Gaussian Mixture Regression on Symmetric Positive Definite Matrices Manifolds: Application to Wrist Motion Estimation with sEMG

Noémie Jaquier and Sylvain Calinon

***Abstract*— In many sensing and control applications, data are represented in the form of symmetric positive definite (SPD) matrices. Considering the underlying geometry of this data space can be beneficial in many robotics applications. In this paper, we present an extension of Gaussian mixture regression (GMR) with input and/or output data on SPD manifolds. As the covariance of SPD datapoints is a 4th-order tensor, we develop a method for parallel transport of high order covariances on SPD manifolds. The proposed approach is experimented in the context of prosthetic hands, with the estimation of wrist movements based on spatial covariance features computed from surface electromyography (sEMG) signals.**

I. INTRODUCTION

Symmetric positive definite (SPD) matrices are encountered in various domains. For instance, in medical imaging, diffusion tensor magnetic resonance imaging (DT-MRI) provides measurements in the form of SPD matrices [1], [2]. Covariance features provide also a robust representation for object tracking [3] or for analyzing electroencephalography (EEG) signals [4]. Considering the underlying Riemannian geometry has been proven beneficial for many applications, such as interpolation and regression in the space of SPD matrices [5], [6].

Gaussian mixture regression (GMR) exploits the Gaussian conditioning theorem to estimate the distribution of output data given input data. A Gaussian mixture model (GMM) is first estimated to encode the joint distribution of input and output datapoints, e.g. with an Expectation-Maximization (EM) algorithm. The output is then estimated as a linear combination of expected output Gaussian distributions, given observed inputs. This regression method provides a fast and efficient way to estimate multivariate output data from multivariate input data in the form of Gaussian distributions with full covariances. The approach does not learn the regression function directly, but instead relies on the learned joint distribution. The approach is widely used in imitation learning to generate robot motions [7].

The formulation of Gaussian distributions in standard GMM and GMR assumes data in the Euclidean space. Recently, probabilistic encoding using GMM/GMR was extended to orientation data (quaternions on hypersphere manifolds) [8]. This work demonstrated that parallel transport of covariance matrices —a standard technique in differential geometry— is essential to extend GMR to Riemannian manifolds. In this paper, we extend the generalization of GMM and GMR to data in the form of SPD matrices. To achieve this, we characterize the (co)variability of 2nd-order tensors (matrices) by a 4th-order covariance tensor. In order to encode the variability with full covariances, we develop an efficient parallel transport method for 4th-order covariance tensors of SPD matrices. We exploit the proposed approach in the context of prosthetic hands, with the goal of identifying wrist movements from spatial covariance features acquired by surface electromyography.

The paper is organized as follows: Section II motivates the use of GMR on SPD matrices and explains the need of inversion and parallel transport for 4th-order covariance tensors. Related work is described in Section III. Section IV introduces Riemannian manifolds, tensor representation and the normal distribution of SPD matrices. The proposed approach is described in Section V and evaluated on sEMG signals in Section VI. Section VII concludes the paper.

II. MOTIVATING EXAMPLE

The extension of GMM/GMR to Riemannian manifolds allows the retrieval of smooth trajectories for data that do not belong to the Euclidean space. The process remains similar: the joint density distribution of a set of input and output data is modeled in a GMM. The expected conditional distribution of the output, given a new input, is then estimated using GMR. The components of a GMM are characterized by their centers and covariance matrices.

To apply GMR, we have to invert and parallel transport parts of the covariance matrices obtained by the GMM, see [8] for details. In the case of SPD manifolds, the center of a GMM component is a SPD matrix, and its variability is represented by a 4th-order tensor (4-way array of data). The inverse and parallel transport of 4th-order tensors are not trivial operations, except for the identity tensor (eventually multiplied by a constant). In this case, both operations result in the identity tensor.

In standard GMM, it is well known that forcing covariances to be isotropic can have limited representation power, particularly in applications requiring coordination patterns to be modeled. Similarly, forcing higher order GMMs to have isotropic tensors as covariances can be limited for the encoding of coordination information, by decoupling each dimension.

Fig. 1 presents an example motivating the use of full covariances or tensors in a GMM, in the special case of trajectory encoding and retrieval (i.e., with a time signal

This work was supported by the SNSF/DFG project "TACT-HAND: Improving control of prosthetic hands using tactile sensors and realistic machine learning" (Swiss National Science Foundation project 200021E-160665).

The authors are with the Idiap Research Institute, Martigny, Switzerland. {noemie.jaquier, sylvain.calinon}@idiap.ch

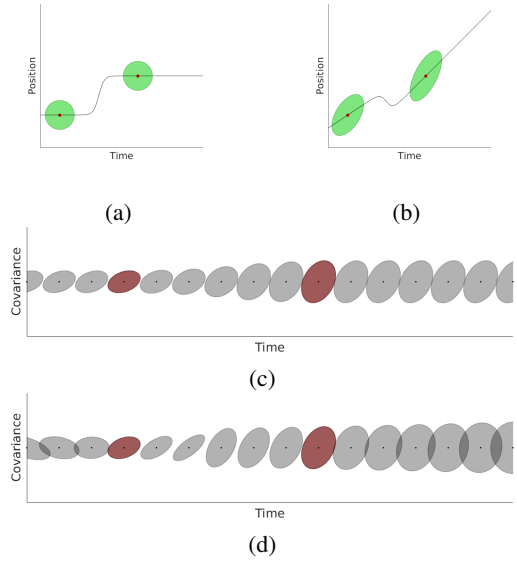


Fig. 1: Examples of Gaussian mixture regression results, where time inputs are used to retrieve a 1D trajectory (*a,b*) or a sequence of 2×2 SPD matrices (*c,d*). The joint density is modeled by a 2-component GMM. (*a,b*) The retrieved trajectory are in black lines, while the centers and covariances are represented by green ellipses with red nucleus (equiprobable contour of one standard deviation). (*c,d*) The retrieved sequence of SPD matrices and the centers of the GMM components are represented in gray and red ellipsoids, respectively. Note that the 4th-order covariance is not depicted in the graph.

as input). We can see in Fig. 1a, 1c that with isotropic Gaussians, the retrieved trajectories show a smooth transition between the components of the GMM, but remain constant elsewhere, also when extrapolating outside the range of the Gaussians. When full covariances are used to model the joint distribution, the retrieved data are characterized by a local linear trend that better models the evolution of the covariance, for both interpolation and extrapolation, see Fig. 1b, 1d. The benefits of using full covariances in GMMs motivates the need to develop inversion and parallel transport methods for full covariance tensors.

III. RELATED WORK

Ghahramani and Jordan were the first to formulate Gaussian mixture regression (GMR) by studying regression employing a mixture of Gaussians in Euclidean space [9]. In [10], Simo-Serra *et al.* propose an extension of EM algorithm to Riemannian manifold for data in vector form. Their approach allows each distribution to be located on its own tangent space. In [11], Kim *et al.* also reformulates GMM to handle the space of rotation in $\mathbb{R}^3$. Both publications present methods for regression from a mixture of Gaussians on Riemannian manifolds, although they only partially exploit the manifold structure in Gaussian conditioning. In [8], Zeestraten *et al.* extend probabilistic encoding using Gaussians to Riemannian manifolds with data represented in vector form. They demonstrate that the parallel transport of covariance matrices is essential for Gaussian conditioning on Riemannian manifolds.

In [12], Said *et al.* introduce Riemannian Gaussian distribution and propose an implementation of EM for Gaussian mixture on SPD manifolds. However, their formulation only mildly characterize the (co)variability of SPD matrices as it is represented by a scalar. In [13], Zhan and Ma propose a modified EM algorithm for GMM on SPD manifolds, where SPD matrices are converted to vectors. Their covariance is then expressed by a matrix of higher dimensions, which is problematic for covariance transport. In [1], Basser and Pajevic present a normal distribution on SPD manifold where the covariability of matrices has the form of a 4th-order tensor. We use this formulation to extend Gaussian mixture model and Gaussian mixture regression to SPD manifolds. We also extend the parallel transport of elements in the form of vectors presented in [14] to SPD manifolds.

Related work concerning the proposed experiment is described in Section VI.

IV. PRELIMINARIES

In this section, we first introduce useful notions of geometry on Riemannian manifolds, particularly on SPD manifolds. Interested readers are referred to [15], [16] for more information. We then introduce tensor-based representation and discuss the normal distribution for SPD matrices as well as the representation of covariance for SPD matrices.

Scalars are denoted by lower case letters x, vectors by boldface lower case letters $\boldsymbol{x}$, matrices by boldface uppercase letters $\boldsymbol{X}$, tensors by boldface calligraphic letters $\boldsymbol{\mathcal{X}}$, and dimensions by uppercase letters. Manifolds and their tangent spaces are designated by calligraphic letters $\mathcal{M}$ and $\mathcal{TM}$, respectively.

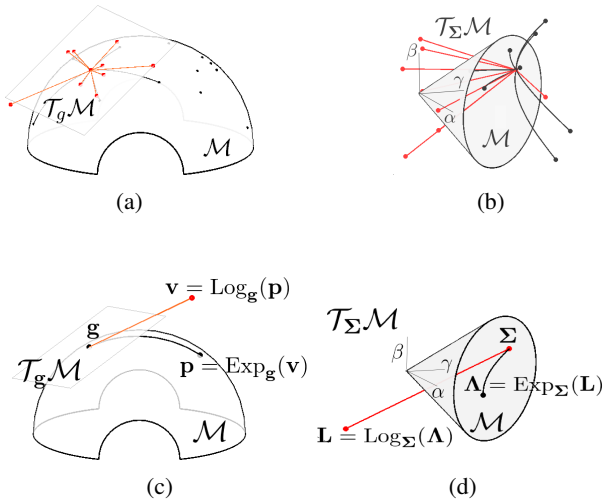
A. Riemannian Manifold of SPD Matrices

A manifold $\mathcal{M}$ is a mathematical space for which each point locally resembles a Euclidean space. For each point $\boldsymbol{p} \in \mathcal{M}$, there exists a tangent space $\mathcal{T}_{\boldsymbol{p}}\mathcal{M}$ made of the tangent vectors of all the curves passing through that point, e.g. Fig. 2a. A Riemannian manifold is a differentiable manifold for which each tangent space is equipped with a Riemannian metric, i.e. a positive definite inner product.

The SPD manifold $\mathcal{S}^D_{++}$ is the space of $D \times D$ SPD matrices. A real symmetric matrix $\boldsymbol{\Sigma}$ is positive definite if $\boldsymbol{v}^\top\boldsymbol{\Sigma}\boldsymbol{v} > 0,\ \forall \boldsymbol{v} \neq \boldsymbol{0}$. The tangent space at any point $\boldsymbol{\Sigma} \in \mathcal{S}^D_{++}$ is the space of symmetric matrices Sym^D. The space of SPD matrices forms the interior of a convex cone in the space Sym^D (see Fig. 2b). The invariant Riemannian metric generally used for SPD manifolds is the inner product between two matrices $\boldsymbol{V}_1,\ \boldsymbol{V}_1 \in \mathcal{T}_{\boldsymbol{\Sigma}}\mathcal{S}^D_{++}$ such that [16]

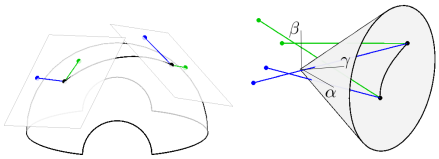
$$<\boldsymbol{V}_1, \boldsymbol{V}_2>_{\boldsymbol{\Sigma}} \ = \ \text{tr}(\boldsymbol{\Sigma}^{-\frac{1}{2}}\boldsymbol{V}_1\boldsymbol{\Sigma}^{-1}\boldsymbol{V}_2\boldsymbol{\Sigma}^{-\frac{1}{2}}).$$

The minimum length curves between two points on a Riemannian manifold are called geodesics. Similarly to straight lines in Euclidean space, the second derivative is zero everywhere along a geodesic. The exponential map $\text{Exp}_{\boldsymbol{p}} : \mathcal{T}_{\boldsymbol{g}}\mathcal{M} \rightarrow \mathcal{M}$ maps a point $\boldsymbol{v}$ in the tangent space to a point $\boldsymbol{p}$ on the manifold, so that is lies on the geodesic starting at $\boldsymbol{g}$ in the direction $\boldsymbol{v}$ and such that the geodesic distance between $\boldsymbol{g}$ and $\boldsymbol{p}$ is equal to the Euclidean distance between $\boldsymbol{g}$ and $\boldsymbol{v}$



(a) (b) (c) (d)

Fig. 2: Manifold representations and mappings for $\mathcal{S}^2$ spheres (left column) and $\mathcal{S}^2_{++}$ SPD matrices (right column). The manifold $\mathcal{S}^2$ is here employed to facilitate the visualization of the manifold characteristics. Black and red points belong to $\mathcal{M}$ and $\mathcal{TM}$, respectively. (*a*) Representation of $\mathcal{S}^2$ and its tangent space. (*b*) Representation of $\mathcal{S}^2_{++}$ embedded in its tangent space Sym^2. One point on the graph corresponds to a SPD matrix $\begin{pmatrix} \alpha & \beta \\ \beta & \gamma \end{pmatrix}$. (*c,d*) Exponential and logarithm maps on $\mathcal{S}^2$ and $\mathcal{S}^2_{++}$, providing one to one mapping between manifold and tangent space.



(a) (b)

Fig. 3: Parallel transport of vectors (*a*), and matrices (*b*) (green and blue lines) between tangent spaces of $\mathcal{S}^2$ and $\mathcal{S}^2_{++}$, respectively. The inner product, i.e. the angle, between elements in the tangent space is constant during parallel transport.

(see Fig. 2c). The inverse map is called the logarithm map $\mathrm{Log}_{\boldsymbol{p}} : \mathcal{M} \rightarrow \mathcal{T}_{\boldsymbol{g}}\mathcal{M}$. For the given metric, the exponential and logarithm maps on SPD manifold are (see also Fig. 2d)

$$\mathbf{\Lambda} = \mathrm{Exp}_{\mathbf{\Sigma}}(\boldsymbol{L}) = \mathbf{\Sigma}^{\frac{1}{2}} \exp(\mathbf{\Sigma}^{-\frac{1}{2}} \boldsymbol{L} \mathbf{\Sigma}^{-\frac{1}{2}}) \mathbf{\Sigma}^{\frac{1}{2}},$$
$$\boldsymbol{L} = \mathrm{Log}_{\mathbf{\Sigma}}(\mathbf{\Lambda}) = \mathbf{\Sigma}^{\frac{1}{2}} \log(\mathbf{\Sigma}^{-\frac{1}{2}} \mathbf{\Lambda} \mathbf{\Sigma}^{-\frac{1}{2}}) \mathbf{\Sigma}^{\frac{1}{2}}.$$

Parallel transport $\Gamma_{\boldsymbol{g}\rightarrow\boldsymbol{h}} : \mathcal{T}_{\boldsymbol{g}}\mathcal{M} \rightarrow \mathcal{T}_{\boldsymbol{h}}\mathcal{M}$ moves vectors between tangent spaces such that the inner product between two vectors in a tangent space is conserved, see Fig. 3a. The parallel transport of $\boldsymbol{V} \in \mathcal{T}_{\mathbf{\Sigma}}\mathcal{S}^D_{++}$ to $\mathcal{T}_{\mathbf{\Lambda}}\mathcal{S}^D_{++}$ is given by

$$\Gamma_{\mathbf{\Sigma}\rightarrow\mathbf{\Lambda}}(\boldsymbol{V}) = \boldsymbol{A}_{\mathbf{\Sigma}\rightarrow\mathbf{\Lambda}}\ \mathbf{\Sigma}\ \boldsymbol{A}^{\top}_{\mathbf{\Sigma}\rightarrow\mathbf{\Lambda}}, \tag{1}$$

with $\boldsymbol{A}_{\mathbf{\Sigma}\rightarrow\mathbf{\Lambda}} = \mathbf{\Sigma}^{\frac{1}{2}} \exp(\frac{1}{2}\mathbf{\Sigma}^{-\frac{1}{2}} \boldsymbol{V} \mathbf{\Sigma}^{-\frac{1}{2}}) \mathbf{\Sigma}^{-\frac{1}{2}}$, see Fig. 3b.

B. Tensor Representation

Tensors are generalization of matrices to higher order. Vectors and matrices may respectively be seen as 1st and 2nd-order tensors.

The tensor product of two tensors $\boldsymbol{\mathcal{U}} \in \mathbb{R}^{I_1 \times \ldots \times I_m}$, $\boldsymbol{\mathcal{V}} \in \mathbb{R}^{J_1 \times \ldots \times J_n}$ is $\boldsymbol{\mathcal{W}} = \boldsymbol{\mathcal{U}} \otimes \boldsymbol{\mathcal{V}} \in \mathbb{R}^{I_1 \times \ldots \times I_m \times J_1 \times \ldots \times J_n}$, where

$$\boldsymbol{\mathcal{W}}_{i_1,\ldots,i_m,j_1,\ldots,j_n} = \boldsymbol{\mathcal{U}}_{i_1,\ldots,i_m}\ \boldsymbol{\mathcal{V}}_{j_1,\ldots,j_n}.$$

The tensor product is a multilinear generalization of the outer product of two vectors $\boldsymbol{u} \otimes \boldsymbol{v}$, equivalent to the matrix multiplication $\boldsymbol{u}\boldsymbol{v}^{\top}$.

Similarly to the computation of the covariance of vectors, the 4th-order covariance tensor of matrices is defined as

$$\boldsymbol{\mathcal{S}} = \frac{1}{N-1} \sum_{n=1}^{N} \boldsymbol{X}_n \otimes \boldsymbol{X}_n,$$

where $\boldsymbol{X}_n$ is the n-th datapoint and N is the total number of datapoints.

We follow the simple and convenient convention employed in [3], by denoting the element (p, q, i, j) of a 4th-order tensor $\boldsymbol{\mathcal{S}}$ by $\boldsymbol{\mathcal{S}}^{ij}_{pq}$, with two covariant indices (p, q) and two contravariant indices (i, j). The element (i, j) of a matrix $\boldsymbol{X}$ is denoted by $\boldsymbol{X}_{ij}$ with two covariant indices (i, j). A tensor contraction between two tensors is performed when one or more contravariant and covariant indices are identical. For example, the tensor contraction of $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{D\times D\times D\times D}$ and $\boldsymbol{X} \in \mathbb{R}^{D\times D}$ will be denoted

$$\boldsymbol{\mathcal{S}}\boldsymbol{X} = \sum_{i=1}^{D}\sum_{j=1}^{D} \boldsymbol{\mathcal{S}}^{ij}_{pq} \boldsymbol{X}_{ij},$$

where the typeface helps at disambiguating a tensor operation from a matrix operation.

Analogously to the representation used for 3rd-order tensor [17], we represent 4th-order tensor by separating the different fibers with bars, so that $\boldsymbol{\mathcal{S}}$ can also be visualized as

$$\boldsymbol{\mathcal{S}} = \left(\begin{array}{c|c|c} \boldsymbol{\mathcal{S}}^{11}_{::} & \ldots & \boldsymbol{\mathcal{S}}^{1J}_{::} \\ \hline \vdots & \ddots & \vdots \\ \hline \boldsymbol{\mathcal{S}}^{J1}_{::} & \ldots & \boldsymbol{\mathcal{S}}^{IJ}_{::} \end{array}\right), \text{ with } \boldsymbol{\mathcal{S}}^{ij}_{::} = \begin{pmatrix} \boldsymbol{\mathcal{S}}^{ij}_{11} & \ldots & \boldsymbol{\mathcal{S}}^{ij}_{1Q} \\ \vdots & \ddots & \vdots \\ \boldsymbol{\mathcal{S}}^{ij}_{P1} & \ldots & \boldsymbol{\mathcal{S}}^{ij}_{PQ} \end{pmatrix}.$$

C. The Normal Distribution for SPD matrices

Random vectors and covariance matrices of a multivariate normal distribution can be interpreted as 1st and 2nd-order tensors. By adopting the above notation, the statement can be generalized to a tensor-variate normal distribution of a random 2nd-order tensor $\boldsymbol{X}$ with mean $\boldsymbol{M}$ and 4th-order covariance tensor $\boldsymbol{\mathcal{S}}$ given by

$$\mathcal{N}(\boldsymbol{X}|\boldsymbol{M}, \boldsymbol{\mathcal{S}}) = \frac{1}{\sqrt{(2\pi)^{\tilde{D}}|\boldsymbol{\mathcal{S}}|}}\ e^{-\frac{1}{2}(\boldsymbol{X}-\boldsymbol{M})\boldsymbol{\mathcal{S}}^{-1}(\boldsymbol{X}-\boldsymbol{M})}.$$

As we are interested in using GMM with full covariances to perform GMR on SPD manifolds, we have to compute tensor-variate normal density function with full covariance tensor of SPD matrices, where the computation of inverses and determinants are required. As $\boldsymbol{X} \in \mathcal{S}^D_{++}$ are symmetric matrices, our covariance tensor is *supersymmetric* and we can use the simplification proposed in [1] to compute its inverse and determinant. We next briefly summarize this approach.

When $\boldsymbol{X}$ is a symmetric 2nd-order tensor of dimension D (e.g. $\boldsymbol{X} \in \mathbb{R}^{D\times D}$ such that $\boldsymbol{X}_{ij} = \boldsymbol{X}_{ji}$), the covariance tensor $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{D\times D\times D\times D}$ inherits symmetries such that $\boldsymbol{\mathcal{S}}^{mn}_{ij} = \boldsymbol{\mathcal{S}}^{mn}_{ji} = \boldsymbol{\mathcal{S}}^{nm}_{ij}$ and $\boldsymbol{\mathcal{S}}^{mn}_{ij} = \boldsymbol{\mathcal{S}}^{ij}_{mn}$. Any 4th-order tensor $\boldsymbol{\mathcal{S}}$ satisfying those symmetry properties can be mapped into a

symmetric positive definite 2nd-order tensor $\boldsymbol{\Sigma} \in \mathbb{R}^{\tilde{D}\times\tilde{D}}$ with $\tilde{D} = D + D(D-1)/2$. Similarly, $\boldsymbol{X}$ can be written as a $\tilde{D}$-dimensional vector $\boldsymbol{x}$ such that the tensor contraction $\boldsymbol{X}\boldsymbol{\mathcal{S}}^{-1}\boldsymbol{X}$ is equal to $\boldsymbol{x}^\top\boldsymbol{\Sigma}^{-1}\boldsymbol{x}$, and such that the normal probability density function $\mathcal{N}(\boldsymbol{X}|\boldsymbol{M},\boldsymbol{\mathcal{S}})$ is equivalent to

$$\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^{\tilde{D}}|\boldsymbol{\Sigma}|}}\, e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}.$$

For simplicity, we take the example of $D = 3$, which can be generalized to higher dimensions. $\boldsymbol{X}$ is in this case written as a 6-dimensional column vector

$$\boldsymbol{x} = (\boldsymbol{X}_{ii},\ \boldsymbol{X}_{jj},\ \boldsymbol{X}_{kk},\ \sqrt{2}\boldsymbol{X}_{ij},\ \sqrt{2}\boldsymbol{X}_{ik},\ \sqrt{2}\boldsymbol{X}_{jk})^\top.$$

The 3-dimensional 4h-order tensor is converted to the 6-dimensional 2nd-order tensor by defining $\boldsymbol{\Sigma}$ as

$$\begin{pmatrix}
\boldsymbol{\mathcal{S}}^{ii}_{ii} & \boldsymbol{\mathcal{S}}^{jj}_{ii} & \boldsymbol{\mathcal{S}}^{kk}_{ii} & \sqrt{2}\boldsymbol{\mathcal{S}}^{ij}_{ii} & \sqrt{2}\boldsymbol{\mathcal{S}}^{ik}_{ii} & \sqrt{2}\boldsymbol{\mathcal{S}}^{jk}_{ii} \\
\boldsymbol{\mathcal{S}}^{jj}_{ii} & \boldsymbol{\mathcal{S}}^{jj}_{jj} & \boldsymbol{\mathcal{S}}^{kk}_{jj} & \sqrt{2}\boldsymbol{\mathcal{S}}^{ij}_{jj} & \sqrt{2}\boldsymbol{\mathcal{S}}^{ik}_{jj} & \sqrt{2}\boldsymbol{\mathcal{S}}^{jk}_{jj} \\
\boldsymbol{\mathcal{S}}^{kk}_{ii} & \boldsymbol{\mathcal{S}}^{kk}_{jj} & \boldsymbol{\mathcal{S}}^{kk}_{kk} & \sqrt{2}\boldsymbol{\mathcal{S}}^{ij}_{kk} & \sqrt{2}\boldsymbol{\mathcal{S}}_{kkik} & \sqrt{2}\boldsymbol{\mathcal{S}}^{jk}_{kk} \\
\sqrt{2}\boldsymbol{\mathcal{S}}^{ij}_{ii} & \sqrt{2}\boldsymbol{\mathcal{S}}^{ij}_{jj} & \sqrt{2}\boldsymbol{\mathcal{S}}^{ij}_{kk} & 2\boldsymbol{\mathcal{S}}^{ij}_{ij} & 2\boldsymbol{\mathcal{S}}^{ik}_{ij} & 2\boldsymbol{\mathcal{S}}^{jk}_{ij} \\
\sqrt{2}\boldsymbol{\mathcal{S}}^{ik}_{ii} & \sqrt{2}\boldsymbol{\mathcal{S}}^{ik}_{jj} & \sqrt{2}\boldsymbol{\mathcal{S}}^{ik}_{kk} & 2\boldsymbol{\mathcal{S}}^{ik}_{ij} & 2\boldsymbol{\mathcal{S}}^{ik}_{ik} & 2\boldsymbol{\mathcal{S}}^{jk}_{ik} \\
\sqrt{2}\boldsymbol{\mathcal{S}}^{jk}_{ii} & \sqrt{2}\boldsymbol{\mathcal{S}}^{jk}_{jj} & \sqrt{2}\boldsymbol{\mathcal{S}}^{jk}_{kk} & 2\boldsymbol{\mathcal{S}}^{jk}_{ij} & 2\boldsymbol{\mathcal{S}}^{jk}_{ik} & 2\boldsymbol{\mathcal{S}}^{jk}_{jk}
\end{pmatrix}. \tag{2}$$

Similarly to the eigenvalue decomposition of matrices, the eigenvalues λ and eigentensors $\boldsymbol{V} \in \mathbb{R}^{D\times D}$ of a 4th-order tensor $\boldsymbol{\mathcal{S}}$ satisfy the fundamental equation $\boldsymbol{\mathcal{S}}\boldsymbol{V} = \lambda\boldsymbol{V}$. They may also be found using the correspondence between $\boldsymbol{\mathcal{S}}$ and $\boldsymbol{\Sigma}$. First, the eigenvalues of the 4th-order covariance and the 2nd-order covariance are the same. Then, given $\boldsymbol{v} = (\boldsymbol{v}_{xx}, \boldsymbol{v}_{yy}, \boldsymbol{v}_{zzx}, \boldsymbol{v}_{xy}, \boldsymbol{v}_{xz}, \boldsymbol{v}_{yz})^\top$ one eigenvector of $\boldsymbol{\Sigma}$, the corresponding eigentensor of $\boldsymbol{\mathcal{S}}$ is

$$\boldsymbol{V} = \begin{pmatrix}
\boldsymbol{v}_{xx} & \frac{1}{\sqrt{2}}\boldsymbol{v}_{xy} & \frac{1}{\sqrt{2}}\boldsymbol{v}_{xz} \\
\frac{1}{\sqrt{2}}\boldsymbol{v}_{xy} & \boldsymbol{v}_{yy} & \frac{1}{\sqrt{2}}\boldsymbol{v}_{yz} \\
\frac{1}{\sqrt{2}}\boldsymbol{v}_{xz} & \frac{1}{\sqrt{2}}\boldsymbol{v}_{yz} & \boldsymbol{v}_{zz}
\end{pmatrix}. \tag{3}$$

The inverse and determinant of the covariance $\boldsymbol{\mathcal{S}}$ are then computed using the eigentensors and eigenvalues with

$$\boldsymbol{\mathcal{S}}^{-1} = \sum_k \lambda_k^{-1}\, \boldsymbol{V}_k \otimes \boldsymbol{V}_k, \text{ and } |\boldsymbol{\mathcal{S}}| = \sum_k \lambda_k.$$

V. REGRESSION ON SPD MANIFOLDS

We introduce here a parallel transport method for 4th-order full covariances, which will be used in Gaussian mixture regression (GMR). We then present extensions of Gaussian mixture model (GMM) and GMR to SPD manifolds. Finally, we show that our approach can be used for applications where input and output data belong to different manifolds.

A. Parallel Transport of 4th-Order Covariances

In the case of a Riemannian manifold where the elements are represented by vectors, covariance matrices can be *parallel transported* between tangent spaces by transporting their eigenvectors [14]. We extend this approach to SPD manifolds.

Our goal is to parallel transport the 4th-order covariance tensor $\boldsymbol{\mathcal{S}}$ from $\mathcal{T}_{\boldsymbol{\Sigma}}\mathcal{S}^D_{++}$ to $\mathcal{T}_{\boldsymbol{\Lambda}}\mathcal{S}^D_{++}$. The eigenvalues and eigentensors of $\boldsymbol{\mathcal{S}}$ are computed from the eigendecomposition of the low order covariance using (2) and (3). Let $\tilde{\boldsymbol{V}}_k = \Gamma_{\boldsymbol{\Sigma}\to\boldsymbol{\Lambda}}(\boldsymbol{V}_k)$ be the k-th parallel transported eigentensor with (1), and λ_k the k-th eigenvalue. The parallel transported 4th-order covariance tensor is then given by

$$\tilde{\boldsymbol{\mathcal{S}}} \;=\; \Gamma_{\boldsymbol{\Sigma}\to\boldsymbol{\Lambda}}(\boldsymbol{\mathcal{S}}) \;=\; \sum_k \lambda_k\, \tilde{\boldsymbol{V}}_k \otimes \tilde{\boldsymbol{V}}_k. \tag{4}$$

B. Gaussian Mixture Model on SPD manifolds

Similarly to multivariate distributions (see [8], [10], [18]), tensor-variate distributions maximizing the entropy in the tangent space can be approximated by

$$\mathcal{N}_{\mathcal{M}}(\boldsymbol{X}|\boldsymbol{M},\boldsymbol{\mathcal{S}}) = \frac{1}{\sqrt{(2\pi)^{\tilde{D}}|\boldsymbol{\mathcal{S}}|}}\, e^{-\frac{1}{2}\mathrm{Log}_{\boldsymbol{X}}(\boldsymbol{M})\,\boldsymbol{\mathcal{S}}^{-1}\,\mathrm{Log}_{\boldsymbol{X}}(\boldsymbol{M})},$$

where $\boldsymbol{X} \in \mathcal{M}$ is a point of the manifold, $\boldsymbol{M} \in \mathcal{M}$ is the origin of the tangent space, and $\boldsymbol{\mathcal{S}} \in \mathcal{T}_{\boldsymbol{M}}\mathcal{M}$ is the covariance tensor defined in this tangent space.

Similarly to the Euclidean case, a GMM on SPD manifold is defined by

$$p(\boldsymbol{X}) = \sum_{k=1}^{K} \pi_k \mathcal{N}_{\mathcal{M}}(\boldsymbol{X}|\boldsymbol{M}_k,\boldsymbol{\mathcal{S}}_k),$$

with K the number of components and π_k the mixing coefficients (priors) such that $\sum_k \pi_k = 1$. The parameters of the GMM on the manifold can be estimated by Expectation-Maximization (EM) algorithm, by extending the approach in [10] to SPD manifolds.

We first compute the responsibility of each component k in the E-step with

$$p(k|\boldsymbol{X}_i) = \frac{\pi_k\, \mathcal{N}_{\mathcal{M}}(\boldsymbol{X}_i|\boldsymbol{M}_k,\boldsymbol{\mathcal{S}}_k)}{\sum_{j=1}^{K} \pi_j\, \mathcal{N}_{\mathcal{M}}(\boldsymbol{X}_i|\boldsymbol{M}_j,\boldsymbol{\mathcal{S}}_j)},$$
$$N_k = \sum_{i=1}^{N} p(k|\boldsymbol{X}_i).$$

We then update the mean $\boldsymbol{M}_k$, the covariance tensor $\boldsymbol{\mathcal{S}}_k$ and the prior π_k for each component during the M-step with

$$\boldsymbol{M}_k \leftarrow \frac{1}{N_k}\mathrm{Exp}_{\boldsymbol{M}_k}\Big(\sum_{i=1}^{N} p(k|\boldsymbol{X}_i)\ \mathrm{Log}_{\boldsymbol{M}_k}(\boldsymbol{X}_i)\Big),$$
$$\boldsymbol{\mathcal{S}}_k \leftarrow \frac{1}{N_k}\sum_{i=1}^{N} p(k|\boldsymbol{X}_i)\ \mathrm{Log}_{\boldsymbol{M}_k}(\boldsymbol{X}_i) \otimes \mathrm{Log}_{\boldsymbol{M}_k}(\boldsymbol{X}_i),$$
$$\pi_k \leftarrow \frac{N_k}{N}.$$

Note that at each M-step, the update of the means $\boldsymbol{M}_k$ is performed iteratively until convergence, and $\boldsymbol{\mathcal{S}}_k$ is computed once, after convergence of $\boldsymbol{M}_k$.

C. Gaussian Mixture Regression (GMR) on SPD manifolds

GMR computes the conditional distribution $p(\boldsymbol{X}_{\mathcal{O}\mathcal{O}}|\boldsymbol{X}_{\mathcal{I}\mathcal{I}})$ of the joint distribution $p(\boldsymbol{X})$. We denote the block decom-

position of the datapoints, means and covariances as

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{X}_{\mathcal{I}\mathcal{I}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{X}_{\mathcal{O}\mathcal{O}} \end{pmatrix}, \boldsymbol{M} = \begin{pmatrix} \boldsymbol{M}_{\mathcal{I}\mathcal{I}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{M}_{\mathcal{O}\mathcal{O}} \end{pmatrix},$$

$$\boldsymbol{\mathcal{S}} = \left(\begin{array}{cc|cc} \boldsymbol{\mathcal{S}}^{\mathcal{I}\mathcal{I}}_{\mathcal{I}\mathcal{I}} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\mathcal{S}}^{\mathcal{O}\mathcal{O}}_{\mathcal{I}\mathcal{I}} & \boldsymbol{0} & \boldsymbol{0} \\ \hline \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{\mathcal{S}}^{\mathcal{I}\mathcal{I}}_{\mathcal{O}\mathcal{O}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{\mathcal{S}}^{\mathcal{O}\mathcal{O}}_{\mathcal{O}\mathcal{O}} \end{array}\right).$$

With this decomposition, manifold functions are applied individually on input and output parts, e.g. $\text{Exp}_{\boldsymbol{M}_k}(\boldsymbol{X})$ is equivalent to $\begin{pmatrix} \text{Exp}_{\boldsymbol{M}_{\mathcal{I}\mathcal{I}}}(\boldsymbol{X}_{\mathcal{I}\mathcal{I}}) & \boldsymbol{0} \\ \boldsymbol{0} & \text{Exp}_{\boldsymbol{M}_{\mathcal{O}\mathcal{O}}}(\boldsymbol{X}_{\mathcal{O}\mathcal{O}}) \end{pmatrix}$.

Similarly to GMR in Euclidean space [7] and in manifolds where data are represented by vectors [8], GMR on SPD manifold approximates the conditional distribution by a single Gaussian $p(\boldsymbol{X}_{\mathcal{O}\mathcal{O}}|\boldsymbol{X}_{\mathcal{I}\mathcal{I}}) \sim \mathcal{N}(\hat{\boldsymbol{M}}_{\mathcal{O}\mathcal{O}}, \hat{\boldsymbol{\mathcal{S}}}^{\mathcal{O}\mathcal{O}}_{\mathcal{O}\mathcal{O}})$.

The mean $\hat{M}_{\mathcal{O}\mathcal{O}}$ is computed iteratively until convergence with

$$\boldsymbol{\Delta}_k = \text{Log}_{\hat{\boldsymbol{M}}_{\mathcal{O}\mathcal{O}}}(\boldsymbol{M}_{\mathcal{O}\mathcal{O},k}) - \tilde{\boldsymbol{\mathcal{S}}}^{\mathcal{I}\mathcal{I}}_{\mathcal{O}\mathcal{O},k} \, \tilde{\boldsymbol{\mathcal{S}}}^{\mathcal{I}\mathcal{I}-1}_{\mathcal{I}\mathcal{I},k} \, \text{Log}_{\hat{X}_{\mathcal{I}\mathcal{I}}}(\boldsymbol{M}_{\mathcal{I}\mathcal{I},k}),$$

$$\hat{\boldsymbol{M}}_{\mathcal{O}\mathcal{O}} \leftarrow \text{Exp}_{\hat{\boldsymbol{M}}_{\mathcal{O}\mathcal{O}}}\Big(\sum_k h_k \boldsymbol{\Delta}_k\Big),$$

where h_k are the responsibilities of the GMM component k computed with

$$h_k = \frac{\pi_k \, \mathcal{N}(\boldsymbol{X}_{\mathcal{I}\mathcal{I}}|\boldsymbol{M}_{\mathcal{I}\mathcal{I},k}, \boldsymbol{\mathcal{S}}^{\mathcal{I}\mathcal{I}}_{\mathcal{I}\mathcal{I},k})}{\sum_{j=1}^{K} \pi_j \, \mathcal{N}(\boldsymbol{X}_{\mathcal{I}\mathcal{I}}|\boldsymbol{M}_{\mathcal{I}\mathcal{I},j}, \boldsymbol{\mathcal{S}}^{\mathcal{I}\mathcal{I}}_{\mathcal{I}\mathcal{I},j})},$$

and $\tilde{\boldsymbol{\mathcal{S}}}$ is the parallel transported covariance tensor

$$\tilde{\boldsymbol{\mathcal{S}}} = \Gamma_{\boldsymbol{M} \rightarrow \hat{\boldsymbol{X}}}(\boldsymbol{\mathcal{S}}) \quad \text{with} \quad \hat{\boldsymbol{X}} = \begin{pmatrix} \boldsymbol{X}_{\mathcal{I}\mathcal{I}} & \boldsymbol{0} \\ \boldsymbol{0} & \hat{\boldsymbol{M}}_{\mathcal{O}\mathcal{O}} \end{pmatrix},$$

computed with (4). The covariance $\hat{\boldsymbol{\mathcal{S}}}^{\mathcal{O}\mathcal{O}}_{\mathcal{O}\mathcal{O}}$ defined in the tangent space of the estimated center can then be estimated with

$$\hat{\boldsymbol{\mathcal{S}}}^{\mathcal{O}\mathcal{O}}_{\mathcal{O}\mathcal{O}} = \sum_k h_k \left(\tilde{\boldsymbol{\mathcal{S}}}^{\mathcal{O}\mathcal{O}}_{\mathcal{O}\mathcal{O},k} - \tilde{\boldsymbol{\mathcal{S}}}^{\mathcal{I}\mathcal{I}}_{\mathcal{O}\mathcal{O},k} \tilde{\boldsymbol{\mathcal{S}}}^{\mathcal{I}\mathcal{I}-1}_{\mathcal{I}\mathcal{I},k} \tilde{\boldsymbol{\mathcal{S}}}^{\mathcal{O}\mathcal{O}}_{\mathcal{I}\mathcal{I},k} + \boldsymbol{\Delta}_k \otimes \boldsymbol{\Delta}_k\right).$$

D. Extension of GMR to Other Manifolds

In many applications, the input and output data do not belong to the same manifold, e.g. [1], [4]. Gaussian mixture regression as presented in previous sections can be used with input or output data in the form of scalars or vectors by defining the corresponding part of $\boldsymbol{X}$ either as a scalar with $\boldsymbol{X}_{\mathcal{I}\mathcal{I}} = x_{\mathcal{I}}$, or as a diagonal matrix by placing the elements of the vector in the main diagonal with $\boldsymbol{X}_{\mathcal{I}\mathcal{I}} = \text{diag}(\boldsymbol{x}_{\mathcal{I}})$, see for example [19]. Data on several manifolds can also be combined to form the input or output parts of $\boldsymbol{X}$. The mapping functions and the parallel transport applied on the input or output part of the datapoint $\boldsymbol{X}$, centers $\boldsymbol{M}$ and covariance tensors $\boldsymbol{\mathcal{S}}$ must be adapted consequently.

VI. APPLICATION

Electroencephalography (EEG) or electromyography (EMG) features are often represented in the form of covariances matrices as this method reduces the impact of noisy samples and is straightforward to fuse different features [20]. We focus here on the application of sEMG in the context of prosthetic hand control. In order to test the applicability and accuracy of the presented technique, we applied it to the Ninapro database [21].

We give here a short description of the data and we describe how the proposed method is applied to sEMG. We then present the results and compare the performance of GMR on SPD manifold with classical GMR.

A. Data Description

The second iteration of the publicly available Ninapro database [21] comprises recordings from 40 able-bodied subjects performing 50 movements including isometric hand configurations, functional movements and grasping of common household objects. Each movement was repeated 6 times. Muscular activity was recorded using 12 state-of-the-art surface electromyography (sEMG) sensors sampled at 2 kHz. A 22-sensor data glove was used to record hand kinematics activity.

In this experiment, we focused on the estimation of three wrist movements: extension, flexion and supination. In the context of prosthetics, we are interested in predicting which movement is performed but also the transitions between movements, i.e. transitions from rest poses to wrist movements and from wrist movements to rest poses.

All sEMG channels were standardized to have zero mean and unit variance, based on statistics computed on data from the training set. Dimensionality reduction was then performed using principal component analysis (PCA). Spatial covariance matrices were segmented using a sliding window of 400 ms with an increment of 10 ms. The output for each spatial covariance is a continuous 4-dimensional vector $\boldsymbol{y} = (y_{\text{rest}}, y_{\text{sup}}, y_{\text{ext}}, y_{\text{flex}})^\top$ where $y_i = \frac{N_i}{N} \in [0, 1]$, with N_i the number of sample with label i in the sliding window, and N the total number of samples of the sliding window.

The data for each subject are split into training and testing sets based on repetitions: the second and fifth repetitions for each movement were used for testing while the training set is comprised of the remaining repetitions. Both training and testing sets were subsampled at interval of 10 samples in order to ensure computational feasibility.

B. Results

We applied GMR on SPD manifold to predict wrist movements from spatial covariances computed with sEMG data. We compared the results with those obtained by applying GMR directly on reduced sEMG data in Euclidean space. The different models were initialized using K-means and the numbers of Gaussians were determined using cross-validation applied on the training set.

Table I shows three examples of typical average and standard deviation of the root-mean-square error (RMSE) values obtained by applying GMR on SPD manifold and on Euclidean space for each wrist movement. Each model was trained three times with different initializations. GMR on SPD manifold was more effective than GMR on Euclidean space for most of the 40 subjects: on average, it reduces the error by 6%, 4%, 1% and 1% for rest, wrist supination,

TABLE I: Root mean square error (RMSE) for three participants and each wrist movement.

	Rest	Wr. supination	Wr. extension	Wr. flexion
$\mathcal{S}^D_{++}$	0.29 ± 0.00	0.18 ± 0.00	0.25 ± 0.00	0.27 ± 0.00
$\mathbb{R}$	0.47 ± 0.00	0.31 ± 0.00	0.33 ± 0.00	0.33 ± 0.00
$\mathcal{S}^D_{++}$	0.32 ± 0.02	0.29 ± 0.14	0.36 ± 0.07	0.43 ± 0.13
$\mathbb{R}$	0.46 ± 0.00	0.34 ± 0.00	0.35 ± 0.00	0.35 ± 0.00
$\mathcal{S}^D_{++}$	0.36 ± 0.02	0.22 ± 0.01	0.31 ± 0.00	0.29 ± 0.00
$\mathbb{R}$	0.42 ± 0.00	0.42 ± 0.00	0.43 ± 0.00	0.43 ± 0.00

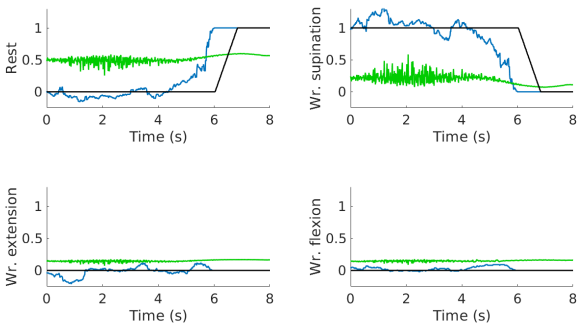


Fig. 4: Comparison between the movements predicted by GMR on SPD manifold (blue line) and GMR in Euclidean space (green line) during a transition from wrist supination to rest. The reference is the black line. We can see that GMR on SPD manifold is more efficient at predicting a transition than GMR in Euclidean space that tends to favor rest poses in most of the cases.

extension and flexion, respectively. However, the error reduction can attain 25% for some subjects and movements. The outputs found by GMR in Euclidean space tend to vary only a little compared to the real value. The expected output is often dominated by the rest pose, whose value is high compared to the others. GMR on SPD manifold is also generally more efficient than GMR in Euclidean space to predict transitions between movements (see Fig. 4).

VII. CONCLUSION

In this paper, we showed how Gaussian mixture model (GMM) and Gaussian mixture regression (GMR) can be extended to the manifold of symmetric positive definite matrices. We exploited the fact that the 4th-order covariance of a symmetric matrix inherits useful symmetry properties. This allows us to reduce it to a 2nd-order tensor to find its eigentensors and design a parallel transport scheme exploiting this structure. Our approach is flexible as it allows us to easily combine data in the form of scalar, vector or SPD matrices on different manifolds. It is extensible to other Riemannian manifolds where the data are represented in the form of matrices, as long as a method is available to find the eigentensors of their covariance.

We experimented the approach in the context of prosthetic hands, with the regression of wrist movements from sEMG data in the form of spatial covariance. The proposed method improved the detection of wrist movement for most of the subjects and proved to be efficient to detect transition between movements. In future work, we plan to extend the regression problem to fingers movements. We will also explore other forms of covariance features typically found in sEMG analysis, such as root mean square (RMS) or sEMG histogram (HIST) features.

REFERENCES

[1] P. Basser and S. Pajevic, "Spectral decomposition of a 4th-order covariance tensor: Applications to diffusion tensor MRI," *Signal Processing*, vol. 87, no. 2, pp. 220–236, 2007.

[2] M. Han and F. C. Park, "DTI segmentation and fiber tracking using metrics on multivariate normal distributions," *Journal of Mathematical Imaging and Vision*, vol. 49, pp. 317–334, 2014.

[3] A. Tyagi and J. W. Davis, "A recursive filter for linear systems on Riemannian manifolds," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Anchorage, Alaska, June 2008, pp. 1–8.

[4] F. Yger, F. Lotte, and M. Sugiyama, "Averaging covariance matrices for EEG signal classification based on the CSP: an empirical study," in *Proc. European Signal Processing Conf. (EUSIPCO)*, Nice, France, 2015, pp. 2721–2725.

[5] M. Moakher and P. Batchelor, "Symmetric positive-definite matrices: From geometry to applications and visualization," *Visualization and Processing of Tensor Fields*, pp. 285–298, 2006.

[6] G. Meyer, S. Bonnabel, and R. Sepulchre, "Regression on fixed-rank positive semidefinite matrices: a Riemannian approach," *Journal of Machine Learning Research*, vol. 12, pp. 593–625, 2012.

[7] S. Calinon, "A tutorial on task-parametrized movement learning and retrieval," *Intelligent Service Robotics*, vol. 9, no. 1, pp. 1–29, 2016.

[8] M. J. A. Zeestraten, I. Havoutis, J. Silvério, S. Calinon, and D. G. Caldwell, "An approach for imitation learning on Riemannian manifolds," *IEEE Robotics and Automation Letters (RA-L)*, vol. 2, no. 3, pp. 1240–1247, June 2017.

[9] Z. Ghahramani and M. Jordan, "Supervised learning from incomplete data via an EM approach," *Advances in neural information processing systems*, vol. 6, pp. 120–127, 1994.

[10] E. Simo-Serra, C. Torras, and F. Moreno-Noguer, "3D human pose tracking priors using geodesic mixture models," *International Journal of Computer Vision*, vol. 122, no. 2, pp. 388–408, 2017.

[11] S. Kim, R. Haschke, and H. Ritter, "Gaussian mixture model for 3-DoF orientations," *Robotics and Autonomous Systems*, vol. 87, pp. 28–37, 2017.

[12] S. Said, L. Bombrun, Y. Berthoumieu, and J. H. Manton, "Riemannian Gaussian distributions on the space of symmetric positive definite matrices," *IEEE Trans. on Information Theory*, vol. 63, no. 4, pp. 2153–2170, 2017.

[13] X. Zhan and B. Ma, "Gaussian mixture model on tensor field for visual tracking," *IEEE Signal Processing Letters*, vol. 19, no. 11, pp. 733–736, 2012.

[14] O. Freifeld, S. Hauberg, and M. J. Black, "Model transport: Towards scalable transfer learning on manifolds," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Columbus, OH, USA, June 2014, pp. 1378–1385.

[15] P. A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2007.

[16] X. Pennec, P. Fillard, and N. Ayache, "A Riemannian framework for tensor computing," *Intl Journal on Computer Vision*, vol. 66, no. 1, pp. 41–66, 2006.

[17] L. Lathauwer, B. Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM Review*, vol. 21, no. 4, pp. 1253–1278, 1999.

[18] G. Dubbelman, "Intrinsic statistical techniques for robust pose estimation," PhD thesis, University of Amsterdam, Netherlands, 2011.

[19] L. Rozo, N. Jaquier, S. Calinon, and D. G. Caldwell, "Learning manipulability ellipsoids for task compatibility in robot manipulation," in *Proc. IEEE/RSJ Intl Conf. on Intelligent Robots and Systems (IROS)*, Vancouver, Canada, September 2017.

[20] M. Harandi, C. Sanderson, A. Sanin, and B. Lovell, "Spatio-temporal covariance descriptors for action and gesture recognition," in *Proc. of IEEE Workshop on Applications of Computer Vision (WACV)*, Clearwater, USA, 2013.

[21] M. Atzori, A. Gijsberts, C. Castellini, B. Caputo, A. G. M. Hager, S. Elsig, G. Giatsidis, B. F., and H. Müller, "Electromyography data for non-invasive naturally-controlled robotic hand prostheses," *Scientific Data*, vol. 1, no. 140053, pp. 1–13, 2014.