Procrustes Analysis on the Manifold of SPSD Matrices for Data Sets Alignment

Almog Lahav and Ronen Talmon, *Senior Member, IEEE*

***Abstract*—In contemporary high-dimensional data analysis, intrinsically similar and related data sets are often significantly different due to various undesired factors that could arise from different acquisition equipment, calibration, environmental conditions, and many other sources of batch effects. Therefore, the task of aligning such data sets has become ubiquitous. In this work, we present a method for the alignment of different, but related, sets of Symmetric Positive Semidefinite (SPSD) matrices, which constitute a commonly-used family of features, e.g., covariance and correlation matrices, various kernels, and prototypical graph and network representations. Our method does not require any a-priori correspondence, and it is based on non-Euclidean Procrustes Analysis (PA) using a particular Riemannian geometry of SPSD matrices. While the derivation is focused on the manifold of SPSD matrices, we show that our alignment method can be applied directly in the original high-dimensional data space, when considering SPSD features that are sample covariance matrices. We demonstrate the advantage of our approach over competing methods in simulations and in an application to Brain-Computer Interface (BCI) with electroencephalographic (EEG) recordings.**

***Index Terms*—Domain adaptation, symmetric positive semidefinite matrices, Riemannian geometry, BCI, EEG.**

I. Introduction

A LONGSTANDING problem in data science is how to represent, analyse, and process a union of different, but related, data sets. Often, due to the inherent heterogeneity of many types of data sets, useful representations usually cannot be achieved simply by merging multiple data sets into one big data set. Furthermore, a model learned from one data set is typically not appropriate as-is for the analysis of another. The density of the data sets could be different as a result of a broad variety of application-dependent differences, e.g., the recording conditions, the technology of the data acquisition, the production process and calibration of the recording device, the population of subjects, the physical conditions, etc. In order to exploit the model learned from one data set for analysis tasks in another data set, or to jointly process and analyze them, prior alignment of the data sets is required.

One approach for alignment, which is primarily based on geometric considerations, is Procrustes Analysis (PA) [1]. Originated in shape analysis, PA aims to adjust two or more shapes in order to facilitate a meaningful comparison between them. Given pairs of landmarks from both shapes, the adjustment is typically performed by applying three transformations: centering, scaling, and rotation, such that the distances between the landmarks are minimized. Extending this idea from shapes to high-dimensional point clouds facilitated an appealing data alignment approach, which is simple, efficient, and mathematically tractable, and it does not require any rigid a-priori model assumptions or estimates of the whole distribution of the data. Indeed, data alignment using PA has been successfully applied to various fields, including Brain-Computer Interface (BCI) [2], genetics and bioinformatics [3], [4], [5], indoor navigation [6], face recognition [7], and hierarchical representation [5], [8], to name but a few.

The application of PA to high-dimensional data sets poses challenges since such data typically do not live in a Euclidean space. The recent common practice for high-dimensional data analysis is based on the manifold assumption, i.e., to assume the existence of an intrinsic low-dimensional manifold underlying the data [9], [10], [11], [12]. However, learning the manifold from observations might be impractical in some real-world problems. An alternative approach is to use informative features with a-priori known and useful geometry. One natural example of such features is covariance matrices that embody multivariate associations. If the covariance matrices are full rank, they live on a Riemannian manifold, also known as the Symmetric Positive Definite (SPD) cone [13], [14]. Rodrigues et al. [2] presented PA based on the Riemannian geometry of SPD matrices, with application to BCI involving electroencephalographic (EEG) recordings of different subjects. Yet, many types of data sets, such as gene expression data [15] and hyper-spectral imaging data [16], [17], have a low dimensional structure, and therefore, the covariance matrices stemming from such data are rank deficient, i.e., they are Symmetric Positive Semidefinite (SPSD) matrices. Other examples of SPSD features are kernels, similarity matrices, and graph Laplacians, where the multiplicity of the zero eigenvalue equals the number of connected components of the associated graph [18].

In this work, we address the problem of aligning high-dimensional data sets by solving a new PA problem designed specifically for SPSD matrices with a fixed rank [19]. Given two

Manuscript received 10 July 2022; revised 31 January 2023; accepted 16 April 2023. Date of publication 2 May 2023; date of current version 1 June 2023. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Marco Felipe Duarte. This work was supported by the European Union's Horizon 2020 research and innovation programme under Grant Agreement 802735-ERC-DIFFOP. *(Corresponding author: Ronen Talmon.)*

The authors are with the Viterbi Faculty of Electrical and Computer Engineering, Technion—Israel Institute of Technology, Haifa 32000, Israel (e-mail: salmogl@campus.technion.ac.il; ronen@ef.technion.ac.il).

This article has supplementary downloadable material available at https://doi.org/10.1109/TSP.2023.3272159, provided by the authors.

Digital Object Identifier 10.1109/TSP.2023.3272159

sets of SPSD matrices, we aim to align the sets by matching their statistical moments and available labels. For this purpose, we derive the three geometric operations composing PA: centering, scaling, and rotation, by using the Riemannian framework of SPSD matrices presented in [19], [20]. Broadly, our proposed centering is based on Riemannian mean subtraction derived using parallel transport. The scaling is based on matching the Riemannian dispersion, which is carried out with respect to the geodesic path. Lastly, the rotation makes use of a small number of landmarks and is designed to match the first- and second-order statistical features of the sets. Conceptually, our work could be viewed as an extension of [2] from SPD geometry to SPSD geometry. Seemingly, the difference between SPD geometry and SPSD geometry is subtle and could be technically solved by an appropriate regularization, yet, it is in fact fundamental and involves a completely different geometry.

We demonstrate the usefulness of our approach in simulations and in an application to BCI involving EEG recordings. Recently, the spatial covariance matrix of EEG recordings has been shown to be an informative feature for BCI applications, leading to state-of-the-art performance [21], [22], [23], [24]. Here, we show that using our method to align sets of covariance matrices of EEG recordings acquired from different subjects is beneficial, facilitating improved BCI performance compared to other alignment methods.

Seemingly, one limitation of the proposed approach, when it is applied to aligning data through the alignment of the SPSD features of the data, is that every downstream task is also restricted to using these aligned SPSD features. We show that if the SPSD features are computed as the sample covariance matrices, then the centering and rotation steps of the proposed PA, which are derived following SPSD geometry considerations, can be applied directly in the original data space. Consequently, in this case, we are not restricted to using the SPSD features in downstream tasks following the proposed alignment, in contrast to other related algorithms for data alignment [2], [20], [25].

The remainder of the paper is organized as follows. In Section II, we present the relevant mathematical background. Section III presents the problem formulation. The proposed method is presented in Section IV. In Section V, we show how our method can be applied directly to high-dimensional data sets with a low-dimensional structure. In Section VI, we showcase the performance of the proposed algorithm in applications to simulated data and to real EEG recordings. Section VII concludes the paper.

II. Mathematical Background

We present a brief mathematical background on the three Riemannian manifolds considered in this paper. For more details, we refer the readers to [14], [19], [20], [26].

A. The Manifold of SPD Matrices

We denote the set of all $r \times r$ SPD matrices by

$$\mathcal{P}_r = \left\{ P \in \mathbb{R}^{r\times r} : P = P^T, P \succ 0 \right\}. \tag{1}$$

The tangent space $\mathcal{T}_P\mathcal{P}_r$ at a point $P \in \mathcal{P}_r$ is the set of all symmetric matrices

$$\mathcal{T}_P\mathcal{P}_r = \left\{ S \in \mathbb{R}^{r\times r} : S = S^T \right\}.$$

For $S_1, S_2 \in \mathcal{T}_P\mathcal{P}_r$, the affine invariant metric is defined by [13]

$$\langle S_1, S_2 \rangle_P = \left\langle P^{-\frac{1}{2}} S_1 P^{-\frac{1}{2}}, P^{-\frac{1}{2}} S_2 P^{-\frac{1}{2}} \right\rangle, \tag{2}$$

where $\langle A, B\rangle = \mathrm{Tr}\{A^T B\}$. The set $\mathcal{P}_r$ in (1) with the inner product (2) constitutes a Riemannian manifold. For $P_1, P_2 \in \mathcal{P}_r$, the map of P_2 to the tangent space $\mathcal{T}_{P_1}\mathcal{P}_r$ is given by the following explicit expression of the Logarithmic map

$$S = \mathrm{Log}_{P_1}(P_2) = P_1^{1/2} \log(P_1^{-1/2} P_2 P_1^{-1/2}) P_1^{1/2}. \tag{3}$$

Its inverse is given by the following explicit expression of the Exponential map

$$P_2 = \mathrm{Exp}_{P_1}(S) = P_1^{1/2} \exp(P_1^{-1/2} S P_1^{-1/2}) P_1^{1/2}. \tag{4}$$

The geodesic path from $P_1 \in \mathcal{P}_r$ to $P_2 \in \mathcal{P}_r$ is given by

$$\begin{aligned}\gamma_{P_1\to P_2}(t) &= \mathrm{Exp}_{P_1}\left(t \mathrm{Log}_{P_1}(P_2)\right) \\ &= P_1^{1/2} \left(P_1^{-1/2} P_2 P_1^{-1/2}\right)^t P_1^{1/2},\ t \in [0,1]. \end{aligned} \tag{5}$$

Intuitively, the logarithmic map could be interpreted as follows. If we move from a base point P_1 with initial velocity $\mathrm{Log}_{P_1}(P_2)$ and maintain constant speed along the geodesic, then after one time step, we arrive at P_2. The square of the arc length of the geodesic path in (5) is given by

$$\begin{aligned} d^2_{\mathcal{P}}(P_1, P_2) &= \left\| \log\left(P_1^{-1/2} P_2 P_1^{-1/2}\right) \right\|_F^2 \\ &= \sum_{i=1}^{r} \log^2(\lambda_i), \end{aligned} \tag{6}$$

where λ_i are the eigenvalues of $P_1^{-1} P_2$. The arc length in (6) defines an affine-invariant distance, to which we will refer as the Riemannian distance in $\mathcal{P}_r$. We note that the affine-invariant property of this distance can be explicitly written by $d^2_{\mathcal{P}}(P_1, P_2) = d^2_{\mathcal{P}}(A^T P_1 A, A^T P_2 A)$ for every invertible matrix A.

In the context of PA, we will consider the Fréchet mean, the dispersion, and the following transport map.

Definition 1 (Fréchet mean and dispersion):

1) The *Fréchet mean* (first order moment) of a set $\{x_i \in \mathcal{M}\}_{i=1}^N$ is defined by

$$\overline{x} = M\left(\{x_i\}\right) := \arg\min_{x\in\mathcal{M}} \sum_{i=1}^{N} d^2_R\left(x, x_i\right), \tag{7}$$

where d_R is the Riemannian distance on $\mathcal{M}$.

2) The *dispersion* (second order moment) of a set $\{x_i \in \mathcal{M}\}_{i=1}^N$ is defined by

$$D(\{x_i\}) = \frac{1}{N-1} \sum_{i=1}^{N} d^2_R(\overline{x}, x_i). \tag{8}$$

Definition 2 (Isometric Transport Map [20]): Let $\{x_i \in \mathcal{M}\}_{i=1}^N$ be a set of points on a Riemannian manifold $\mathcal{M}$ with

mean $M(\{x_i\}) = \overline{x}$. We call the map $\widetilde{\Gamma}^{+}_{\overline{x}\to x_0} : \mathcal{M} \to \mathcal{M}$ of the set $\{x_i\}$ to the point $x_0 \in \mathcal{M}$ an isometric transport map if:

1) It satisfies: $M(\{\widetilde{\Gamma}^{+}_{\overline{x}\to x_0}(x_i)\}_{i=1}^{N}) = x_0$.
2) It is an isometry, i.e., it preserves pairwise distances:

$$d_{\mathcal{M}}(x_i, x_j) = d_{\mathcal{M}}\left(\widetilde{\Gamma}^{+}_{\overline{x}\to x_0}(x_i), \widetilde{\Gamma}^{+}_{\overline{x}\to x_0}(x_j)\right).$$

Note that these definitions of the Fréchet mean, the dispersion, and the isometric transport map are not specific to the SPD manifold, and they will be used in the context of other manifolds as well.

Let $\{P_i \in \mathcal{P}_r\}$ be a set with mean $M(\{P_i\}) = \overline{P}$. Yair et al. showed in [27] that the map

$$\widetilde{\Gamma}^{+}_{\overline{P}\to P_0}(P_i) = T_p P_i T_p^T, \tag{9}$$

where $T_p = (P_0 \overline{P}^{-1})^{1/2}$, is an isometric transport map on $\mathcal{P}_r$. The map in (9) is tightly related to Parallel Transport (PT) with one important distinction: it maps points from the manifold to the manifold, while PT maps points from tangent space to tangent space. Broadly, PT is the process of transporting a vector along a curve of the manifold such that the covariant derivative of the transported vector along the curve is zero (see [28] for more details). The map in (9) is equivalent to mapping $P_i \in \mathcal{P}_r$ to the tangent space at $\overline{P}$, parallel transporting the mapped vector along the geodesic path from $\overline{P}$ to P_0, and mapping the result back to manifold $\mathcal{P}_r$.

B. The Grassmann Manifold

Let $\mathcal{V}_{d,r}$ be the set of all $d \times r$ matrices, $r < d$, whose r columns are orthonormal vectors in $\mathbb{R}^d$. Let $\mathcal{O}_r$ be the set of all $r \times r$ orthogonal matrices, such that any $O \in \mathcal{O}_r$ satisfies $OO^T = O^T O = I$. Following [26], we view the Grassmann manifold as the quotient space $\mathcal{G}_{d,r} = \mathcal{V}_{d,r}/\mathcal{O}_r$. A point on $\mathcal{G}_{d,r}$ is an r-dimensional subspace of $\mathbb{R}^d$, and it is represented by an equivalence class

$$[G] = \{GO : O \in \mathcal{O}_r\},$$

where $G \in \mathcal{V}_{d,r}$. Let $G_\perp \in \mathbb{R}^{d\times(d-r)}$ be the orthogonal complement of G, i.e., $[G \quad G_\perp] \in \mathcal{O}_d$. The tangent space at $[G] \in \mathcal{G}_{d,r}$ is given by

$$\mathcal{T}_G\mathcal{G}_{d,r} = \left\{\Delta = G_\perp B \,\middle|\, B \in \mathbb{R}^{(d-r)\times r}\right\}.$$

We consider the inner product proposed in [26]

$$\langle \Delta_1, \Delta_2 \rangle_G = \mathrm{Tr}\left\{\Delta_1^T \Delta_2\right\} = \mathrm{Tr}\{B_1^T B_2\},$$

where $\mathcal{T}_{[G]}\mathcal{G}_{d,r} \ni \Delta_1 = G_\perp B_1$ and $\mathcal{T}_{[G]}\mathcal{G}_{d,r} \ni \Delta_2 = G_\perp B_2$. Note that $G_\perp$ and B are not uniquely identified, but any choice results in the same inner product. The geodesic path from $[G_1] \in \mathcal{G}_{d,r}$ to $[G_2] \in \mathcal{G}_{d,r}$ is given by

$$\gamma_{G_1\to G_2}(t) = \mathrm{Exp}_{G_1}\left(t\mathrm{Log}_{G_1}(G_2)\right), \qquad t \in [0,1]. \tag{10}$$

where Log and Exp denote the Logarithmic and Exponential maps on the Grassmann manifold (for their explicit expressions, see [20]). The arc length of the geodesic path in (10) specifies a Riemannian distance. The square of the arc length is given by

$$d_{\mathcal{G}}^2(G_1, G_2) = \|\Theta\|_F, \tag{11}$$

where Θ represents the angles between the subspaces and can be computed by the following Singular Value Decomposition (SVD) $G_1^T G_2 = O_1(\cos\Theta)O_2^T$.

Let $\{G_i \in \mathcal{G}_{d,r}\}$ be a set with mean $M(\{G_i\}) = [\overline{G}]$. In [20], it was shown that the following map is an isometric transport map on $\mathcal{G}_{d,r}$

$$\widetilde{\Gamma}^{+}_{\overline{G}\to G_0}(G_i) = Q_0 \overline{Q}^T G_i, \tag{12}$$

where $\overline{Q} = [\overline{G} \quad \overline{G}_\perp]$, $\overline{G}_\perp$ is the orthogonal complement of $\overline{G}$, $Q_0 = \overline{Q}\exp\left(\begin{bmatrix}\mathbf{0} & -B^T\\ B & \mathbf{0}\end{bmatrix}\right)$, and $B = \overline{G}_\perp^T \mathrm{Log}_{\overline{G}}(G_0)$. For simplicity, we denote the isometric transport map on $\mathcal{G}_{d,r}$ by $T_g = Q_0 \overline{Q}^T$.

C. The Manifold of SPSD Matrices

We denote the set of all $d \times d$ SPSD matrices with a fixed rank $r < d$ by

$$\mathcal{S}_{d,r}^{+} = \left\{C \in \mathbb{R}^{d\times d} : C = C^T, C \succeq 0, \mathrm{rank}(C) = r\right\}.$$

We follow the work of Bonnabel and Sepulchre [19], which considers the following quotient manifold representation

$$\mathcal{S}_{d,r}^{+} \cong (\mathcal{V}_{d,r} \times \mathcal{P}_r)/\mathcal{O}_r.$$

Since any $C \in \mathcal{S}_{d,r}^{+}$ can be decomposed as

$$C = URU^T,$$

where $U \in \mathcal{V}_{d,r}$ and $R \in \mathcal{P}_r$, C can be represented by

$$C \cong (U, R), \tag{13}$$

where $\mathcal{V}_{d,r} \times \mathcal{P}_r$ is termed the structure space representation [19].

The representation in the structure space (13) is not unique because any orthogonal matrix $O \in \mathcal{O}_r$ satisfies

$$C = URU^T = (UO)\left(O^T RO\right)(UO)^T,$$

and therefore

$$C \cong \left(UO, O^T RO\right),$$

so that U in (13) is considered as a point on the Grassmann manifold $\mathcal{G}_{d,r}$.

The fact that the structure space is not unique makes the practical use of a set of SPSD matrices challenging. Recently in [20], a canonical representation of a set of SPSD matrices in the structure space was proposed, such that every matrix in the set has a unique structure space representation, and it was shown to be useful when SPSD matrices are considered as data features. Given a set of SPSD matrices $\mathcal{C} = \{C_i\}$, we denote the canonical structure space representation by

$$C_i \cong (G_i, P_i), \tag{14}$$

where the computation of (G_i, P_i) is given in Algorithm 1.

By [19, Thm. 1], the space $\mathcal{S}_{d,r}^{+}$ equipped with the following metric

$$\langle(\Delta_1, S_1), (\Delta_2, S_2)\rangle_{(U,R)} = \langle\Delta_1, \Delta_2\rangle_U + k\,\langle S_1, S_2\rangle_R \quad (15)$$

which is the sum of the metrics in $\mathcal{G}_{d,r}$ and $\mathcal{P}_r$ with some parameter $k > 0$, is a Riemannian manifold with horizontal space

$$\mathcal{T}_{(U,R)}\mathcal{S}_{d,r}^{+} = \{(\Delta, S) : \Delta \in \mathcal{T}_U\mathcal{G}_{d,r}, S \in \mathcal{T}_R\mathcal{P}_r\}. \quad (16)$$

We note that there exist in the literature other metrics for $\mathcal{S}_{d,r}^{+}$, e.g., those presented in [29], [30]. We choose this particular representation and its associated metric for three reasons. First, it provides a metric, which generalizes the Affine Invariant Riemannian Metric (AIRM) for SPD matrices (2), since it is invariant to orthogonal transformations, scaling, and pseudoinversion [19]. Second, it was shown in [20] that this metric provides better empirical results compared to the metric proposed in [30]. Third, this geometry enables the computation of essential components, such as the Fréchet mean, and as far as we know, there is no algorithm for computing the Fréchet mean using the geometry proposed in [29].

In [19], a special path of interest between two points in $\mathcal{S}_{d,r}^{+}$, $C_1 \cong (U_1, R_1)$ and $C_2 \cong (U_2, R_2)$, was proposed. This path is based on a horizontal geodesic in the structure space, and therefore, we first find two representatives of C_1 and C_2 in the structure space that are connected by horizontal geodesics. Define the rotation of U_2 with respect to U_1 by

$$\widetilde{U}_2 = \Pi_{U_1}(U_2) := U_2 O_2 O_1^T, \quad (17)$$

where $U_1^T U_2 = O_1 \Sigma O_2^T$ is an SVD. The matrix C_2 can be represented also by $C_2 \cong (\widetilde{U}_2, \widetilde{R}_2)$, where $\widetilde{R}_2 = \widetilde{U}_2^T C_2 \widetilde{U}_2$. Then, by [19, Thm. 2], the path between C_1 and C_2, given by

$$\widehat{\gamma}_{C_1 \to C_2}(t) = U(t) R(t) U(t)^T, \quad t \in [0, 1], \quad (18)$$

where $U(t) = \gamma_{U_1 \to \widetilde{U}_2}(t)$ and $R(t) = \gamma_{R_1 \to \widetilde{R}_2}(t)$, admits the following properties. First, it connects C_1 and C_2, i.e., $\widehat{\gamma}_{C_1 \to C_2}(0) = C_1$ and $\widehat{\gamma}_{C_1 \to C_2}(1) = C_2$, and $\widehat{\gamma}_{C_1 \to C_2}(t) \in \mathcal{S}_{d,r}^{+}$ for all $t \in [0, 1]$. Second, the path $(U(t), R(t))$ is a horizontal lift of $\widehat{\gamma}_{C_1 \to C_2}(t)$ and it is a geodesic in the structure space. Third, the squared total length of $\widehat{\gamma}_{C_1 \to C_2}(t)$ is given by

$$l^2\left(\widehat{\gamma}_{C_1 \to C_2}\right) = d_{\mathcal{G}}^2\left(U_1, \widetilde{U}_2\right) + k d_{\mathcal{P}}^2\left(R_1, \widetilde{R}_2\right), \quad k > 0. \quad (19)$$

The path $\widehat{\gamma}_{C_1 \to C_2}(t)$ is not necessarily a geodesic path in the Riemannian manifold $\mathcal{S}_{d,r}^{+}$ with the metric in (15). In addition, its length in (19) is not a distance since it does not satisfy the triangle inequality. Nevertheless, it was shown theoretically in [19] and empirically in [20] that the length in (19) is a meaningful measure of dissimilarity.

We consider a map of $C_2 \cong (\widetilde{U}_2, \widetilde{R}_2)$ to the horizontal space $\mathcal{T}_{(U_1,R_1)}\mathcal{S}_{d,r}^{+}$ by the corresponding logarithmic maps in the structure space as follows

$$\widehat{\mathrm{L}}_{(U_1,R_1)}(\widetilde{U}_2, \widetilde{R}_2) = \left(\mathrm{Log}_{U_1}\left(\widetilde{U}_2\right), \mathrm{Log}_{R_1}\left(\widetilde{R}_2\right)\right). \quad (20)$$

Informally, in the remainder of this paper, we view the horizontal space in (16) as the tangent space, the path $\widehat{\gamma}_{C_1 \to C_2}(t)$ in (18) as an approximate geodesic path, and the map in (20) as an approximate logarithmic map in $\mathcal{S}_{d,r}^{+}$.

Algorithm 1: Canonical Representation for a Set of SPSD Matrices [20].

Input: $\mathcal{C} = \{C_i \in \mathcal{S}_{d,r}^{+}\}_{i=1}^{N}$
Output: A canonical representation $\mathcal{C} = \{C_i \cong (G_i, P_i)\}_{i=1}^{N}$
1: **for** $i = 1$ to N **do**
2: Compute $U_i \in \mathbb{R}^{d \times r}$ the r first eigenvectors of C_i
3: **end for**
4: Compute the Grassman mean $\overline{U} = M(\{U_i\}_{i=1}^{N})$
5: **for** $i = 1$ to N **do**
6: Rotate U_i: $G_i = \Pi_{\overline{U}}(U_i)$ ▷ see (17)
7: Set $P_i = G_i^T C_i G_i$
8: **end for**
9: **return** $\{(G_i, P_i)\}_{i=1}^{N}$

III. Problem Formulation

Consider two data sets: the source set

$$\mathcal{X} = \{X_i\}_{i=1}^{N_x},$$

and the target set

$$\mathcal{Y} = \{Y_i\}_{i=1}^{N_y},$$

where X_i and Y_i are $d \times d$ SPSD matrices with rank $r < d$, which are viewed as data points. Suppose that each set consists of L classes. We consider a semi-supervised setting, where the labels of only a small subset $\mathcal{X}_l \subset \mathcal{X}$ and the labels of only a small subset $\mathcal{Y}_l \subset \mathcal{Y}$ are known, and they are used for the proposed data alignment. We assume that the statistical distributions of $\mathcal{X}$ and $\mathcal{Y}$ are different due to various extrinsic factors, yet the sets are intrinsically related.

To formulate our goal, we follow [2] and parametrize the statistical distributions of $\mathcal{X}$ and $\mathcal{Y}$ using their first- and second-order moments, which are defined in the Riemannian sense as follows.

Let $\mathcal{M}$ be the SPSD manifold. Since the length in (19) is not a distance, the Fréchet mean in (7) cannot be used as is. Instead, we use the mean proposed in [31] which is computed as follows. Let $C_i \cong (G_i, P_i)$ be the canonical representation obtained by applying Algorithm 1 to a set of SPSD matrices $\mathcal{C} = \{C_i\}_{i=1}^{N}$. The mean of $\mathcal{C}$ is given by

$$\overline{C} \cong (\overline{G}, \overline{P}), \quad (21)$$

where $\overline{G} = M(\{G_i\})$ and $\overline{P} = M(\{P_i\})$.

For computing the dispersion of $\mathcal{C}$, we use (19) as a measure of dissimilarity between $\overline{C}$ and C_i instead of d_R in (22) because d_R is not defined, giving rise to the following alternative dispersion

$$\sigma^2 = \frac{1}{N-1}\sum_{i=1}^{N} l^2\left(\widehat{\gamma}_{\overline{C} \to C_i}\right). \quad (22)$$

Now, equipped with the mean and dispersion of a set, we parametrize the statistical distributions of $\mathcal{X}$ and $\mathcal{Y}$ as follows

$$\Omega_{\mathcal{X}} = \left\{\overline{X}, M_1^x, M_2^x, \ldots, M_L^x, \sigma_x\right\}$$
$$\Omega_{\mathcal{Y}} = \left\{\overline{Y}, M_1^y, M_2^y, \ldots, M_L^y, \sigma_y\right\},$$

where $\overline{X}$ and $\overline{Y}$ are the means of $\mathcal{X}$ and $\mathcal{Y}$, respectively, M_i^x and M_i^y are the means of the i_{th} class in the sets $\mathcal{X}$ and $\mathcal{Y}$, respectively, and σ_x and σ_y are the dispersions of $\mathcal{X}$ and $\mathcal{Y}$, respectively.

Our goal is to align $\mathcal{X}$ and $\mathcal{Y}$ such that the above parametrization of their statistical distributions coincide. After alignment, a model learned from one set, say $\mathcal{X}$, can be exploited for an analysis or learning task applied to the other set, say $\mathcal{Y}$, despite their heterogeneity. For example, let $f : \mathcal{X} \to \{1, 2, \ldots, L\}$ be a classification function defined on the set $\mathcal{X}$. A good alignment of $\mathcal{X}$ and $\mathcal{Y}$ facilitates good classification results simply by applying f to points in $\mathcal{Y}$ after alignment.

IV. Proposed Method

Following standard PA [1], the proposed alignment of the two sets $\mathcal{X}$ and $\mathcal{Y}$ is composed of three operations on the manifold $\mathcal{S}_{d,r}^+$: *centering*, *scaling*, and *rotation*. In this section, we present their proposed implementation on $\mathcal{S}_{d,r}^+$.

A. Centering

We center the sets $\mathcal{X}$ and $\mathcal{Y}$ by subtracting their respective means using a transport map defined as follows. Let $\{C_i \cong (G_i, P_i)\}$ be the canonical representation of a set with a mean $M(\{C_i\}) = \overline{C} \cong (\overline{G}, \overline{P})$, and let $C_0 \cong (G_0, P_0)$ be another point on $\mathcal{S}_{d,r}^+$ such that $G_0 = \Pi_{\overline{G}}(G_0)$ and $P_0 = G_0^T C_0 G_0$.

Definition 3 (Rigid Transport Map): We call the map $\Psi_{\overline{C}\to C_0} : \mathcal{S}_{d,r}^+ \to \mathcal{S}_{d,r}^+$ a Rigid transport map if the following two conditions hold.

1) It satisfies: $M(\Psi_{\overline{C}\to C_0}(\{C_i\})) = C_0$
2) It preserve the length of the curve between $\overline{C}$ and C_i $\forall i$. i.e.:

$$l^2\left(\widehat{\gamma}_{\overline{C}\to C_i}\right) = l^2\left(\widehat{\gamma}_{C_0\to\Psi_{\overline{C}\to C_0}(C_i)}\right) \quad \forall i.$$

Note that a rigid transform map (Definition 3) differs from an isometric transform map (Definition 2) by requiring geodesic length rather than distance preservation.

We define the following map:

$$\widetilde{\Gamma}^+_{\overline{C}\to C_0}(C_i) \cong \left(\widetilde{\Gamma}^+_{\overline{G}\to G_0}(G_i), \widetilde{\Gamma}^+_{\overline{P}\to P_0}(P_i)\right) = (T_g G_i, T_p P_i T_p^T), \tag{23}$$

where the matrices T_g and T_p are given in Sections II-A and II-B, respectively.

Proposition 1: The map $\widetilde{\Gamma}^+_{\overline{C}\to C_0}$ is a rigid transport map.

See Appendix A for the proof of Proposition 1.

Let $C_0 = \mathrm{diag}([\mathbf{1}_r \quad \mathbf{0}_{d-r}])$ denote the origin of the space. We center $\mathcal{X}$ and $\mathcal{Y}$ by applying $\widetilde{\Gamma}^+_{\overline{C}\to C_0}$

$$X_i^{(\mathrm{ctr})} = \widetilde{\Gamma}^+_{\overline{X}\to C_0}(X_i),\ \forall X_i \in \mathcal{X}$$
$$Y_j^{(\mathrm{ctr})} = \widetilde{\Gamma}^+_{\overline{Y}\to C_0}(Y_j),\ \forall Y_i \in \mathcal{Y}. \tag{24}$$

We denote the sets after centering by

$$\mathcal{X}^{(\mathrm{ctr})} = \left\{X_i^{(\mathrm{ctr})} \cong \left(G_i^{(\mathrm{ctr})}, P_i^{(\mathrm{ctr})}\right)\right\}$$
$$\mathcal{Y}^{(\mathrm{ctr})} = \left\{Y_j^{(\mathrm{ctr})} \cong \left(V_j^{(\mathrm{ctr})}, R_j^{(\mathrm{ctr})}\right)\right\}, \tag{25}$$

whose means according to Proposition 1 are C_0.

B. Scaling

We propose to scale the centered source set $\mathcal{X}^{(\mathrm{ctr})}$ such that its dispersion after scaling matches the dispersion of the centered target set $\mathcal{Y}^{(\mathrm{ctr})}$, namely σ_y. This operation is carried out by sampling the geodesic path (18) between the origin C_0 and $X_i^{(\mathrm{ctr})} \cong (G_i^{(\mathrm{ctr})}, P_i^{(\mathrm{ctr})})$. Let (G_0, P_0) be the structure space representation of C_0, and let $\widehat{\gamma}_{C_0\to X_i^{(\mathrm{ctr})}}(t) \cong (G_i(t), P_i(t))$ be the geodesic path between C_0 and $X_i^{(\mathrm{ctr})}$, whose length is given by

$$l^2\left(\widehat{\gamma}_{C_0\to X_i^{(\mathrm{ctr})}}\right) = d_{\mathcal{G}}^2\left(G_0, G_i^{(\mathrm{ctr})}\right) + k d_{\mathcal{P}}^2\left(P_0, P_i^{(\mathrm{ctr})}\right). \tag{26}$$

Consider the structure space representation of $\mathcal{X}^{(\mathrm{ctr})}$ and $\mathcal{Y}^{(\mathrm{ctr})}$ defined in (25). Let σ_g and σ_v be the dispersion of $\{G_i^{(\mathrm{ctr})}\}$ and $\{V_i^{(\mathrm{ctr})}\}$ on $\mathcal{G}_{d,r}$, respectively, and let σ_p and σ_r be the dispersion of $\{P_i^{(\mathrm{ctr})}\}$ and $\{R_i^{(\mathrm{ctr})}\}$ on $\mathcal{P}_r$, respectively. We define the scaling of $X_i^{(\mathrm{ctr})}$ by

$$X_i^{(\mathrm{scl})} \cong \left(G_i^{(\mathrm{scl})}, P_i^{(\mathrm{scl})}\right) = \left(G_i\left(t = \frac{\sigma_v}{\sigma_g}\right), P_i\left(t = \frac{\sigma_r}{\sigma_p}\right)\right),$$

such that the scaled source set is given by $\mathcal{X}^{(\mathrm{scl})} = \{X_i^{(\mathrm{scl})}\}$. The dispersion after scaling is given by

$$\sigma^2\left(\mathcal{X}^{(\mathrm{scl})}\right) = \sum_{i=1}^{N_x} l^2\left(\widehat{\gamma}_{C_0\to X_i^{(\mathrm{scl})}}\right).$$

Proposition 2: Then the dispersion of the set $\mathcal{X}^{(\mathrm{scl})}$ is equal to the dispersion of the set $\mathcal{Y}^{(\mathrm{ctr})}$, i.e.,

$$\sigma^2\left(\mathcal{X}^{(\mathrm{scl})}\right) = \sigma_y^2.$$

See Appendix B for the proof of Proposition 2.

C. Rotation

According to the problem formulation presented in Section III, the sets $\mathcal{X}$ and $\mathcal{Y}$ are composed of L classes. In this step, we propose to match the means of the classes such that the classes in both sets coincide.

Definition 4 (Rotation): Let $\{C_i \cong (G_i, P_i)\} \subset \mathcal{S}_{d,r}^+$ be the canonical representation of a set with a mean $M(\{C_i\}) = \overline{C}$. We

call a map $R : \mathcal{S}^+_{d,r} \to \mathcal{S}^+_{d,r}$ a rotation about $\overline{C}$ if the following properties are satisfied:

1) R preserves the length of the path between all pairs of points:
$$l^2\left(\widehat{\gamma}_{C_i \to C_j}\right) = l^2\left(\widehat{\gamma}_{R(C_i)\to R(C_j)}\right), \quad \forall i, j.$$
2) R maps $\overline{C}$ to itself, i.e., $R(\overline{C}) = \overline{C}$.
3) R preserves the length of the path to $\overline{C}$:
$$l^2\left(\widehat{\gamma}_{C_i \to \overline{C}}\right) = l^2\left(\widehat{\gamma}_{R(C_i)\to \overline{C}}\right), \quad \forall i.$$

Let $\mathrm{SO}(d)$ be the set of all special orthogonal $d \times d$ matrices, i.e., orthogonal matrices with determinant 1. The following map is a rotation about the origin C_0 for any $C \cong (G, P)$:
$$R(C) \cong (O_g G, O_p P O_p^T), \tag{27}$$
where $O_p \in \mathrm{SO}(r)$, and $O_g \in \mathrm{SO}(d)$ is of the form
$$O_g = \begin{bmatrix} O_{g_1} & \mathbf{0} \\ \mathbf{0} & O_{g_2} \end{bmatrix},$$
where $O_{g_1} \in \mathrm{SO}(r)$ and $O_{g_2} \in \mathrm{SO}(d-r)$. In the Supplementary Materials, we show that the map in (27) indeed satisfies all the three properties of Definition 4.

To find the best R, which describes the relation between $\mathcal{X}^{(\mathrm{scl})}$ and $\mathcal{Y}^{(\mathrm{ctr})}$, we search for the rotation which matches the classes means of $\mathcal{X}^{(\mathrm{scl})}$ to those of $\mathcal{Y}^{(\mathrm{ctr})}$ by minimizing the following criterion
$$\min_R \sum_{c=1}^{L} l^2(\widehat{\gamma}_{R(M_c^x)\to M_c^y}), \tag{28}$$
where $M_c^x \cong (G_c^x, P_c^x)$ and $M_c^y \cong (G_c^y, P_c^y)$ are the mean of the c-th class of $\mathcal{X}^{(\mathrm{scl})}$ and $\mathcal{Y}^{(\mathrm{ctr})}$, respectively. We note that in contrast to the previous steps (centering and scaling), which are completely label-free, this step requires subsets of $\mathcal{X}$ and $\mathcal{Y}$ that are sufficiently large and allow for accurate estimations of M_c^x and M_c^y, respectively. Alternatively, an unsupervised rotation by matching the second-order moments of the data sets as in [5], [27], can be considered.

By using the arc length in (19) and the form of R in (27), the criterion in (28) can be decoupled to two minimization problems
$$\min_{O_g \in \mathrm{SO}(d)} \sum_{c=1}^{L} d^2_{\mathcal{G}}(G_c^y, O_g G_c^x) \tag{29}$$
$$\min_{O_p \in \mathrm{SO}(r)} \sum_{c=1}^{L} d^2_{\mathcal{P}}(P_c^y, O_p P_c^x O_p^T). \tag{30}$$
We solve (29) and (30) by using the steepest descent algorithm on the Riemannian manifold of all rotation matrices, which was implemented in [32]. This algorithm computes the Riemannian gradient by mapping the Euclidean gradient to the tangent space of the manifold. The Euclidean gradient of (30) is given in [2] by
$$\nabla_{O_p} \sum_{c=1}^{L} d^2_{\mathcal{P}}(P_c^y, O_p P_c^x O_p^T) = 4 \sum_{c=1}^{L} \log\left(\left(P_c^y\right)^{-1} O_p P_c^x O_p^T\right) O_p. \tag{31}$$
For the Euclidean gradient computation of (29), see Supplementary Materials. We note that the convergence of the steepest descent algorithm implemented in [32] to a global minimum is not guaranteed. However, we will show experimentally that it performs well on both simulated and real problems.

We note that the matrices O_g and O_p, as well as T_g and T_p, are invertible matrices. This gives rise to the following insight. Suppose $\mathcal{X}$ and $\mathcal{Y}$ are represented canonically in the structure space and have the same dispersion, i.e. $\sigma_x = \sigma_y$, such that the discrepancies between the statistics of $\mathcal{X}$ and $\mathcal{Y}$ are indeed expressed by transportations, $\widetilde{T}_g$ and $\widetilde{T}_p$, and rotations, $\widetilde{O}_g$ and $\widetilde{O}_p$. Consequently, the centering step in Section IV-A and the rotation step in this section, can be viewed as the estimation of the inverse of $\widetilde{T}_g$, $\widetilde{T}_p$, $\widetilde{O}_g$ and $\widetilde{O}_p$, and the proposed PA can potentially match the statistical distributions.

V. DA in the Original Data Space

In this work, we consider the SPSD matrices as data features. Despite being quite broad (e.g., low-rank covariance matrices, graph Laplacians, kernel and similarity matrices), the focus on SPSD matrices applies not only to the current context of data alignment, but also to every subsequent downstream task that is restricted to working with particular SPSD features.

As a remedy, in this section, we show that the *centering* and *rotation* steps can be applied equivalently in the original data space rather than to the SPSD sample covariance matrices. We remark that we did not find such an equivalent implementation of the *scaling* step.

Consider two sets of data sets: $\mathcal{D}_x = \{D_{x_i} \in \mathbb{R}^{d \times n_{x_i}}\}_{i=1}^{N_x}$ and $\mathcal{D}_y = \{D_{y_i} \in \mathbb{R}^{d \times n_{y_i}}\}_{i=1}^{N_y}$, where the columns of D_{x_i} and D_{y_i} are samples in $\mathbb{R}^d$ with zero mean. The sample covariance matrices of D_{x_i} and D_{y_i} are given by
$$\Sigma_{X_i} = \frac{1}{n_{x_i} - 1} D_{x_i} D_{x_i}^T$$
$$\Sigma_{Y_i} = \frac{1}{n_{y_i} - 1} D_{y_i} D_{y_i}^T,$$
respectively. We assume that $\Sigma_{X_i}, \Sigma_{Y_i} \in \mathcal{S}^+_{d,r}$, and denote the sets of the sample covariance matrices computed from $\mathcal{D}_x$ and $\mathcal{D}_y$, by $\mathcal{X} = \{\Sigma_{X_i}\}_{i=1}^{N_x}$ and $\mathcal{Y} = \{\Sigma_{Y_j}\}_{j=1}^{N_y}$. Let (G_i, P_i) be the canonical representation of Σ_{X_i}, obtained by applying Algorithm 1 to the set $\mathcal{X}$. To apply alignment directly to $\mathcal{D}_x$ we first compute the centering matrices of the set $\mathcal{X}$, T_{p_x} and T_{g_x}, and the centering matrices of the set $\mathcal{Y}$, T_{p_y} and T_{g_y}, as described in Section IV-A. Second, we compute the rotation matrices O_g and O_p as described in Section IV-C. Then, we apply three consecutive steps: centering, rotation, and transportation to the mean of $\mathcal{Y}$, which are given by
$$\widehat{\Sigma}_{X_i} := \widetilde{\Gamma}^+_{C_0 \to \overline{Y}}\left(R\left(\widetilde{\Gamma}^+_{\overline{X} \to C_0}\left(\Sigma_{X_i}\right)\right)\right), \tag{32}$$
where $\overline{X}$ and $\overline{Y}$ are the means of $\mathcal{X}$ and $\mathcal{Y}$, respectively.

Proposition 3: The transformation of D_{x_i}
$$\widehat{D}_{x_i} = \left(T_{g_y}^{-1} O_g T_{g_x}\right) G_i \left(T_{p_y}^{-1} O_p T_{p_x}\right) G_i^T D_{x_i} \tag{33}$$

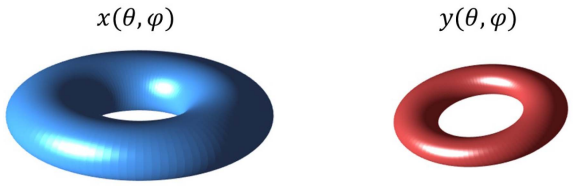

Fig. 1. Two tori $x(\theta,\varphi)$ and $y(\theta,\varphi)$ defined by the parametrization given in (34).

satisfies

$$\frac{1}{n_{x_i}-1}\widehat{D}_{x_i}\widehat{D}_{x_i}^T=\widehat{\Sigma}_{X_i}.$$

The proof appears in Appendix E.

Proposition 3 implies that when we apply the transformation in (33) to D_{x_i}, the relation between the sample covariance of the data before and after the transformation is given by centering, rotation about the origin, and transportation to the mean $\overline{Y}$. Consequently, after the transformation in (33), the covariance matrices characterizing the data from both sets $\mathcal{D}_x$ and $\mathcal{D}_y$ are aligned.

The implication of Proposition 3 is that we are not restricted to using SPSD features in downstream tasks following the proposed alignment, in contrast to other related algorithms for data alignment, such as [2], [20], [25]. Section VI-B illustrates the application of this result.

VI. Experimental Results

A. Toy Problem – Alignment

Consider two tori with the following parametrization

$$x(\theta,\varphi)=\begin{bmatrix}(4+2\cos(\theta))\cos(\varphi)\\(4+2\cos(\theta))\sin(\varphi)\\2\sin(\theta)\end{bmatrix}$$

$$y(\theta,\varphi)=O\begin{bmatrix}(3+\cos(\theta))\cos(\varphi)\\(3+\cos(\theta))\sin(\varphi)\\\sin(\theta)\end{bmatrix}, \tag{34}$$

where $O\in\mathrm{SO}(3)$ is picked arbitrarily and given by

$$O=\begin{bmatrix}0.7651 & -0.5426 & 0.3468\\0.4314 & 0.8317 & 0.3495\\-0.4781 & -0.1177 & 0.8704\end{bmatrix}.$$

Fig. 1 presents both tori $x(\theta,\varphi)$ and $y(\theta,\varphi)$, where it can be seen that $y(\theta,\varphi)$ is a scaled and rotated version of $x(\theta,\varphi)$.

For each torus, we generate a set of covariance matrices as follows. Let $[\theta,\varphi]^T$ be a random variable with a uniform distribution $U[0,\pi/2]\times[0,\pi/2]$. Each realization $[\varphi_i,\theta_i]^T$ of $[\varphi,\theta]^T$ defines a point on the torus. We collect $N=500$ points from $x(\theta,\varphi)$: $\{x_i=x(\theta_i,\varphi_i)\}_{i=1}^{500}$. For each point x_i, we compute the sample covariance

$$X_i=\frac{1}{|\mathcal{N}_i|-1}\sum_{n_{i_j}\in\mathcal{N}_i}\left(n_{i_j}-x_i\right)\left(n_{i_j}-x_i\right)^T, \tag{35}$$

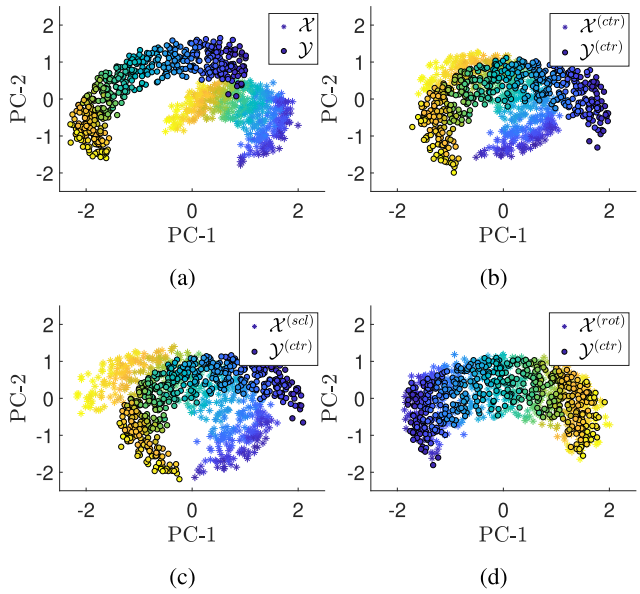

Fig. 2. The PC of the covariance matrices after mapping to the tangent space of $\mathcal{S}_{3,2}^+$: (a) $\mathcal{X}$ and $\mathcal{Y}$ before alignment, (b) $\mathcal{X}^{(\mathrm{ctr})}$ and $\mathcal{Y}^{(\mathrm{ctr})}$ after centering, (c) $\mathcal{X}^{(\mathrm{scl})}$ and $\mathcal{Y}^{(\mathrm{ctr})}$ after scaling and (d) $\mathcal{X}^{(\mathrm{rot})}$ and $\mathcal{Y}^{(\mathrm{ctr})}$ after rotation. Points are colored according to their angle φ.

where $n_{i_j}=x(\theta_{i_j},\varphi_{i_j})\in\mathcal{N}_i$ and $[\theta_{i_j},\varphi_{i_j}]^T$ is a realization of a Gaussian variable centered around $[\theta_i,\varphi_i]^T$ with a covariance $\Sigma=10^{-4}I$.

The above procedure is repeated for the other torus $y(\theta,\varphi)$, resulting in another set of $N=500$ points $[\varphi_i,\theta_i]^T$ with their associated covariance matrices Y_i. The obtained two sets of covariance matrices are denoted by $\mathcal{X}=\{X_i\}_{i=1}^{500}$ and $\mathcal{Y}=\{Y_i\}_{i=1}^{500}$. Since the considered torus is a two-dimensional surface embedded in $\mathbb{R}^3$, the sample covariance matrices X_i and Y_i computed based on a small neighborhood are approximately in $\mathcal{S}_{3,2}^+$.

Our goal in this context could be stated as follows. We wish to align $\mathcal{X}$ and $\mathcal{Y}$, such that the covariance matrices X_i and Y_j of any two close points in the (hidden) parameter space $[\theta_i,\varphi_i]^T$ and $[\theta_j,\varphi_j]^T$, where each lies on a different torus, will be close.

We apply our method and get a set of covariance matrices after each of the three steps: centering, scaling, and rotation. For the rotation step, we divide the covariance matrices into four classes according to their respective angles θ_i and φ_i. Each class corresponds to one of the four combinations $\theta_i\lessgtr\pi/4$ and $\varphi_i\lessgtr\pi/4$. We use 50 labels from the target set $\mathcal{Y}$, which is 10% of the set size.

To visualize the resulting sets after adaptation, we use the approximate logarithmic map in (20) to map the matrices to the tangent space. After mapping to the tangent space, the matrices are viewed as vectors in a linear space, where we can apply Principal Components Analysis (PCA) and display the Principal Components (PC).

Fig. 2 shows the PC before and after each of the steps of the proposed method: (a) $\mathcal{X}$ and $\mathcal{Y}$ before alignment, (b) $\mathcal{X}^{(\mathrm{ctr})}$ and $\mathcal{Y}^{(\mathrm{ctr})}$ after centering, (c) $\mathcal{X}^{(\mathrm{scl})}$ and $\mathcal{Y}^{(\mathrm{ctr})}$ after scaling, and

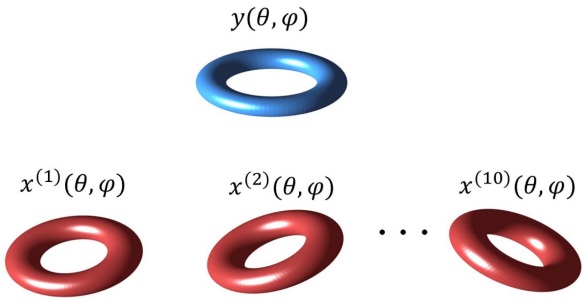

Fig. 3. The torus $y(\theta,\varphi)$ in blue and the rotated tori $x^{(j)}(\theta,\varphi)$, $j \in \{1,2,10\}$, in red.

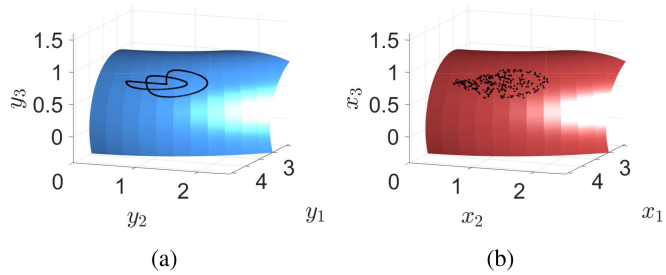

Fig. 4. Zoom in to the tori: (a) $y(\theta,\varphi)$ and (b) $x^{(1)}(\theta,\varphi)$. The black points are samples of the datasets D_{y_i} and $D_{x_i}^{(1)}$, which are defined on $y(\theta,\varphi)$ and $x^{(1)}(\theta,\varphi)$, respectively.

(d) $\mathcal{X}^{(\text{rot})}$ and $\mathcal{Y}^{(\text{ctr})}$ after rotation. Each point in Fig. 2 represents a covariance matrix, and it is colored according to its respective (hidden) angle φ. We observe that the proposed method indeed aligns $\mathcal{X}$ and $\mathcal{Y}$. Furthermore, the correspondence between the colors of the points from both sets (representing the angle φ) illustrates that the obtained alignment respects the intrinsic low-dimensional structure of the data (in this case, the hidden parametrization by φ and θ).

B. Toy Problem – Denoising in the Original Data Space

In Section V, we showed that the proposed alignment can be applied in the original feature space, in which the data is given, thereby not restricting downstream tasks to the use of SPSD features. In this section, we demonstrate this capability using a subsequent denoising task following the alignment.

Consider the following (reference) torus

$$y(\theta,\varphi) = \begin{bmatrix} (3+\cos(\theta))\cos(\varphi) \\ (3+\cos(\theta))\sin(\varphi) \\ \sin(\theta) \end{bmatrix}$$

and the following ten rotated tori

$$x^{(j)}(\theta,\varphi) = O^{(j)} y(\theta,\varphi),$$

where $O^{(j)} \in \text{SO}(3)$, $1 \le j \le 10$, are picked arbitrarily. Suppose each torus represents a different domain. Fig. 3 shows the reference torus $y(\theta,\varphi)$ in blue, and a few rotated tori $x^{(j)}(\theta,\varphi)$ in red.

A data set D_{y_i} in the domain of the torus $y(\theta,\varphi)$ is given by a set of samples from a signal defined on $y(\theta,\varphi)$:

$$D_{y_i} = \{y(\theta_i(t_n), \varphi_i(t_n))\}_{n=1}^{300},$$

where

$$\begin{bmatrix} \theta_i(t) \\ \varphi_i(t) \end{bmatrix} = \begin{bmatrix} \theta_i(0) + \frac{1}{10}(2t + \cos(5\pi t) + 0.2\cos(10\pi t)) \\ \varphi_i(0) + 0.3t\sin(5\pi t) \end{bmatrix},$$

and $[\theta_i(0), \varphi_i(0)]^T$ are sampled from a uniform grid in the square $[0, \pi/2] \times [0, \pi/2]$. Data sets $D_{x_i}^{(j)}$ in the domain of the tori $x^{(j)}(\theta,\varphi)$ are given by sets of samples from noisy signals defined on $x^{(j)}(\theta,\varphi)$ by

$$D_{x_i}^{(j)} = \left\{x^{(j)}(\theta_i(t_n) + \sigma_{\theta_i}(t_n), \varphi_i(t_n) + \sigma_{\varphi_i}(t_n))\right\}_{n=1}^{300},$$

where $\sigma_{\theta_i}(t_n)$ and $\varphi_{\theta_i}(t_n)$ are realizations of a Gaussian variable with zero mean and variance $\sigma^2 = 4 \cdot 10^{-4}$, such that the input Signal-to-Noise Ratio (SNR) (before denoising) is 26.2 dB. Fig. 4 presents examples for: (a) a set of samples from the clean signal D_{y_i}, and (b) a set of samples from the noisy signal $D_{x_i}^{(1)}$.

Following the notation in Section V, we denote $\mathcal{D}_y = \{D_{y_i}\}$ and $\mathcal{D}_x^{(j)} = \{D_{x_i}^{(j)}\}$. To attenuate the noise in $\mathcal{D}_x^{(j)}$, we first apply the transformation in (33) to adapt $D_{x_i}^{(j)}$ to $\mathcal{D}_{y_i}$ and get $\widehat{D}_{x_i}^{(j)}$, and then, we average over all the noisy (and adapted) signals: $\widetilde{D}_{x_i} = \frac{1}{10}\sum_{j=1}^{10} \widehat{D}_{x_i}^{(j)}$. To illustrate the importance of data alignment for denoising, we compare between the averaged signal without alignment and the averaged signal after alignment. Fig. 5 shows the denoising results with and without the proposed alignment. The clean signal $D_{x_i}(t)$ and the filtered signal $\widetilde{D}_{y_i}(t)$ are depicted by the blue curve and the red curve, respectively. Each row in the figure presents a different trajectory of $D_{y_i}(t)$ and $\widetilde{D}_{x_i}(t)$. The left and right columns show the denoising results without alignment and the denoising results by using the proposed alignment, respectively.

We observed that by using the proposed alignment prior to averaging (denoising), we obtain a better noise attenuation. Furthermore, when using alignment, the structure of the signal is preserved while the averaging of the signals $D_{x_i}^{(j)}(t)$ without a prior alignment hinders the structure of the signal. To quantitatively measure the denoising results, we compute the SNR of the resulting signal. The SNR obtained by the denoising after alignment is 42 dB, whereas the SNR obtained by the denoising without alignment is only 20.6 dB.

C. Application to BCI

One of the paradigms of BCI is the Steady State Visually Evoked Potential (SSVEP), which is based on the fact that when a user is looking at a flickering object at some frequency, this frequency manifests itself in the spectrum of the EEG signal. In typical SSVEP applications, a user is exposed to several visual stimuli with different frequencies, where each stimulus corresponds to a different command. The user chooses the desired command by looking at the associated stimulus, and the

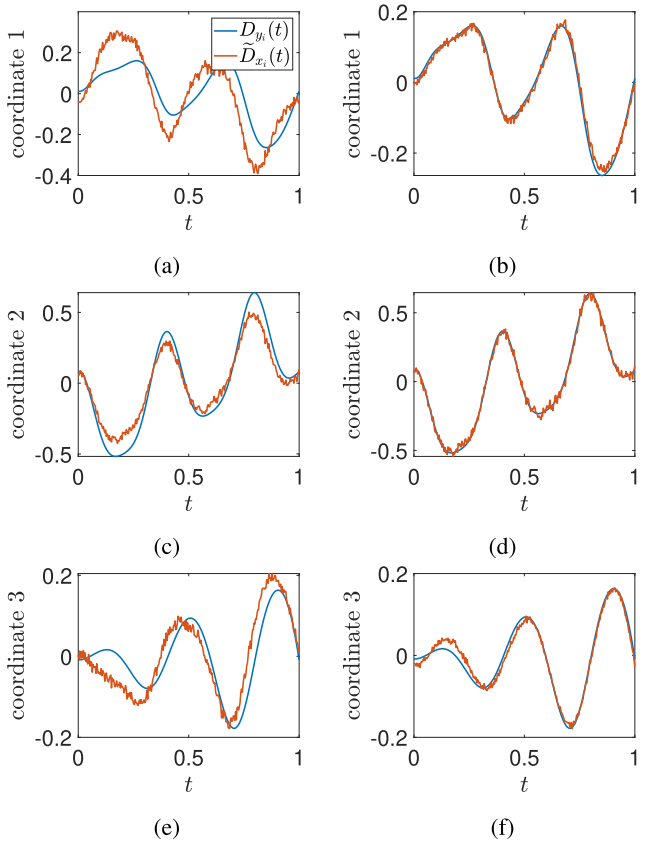

Fig. 5. Denoising results. The clean signal $D_{x_i}(t)$ and the filtered signal $\widetilde{D}_{y_i}(t)$ are represented by the blue curve and the red curve, respectively. Each row presents a different trajectory of $D_{x_i}(t)$ and $\widetilde{D}_{y_i}(t)$. The left and the right columns, show the denoising results without alignment and the denoising results by using alignment, respectively.

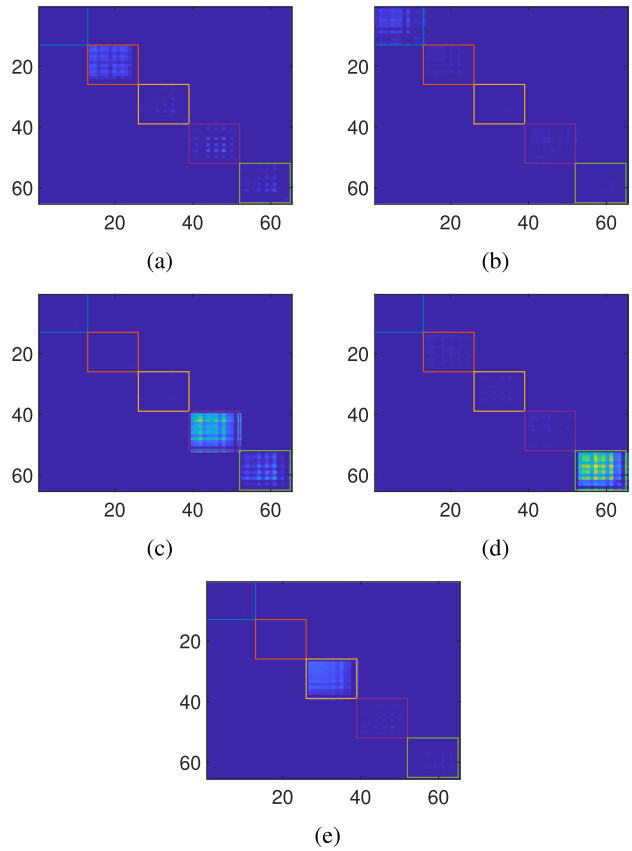

Fig. 6. The covariance matrix of EEG signals recorded in five trials. In each trial the subject was exposed to a flickering light with a different frequency f_k.

system executes that command by identifying the corresponding frequency in the EEG signal.

We apply our method to adapt the EEG recordings of one subject to another. In our experiment, we use the EEG data from the first SSVEP experiment in the MAMEM database [33], [34], which consists of the recordings of 11 subjects. Each subject was exposed in each trial to one of five flickering lights with a different frequency $\{f_k\}_{k=1}^5$. The EEG signals were recorded during the experiment in 256 channels. However, we use only the 13 most informative channels according to the performance analysis reported in [33]. The task in this experiment is to classify the EEG signals of each trial according to the flickering frequencies. We note that in the technical report published with the MAMEM database [33], subjects 3, 5, and 8 were marked as outliers.

In the context of this paper, we will show that there is a significant difference between the data recorded from different subjects, and therefore, it could be beneficial to view each subject as a domain. Subsequently, we will use our method for the purpose of adapting data from two subjects (i.e., two domains), which allows for high-quality classification of EEG signals of one subject by a classifier trained on data from another subject. We note that for fair comparisons with the competing methods, we assume that all the labels of $\mathcal{X}$ are given, rather than the labels of a subset $\mathcal{X}_l \subset \mathcal{X}$.

Let $D \in \mathbb{R}^{N_c \times N_t}$ be the EEG signals recorded in one trial after mean subtraction, where $N_c = 13$ is the number of channels and $N_t = 1250$ is the number of time samples. For preprocessing, we use the mean subtraction and denoising algorithm AMUSE [35], which are implemented in the SSVEP toolbox [36] released as a part of the MAMEM database. Next, we filter D with five band-pass filters centered around the flickering frequency f_k. Let $D_k \in \mathbb{R}^{N_c \times N_t}$ be the EEG signals filtered with the k-th filter. We use the covariance matrix proposed in [37], which is block diagonal with the sample covariance matrix $\Sigma_k = \frac{1}{N_t - 1} D_k D_k^T$, $k = 1, \ldots, 5$, in each block, i.e.,

$$\Sigma = \begin{pmatrix} \Sigma_1 & & & \\ & \Sigma_2 & & \\ & & \ddots & \\ & & & \Sigma_5 \end{pmatrix} \in \mathbb{R}^{65 \times 65}. \tag{36}$$

The dimension of Σ is 65×65 since there are 5 flickering frequencies, and therefore we have $5 \cdot N_c = 65$ filtered signals. Fig. 6 shows the covariance matrix of the EEG signals recorded in five trials. In each trial, the subject was exposed to a flickering light with a different frequency f_k. It can be observed that in each covariance matrix, a different Σ_k is dominant.

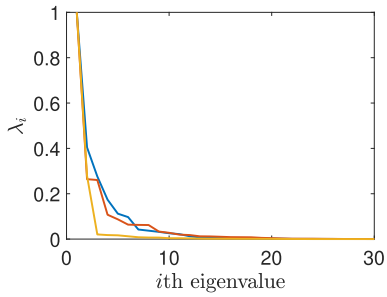

Fig. 7. The eigenvalues decay of three different covariance matrices computed for three different trials of the SSVEP experiment.

In Fig. 7, we plot the eigenvalues decay of three different covariance matrices Σ (36), where each matrix is computed based on a different trial. We report that such decays are prototypical. We see that the eigenvalues (sorted in descending order) have small values, i.e., $\lambda_i \approx 0$, for $i > 12$, implying that the rank of the matrices is approximately $r = 12$. As a result, we consider the covariance matrices as SPSD matrices of fixed rank $r = 12$.

We compute the covariance matrix given in (36) for each trial of each subject, and denote the covariance matrix of the i-th trial and the l-th subject by $X_i^{(l)} \in \mathbb{S}^+_{65,12}$. Let $X_i^{(l)} \cong (G_i^{(l)}, P_i^{(l)})$ be the canonical representation. Throughout this experiment, we set k in (15) such that the dispersion of $\{(G_i^{(l)})\}_i$ is the same as the dispersion of $\{(P_i^{(l)})\}_i$ multiplied by k, balancing the distances in $\mathcal{G}_{d,r}$ and the distances in $\mathcal{P}_r$.

To illustrate the advantage of using the Riemannian geometry of SPSD matrices, we compare it to the Riemannian geometry of SPD matrices. We apply two logarithmic maps to the set $\mathcal{X}^{(6)} = \{X_i^{(6)}\}$: (a) the logarithmic map of the SPD manifold given in (3), and (b) the approximation for the logarithmic map of the SPSD manifold given in (20). Then, we compute the two PCs in the obtained tangent space. Note that in the tangent space, the matrices are viewed as vectors in a linear space, where the standard PCA can be applied. Fig. 8(a) and (b) present the resulting PC computed in the tangent space of the SPD manifold and the SPSD manifold, respectively. Each point represents a matrix of a single trial, which is colored by its respective label, i.e., the stimulus frequency. One common practice to counter the fact that the matrices are rank deficient is the Ledoit-Wolf estimator for high-dimensional covariance matrices [38]. To compare the SPSD geometry to the SPD geometry after applying the Ledoit-Wolf estimator, we use the implementation of [38] in [39] for the estimation of the covariance matrices $\Sigma_k, k = 1, \ldots, 5$ in (36). Fig. 8(c) presents the PCs obtained in the tangent space of the SPD manifold when the covariance matrices were estimated by the Ledoit-Wolf algorithm. Fig. 8(d) presents the PC obtained by stacking each matrix into one vector and then applying PCA, namely, using Euclidean geometry.

Each point in Fig. 8 represents a covariance $X_i^{(6)}$, and it is colored according to the corresponding flickering frequency. We observe that the Riemannian geometry of SPSD matrices provides better separation between the frequencies compared to the separation obtained by the Riemannian geometry of SPD matrices or by the standard Euclidean Geometry.

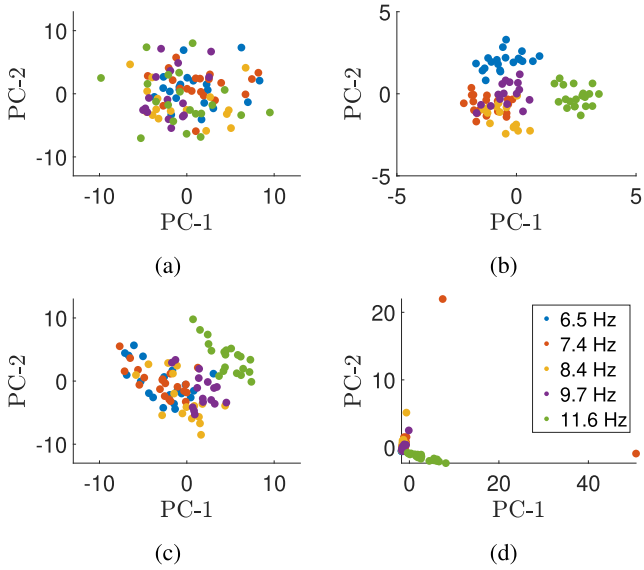

Fig. 8. The PC of the covariance matrices in the set $\mathcal{X}^{(6)}$ after mapping to the tangent space by using: (a) the logarithmic map of the SPD manifold given in (3), (b) the approximation for the logarithmic map of the SPSD manifold proposed in [20], (c) the logarithmic map of the SPD manifold given in (3), where the covariance matrices were estimated using [38]. (d) presents the PC obtained by stacking each matrix into one vector and then applying PCA. Points are colored according to the flickering frequency classes.

TABLE I
COMPARISON BETWEEN THE MEAN ACC OBTAINED BY USING DIFFERENT GEOMETRIES

Geometry	Mean ACC
Euclidean	0.81
SPD	0.33
SPD (LW)	0.82
SPSD	**0.89**

To objectively measure the degree of separation between the frequencies for each of the subjects (except the outlier subjects 3, 5, and 8), we train a Minimum-Distance to Mean (MDM) classifier [21] with the labels of 80% of the trials, and then classify the rest of the trials. The MDM classifier associates a sample to the class with the closest mean. We repeat this experiment 20 times for each subject and compute the mean Classification Accuracy (ACC). Table I shows the mean ACC obtained by using each of the different geometries: Euclidean, SPD with sample covariance matrices, SPD with covariance matrices estimated using [38], and the SPSD geometry proposed by [19]. We see that the MDM classifier obtains the highest mean ACC when using the SPSD geometry.

We note that Fig. 8 visualizes covariance matrices of only one subject. To illustrate the need for data alignment, we visualize the covariance matrices of two different subjects. Fig. 9 presents the tSNE representation [40] computed in the tangent space of data from subjects 2 and 4, which were arbitrarily chosen. Each point represents a covariance matrix of one trial, and it is colored according to the flickering frequency. The covariance matrices of subject 2 are marked by asterisks and the covariance matrices of subject 4 are marked by circles. We observe that trials of the

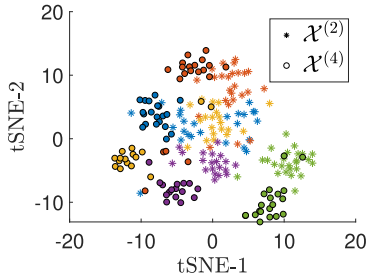

Fig. 9. A two-dimensional tSNE representation of the covariance matrices of two subjects. Each point represents a covariance matrix of one trial, and it is colored according to the flickering frequency class. The covariance matrices of subject 2 are marked by asterisks and the covariance matrices of subject 4 are marked by circles.

TABLE II
MEAN ACC AND STANDARD DEVIATION (STD) AFTER ALIGNMENT BETWEEN ALL PAIRS OF SUBJECTS. NOTE THAT ALGORITHMS WITH ASTERISK (*) ARE COMPLETELY UNSUPERVISED

Algorithm	Mean ACC $\pm$ STD
PA on $\mathcal{P}_d$ [2]	0.71 ± 0.05
Naive union (*)	0.74 ± 0.18
Transport on $\mathcal{S}^+_{d,r}$ (*) [20]	0.77 ± 0.15
PA on $\mathcal{S}^+_{d,r}$ (ours, centering & rotation)	$\mathbf{0.89 \pm 0.09}$
PA on $\mathcal{S}^+_{d,r}$ (ours, complete PA)	0.77 ± 0.11

same flickering frequencies recorded for different subjects do not reside in the same vicinity. The variability between the two subjects limits the ability to identify the flickering frequency in the EEG signal of one subject by training a classifier on signals from another subject. For example, a classifier trained on the recordings of subject 4, obtains a considerably low ACC (0.64) when applied to recordings of subject 2.

We apply the proposed alignment algorithm to align the covariance matrices of all pairs of subjects, where one subject is the source and the other is the target. For the rotation step, we use the labels of 5 trials from each class in the source set. To quantitatively evaluate the alignment of each pair of subjects, we train a MDM classifier [21] on the target set and test it on the source set. We compare the obtained ACC of four algorithms for data alignment: (a) PA on $\mathcal{P}_d$ [2], (b) a naive union of SPSD matrices without alignment, (c) Transportation on $\mathcal{S}^+_{d,r}$ [20] and (d) PA on $\mathcal{S}^+_{d,r}$ (ours). To compare the algorithms numerically, we compute the average accuracy of all pairs excluding the outlier subjects 3, 5 and 8, where our algorithm is applied both with and without the scaling step. Table II shows the average accuracy obtained by the four algorithms. Our algorithm obtains a comparable ACC to the ACC obtained by [20], and in contrast to the experiment in Section VI-A, it obtains the highest accuracy without the scaling step. Fig. 10 presents the confusion matrix for each of the four algorithms consisting of the ACC of all pairs of subjects, excluding subjects 3, 5, and 8. Table III presents the average precision, recall, and F1 measure of each class, obtained by the tested algorithms.

Fig. 6 implies that a specific classifier for the SSVEP application could be implemented as follows:

$$\hat{k} = \underset{k}{\operatorname{argmax}} \quad \|\Sigma_k\|_F, \tag{37}$$

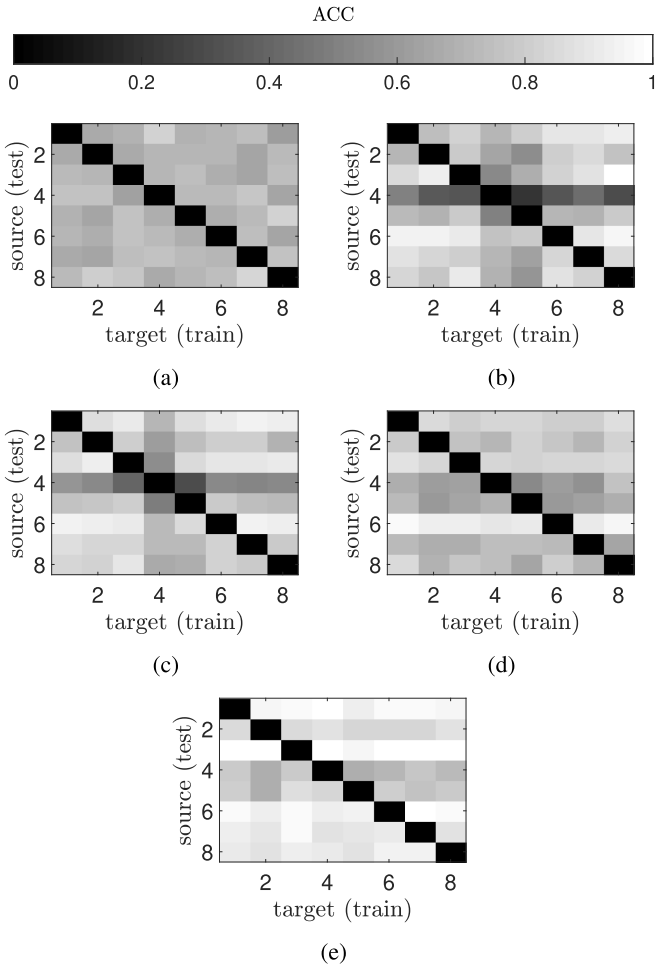

Fig. 10. The confusion matrix obtained by: (a) PA on $\mathcal{P}_d$ [2], (b) a naive union of SPSD matrices, (c) Transportation on $\mathcal{S}^+_{d,r}$ [20], (d) PA on $\mathcal{S}^+_{d,r}$ (ours) and (e) PA on $\mathcal{S}^+_{d,r}$ without scaling (ours). The (i, j) entry is the classification accuracy obtained for the ith subject by a classifier that was trained on the jth subject.

TABLE III
CLASSIFICATION RESULTS. MEAN PRECISION, RECALL, AND F1, AFTER ALIGNMENT BETWEEN ALL PAIRS OF SUBJECTS. NOTE THAT ALGORITHMS WITH ASTERISK (*) ARE COMPLETELY UNSUPERVISED

		PA on $\mathcal{P}_d$ [2]	Naive union (*)	Transport on $\mathcal{S}^+_{d,r}$ (*) [20]	PA on $\mathcal{S}^+_{d,r}$ (ours)
Class 1	Pr	0.69	0.76	0.75	**0.87**
	Re	0.81	0.70	0.73	**0.88**
	F1	0.74	0.69	0.73	**0.87**
Class 2	Pr	0.72	0.75	0.76	**0.89**
	Re	0.68	0.81	0.78	**0.90**
	F1	0.69	0.75	0.77	**0.89**
Class 3	Pr	0.72	0.64	0.68	**0.84**
	Re	0.61	0.60	0.68	**0.82**
	F1	0.65	0.59	0.69	**0.83**
Class 4	Pr	0.84	0.82	0.80	**0.88**
	Re	0.66	0.77	0.81	**0.87**
	F1	0.73	0.77	0.80	**0.87**
Class 5	Pr	0.72	0.84	0.90	**0.95**
	Re	0.80	0.78	0.85	**0.94**
	F1	0.76	0.79	0.86	**0.94**
Mean	Pr	0.74	0.76	0.78	**0.89**
	Re	0.71	0.73	0.77	**0.88**
	F1	0.71	0.72	0.77	**0.88**

TABLE IV
COMPARISON BETWEEN THE ACC OBTAINED BY THE PROPOSED ALGORITHM AND THE ACC OBTAINED BY THE BASELINE CLASSIFIER IN (37) USING SUBJECT 4 AS THE TARGET SET

Subject	Proposed	Baseline
1	0.97	0.93
2	0.85	0.88
3	0.65	0.42
5	0.38	0.25
6	0.79	0.83
7	0.87	0.82
8	0.46	0.16
9	0.97	0.96
10	0.97	0.91
11	0.93	0.95
Mean	0.79	0.71

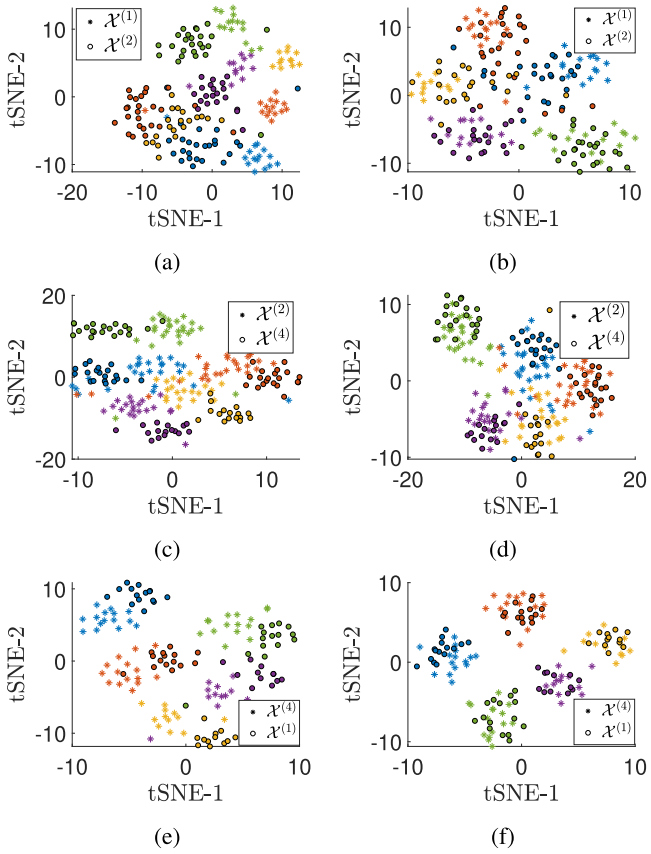

Fig. 11. A two dimensional representation of the covariance matrices of three pairs of subjects before alignment (left column) and after alignment (right column). Each point represents a covariance matrix of one trial, and it is colored according to the flickering frequency class. Covariance matrices from the source set are marked by asterisks and covariance matrices from the target set are marked by circles.

where $\|\cdot\|_F$ is the Frobenius norm, and Σ_k is the k-th block of Σ defined in (36). We consider the classifier in (37) as a baseline classifier for this application. To compare our method with this baseline, we designate the recordings of subject 4 as the target set and train the MDM classifier on this set. Then we classify the recordings of all the other subjects. Table IV compares the obtained ACC with the ACC obtained by the baseline classifier (37). We see that for most of the subjects, the proposed alignment improves the classification results. The mean ACC is improved by 8%. In practice, a subject should be considered as a (good) target if the intra-subject covariance matrices are well separated into the different classes. Specifically, when using the MDM classifier, this separation can be evaluated by the percentage of the covariance matrices for which the closest mean is the mean of the true class. For subject 4 (and subject 11 as well), 100% of the covariance matrices satisfy this condition, and therefore, this is a good choice of a target set. We repeat the above experiment, where each time the recordings of a different subject are designated as the target set. We report that the mean ACC is improved by the alignment for all the subjects, except subjects 3, 5 and 8, which are the outlier subjects.

Same as in Fig. 9, in Fig. 11 we present the tSNE representation of covariance matrices from three pairs of subjects before and after alignment in the left and right columns, respectively. Each point represents a covariance matrix of one trial, and it is colored according to the flickering frequency. Covariance matrices from the source set are marked by asterisks and covariance matrices from the target set are marked by circles. It can be observed that before alignment, points from the source set and points from the target set of the same class, reside in different regions. In contrast, after applying the proposed alignment, the classes of the source set coincide with the classes of the target set.

VII. CONCLUSION

In this work, we propose an algorithm for data alignment based on the Riemannian geometry of SPSD matrices. We show that this algorithm can be applied not only to SPSD matrices but also to high-dimensional data sets with a low-dimensional structure, by performing the centering and rotation steps in the data space using geometric considerations stemming from their corresponding sample covariance matrices. The advantage of the proposed algorithm is illustrated in both simulated and real data.

Our method relies on the assumption that the rank and the dimension of the SPSD matrices in a given set are fixed. In future work, we plan to address the case where SPSD matrices have different ranks or different dimensions [25], corresponding for example to a different number of electrodes in two different EEG experiments. In addition, our algorithm only takes into account the first and the second moments of the statistical distribution of a given data set, while in future work, we will investigate the incorporation of higher moments.

APPENDIX A
PROOF OF PROPOSITION 1

Proof: For the proof of property 1 in Definition 3 see [20]. Next, we prove property 2 in Definition 3. Let $\{C_i \cong (G_i, P_i)\} \subset \mathcal{S}^+_{d,r}$ be the canonical representation of a set with a mean $M(\{C_i\}) = \overline{C} \cong (\overline{G}, \overline{P})$, and let $C_0 \cong (G_0, P_0)$ be another point on $\mathcal{S}^+_{d,r}$ such that $G_0 = \Pi_{\overline{G}}(G_0)$ and $P_0 = G_0^T C_0 G_0$. The length of the curve $\widehat{\gamma}_{\overline{C}\to C_i}$ is given by (19)

$$l^2\left(\widehat{\gamma}_{\overline{C}\to C_i}\right) = d^2_{\mathcal{G}}\left(\overline{G}, G_i\right) + k d^2_{\mathcal{P}}\left(\overline{P}, P_i\right), \quad k > 0.$$

Next, we show that

$$l^2\left(\widehat{\gamma}_{\overline{C}\to C_i}\right) = l^2\left(\widehat{\gamma}_{C_0\to\widetilde{\Gamma}^{+}_{\overline{C}\to C_0}(C_i)}\right) \quad \forall i. \tag{38}$$

The structure space representation of the transported set $\widetilde{\Gamma}^{+}_{\overline{C}\to C_0}(C_i)$ is given by $(T_gG_i, T_pP_iT_p^T)$, where the matrices T_g and T_p are give in Sections II-A and II-B, respectively. To use (19) for the computation of the right hand side of (38), we need to insure that T_gG_i satisfies

$$\Pi_{G_0}(T_gG_i) = T_gG_i.$$

By definition, $\Pi_{G_0}(T_gG_i) = (T_gG_i)O_1O_2^T$, where $G_0^TT_gG_i = O_1\Sigma O_2^T$ is an SVD. Since $G_0 = T_g\overline{G}$ (see [20]), we have

$$G_0^TT_gG_i = \overline{G}^TT_g^TT_gG_i = \overline{G}^TG_i. \tag{39}$$

Therefore, O_1 and O_2 are given by the left and the right orthogonal matrices of the SVD: $\overline{G}^TG_i = \overline{O}\Sigma O_i^T$, respectively. Hence

$$\Pi_{G_0}(T_gG_i) = (T_gG_i)\overline{O}O_i^T = T_gG_i,$$

The last equality is due to the fact that $\Pi_{\overline{G}}(G_i) = G_i\overline{O}O_i^T = G_i$, since (G_i, P_i) is the canonical representation of C_i. Now, we use (19) to compute the right hand side of (38)

$$\begin{aligned} &l^2\left(\widehat{\gamma}_{C_0\to\widetilde{\Gamma}^{+}_{\overline{C}\to C_0}(C_i)}\right) \\ &= d_{\mathcal{G}}^2\left(G_0, T_gG_i\right) + kd_{\mathcal{P}}^2\left(P_0, T_pP_iT_p^T\right) \\ &= d_{\mathcal{G}}^2\left(T_g\overline{G}, T_gG_i\right) + kd_{\mathcal{P}}^2\left(T_p\overline{P}T_p^T, T_pP_iT_p^T\right) \\ &= d_{\mathcal{G}}^2\left(\overline{G}, G_i\right) + kd_{\mathcal{P}}^2\left(\overline{P}, P_i\right) \\ &= l^2\left(\widehat{\gamma}_{\overline{C}\to C_i}\right) \end{aligned}$$

The third equality holds since $d_{\mathcal{P}}$ is an affine-invariant distance and $d_{\mathcal{G}}$ is invariant to orthogonal matrix multiplication. □

Appendix B
Proof of Proposition 2

To show that $\sigma^2(\mathcal{X}^{(\text{scl})}) = \sigma_y^2$ we use the following lemmas.

Lemma 1: Let $G(t)$ be a geodesic path between $G_1 \in \mathcal{G}_{d,r}$ and $G_2 \in \mathcal{G}_{d,r}$, where $G_2 = \Pi_{G_1}(G_2)$. The geodesic path $G(t)$ satisfies

$$G(t) = \Pi_{G_1}\left(G(t)\right).$$

See Appendix C for the proof. According to Lemma 1, $G_i^{(\text{scl})}$ satisfies $G_i^{(\text{scl})} = \Pi_{G_0}(G_i^{(\text{scl})})$. Therefore, the length of the geodesic path between C_0 and $X_i^{(\text{scl})}$ is given by

$$l^2\left(\widehat{\gamma}_{C_0\to X_i^{(\text{scl})}}\right) = d_{\mathcal{G}}^2\left(G_0, G_i^{(\text{scl})}\right) + kd_{\mathcal{P}}^2\left(P_0, P_i^{(\text{scl})}\right).$$

Lemma 2: Let $G(t)$ be a geodesic path between two points $G_1, G_2 \in \mathcal{G}_{d,r}$. The distance between G_1 and $G(t)$ is given by

$$d_{\mathcal{G}}^2\left(G_1, G(t)\right) = t^2\|\Theta\|_F^2 \tag{40}$$

where $G_1^TG_2 = O_1(\cos\Theta)O_2^T$ is an SVD.

See Appendix D for the proof.

We note that an analogous result to Lemma 2 in the SPD manifold can be obtained by substituting (5) in (6)

$$d_{\mathcal{P}}^2\left(P_1, P(t)\right) = t^2\sum_{i=1}^{r}\log^2(\lambda_i), \tag{41}$$

where λ_i are the eigenvalues of $P_1^{-1}P_2$. By using Lemmas 1 and 2, we now prove Proposition 2.

Proof:

$$\begin{aligned} &\sigma^2\left(\mathcal{X}^{(\text{scl})}\right) = \\ &= \sum_{i=1}^{N_x} l^2\left(\widehat{\gamma}_{C_0\to X_i^{(\text{scl})}}\right) \\ &= \sum_{i=1}^{N_x} d_{\mathcal{G}}^2\left(G_0, G_i\left(t=\frac{\sigma_v}{\sigma_g}\right)\right) + kd_{\mathcal{P}}^2\left(P_0, P_i\left(t=\frac{\sigma_r}{\sigma_p}\right)\right) \\ &= \frac{\sigma_v^2}{\sigma_g^2}\sum_{i=1}^{N_x} d_{\mathcal{G}}^2\left(G_0, G_i^{(\text{ctr})}\right) + \frac{\sigma_r^2}{\sigma_p^2}\sum_{i=1}^{N_x} d_{\mathcal{P}}^2\left(P_0, P_i^{(\text{ctr})}\right) \\ &= \sigma_v^2 + k\sigma_r^2 = \sigma_y^2. \end{aligned}$$

The second equality is due to Lemma 1 and (41), and the third equality is due to Lemma 2. □

Appendix C
Proof of Lemma 1

Proof: Let $G_{1,\perp}$ be the orthogonal complement of G_1, such that $[G_1 \quad G_{1,\perp}] \in \mathcal{O}_d$. The geodesic path $G(t)$ is given by [26]

$$G(t) = G_1V\cos(\Sigma t)V^T + U\sin(\Sigma t)V^T, \tag{42}$$

where $U\Sigma V^T$ is the compact SVD of $G_{1,\perp}B$, $B \in \mathbb{R}^{(d-r)\times r}$. Recall that

$$\Pi_{G_1}\left(G(t)\right) = G(t)O_tO_1^T,$$

where $G_1^TG(t) = O_1SO_t^T$ is an SVD (see (17)). To find the SVD of $G_1^TG(t)$, we multiply (42) by G_1^T from the left

$$\begin{aligned} G_1^TG(t) &= G_1^TG_1V\cos(\Sigma t)V^T + G_1^TU\sin(\Sigma t)V^T \\ &= V\cos(\Sigma t)V^T + G_1^TU\sin(\Sigma t)V^T. \end{aligned}$$

The second term equals zero since the columns of U are in the subspace of $G_{1,\perp}$. Therefore, the SVD of $G_1^TG(t)$ is given by

$$G_1^TG(t) = V\cos(\Sigma t)V^T,$$

and

$$\Pi_{G_1}\left(G(t)\right) = G(t)VV^T = G(t).$$

□

Appendix D
Proof of Lemma 2

Proof: Let H be the direction of $G(t)$ at $t = 0$, i.e., $\dot{G}(0) = H$. The path length between G_1 and $G(t)$ is given by [26]

$$d_{\mathcal{G}}\left(G_1, G(t)\right) = t\left\|\Sigma\right\|_F, \tag{43}$$

where $H = U\Sigma V^T$ is the compact SVD of H. As noted in [26], (43) holds only for small enough t such that there are no conjugate points. In our context, it theoretically limits the ratio $t = \sigma_v/\sigma_g$, however, our empirical tests did not find it limiting in practice.

For $t = 1$ we have $G(1) \in [G_2]$. By substituting $t = 1$ in (43) we get

$$d_{\mathcal{G}}\left(G_1, G(1)\right) = d_{\mathcal{G}}(G_1, G_2) = \|\Sigma\|_F . \tag{44}$$

On the other hand, the arc length of the geodesic path between the points G_1 and G_2 is given by

$$d_{\mathcal{G}}\left(G_1, G_2\right) = \|\Theta\|_F ,$$

where $G_1^T G_2 = O_1(\cos\Theta)O_2^T$ is an SVD. Therefore, we get $\|\Theta\|_F = \|\Sigma\|_F$, and (43) becomes

$$d_{\mathcal{G}}\left(G_1, G(t)\right) = t\|\Theta\|_F . \tag{45}$$

□

Appendix E
Proof of Proposition 3

Proof: First, we explicitly write $\widehat{\Sigma}_{X_i}$ in (32):

$$\begin{aligned}
\widehat{\Sigma}_{X_i} &:= \widetilde{\Gamma}^{+}_{C_0 \to \overline{Y}} \left(R \left(\widetilde{\Gamma}^{+}_{\overline{X} \to C_0} \left(\Sigma_{X_i} \right) \right) \right) \\
&\cong \left(T_{g_y}^{-1} O_g T_{g_x} G_i, T_{p_y}^{-1} O_p T_{p_x} P_i T_{p_x}^T O_p^T T_{p_y}^{-T} \right) \\
&\cong \left(T_{g_y}^{-1} O_g T_{g_x} G_i \right) \left(T_{p_y}^{-1} O_p T_{p_x} P_i T_{p_x}^T O_p^T T_{p_y}^{-T} \right) \cdot \\
&\quad \cdot \left(T_{g_y}^{-1} O_g T_{g_x} G_i \right)^T ,
\end{aligned}$$

Then, we have

$$\begin{aligned}
&\frac{1}{n_{x_i} - 1} \widehat{D}_{x_i} \widehat{D}_{x_i}^T \\
&= \frac{1}{n_{x_i} - 1} \left(T_{g_y}^{-1} O_g T_{g_x} \right) G_i \left(T_{p_y}^{-1} O_p T_{p_x} \right) G_i^T \left(D_{x_i} D_{x_i}^T \right) \cdot \\
&\quad \cdot G_i \left(T_{p_y}^{-1} O_p T_{p_x} \right)^T G_i^T \left(T_{g_y}^{-1} O_g T_{g_x} \right)^T \\
&= \left(T_{g_y}^{-1} O_g T_{g_x} \right) G_i \left(T_{p_y}^{-1} O_p T_{p_x} \right) G_i^T \left(G_i P_i G_i^T \right) \cdot \\
&\quad \cdot G_i \left(T_{p_y}^{-1} O_p T_{p_x} \right)^T G_i^T \left(T_{g_y}^{-1} O_g T_{g_x} \right)^T \\
&= \left(T_{g_y}^{-1} O_g T_{g_x} G_i \right) \left(T_{p_y}^{-1} O_p T_{p_x} P_i T_{p_x}^T O_p^T T_{p_y}^{-T} \right) \cdot \\
&\quad \cdot \left(T_{g_y}^{-1} O_g T_{g_x} G_i \right)^T \\
&= \widehat{\Sigma}_{X_i}
\end{aligned}$$

The second equality is due to the definition $\Sigma_{X_i} = \frac{1}{n_{x_i}-1} D_{x_i} D_{x_i}^T = G_i P_i G_i^T$. The third equality is due to the fact that $G_i \in \mathcal{V}_{d,r}$ and therefore, $G_i^T G_i = I_{r,r}$. □

Acknowledgment

We wish to thank the associate editor and the anonymous reviewers for their helpful comments.

References

[1] J. C. Gower and G. B. Dijksterhuis, *Procrustes Problems*, vol. 30. London, U.K.: Oxford Univ. Press, 2004.

[2] P. L. C. Rodrigues, C. Jutten, and M. Congedo, "Riemannian procrustes analysis: Transfer learning for brain computer interfaces," *IEEE Trans. Biomed. Eng.*, vol. 66, no. 8, pp. 2390–2401, Aug. 2019.

[3] C. Wang et al., "Comparing spatial maps of human population-genetic variation using procrustes analysis," *Stat. Appl. Genet. Mol. Biol.*, vol. 9, no. 1, 2010.

[4] T. Turki, Z. Wei, and J. T. Wang, "A transfer learning approach via procrustes analysis and mean shift for cancer drug sensitivity prediction," *J. Bioinf. Comput. Biol.*, vol. 16, no. 03, 2018, Art. no. 1840014.

[5] Y.-W. E. Lin, Y. Kluger, and R. Talmon, "Hyperbolic procrustes analysis using Riemannian geometry," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, vol. 34, pp. 5959–5971.

[6] H. Zou, B. Huang, X. Lu, H. Jiang, and L. Xie, "A robust indoor positioning system based on the procrustes analysis and weighted extreme learning machine," *IEEE Trans. Wireless Commun.*, vol. 15, no. 2, pp. 1252–1266, Feb. 2016.

[7] Y. Tai, J. Yang, Y. Zhang, L. Luo, J. Qian, and Y. Chen, "Face recognition with pose variations and misalignment via orthogonal procrustes regression," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2673–2683, Jun. 2016.

[8] P. Tabaghi and I. Dokmanić, "On procrustes analysis in hyperbolic space," *IEEE Signal Process. Lett.*, vol. 28, pp. 1120–1124, 2021.

[9] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[10] R. R. Coifman and S. Lafon, "Diffusion maps," *Appl. Comput. Harmon. Anal.*, vol. 21, no. 1, pp. 5–30, 2006.

[11] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[12] J. B. Tenenbaum, V. D. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[13] X. Pennec, P. Fillard, and N. Ayache, "A Riemannian framework for tensor computing," *Int. J. Comput. Vis.*, vol. 66, no. 1, pp. 41–66, 2006.

[14] R. Bhatia, *Positive Definite Matrices*, vol. 24.Princeton, NJ, USA: Princeton Univ. Press, 2009.

[15] A. Kapur, K. Marwah, and G. Alterovitz, "Gene expression prediction using low-rank matrix completion," *BMC Bioinf.*, vol. 17, no. 1, 2016, Art. no. 243.

[16] A. Halimi, P. Honeine, M. Kharouf, C. Richard, and J. Tourneret, "Estimating the intrinsic dimension of hyperspectral images using a noise-whitened eigengap approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 7, pp. 3811–3821, Jul. 2016.

[17] Y. Niu and B. Wang, "Hyperspectral anomaly detection based on low-rank representation and learned dictionary," *Remote Sens.*, vol. 8, no. 4, 2016, Art. no. 289.

[18] F. R. Chung and F. C. Graham, *Spectral Graph Theory*. Providence, RI, USA: Amer. Math. Soc., 1997.

[19] S. Bonnabel and R. Sepulchre, "Riemannian metric and geometric mean for positive semidefinite matrices of fixed rank," *SIAM J. Matrix Anal. Appl.*, vol. 31, no. 3, pp. 1055–1070, 2010.

[20] O. Yair, A. Lahav, and R. Talmon, "Symmetric positive semi-definite Riemannian geometry with application to domain adaptation," 2020, *arXiv:2007.14272*.

[21] A. Barachant, S. Bonnet, M. Congedo, and C. Jutten, "Multiclass brain computer interface classification by Riemannian geometry," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 4, pp. 920–928, Apr. 2012.

[22] F. Yger, M. Berar, and F. Lotte, "Riemannian approaches in brain-computer interfaces: A review," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 10, pp. 1753–1762, Oct. 2017.

[23] M. Congedo, A. Barachant, and R. Bhatia, "Riemannian geometry for EEG-based brain-computer interfaces; a primer and a review," *Brain-Comput. Interfaces*, vol. 4, no. 3, pp. 155–174, 2017.

[24] F. Lotte et al., "A review of classification algorithms for EEG-based brain–computer interfaces: A 10 year update," *J. Neural Eng.*, vol. 15, no. 3, 2018, Art. no. 031005.
[25] P. Rodrigues, M. Congedo, and C. Jutten, "Dimensionality transcending: A method for merging BCI datasets with different dimensionalities," *IEEE Trans. Biomed. Eng.*, vol. 68, no. 2, pp. 673–684, Feb. 2021.
[26] A. Edelman, T. A. Arias, and S. T. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM J. Matrix Anal. Appl.*, vol. 20, no. 2, pp. 303–353, 1998.
[27] O. Yair, M. Ben-Chen, and R. Talmon, "Parallel transport on the cone manifold of SPD matrices for domain adaptation," *IEEE Trans. Signal Process.*, vol. 67, no. 7, pp. 1797–1811, Apr. 2019.
[28] S. Sra and R. Hosseini, "Conic geometric optimization on the manifold of positive definite matrices," *SIAM J. Optim.*, vol. 25, no. 1, pp. 713–739, 2015.
[29] B. Vandereycken, P.-A. Absil, and S. Vandewalle, "A Riemannian geometry with complete geodesics for the set of positive semidefinite matrices of fixed rank," *IMA J. Numer. Anal.*, vol. 33, no. 2, pp. 481–514, Apr. 2013.
[30] E. Massart and P.-A. Absil, "Quotient geometry with simple geodesics for the manifold of fixed-rank positive-semidefinite matrices," *SIAM J. Matrix Anal. Appl.*, vol. 41, no. 1, pp. 171–198, 2020.
[31] S. Bonnabel, A. Collard, and R. Sepulchre, "Rank-preserving geometric means of positive semi-definite matrices," *Linear Algebra its Appl.*, vol. 438, no. 8, pp. 3202–3216, 2013.
[32] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre, "Manopt, a matlab toolbox for optimization on manifolds," *J. Mach. Learn. Res.*, vol. 15, no. 42, pp. 1455–1459, 2014. [Online]. Available: https://www.manopt.org
[33] V. P. Oikonomou et al., "Comparative evaluation of state-of-the-art algorithms for SSVEP-based BCIs," 2016, *arXiv:1602.00904*.
[34] A. L. Goldberger et al., "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.
[35] P. Martinez, H. Bakardjian, and A. Cichocki, "Fully online multicommand brain-computer interface with visual neurofeedback using SSVEP paradigm," *Comput. Intell. Neurosci.*, vol. 2007, 2007, Art. no. 094561.
[36] G. Liaros et al., "SSVEP-EEG-processingtoolbox," 2016. [Online]. Available: https://github.com/MAMEM/ssvep-eeg-processing-toolbox
[37] M. Congedo, "EEG source analysis," 2013. [Online]. Available: https://tel.archives-ouvertes.fr/tel-00880483/document
[38] O. Ledoit and M. Wolf, "A well-conditioned estimator for large-dimensional covariance matrices," *J. Multivariate Anal.*, vol. 88, no. 2, pp. 365–411, 2004.
[39] O. Ledoit, "Covshrinkage," 2022. [Online]. Available: https://github.com/oledoit/covShrinkage/releases/tag/1.1.0
[40] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.