Journal of AI and Data Mining
Vol 6, No 2, 2018, 321-334
DOI: 10.22044/JADM.2017.1000

A Geometry Preserving Kernel over Riemannian Manifolds

Kh. Sadatnejad, S. Shiry Ghidary* and M. rahmati

Computer Engineering & Information Technology, Amirkabir University of Technology, Tehran, Iran.

Received 02 November 2015; Revised 01 March 2017; Accepted 08 April 2017
**Corresponding author: Shiry@aut.ac.ir (S. Shiry).*

Abstract
Kernel trick and projection to tangent spaces are two choices for linearizing the data points lying on Riemannian manifolds. These approaches are used to provide the pre-requisites for applying the standard machine learning methods on Riemannian manifolds. Classical kernels implicitly project data to a high-dimensional feature space without considering the intrinsic geometry of the data points. Projection to tangent spaces truly preserves topology along radial geodesics. In this paper, we propose a method for extrinsic inference on Riemannian manifold based on the kernel approach. We show that computing the Gramian matrix using geodesic distances, on a complete Riemannian manifold with unique minimizing geodesic between each pair of points, provides a feature mapping that is proportional with the topology of data points in the input space. The proposed approach is evaluated on real datasets composed of EEG signals of patients with two different mental disorders, texture, and visual object classes. To assess the effectiveness of our scheme, the extracted features are examined by other state-of-the-art techniques for extrinsic inference over symmetric positive definite (SPD) Riemannian manifold. The experimental results obtained show the superior accuracy of the proposed approach over approaches that use the kernel trick to compute similarity on SPD manifolds without considering the topology of dataset or partially preserving the topology.

Keywords: *Kernel Trick, Riemannian Manifold, Geometry Preservation, Gramian Matrix.*

1. Introduction

Many problems in computer vision and signal processing lead to handling non-linear manifolds. Two different approaches in analysis over manifolds are reported in the literature. In one approach, the data points lie on a non-linear manifold that is embedded in R^n. The other approach corresponds to the cases where the data points do not form a vector space but lie on a non-linear manifold with a known structure. In the former approach, the structure of manifolds is unknown; therefore, the manifolds are modeled by graph, and the geodesic distances are approximated by the shortest path on the graph. The manifold learning techniques such as locally linear embedding (LLE) [45], Hessian LLE (HLLE) [43], local tangent space alignment (LTSA) [44], Laplacian eigenmap (LE) [46], non-negative patch alignment framework (NPAF) [47], and Isomap [49] are some methods of this approach that try to extract low-dimensional manifold from high-dimensional data while the topological structure of the manifold is preserved. The difference between these methods is in the geometrical property that they try to preserve. The latter approach that appears in many problems of computer vision consists of analysis over manifolds with well-studied geometries. The exact geometry of these manifolds can be achieved by closed-form formulae for the Riemannian operations [36]. Orthogonal matrices that form Grassmann manifold, 3D rotation matrices that form a special orthogonal group (SO(3)), and normalized histograms that form unit n-sphere (S^n) are some instances of the latter approach. The symmetric positive definite (SPD) matrices are another example that form a Riemannian manifold. Covariance region descriptors [1, 3, 5, 6, 9, 23, 25, 26, 28, 30], diffusion tensors [15], and structure tensors [36] provide SPD matrices in the computer vision and signal processing applications.

Since SPD matrices can be formulated as a Riemannian manifold [5], classical machine-learning methods that assume data points form a vector space have to deal with some challenges to be applicable on this manifold. Projecting manifold data points to tangent spaces using Riemannian *log* map [5] and embedding into Reproducing Kernel Hilbert Space (RKHS) using kernel functions [3, 7, 35] are two existing approaches in the literature to address the above issue. The Riemannian logarithmic map projects points lying over the manifold to the Euclidean space; therefore, the Euclidean-based learning techniques can be applied to the manifold data points. Iterative projections by Riemannian exponential and logarithmic map in this approach impose computational load to the learning process. On the other hand, approximating true geodesic distance between manifold points using associated Euclidean distance in tangent space preserves the manifold structure partially.

To overcome these limitations, using the kernel, the latter approach is applied to implicitly map manifold points into RKHS using the kernel function. The classical kernel functions do not consider the topology of data points on the manifold. Using the Euclidean distance in computing dissimilarities on manifolds may corrupt the intrinsic geometry of manifolds in feature space.

Harandi et al. [7] and Jayasumana et al. [35] considered the geometry of the manifold of SPD matrices by computing the similarities based on the geodesic distances. Using Gaussian kernel based on distances computed using different Riemannian metrics is the proposed approach in these two research works. The drawback of this approach is missing the non-linear structure of the data points in the feature space resulted by Gaussian kernel.

Vemulapalli et al. [52], Wang et al. [53], and Huang et al. [54] addressed the issue of learning over Riemannian manifold as a kernel-learning and metric-learning problem. All the proposed approaches are based on projecting all the data points in a single tangent space using the Riemannian log map. Vemulapalli et al. [52] considered the topology of data points in input space and their discrimination in feature space in the kernel-learning process. The base kernels that they applied in the learning process were based on projecting all the points in a single tangent space. In addition, using LEM_RBF [52] as a base kernel in their proposed approach leads to a non-linear feature space, while the geometry of the feature space is not considered in their proposed approach.

The Wang et al.'s proposed approach [53] for learning over SPD manifold is relied on projecting the data points to a tangent space and using linear discriminant analysis and partial least square in the resulting Euclidean space.

Huang et al. [54] addressed the learning over Riemannian manifold as a metric learning problem. They projected all the data points in a single tangent space, and then projected the data points in another Euclidean space with more discriminability.

All these methods inherit the shortcomings of projection to a tangent space approach.

Due to the smooth changes of labels on the manifolds that were confirmed by the compactness hypothesis, preserving the topology of manifolds in projection to Euclidean space is effective on the efficiency of the classical learning methods. Therefore, in this work, we try to provide the pre-requisites for applying the classical machine-learning methods on SPD manifolds by learning a kernel that preserves the geometry of manifolds. The concept of preserving geometry may incorrectly suggest manifold learning techniques. Since the main challenge of manifold learning techniques is preserving geometry, to clarify the distinction between geometry based kernel on SPD manifold and manifold learning techniques on a non-linear manifold with specified geometry, in this work, some experiments were done on the SPD manifold.

The main contribution of this paper is to introduce an appropriate base kernel over the manifold of SPD matrices with the aim of considering the topology of data points in input space and its geometry in feature space. We use the properties of SPD Riemannian manifolds in the proposed kernel. The exact geodesic distance between any two points is computable using Riemannian metric. We compute Gramian matrix of projections at feature space. This method uses the geodesic distance to preserve the topology of data points in the feature space, the same as topology on the manifold. All kernel-based methods that are formulated based on the inner product of samples are applicable to implicit feature space by applying Gramian matrix instead of explicit coordinate of samples. The proposed kernel over SPD manifold is used for extrinsic inference.

This paper is organized as what follows. The related literature is reviewed in section 2. In section 3, we review the mathematical preliminaries that are required to become familiar with Riemannian geometry. In section 4, we describe our contribution for providing the pre-requisites for learning over the SPD Riemannian manifold including computing the Gramian matrix of training data and its generalization to test samples. The experiments on real datasets are presented in section 5, and are discussed in section 6. Finally, we conclude this paper in section 7.

2. Related works

There is a rich literature regarding kernel learning and also manifold learning. A thorough review on these topics is beyond the scope of this paper. Recently, different useful applications have used covariance matrices for describing objects. These applications lead to applying machine-learning methods on an SPD manifold. In this study, we review some research works that rely on learning on SPD manifold.

As mentioned in section 1, learning on Riemannian manifolds relies on transferring the manifold data points to a vector space [3, 5, 7]. At the approach that linearization is done by mapping tangent spaces using the Riemannian *log* map, the true geodesic distance between the points lying on different radial geodesics would not be preserved. Therefore, the intrinsic geometry is not preserved completely in projection to the tangent space. Porikli et al. [5, 27, 29, 31] applied the ensemble-based techniques to overcome the weakness of projection to tangent space for classifying the data lying on the SPD Riemannian manifold. Computing geometric mean that is the base point of weak learners imposes a computational load to the learner. Barachant et al. [9] projected the data points to the tangent space at global geometric mean, and then used classical classifiers for discrimination. It is obvious that mapping all points to a single tangent space in the case that all the data points do not lie on the same radial geodesic cannot preserve the global topology of the dataset, and may bring poor results. In another research work, Barachant et al. [3] used a combination of two existing approaches for linearizing Riemannian manifolds. They applied a kernel [55] that was based on the geometry of the data, and examined it in BCI application. They applied Riemannian metric to compute the inner product in the tangent space at geometric mean. Unfortunately, in the case that the data points are mapped globally to a single tangent space, the inner product between points on different geodesics are not induced from the true geodesic distance between them and depends on the base point. Therefore, the implicit mapping of their proposed kernel can change the intrinsic topology of the manifold. Harandi et al. [7] proposed a kernel that applied a true geodesic distance between points to compute the inner product in the Hilbert space. Applying an exponential map with an arbitrary bandwidth was their choice in computing the inner product. Sensitivity to kernel's bandwidth [2] and choosing this kernel without fine tuning of its parameter can change the geometry of the dataset in feature space such that degrade the performance. Since the proposed kernel puts the data points on the surface of a sphere, applying the methods that rely on Euclidean metric can bring poor results in the resulting non-linear feature space. Early research works show that considering the geometry of data points in feature space can improve the accuracy of classification [32]. A traditional example of using kernel for linearization is kernel PCA. Applying kernel PCA as a method for dimensionality and noise reduction on non-linear data points relies on the assumption that the data points are flattened in feature space using the kernel function. The kernel type and its parameters are arbitrary and mainly motivated by the hope that the induced mapping linearizes the underlying manifold [8]. Since the geometrical interpretation of the various kernels is difficult, and strongly depends on its parameters, applying inappropriate kernels may cause unfortunate results [2], [34]. In the case that the local principal components of the feature space is not in the direction of global principal components of full manifold, the kernels do not linearize accurately; therefore, poor results are obtained. For example, Gaussian kernel, as defined in (1), brings a non-linear feature space. It puts the data points on the surface of a sphere and modifies the Euclidean distance in such a way that the samples that are far apart become orthonormal, and the points that are very close to each other tend to lie on the same point.

$$K(X_i, X_j) = \exp(-\| X_i - X_j \|_2^2 / \sigma^2) \qquad (1)$$

By changing the value of the variance parameter of Gaussian kernel, the geometry of the feature space changes accordingly [2]. Since the actual geometry of data points may not be preserved through linearization by this kernel, the learners that are trained at the transformed space may bring poor results [2], [8].

The weakness of projection to tangent space in mapping to Euclidean space, and the drawbacks of classical kernels show the necessity of proposing appropriate techniques for linearizing non-linear manifolds with a known structure. The compactness hypothesis that states similar objects has a close representation, and smooth changes of labels over manifold are our motivations for preserving geometry in projection to feature space.

3. Background

In this section, we review some basic concepts in Riemannian geometry that are necessary for reading the paper. We introduce the metric, which is used on SPD matrix space in this paper and its associated $\log$ and $\exp$ map.

3.1. Mathematical preliminaries

A homeomorphism is a continuous bijective map whose inverse is continuous. A topological manifold is a connected Hausdorff space that for every point of the manifold, there is a neighborhood U, which is homeomorphic to an open subset V of R^d. The homeomorphism between these two sets U and $(\phi : U \to V)$ is called a (coordinate) chart. A family of charts that provides an open-covering of the manifold is called an atlas, $\{U_\alpha, \phi_\alpha\}$. A differentiable manifold is a manifold with an atlas such that all transitions between the coordinate charts are differentiable of class C^∞.

$$\phi_\beta \circ \phi_\alpha^{-1} : \phi_\alpha(U_\alpha \cap U_\beta) \to \phi_\beta(U_\alpha \cap U_\beta) \qquad (2)$$

where, ϕ_β and ϕ_α are the coordinate charts corresponding to the U_α and U_β neighborhoods on the manifold. A Riemannian manifold $(\mathrm{M,g})$ is a differentiable manifold M that is endowed with a smooth inner product (Riemannian metric $g(u,v)$) on each tangent space $T_X M$. The inner product (Riemannian metric) in Riemannian manifolds is a metric that allows measuring similarity or dissimilarity of two points on the manifold [11, 12, 17].

A curve $\gamma : I \subset R \to M$ is a geodesic if the rate of change of $\dot{\gamma}$ has no component along the manifold for all $t \in I$ or $\ddot{\gamma}$ is 0 [22]. Given a vector v in the tangent space $T_X M$, there is a geodesic $\gamma(t)$ that is characterized by its length, where geodesic issued from $\gamma(0) = X$, and $\dot{\gamma} = v / \| v \|$. Two points on the manifold may have multiple geodesic between them but the one that minimizes the length is called the minimizing geodesic. In a geodesically complete manifold, each pair of points admits minimizing geodesic. Minimizing geodesic between points may not be unique [22].

The exponential map, $\exp_X(v)$, maps a tangent vector $v \in T_X M$ into a point Y on the manifold. Its inverse is called logarithm map, $\log_X(Y)$, which maps a point on the manifold to a point at tangent space.

The point lying on the geodesic that passes through X with tangent vector v has $dist(X,Y) = \| v \| = <v,v>^{1/2}$.

The radial geodesics are all the geodesics that pass through X. Normal coordinates with center X is the local coordinates defined by the chart $(U, \exp_X^{-1})$. Normal coordinates can preserve the distances on radial geodesics. For example, a sphere that is unfolded onto a plane in normal coordinates can preserve the distances on great circles [13, 19, 22].

3.2. Mappings and distance in SPD matrix space

In this paper, we use the covariance matrices as the descriptors of data points. The Riemannian metric, exponential and logarithm map, and geodesic distance on symmetric positive definite matrix space are defined as what follow.

An invariant Riemannian metric or inner product on the tangent space of the symmetric positive definite matrices is defined as ([14, 15, 24]):

$$< y, z >_X = trace(X^{-1/2} y X^{-1} z X^{-1/2}) \qquad (3)$$

where, y and z are two tangent vectors in the tangent space formed at X point over Riemannian manifold. The Riemannian exponential map is defined as:

$$\exp_X(y) = X^{1/2} \exp(X^{-1/2} y X^{-1/2}) X^{1/2} \qquad (4)$$

where, y is a tangent vector and X is a base point over the manifold. The Riemannian $\log$ map on a point on the Riemannian manifold is defined as:

$$\log_X(Y) = X^{1/2} \log(X^{-1/2} Y X^{-1/2}) X^{1/2} \qquad (5)$$

where, X and Y are two points on the manifold, and matrix exponential and logarithm are calculated as:

$$\exp \Sigma = \sum_{k=0}^{\infty} \Sigma^k / k! = U \exp(D) U^T, \Sigma = UDU^T \qquad (6)$$

$$\log \Sigma = \sum_{k=1}^{\infty} (-1)^{k-1} (\Sigma - I)^k / k = U \log(D) U^T,$$
$$\Sigma = UDU^T$$

In (6), it is assumed that Σ is decomposed into eigenvalues and vectors. Note that the $\exp$

operator on matrices always exists, while the $\log$ operator is only defined on symmetric matrix with positive eigenvalues [24].
The distance between two points on SPD manifold associated with the Riemannian metric is computed by:

$$d_G^2(X,Y) = <\log_X(Y), \log_X(Y)>_X \quad (7)$$
$$= trace(\log^2(X^{-1/2}YX^{-1/2}))$$

In the tensor space with the metric (3), there is one and only one minimizing geodesic between any two tensors. The Riemannian $\log$ map is defined uniquely at all points on the manifold, and the exponential map is global diffeomorphism [8, 15].

4. Global geometry preserving kernel

In this section, we describe our method for providing the pre-requisites for learning in the space of SPD matrices using the properties of Riemannian manifolds. This mapping implicitly transfers the data points to a vector space, while the intrinsic geometry of the dataset is preserved by preserving the geodesic distances. First, we describe the proposed algorithm, which is used to compute the Gramian matrix of a set of points on the SPD Riemannian manifold at an implicit linearized space, and then investigate its generalization to unseen cases. We call the proposed kernel GGPK, which is the abbreviation of the global geometry preserving kernel.

4.1. Flattening an SPD Riemannian manifold

Let $P = \{X_i\}_{i=1}^N$ be the set of points on a Riemannian manifold. The geodesic distance between two points X_i and X_j on Riemannian manifold is computed by mapping to tangent space at one of these points and computing the length of the tangent vector that joins $\log_{X_i}(X_i)$ to $\log_{X_i}(X_j)$, which is given in (7). Assume that the pairwise squared geodesic distances stored in an N-by-N matrix D_G is given as:

$$D_G = [d_G^2(X_i, X_j)]_{1 \le i,j \le N} \quad (8)$$

where, d_G denotes the geodesic distance between two points on the manifold. The symmetric positive definite matrix space with the associated metric is a geodesically complete manifold, and has the structure of a curved vector space [14]. Satisfaction of the manifold assumption implies that defining geometry based on distance along the manifold and preserving it in feature space can bring appropriate projection for classification. Therefore, the distance between the two points $\phi(X_i)$ and $\phi(X_j)$ in the feature space is defined as:

$$d_E^2(\phi(X_i), \phi(X_j)) = \| \phi(X_i) - \phi(X_j) \|_2^2 \quad (9)$$
$$= d_G^2(X_i, X_j),$$
$$D_E \leftarrow D_G$$

where, X_i and X_j are the points on the manifold, ϕ is an implicit feature mapping from SPD Riemannian manifold to a Euclidean space for developable manifolds or a pseudo-Euclidean space for non-developable manifolds, d_G denotes the geodesic distance on the manifold, and d_E denotes the Euclidean distance in the feature space, which is L_2 norm of dissimilarity. D_G denotes a matrix of geodesic distances on SPD manifold that is assigned to the matrix of Euclidian distances between points in the feature space, D_E. This assignment is done implicitly using the kernel function. We recall that:

$$\| \phi(X_i) - \phi(X_j) \|_2^2 = \quad (10)$$
$$< \phi(X_i) - \phi(X_j), \phi(X_i) - \phi(X_j) > =$$
$$< \phi(X_i), \phi(X_i) > + < \phi(X_j), \phi(X_j) >$$
$$- 2 < \phi(X_i) - \phi(X_j) >$$

Thus:

$$< \phi(X_i) - \phi(X_j) > = -(\| \phi(X_i) - \phi(X_j) \|_2^2 - \quad (11)$$
$$< \phi(X_i), \phi(X_i) > - < \phi(X_j), \phi(X_j) >)/2 =$$
$$-(d_E^2(\phi(X_i), \phi(X_j)) - < \phi(X_i), \phi(X_i) >$$
$$- < \phi(X_j), \phi(X_j) >)/2$$

Since ϕ function, and consequently, the coordinate of points in the feature space are unknown, computing the inner product between any two points in the projected space is done implicitly using double centering [8], [49], [51] on D_E. The double centering is performed by subtracting the means of the elements of each row and column, and adding the mean of all of the entries of D_E to the corresponding element of D_E [8]. $\phi(X_i)$ is assumed to be centered. This assumption has no effect on the distances:

$$d_E^2(\phi(X_i), \phi(X_j)) = \| \phi(X_i) - \phi(X_j) \|_2^2 \quad (12)$$
$$= \| (\phi(X_i) - c) - (\phi(X_j) - c) \|_2^2$$

where, c is a constant translation vector. Thus we have:

$$\mu_r(i)=\sum_{j=1}^{N} d_E^2(\phi(X_i),\phi(X_j))/N = \sum_{j=1}^{N}\|\phi(X_i)-\phi(X_j)\|_2^2/N = \sum_{j=1}^{N}<\phi(X_i)-\phi(X_j),\phi(X_i)-\phi(X_j)>/N = \;<\phi(X_i),\phi(X_i)>-2<\phi(X_i),\sum_{j=1}^{N}\phi(X_j)/N> + \sum_{j=1}^{N}<\phi(X_j),\phi(X_j)>/N = \;<\phi(X_i),\phi(X_i)>+\sum_{j=1}^{N}<\phi(X_j),\phi(X_j)>/N \quad (13)$$

where, N denotes the number of data points, and $\mu_r(i)$ denotes the mean of the i th row of D_E. Since D_E is a symmetric matrix, the mean of the j th column, $\mu_c(j)$, can be computed as:

$$\mu_c(j)=<\phi(X_j),\phi(X_j)> + \sum_{i=1}^{N}<\phi(X_i),\phi(X_i)>/N \quad (14)$$

and the mean of all of the entries of D_E, μ is:

$$\mu=\sum_{i=1}^{N}\sum_{j=1}^{N} d_E^2(\phi(X_i),\phi(X_j))/N^2 =\sum_{i=1}^{N}\sum_{j=1}^{N}\|\phi(X_i)-\phi(X_j)\|_2^2/N^2 =\sum_{i=1}^{N}<\phi(X_i),\phi(X_i)>/N+\sum_{j=1}^{N}<\phi(X_j),\phi(X_j)>/N \quad (15)$$

Thus:

$$\mu_r(i)+\mu_c(j)-\mu=<\phi(X_i),\phi(X_i)> + <\phi(X_j),\phi(X_j)> =\sum_{i=1}^{N}\sum_{j=1}^{N}<\phi(X_i)-\phi(X_j),\phi(X_i)-\phi(X_j)>/N^2 =\sum_{i=1}^{N}<\phi(X_i),\phi(X_i)>/N-2\sum_{j=1}^{N}<\phi(X_i),0>/N +\sum_{j=1}^{N}<\phi(X_j),\phi(X_j)>/N \quad (16)$$

Using (11) and (16), we have:

$$<\phi(X_i),\phi(X_j)>=-1/2(d_E^2(\phi(X_i),\phi(X_j)) -\mu_r(i)-\mu_c(j)+\mu) \quad (17)$$

Since D_E and the average of each row, column, and all the elements of D_E are computable, therefore, an N-by-N Gramian matrix can be defined as:

$$G=[<\phi(X_i),\phi(X_j)>]_{1\le i,j\le N} \quad (18)$$

Gramian matrix, G, which can be computed based on the computable terms, is a similarity measure on feature space, induced from intrinsic dissimilarity in input space, and can be used as a non-parametric kernel in kernel-based methods.

4.2. Generalization to test points

To generalize the proposed non-parametric kernel to unseen data, we need to update the components that are used in computing the kernel in learning process. To improve the computational complexity of generalization to test samples, the mean values of rows, columns, and all the entries of the D_E matrix for the training dataset are saved.

The inner product between a test sample X and the previous training samples is computed by updating the geodesic and Euclidean distance matrices:

$$D_G \leftarrow \begin{bmatrix} [D_G] & d^2{}_G(X_i,X) \\ d^2{}_G(X,X_j) & 0 \end{bmatrix}_{1\le i,j\le N}, \quad (19)$$
$$D_E \leftarrow D_G$$

Thus the mean values of each row (μ_r), column (μ_c), and the mean of all the entries of D_E (μ) are updated as follow:

$$\mu_r(i)\leftarrow N*\mu_r(i)+d_E{}^2(\phi(X_i),\phi(X))/(N+1) \quad (20)$$
$$\mu_c(j)\leftarrow N*\mu_c(j)+d_E{}^2(\phi(X),\phi(X_j))/(N+1)$$
$$\mu\leftarrow N^2*\mu+2\sum_{i=1}^{N} d_E{}^2(\phi(X_i),\phi(X))/(N+1)^2$$

where, $\mu_r(i)$ denotes the mean of the i^{th} row and j^{th} column. The mean values of row and column, which corresponds to the new sample, are computed as:

$$\mu_r(N+1)=\sum_{j=1}^{N} d_E{}^2(\phi(X),\phi(X_j))/(N+1) \quad (21)$$
$$\mu_c(N+1)\leftarrow\mu_r(N+1)$$

and the inner product corresponding to the new sample and the other observations is computed as follows:

$$\forall j=1..N, G(X,X_j)=<\phi(X),\phi(X_j)>= -1/2(d_E^2(\phi(X),\phi(X_j))-\mu_r(N+1)-\mu_c(j)+\mu) \quad (22)$$

In the case of developable manifolds, since manifolds have isometry with Euclidean space, double centering brings inner product in a Euclidean space. Assuming $V=[v_1....v_N]^T$, where $v_i\in R$, $1\le i\le N$, so:

$$V^TGV=[v_1....v_N][<\phi(X_i),\phi(X_j)>]_{1\le i,j\le N}[v_1....v_N]^T =\sum_{i=1}^{N}\sum_{j=1}^{N} v_i<\phi(X_i),\phi(X_j)>v_j =\sum_{i=1}^{N} v_i<\phi(X_i),\sum_{j=1}^{N} v_j\phi(X_j)> =<\sum_{i=1}^{N} v_i\phi(X_i),\sum_{j=1}^{N} v_j\phi(X_j)>=\|\sum_{i=1}^{N} v_i\phi(X_i)\|_2^2\ge 0 \quad (23)$$

As $V^TGV\ge 0$ thus G matrix satisfies the Mercer's condition, and can be used as a kernel for mapping to RKHS. In the case of non-developable manifolds, due to the intrinsic curvature of the manifold, the Gramian matrix does not satisfy the Mercer's condition.

Using the proposed topology preserving kernel that induces similarities from the distance along the manifold, every kernel-based method that is formulated using the inner product of samples can be used for inference (i.e. clustering, classification, …) on the proposed implicit feature space. For example, the kernel support vector machine (SVM) [10], [18], [21], which is a suitable choice for complex datasets due to its robustness, was used in our experiments. Applying other kernels without considering their type and parameters that determine the topology of data points in feature space may bring undesirable overlapping of points, and may produce weak results.

5. Results

We applied the linear discriminant analysis (LDA) [50] and SVM as the discriminative methods using different kernels on several real datasets; the characteristics of datasets and also the experimental results are reported in this section. To clarify the difference between the proposed kernel over SPD manifold and the classical manifold-learning techniques, a comparison between them is made.

5.1. EEG datasets and pre-processing

Two-class EEG datasets are used in this work. The participants of this study were 43 children and adolescents (21 cases of ADHD, 22 patients with BMD) ranged from 10 to 22 years old. The diagnosis is based on the DSM_IV criterion [4], [20]. For each patient, within three minutes, the EEG signals were recorded in eyes-open and eyes-closed resting conditions. These signals were recorded using 22 electrodes according to the 10-20 international recording system. Impedances of electrodes were lower than 10 KΩ through the recording, and the sampling rate was 250 Hz. In the pre-processing phase, the signals were filtered by a Butterworth low-pass filter (order 7) with 40 Hz cut-off frequency to remove the additive high-frequency noises [20].

The feature vectors were generated by estimating the empirical covariance matrix between channels [9].

In the cases that covariance matrices had eigenvalues less than or equal to zero, we changed the eigenvalues such that all of them became positive, and scaled them such that the distance between eigenvalues was preserved. For this purpose, we added the absolute value of the minimum of eigenvalues to all the eigenvalues, increased them with a small positive value, and reconstructed the matrix with this new eigenvalues and previous eigenvectors.

$$C_{new} = U diag(\lambda_1 + |\min(\lambda_{\min}(C),0)| + \varepsilon \\ ,\dots,\lambda_{n+} |\min(\lambda_{\min}(C),0)| + \varepsilon) U^T \tag{24}$$

where, $C = U\Lambda U^T$, Λ is a diagonal matrix whose diagonal entries are the eigenvalues of C (denoted as λ_i) and U is the matrix of eigenvectors of C . ε is a small positive value.

With this modification, the distance between different eigenvalues are preserved, and the matrix becomes positive definite.

To remove the dependency between the train and test samples, the leave-one-out cross-validation method was performed. In each round, one patient was dedicated as test set and the others were considered as a validation and train set [20].

Ensemble-based techniques, as a promising approach for improving analysis on EEG datasets, are applied in different applications such as BCI, and mental disorder recognition [39- 41]. These techniques improve the accuracy and stability of the algorithms. Avoiding over-fitting and reducing variance are some other advantages that have been reported for ensemble-based techniques. In experiments on the EEG datasets, different classifiers were aggregated using an ensemble-based technique. These classifiers were trained on different subsets of EEG channels. Since the high dimensionality of the covariance matrix of all channels leads to the problem of curse of dimensionality, we generated multiple views on the EEG datasets. The covariance matrices of multiple subsets of channels, composed of 2 or 3 channels, were estimated separately, and then the learning procedure in each of these views was performed. Finally, the results of different views were combined using the majority voting technique. F7-FZ, F3-F7, FP2-F7, T3-F7, and FZ-CZ-F7 indicate the selected channel name in international 10-20 systems. In this work, the channel selection was performed experimentally. The subsets corresponding to different positions on the scalp were selected randomly and used for training the classifiers. These classifiers were tested on the validation set. Some of the selected subsets that on average led to a higher accuracy on the validation set were selected for our experiments.

5.2. Texture classification

In this experiment, we applied the Brodatz texture dataset [33]. 12 different types of textures were used in the learning process. All textures were gray-scale images that were resized to 512 × 512 pixels. Each image was divided into four equal parts. For each image, two parts that were 256 × 256 pixels were devoted as the training set, and the remaining made the test set. To describe each part of the image, covariance matrices in windows with random height, width, and center were computed.

In these experiments, 10 random subsets were selected for describing each part of the image. Each pixel was described using $[I(x,y),|\partial I/\partial x|,|\partial I/\partial y|,|\partial^2 I/\partial x^2|,|\partial^2 I/\partial y^2|]$. Thus the experimental covariance matrix in each window that was computed by 24 would be a 5 × 5 matrix [7]:

$$C_w = \sum_{i=1}^{N}(F_i - \mu_w)(F_i - \mu_w)^T/(N-1) \tag{25}$$

where, N denotes the number of pixels in each window, F_i is a feature vector that describes the ith pixel of the window w and μ_w shows the mean value in that window.

5.3. Visual object classes

The main goal of this experiment is to recognize the objects from a number of visual object classes in realistic scenes without pre-segmenting the objects. PASCAL VOC 2012 that includes person, animal, vehicle, and indoor categories with twenty object classes are used in this work [42]. For each class, the presence/absence of an example of that class in the test images is determined by a binary classifier. To describe each image, the covariance matrices of pixels, which are described using $[I_R(x,y),I_G(x,y),I_B(x,y),|\partial I_R/\partial y|,|\partial I_G/\partial x|,|\partial I_G/\partial y|,|\partial I_B/\partial x|,|\partial I_B/\partial y|]$, are computed by (25). Descriptors would be a 9 × 9 matrix. Parameters are tuned on the validation set and evaluated in a subset with 1200 instances of the test set.

5.4. Experimental results

In this work, the extracted features from different classes are classified by kNN, SVM, LDA, and kernel LDA and kernel SVM with different kernels. For fine tuning the penalty term of SVM and Lagrange multiplier in KLDA, a wide range of values is assessed. The optimal performance on the validation set determines the suitable values for these terms. In the case that the kernel methods have parameters such as the variance parameter in RBF and GGK kernels, these parameters are tuned by assessing the performance on the validation set.

Accuracy of different classifiers on different subsets of channels on eyes-open and eyes-closed datasets and accuracy of an ensemble of these learners are reported in tables 1 and 2.

Table 1. Accuracy of different classifiers (1-NN, 3-NN, linear SVM, SVM with RBF, TSK, GGPK, and GGK kernels) on different subsets of EEG signals of ADHD and BMD patients at eyes-open resting condition.

Classifiers	Channel subsets					MajorityVote
	F7-FZ	F3-F7	FP2-F7	T3-F7	FZ-CZ-F7	
1-NN	72.09%	60.47%	65.12%	67.44%	62.79%	72.09%
3-NN	55.81%	76.74%	67.44%	74.42%	67.44%	76.74%
Linear SVM	72.09%	86.05%	62.79%	55.81%	81.40%	86.05%
SVM-RBF	79.07%	86.05%	72.09%	76.74%	79.07%	86.05%
SVM-TSK [3]	74.42%	81.40%	69.77%	72.09%	79.07%	81.45%
SVM-GGK[7]	81.40%	86.05%	**81.40**%	88.37%	81.40%	86.05%
SVM-GGPK	**93.02**%	**95.35**%	79.07%	**93.02**%	**86.05**%	**95.35**%
LDA	67.44%	76.74%	62.79%	55.81%	74.42%	83.72%
LDA_TSK	72.09%	62.79%	**74.42%**	65.12%	72.09%	81.40%
LDA_GGK	79.07%	81.40%	69.77%	76.74%	79.07%	81.40%
LDA-GGPK	**81.40**%	**81.40**%	67.44%	**83.72**%	**81.40**%	**86.05**%

Table 2. Accuracy of different classifiers (1-NN, 3-NN, linear SVM, SVM with RBF, TSK, GGPK, and GGK kernels) on different subsets of EEG signals of ADHD and BMD patients at eyes-closed resting condition.

Classifiers	Channel subsets					Majority Vote
	F7-FZ	F3-F7	FP2-F7	T3-F7	FZ-CZ-F7	
1-NN	67.44%	58.14%	79.07%	72.09%	79.07%	79.07%
3-NN	67.44%	67.44%	76.74%	67.44%	74.42%	74.42%
Linear SVM	62.79%	69.77%	65.12%	67.44%	72.09%	67.44%
SVM-RBF	72.09%	72.09%	79.07%	76.74%	69.77%	72.09%
SVM-TSK [3]	69.77%	69.77%	76.74%	65.12%	74.42%	74.42%
SVM-GGK[7]	79.07%	69.77%	88.37%	**83.72**%	81.40%	83.72%
SVM-GGPK	**86.05**%	**76.74%**	**88.37%**	79.07%	**86.05**%	**88.37%**
LDA	46.51%	69.77%	65.12%	65.12%	72.09%	72.09%
LDA_TSK	69.77%	60.47%	72.09%	69.77%	81.40%	74.42%
LDA_GGK	81.40%	69.77%	83.72%	76.74%	81.40%	81.40%
LDA-GGPK	72.09%	72.09%	81.40%	76.74%	83.72%	86.05%

Tables 3 and 7 contain accuracy of classification on Brodatz texture and PASCAL VOC2012 dataset, respectively. Comparison between the proposed and some other topology preserving kernels on Riemannian manifolds are reported in these tables.
The TSK kernel, which partially preserves the topology [3] and Gaussian kernel using geodesic distance (GGK) [7], are geometric kernels that are used for comparison with GGPK.
The effectiveness of linearization and preserving the global topology of the dataset by GGPK is compared with RBF and Linear SVM that does not consider the intrinsic geometry of the dataset.

The manifold learning methods such as LLE, HLLE, LE, Isomap, NPAF, and LTSA are used as a feature extractor on covariance matrices. Intrinsic dimensionality of the target is determined by maximum likelihood intrinsic dimensionality estimator (MLE) [37]. SVM with RBF kernel is used for classification. Comparison between the proposed approach and the results evolved on a reduced dataset by the manifold learning techniques are mentioned in tables 4, 5, 6, and 8. These experiments run over random subsets of Brodatz texture dataset, subsets of EEG dataset, and VOC 20012 dataset.

Table 3. Accuracy of linear SVM, SVM with RBF, TSK [3], and GGPK kernels on 12 different types of textures of Brodatz texture dataset.

Classifiers	Accuracy
Linear SVM	74.58%
SVM-RBF	80.83%
SVM-TSK	86.67%
SVM-GGPK	90.00%

Table 4. Accuracy of SVM with RBF kernel trained on features extracted using LLE, HLLE, LE, LTSA, Isomap, and NPAF from different textures from Brodatz dataset.

Classifiers	Texture No.					
	1-2	11-12	5-6	1-2-3	1-2-3-4-5-6	1-2-3-4-5-6-7-8-9-10-11-12
SVM-GGPK	98.33 %	99.17%	100.0%	92.22%	90.83%	90.00%
LLE+SVM-RBF	75.83 %	77.50%	61.67%	73.33%	30.55%	29.17%
HLLE+SVM-RBF	59.17 %	50.83%	51.67%	55.57%	34.44%	27.22%
LE+SVM-RBF	80.00 %	84.17%	65.83%	80.83%	44.72%	35.41%
LTSA+SVM-RBF	50.00 %	55.83%	61.67%	54.81%	34.72%	27.08%
Isomap + SVM-RBF	75.00 %	51.67%	55.00%	70.56%	37.22%	18.47%
NPAF + SVM-RBF	87.50 %	61.67%	70.83%	75.56%	41.94%	33.33%

Table 5. Accuracy of SVM with RBF kernel trained on features extracted using LLE, HLLE, LE, LTSA, Isomap, and NPAF on different subsets of EEG signal of ADHD and BMD patients at eye-open resting condition.

Learning Techniques	Channel Subsets		
	All channels	Fp1, Fp2, Fpz, F3, F4, F7, F8, FZ, C3, C4, CZ, T3	T4, T5, T6, P3, P4, PZ, O1, O2
SVM-GGPK	**83.72%**	**74.42%**	**83.72%**
LLE+SVM-RBF	35.00%	55.00%	58.14%
HLLE+SVM-RBF	25.58%	46.51%	46.51%
LE+SVM-RBF	69. 77%	67.44%	79.07%
LTSA+SVM-RBF	72.09%	30.23%	72.42%
Isomap + SVM-RBF	67.44%	44.19%	62.79 %
NPAF + SVM-RBF	74.42%	62.79%	48.84%

Table 6. Accuracy of SVM with RBF kernel trained on features extracted using LLE, HLLE, LE, LTSA, Isomap, and NPAF on different subsets of EEG signal of ADHD and BMD patients at eye-closed resting condition.

Learning Techniques	Channel subsets		
	All channels	Fp1, Fp2, Fpz, F3, F4, F7, F8, FZ, C3, C4, CZ, T3	T4, T5, T6, P3, P4, PZ, O1, O2
SVM-GGPK	**83.72%**	**83.72%**	**81.40%**
LLE+SVM-RBF	48.84%	44.19%	51.16%
HLLE+SVM-RBF	46.51%	46.51%	41.86%
LE+SVM-RBF	48.84%	67.44%	48.84%
LTSA+SVM-RBF	39.53%	37.21%	44.19%
Isomap + SVM-RBF	32.56%	58.14%	30.23%
NPAF + SVM-RBF	46.51%	62.79%	51.16%

6. Discussion

In our experiments, several real-world datasets and classifiers were used to evaluate several kernel functions and manifold learning techniques. From these experiments, the following results were achieved:

The superiority of SVM-GGPK and LDA-GGPK over Linear SVM and LDA (Tables 1, 2, 3, and 7) shows the effectiveness of the proposed approach, and implies that measuring dissimilarities using the Euclidean distance in non-linear feature space does not reflect dissimilarities truly. The superiority of SVM-GGPK and LDA-GGPK over kNN (Tables 1, 2) and SVM-RBF (Tables 1, 2, 3, 7), which use Euclidean distance for measuring dissimilarities, approves this finding. The geometry-based kernels such as TSK, GGK, and GGPK gain higher discrimination rates in comparison with the RBF and linear kernels. This means that considering the geometry of data points in input space can be effective at learning kernel and outperforms generalization of the classifiers.

The proposed kernel has no parameter, which is one of its superiorities over the RBF and GGK kernels whose performances strongly depend on the bandwidth of the kernel.

Table 7. Accuracy of SVM with linear, RBF, GGK, and GGPK kernels trained on PASCALVOC2012 dataset.

Learning Techniques	Class name			
	Aeroplane	Bird	Bottle	Car
SVM-Linear	80.75%	40.00%	14.33%	61.17%
SVM-RBF	78.25%	91.25%	19.75%	60.42%
SVM-GGK	84.76%	94.60%	21.33%	**67.33%**
SVM-GGPK	**87.75%**	**94.83%**	**24.00%**	66.83%

Table 8. Accuracy of SVM with RBF kernel trained on features extracted using LLE, HLLE, LE, LTSA, Isomap, and NPAF on subsets of PASCALVOC2012 dataset.

Learning Techniques	Class name			
	Aeroplane	Bird	Bottle	Car
SVM-GGPK	**87.75%**	**94.83%**	24.00%	**66.83%**
LLE+SVM-RBF	50.67%	49.83%	49.50%	50.67%
HLLE+SVM-RBF	48.75%	53.08%	49.75%	50.08%
LE+SVM-RBF	50.67%	50.33%	48.75%	49.67%
LTSA+SVM-RBF	54.50%	51.83%	48.00%	44.42%
Isomap + SVM-RBF	64.08%	49.67%	37.83%	50.25%
NPAF + SVM-RBF	70.67%	52.92%	**55.00%**	43.67%

Table 9. p-value resulted by applying paired t-Test for comparison between SVM-GGPK and other compatitors on ADHD/BMD dataset in classification problem.

	SVM-GGPK/ SVM-GGK	SVM-GGPK/ SVM-TSK	SVM-GGPK/ SVM-RBF	SVM-GGPK/ Linear SVM	SVM-GGPK/ 3-NN	SVM-GGPK/ 1-NN
Eyes-open	0.0293	0.0013	0.0011	0.0195	0.0021	3.6875e-04
Eyes-closed	0.1576	2.1248e-04	0.0080	0.0019	1.9118e-04	0.0035

Table 10. p-value resulted by applying paired t-Test for comparison between SVM-GGPK and other compatitors on Brodatz texture dataset in dimensionality reduction problem.

	SVM-GGPK/ LLE+SVM-RBF	SVM-GGPK/ HLLE+SVM-RBF	SVM-GGPK/ LE+SVM-RBF	SVM-GGPK/ LTSA+SVM-RBF	SVM-GGPK/ Isomap+SVM-RBF	SVM-GGPK/ NPAF+SVM-RBF
Brodatz	0.0054	7.1445e-05	0.0092	8.6519e-05	0.0024	0.0061

Experiments show the superiority of the proposed approach over the techniques that rely on manifold learning. Conventional manifold learning techniques are applicable only on the cases in which a manifold is embedded in the Euclidean space. In this work, our input space is composed of symmetric positive definite matrices. Since the features can be formulated as a Riemannian manifold and live in a non-Euclidean space, applying the classical manifold learning methods on this manifold is not compatible with the pre-requisites of the conventional manifold learning techniques. Weak generalization of manifold learning-based methods, which are reported in tables 4, 5, 6, and 8, confirm this fact. Therefore, to apply the manifold learning methods over Riemannian manifolds, it is required to modify some parts of these methods that depend on the manifold structure [36]. Some reasons that lead to inconvenience of the manifold learning techniques that are examined in this study over Riemannian manifold are listed what follows.

LE tries to preserve locality in projection to the low-dimensional space and uses the Laplacian matrix for representing manifold. The shortcoming of LE on Riemannian manifolds is the result of approximating true geodesic distance by graph distance.

LLE computes a weight matrix such that a data point can be constructed as a linear combination of its neighbors, and its aim is to preserve local linearity in a low-dimensional space. In the Euclidean case, this aim is achieved by solving a least-squares problem, while in the Riemannian case, it is required to solve an interpolation problem on the manifold. The cost function that should be minimized and the interpolation on the Riemannian manifold are some challenges that make LLE on Riemannian manifold different from the classical one.

A learning process in HLLE consists of computing the mean and a set of principal components from the neighborhood of each point. In the Euclidean case, this can be done using PCA, while on the Riemannian manifolds, computing mean can be done in an iterative procedure, and computing principal components on the manifold has some challenges. For example, the principal geodesic analysis [38] was proposed to compute the principal components on Riemannian manifolds.

In the case of LTSA, in the first stage, a local parameterization of data points should be provided. This stage is computed by the assumption that the data points are embedded in the Euclidean space, and the Taylor series expansion in the Euclidian space around the base point of tangent space lead to finding local coordinates at the corresponding tangent space that is computed using PCA. Since LTSA estimates the tangent space of the Riemannian manifold at a point using available data samples in the neighborhood of the base point, sampling conditions such as the sampling extent and density affect the estimated tangent space. Running PCA on some instances of the Riemannian manifold leads to inaccurate local information, which brings poor results in classification.

Isomap tries to preserve the global geometry in projection to the low-dimensional space and use the geodesic distance for capturing the intrinsic geometry of the manifold. Isomap represents the manifold using a graph on the available data points and approximates the geodesic distance using graph distance. The density of input data and bad sampling may lead to disconnectivity of graph and partial covering over training data. Over-estimation of geodesic distance and linear shortcuts near regions of high surface curvature are two disadvantages of Isomap that are the result of the estimation of geodesic distance by graph distance. These shortcomings can lead to overlapping of data points, and may decrease generalization of learners over SPD manifold.

Manifold learning techniques, which are not compatible with SPD Riemannian manifold, may corrupt the topology of data points. In multi-class cases, by increasing the number of classes, mapping to low-dimensional space cause more overlapping between different classes, and lead to weakness of classifiers.

To show the statistical significance of superiority of the proposed approach, we apply the statistical test on the ADHD/BMD dataset in two eyes-closed and eyes-open resting condition in classification problem (Table 9) and on Brodatz texture dataset in dimensionality reduction problem (Table 10). The resulting p-values in most cases indicate the significant superiority of the methods that relied on using GGPK kernel in both the classification and dimensionality reduction problems.

7. Conclusion

In this paper, we proposed a global projection technique for mapping points lying on the SPD Riemannian manifold to feature space such that the topology of input space is preserved. Learning kernel over SPD manifold by computing the Gramian matrix, based on squared geodesic distance, was our contribution.

Superiority over approaches that partially preserve topology such as approaches that are relied on projection to tangent space or approaches that do not preserve topology such as some Euclidean distance-based kernels shows effectiveness of the preserving topology.
In comparison with methods that are based upon the traditional manifold learning techniques, superiorities are observed in the experiments. The shortcoming of manifold learning methods over SPD manifold can be the result of living SPD manifold in non-Euclidean space, while these methods do computation with the assumption that data points live in the Euclidean space.

References

[1] Pennec, X. (2006). Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements. Journal of Mathematical Imaging and Vision, vol. 25, no. 1, pp. 127-154.

[2] Wang, J., Lu, H., Plataniotis, K. N., & Lu, J. (2009). Gaussian kernel optimization for pattern classification. Pattern Recognition, vol. 42, no. 7, pp. 1237-1247.

[3] Barachant, A., Bonnet, S., Congedo, M., & Jutten, C. (2012). BCI Signal Classification using a Riemannian-based kernel. In 20th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), pp. 97-102.

[4] Sadatnezhad, K., Boostani, R., & Ghanizadeh, A. (2011). Classification of BMD and ADHD patients using their EEG signals. Expert Systems with Applications, vol. 38, no. 3, pp. 1956-1963.

[5] Tuzel, O., Porikli, F., & Meer, P. (2008). Pedestrian detection via classification on riemannian manifolds. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no.10, pp. 1713-1727.

[6] Subbarao, R., & Meer, P. (2009). Nonlinear mean shift over Riemannian manifolds. International Journal of Computer Vision, vol. 84, no. 1, pp. 1-20.

[7] Harandi, M. T., Sanderson, C., Wiliem, A., & Lovell, B. C. (2012). Kernel analysis over Riemannian manifolds for visual recognition of actions, pedestrians and textures. IEEE Workshop in Applications of Computer Vision (WACV), pp. 433-439, 2012.

[8] Lee, J. A., & Verleysen, M. (2007). Nonlinear dimensionality reduction. Springer Science & Business Media.

[9] Barachant, A., Bonnet, S., Congedo, M., & Jutten, C. (2012). Multiclass brain–computer interface classification by Riemannian geometry. IEEE Transactions on Biomedical Engineering, vol. 59, no. 4, pp. 920-928.

[10] Friedman, J., Hastie, T., & Tibshirani, R. (2001). The elements of statistical learning. Springer, Berlin: Springer series in statistics.

[11] Lee, J. M. (2006). Riemannian manifolds: an introduction to curvature. Springer Science & Business Media.

[12] Jost, J. (2008). Riemannian geometry and geometric analysis. Springer Science & Business Media.

[13] O'neill, B. (1983). Semi-Riemannian Geometry with Applications to Relativity. Academic press.

[14] Rossmann, W. (2002). Lie groups: an introduction through linear groups. Oxford University Press.

[15] Pennec, X., Fillard, P. & Ayache, N. (2006). A Riemannian framework for tensor computing. International Journal of Computer Vision, vol. 66, no. 1, pp. 41-66.

[16] Kim, S. J., Magnani, A., & Boyd, S. (2006). Optimal kernel selection in kernel fisher discriminant analysis. In Proceedings of the 23rd international conference on Machine learning ACM, pp. 465-472.

[17] Lin, T., & Zha, H. (2008). Riemannian manifold learning. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 5, pp. 796-809.

[18] Bottou, L., & Lin, C. J. (2007). Support vector machine solvers. Large scale kernel machines, pp. 301-320.

[19] Gallier, J. (2011). Geometric methods and applications: for computer science and engineering. Springer Science & Business Media.

[20] Sadatnezhad, K., Boostani, R., & Ghanizadeh, A. (2010). Proposing an adaptive mutation to improve XCSF performance to classify ADHD and BMD patients. Journal of neural engineering, vol. 7, no. 6, pp. 066006.

[21] Cristianini, N., & Shawe-Taylor, J. (2000). An introduction to support vector machines and other kernel-based learning methods. Cambridge university press.

[22] Dey, T. K., & Li, K. (2009). Cut locus and topology from surface point data. In Proceedings of the twenty-fifth annual symposium on Computational geometry ACM, pp. 125-134.

[23] Porikli, F., Tuzel, O., & Meer, P. (2006). Covariance tracking using model update based on lie algebra. In Computer Vision and Pattern Recognition, IEEE Computer Society Conference, vol. 1, pp. 728-735.

[24] Förstner, W., & Moonen, B. (2003). A metric for covariance matrices. Springer Berlin Heidelberg. In Geodesy-The Challenge of the 3rd Millennium.

[25] Tuzel, O., Porikli, F., & Meer, P. (2006). Region covariance: A fast descriptor for detection and

classification. Springer Berlin Heidelberg. In Computer Vision–ECCV, pp. 589-600.

[26] Tuzel, O., Subbarao, R., & Meer, P. (2005). Simultaneous multiple 3D motion estimation via mode finding on Lie groups. Tenth IEEE International Conference in Computer Vision, vol. 1, pp. 18-25.

[27] Tuzel, O., Porikli, F., & Meer, P. (2007). Human detection via classification on riemannian manifolds. In Computer Vision and Pattern Recognition, IEEE Conference on pp. 1-8.

[28] Guo, K., Ishwar, P., & Konrad, J. (2010). Action recognition using sparse representation on covariance manifolds of optical flow. In Advanced Video and Signal Based Surveillance (AVSS), Seventh IEEE International Conference, pp. 188-195.

[29] Tosato, D., Farenzena, M., Spera, M., Murino, V., & Cristani, M. (2010). Multi-class classification on riemannian manifolds for video surveillance. Springer Berlin Heidelberg. In Computer Vision–ECCV, pp. 378-391.

[30] Li, X., Hu, W., Zhang, Z., Zhang, X., Zhu, M., & Cheng, J. (2008). Visual tracking via incremental log-euclidean riemannian subspace learning. In Computer Vision and Pattern Recognition. IEEE Conference, pp. 1-8.

[31] Sanin, A., Sanderson, C., Harandi, M. T., & Lovell, B. C. (2012). K-tangent spaces on Riemannian manifolds for improved pedestrian detection. 19th IEEE International Conference on Image Processing (ICIP), pp. 473-476.

[32] Courty, N., Burger, T., & Marteau, P. F. (2012). Geodesic analysis on the Gaussian RKHS hypersphere. Springer Berlin Heidelberg. In Machine Learning and Knowledge Discovery in Databases, pp. 299-313.

[33] Randen, T., & Husoy, J. H. (1999). Filtering for texture classification: A comparative study. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 21, no. 4, pp. 291-310.

[34] Xiong, H., Swamy, M. N. S., & Ahmad, M. O. (2005). Optimizing the kernel in the empirical feature space. IEEE Transactions on Neural Networks, vol. 16, no.2, pp. 460-474.

[35] Jayasumana, S., Hartley, R., Salzmann, M., Li, H., & Harandi, M. (2013). Kernel methods on the riemannian manifold of symmetric positive definite matrices. IEEE Conference in Computer Vision and Pattern Recognition (CVPR), pp. 73-80.

[36] Goh, A., & Vidal, R. (2008). Clustering and dimensionality reduction on Riemannian manifolds. IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-7.

[37] Lim, I. S., de Heras Ciechomski, P., Sarni, S., & Thalmann, D. (2003). Planar arrangement of high-dimensional biomedical data sets by isomap coordinates. In Computer-Based Medical Systems, 16th IEEE Symposium, pp. 50-55.

[38] Fletcher, P. T., & Joshi, S. (2007). Riemannian geometry for the statistical analysis of diffusion tensor data. Signal Processing, vol. 87, no. 2, pp. 250-262.

[39] Polikar, R. (2006). Ensemble based systems in decision making. Circuits and Systems Magazine, vol. 6, no. 3, pp. 21-45.

[40] Sun, S., Zhang, C., & Zhang, D. (2007). An experimental evaluation of ensemble methods for EEG signal classification. Pattern Recognition Letters, vol. 28, no. 15, pp. 2157-2163.

[41] Polikar, R., Topalis, A., Parikh, D., Green, D., Frymiare, J., Kounios, J., & Clark, C. M. (2008). An ensemble based data fusion approach for early diagnosis of Alzheimer’s disease. Information Fusion, vol. 9, no. 1, pp. 83-95.

[42] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2012). The PASCAL visual object classes challenge results.

[43] Donoho, D. L., & Grimes, C. (2003). Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. Proceedings of the National Academy of Sciences, vol. 100, no. 10, pp. 5591-5596.

[44] Zhang, Z. Y., & Zha, H. Y. (2004). Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. Journal of Shanghai University (English Edition), vol. 8, no. 4, pp. 406-424.

[45] Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. Science, vol. 290, no. 5500, pp. 2323-2326.

[46] Belkin, M., & Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. Neural computation, vol. 15, no.6, pp. 1373-1396.

[47] Guan, N., Tao, D., Luo, Z., & Yuan, B. (2011). Non-negative patch alignment framework. IEEE Transactions on Neural Networks, vol. 22, no. 8, pp. 1218-1230.

[48] Van der Maaten, L. (2013), Affiliation: Delft University of Technology. Matlab Toolbox for Dimensionality Reduction (v0.8.1b).

[49] Yan, S., Xu, D., Zhang, B., Zhang, H. J., Yang, Q., & Lin, S. (2007). Graph embedding and extensions: a general framework for dimensionality reduction. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 1, pp. 40-51.

[50] Scholkopft, B., & Mullert, K. R. (1999). Fisher discriminant analysis with kernels. Neural networks for signal processing IX.

[51] Kung, S. Y. (2014). Kernel methods and machine learning. Cambridge University Press.

[52] Vemulapalli, R., Pillai, J. K., & Chellappa, R. (2013). Kernel learning for extrinsic classification of

manifold features. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1782-1789.

[53] Wang, R., Guo, H., Davis, L. S., & Dai, Q. (2012). Covariance discriminative learning: A natural and efficient approach to image set classification. In Computer Vision and Pattern Recognition, pp. 2496-2503.

[54] Huang, Z., Wang, R., Shan, S., Li, X., & Chen, X. (2015). Log-Euclidean Metric Learning on Symmetric Positive Definite Manifold with Application to Image Set Classification, pp. 720-729.

[55] Zare, T., Sadeghi, M. T., Abutalebi, H. R., & Kittler, J. (2017). Composite Kernel Optimization in Semi-Supervised Metric. Journal of AI and Data Mining, vol. 5, no. 2, pp. 259-273.

نشریه هوش مصنوعی و داده کاوی

معرفی یک هسته مبتنی بر هندسه ددگان روی منیفلد های ریمانی

سیده خدیجه سادات نژاد، سعید شیری قیداری* و محمد رحمتی

دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، تهران، ایران.

ارسال ۲۰۱۵/۱۱/۰۲؛ بازنگری ۲۰۱۷/۰۳/۰۱؛ پذیرش ۲۰۱۷/۰۴/۰۸

چکیده:

حقه هسته و نگاشت به صفحات مماسی دو گزینه به منظور خطی‌سازی داده‌های واقع بر منیفلد ریمانی می‌باشند. خطی‌سازی منیفلد بـه منظـور فـراهم نمودن پیش شرط‌های لازم جهت اعمال تکنیک‌های استاندارد یادگیری ماشین روی منیفلدهای ریمانی ضروری می‌باشد. هسته‌های کلاسـیک صـرفنظر از توپولوژی داده‌ها در فضای ورودی به طور ضمنی داده‌ها را به فضای ویژگی نگاشت می‌نمایند. خطی‌سازی با استفاده از نگاشـت بـه فضـاهای مماسـی، هندسه داده‌ها را تنها در امتداد ژئودزیک‌های شعاعی حفظ می‌نماید. در این مقاله یک روش یادگیری بر منیفلدهای ریمـانی مبتنـی بـر رویکـرد هسـته پیشنهاد گردید. محاسبه ماتریس گرامیان با استفاده از فواصل ژئودزیک روی یک منیفلد ریمانی کامل، با ژئودزیـک کمینـه یکتـا بـین هـر زوج از نقـاط روی منیفلد، یک نگاشت ویژگی فراهم می‌نماید که متاثر از توپولوژی داده‌ها می‌باشد. رویکـرد پیشـنهادی روی مجموعـه داده‌هـای حقیقـی متشـکل از سیگنالهای بیماران با دو اختلال روانی متفاوت، کلاسهای اشیاء بصری و بافت مورد ارزیابی قرارگرفت. برای ارزیابی کارایی رویکـرد پیشـنهادی، ویژگـی-های استخراج شده با روش‌های برتر در این حوزه روی منیفلد ماتریس‌های متقارن و مثبت معین مورد ارزیابی قرار گرفت. نتایج تجربی بـه دسـت آمـده برتری دقت روش پیشنهادی در قیاس با روش‌هایی که به توپولوژی مجموعه‌داده توجه نمی‌نمایند یا به طور جزئی توجه می‌نمایند را نشان می‌دهد.

کلمات کلیدی: حقه هسته، منیفلد ریمانی، حفظ هندسه، ماتریس گرامیان.