

Rate-Invariant Analysis of Covariance Trajectories

Zhengwu Zhang · Jingyong Su · Eric Klassen · Huiling Le · Anuj Srivastava

Received: date / Accepted: date

Abstract Statistical analysis of dynamic systems, such as videos and dynamic functional connectivity, is often translated into a problem of analyzing trajectories of relevant features, particularly covariance matrices. As an example, in video-based action recognition, a natural mathematical representation of activity videos is as parameterized trajectories on the set of symmetric, positive-definite matrices (SPDMs). The variable execution-rates of actions, implying arbitrary parameterizations of trajectories, complicates their analysis and classification. To handle this challenge, we represent covariance trajectories using *transported square-root vector fields* (TSRVFs), constructed by parallel trans-

lating scaled-velocity vectors of trajectories to their starting points. The space of such representations forms a vector bundle on the SPDM manifold. Using a natural Riemannian metric on this vector bundle, we approximate geodesic paths and geodesic distances between trajectories in the quotient space of this vector bundle. This metric is invariant to the action of the reparameterization group, and leads to a rate-invariant analysis of trajectories. In the process, we remove the parameterization variability and temporally register trajectories during analysis. We demonstrate this framework in multiple contexts, using both generative statistical models and discriminative data analysis. The latter is illustrated using several applications involving video-based action recognition and dynamic functional connectivity analysis.

Keywords SPDM Riemannian Structure · SPDM Parallel Transport · Invariant Metrics · Covariance Trajectories · Trajectories on manifolds · Vector bundles · Rate-invariant classification

Z. Zhang
Department of Biostatistics and Computational Biology, NY, USA
E-mail: zhengwu_zhang@urmc.rochester.edu

J. Su
Department of Mathematics and Statistics, Texas Tech University, Lubbock, TX, USA.
E-mail: jingyong.su@ttu.edu

E. Klassen
Department of Mathematics, Florida State University, Tallahassee, FL, USA.
E-mail: klassen@math.fsu.edu

H. Le
School of Mathematics Sciences, University of Nottingham, Nottingham, UK
E-mail: huiling.le@nottingham.ac.uk

A. Srivastava
Department of Statistics, Florida State University, Tallahassee, FL, USA.
E-mail: anuj@stat.fsu.edu

1 Introduction

The problem of studying of dynamical systems using image sequences (such as videos) is both important and challenging. It has applications in many areas including video surveillance, lip reading, pedestrian tracking, hand-gesture recognition, human-machine interfaces, brain functional connectivity analysis and medical diagnosis. Since the size of video data is generally very high, the task of video classification is often performed by extracting certain low-dimensional features of interest – geometric, motion, colorimetric features, etc – from each frame and then forming temporal sequences of these features for full videos. Consequently, analysis of

videos get replaced by analysis of longitudinal observations in a certain feature space. (Some papers (e.g. [13, 40]) discard temporal structure by pooling all the feature together but that represents a severe loss of information.) Since many features are naturally constrained to lie on nonlinear manifolds, the corresponding video representations form parameterized trajectories on these manifolds. Examples of these manifolds include unit spheres, Grassmann manifolds [16], Lie groups [7], and the space of probability distributions.

One of the most common and effective features in image analysis is a covariance matrix, as shown via applications in medical imaging [3, 31] and computer vision [15, 17, 23, 24, 39, 40]. These matrices are naturally constrained to be symmetric positive-definite matrices (SPDMs) and have also played a prominent role as region descriptors in texture classification, object detection, object tracking, action recognition and face recognition. Tuzel et al. [40] introduced the concept of *covariance tracking* where they extracted a covariance matrix for each video frame and studied the temporal evolution of this matrix in the context of pedestrian tracking in videos. Since the set of SPDMs is a well known set, denoted by $\tilde{\mathcal{P}}$ (or $\tilde{\mathcal{P}}(n)$ when the dimension of the SPDM manifold is specified as n), a video segment can be represented as a (parameterized) trajectory in $\tilde{\mathcal{P}}$. In the brain functional connectivity analysis, the instantaneous connectivity, extracted from functional magnetic resonance imaging (fMRI) data, is typically represented as a SPDM [8, 9]. Therefore, a dynamic evolution of connectivity can be naturally represented as a trajectory on the set of SPDMs. In this paper we focus on the problem of statistical analysis of actions and functional brain connectivity by treating them as parameterized trajectories in $\tilde{\mathcal{P}}$. Fig. 1 shows some examples of video frames for the two applications studied in this paper: visual-speech recognition and hand-gesture.

One challenge in characterizing activities as trajectories comes from the variability in execution rates. The execution rate of an activity dictates the parameterization of the corresponding trajectory. The execution rates for different observations are quite different, even if the activities belong to the same class. Different execution rate implies that the trajectories go through the same sequences of points in $\tilde{\mathcal{P}}$ but at different times. Consequently, directly analyzing such trajectories without temporal alignment, e.g. comparing the difference, and calculating point-wise mean and covariance, can be erroneous. This is because the mean may not be a representative of individual trajectories, and the variance is often artificially inflated.

To make these issues precise, we develop some notation first. Let $\alpha : [0, 1] \rightarrow \tilde{\mathcal{P}}$ be a trajectory and let

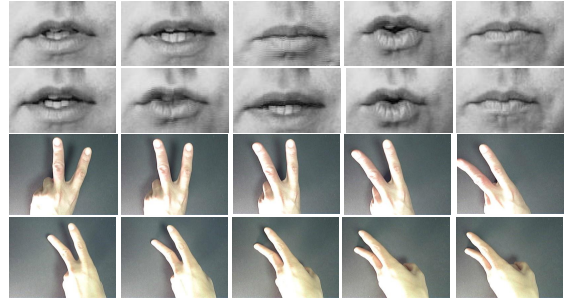


Fig. 1 Examples of video frames in visual-speech recognition (first two rows) and hand-gesture classification (last two).

$\gamma : [0, 1] \rightarrow [0, 1]$ be a positive diffeomorphism such that $\gamma(0) = 0$ and $\gamma(1) = 1$. This γ plays the role of a time-warping function, or a re-parameterization function, so that the composition $\alpha \circ \gamma$ is now a time-warped or re-parameterized version of α . In other words, the trajectory $\alpha \circ \gamma$ goes through the same set of points as α but at a different rate (speed). Some kind of temporal registration is necessary to deal with this so-called *phase variability*.

There are two types of registration problems for trajectories. Firstly, the pairwise registration defined as follows. Let $\alpha_1, \alpha_2 : [0, 1] \rightarrow \tilde{\mathcal{P}}$ be two trajectories in $\tilde{\mathcal{P}}$. The process of registration of α_1 and α_2 is to find a time warping γ such that $\alpha_1(t)$ is optimally registered to $\alpha_2(\gamma(t))$ for all $t \in [0, 1]$, with optimality defined using an objective function. Another type is multiple registration: let $\alpha_1, \alpha_2, \dots, \alpha_n$ be n trajectories on $\tilde{\mathcal{P}}$, and we want to find out time warpings $\gamma_1, \gamma_2, \dots, \gamma_n$ such that for all t , the variables $\{\alpha_i(\gamma_i(t))\}_{i=1}^n$ are optimally registered. A solution for pairwise registration can be used to solve the multiple registration problem using an iterative solution – for the given trajectories, first define a *template* trajectory and then align each given trajectory to this template in a pairwise fashion. One way of defining this template is to use the mean of given trajectories under an appropriately chosen metric.

Notice that the problem of *comparisons of trajectories* is different from the problem of *curve fitting* or *trajectory estimation* from noisy data. Many papers have studied spline-type solutions for fitting curves to discrete, noisy data points on manifolds [20, 25, 30, 33, 37] but in this paper we assume that the trajectories are already available through some other means.

1.1 Past Work & Their Limitations

There are very few papers in the literature for analyzing – in the sense of comparing, averaging or clustering

– trajectories on nonlinear manifolds. What one may consider a very natural approach actually has limitations when seeking parametrization invariance. Let $d_{\tilde{\mathcal{P}}}$ denote the geodesic distance resulting from the chosen Riemannian metric on $\tilde{\mathcal{P}}$. It can be shown that the quantity $\int_0^1 d_{\tilde{\mathcal{P}}}(\alpha_1(t), \alpha_2(t)) dt$ forms a proper distance on the set $\tilde{\mathcal{P}}^{[0,1]}$, the space of all trajectories on $\tilde{\mathcal{P}}$. For example, [22] uses this metric, combined with the arc-length distance on \mathbb{S}^2 , to cluster hurricane tracks. However, this metric is not immune to different temporal evolutions of hurricane tracks. Handling this variability requires performing some kind of temporal alignment. It may be tempting to use the following modification of this distance to align two trajectories:

$$\inf_{\gamma \in \Gamma} \left(\int_0^1 d_{\tilde{\mathcal{P}}}(\alpha_1(t), \alpha_2(\gamma(t))) dt \right), \quad (1)$$

but this can lead to degenerate solutions (also known as the *pinching problem*, described for real-valued functions in [32, 35]). Pinching implies that a severely distorted γ is used to eliminate (or minimize) those parts of α_2 that do not match with α_1 , which can be done even when α_2 is significantly different from α_1 . While this degeneracy can be avoided using a regularization penalty on γ , some of the other problems remain, including the fact that the solution is not symmetric.

A recent solution, presented in [38, 39], develops the concept of elastic trajectories to deal with the parametrization variability. It represents each trajectory by its transported square-root vector field (TSRVF) defined as:

$$h_{\alpha}(t) = \left(\frac{\dot{\alpha}(t)}{\sqrt{|\dot{\alpha}(t)|}} \right)_{\alpha(t) \rightarrow c} \in \mathcal{T}_c(\tilde{\mathcal{P}}),$$

where c is pre-determined but arbitrary reference point on $\tilde{\mathcal{P}}$ and \rightarrow denotes a parallel transport of the vector $\dot{\alpha}(t)$ from the point $\alpha(t)$ to c along a geodesic path. A trajectory is mapped into a curve in the tangent space $\mathcal{T}_c(\tilde{\mathcal{P}})$ and one can compare/align these curves using the \mathbb{L}^2 norm, denoted by $\|\cdot\|$, on that vector space. More precisely, the quantity $\inf_{\gamma} \|h_{\alpha_1} - h_{\alpha_2 \circ \gamma}\|$ provides not only a criterion for optimality of γ but also a proper metric for averaging and other statistical analyses. This TSRVF representation is an extension of the SRVF framework used for elastic shape analysis of curves in Euclidean spaces [36]. There are two main limitations of this mathematical representation. One is that the choice of reference point, c , is left arbitrary. The results can potentially change with c , which make it difficult to interpret the results. A bigger issue is that the transport of tangent vectors $\dot{\alpha}(t)$ to c ,

along geodesics, can introduce large distortion, especially when the point $\alpha(t)$ is far from c on the manifold.

Since our original formulation [41], Briggant et al. [5, 6] have also used a similar Riemannian structure for comparing trajectories. However, their representations are based on a direct analysis of the vector fields $\frac{\dot{\alpha}(t)}{\sqrt{|\dot{\alpha}(t)|}}$, i.e. without any parallel translation, and the space of such representations is the the space of trajectories in full tangent bundle of the manifold. As described next, the proposed representation in our paper is a curve in a tangent space and, thus, the space of representations is a vector bundle, a proper subset of the tangent bundle used in [5, 6]. Consequently, the resulting Riemannian metric and geodesic paths are different in the two sets of works. A major limitation of [5, 6] is that while they use parametrization-invariant metrics, they do not explicitly solve for the temporal registration across trajectories. This registration is, in fact, the main reason for choosing invariant metrics in the first place, and is a major contribution of the current paper.

1.2 Our Approach

We introduce a novel mathematical representation of trajectories that does not require a choice of c . In this representation, the trajectories are still represented by their transported vector fields but not at the global reference point. For each trajectory α_i , the reference point is chosen to be its starting point $\alpha_i(0)$, and the *transport is performed along the trajectory itself*. In other words, for each t , the velocity vector $\dot{\alpha}_i(t)$ is transported along α to the tangent space of the starting point $\alpha_i(0)$. As a consequence, the trajectory α gets mapped into a curve in the tangent space $T_{\alpha(0)}(\tilde{\mathcal{P}})$. This idea has been used previously in [25] and others, for some shape manifolds, and results in a relatively stable curve with smaller distortions than the TSRVFs of [38]. However, these previous papers do not provide re-parameterization invariance in their analysis. In contrast, we develop a metric-based framework for comparing, averaging, and modeling such curves in a manner that is invariant to their re-parameterizations. Consequently, this framework provides a natural solution for removal of rate, or *phase*, variability from trajectory data.

The main contributions of this paper are:

1. Provides a novel representation for parametrization-invariant analysis of trajectories on manifolds. It results in a significant improvement over [38, 39] in the sense that the new representation forms a vector bundle of the manifold, rather than a pre-determined tangent space.

2. Introduces a re-parameterization invariant metric on the vector bundle and uses that metric to generate temporal alignments, and rate-invariant sample summary of trajectories on manifolds.
3. Provides efficient algorithms for computation of geodesic parallel-transport of the vector field along α to the starting point p according to: for each $\tau \in [0, 1]$, $q(\tau) = (\frac{\dot{\alpha}(\tau)}{\sqrt{|\dot{\alpha}(\tau)|}})_{\alpha(\tau) \rightarrow p} \in \mathcal{T}_p(M)$, where $|\cdot|$ denotes the norm that is defined through the Riemannian metric on M .
4. Demonstrates these ideas by successfully analyzing covariance trajectories, using data from video-based action recognition and dynamic brain functional connectivity analysis. In the process, it utilizes a non-standard metric on SPDMS that provides all necessary tools in $\tilde{\mathcal{P}}$ for this framework.

The rest of this paper is organized as follows. In Section 2, we introduce our framework for aligning, averaging and comparing of trajectories on a general manifold M . Since we mainly focus on covariance trajectories as an application, in Section 4, we introduce a Riemannian structure on the set $\tilde{\mathcal{P}}$ of SPDMS. Sections 5 and 7 provide some illustrations of the proposed framework seeking statistical summaries and generative modeling of 3×3 SPDM trajectories. In Section 8, we demonstrate the proposed work with real-world data involving video-based action recognition and dynamic functional brain network analysis.

2 Analysis of Trajectories on Manifolds

In this section we derive a framework for comparing trajectories on a general Riemannian manifold M .

2.1 Representation of Trajectories

Let α denote a piecewise C^1 trajectory on a Riemannian manifold M . That is, $\alpha : [0, 1] \rightarrow M$ such that there are finitely many points $0 = t_0 < t_1 < \dots < t_n = 1$ such that on each $[t_{i-1}, t_i]$ α is a C^1 curve (one-sided derivatives at each end). Let \mathcal{F}_p be all such piecewise C^1 trajectories starting at p , and let $\mathcal{F} = \coprod_{p \in M} \mathcal{F}_p$.

Define Γ to be the set of all orientation preserving diffeomorphisms of $[0, 1]$: $\Gamma = \{\gamma : [0, 1] \rightarrow [0, 1] | \gamma(0) = 0, \gamma(1) = 1, \gamma \text{ is a diffeomorphism}\}$. Γ forms a group under the composition operation. If α is a trajectory on M , then $\alpha \circ \gamma$ is a trajectory that follows the same sequence of points as α but at the evolution rate governed by γ . More technically, the group Γ acts on \mathcal{F} , $\mathcal{F} \times \Gamma \rightarrow \mathcal{F}$, according to $(\alpha, \gamma) = \alpha \circ \gamma$.

We introduce a new representation of trajectories that will be used to compare and register them. We assume that for any two points $\alpha(\tau_1), \alpha(\tau_2) \in M, \tau_1 \neq \tau_2$, we have a mechanism for parallel transporting any vector $v \in \mathcal{T}_{\alpha(\tau_1)}(M)$ **along** α from $\alpha(\tau_1)$ to $\alpha(\tau_2)$, denoted by $(v)_{\alpha(\tau_1) \rightarrow \alpha(\tau_2)}$.

Definition 1 Let $\alpha : [0, 1] \rightarrow M$ denote a piecewise C^1 trajectory starting with $p = \alpha(0)$. Given a trajectory α , and the velocity vector field $\dot{\alpha}$, define its transported square-root vector field (TSRVF) to be a scaled parallel-transport of the vector field along α to the starting point p according to: for each $\tau \in [0, 1]$, $q(\tau) = (\frac{\dot{\alpha}(\tau)}{\sqrt{|\dot{\alpha}(\tau)|}})_{\alpha(\tau) \rightarrow p} \in \mathcal{T}_p(M)$, where $|\cdot|$ denotes the norm that is defined through the Riemannian metric on M .

This representation is motivated from some similar but distinct ideas used in the past literature. Firstly, it relates to the notion of *unrolling* introduced by Jupp and Kent [20] for spherical manifolds. Starting with a piecewise C^1 curve α on a sphere, they constructed a curve in \mathbb{R}^2 , called the *unrolling* of α as follows. They define the unrolled curve as the integral of the curve in $T_p(M)$ generated by the parallel translation of $\dot{\alpha}(t)$ along α to p . That is, $\nu : [0, 1] \rightarrow T_p(M)$ is the unrolling of α , where $\nu(t) = \int_0^t (\dot{\alpha}(s)_{\alpha(s) \rightarrow p}) ds$. The difference between unrolling and TSRVF is the use of $\sqrt{|\dot{\alpha}(t)|}$ in the denominator of TSRVF and the extra integral present in unrolling. Secondly, it is similar to the TSRVF in [38] with the difference that in [38] the transport was along **geodesics** to a reference point c , but here the parallel transport is along α (to the starting point p). This reduces distortion in representation relative to the parallel transport of [38] to a faraway reference point.

This TSRVF representation maps a trajectory α on M to a curve q in $\mathcal{T}_p(M)$. For any point $p \in M$, let \mathbb{B}_p be the set of functions on the tangent space $T_p(M)$ of the type: $v : [0, 1] \rightarrow T_p(M)$ is in \mathbb{B}_p if there are finitely many points $0 = t_0 < t_1 < \dots < t_n = 1$ such that, on each $[t_{i-1}, t_i]$, v is continuous, and $\lim_{t \rightarrow t_i^-}$ exists. The space of interest, then, becomes an infinite-dimensional vector bundle $\mathbb{B} = \coprod_{p \in M} \mathbb{B}_p$, which is the indexed union of \mathbb{B}_p for every $p \in M$. We note in passing that \mathbb{B}_p is a subspace of the Hilbert space $\mathbb{L}^2([0, 1], \mathcal{T}_p(M))$, the set of all square-integrable curves in $\mathcal{T}_p(M)$.

There is a bijection between \mathcal{F}_p and \mathbb{B}_p . This result is straightforward except for the following point. If $\alpha \in \mathcal{F}_p$ has a bend at t_0 , then $\dot{\alpha}(t_0)$ does not exist. To define the corresponding TSRVF q , we can take $q(t_0) = \lim_{t \rightarrow t_0^+} (\dot{\alpha}(t) / \sqrt{|\dot{\alpha}(t)|})_{\alpha \rightarrow p}$, and the resulting $q \in \mathbb{B}_p$. As a corollary to this result, the TSRVF representation is bijective: any $\alpha \in \mathcal{F}$ is uniquely represented by a pair $(p, q(\cdot)) \in \mathbb{B}$, where $p \in M$ is the starting point, $q \in \mathbb{B}_p$ is its TSRVF. We can reconstruct the trajectory from (p, q) using the covariant integral (see Algorithm 3 for a numerical implementation).

2.2 Riemannian Structure on \mathbb{B}

In order to compare trajectories, we will compare their corresponding representations in \mathbb{B} and that requires a Riemannian structure on \mathbb{B} . Let α_1, α_2 be two trajectories on M , with starting points p_1 and p_2 , respectively, and let the corresponding TSRVFs be q_1 and q_2 . Now α_1, α_2 are represented as two points in the vector bundle $(p_1, q_1), (p_2, q_2) \in \mathbb{B}$ over M . This representation space is an infinite-dimensional vector bundle, whose fiber over each point p in M is \mathbb{B}_p .

We impose the following Riemannian structure on \mathbb{B} . For an element $(x, v(\cdot))$ in \mathbb{B} , where $x \in M$, $v \in \mathbb{B}_x$, we naturally identify the tangent space at (x, v) to be: $\mathcal{T}_{(x,v)}(\mathbb{B}) \cong \mathcal{T}_x(M) \oplus \mathbb{B}_x$. To see this, suppose we have a curve in \mathbb{B} given by $(x(s), v(s, \tau))$, $s, \tau \in [0, 1]$. The velocity vector to this curve at $s = 0$ is given by $(x_s(0), \nabla_{x_s} v(0, \cdot)) \in \mathcal{T}_x(M) \oplus \mathbb{B}_x$, where x_s denotes dx/ds , and ∇_{x_s} denotes *covariant differentiation of tangent vectors*. The Riemannian inner product on \mathbb{B} is defined in an obvious way: If $(u_1, w_1(\cdot))$ and $(u_2, w_2(\cdot))$ are both elements of $\mathcal{T}_{(x,v)}(\mathbb{B}) \cong \mathcal{T}_x(M) \oplus \mathbb{B}_x$, define

$$\langle (u_1, w_1(\cdot)), (u_2, w_2(\cdot)) \rangle = (u_1 \cdot u_2) + \int_0^1 (w_1(\tau) \cdot w_2(\tau)) d\tau, \quad (2)$$

where the inner products on the right denote the original Riemannian metric in $\mathcal{T}_x(M)$.

For given two points (p_1, q_1) and (p_2, q_2) on \mathbb{B} , we want to find the geodesic path connecting them. Let $(x(s), v(s, \cdot))$, $s \in [0, 1]$ be a path with $(x(0), v(0, \cdot)) = (p_1, q_1)$ and $(x(1), v(1, \cdot)) = (p_2, q_2)$. We have the following characterization of geodesics on \mathbb{B} .

Theorem 1 *A parameterized path $[0, 1] \rightarrow \mathbb{B}$ given by $s \mapsto (x(s), v(s, \tau))$ on \mathbb{B} (where the variable τ corresponds to the parametrization in \mathbb{B}_x), is a geodesic in \mathbb{B} if and only if:*

$$\begin{aligned} \nabla_{x_s} x_s + \int_0^1 R(v, \nabla_{x_s} v)(x_s) d\tau &= 0 & \text{for every } s, \\ \nabla_{x_s} (\nabla_{x_s} v)(s, \tau) &= 0 & \text{for every } s, \tau. \end{aligned} \quad (3)$$

Here $R(\cdot, \cdot)(\cdot)$ denotes the Riemannian curvature tensor, x_s denotes dx/ds , and ∇_{x_s} denotes the covariant differentiation of tangent vectors on tangent space $\mathcal{T}_{x(s)}(M)$.

Proof: We will prove this theorem in two steps.

(1) First, we consider a simpler case where the space of interest is the tangent bundle TM of the Riemannian manifold M . An element of TM is denoted by (x, v) , where $x \in M$ and $v \in \mathcal{T}_x(M)$. It is natural to identify $\mathcal{T}_{(x,v)}(TM) \cong \mathcal{T}_x(M) \oplus \mathcal{T}_x(M)$. The Riemannian inner

product on TM is defined in the obvious way: If (u_1, w_1) and (u_2, w_2) are both elements of $\mathcal{T}_{(x,v)}(TM)$, define

$$\langle (u_1, w_1), (u_2, w_2) \rangle = u_1 \cdot u_2 + w_1 \cdot w_2$$

and, again, the inner products on the right denote the original Riemannian metric on M . Suppose we have a path in $[0, 1] \rightarrow TM$ given by $s \mapsto (x(s), v(s))$. We define the energy of this path by

$$E = \int_0^1 (x_s \cdot x_s + \nabla_{x_s} v \cdot \nabla_{x_s} v) ds.$$

The integrand is the inner product of the velocity vector of the path with itself. It is a standard result that a geodesic on TM can be characterized as a path that is a critical point of this energy function on the set of all paths between two fixed points in TM . To derive local equations for this geodesic, we now assume we have a parameterized family of paths denoted by $(x(s, t), v(s, t))$, where s is the parameter of each individual path in the family (as above) and the variable t tells us which path in the family we are in. Assume $0 \leq s \leq 1$ and t takes values on $(-\delta, \delta)$ for some small $\delta > 0$. We want all the paths in this family to start and end at the same points of TM , so assume that $(x(0, t), v(0, t))$ and $(x(1, t), v(1, t))$ are constant functions of t . The energy of the path with index t is given by:

$$E(t) = \int_0^1 (x_s \cdot x_s + \nabla_{x_s} v \cdot \nabla_{x_s} v) ds.$$

To simplify notation in what follows, we will write ∇_s for ∇_{x_s} and ∇_t for ∇_{x_t} . To establish conditions for (x, v) to be critical, we take the derivative of $E(t)$ with respect to t at $t = 0$:

$$E'(0) = 2 \int_0^1 [(\nabla_t x_s \cdot x_s) + (\nabla_t (\nabla_s v) \cdot \nabla_s v)] ds.$$

We will use two elementary facts: (a) $\nabla_t(x_s) = \nabla_s(x_t)$ and (b) $R(x_t, x_s)(v) = \nabla_t(\nabla_s v) - \nabla_s(\nabla_t v)$, without presenting their proofs. Plugging these facts into the above calculation, we get $E'(0)$ to be:

$$\begin{aligned} & 2 \int_0^1 [\nabla_s x_t \cdot x_s + R(x_t, x_s)(v) \cdot \nabla_s v + \nabla_s(\nabla_t v) \cdot \nabla_s v] ds \\ &= 2 \int_0^1 [(-\nabla_s x_s \cdot x_t) + R(x_t, x_s)(v) \cdot \nabla_s v + (-\nabla_s(\nabla_s v) \cdot \nabla_t v)] ds. \end{aligned}$$

The second equality comes from using integration by parts on the first and third term, taking into account the fact that x_t and $\nabla_t v$ vanish at $s = 0, 1$, (since all the paths begin and end at the same point). Now, using the

standard identities $R(X, Y)(Z) \cdot W = R(Z, W)(X) \cdot Y$ and $R(X, Y)(Z) \cdot W = -R(X, Y)(W) \cdot Z$, we obtain:

$$\begin{aligned} E'(0) &= 2 \int_0^1 [(-\nabla_s x_s \cdot x_t) + (-R(v, \nabla_s v)(x_s) \cdot x_t) \\ &\quad + (-\nabla_s(\nabla_s v) \cdot \nabla_t v)] ds \\ &= -2 \int_0^1 [(\nabla_s x_s + R(v, \nabla_s v)(x_s)) \cdot x_t + (\nabla_s(\nabla_s v) \cdot \nabla_t v)] ds \\ &= -2 \int_0^1 (\nabla_s x_s + R(v, \nabla_s v)(x_s)) \cdot x_t ds \\ &\quad - 2 \int_0^1 \nabla_s(\nabla_s v) \cdot \nabla_t v ds . \end{aligned}$$

Now, $(x(s), v(s))$ is critical for E if and only if $E'(0) = 0$ for every possible variation x_t of x and $\nabla_t(v)$ of v , which is clearly true if and only if

$$\nabla_s x_s + R(v, \nabla_s v)(x_s) = 0 \text{ and } \nabla_s(\nabla_s v) = 0.$$

Thus we have derived the geodesic equations for TM .

(2) Now we consider the case of the infinite dimensional vector bundle $\mathbb{B} \rightarrow M$ whose fiber over $x \in M$ is $\mathbb{L}^2(I, \mathcal{T}_x(M))$, $I = [0, 1]$. A point in \mathbb{B} is denoted by $(x, v(\tau))$, where the variable τ corresponds to the I -parameter in $\mathbb{L}^2(I, \mathcal{T}_x(M))$. The tangent space to \mathbb{B} at $(x, v(\tau))$ is $\mathcal{T}_x(M) \oplus \mathbb{L}^2(I, \mathcal{T}_x(M))$. Suppose $(u_1, w_1(\tau))$ and $(u_2, w_2(\tau))$ are elements of this tangent space and we use the Riemannian metric:

$$\langle (u_1, w_1(\tau)), (u_2, w_2(\tau)) \rangle = u_1 \cdot u_2 + \int_0^1 w_1(\tau) \cdot w_2(\tau) d\tau.$$

Now we want to work out the local equations for geodesics in \mathbb{B} . A path in \mathbb{B} is denoted by $(x(s), v(s, \tau))$. The energy calculation is basically the same as above but surround everything with integration with respect to τ . So, it starts out with

$$\begin{aligned} E &= \int_0^1 \left(x_s \cdot x_s + \int_0^1 \nabla_s v \cdot \nabla_s v d\tau \right) ds \\ &= \int_0^1 \int_0^1 (x_s \cdot x_s + \nabla_s v \cdot \nabla_s v) ds d\tau. \end{aligned}$$

(Of course $x_s \cdot x_s$ does not involve the parameter τ , but surrounding it with $\int_0^1 \dots d\tau$ does not change its value!)

In order to perform variational calculus, we now consider a parametrized family of such paths, denoted by $(x(s, t), v(s, t, \tau))$ where we assume that $x(0, t)$ and $x(1, t)$ are constant functions of t , and for each τ , $v(0, t, \tau)$ and $v(1, t, \tau)$ are constant functions of t , since we want every path in our family to start and end at the same points of \mathbb{B} .

Then, following through the computation exactly as in earlier case, we obtain

$$\begin{aligned} E'(0) &= -2 \int_0^1 \left(\nabla_s x_s + \int_0^1 R(v, \nabla_s v)(x_s) d\tau \right) \cdot x_t ds \\ &\quad - 2 \int_0^1 \int_0^1 \nabla_s(\nabla_s v) \cdot \nabla_t v d\tau ds. \end{aligned}$$

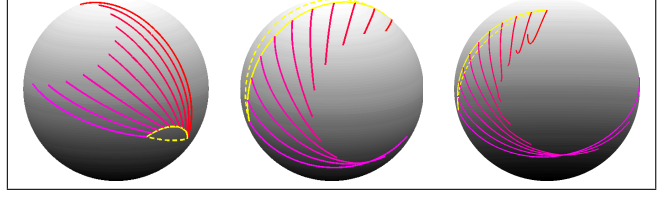


Fig. 2 Examples of geodesic between two trajectories on \mathbb{S}^2 . The yellow solid line denotes the baseline $x(s)$ and yellow dash line shows the geodesic on \mathbb{S}^2 as a comparison.

In order for our path $(x(s), v(s, \tau))$ to be critical for E , $E'(0)$ must vanish for every variation $x_t(s)$ of $x(s)$ and $\nabla_t(v(s, \tau))$ of $v(s, \tau)$, which is clearly true if and only if

$$\begin{aligned} \nabla_s x_s + \int_0^1 R(v, \nabla_s v)(x_s) d\tau &= 0, \text{ for every } s \\ \nabla_s(\nabla_s v) &= 0, \text{ for every } s \text{ and every } \tau . \end{aligned}$$

Q.E.D

The geodesic path $(x(s), v(s, \tau))$ can be intuitively understood as follows: (1) $x(s)$ is a *baseline* curve on M connecting p_1 and p_2 , and the covariant differentiation of x_s at the tangent space of $\mathcal{T}_{x(s)}(M)$ equals the negative integral of the Riemannian curvature tensor $R(v(s, \tau), \nabla_{x_s} v(s, \tau))(x_s)$ with respect to τ . In other words, the values of v at each τ equally determine the geodesic acceleration of $x(s)$ in the first equation. (2) The second equation leads to a fact that v is covariantly linear, i.e. $v(s, \tau) = a(s, \tau) + sb(s, \tau)$ and $\nabla_{x_s} a = \nabla_{x_s} b = 0$ for every s and τ . For a geodesic path connecting (p_1, q_1) and (p_2, q_2) , it is natural to let $a(s, \tau) = q_1(\tau)_{x(0) \rightarrow x(s)}$ and $b(s, \tau) = w(\tau)_{x(0) \rightarrow x(s)}$, where $q_1(\tau)_{x(0) \rightarrow x(s)}$ and $w(\tau)_{x(0) \rightarrow x(s)}$ represent the parallel transport of $q_1(\tau)$ and $w(\tau)$ along x from $x(0)$ to $x(s)$, and w is the difference between the TSRVFs q_2 and q_1 in $\mathcal{T}_{x(0)}(M)$, defined as $(q_2)_{x(1) \rightarrow x(0)} - q_1$. In Fig. 2, we illustrate geodesic paths between some arbitrary trajectories on $M = \mathbb{S}^2$. In each case, the yellow solid line denotes the baseline $x(s)$ and the intermediate lines are the covariant integrals (in Algorithm 3) of $v(s, \cdot)$ with starting point $x(s)$. As comparison, the dash yellow line shows the standard geodesic curve between starting points p_1 and p_2 in \mathbb{S}^2 .

Theorem 1 is only a characterization of geodesics but does not provide explicit expressions for computing them. In the following section, we develop a numerical solution for constructing geodesics in \mathbb{B} .

2.3 Numerical Computations of Geodesics in \mathbb{B}

Here we develop a numerical approach for computing geodesic paths in the representation space. To simplify

discussion, we will assume that the original trajectories on M are not only piecewise C^1 but also piecewise geodesic. This implies that the corresponding TSRVFs are piecewise constant. (This restriction was also discussed for unrolling of spherical curves in [20].) Therefore, our focus in this section will be on piecewise constant TSRVFs.

There are two main approaches in numerical construction of geodesic paths on manifolds. The first approach, called *path-straightening*, initializes the search with an arbitrary path, between the given two points on the manifold, and then iteratively “straightens” it until a geodesic is reached. The second approach, called the *shooting method*, tries to “shoot” a geodesic from the first point, iteratively adjusting the shooting direction, so that the resulting geodesic passes through the second point. In this paper, we use the shooting method to construct geodesic paths in \mathbb{B} .

In order to implement the shooting method, we need the exponential map on \mathbb{B} . Given a point $(p, q) \in \mathbb{B}$ and a tangent vector $(u, w) \in \mathcal{T}_{(p,q)}(\mathbb{B})$, the exponential map $\exp_{(p,q)}(s(u, w))$ for $s \in [0, 1]$ gives a geodesic path $(x(s), v(s))$ in \mathbb{B} (for notational simplicity, we will use $(x(s), v(s))$ to denote $(x(s), v(s, \cdot))$). Equation 3 helps us with this construction as follows. The two equations essentially provide expressions for second-order covariant derivatives of x and v components of the path. Therefore, using numerical techniques, we can perform covariant integration of these quantities to recover the path itself.

Note that we assume that v and w are piecewise constant over the same partition of $[0, 1]$. Furthermore, using the re-parameterization group introduced later, we can also assume that this partition is a uniform partition of $[0, 1]$. Note that addition and subtraction of piecewise constant functions with identical partitions simplify to these operations restricted to only the mid-points of the intervals.

In this setup, Algorithm 1 corresponds to the Euler’s method for numerical integration of an ordinary differential equation and, thus, follows a standard convergence analysis.

Once we have a procedure for the exponential map, we can establish the shooting method for finding geodesics. Let (p_1, q_1) be the starting point and (p_2, q_2) be the target point. The shooting method iteratively updates the tangent or shooting vector (u, w) on $\mathcal{T}_{(p_1, q_1)}(\mathbb{B})$ such that $\exp_{(p_1, q_1)}((u, w)) = (p_2, q_2)$. Then, the geodesic between (p_1, q_1) and (p_2, q_2) is given by $(x(s), v(s)) = \exp_{(p_1, q_1)}(s(u, w))$, $s \in [0, 1]$. The key step here is to use the current discrepancy between the point reached, $\exp_{(p_1, q_1)}((u, w))$, and the target, (p_2, q_2) , to update the shooting vector (u, w) , at each iteration. There are

Algorithm 1 Numerical Implementation of Exponential map on \mathbb{B}

Let the initial point be $(x(0), v(0)) \in \mathbb{B}$ and the tangent vector be $(u, w) \in \mathcal{T}_{(x(0), v(0))}(\mathbb{B})$. We have $x_s(0) = u$, $\nabla_{x_s} v(s)|_{s=0} = w$. We will approximate this map using n steps and let $\epsilon = \frac{1}{n}$. Then, for $i = 1, \dots, n$ the exponential map $(x(i\epsilon), v(i\epsilon)) = \exp_{(x(0), v(0))}(i\epsilon(u, w))$ is given as:

1. Set $x(\epsilon) = \exp_{x(0)}(\epsilon x_s(0))$, where $x_s(0) = u$, and $v(\epsilon) = (v^\parallel + \epsilon w^\parallel)$, where v^\parallel and w^\parallel are parallel transports of $v(0)$ and w along path x from $x(0)$ to $x(\epsilon)$, respectively.
2. For each $i = 1, 2, \dots, n-1$, calculate

$$x_s(i\epsilon) = [x_s((i-1)\epsilon) + \epsilon \nabla_{x_s} x_s((i-1)\epsilon)]_{x((i-1)\epsilon) \rightarrow x(i\epsilon)},$$

where $\nabla_{x_s} x_s((i-1)\epsilon) = -R(v((i-1)\epsilon), \nabla_{x_s} v((i-1)\epsilon))(x_s((i-1)\epsilon))$ is given by the first equation in Theorem 1. It is easy to show that $R(v((i-1)\epsilon), \nabla_{x_s} v((i-1)\epsilon)) = R(v^\parallel + \epsilon(i-1)w^\parallel, w^\parallel) = R(v^\parallel, w^\parallel)$, where $v^\parallel = v(0)_{x(0) \rightarrow x((i-1)\epsilon)}$, and $w^\parallel = w_{x(0) \rightarrow x((i-1)\epsilon)}$.

3. Obtain $x((i+1)\epsilon) = \exp_{x(i\epsilon)}(\epsilon x_s(i\epsilon))$, and $v((i+1)\epsilon) = v^\parallel + (i+1)\epsilon w^\parallel$, where $v^\parallel = v(0)_{x(0) \rightarrow x((i+1)\epsilon)}$, and $w^\parallel = w_{x(0) \rightarrow x((i+1)\epsilon)}$.
-

several possibilities for performing the updates and we discuss one here. Since we have two components to update, u and w , we will update them separately: (1) Fix w and update u . For the u component, the increment can come from parallel translation of the vector $\exp_{\tilde{p}}^{-1}(p_2)$ (the difference between the reached point \tilde{p} and the target point p_2) from \tilde{p} to p_1 , where \tilde{p} is the first component of reached point $\exp_{(p_1, q_1)}((u, w))$. (2) Fix u and update w . For the w component, we can take the difference between q_2 and the second component of the point reached (denoted as \tilde{q}) as the increment. This is done by parallel translating \tilde{q} to $\mathcal{T}_{p_2}(M)$ (the same space as q_2) and calculate the difference, and then parallel translate the difference to $\mathcal{T}_{p_1}(M)$ to update w .

Once again we will assume that the TSRVFs q_1 and q_2 are piecewise constant curves on a uniform partition of the interval $[0, 1]$. Numerical accuracy of this shooting algorithm naturally depends on the numerical accuracy of Algorithm 1.

Recall that trajectories on M and their representations in \mathbb{B} are bijective. For each pair $(p, q) \in \mathbb{B}$, one can reconstruct the corresponding trajectory α using covariant integration. A numerical implementation of this procedure is summarized in Algorithm 3. Similar to Algorithm 1, Algorithm 3 is also an Euler’s method for numerical integration of an ordinary differential equation and, thus, follows a standard convergence analysis.

Algorithm 2 allows us to calculate the geodesic between two points in \mathbb{B} . So, for each point along the geodesic $(x(s), v(s))$ in \mathbb{B} , one can easily reconstruct the trajectory on M using Algorithm 3. Here, one sets

Algorithm 2 Shooting algorithm for calculating geodesic on \mathbb{B}

Given $(p_1, q_2), (p_2, q_2) \in \mathbb{B}$, select one point, say (p_1, q_1) , as the starting point and the other, (p_2, q_2) , as the target point. The shooting algorithm for calculating the geodesic from (p_1, q_1) to (p_2, q_2) is:

1. Initialize the shooting direction: find the tangent vector u at p_1 such that the exponential map $\exp_{p_1}(u) = p_2$ on the manifold M . Parallel transport q_2 to the tangent space of p_1 along the shortest geodesic between p_1 and p_2 , denoted as q_2^\parallel . Initialize $w = q_2^\parallel - p_1$. Now we have a pair $(u, w) \in \mathcal{T}_{(p_1, q_1)}(\mathbb{B})$.
 2. Construct a geodesic starting from (p_1, q_1) in the direction (u, w) using the numerical exponential map in Algorithm 1. Let us denote this geodesic path as $(x(s), v(s))$, where s is the time parameter for the geodesic path.
 3. If $(x(1), v(1)) = (p_2, q_2)$, we are done. If not, measure the discrepancy between $(x(1), v(1))$ and (p_2, q_2) using a simple measure, e.g. the \mathbb{L}^2 distance.
 4. Iteratively, update the shooting direction (u, w) to reduce the discrepancy to zero. This update can be done using a two-stage approach: (1) fix u and update w until converge; (2) fix w and update u until converge.
-

Algorithm 3 Covariant integral of q along α

Given a piecewise constant TSRVF q sampled at a uniform partition of size T , $\{t\delta | t = 0, 1, \dots, T-1\}$, $\delta = 1/T$, and the starting point p :

1. Set $\alpha(0) = p$, and compute $\alpha(\delta) = \exp_{\alpha(0)}(\delta q(0)|q(0)|)$, where \exp denotes the exponential map on M .
2. For $t = 1, 2, \dots, T-1$
 - (a) Parallel transport $q(t\delta)$ to $\alpha(t\delta)$ along the current trajectory from $\alpha(0)$ to $\alpha(t\delta)$, and call it $q^\parallel(t\delta)$.
 - (b) Compute

$$\alpha((t+1)\delta) = \exp_{\alpha(t\delta)}(\delta q^\parallel(t\delta)|q^\parallel(t\delta)|).$$

$x(s)$ as the starting point and $v(s)$ as the TSRVF of the trajectory.

2.4 Geodesic Distance on \mathbb{B}

Using the chosen Riemannian metric on \mathbb{B} (defined in Eqn. 2), the geodesic distance between any two points in \mathbb{B} is defined as the following.

Definition 2 Given two trajectories α_1, α_2 and their representations $(p_1, q_1), (p_2, q_2) \in \mathbb{B}$, and let $(x(s), v(s)) \in \mathbb{B}$, $s \in [0, 1]$ be the geodesic between (p_1, q_1) and (p_2, q_2) on \mathbb{B} , the geodesic distance is given as:

$$d_c((p_1, q_1), (p_2, q_2)) = \sqrt{l_x^2 + \int_0^1 |q_1^\parallel(\tau) - q_2(\tau)|^2 d\tau} \quad (4)$$

This distance has two components: (1) the length between the starting points on M , $l_x = \int_0^1 |\dot{x}(s)| ds$; and

(2) the standard \mathbb{L}^2 norm on \mathbb{B}_{p_2} between the TSRVFs of the two trajectories, where q_1^\parallel represents the parallel transport of $q_1 \in \mathbb{B}_{p_1}$ along x to \mathbb{B}_{p_2} . Since we have a numerical approach for approximating the geodesic, the same algorithm can also provide an estimate for the geodesic distance.

3 Analysis of Trajectories Modulo Re-Parameterization

The main motivation of using TSRVF representation for trajectories on M and constructing the distance d_c to compare two trajectories comes from the following. If a trajectory α is warped by γ , resulting in $\alpha \circ \gamma$, what is the TSRVF of the time-warped trajectory? The new TSRVF is given by:

$$\begin{aligned} q_{\alpha \circ \gamma}(t) &= \left(\frac{(\dot{\alpha}(\gamma(t))\dot{\gamma}(t))}{\sqrt{|\dot{\alpha}(\gamma(t))\dot{\gamma}(t)|}} \right)_{\alpha(\gamma(t)) \rightarrow p} \\ &= \left(\frac{(\dot{\alpha}(\gamma(t)))\sqrt{\dot{\gamma}(t)}}{\sqrt{|\dot{\alpha}(\gamma(t))|}} \right)_{\alpha(\gamma(t)) \rightarrow p} \\ &= q_\alpha(\gamma(t))\sqrt{\dot{\gamma}(t)} \equiv (q_\alpha * \gamma)(t). \end{aligned}$$

Theorem 2 For any two trajectories $\alpha_1, \alpha_2 \in \mathcal{F}$ and their representations $(p_1, q_1), (p_2, q_2) \in \mathbb{B}$, the metric d_c satisfies $d_c((p_1, q_{\alpha_1 \circ \gamma}), (p_2, q_{\alpha_2 \circ \gamma})) = d_c((p_1, q_1), (p_2, q_2))$, for any $\gamma \in \Gamma$

Proof: First, if a trajectory is warped by $\gamma \in \Gamma$, the resulting trajectory is $\alpha \circ \gamma$, i.e. γ acts on the space \mathbb{B} by $(p, q) * \gamma = (p, q * \gamma)$. The differential of this action is the map $\mathcal{T}_{(p, q)}(\mathbb{B}) \rightarrow \mathcal{T}_{(p, q * \gamma)}(\mathbb{B})$ given by $(u, w) \mapsto (u, w * \gamma)$. We prove that this differential preserves our Riemannian inner product (Eqn. 2) as follows: let (u_1, w_1) and (u_2, w_2) be two tangent vectors on $\mathcal{T}_{(p, q)}(\mathbb{B})$; it follows that

$$\begin{aligned} &\langle (u_1, w_1 * \gamma), (u_2, w_2 * \gamma) \rangle \\ &= u_1 \cdot u_2 + \int_0^1 w_1(\gamma(t))\sqrt{\dot{\gamma}(t)}w_2(\gamma(t))\sqrt{\dot{\gamma}(t)}dt \\ &= u_1 \cdot u_2 + \int_0^1 w_1(\gamma)w_2(\gamma)d\gamma \quad (5) \\ &= u_1 \cdot u_2 + \int_0^1 w_1(s)w_2(s)ds \\ &= \langle (u_1, w_1), (u_2, w_2) \rangle, \end{aligned}$$

Since Γ acts on \mathbb{B} by isometries, i.e. preserving the Riemannian inner product, it follows immediately that it takes geodesics to geodesics, and preserves geodesic distance. **Q.E.D.**

Theorem 2 reveals the advantage of using TSRVF representation: the action of Γ on \mathbb{B} under the metric d_c is by isometries. The isometry property of time-warping

action under the metric d_c allows us to compare trajectories in such a manner that the resulting comparison is invariant to the time warping. This is achieved through defining a distance in the quotient space of reparameterization group.

3.1 Theoretical Setup

To form the quotient space of \mathbb{B} modulo the reparameterization group, we take the approach presented in several previous papers, including [27]. While [27] considers shapes of curves in Euclidean domains, these ideas naturally extend to the nonlinear manifolds also. The approach is to introduce a set $\tilde{\Gamma}$ as the set of all non-decreasing, absolutely continuous functions γ on $[0, 1]$ such that $\gamma(0) = 0$ and $\gamma(1) = 1$. This set is a semi-group with the composition operation (it does not have a well-defined inverse). It can be shown that Γ is a dense subset of $\tilde{\Gamma}$. For any $q \in \mathbb{L}^2([0, 1], T_p(M))$, let $[q]_{\tilde{\Gamma}}$ denote the set $\{(q * \gamma) | \gamma \in \tilde{\Gamma}\}$. This is a closed set [27, 38], while the orbit of q under Γ is not, and therefore we choose to work with the former, at least for the formal development. (However, in practice, we approximate solutions using the elements of Γ .) Note that the actions of Γ and $\tilde{\Gamma}$ on q are exactly same as if α was a Euclidean curve, as the kind studied in [27]. Therefore, borrowing results from [27], the closure of Γ -orbit of q is equal to the $\tilde{\Gamma}$ -orbit of q . Consequently, will call the set $[q]_{\tilde{\Gamma}}$ a *closed-up orbit* of q . We define the quotient space $\mathbb{B}/\tilde{\Gamma}$ as the set of all closed-up orbits, with each orbit being:

$$[(p, q)] \equiv (p, [q]) = \{(p, (q * \gamma)) | \gamma \in \tilde{\Gamma}\} .$$

To understand the concept of a closed-up orbit, one can view it as an equivalence class under the following relation. For any two trajectories α_1, α_2 and their representations in \mathbb{B} , $(p_1, q_1), (p_2, q_2)$, we define them to be equivalent when: (1) $p_1 = p_2$; and (2) there exists a sequence $\gamma_i \in \tilde{\Gamma}$ such that $q_{\alpha_2 \circ \gamma_i}$ converges to q_1 . In other words, if two trajectories have the same starting point, and the TSRVF of one can be time-warped into the TSRVF of the other, using a sequence of time-warpings, then these two trajectories are deemed equivalent to each other. Theorem 2 states that if two trajectories are warped by the same γ function, the distance d_c between them remains the same. In other words, the closed-up orbits in \mathbb{B} are “parallel” to each other.

The main reason for introducing the quotient space $\mathbb{B}/\tilde{\Gamma}$ is to define a proper distance on it and to compute geodesic paths between its elements with respect to this distance for the purposes of statistical analysis. We define a metric on the quotient space $\mathbb{B}/\tilde{\Gamma}$ using the inherent Riemannian metric from \mathbb{B} , as follows.

Definition 3 *The geodesic distance d_q on $\mathbb{B}/\tilde{\Gamma}$ is the shortest distance between two closed-up orbits in \mathbb{B} , given as*

$$d_q((p_1, [q_1]), (p_2, [q_2])) = \inf_{\gamma_1, \gamma_2 \in \tilde{\Gamma}} d_c((p_1, (q_1 * \gamma_1)), (p_2, (q_2 * \gamma_2))) \quad (6)$$

For a similar representation, [38] established that the induced distance is a proper distance on the set of closed-up orbits and that same proof applies to the current context also. It is also similar to the theory described for Euclidean curves in [27].

In order to compute geodesics paths in $\mathbb{B}/\tilde{\Gamma}$, one can solve for the optimization problem stated in Eqn. 6 and use the optimal points to form geodesics in the upper space \mathbb{B} . That is, for any $\alpha_1, \alpha_2 \in \mathbb{B}$, and the corresponding representations $(p_1, q_1), (p_2, q_2)$, we first solve for

$$(\hat{\gamma}_1, \hat{\gamma}_2) = \operatorname{argmin}_{\gamma_1, \gamma_2 \in \tilde{\Gamma}} d_c((p_1, (q_1 * \gamma_1)), (p_2, (q_2 * \gamma_2))) . \quad (7)$$

Then, we simply compute a geodesic path between $(p_1, q_1 * \hat{\gamma}_1)$, and $(p_2, q_2 * \hat{\gamma}_2)$ in \mathbb{B} , as described in the previous section.

3.2 Numerical Approximations

Conceptually, the geodesic and the geodesic distance between closed-up orbits $(p_1, [q_1])$ and $(p_2, [q_2])$ are defined by optimizing over geodesics between all possible cross pairs in sets $(p_1, [q_1])$ and $(p_2, [q_2])$. This, in turn, requires a double optimization on the set $\tilde{\Gamma}$, as stated in Definition 3. We now look at the computational aspects of this definition and seek some faster approximations.

Firstly, since Γ is dense in $\tilde{\Gamma}$, we can compute the geodesic distance $d_q((p_1, [q_1]), (p_2, [q_2]))$ using only a single optimization on the group Γ . This is because:

$$\begin{aligned} & \operatorname{argmin}_{\gamma_1, \gamma_2 \in \tilde{\Gamma}} d_c((p_1, (q_1 * \gamma_1)), (p_2, (q_2 * \gamma_2))) \\ &= \operatorname{argmin}_{\gamma_1, \gamma_2 \in \Gamma} d_c((p_1, (q_1 * \gamma_1)), (p_2, (q_2 * \gamma_2))) \\ &= \inf_{\gamma \in \Gamma} d_c((p_1, q_1), (p_2, (q_2 * \gamma))) . \end{aligned} \quad (8)$$

There is no approximation here and the infimum on a single Γ is much faster compared to the double optimization on $\tilde{\Gamma}$.

If we further assume that the trajectories α_1, α_2 are piecewise geodesic and, thus, their TSRVFs are piecewise constants, then some additional results hold. The paper [27] provided an exact approach for optimal alignment of SRVFs of piecewise linear curves in Euclidean spaces. Since TSRVFs are Euclidean curves, that approach can be easily adapted to solve for the optimal

alignment in Eqn. 7. This would provide optimal alignment and consequently a precise geodesic path between $(p_1, [q_1])$ and $(p_2, [q_2])$.

However, this approach can be slow in practice, and one can speed the implementation using a single optimization according to Eqn. 8. That is, we can solve for a single $\hat{\gamma}$ and match the point $\alpha_1(t)$ to the point $\alpha_2(\hat{\gamma}(t))$. While this approach is much faster than the joint optimization, a drawback here is that there is no guarantee that we are close to an optimal matching. However, in practice, we have found that matchings and geodesic paths obtained this way are quite similar to the optimal solutions in most real data. To solve Eqn. 8, it is equivalent to optimize the following equation:

$$\min_{(x,v),\gamma} \left(l_x^2 + \int_0^1 \|q_{1,x}^{\parallel}(t) - (q_2 * \gamma)(t)\|^2 dt \right), \quad (9)$$

where (x, v) is the path between (p_1, q_1) and $(p_2, q_{\alpha_2 \circ \gamma})$, and $q_{1,x}^{\parallel}$ means parallel transport q_1 along x to \mathbb{B}_{p_2} . Note that the time-warping γ acting on α_2 changes the underlying geodesic (x, v) between two trajectories. Algorithm 4 describes a numerical solution for optimizing Eqn. 9 on a general manifold M .

Algorithm 4 Pairwise registration of two trajectories on M

Represent two trajectories α_1, α_2 by their TSRVFs, (p_1, q_1) and (p_2, q_2) . Initialize $\gamma^* = \gamma_{id}$, and set $itermax = K$ (a large integer), $iter = 1$ and a small $\epsilon > 0$.

1. Select one point, say (p_1, q_1) , as the starting point and the other, (p_2, \tilde{q}_2) , as the target point, where \tilde{q}_2 denotes $(q_{\alpha_2} * \gamma)$, for $\gamma \in \Gamma$. In this step, let $\gamma = \gamma_{id}$.
 2. Obtain $(u, w) \in \mathcal{T}_{(p_1, q_1)}(\mathbb{B})$ such that $\exp_{(p_1, q_1)}(s(u, w)) = (x(s), v(s))$, $s \in [0, 1]$ and $(x(1), v(1)) = (p_2, \tilde{q}_2)$.
 3. Parallel transport \tilde{q}_2 to the tangent space $\mathcal{T}_{p_1}(M)$ along $x(s)$, denoted as \tilde{q}_2^{\parallel} . Align \tilde{q}_2^{\parallel} to q_1 using Dynamic Programming Algorithm and obtain the optimal warping function γ .
 4. Update $\gamma^* = \gamma^* \circ \gamma$ by composition. If $\|\gamma - \gamma_{id}\| < \epsilon$ or $iter > itermax$ stop. Else, set $\tilde{q}_2 = (q_{\alpha_2} * \gamma)$, $iter = iter + 1$ and go back to step 3.
-

Note that in Algorithm 4, *Step 2* corresponds to the first argument (x, v) and *Step 3* corresponds to the second argument γ in Eqn. 9, respectively. The optimization over the warping function in *Step 3* is achieved using the Dynamic Programming Algorithm (page 435-436 in [35]). Here one samples the interval $[0, 1]$ using N discrete points and then restricts to only piecewise linear γ 's that pass through that $N \times N$ grid. In practice, Algorithm 4 typically takes a few iterations to converge.

Since Algorithm 4 involves multiple evaluations of the exponential map and dynamic programming alignment, it is still not computationally very efficient. We

further speed up this computation as follows: find the baseline $x(s)$ connecting two trajectories first (using geodesic between $\alpha_1(0)$ and $\alpha_1(1)$ on M) and then align their TSRVFs accordingly. This substantially speeds up the solution albeit at the cost of diverging from the optimal solution stated under the theoretical formulation. In the experimental results presented later, we use this method to speed up registration and comparison.

4 Riemannian Structure on $\tilde{\mathcal{P}}$

Next we discuss the geometry of $\tilde{\mathcal{P}}$ and impose a Riemannian structure that facilitates our analysis of trajectories on $\tilde{\mathcal{P}}$. Several past papers have studied the space of SPDMS as a nonlinear manifold and have imposed metric structures on that manifold [3, 12, 19, 31, 34]. While they mostly seek to define distances on this set, a few of these distances originate from a Riemannian structure with expressions for geodesics and exponential maps. However, the most common Riemannian framework [31] does not provide expressions for all desired items that are needed in our context, especially expressions for parallel transport and Riemannian curvature tensor. Therefore, we choose a more recent Riemannian structure that was introduced in [37], and subsequently used in [38]. We will summarize the main results here and refer the reader for more details to these papers and two supplementary files.

Let $\tilde{\mathcal{P}}$ be the space of $n \times n$ SPDMS, and let \mathcal{P} be its subset of matrices with determinant one. The idea is to first identify the space \mathcal{P} with the quotient space $SL(n)/SO(n)$ and borrow the Riemannian structure from the latter directly. Then, one can straightforwardly extend the Riemannian structure on \mathcal{P} to $\tilde{\mathcal{P}}$. The process starts by choosing a Riemannian metric on G as follows: for any point $G \in SL(n)$ the metric is defined by pulling back the tangent vectors under G^{-1} to I , and then using the *trace* metric (see more details in the Section 1 of Supplementary Material I). This definition leads to expressions for the exponential map, its inverse, parallel transport of tangent vectors, and the Riemannian curvature tensor on $SL(n)$. It also induces a Riemannian structure on the quotient space $SL(n)/SO(n)$ in a natural way because the chosen metric is invariant to the action of $SO(n)$ on $SL(n)$. Finally, these results are transferred to \mathcal{P} using the mapping

$$\pi : SL(n)/SO(n) \rightarrow \mathcal{P}, \pi([G]) = \sqrt{\tilde{G}\tilde{G}^t},$$

for any $\tilde{G} \in [G]$. One can check that this map is well defined and is a diffeomorphism, by letting $\tilde{G} = PS$ (polar decomposition), and then $\pi([G]) = \sqrt{\tilde{G}\tilde{G}^t} =$

$\sqrt{PSS^tP} = P$. This square-root is the symmetric, positive-definite square-root of a symmetric matrix. The inverse map of π is given by: $\pi^{-1}(P) = [P] \equiv \{PS|S \in SO(n)\} \in SL(n)/SO(n)$. This establishes a one-to-one correspondence between the quotient space $SL(n)/SO(n)$ and \mathcal{P} . In turn, this correspondence is used to derive required expressions for geodesics, exponential map, curvature tensor, etc, on \mathcal{P} . We refer the reader to the supplementary material I for more details.

These results are then extended to the set $\tilde{\mathcal{P}}$ using a product map. Since for any $\tilde{P} \in \tilde{\mathcal{P}}$ we have $\det(\tilde{P}) > 0$, we can express $\tilde{P} = (P, \frac{1}{n} \log(\det(\tilde{P})))$ with $P = \frac{\tilde{P}}{\det(\tilde{P})^{1/n}} \in \mathcal{P}$. Thus, $\tilde{\mathcal{P}}$ is identified with the product space of $\mathcal{P} \times \mathbb{R}$. To define a metric on this product space, we can use the square-root of sum of squares of the individual metrics but with arbitrary weights. Here we use the weight $1/n$ for the determinant term. We summarize expressions for the required mathematical tools on $\tilde{\mathcal{P}}$:

1. **Exponential map:** Give $\tilde{P} \in \tilde{\mathcal{P}}$ and a tangent vector $\tilde{V} \in \mathcal{T}_{\tilde{P}}(\tilde{\mathcal{P}})$. We denote $\tilde{V} = (V, v)$, where $V \in \mathcal{T}_P(\mathcal{P})$, $P = \tilde{P}/\det(\tilde{P})^{1/n}$ and $v = \frac{1}{n} \log(\det(\tilde{P}))$. The exponential map $\exp_{\tilde{P}}(\tilde{V})$ is given as $e^v \exp_P(V)$, where $\exp_P(V) = \sqrt{Pe^{2P^{-1}VP^{-1}}P}$.
2. **Geodesic distance:** For any $\tilde{P}_1, \tilde{P}_2 \in \tilde{\mathcal{P}}$, the squared geodesic distance between them is: $d_{\tilde{\mathcal{P}}}(\tilde{P}_1, \tilde{P}_2)^2 = d_{\mathcal{P}}(P_1, P_2)^2 + \frac{1}{n}(\log(\det(\tilde{P}_2)) - \log(\det(\tilde{P}_1)))^2$, where $P_{12} = \sqrt{P_1^{-1}P_2^2P_1^{-1}}$ and $d_{\mathcal{P}}(P_1, P_2) = \|A_{12}\|$ for $e^{A_{12}} = P_{12} \in \mathcal{P}$.
3. **Inverse exponential map:** For any $\tilde{P}_1, \tilde{P}_2 \in \tilde{\mathcal{P}}$, the inverse exponential map $\exp_{\tilde{P}_1}^{-1}(\tilde{P}_2) = \tilde{V} \equiv (V, v)$, where $V = P_1 \log(\sqrt{P_1^{-1}P_2^2P_1^{-1}})P_1$ and $v = \frac{1}{n} \log(\det(P_2)) - \frac{1}{n} \log(\det(P_1))$.
4. **Parallel transport:** For any $\tilde{P}_1, \tilde{P}_2 \in \tilde{\mathcal{P}}$ and a tangent vector $\tilde{V} = (V, v) \in \mathcal{T}_{\tilde{P}_1}(\tilde{\mathcal{P}})$, the parallel transport of \tilde{V} along the geodesic from \tilde{P}_1 to \tilde{P}_2 is: $(P_2T_{12}^TBT_{12}P_2, v)$, where $B = P_1^{-1}VP_1^{-1}$, $T_{12} = P_{12}^{-1}P_1^{-1}P_2$ and $P_{12} = \sqrt{P_1^{-1}P_2^2P_1^{-1}}$.
5. **Riemannian curvature tensor:** For any $\tilde{P} \in \tilde{\mathcal{P}}$, and tangent vectors $\tilde{X} = (X, x), \tilde{Y} = (Y, y)$ and $\tilde{Z} = (Z, z) \in \mathcal{T}_{\tilde{P}}(\tilde{\mathcal{P}})$, the Riemannian curvature tensor is given by $R(\tilde{X}, \tilde{Y})(\tilde{Z}) = -P[[A, B], C]$, where $A = P^{-1}XP^{-1}$, $B = P^{-1}YP^{-1}$, $C = P^{-1}ZP^{-1}$ and $[A, B] = AB - BA$.

Remark 1 The Riemannian structure used here and the one used previously [31] are both derived from the same

induced structure on the quotient space $SL(n)/SO(n)$. The difference lies in the mapping used to map the metric from $SL(n)/SO(n)$ to \mathcal{P} . In [31], the mapping from the quotient space \mathcal{P} is GG^T , leading to the relationship:

$$\begin{array}{ccc} SL(n) & \xleftarrow{\sqrt{P}} & \\ \downarrow & & \searrow \\ SL(n)/SO(n) & \xrightarrow[\pi_1^{-1}(P)=[\sqrt{P}]]{\pi_1([G])=GG^T} & \mathcal{P} \end{array}$$

while in our approach, this mapping is $\sqrt{GG^T}$, leading to the picture:

$$\begin{array}{ccc} SL(n) & \xleftarrow{P} & \\ \downarrow & & \searrow \\ SL(n)/SO(n) & \xrightarrow[\pi_2^{-1}(P)=[P]]{\pi_2([G])=\sqrt{GG^T}} & \mathcal{P} \end{array}$$

The main motivation for the current choice of mapping, and the resulting Riemannian metric on \mathcal{P} , is as follows. Consider any $G \in SL(n)$. It is an important fact that $\sqrt{GG^T}$ is the only element in $[G]$ that is also in \mathcal{P} . So, it is a very natural idea to represent the equivalence class $[G]$ with $\sqrt{GG^T}$ giving a representation of $SL(n)/SO(n)$ using \mathcal{P} . Note that GG^T , used in [31], is generally not an element of $[G]$. In view of this simplification, i.e. the identity mapping from \mathcal{P} to $SL(n)$, \mathcal{P} can be viewed as a subset of $SL(n)$. Thus, all the relevant expressions can be derived under this identification rather than treating \mathcal{P} as a separate space. Specifically, we have readily available expressions for geodesic, geodesic distance, exponential map and its inverse, parallel transport, and Riemannian curvature tensor on \mathcal{P} viewed as a subset of $SL(n)$.

5 Demonstration of Numerical Procedures

In this section, we demonstrate the numerical procedures of the proposed framework on simulated covariance trajectories. We used $M = \mathcal{P}(3)$, the set of 3×3 SPDMS with determinant one. Extension to $\tilde{\mathcal{P}}$ is straightforward.

Geodesic computation: As a first example, we compute the geodesic between two arbitrary trajectories using the numerical method in Algorithm 2. Fig. 3 shows the result. In this plot, each matrix is visualized by an ellipsoid and a trajectory in $\mathcal{P}(3)$ by a sequence of ellipsoids. The top row shows two original trajectories α_1 and α_2 with representations (p_1, q_1) and (p_2, q_2) . The next row shows the baseline path $x(s)$ associated with the geodesic between α_1 and α_2 , and the end point

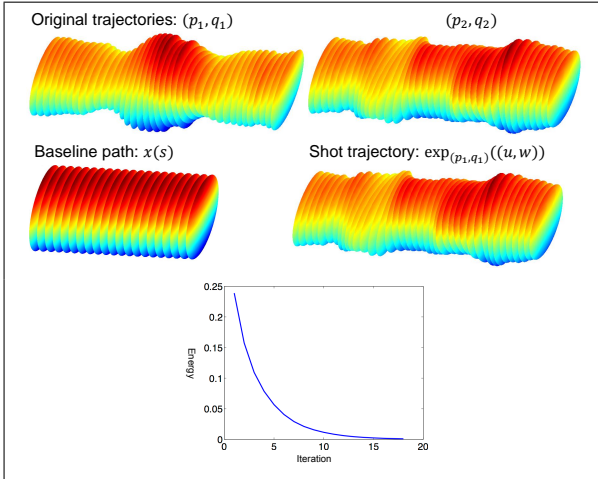


Fig. 3 Example of calculating geodesic using shooting method for trajectories on \mathcal{P} . The first row shows the original trajectory (p_1, q_1) and the target trajectory (p_2, q_2) . The second and third row show some results obtained from Algorithm 2: baseline curve $x(s)$ connecting p_1 and p_2 on \mathcal{P} , the final shot trajectory $\exp_{(p_1, q_1)}(u, w)$ and the \mathbb{L}^2 discrepancy between the shot trajectory and the target trajectory (p_2, q_2) v.s. number of iterations.

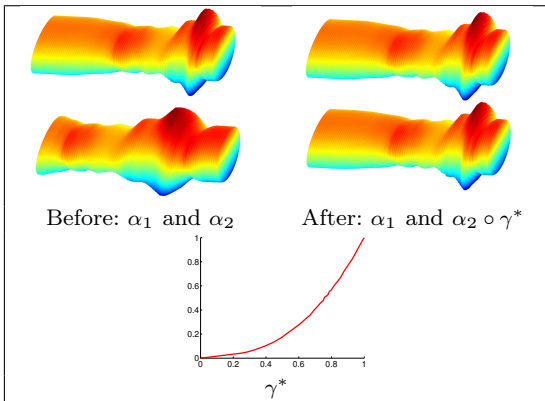


Fig. 4 Pairwise registration of two trajectories α_1 (first row) and α_2 (second row). The bottom panel shows the warping function γ to warp α_2 to α_1

of the geodesic, i.e. $\exp_{(p_1, q_1)}(u, w)$). Here we selected (p_1, q_1) as the starting point and computed the shooting direction (u, w) such that $\exp_{(p_1, q_1)}(u, w) \approx (p_2, q_2)$. The bottom panel shows the evolution of \mathbb{L}^2 norm between the shot trajectory and the target (p_2, q_2) during the shooting algorithm.

Temporal alignment: Next, we present an example of aligning two trajectories α_1 and α_2 in $\mathcal{P}(3)$ in Fig. 4. This alignment is based on particularization of Algorithm 4 to $M = \mathcal{P}(3)$. As the figure shows, the two trajectories are very well aligned as a result.

Computation of summary statistics: Finally, we focus on the problem of generating statistical sum-

maries of covariance trajectories. Since d_q defines a metric in the quotient space $\mathbb{B}/\tilde{\Gamma}$, this framework allows us to perform statistical analysis of multiple trajectories in $\mathbb{B}/\tilde{\Gamma}$. Given a set of trajectories $\{\alpha_i, i = 1 \dots k\}$, we are interested in computing the average of these trajectories and using it as a template for registering these trajectories. The sample mean can be approximated through:

$$(\mu_p, [\mu_q]) = \operatorname{argmin}_{(p, [q]) \in \mathbb{B}/\tilde{\Gamma}} \sum_{i=1}^n d_q((p, [q]), (p_i, [q_{\alpha_i}]))^2. \quad (10)$$

Note that $(\mu_p, [\mu_q])$ is an orbit (equivalence class of trajectories) and one can select any element of this orbit as a template to align multiple trajectories.

Algorithm 5 Calculation of Mean Trajectory

For each α_i , compute its mathematical representation (p_i, q_i) . Let (μ_p^j, μ_q^j) , $j = 0$ be the initial estimate of the mean (e.g. we can choose one of the trajectories). Set small $\epsilon, \epsilon_1, \epsilon_2 > 0$.

1. For $i = 1$ to n , align each trajectory (p_i, q_i) to (μ_p^j, μ_q^j) according to Algorithm 4, denoted as (p_i, \tilde{q}_i) . Algorithm 4 also gives us the inverse exponential map: $(u_i, w_i) = \exp_{(\mu_p^j, \mu_q^j)}^{-1}(p_i, \tilde{q}_i)$.
 2. Compute the average direction: $\bar{u} = \frac{1}{n} \sum_{i=1}^n u_i$, $\bar{w} = \frac{1}{n} \sum_{i=1}^n w_i$.
 3. If $\|\bar{u}\| < \epsilon_1$ and $\|\bar{w}\| < \epsilon_2$, stop. Else, update (μ_p^j, μ_q^j) in the direction of (\bar{u}, \bar{w}) using exponential map: $(\mu_p^{j+1}, \mu_q^{j+1}) = \exp_{(\mu_p^j, \mu_q^j)}(\epsilon \bar{u}, \epsilon \bar{w})$, where ϵ is the step size. We suggest using $\epsilon = 0.5$.
 4. Set $j = j + 1$, return to step 2.
-

For discussions on existence of this Riemannian sample mean and convergence of Algorithm 5 to a limit, we refer the reader to [1, 21].

After Algorithm 5 coverages, one can compute the covariant integral of (μ_p, μ_q) using Algorithm 3, denoted by μ , which is the mean of $\{\alpha_1, \alpha_2, \dots, \alpha_n\}$. Fig. 5 shows an example of calculating the mean of given trajectories. The upper part of the figure shows four simulated trajectories. The bottom part shows the mean trajectory in two situations: in \mathbb{B} using d_c (without temporal alignment) and in $\mathbb{B}/\tilde{\Gamma}$ under d_q (with alignment). One can see that under d_q the structures along the trajectories are better preserved.

6 Comparison with Previous Work

The proposed framework is an improvement of [38] in the following sense. It preserves the invariance properties achieved in [38], but does not require choosing a global reference point. Also, this framework naturally includes the difference between the starting points of

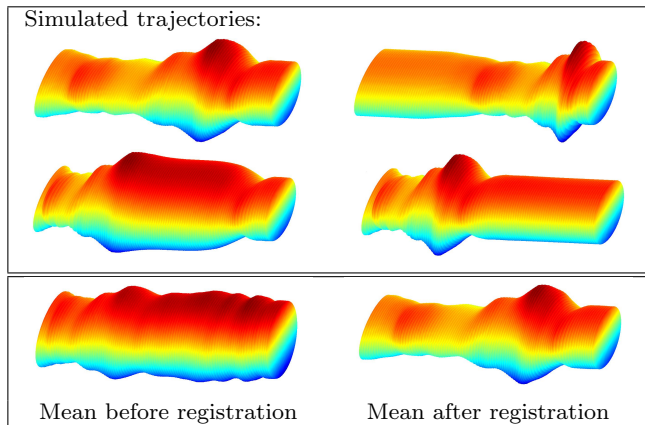


Fig. 5 Example of calculating the mean trajectory. The upper panel shows simulated trajectories, and the bottom panel shows means before and after alignment.

two trajectories that was ignored in [38]. Since the velocity vectors here are transported to the starting point of a trajectory, along that trajectory, as opposed to a transport to an arbitrary reference point in [38], this representation is more stable.

To quantitatively compare with [38,39], we performed the same visual speech recognition task utilizing the same subset of OuluVS dataset [39,42]. Here, we briefly introduce the experiment setup and more details can be found in [39]. OuluVS dataset includes 20 speakers, each uttering 10 everyday greetings five times: *Hello, Excuse me, I am sorry, Thank you, Good bye, See you, Nice to meet you, You are welcome, How are you, Have a good time*. Thus, the database has a total of 1000 videos. All the image sequences are segmented, having the mouth region determined by manually labeled eye positions in each frame [43]. Some examples of the segmented mouth images are shown in Fig. 6. We performed the experiment on a subset of the dataset, which contains 800 video sequences by removing some short videos [39]. The same covariance matrix features as [39] were extracted to represent each video. The resulting trajectories in $\tilde{\mathcal{P}}(7)$ are aligned using Algorithm 4 and compared using distance d_q defined in Eqn. 6.

In Fig. 7 (a), we show some optimal γ 's obtained to align one video of phrase (“excuse me”) to other videos of the same phrase spoken by the same person. One can see that there exist temporal differences in the original videos and they need to be aligned before further analysis. In (b), we show the histogram of $(d_c - d_q)/\max(d_c, d_q)$'s (the relative distance changes before and after alignment). In this case, each person has 50 videos, and we can calculate $(50 \times 49)/2$ pairwise distances before and after alignment, and their differences. For all 20 persons in this dataset, we have

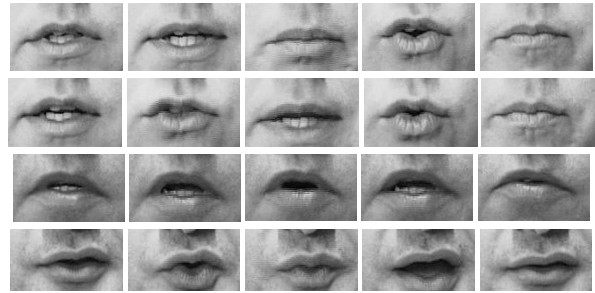


Fig. 6 Examples of down sampled video sequences in OuluVS dataset. The first and second row show one person's two speech samples of the phrase “Nice to meet you”; the third and fourth row show the phrases “How are you” and “Good bye” uttered by different persons.

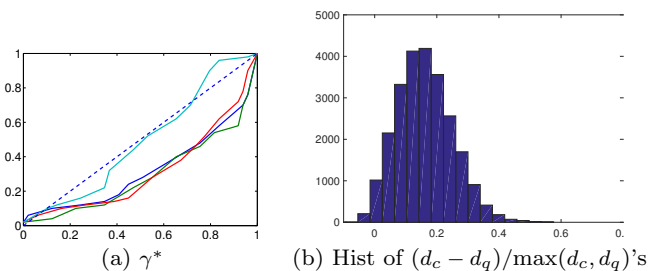


Fig. 7 (a) shows the optimal γ 's obtained to align one video of phrase (“excuse me”) to the other four videos of the same phrase spoken by the same person. (b) shows the histogram of $(d_c - d_q)/\max(d_c, d_q)$'s (The relative distance changes before and after alignment).

Table 1 Comparison of SDT performance on OuluVS.

Method		1NN Rate
Su et al. [39]	before alignment	33.8%
	after alignment	70.5%
Our method	before alignment	41.0%
	after alignment	78.6%

$20 \times (50 \times 49)/2 = 24500$ such differences. From the histogram of the relative changes, one can see that after our alignment, the distances (d_q 's) consistently become smaller.

Table 1 shows the average first nearest neighbor (1NN) classification rate of our method and [39]. Our method has the classification rate of 78.6%, which is 8.1% better than [39]'s. These results indicate that the new representation of trajectories and analysis framework have better discriminative power even before alignment comparing with the reference point based method in [38,39]. In addition, there is a 37.6% improvement due to alignment (registration), which demonstrates the importance of removing temporal difference in comparing of the dynamic systems in computer vision.

7 Application 1: Generative Modeling of Trajectories

Algorithm 5 results in several quantities of interest: (i) the mean trajectory $(\mu_p, [\mu_q])$, (ii) the aligned trajectories (p_i, \tilde{q}_i) , and (iii) the shooting vectors from the mean to (p_i, \tilde{q}_i) , denoted as (u_i, w_i) . Since our representation is invertible, one can develop generative models of given trajectories using these quantities. For example, we can use advanced statistical methods to infer the distribution of a set of trajectories, draw random samples and perform statistical inference. We illustrate these ideas using a simple example on $\mathcal{P}(3)$, as it is easier to visualize.

As described in the supplementary material I, the tangent element $X \in T_{\mu_p}(\mathcal{P})$ can be identified as $\mu_p B \mu_p$, where $B \in T_I(\mathcal{P}) = \{A | A^t = A \text{ and } \text{tr}(A) = 0\}$. For matrix size three, elements of $T_I(\mathcal{P}(3))$ has only five degrees of freedom. Let $\phi : T_I(\mathcal{P}(3)) \mapsto \mathbb{R}^5$ be an embedding given by $\phi(A) = [a_{11}, a_{12}, a_{13}, a_{22}, a_{23}]^T$. Given tangent vectors (u_i, w_i) for $i = 1, \dots, n$ in $T_{(\mu_p, \mu_q)}(\mathbb{B})$, we have $u_i \in T_{\mu_p}(\mathcal{P}(3))$ and $w_i \in \mathbb{L}^2([0, 1], T_{\mu_p}(\mathcal{P}(3)))$. We transform (u_i, w_i) into (u_i^0, w_i^0) such that $u_i^0 \in T_I(\mathcal{P}(3))$ and $w_i^0 \in \mathbb{L}^2([0, 1], T_I(\mathcal{P}(3)))$ according to

$$u_i = \mu_p u_i^0 \mu_p \text{ and } w_i(s) = \mu_p w_i^0(s) \mu_p, \text{ for } s \in [0, 1],$$

and perform the statistical modeling in $T_I(\mathcal{P}(3))$. These elements are further mapped into \mathbb{R}^5 and functions in \mathbb{R}^5 , respectively, using ϕ . Statistical modeling of the trajectories in $\mathcal{P}(3)$ becomes of modeling points in \mathbb{R}^5 and \mathbb{L}^2 functions in \mathbb{R}^5 .

Next, we consider the problem of fitting a distribution to the given sample trajectories. For the first component u_i , we use a simple multivariate Gaussian distribution. For the second component w_i , we follow a similar procedure as [26] to define a Gaussian distribution in a principal subspace and then map it back to the trajectory space. Let the trajectory α_i be sampled with a finite number of points, say m , we then calculate the the sample covariance matrix $\mathbf{K} \in \mathbb{R}^{5m \times 5m}$ similar to [26]. Let $\mathbf{K} = \mathbf{U} \Sigma \mathbf{U}^T$ be the singular value decomposition of \mathbf{K} , and let \mathbf{U}_r , the first r columns of \mathbf{U} , span the principal subspace of the observed data. The principal scores of each data can be calculated by projecting each $\phi(w_i^0)$ to this principal subspace, and we apply a Gaussian distribution to model the variation of these principal scores. Actually, \mathbf{U}_1 describes the first principal direction (PC), together with the variation $S(1, 1)$, and one can visualize the variability of trajectories along the first PC \mathbf{U}_1 .

In this simulation study we generate 50 random trajectories in $\mathcal{P}(3)$, some of them are shown in Fig. 8(a).

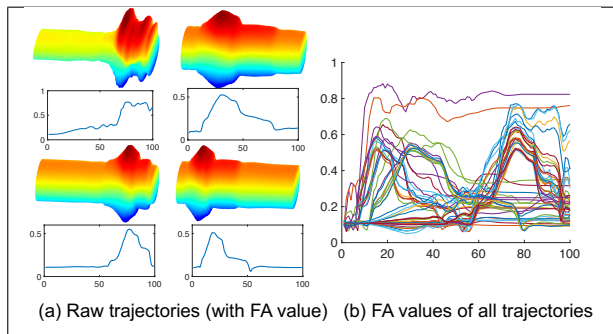


Fig. 8 Example of simulated trajectories.

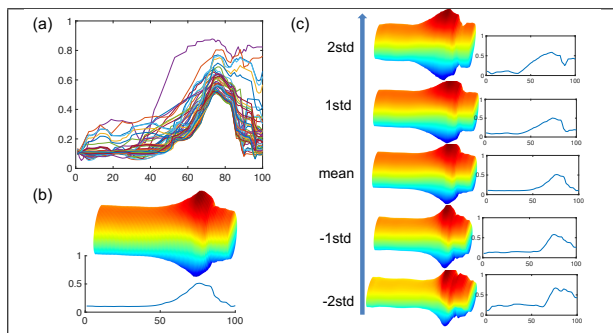


Fig. 9 Statistical modeling on the tangent space of the mean trajectory. (a) shows the FA curves of the aligned trajectories to the mean; (b) shows the mean trajectory and its FA curve; and (c) shows the first PC direction.

To give an idea about variability along these trajectories, we compute the fractional anisotropy (FA) [4] value of each SPDM in a trajectory, and visualize it as a scalar function of time. FA is a scalar value between zero and one that describes the anisotropy of the given SPDM. In Fig. 8 (b), we show FA curves of the 50 simulated trajectories. We can see that the bump's differ in locations and heights across the 50 trajectories. Using Algorithm 5, we calculate the mean trajectory, the aligned trajectories (p_i, \tilde{q}_i) , and the shooting vectors (u_i, w_i) . Fig. 9 (a) shows FA curves of the aligned trajectories, and (b) shows the mean trajectory and its FA curve. A PCA of shooting vectors leads to dimension reduction of data, which is necessary for reaching an efficient statistical model on trajectories. Fig. 9 (c) shows the variability of trajectories along the first PC of the given data.

Next we demonstrate a potential approach for imposing stochastic models on the space of trajectories. Since our representation of trajectories has two components, p and q , we impose individual stochastic models on the corresponding tangent representations u and w . We choose a simple univariate Gaussian model on u and similar multivariate Gaussian model on four princi-

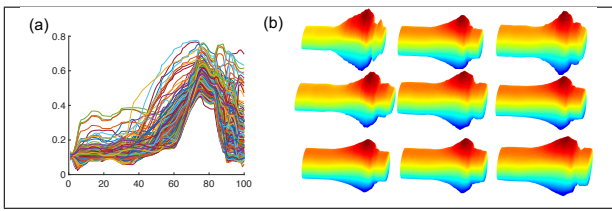


Fig. 10 Randomly sampled trajectories from the fitted model. (a) shows the FA curves of the 200 simulated trajectories and (b) shows 9 example trajectories.

pal components of w . We randomly sample (u, w) from the fitted distributions and utilize the exponential map defined in Algorithm 2 to map (u, w) to a random trajectory α in $\mathcal{P}(3)$. Fig. 10 shows some random samples from the Gaussian models discussed above. In (a), we show the FA curves of the 200 sampled trajectories and in (b) we show 9 random trajectories. Notice that, the statistical model is in \mathbb{B}/\tilde{T} , so we do not consider the variation of the wrapping functions γ . To build a more complex model that considers the variation of γ , we refer the reader to Chapter 7 in [35].

8 Application 2: Discriminative Analysis of Trajectories

Now we turn to evaluation of the framework developed for discriminative pattern recognition. We try two applications here: (1) action recognition using videos: a hand-gesture recognition application, and (2) dynamic functional brain connectivity study.

8.1 Hand Gesture Recognition

Hand gesture recognition using videos is an important research area since gesture is a natural way of communication and expressing intentions. People use gestures to depict sign language for deaf, convey messages in loud environment and to interface with computers. In this section, we are interested in applying our framework in video-base (dynamic) hand gesture recognition. We use the *Cambridge hand-gesture dataset* [24] which has 900 video sequences with nine different hand gestures: 100 video sequences for each gesture. The nine gestures result from 3 primitive hand shapes and 3 primitive motions, and as collected under different illumination conditions. Some example gestures are shown in Fig. 11. The gestures are imaged under five different illuminations, labeled as Set1, Set2, ..., Set5.

In addition to the illumination variability, the main challenge here comes from the fact that hands in this database are not well aligned, e.g. the proportion of a

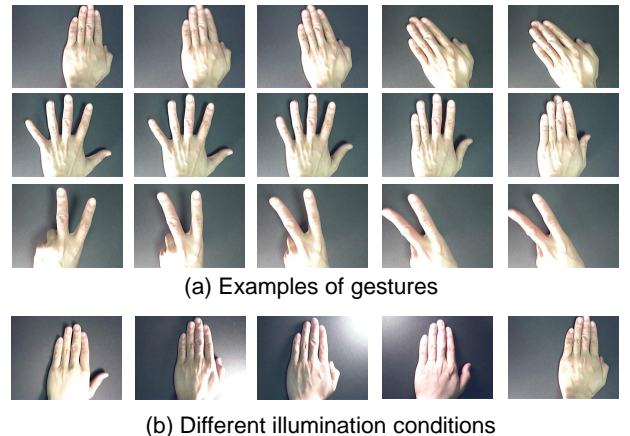


Fig. 11 (a) shows three examples of gestures in the Cambridge hand-gesture database. (b) shows the five different illumination conditions in the database.

hand in an image and the location of the hand are different in different video sequences. To reduce these effects we evenly split one image into four quadrants (upper-left, upper-right, bottom-left, bottom-right) with some overlaps. Each of the four quadrants is represented by a sequence of covariance matrices in $\tilde{\mathcal{P}}$. In this experiment, we use HOG features [10] to form a covariance matrix per image quadrant as follows. We use 2×2 blocks of 8×8 pixel cells with 7 histogram channels to form HOG features. Those HOG features are then used to generate a 7×7 covariance matrix for each quadrant of each frame. Thus, our representation of a video is now given by $t \mapsto \alpha(t) \in \tilde{\mathcal{P}}(7)^4$.

Since we have split each hand gesture into four dynamic parts, the total distance between any two hand gestures is a composite of four corresponding distances. For each corresponding dynamic quadrant, e.g. the upper-left part, we first align a pair of videos (using Algorithm 4) and then compare them using the metric d_q , denoted by d_{upl}^q . The final distance is obtained using an weighted average of the four parts: $d = \lambda_1 d_{upl}^q + \lambda_2 d_{upr}^q + \lambda_3 d_{downl}^q + \lambda_4 d_{downr}^q$ and $\sum_{i=1}^4 \lambda_i = 1$. For an unsupervised study, we set $\lambda_i = 1/4$ for $i = 1, \dots, 4$. In another setting, we learn a different weight for each illumination to make d be more robust to the registration and illumination issues. In our experiment, we randomly selected half of the data (90 video sequences) as the training data, and the other half of the data are used for the testing. Table 2 shows our results using the nearest neighbor classifier on all five sets. One can see that the alignment can significantly improve the classification rate on every illumination condition in both supervised and unsupervised settings. We also reported

Table 2 Recognition results on the Cambridge Hand-Gesture dataset. AA represents after alignment and BA represents before alignment.

Method		Set1	Set2	Set3	Set4	Set5
TCCA [23]		81%	81%	78%	86%	-
RLPP [17]		86%	86%	85%	88%	-
PM 1-NN [29]		89%	86%	89%	87%	-
PMLSR [28]		93%	89%	91%	94%	-
kgLC [16]		96%	94%	96%	98%	-
Our	BA	94%	91%	90%	88%	77%
	AA (unsup.)	98%	95%	93%	97%	94%
	AA (sup.)	99%	97%	97%	96%	98%

the state-of-the-art results on this database [16, 28, 29]. One can see that with only a 7×7 covariance feature per quadrant per frame, we are able to achieve a classification result that is equivalent or better than the state-of-the-art results. The classification result may further be improved using a more discriminative feature or better classifier.

8.2 Dynamic Functional Connectivity Study

Another interesting application of the proposed framework is in analyzing functional brain connectivity using functional magnetic resonance imaging (fMRI) data. Functional Connectivity (FC) is defined as statistical dependencies among remote neurophysiological events. These dependencies are expressed as quantifications of similarity, or correlations, between simultaneous functional measurements of neuronal activities across regions in human brain. The short-term FC is often represented as a covariance or correlation matrix of fMRI data over a small time window, with the matrix size being the number of brain regions. In the early studies, FC associated with individual tasks or stimuli was treated as fixed or static over time. However, later studies [18] revealed that FC is a dynamic process and evolves over time. Therefore, it is natural to represent FC observed over a long interval as an indexed sequence of covariance matrices, or as a covariant trajectory. Consequently, we can use the method developed in this paper for comparing and analyzing such FC.

We present some experimental results using data from the Human Connectome Project (HCP). For the first experiment, we select fMRI data for 20 subjects during resting state and during performances of various other tasks. The data are aligned and denoised using HCP data preprocessing pipeline [14], and the Destrieux atlas [11] is used to parcellate cortical regions into 74 nodes per hemisphere. We choose 10 regions, including the inferior frontal gyrus and sulcus,

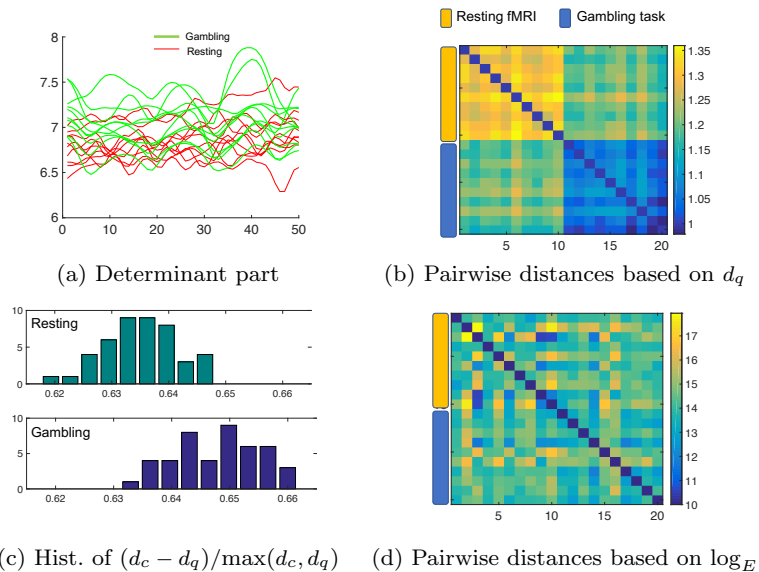


Fig. 12 (a) shows the determinant part ($\frac{1}{n} \log(\det(\tilde{P}))$) of the 20 dynamic brain subnetworks. (b) shows the pairwise distances between the 20 subnetworks after alignment. (c) shows the histogram of $(d_c - d_q)/\max(d_c, d_q)$. (d) shows the pairwise distances calculated based on \log_E metric.

and transverse temporal sulcus, that are related to the go/no tasks [2]. The connectivity of these regions is represented as a trajectory of covariance matrices between the regions. We use a sliding window [18] to calculate a covariance trajectory. For comparisons, we have selected two tasks: resting state fMRI and gambling task fMRI, and 10 trajectories for each task.

The results are shown in Fig. 12. In panel (a), we show the determinant ($\frac{1}{n} \log(\det(\tilde{P}(t)))$), as function of t , of these 20 covariance trajectories. We can clearly see fluctuations in the dynamics of the determinant part over time. Since the gambling task is carefully designed to repeat the reward, neutral and loss blocks, we see similar periodic fluctuations for different subjects (but there are some temporal misalignment due to inter-subject variability). This dynamic pattern seems different from that for the resting state fMRI. Fig. 12 (b) shows the pairwise distances d_q between the 20 trajectories. The block pattern in this distance matrix is an evidence that the resting state are very different from the gambling task state. The dynamic FC trajectories are more homogeneous in the gambling task (after temporal alignment). The temporal alignment plays an important role in studying dynamic FC. Panel (c) shows the histograms of percentages of distance changes before and after alignment for resting and gambling task. We see a significant reduction in distances in both cases, but the percentage reduction for task trajectories is more than that of the resting ones. This is due to the

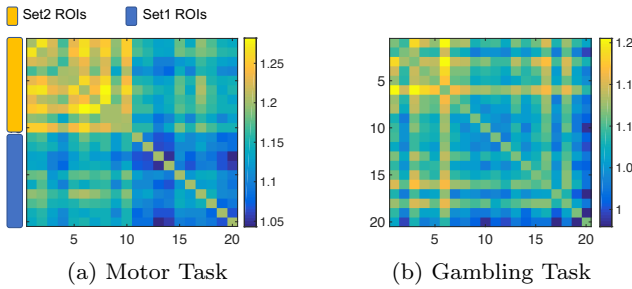


Fig. 13 The pairwise distances (based on d_q) between covariance trajectories generated from different set of nodes. (a) shows the distance in the motor task for 20 trajectories after the alignment. (b) shows the distance in the gambling task.

fact that the gambling task is performed under a well designed rule, and although there are some temporal misalignment, everyone in the experiment exhibits similar dynamic FC among the selected ROIs. For comparison, we also compute pairwise trajectory distances under the log-Euclidean distance [44]. That is, for any two trajectories α_1, α_2 , the distance between them is calculated as $d_{\log_E}(\alpha_1, \alpha_2) = \int_0^1 \|\log(\alpha_1(t)) - \log(\alpha_2(t))\| dt$. Panel (d) shows the pairwise distance matrix calculated using d_{\log_E} for the same data. One can see that there is no block pattern there, indicating the lack of power of this metric in dynamic FC study.

In the previous experiment, the 10 selected regions are marked as “Set1 ROIs”. We take another set of 10 regions (postcentral and precentral gyrus, and central and precentral sulcus, that are part of the motor cortex) to generate another set of covariance trajectories. These 10 regions are denoted as “Set2 ROIs”. We compare the dynamic FC generated by these two sets of ROIs under different tasks, and the results are shown in Fig. 13. The results show that while the dynamic FC are very different for the two sets of ROIs for the motor task, the connectivities are not that well separated for the gambling task. More importantly, these experiments demonstrate the potential of the proposed framework in performing statistical analysis of the dynamic functional connectivities and in linking dynamic connectivity to specific tasks, for example, for prediction purposes. There are many other interesting and critical problems that can be explored using the developed framework.

9 Conclusion

In summary, we have proposed metric-based approach for simultaneous alignment and comparisons of trajectories on $\tilde{\mathcal{P}}$, the Riemannian manifold of covariance ma-

trices (SPDMs). In order to facilitate our analysis, we impose a Riemannian structure on this manifold that facilitates explicit expressions for geometric quantities, such as parallel transport and Riemannian curvature tensor. For analyzing covariance trajectories, the basic idea is to represent each trajectory by a starting point $\tilde{P} \in \tilde{\mathcal{P}}$ and a TSRVF which is a curve in the tangent space $\mathcal{T}_{\tilde{P}}(\tilde{\mathcal{P}})$. The metric for comparing these elements is a composite of: (a) the length of the path between the starting points and (b) the difference introduced by the TSRVFs. The search for optimal path, or a geodesic, is based on a shooting method, that in itself uses geodesic equations for computing the exponential map. Using a numerical implementation of the exponential map, we derive numerical solutions for pairwise alignment of covariance trajectories and to quantify their differences using a rate-invariant distance. We have applied this framework to two scenarios: (1) covariance tracking in video data, with an application to the hand-gesture recognition, and (2) dynamic functional connectivity study in fMRI data. The advantages and potential applications of the proposed framework have been demonstrated in these experiments.

References

1. Afsari, B., Tron, R., Vidal, R.: On the convergence of gradient descent for finding the Riemannian center of mass. *SIAM Journal on Control and Optimization* **51**, 2230–2260 (2013)
2. Aron, A.R., Robbins, T.W., Poldrack, R.A.: Inhibition and the right inferior frontal cortex. *Trends Cogn. Sci. (Regul. Ed.)* **8**(4), 170–177 (2004)
3. Arsigny, V., Fillard, P., Pennec, X., Ayache, N.: Log-Euclidean metrics for fast and simple calculus on diffusion tensors. *Magnetic Resonance in Medicine* **56**(2), 411–421 (2006)
4. Basser, P.J., Pierpaoli, C.: Microstructural and physiological features of tissues elucidated by quantitative-diffusion-tensor MRI. *J Magn Reson B* **111**(3), 209–219 (1996)
5. Briggant, A.L.: Computing distances and geodesics between manifold-valued curves in the SRV framework. *arXiv:1601.02358* (2016)
6. Briggant, A.L., Arnaudon, M., Barbaresco, F.: Reparameterization invariant metric on the space of curves. *arXiv:1507.06503* (2015)
7. Celledoni, E., Eslitzbichler, M., Schmeding, A.: Shape analysis on lie groups with applications in computer animation. *Journal of Geometric Mechanics* **8**(3), 273–304 (2016)
8. Dai, M., Zhang, Z., Srivastava, A.: Testing stationarity of brain functional connectivity using change-point detection in fMRI data. In: *CVPR Workshops Diff-CVML*, pp. 981–989 (2016)
9. Dai, M., Zhang, Z., Srivastava, A.: Discovering change-point patterns in dynamic functional brain connectivity of a population. In: *IPMI* (2017)

10. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: International Conference on CVPR, vol. 2, pp. 886–893 (2005)
11. Destrieux, C., Fischl, B., Dale, A., Halgren, E.: Automatic parcellation of human cortical gyri and sulci using standard anatomical nomenclature. *Neuroimage* **53**(1), 1–15 (2010)
12. Dryden, I.L., Koloydenko, A.A., Zhou, D.: Non-Euclidean statistics for covariance matrices with applications to diffusion tensor imaging. *Annals of Applied Statistics* **3**(3), 1102–1123 (2009)
13. Faraki, M., Palhang, M., Sanderson, C.: Log-euclidean bag of words for human action recognition. *IET Computer Vision* **9**(3), 331–339 (2014)
14. Glasser, M.F., Sotiropoulos, S.N., Wilson, J.A., Coalson, T.S., Fischl, B., Andersson, J.L., Xu, J., Jbabdi, S., Webster, M., Polimeni, J.R., Van Essen, D.C., Jenkinson, M.: The minimal preprocessing pipelines for the Human Connectome Project. *Neuroimage* **80**, 105–124 (2013)
15. Guo, K., Ishwar, P., Konrad, J.: Action recognition in video by sparse representation on covariance manifolds of silhouette tunnels. In: Proceedings of the 20th International Conference on Recognizing Patterns in Signals, Speech, Images, and Videos, pp. 294–305 (2010)
16. Harandi, M., Hartley, R., Shen, C., Lovell, B., Sanderson, C.: Extrinsic methods for coding and dictionary learning on Grassmann manifolds. *International Journal of Computer Vision* **114**(2), 113–136 (2015)
17. Harandi, M.T., Sanderson, C., Wiliem, A., Lovell, B.C.: Kernel analysis over Riemannian manifolds for visual recognition of actions, pedestrians and textures. In: Proceedings of the 2012 IEEE Workshop on the Applications of Computer Vision, pp. 433–439 (2012)
18. Hutchison, R.M., et al.: Dynamic functional connectivity: Promise, issues, and interpretations. *NeuroImage* **80**, 360–378 (2013)
19. Jost, J.: *Riemannian Geometry and Geometric Analysis*. Springer (2005)
20. Jupp, P.E., Kent, J.T.: Fitting smooth paths to spherical data. *Journal of the Royal Statistical Society. Series C (Applied Statistics)* **36**(1), 34–46 (1987)
21. Kendall, W.S.: Probability, convexity, and harmonic maps with small image i: uniqueness and fine existence. *Proceedings of the London Mathematical Society* **3**(2), 371–406 (1990)
22. Kendall, W.S.: Barycenters and hurricane trajectories. *arXiv* **1406.7173** (2014)
23. Kim, T.K., Cipolla, R.: Canonical correlation analysis of video volume tensors for action categorization and detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **31**(8), 1415–1428 (2009)
24. Kim, T.K., Wong, K.Y.K., Cipolla, R.: Tensor canonical correlation analysis for action classification. In: IEEE Conference on CVPR, pp. 1–8 (2007)
25. Kume, A., Dryden, I.L., Le, H.: Shape-space smoothing splines for planar landmark data. *Biometrika* **94**, 513–528 (2007)
26. Kurtek, S., Srivastava, A., Klassen, E., Ding, Z.: Statistical modeling of curves using shapes and related features. *J. of the American Statistical Association* **107**(499), 1152–1165 (2012)
27. Lahiri, S., Robinson, D., Klassen, E.: Precise matching of PL curves in R^N in square-root velocity framework. *Geometry, Imaging, and Computing* **2**(3), 133–186 (2015)
28. Lui, Y.M.: Human gesture recognition on product manifolds. *Journal of Machine Learning Research* **13**(1), 3297–3321 (2012)
29. Lui, Y.M., Beveridge, J., Kirby, M.: Action classification on product manifolds. In: IEEE Conference on CVPR, pp. 833–839 (2010)
30. Morris, R.J., Kent, J., Mardia, K.V., Fidrich, M., Aykroyd, R.G., Linney, A.: Analysing growth in faces. In: International conference on Imaging Science, Systems and Technology (1999)
31. Pennec, X., Fillard, P., Ayache, N.: A Riemannian framework for tensor computing. *Int. J. Comput. Vision* **66**(1), 41–66 (2006)
32. Ramsay, J.O., Silverman, B.W.: *Functional Data Analysis*. Springer Series in Statistics. Springer (2005)
33. Samir, C., Absil, P.A., Srivastava, A., Klassen, E.: A gradient-descent method for curve fitting on Riemannian manifolds. *Foundations of Computational Mathematics* **12**(1), 49–73 (2012)
34. Schwartzman, A., Mascarenhas, W., Taylor, J.: Inference for eigenvalues and eigenvectors of Gaussian symmetric matrices. *Annals of Statistics* **36**(6), 2886–2919 (2008)
35. Srivastava, A., Klassen, E.: *Functional and Shape Data Analysis*. Springer (2016)
36. Srivastava, A., Klassen, E., Joshi, S., Jermyn, I.: Shape analysis of elastic curves in Euclidean spaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33**(7), 1415–1428 (2011)
37. Su, J., Dryden, I.L., Klassen, E., Le, H., Srivastava, A.: Fitting optimal curves to time-indexed, noisy observations on nonlinear manifolds. *Journal of Image and Vision Computing* **30**(6-7), 428–442 (2012)
38. Su, J., Kurtek, S., Klassen, E., Srivastava, A.: Statistical analysis of trajectories on Riemannian manifolds: Bird migration, hurricane tracking and video surveillance. *The Annals of Applied Statistics* **8**(1), 530–552 (2014)
39. Su, J., Srivastava, A., de Souza, F., Sarkar, S.: Rate-invariant analysis of trajectories on Riemannian manifolds with application in visual speech recognition. In: 2014 IEEE Conference on CVPR, pp. 620–627 (2014)
40. Tuzel, O., Porikli, F., Meer, P.: Region covariance: A fast descriptor for detection and classification. In: 9th European Conference on Computer Vision, pp. 589–600 (2006)
41. Zhang, Z., Su, J., Klassen, E., Le, H., Srivastava, A.: Video-based action recognition using rate-invariant analysis of covariance trajectories. *arXiv:1503.06699* (2015)
42. Zhao, G., Barnard, M., Pietikäinen, M.: Lipreading with local spatiotemporal descriptors. *IEEE Transactions on Multimedia* **11**(7), 1254–1265 (2009)
43. Zhao, G., Pietikäinen, M., Hadid, A.: Local spatiotemporal descriptors for visual recognition of spoken phrases. In: Proceedings of the International Workshop on Human-centered Multimedia, HCM '07, pp. 57–66 (2007)
44. Zhou, D., Dryden, I., Koloydenko, A., et al.: Non-euclidean analysis for covariance matrix data with application to diffusion tensor images (2013)