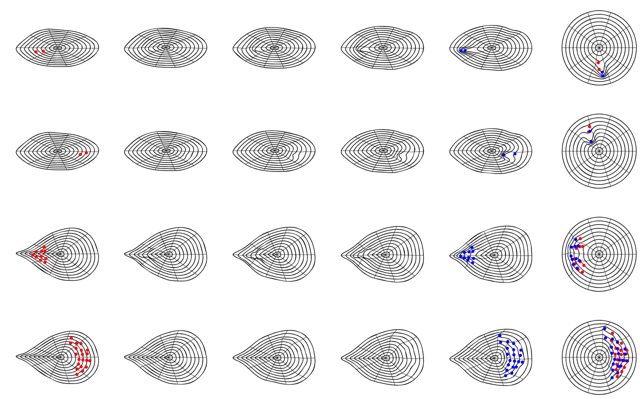

Fig. 24 Geodesic between f_1 and $f_2 \circ \hat{\gamma}$ for leaf shapes. The last panel: $\hat{\gamma}$ and landmarks

The main limitations of the method presented here are the following. These limitations also naturally point to items for future research in this area.

1. As mentioned in the paper, the geometry of the range space of G has not been respected or utilized in the current implementation for geodesics in $\mathcal{G}$. One can explore the use of this geometry and develop more *intrinsic* techniques for such geodesics.
2. Ideally, the objective function used for registration should be the geodesic distance $d_{\mathcal{G}}$ defined on the preshape space. In this paper, we use a simplified surrogate that is computationally efficient to optimize. Given recent advances in optimization techniques on nonlinear manifolds, one can explore the use of $d_{\mathcal{G}}$ for registering points across objects.
3. In the current implementation of optimization over Γ, the chosen basis for $T_{\gamma_{id}}(\Gamma)$ does not allow the movement of the points on the boundary of the domain. This constraint can be easily relaxed, and the method extended.

A crucial part of the shape analysis framework is finding a proper basis for the gradient descent algorithms for registration. For the optimization over Γ, the classic bases such as the Fourier basis or the polynomial basis are not as practical as they do not efficiently capture complex deformations. Thus, we propose a novel method to formulate the basis using landmark registration via large deformation diffeomorphisms [14], and it gives satisfactory results. The whole framework can be easily generalized to any objects defined on D such as $f : D \to \mathbb{R}^{1,2,3}$.

Acknowledgements This research was supported in part by the Grants NSF DMS-1621787, NSF CDS&E DMS 1953087, and NIH R01 GM126558. The authors thank Prof. Eric Klassen of FSU for fruitful discussions on this topic and for providing feedback on this research.

A Appendix: Geometry of $\mathrm{Sym}^+(n)$

For any $A \in \mathrm{Sym}^+(n)$ and $U, V \in T_A\mathrm{Sym}^+(n)$, consider the following Riemannian metric defined on Sym^+(n)

$$\langle U, V\rangle_A = \left(\mathrm{tr}\left(A^{-1}U_0A^{-1}V_0\right) + \kappa\,\mathrm{tr}\left(A^{-1}U\right)\mathrm{tr}\left(A^{-1}V\right)\right)\sqrt{\det A}, \tag{10}$$

where $\kappa > 0$, $U_0 = U - \frac{1}{n}\mathrm{tr}(A^{-1}U)A$ and $V_0 = V - \frac{1}{n}\mathrm{tr}(A^{-1}V)A$ satisfying $\mathrm{tr}(A^{-1}U_0) = 0$ and $\mathrm{tr}(A^{-1}V_0) = 0$. This Riemannian metric was first introduced in [12] and is termed the *split Ebin metric*. The geometric structure of Sym^+(n) endowed with the split Ebin metric was studied thoroughly in [12] and we borrow some results from it directly.

Theorem 3 *Let $A \in \mathrm{Sym}^+(n)$ and $K \in T_A\mathrm{Sym}^+(n)$. Define $q = 1 + \frac{\mathrm{tr}(A^{-1}K)}{4}$, $\theta = \frac{\sqrt{\kappa^{-1}\mathrm{tr}(A^{-1}K_0A^{-1}K_0)}}{4}$, where $K_0 = K - \frac{1}{n}\mathrm{tr}(A^{-1}K)A$ satisfying $\mathrm{tr}(A^{-1}K_0) = 0$. Then, the exponential map starting at A in the direction of K is given by*

$$\mathrm{Exp}_A(K) = \begin{cases} (q^2+\theta^2)^{\frac{2}{n}}\, A \exp\left(\frac{\arctan(\theta/q)}{\theta}A^{-1}K_0\right) & \text{if } K_0 \neq 0,\\ q^{\frac{4}{n}}A & \text{if } K_0 = 0. \end{cases} \tag{11}$$

Theorem 4 *Let $A, B \in \mathrm{Sym}^+(n)$, we have $K = A\log(A^{-1}B) \in \mathcal{U} = T_A\mathrm{Sym}^+(n)\backslash(-\infty, -4/n]A$. Then the inverse of the exponential map is given by the following:*

$$\mathrm{Exp}_A^{-1}(B) = \begin{cases} \frac{4}{n}(\beta\cos\theta - 1)A + \frac{1}{\theta}\beta\sin\theta K_0 & \text{if } K_0 \neq 0,\\ \frac{4}{n}(\beta - 1)A & \text{if } K_0 = 0, \end{cases} \tag{12}$$

where $\beta = \exp\left(\frac{\mathrm{tr}(A^{-1}K)}{4}\right)$, K_0 and θ are define as Theorem 3.

Then, the geodesic on $\mathrm{Sym}^+(n)$ can be solved explicitly.

Theorem 5 *Let $A, B \in \mathrm{Sym}^+(n)$, $u \triangleq \mathrm{Exp}_A^{-1}(B)$, the geodesic between A and B is given as*

$$\phi(t) = \mathrm{Exp}_A(ut), t \in [0, 1]. \tag{13}$$

Let $\mathcal{M}$ be the space of positive semi-definite symmetric matrices. Then the metric completion of $\mathrm{Sym}^+(n)$ is given by $\mathcal{M}/\sim$, where $A \sim B$ if they are both degenerate.

When $\kappa = \frac{1}{n}$, we call the Riemannian metric in Eq. 10 as the standard Ebin metric. It is the usual metric on the space of all Riemannian metrics considered by [8,10,11]. [7] gives the geodesic distance between any two matrices in $\mathrm{Sym}^+(n)$ of the standard Ebin metric as follows.

Theorem 6 *For $A, B \in \mathrm{Sym}^+(n)$. Let $K = A\log(A^{-1}B)$. Then the square of the geodesic distance for the metric (10) between A and B is given by*

$$d(A, B)^2 = \frac{16}{n}\left(\sqrt{\det(A)} - 2\sqrt[4]{\det(A)}\sqrt[4]{\det(B)}\cos\theta + \sqrt{\det(B)}\right), \tag{14}$$

where $\theta = \min\left\{\pi, \frac{\sqrt{n\mathrm{tr}(A^{-1}K_0A^{-1}K_0)}}{4}\right\}$, K_0 is defined as Theorem 3.

In this paper, we use the standard Ebin metric ($\kappa = \frac{1}{2}$) for computation.

Using these point wise geodesics, we can compute a geodesic path between any two elements of $\mathcal{G}$. For any $g_1, g_2 \in \mathcal{G}$, let $\Phi(s, t) = \phi(t)$, where ϕ is a parameterized geodesic between $g_1(s)$ and $g_2(s)$ in $\mathrm{Sym}^+(2)$, for all $s \in D$.

References

1. Bauer, M., Bruveris, M., Marsland, S., Michor, P.W.: Constructing reparameterization invariant metrics on spaces of plane curves. Differ. Geom. Appl. **34**, 139–165 (2014)
2. Bauer, M., Bruveris, M., Michor, P.W.: R-transforms for sobolev h^2-metrics on spaces of plane curves. Geom. Imaging Comput. **1**(1), 1–56 (2014)
3. Bauer, M., Klassen, E., Preston, S.C., Su, Z.: A diffeomorphism-invariant metric on the space of vector-valued one-forms. arXiv preprint arXiv:1812.10867, (2018)
4. Bauer, M., Joshi, S., Modin, K.: Diffeomorphic density registration. In Riemannian Geometric Statistics in Medical Image Analysis, pp 577 – 603. Academic Press, (2020)
5. Beg, M.F., Miller, I., Trouvé, A., Younes, L.: Computing large deformation metric mappings via geodesic flows of diffeomorphisms. Int. J. Comput. Vis. **61**(2), 139–157 (2005)
6. Brigant, A.L.: Computing distances and geodesics between manifold-valued curves in the SRV framework. arXiv:1601.02358, (2016)
7. Clarke, B.: Geodesics, distance, and the cat (0) property for the manifold of riemannian metrics. Math. Z. **273**(1–2), 55–93 (2013)
8. Ebin, D.G.: The manifold of riemannian metrics. In: Global analysis, berkeley, calif., 1968. In Proc. Sympos. Pure Math. **15**, pp 11–40 (1970)
9. Edwards, K.A., Doescher, L.T., Kaneshiro, K.Y., Yamamoto, D.: A database of wing diversity in the hawaiian drosophila. PLoS ONE **2**(5), e487 (2007)

10. Freed, D.S., Groisser, David, et al.: The basic geometry of the manifold of riemannian metrics and of its quotient by the diffeomorphism group. Mich. Math. J. **36**(3), 323–344 (1989)
11. Gil-Medrano, O., Michor, P.W.: The riemannian manifold of all riemannian metrics. Q. J. Math. **42**(1), 183 (1991)
12. Gil-Medrano, O., Michor, P.W., Neuwirther, M.: Pseudo-riemannian metrics on spaces of bilinear structures. Q. J. Math. Oxford **43**, 201–221 (1992)
13. Gupta, M.D., Nath, U.: Divergence in patterns of leaf growth polarity is associated with the expression divergence of mir396. Plant Cell **27**(10), 2785–2799 (2015)
14. Joshi, S.C., Miller, M.I.: Landmark matching via large deformation diffeomorphisms. IEEE Trans. Image Process. **9**(8), 1357–1370 (2000)
15. Klassen, E., Srivastava, A., Mio, M., Joshi, Shantanu H.: Analysis of planar shapes using geodesic paths on shape spaces. IEEE Trans. Pattern Anal. Mach. Intell. **26**(3), 372–383 (2004)
16. Kurtek, S., Srivastava, A., Klassen, E., Laga, H.: Landmark-guided elastic shape analysis of spherically-parameterized surfaces. Comput. Graph. Forum **32**(2), 429–438 (2013)
17. Laga, H., Xie, Q., Jermyn, I.H., Srivastava, A.: Numerical inversion of srnf maps for elastic shape analysis of genus-zero surfaces. IEEE Trans. Pattern Anal. Mach. Intell. **39**(12), 2451–2464 (2017)
18. Michor, P.W., Mumford, D.: Riemannian geometries on spaces of plane curves. J. Eur. Math. Soc. **8**, 1–48 (2006)
19. Mio, W., Srivastava, A., Joshi, S.: On shape of plane elastic curves. Int. J. Comput. Vis. **73**(3), 307–324 (2007)
20. Srivastava, A., Klassen, E., Joshi, S.H., Jermyn, I.H.: Shape analysis of elastic curves in euclidean spaces. IEEE Trans. Pattern Anal. Mach. Intell. **33**(7), 1415–1428 (2011)
21. Su, J., Kurtek, S., Klassen, E., Srivastava, A.: Statistical analysis of trajectories on Riemannian manifolds: bird migration, hurricane tracking and video surveillance. Ann. Appl. Stat. **8**(1), 530–552 (2014). (**03**)
22. Younes, L., Michor, P.W., Shah, J., Mumford, D., Lincei, R.: A metric on shape space with explicit geodesics. Math. E Appl. **19**(1), 25–57 (2008)
23. Younes, L.: Computable elastic distances between shapes. SIAM J. Appl. Math. **58**(2), 565–586 (1998)

Ruiyi Zhang is currently a post-doc working at Center for Imaging Science of Johns Hopkins University. She received her Ph.D. degree in Statistics from Florida State University in 2019. Her research interest includes medical imaging analysis, functional data analysis, shape analysis and nonparametric methods.

Anuj Srivastava is a Professor of Statistics and a Distinguished Research Professor at the Florida State University. He received his Ph.D. degree in Electrical Engineering from Washington University in St. Louis in 1996 and was a visiting research associate at the Division of Applied Mathematics at Brown University during 1996-1997. Since then, he has been working at FSU. His research areas include statistics on nonlinear manifolds, statistical image understanding, functional data analysis, and statistical shape theory. He has published more than 250 papers in refereed journals and conference proceedings. He is a fellow of the American Association for the Advancement of Science (AAAS), Institute of Electrical and Electronics Engineers (IEEE), American Statistical Association (ASA), and International Association for Pattern Recognition (IAPR). He has been an associate editor for several journals in computer vision and statistics.