Covariance Descriptors on a Gaussian Manifold and their Application to Image Set Classification

Kai-Xuan Chen[a,c], Jie-Yi Ren[a,c], Xiao-Jun Wu[a,c,*], Josef Kittler[b]

[a] *School of IoT Engineering, Jiangnan University, 214122, Wuxi, China*
[b] *Center for Vision, Speech and Signal Processing(CVSSP), University of Surrey, GU2 7XH, Guildford, UK*
[c] *Jiangsu Provincial Engineering Laboratory of Pattern Recognition and Computational Intelligence, Jiangnan University, 214122, Wuxi, China.*

Abstract

Covariance descriptors (CovDs) for image set classification have been widely studied recently. Different from the conventional CovDs, which describe similarities between pixels at different locations, we focus more on similarities between regions that convey more comprehensive information. In this paper, we extract pixel-wise features of image regions and represent them by Gaussian models. We extend the conventional covariance computation onto a special type of Riemannian manifold, namely a Gaussian manifold, so that it is applicable to our image set data representation provided in terms of Gaussian models. We present two methods to calculate a Riemannian local difference vector on the Gaussian manifold (RieLDV-G) and generate our proposed Riemannian covariance descriptors (RieCovDs) using the resulting RieLDV-G. By measuring the recognition accuracy achieved on benchmarking datasets, we demonstrate experimentally the superior performance of our proposed RieCovDs descriptors, as compared with state-of-the-art methods. (*The code is available at:* `https://github.com/Kai-Xuan/RiemannianCovDs`)

Keywords: covariance descriptors, Riemannian local difference vector, Riemannian covariance descriptors, image set classification.

*Corresponding author
Email address: `wu_xiaojun@jiangnan.edu.cn` (Xiao-Jun Wu)

1. Introduction

The bag-of-features (BoF) model [1] has been a commonly used representation for the task of image classification over a decade. However, it is well-known that a codebook obtained by pre-training on one dataset cannot, in general, be used for other datasets [2], and learning a codebook is time consuming, especially for datasets of large size. Different from the BoF-based descriptors, a codebook-free model, called Codebookless Model (CLM) [3], estimates the data model on a set of local features, and generates a data representation directly. The well-known structured descriptors, such as linear subspaces [4, 5], covariance descriptors (CovDs)[6, 7, 8] and single Gaussian model [9, 3], belong to the CLM group of methods. They have been shown to offer powerful representations for high dimensional tasks in computer vision. The subspaces spanned by these structured representations are Grassmannian manifold [4], Symmetric Positive Definite (SPD) manifold [6] and Gaussian manifold [3] respectively.

The covariance descriptors, which estimate the data statistics on sets of local features, have been widely applied to represent visual data for the task of classification. Prior to [6], it was common to characterise local regions within an image by covariance matrices, namely as region CovDs. Different from CovDs for describing image sets, region CovDs extract features that include intensity, position, and derivatives. A feature vector computed at a single pixel in a region is viewed as a sample. Region covariance matrices will often be nonsingular because the feature dimensionality (the number of feature attributes) is usually lower than the number of samples (the number of pixels in the region). In contrast, when CovDs are used to describe image sets [6], samples are the images in the set and features are the raw intensities of the image pixels. The resulting covariance matrices are often singular because the feature dimensionality (the number of the pixels in each image) is usually greater than the number of samples (the number of images). In addition, the conventional CovDs for describing image sets are deficient as they lack the capacity to capture all the available discriminative information. Motivated by this drawback, we present a Riemannian version of the conventional covariance descriptors (RieCovDs) to offer a more discriminative

representation for image sets.

Inspired by the work in [10], which proposed to compute a Riemannian local difference vector (RieLDV) by considering geodesic distance and the gradient of the geodesic distance function to advocate a Riemannian version of the Vector of Locally Aggregated Descriptors (RieVLAD) for visual classification tasks, inhere we propose a novel framework for computing the covariance on a Gaussian manifold [3] and generate RieCovDs to tackle the task of image set classification [6, 11, 12]. The challenge of the covariance computation on the Gaussian manifold [3] lies in the fact that the space of Gaussians is not a linear space, where the deviations from the expected value could simply be computed through the subtraction operation. We first consider RieLDV[10] as a measure of deviations from the expectation of Gaussian models and then present two methods for computing RieLDV on the Gaussian manifold (RieLDV-G). Finally, we define the covariance between two sets of Gaussian models as a special case of their RieLDV-G's inner product.

The contributions of our proposed RieCovDs are three-fold:

1. *We propose to characterize the similarities between image sets by focusing on image regions, which contain more comprehensive information, rather than pixels.*
2. *Two methods are developed for computing a Riemannian local difference vector on the Gaussian manifold.*
3. *A novel framework is provided for computing covariance on the Gaussian manifold and generating the proposed Riemannian covariance descriptors (RieCovDs).*

The rest of this paper is organized as follows. Section 2 presents a review of the related work. In Section 3, we introduce our proposed framework for generating a Riemannian version of covariance descriptors on the Gaussian manifold. Section 4 presents two methods for calculating RieLDV-G. The experimental results can be found in Section 5. We draw our conclusions and present future work in Section 6.

2. Related Work

An alternative, codebook-free modeling approacfh [3], which aims to represent visual information directly, has been attracting increasing attention in recent years. Rubner et al.[13] proposed the Earth Mover's Distance for the task of image retrieval and introduced the concept of image signature. Hamm et al.[4, 5] suggested desribing image data as a collection of linear subspaces instead of vectors and proposed the subspace-based learning approach by assuming the data lies on the Grassmann manifold. Wang et al.[3] represented images with single Gaussian models that were matched using Riemannian metric. Nakayama et al.[14] also used global Gaussians to describe images and matched two representations by using the Kullback-Leibler divergence. Tuzel et al.[7, 8] used covariance matrices to represent regions of an image, and employed Affine Invariant Riemannian Metric [15] for the tasks of object detection and classification. Wang et al.[6] proposed computing the covariance of image pixels intensities at different locations and used the resulting covariance matrices to describe image sets.

Different approaches to visual data representation via covariance computation have been investigated in the context computer vision tasks, including e.g., region covariance matrices for pedestrian detection [8] and material categorization [16], covariance matrices of image pixel intensities for image set or video based classification [6], covariance matrices of the local Brownian motion of water molecules in the diffusion tensor imaging (DTI) [15], and human joint covariances for activity recognition [17]. Quang et al.[18] proposed a novel computational framework, namely Log-Hilbert Schmidt metric, for analyzing infinite-dimensional CovDs, which enhances the discriminative power of low-dimensional CovDs. Harandi et al.[19] mapped CovDs to Reproducing Kernel Hilbert Space and advocated computing and comparing the resulting infinite-dimensional CovDs by using Bregman divergences. However, the last two methods suffer from high computational costs and from the drawbacks of implicit representation. These problems were partly addressed in [16] and [20].

There is another line of research on generating more discriminative data represen-

taion directly. Wang et al.[21] proposed an open framework based on the use of a kernel matrix over feature dimensions as a generic representation and discussed its properties and advantages. Li et al.[22] extended the descriptive granularity of covariance matrix from the traditional pixel-level to a more general patch-level. Chen et al.[23] concentrated on modelling the similarities of sub-image sets instead of those of pixels and proposed a framework to generate a low-dimensional discriminative data representation for describing image sets. Different from the above methods, we propose a mathematical framework for the covariance computation on a Gaussian manifold instead of Euclidean space and advocate the use of Riemannian covariance descriptors (RieCovDs) for describing image sets.

3. Riemannian Covariance Descriptors

In this section, we start by reviewing the conventional covariance computation in a Euclidean space. We then extend the formulation to the task of covariance computation on a Gaussian manifold. Finally, we introduce our proposed RieCovDs for describing image sets.

Notation: We shall denote the Gaussian manifold spanned by k-dimensional Gaussian models by G_k . Further, let S_k^{++} denote the SPD manifold spanned by real value $k \times k$ SPD matrices and S_k the space spanned by real value $k \times k$ symmetric matrices.

3.1. Conventional covariance computation

The covariance between two jointly distributed real-valued random variables X and Y with finite second moments is defined as the expected product of their deviations from their respective expected values[24]: cov(X,Y) = E[(X - E(X))(Y - E(Y))], where E[X] is the expected value of X, also known as the mean of X. X = $\{x_1, x_2, ..., x_n\}$ and Y = $\{y_1, y_2, ..., y_n\}$ can also be sets of discrete scalar variables. Then the sample based covariance can be written in terms of the means E(X) and E(Y) as:

$$\mathrm{cov}(X, Y) = \frac{1}{n-1} \sum_{p=1}^{n} (x_p - \mathrm{E}(X))(y_p - \mathrm{E}(Y)) \tag{1}$$

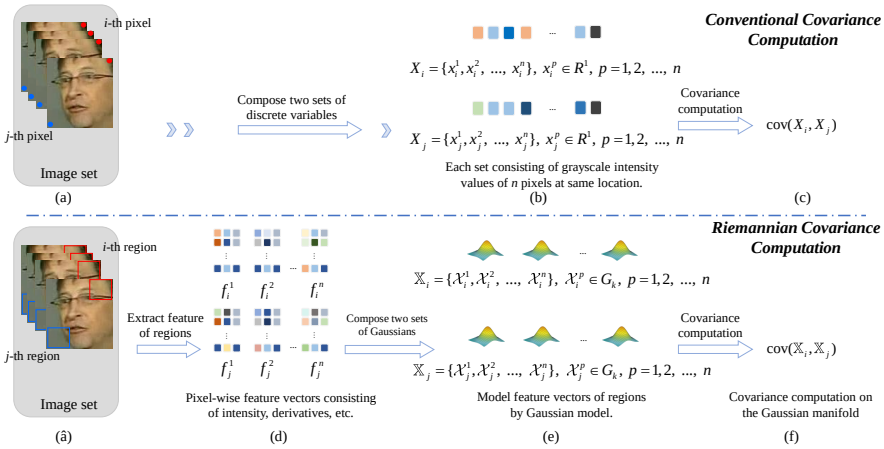

Figure 1: The flow chart of convetional covariance computation and Riemannian covariance computation. Top: conventional covariance computation. Bottom: Riemannian covariance computation. (a) input: image set. (b) two sets of discrete values, each set consisting of n grayscale intensity values of n pixels at the same location. (c) covariance via Eq.(1) between two sets. (â) input: image set. (d) pixel-wise features (position, first order gradients, etc.) of regions. (e) two sets of Gaussian models, each set consisting of n Gaussian models of n regions at the same location. (f) Riemamnnian covariance via Eq.(2) between two sets.

where $\mathrm{E}(X) = \frac{1}{n}\sum_{p=1}^{n} x_p$, $\mathrm{E}(Y) = \frac{1}{n}\sum_{p=1}^{n} y_p$.

3.2. Riemannian covariance computation

Now let us assume that $\mathbb{X}$ = $\{\mathcal{X}_1,\mathcal{X}_2,...,\mathcal{X}_n\}$, $\mathcal{X}_p \in G_k$ and $\mathbb{Y}$ = $\{\mathcal{Y}_1,\mathcal{Y}_2,...,\mathcal{Y}_n\}$, $\mathcal{Y}_p \in G_k$, $p = 1, 2, ..., n$, are two sets of n k-dimensional Gaussian models, which can be considered as the points on a Gaussian manifold [3]. Since the Euclidean operations are not applicable to the points on a Riemannian manifold, the operation of vector subtraction cannot be used for computing deviations from the expectation of Gaussian models. In [10], Faraki et al. defined a Riemannian local difference vector (RieLDV) in terms of geodesic distance and the gradient of geodesic distance functions. We adopt the idea here to define deviations on Gaussian manifold [3]. We propose the covariance computation on Gaussian manifold as:

$$\mathrm{cov}(\mathbb{X}, \mathbb{Y}) = \frac{1}{n-1}\sum_{p=1}^{n} \zeta(\mathcal{X}_p, \mathrm{E}(\mathbb{X}))^T \, \zeta(\mathcal{Y}_p, \mathrm{E}(\mathbb{Y})) \tag{2}$$

where E($\mathbb{X}$) is the expected value of $\mathbb{X}$. The *column vector* $\zeta(\mathcal{X}_p, \mathrm{E}(\mathbb{X}))$ is the RieLDV relative to $\mathrm{E}(\mathbb{X})$. The covariance on a Gaussian manifold is defined as the expected product of the Riemannian local difference vectors defined with respect to the expectation of the Gaussian models. According to [10], $\zeta(\mathcal{X}_p, \mathrm{E}(\mathbb{X}))$ has the following form:

$$\zeta(\mathcal{X}_p, \mathrm{E}(\mathbb{X})) = \delta(\mathcal{X}_p, \mathrm{E}(\mathbb{X})) \frac{\nabla_{\mathrm{E}(\mathbb{X})}\delta^2(\mathcal{X}_p, \mathrm{E}(\mathbb{X}))}{\| \nabla_{\mathrm{E}(\mathbb{X})}\delta^2(\mathcal{X}_p, \mathrm{E}(\mathbb{X})) \|} \tag{3}$$

where δ is the geodesic distance on the curved manifold, and $\nabla_{\mathrm{E}(\mathbb{X})}\delta^2(\mathcal{X}_p, \mathrm{E}(\mathbb{X}))$ [1] is the gradient of the smooth distance function δ at point $E(\mathbb{X})$.

In the next section (Section 4), we introduce a sample representation in terms of a single Gaussian model and develop two methods for calculating the Riemannian local difference vector on the Gaussian manifold (RieLDV-G). Before concluding this section, we discuss how the conventional CovDs and Riemannian CovDs can be used to describe image sets.

[1] It should be noted that when computing the Riemannian gradient, $\nabla_X Y$ does not signify the usual notation in Riemannian geometry, referring to the object Y being covariantly differentiated along the vector field X. Instead, in Eq.(3) the lower index denotes the variables along which one differentiates.

3.3. Riemannian CovDs for describing image set

In the subsection, we follow the steps of Figure 1 to elaborate conventional CovDs and Riemannian CovDs for describing an image set. Given an image set with n image samples, the final representation of an image set is a symmetric matrix obtained via Riemannian covariance computation between regions at different locations instead of pixels.

3.3.1. Conventional CovDs

Conventional CovDs for describing an image set generate representations via the covariance computation between the sets of discrete pixel values. Each set consists of n grayscale pixel intensity values at the same image location. The conventional covariance matrix representation can be written as:

$$\begin{aligned} C &= [C_{i,j}]_{i,j=1,\ldots,d} \\ where \quad C_{i,j} &= \mathrm{cov}(X_i, X_j) \end{aligned} \tag{4}$$

where d is the number of pixels in each image and X_i denotes the set of grayscale intensity values at i-th pixel of n image samples (Figure 1 step(b)). $C_{i,j}$ is the covariance computed between two sets of grayscale values: X_i and X_j (Figure 1 step(c)) defined via Eq.(1). The non-diagonal entry of the resulting covariance matrix C: $C_{i,j}$ is the covariance between X_i and X_j, which represents their respective correlation. The diagonal entries represent the variance of discrete values in each individual set.

3.3.2. Riemannian CovDs

Riemannian CovDs (RieCovDs) describe the correlations between regions instead of pixels and offer a more discriminative representation than the conventional CovDs for the task of image set classification. For the i-th and j-th region of the n image samples (Figure 1 Step(â)), we first extract pixel-wise features (positions, first order gradients, etc.) of the regions (Figure 1 Step(d)) and then use two sets of Gaussian models: $\mathbb{X}_i = \{\mathcal{X}_i^1, \mathcal{X}_i^2, \ldots, \mathcal{X}_i^n\}$ and $\mathbb{X}_j = \{\mathcal{X}_j^1, \mathcal{X}_j^2, \ldots, \mathcal{X}_j^n\}$ (where $\mathcal{X}_i^p \in G_k$ denotes i-th region of p-th image) to represent them (Figure 1 Step(e)). The covariance $\mathrm{cov}(\mathbb{X}_i, \mathbb{X}_j)$ (Figure 1 Step(f)) between $\mathbb{X}_i$ and $\mathbb{X}_j$ is then defined the by Riemannian covariance

Algorithm 1 The proposed RieCovDs algorithm

Input:

- An image set with n images

Output:

- Riemannian CovDs for describing the image set

1: Divide the image into D regions of the same size.

2: For the i-th and j-th region of the image set (Figure 1 step(â)), we first extract pixel-wise features of the regions (Figure 1 Step(d)).

3: Model feature vectors of the regions by a Gaussian model and obtain two sets of Gaussian models: $\mathbb{X}_i$ and $\mathbb{X}_j$ (Figure 1 Step(e)).

4: Calculate the covariance cov($\mathbb{X}_i$,$\mathbb{X}_j$) between $\mathbb{X}_i$ and $\mathbb{X}_j$ (Figure 1 Step(f)).

5: Generate the resulting covariance matrix $\mathbb{C}$ via Eq.(5) .

computation (Eq.(2)). The resulting Riemannian CovDs descriptor of the image set is:

$$\begin{aligned} \mathbb{C} &= [\mathbb{C}_{i,j}]_{i,j=1,\ldots,D} \\ where \quad \mathbb{C}_{i,j} &= \mathrm{cov}(\mathbb{X}_i, \mathbb{X}_j) \end{aligned} \tag{5}$$

In Eq.(5), D is the number of regions in each image and $\mathbb{X}_i$ is the set of Gaussian models at the i-th region of the n images. $\mathbb{C}_{i,j}$ denotes the covariance between the i-th and j-th region. Algorithm 1 is referred to as RieCovDs.

3.4. A comparison with other improved variants of the conventional CovDs for image set coding

The works in [21, 22, 23] are the existing improved versions of the conventional CovDs for describing image sets, producing different representations. It is pertinent to elaborate their connections to and differences from the proposed approach. Wang et al.[21] proposed an open framework based on the use of a kernel matrix defined over feature dimensions as a generic representation. This work used a non-linear kernel matrix as the representation that describes similarities between pixels at different locations

as a preamble to the conventional CovDs [6]. Our work proposes to capture similarities between regions at different locations and uses Gaussian models to represent them. Li et al.[22] extended the descriptive granularity of the covariance matrix from the traditional pixel-level to a more general patch-level. Though this work concentrates on the patch-level covariance computation, it is actually a sum-pooling form of the pixel-level covariances. Different from [22], we extract pixel-level features instead of using intensity values and extend the covariance computation from Euclidean space to a Gaussian manifold. Chen et al.[23] concentrated on similarities between sub-image sets instead of the pixels and proposed a framework, which has been improved in [25], to generate a low-dimensional discriminative data representation for describing image sets. The main difference is that we describe the similarities between the regions at different locations via covariance computation on a Gaussian manifold instead of Log-Euclidean kernel [6] on the SPD manifold.

4. Riemannian local difference vector on Gaussian manifold

In this section, we first present a method of sample representation by a single Gaussian model. Then, we develop two methods that transform the problem of calculating Riemannian local difference vector on Gaussian manifold (RieLDV-G) to the task of calculating Riemannian local difference vector on SPD manifold (RieLDV-S). Finally, we introduce four frequently-used Riemannian metrics on SPD manifold and show how to calculate RieLDV-S by using these metrics.

4.1. Gaussian model

Consider a set of N local features $F = \{f_1, f_2, ..., f_N | f_i \in R^k\}$. Using the maximum likehood method, we model their distribution via the following Gaussian model[26, 3]:

$$\mathcal{N}(f_i|\mu, \Sigma) = \frac{\exp(-\frac{1}{2}(f_i - \mu)^T \Sigma^{-1}(f_i - \mu))}{\sqrt{(2\pi)^k \det(\Sigma)}} \tag{6}$$

where $\mu = \frac{1}{N}\sum_{i=1}^{N} f_i$ and $\Sigma = \frac{1}{N-1}\sum_{i=1}^{N}(f_i - \mu)^T(f_i - \mu)$ are the mean vector and covariance matrix, and det$(\cdot)$ denotes matrix determinant.

Table 1: The Abbreviations of different types of Riemannian local difference vectors.

Abbreviations	Full Names
RieLDV	Riemannian local difference vector
RieLDV-S	Riemannian local difference vector on SPD manifold
RieLDV-G	Riemannian local difference vector on Gaussian manifold
DE-RieLDV-G	Directly calculated Riemannian local difference vector on Gaussian manifold
IE-RieLDV-G	Indirectly calculated Riemannian local difference vector on Gaussian manifold

To calculate RieLDV-G, we develop two methods that transform the problem of RieLDV-G to the task of calculating RieLDV-S. In the first method, called directly calculated RieLDV-G (DE-RieLDV-G), we compute the local difference vector (LDV) relative to the mean vector and RieLDV-S relative to the covariance matrix respectively, and then concatenate these two components as the final DE-RieLDV-G. The second, called indirectly calculated RieLDV-G (IE-RieLDV-G), maps G_k via a Gaussian embdeding [3] into the space of SPD matrices, S_{k+1}^{++}, and then computes RieLDV-S. Finally, we use the resulting RieLDV-S as the IE-RieLDV-G. Table 1 summarizes the above abbreviations.

4.2. Calculating the Riemannian local difference vector on a Gaussian manifold

4.2.1. Directly calculated RieLDV-G (DE-RieLDV-G)

Let us consider a set of n k-dimensional Gaussian models: $\mathbb{X} = \{\mathcal{X}_1, \mathcal{X}_2, ..., \mathcal{X}_n | \mathcal{X}_i \sim \mathcal{N}(\mu_i, \Sigma_i)\}$. We compute RieLDV-G from the mean vectors and covariance matrices. For the set of mean vectors $M = \{\mu_1, \mu_2, ..., \mu_n\}$, the set of local difference vectors in the Euclidean space is defined by:

$$\mathbb{L}_E = \{L_E^1, L_E^2, ..., L_E^n\} \tag{7}$$

where $L_E^i = \mu_i - \tilde{\mu}$, and $\tilde{\mu} = \frac{1}{n}\sum_{p=1}^{n}\mu_p$. The set of covariance matrices $S = \{\Sigma_1, \Sigma_2, ..., \Sigma_n\}$ is considered as a set of points on an SPD manifold. The set of Riemannian local difference vectors on the SPD manifold:

$$\mathbb{L}_S = \{L_S^1, L_S^2, ..., L_S^n\} \tag{8}$$

where $L_S^i = \zeta(\Sigma_i, \mathrm{E}(S))$ is defined via Eq.(3). $\mathrm{E}(S)$ is the Fréchet mean of SPD matrices which will be introduced in Section 4.3. The set of DE-RieLDV-G is expressed as:

$$\begin{aligned} \mathbb{L}_{DG} &= \{L_{DG}^1, L_{DG}^2, ..., L_{DG}^n\} \\ where \quad L_{DG}^i &= [\alpha L_E^i; L_S^i] \end{aligned} \tag{9}$$

In Eq.(9), L_{DG}^i is the concatenation of L_E^i and L_S^i with weight $\alpha > 0$. The relative effect of the local difference vectors in the Euclidean space and RieLDV-S on the SPD manifold on the resulting DE-RieLDV-G can be adjusted by α.

4.2.2. Indirectly calculated RieLDV-G (IE-RieLDV-G)

The space of k-dimensional Gaussian models is a Riemannian manifold. Let $\mathcal{N}(\mu, \Sigma)$ be a Gaussian model with mean vector μ and covariance matrix Σ. Through a function Ω, $\mathcal{N}(\mu, \Sigma)$ is mapped to an affine matrix $\mathcal{A}$. Further, through the function Φ, $\mathcal{A}$ is mapped to an SPD matrix $\mathcal{S}$. It is an element on the SPD manifold S_{k+1}^{++}. By these two functions Ω and Φ, $\mathcal{N}(\mu, \Sigma)$ is uniquely designated as a $(k+1) \times (k+1)$ SPD matrix [27]. With the consideration of the relative effect of the mean vector and the covariance matrix, Wang et al.[3] introduced a parameter $\beta > 0$ in the function Ω:

$$\Omega(\beta) : \mathcal{N}(\mu, \Sigma) \rightarrow \mathcal{A} = \begin{bmatrix} P & \beta\mu \\ 0^T & 1 \end{bmatrix} \tag{10}$$

where P is defined by the Cholesky factorization of Σ, i.e., $\Sigma = PP^T$. Accordingly, the representation of $\mathcal{N}(\mu, \Sigma)$ via the Gaussian manifold embedding [3] has the following form:

$$\mathcal{N}(\mu, \Sigma) \sim \mathcal{S} = \begin{bmatrix} \Sigma + \beta^2\mu\mu^T & \beta\mu \\ \beta\mu^T & 1 \end{bmatrix} \tag{11}$$

Table 2: Metrics and associated gradients on the SPD manifold.

Metric	$\delta^2(X,Y)$	$\nabla_X \delta^2$
AIRM [15]	$\|\log(X^{-1/2}YX^{-1/2})\|_F^2$	$2X^{1/2}\log(X^{-1/2}YX^{-1/2})X^{1/2}$
Stein [28]	$\log\det(\frac{X+Y}{2})$ - $\frac{1}{2}\log\det(XY)$	$X(X+Y)^{-1}X - \frac{1}{2}X$
Jeffrey [29]	$\frac{1}{2}\mathrm{Tr}(X^{-1}Y) + \frac{1}{2}\mathrm{Tr}(Y^{-1}X)$ - k	$\frac{1}{2}X(Y^{-1} - X^{-1}YX^{-1})X$
LEM [30]	$\|\log(Y)$ - $\log(X)\|_F^2$	$X(\log(X)$-$\log(Y))$+$(\log(X)$-$\log(Y))X$

Accordingly, the space of k-dimensional Gaussian models has been embedded into SPD manifold S_{k+1}^{++}. For the set of N k-dimensional Gaussian models: $\mathbb{X} = \{\mathcal{X}_1,\mathcal{X}_2,...,\mathcal{X}_n|\mathcal{X}_i \sim \mathcal{N}(\mu_i,\Sigma_i)\}$, we have a set of their embedding matrices $\mathbb{S} = \{\mathcal{S}_1,\mathcal{S}_2,...,\mathcal{S}_n|\mathcal{S}_i \in S_{k+1}^{++}\}$. The set of IE-RieLDV-G is then given as:

$$\mathbb{L}_{IG} = \{L_{IG}^1, L_{IG}^2, ..., L_{IG}^n\} \tag{12}$$

where L_{IG}^i is the resulting IE-RieLDV-G obtained by calculating the RieLDV-S of i-th embedded SPD matrix $\mathcal{S}_i$ on SPD manifold S_{k+1}^{++}.

4.3. Calculating the Riemannian local difference vector on an SPD manifold

In this section, we introduce Riemannian metrics on an SPD manifold, as well as the corresponding gradients and Riemannian means. Specifically, the expected value of a set of SPD matrices is expressed using the Fréchet mean [31]. A real value $k \times k$ SPD matrix $X \in S_k^{++}$ satisfies $v^T X v \geq 0$ for all non-zero $v \in R^k$. The SPD manifold S_k^{++} can be considered as the interior of a convex cone in the $k(k+1)/2$-dimensional Euclidean space which is a non-linear Riemannian manifold [32, 33]. The Affine Invariant Riemannian Metric (AIRM) is the mostly studied Riemannian metric on SPD manifolds [15]. Beside AIRM, Log-Euclidean Metric (LEM) [30] and two types of Bregman divergence [34], namely Stein [28] and Jeffrey [29] divergence, are also widely used to analyze SPD matrices. Given two SPD matrices X, $Y \in S_k^{++}$, their distance computed via these four metrics (δ_A, δ_L, δ_S, δ_J) is described in Table 2. The gradients of δ_A^2 and δ_L^2 are reported in [15] and [30] respectively, and the gradients of δ_S^2 and δ_J^2 are derived in [35]. The gradients required for calculating RieLDV-S via Eq.(3) are summarized in Table 2.

Let us consider the following cost function for a set of points $\{x_i\}_{i=1}^m$. The Fréchet mean [31] is then defined as the local minimizer of this cost function.

$$c^* = \arg\min_c \sum_{i=1}^{m} \delta^2(c, x_i) \tag{13}$$

For a set of SPD matrices $\{X_i\}_{i=1}^m \in S_k^{++}$, the Fréchet mean with AIRM and Stein divergence are obtained by an iterative approach [15, 35]. An estimate of the Fréchet mean with AIRM at the $t+1$ step is obtained [15]:

$$\mu_{t+1} = \mu_t^{\frac{1}{2}} \exp(\frac{1}{m} \sum_{i=1}^{m} \log(\mu_t^{-\frac{1}{2}} X_i \mu_t^{-\frac{1}{2}})) \mu_t^{\frac{1}{2}} \tag{14}$$

where $\log(\cdot)$ and $\exp(\cdot)$ are the matrix logarithm and matrix exponential respectively. For the Stein divergence, the estimate at iteration $t+1$ is given as:

$$\mu_{t+1} = [\frac{1}{m} \sum_{i=1}^{m} (\frac{X_i + \mu_t}{2})^{-1}]^{-1} \tag{15}$$

The proof is presented in [35]. The Fréchet mean with Jeffrey divergence is obtained analytically as [10]:

$$\mu = P^{-\frac{1}{2}} (P^{\frac{1}{2}} Q P^{\frac{1}{2}})^{\frac{1}{2}} P^{-\frac{1}{2}} \tag{16}$$

where $P = \sum_i X_i^{-1}$ and $Q = \sum_i X_i$. The Fréchet mean with LEM is obtained analytically as [30]:

$$\mu = \exp(\frac{1}{m} \sum_{i=1}^{m} \log(X_i)) \tag{17}$$

For more details on the Fréchet mean with Jeffrey divergence and LEM, the interested readers can refer to [10, 30].

5. Experiments and Analysis

This section presents the results of comparative experiments on datasets relating to the task of image set classification. The four datasets are Cambridge hand-gesture [36], ETH-80 [37], Virus [38] and MDSD [39] datasets respectively.

5.1. Datasets and settings

Our first experiment involves the task of hand gesture recognition. We use the Cambridge hand-gesture dataset [36] that contains nine categories of samples and nine hundred image sets. For each class twenty image sets are chosen randomly as training samples and the remaining eighty image sets are reserved for testing.

In the ETH-80 dataset [37], there are eight categories of samples and eighty image sets. For each class, five image sets are randomly chosen for training and the remaining five image sets are used for testing.

In the Virus cell dataset [38], there are fifteen categories of samples and 100 gray images in each category. We divide the images of each category equally into five different image sets and obtain seventy-five image sets. For each class, three image sets are randomly chosen as training samples and the remaining two image sets for testing.

The MDSD dataset [39] has been used for the task of dynamic scene classification. Following the settings in [40], we test the method based on the protocol of seventy-thirty-ratio (STR) which chooses seven videos for training and three videos for testing in each class. We divide each video into about 60 clips. f_i^p (in figure 1 Step(d)) represents the pixel-wise features of frames at the i-th region of the p-th video clip.

In our experiment, an 11-dimensional local feature vector is extracted from the colour image I as follows:

$$\begin{aligned} F(x,y) = [x, y, I_R(x,y), |\frac{\partial I_R}{\partial x}|, |\frac{\partial I_R}{\partial y}|, I_G(x,y), \\ |\frac{\partial I_G}{\partial x}|, |\frac{\partial I_G}{\partial y}|, I_B(x,y), |\frac{\partial I_B}{\partial x}|, |\frac{\partial I_B}{\partial y}|] \end{aligned} \tag{18}$$

The first two components are the x and y coordinates of a pixel (x, y). $I_R(x, y)$, $I_G(x, y)$, $I_B(x, y)$ represent the corresponding color information and $|\frac{\partial I_R}{\partial x}|$, $|\frac{\partial I_R}{\partial y}|$, $|\frac{\partial I_G}{\partial x}|$, $|\frac{\partial I_G}{\partial y}|$, $|\frac{\partial I_B}{\partial x}|$, $|\frac{\partial I_B}{\partial y}|$ capture the first order gradients of the corresponding colour information. For the gray image (In Virus dataset [38]), the 5-dimensional feature vectors are extracted by:

$$F(x,y) = [x, y, I(x,y), |\frac{\partial I}{\partial x}|, |\frac{\partial I}{\partial y}|] \tag{19}$$

where the last three dimensions are the grayscale intensity and the corresponding first order gradients.

For our proposed RieCovDs, the size of the image is 240×320 on the Cambridge and MDSD datasets, 256×256 on the ETH-80 dataset, and 40×40 on the Virus dataset. We use the sliding window method to obtain the regions at different locations. The size of the sliding window is 60×80, 32×32, 5×5, 60×80 on the Cambridge, ETH-80, Virus and MDSD datasets. The step size is '30', '28', '5', '20' pixels in the horizontal direction and '40', '28', '5', '30' pixels in the vertical direction for these four datasets respectively. We set parameter α in Eq.(9) to be 0.2, 0.9, 1, 0.3 for the four datasets respectively and set β in Eq.(10) to be 0.7, 0.6, 0.4, 0.9 respectively. For each RieLDV-G $\mathbb{L}_G$, the normalization is performed using the transfer function: $sign(\mathbb{L}_G)\sqrt{|\mathbb{L}_G|}$, where $|\cdot|$ denotes an absolute value operation. The gradient of the geodesic distance function at X: $\bigtriangledown_X \delta^2$, is normalized by: $X^{-1/2}\bigtriangledown_X \delta^2 X^{-1/2}$.

5.2. Comparison with the conventional CovDs

For the comparative experiments with the conventional CovDs [6], five algorithms are used to demonstrate the merits of our proposed RieCovDs. They include four nearest neighbor (NN) algorithms based on the above four Riemannian metrics on the SPD manifold, and the well-known SVM classifier [41]. The different classifiers tested in our experiments are referred to as:

NN-AIRM: AIRM based Nearest Neighbor classifier.

NN-Stein: S divergence based Nearest Neighbor classifier.

NN-Jeffrey: J divergence based Nearest Neighbor classifier.

NN-LEM: LEM based Nearest Neighbor classifier.

Ker-SVM: Log-Euclidean Kernel [6] based SVM classifier.

Here, Ker-SVM is a one-vs-all SVM classifier [2] implemented by Wang et al. Following [6], we regularize the conventional covariance matrix by: $C^* = C + \lambda \times Tr(C)I$ and set $\lambda = 10^{-3}$ typically. If the minimum eigenvalue of the representation computed by our proposed RieCovDs is less than 10^{-9}, then we regularize it by setting $\lambda = 10^{-9}$.

[2] http://www.peihuali.org/publications/RAID-G/RIAD-G_V1.zip

For the covariance matrices of the regions and the resulting representation via the Gaussian embedding, we regularize them in a similar manner. The different descriptors in this paper are referred to as:

CovDs: Image set repesented by conventional CovDs [6].

RieCovDs-A-DE/IE: Riemannian CovDs via DE-RieLDV-G/IE-RieLDV-G and using AIRM to compute the component of RieLDV-S.

RieCovDs-L-DE/IE: Riemannian CovDs via DE-RieLDV-G/IE-RieLDV-G and using LEM to compute the component of RieLDV-S.

RieCovDs-S-DE/IE: Riemannian CovDs via DE-RieLDV-G/IE-RieLDV-G and using S divergence to compute the component of RieLDV-S.

RieCovDs-J-DE/IE: Riemannian CovDs via DE-RieLDV-G/IE-RieLDV-G and using J divergence to compute the component of RieLDV-S.

Table 3 shows the average recognition accuracies and standard deviations of different descriptors with different classifiers. The advantage of our proposed RieCovDs is evident from the results obtained using the same classifier. This shows the effectiveness of our proposed RieCovDs. Ker-SVM achieves the best recognition accuracy of 96.93%, 98.55%, 82.17%, 47.21% with RieCovDs-A-IE, RieCovDs-A-IE, RieCovDs-S-IE, RieCovDs-L-DE on the four datasets respectively.

In the next sub-section, we consider the results of the best classifier, namely the Ker-SVM classifier, and compare them with state-of-the-art (SOTA) methods, which include SOTA classifiers and other improved versions of the conventional CovDs for the task of image set classification.

5.3. Comparison with SOTA methods

In this section, we compare with the SOTA classifiers including Covariance Discriminative Learning (COV-LDA, COV-PLS) [6], Log-Euclidean Kernels for Sparse Representation (LogEKSR.Pol, LogEKSR.Exp and LogEKSR.Gau) [42], SPD Manifold Learning based on LEM (SPDML-LEM) [33], Log-Euclidean Metric Learning (LEML, LEML+COV-LDA, LEML+COV-PLS) [43], Riemannian Network for SPD Matrix Learning (SPDNet) [44] and Multiple Manifolds Metric Learning (MMML)

Table 3: A comparison of the conventional CovDs and RieCovDs using different classifiers.

Methods	Descriptors	Cambridge[36]	ETH-80[37]	Virus[38]	MDSD[39]
NN-AIRM	CovDs [6]	55.29±2.53	77.05±4.85	27.00±4.12	13.90±4.30
	RieCovDs-A-DE/IE	94.51±0.96 / 94.49±0.96	93.30±3.84 / 94.95±3.71	52.80±5.82 / 72.50±6.27	33.82±6.45 / 31.33±6.09
	RieCovDs-S-DE/IE	94.48±0.93 / 94.36±0.96	91.70±3.33 / 94.30±3.63	51.23±5.31 / 72.00±6.32	27.41±6.01 / 32.54±6.42
	RieCovDs-J-DE/IE	94.33±0.91 / 94.54±1.04	90.83±3.26 / 91.92±3.71	55.07±4.58 / 72.93±6.48	27.51±5.73 / 24.79±5.69
	RieCovDs-L-DE/IE	95.00±0.90 / 95.09±0.84	93.45±3.79 / 93.85±3.54	52.80±5.52 / 72.30±6.26	34.74±6.31 / 31.18±6.05
NN-Stein	CovDs [6]	43.57±2.76	65.27±5.18	25.80±4.68	13.44±4.39
	RieCovDs-A-DE/IE	92.18±1.16 / 93.65±0.89	93.25±3.85 / 94.58±3.64	46.07±4.33 / 72.90±6.71	34.67±6.42 / 31.08±6.22
	RieCovDs-S-DE/IE	92.54±1.13 / 93.68±0.89	91.65±3.00 / 94.47±3.56	45.97±4.43 / 72.83±6.69	27.49±6.36 / 32.21±6.31
	RieCovDs-J-DE/IE	89.78±1.37 / 93.24±0.97	90.93±3.17 / 92.02±3.55	47.53±4.78 / 74.37±6.59	28.64±5.82 / 25.41±6.23
	RieCovDs-L-DE/IE	93.00±1.00 / 93.90±0.97	92.53±3.58 / 94.03±3.44	46.87±5.56 / 74.53±6.47	34.74±6.77 / 30.13±5.87
NN-Jeffrey	CovDs [6]	83.80±1.51	87.73±4.91	32.20±5.71	19.08±4.99
	RieCovDs-A-DE/IE	89.04±1.30 / 89.70±1.27	93.50±3.62 / 95.12±3.85	61.67±7.07 / 61.93±6.29	32.36±6.04 / 33.74±6.37
	RieCovDs-S-DE/IE	89.89±1.19 / 90.02±1.24	92.52±3.53 / 94.35±3.70	60.90±6.48 / 61.93±6.23	27.36±5.94 / 32.56±6.54
	RieCovDs-J-DE/IE	86.11±1.80 / 88.11±1.40	90.57±3.67 / 91.68±3.84	61.73±6.47 / 61.77±6.29	25.15±5.45 / 20.33±5.09
	RieCovDs-L-DE/IE	89.11±1.25 / 89.65±1.19	93.45±3.62 / 93.70±3.83	63.67±6.35 / 63.37±6.61	33.69±6.07 / 33.51±6.26
NN-LEM	CovDs [6]	71.17±1.87	81.85±5.75	26.30±4.69	14.82±4.72
	RieCovDs-A-DE/IE	94.60±0.95 / 94.38±0.96	93.03±3.71 / 94.80±3.58	51.93±5.08 / 71.50±6.35	34.23±6.32 / 32.54±6.25
	RieCovDs-S-DE/IE	94.59±0.90 / 94.34±0.92	91.15±3.25 / 94.10±3.58	51.07±5.36 / 70.67±6.82	29.79±6.09 / 33.69±6.47
	RieCovDs-J-DE/IE	94.51±0.92 / 94.30±1.05	90.88±2.98 / 91.63±4.16	52.17±5.43 / 72.00±6.80	27.28±5.58 / 26.31±5.86
	RieCovDs-L-DE/IE	95.08±0.92 / 94.76±0.90	92.90±3.65 / 93.93±3.47	53.37±5.29 / 73.00±6.44	34.10±6.24 / 33.74±6.44
Ker-SVM	CovDs [6]	91.67±1.10	92.48±5.08	66.57±5.79	35.38±6.25
	RieCovDs-A-DE/IE	96.89±0.75 / **96.93±0.76**	98.25±2.53 / **98.55±2.17**	73.63±6.01 / 80.60±5.79	46.92±5.87 / 45.51±6.43
	RieCovDs-S-DE/IE	96.36±0.81 / 96.11±0.95	97.87±2.26 / 98.00±2.61	70.30±6.16 / **82.17±5.81**	40.67±6.44 / 42.26±6.28
	RieCovDs-J-DE/IE	94.20±0.96 / 94.09±1.00	94.53±2.90 / 96.52±2.58	78.07±6.01 / 77.73±7.09	38.59±5.31 / 37.97±6.48
	RieCovDs-L-DE/IE	96.86±0.75 / 96.70±0.83	98.35±2.44 / 98.00±2.86	72.70±6.04 / 78.37±5.29	**47.21±5.92** / 45.38±6.16

[45]. For these methods, we first resize all images to 20×20, and then use the intensity values as their features. The latent dimensions in COV-LDA/COV-PLS are set to $c-1/c+1$, where c is the number of classes. For these SOTA classifiers, we report the best accuracies by searching over their various parameter settings.

The methods in [21, 22, 23] are the other improved versions of the conventional CovDs for describing image sets. For the comparative experiments with SOTA descriptors, we use the accuracies of the Ker-SVM classifier tested on these representations as the final results. To generate the descriptor in [21], named CovDs-B [3] in this paper, we use the kernel matrix representation obtained with the RBF kernel as it has been shown to achieve a better accuracy [21]. To generate the descriptor in [23], named CovDs-C [4] , all images are resized to 24×24 and sub-image sets are obtained by a 6×6 sliding window with the spatial step of 2 pixels for CG, ETH-80 and MDSD datasets, and spatial step of 3 pixels for the Virus dataset. For the method in [22], named CovDs-P in this paper, the sliding window size and step size are the same as the settings of CovDs-C on the corresponding dataset of patch-level images. The CovDs-B [21] and CovDs-P [22] descriptors are regularized as the conventional CovDs.

As shown in Table 4, the experimental results demonstrate the superior performance of our method, measured in terms of recognition accuracy. The advantages of our methods are very clear on the Virus and MDSD datasets, where image samples contain a large amount of noise. Our method achieves the best recognition accuracies with low standard deviations on the four datasets.

5.4. Ablation study

The results in Table 3 and Table 4 demonstrate the superior performance of our proposed RieCovDs method. The results have been obtained by fine tuning the various meta parameters of the method, namely the balance parameter α in Eq.(9), β in Eq.(10), the sliding window/step size, the normalization of RieLDV-G $\mathbb{L}_G$ and gradient $\nabla_X \delta^2$.

[3] https://www.uow.edu.au/ leiw/

[4] https://github.com/Kai-Xuan/ComponentSPD/

Table 4: Average recognition accuracies and standard deviations of different classifiers.

Methods	Cambridge [36]	ETH-80 [37]	Virus [38]	MDSD [39]
COV-LDA [6]	90.25±1.64	93.95±4.30	46.40±5.76	34.10±5.90
COV-PLS [6]	88.95±1.26	94.23±4.63	62.84±5.99	36.74±5.62
LogEKSR.Pol [42]	92.32±1.19	95.00±3.28	58.53±6.54	36.23±6.81
LogEKSR.Exp [42]	92.23±1.18	95.10±3.20	59.03±6.38	36.59±6.88
LogEKSR.Gau [42]	92.33±1.18	95.18±3.30	61.80±6.35	37.95±6.83
LEML [43]	88.18±1.29	93.05±3.31	33.00±5.70	25.97±6.79
LEML+COV-LDA [43]	89.09±1.63	95.35±3.50	58.03±5.84	31.92±6.44
LEML+COV-PLS [43]	86.36±1.35	95.83±3.04	59.40±6.22	35.90±6.98
SPDML-LEM [33]	84.03±1.04	90.63±4.19	49.37±7.46	24.23±4.47
SPDNet [44]	92.03±1.46	95.50±3.69	59.70±4.58	33.76±5.04
MMML [45]	92.87±1.39	95.28±3.80	51.13±7.60	31.95±6.26
Ker-SVM + CovDs-B [21]	92.31±1.13	94.18±3.69	73.77±5.82	37.79±5.76
Ker-SVM + CovDs-P [22]	94.34±1.02	94.52±3.64	75.40±6.01	35.54±6.47
Ker-SVM + CovDs-C [23]	93.81±0.91	95.45±2.85	53.63±6.80	38.08±6.05
Ker-SVM + RieCovDs(Ours)	**96.93±0.76**	**98.55±2.17**	**82.17±5.81**	**47.21±5.92**

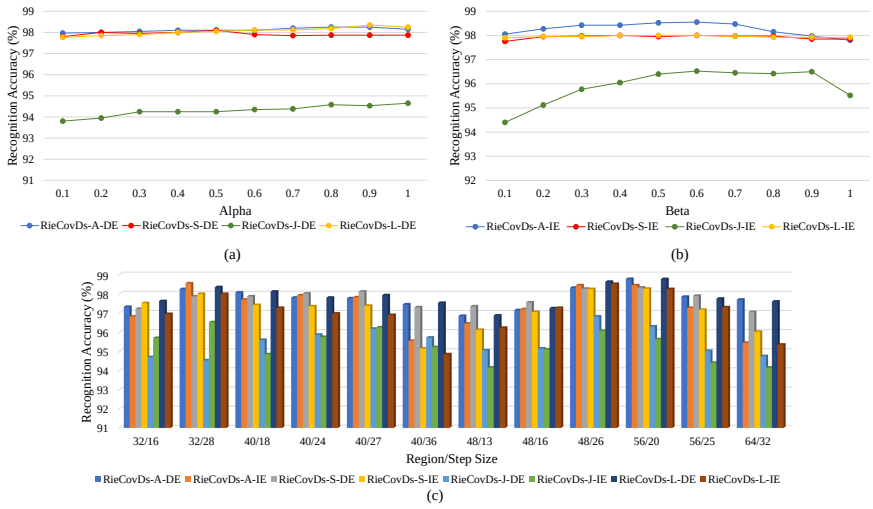

Figure 2: The effect of balance parameter α in Eq.(9), β in Eq.(10) and different region/step sizes on ETH-80 dataset.

Table 5: The effect of four kinds of data preprocessing on the Riemannian covariance.

Descriptors	KORG	NORR	NORG	NORR+NORG
RieCovDs-A-DE	92.48±3.92	94.95±3.59	96.62±2.50	98.25±2.53
RieCovDs-A-IE	94.65±3.26	96.75±2.90	98.17±2.58	98.55±2.17
RieCovDs-S-DE	96.87±2.47	97.17±3.03	95.98±2.82	97.87±2.26
RieCovDs-S-IE	95.58±2.79	95.82±3.14	97.32±2.57	98.00±2.61
RieCovDs-J-DE	88.80±4.09	94.15±3.06	90.40±3.29	94.53±2.90
RieCovDs-J-IE	92.45±3.79	94.67±2.72	92.03±3.62	96.52±2.58
RieCovDs-L-DE	91.35±4.37	94.65±3.59	96.77±2.52	98.35±2.44
RieCovDs-L-IE	92.90±4.49	96.20±2.96	97.65±2.65	98.00±2.86

In order to validate their contributions, we investigate the sensitivity of the recognition accuracies produced by the Ker-SVM classifier on ETH-80 dataset [37] to changes in these parameters. The results of RieCovDs with various α and β are illustrated in Figure 2 (a) and (b). Figure 2 (c) shows the effect of region size and step size on the recognition accuracies. The horizontal ordinate is in the form of 'a/b', which means that the sliding window size is $a \times a$ and step size is b pixels in both two directions. The results in figure 2 show that our proposed method is more sensitive to the sliding window/step size than the balance parameter α and β.

The effect of the four kinds of data preprocessing on Riemannian covariance was measured by comparing the results of the following variants: keeping the original form of RieLDV-G and gradient (KORG), normalizing only RieLDV-G (NORR), normalizing only gradient (NORG), normalizing RieLDV-G and gradient (NORR+NORG). The results are shown in Table 5. From the results, we can conclude that the normalization of RieLDV-G and gradient are the most important steps in the computation of Riemannian covariance.

6. Conclusion

In this paper, we have revisited the problem of representing images sets using covariance type descriptors. We discussed the merits of exploiting region based, rather than pixel based characteristics of image similarities and proposed the use of region

based Gaussian models as a means of image region description. As these regional models lie on a Gaussian manifold, we developed a novel approach to computing covariance on such manifolds. This involved defining the notion of deviations from the mean on Riemannian manifolds in the form of a Riemannian local difference vector, which led to the development Riemannian Covarince Descriptors (RieCovDs). The resulting RieCovDs capture the similarities between image regions more comprehensively than the conventional covariance descriptors. The experimental results show that RieCovDs are more discriminative than the conventional CovDs.

In the era where deep neural network (DNN) representations begin to dominate all machine learning approaches, it would be interesting to know how the proposed methodology would compare with DNN features. In our paper, the comparison with deep networks has been limited to the Riemannian network-based solution for SPD matrix learning (SDPNet in Table 4), with a favourable outcome for our proposed representation. We believe that there may be circumstances, where the volume of training data available would preclude the use of deep neural networks, and in such cases the more powerful representations derived by the proposed method are beneficial. However, when there are no constraints on training data availability, a comprehensive comparison with deep features would be very useful, not only from the point of view of performance, but also the speed of training, and the computational complexity of image set classification. We plan to carry out such a comparative study as part of our future research. We also intend to extend our proposed framework to other types of manifolds and applications.

7. ACKNOWLEDGMENTS

THE PAPER IS SUPPORTED BY THE NATIONAL NATURAL SCIENCE FOUNDATION OF CHINA (GRANT NO.61672265,U1836218), THE 111 PROJECT OF MINISTRY OF EDUCATION OF CHINA (GRANT NO. B12018), UK EPSRC GRANT EP/N007743/1, AND MURI/EPSRC/DSTL GRANT EP/R018456/1.

[1] J. Sivic, A. Zisserman, Video google: A text retrieval approach to object matching in videos, in: null, IEEE, 2003, p. 1470.

[2] W. Zhou, M. Yang, H. Li, X. Wang, Y. Lin, Q. Tian, et al., Towards codebook-free: Scalable cascaded hashing for mobile image search., IEEE Trans. Multimedia 16 (3) (2014) 601–611.

[3] Q. Wang, P. Li, L. Zhang, W. Zuo, Towards effective codebookless model for image classification, Pattern Recognition 59 (2016) 63–71.

[4] J. Hamm, D. D. Lee, Grassmann discriminant analysis: a unifying view on subspace-based learning, in: Proceedings of the 25th international conference on Machine learning, ACM, 2008, pp. 376–383.

[5] J. Hamm, D. D. Lee, Extended grassmann kernels for subspace-based learning, in: Advances in neural information processing systems, 2009, pp. 601–608.

[6] R. Wang, H. Guo, L. S. Davis, Q. Dai, Covariance discriminative learning: A natural and efficient approach to image set classification, in: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE, 2012, pp. 2496–2503.

[7] O. Tuzel, F. Porikli, P. Meer, Region covariance: A fast descriptor for detection and classification, in: European conference on computer vision, Springer, 2006, pp. 589–600.

[8] O. Tuzel, F. Porikli, P. Meer, Pedestrian detection via classification on riemannian manifolds, IEEE Transactions on Pattern Analysis & Machine Intelligence (10) (2008) 1713–1727.

[9] P. Li, Q. Wang, H. Zeng, L. Zhang, Local log-euclidean multivariate gaussian descriptor and its application to image classification, IEEE transactions on pattern analysis and machine intelligence 39 (4) (2017) 803–817.

[10] M. Faraki, M. T. Harandi, F. Porikli, A comprehensive look at coding techniques on riemannian manifolds, IEEE transactions on neural networks and learning systems (99) (2018) 1–12.

[11] P. Zheng, Z.-Q. Zhao, J. Gao, X. Wu, Image set classification based on cooperative sparse representation, Pattern Recognition 63 (2017) 206–217.

[12] H. Tan, Y. Gao, Z. Ma, Regularized constraint subspace based method for image set classification, Pattern Recognition 76 (2018) 434–448.

[13] Y. Rubner, C. Tomasi, L. J. Guibas, The earth mover's distance as a metric for image retrieval, International journal of computer vision 40 (2) (2000) 99–121.

[14] H. Nakayama, T. Harada, Y. Kuniyoshi, Global gaussian approach for scene categorization using information geometry, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE, 2010, pp. 2336–2343.

[15] X. Pennec, P. Fillard, N. Ayache, A riemannian framework for tensor computing, International Journal of computer vision 66 (1) (2006) 41–66.

[16] M. Faraki, M. T. Harandi, F. Porikli, Approximate infinite-dimensional region covariance descriptors for image classification, in: 2015 IEEE international conference on acoustics, speech and signal processing (ICASSP), IEEE, 2015, pp. 1364–1368.

[17] M. E. Hussein, M. Torki, M. A. Gowayyed, M. El-Saban, Human action recognition using a temporal hierarchy of covariance descriptors on 3d joint locations., in: IJCAI, Vol. 13, 2013, pp. 2466–2472.

[18] M. H. Quang, M. San Biagio, V. Murino, Log-hilbert-schmidt metric between positive definite operators on hilbert spaces, in: Advances in Neural Information Processing Systems, 2014, pp. 388–396.

[19] M. Harandi, M. Salzmann, F. Porikli, Bregman divergences for infinite dimensional covariance matrices, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1003–1010.

[20] K.-X. Chen, X.-J. Wu, R. Wang, J. Kittler, Riemannian kernel based nyström method for approximate infinite-dimensional covariance descriptors with application to image set classification, in: 2018 24th International Conference on Pattern Recognition (ICPR), IEEE, 2018, pp. 651–656.

[21] L. Wang, J. Zhang, L. Zhou, C. Tang, W. Li, Beyond covariance: Feature representation with nonlinear kernel matrices, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 4570–4578.

[22] Y. Li, R. Wang, Z. Cui, S. Shan, X. Chen, Spatial pyramid covariance-based compact video code for robust face retrieval in tv-series, IEEE Transactions on Image Processing 25 (12) (2016) 5905–5919.

[23] K.-X. Chen, X.-J. Wu, Component spd matrices: A low-dimensional discriminative data descriptor for image set classification, Computational Visual Media 4 (3) (2018) 245–252.

[24] Y. Dodge, The Oxford dictionary of statistical terms, Oxford University Press on Demand, 2006.

[25] K.-X. Chen, X.-J. Wu, J.-Y. Ren, R. Wang, J. Kittler, More about covariance descriptors for image set coding: Log-euclidean framework based kernel matrix representation, in: Proceedings of the IEEE International Conference on Computer Vision Workshops, 2019, pp. 0–0.

[26] Q. Wang, P. Li, W. Zuo, L. Zhang, Raid-g: Robust estimation of approximate infinite dimensional gaussian with application to material recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4433–4441.

[27] M. Lovric, M. Min-Oo, E. A. Ruh, et al., Multivariate normal distributions parametrized as a riemannian symmetric space, Journal of Multivariate Analysis 74 (1) (2000) 36–48.

[28] S. Sra, A new metric on the manifold of kernel matrices with application to matrix geometric means, in: Advances in neural information processing systems, 2012, pp. 144–152.

[29] Z. Wang, B. C. Vemuri, An affine invariant tensor dissimilarity measure and its applications to tensor-valued image segmentation, in: Computer Vision and Pat-

tern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, Vol. 1, IEEE, 2004, pp. I–I.

[30] V. Arsigny, P. Fillard, X. Pennec, N. Ayache, Geometric means in a novel vector space structure on symmetric positive-definite matrices, SIAM journal on matrix analysis and applications 29 (1) (2007) 328–347.

[31] F. Nielsen, R. Bhatia, Matrix information geometry, Springer, 2013.

[32] M. T. Harandi, M. Salzmann, R. Hartley, From manifold to manifold: Geometry-aware dimensionality reduction for spd matrices, in: European conference on computer vision, Springer, 2014, pp. 17–32.

[33] M. Harandi, M. Salzmann, R. Hartley, Dimensionality reduction on spd manifolds: The emergence of geometry-aware methods, IEEE transactions on pattern analysis and machine intelligence 40 (1) (2018) 48–62.

[34] B. Kulis, M. A. Sustik, I. S. Dhillon, Low-rank kernel learning with bregman matrix divergences, Journal of Machine Learning Research 10 (Feb) (2009) 341–376.

[35] A. Cherian, S. Sra, A. Banerjee, N. Papanikolopoulos, Jensen-bregman logdet divergence with application to efficient similarity search for covariance matrices, IEEE transactions on pattern analysis and machine intelligence 35 (9) (2013) 2161–2174.

[36] T.-K. Kim, R. Cipolla, Canonical correlation analysis of video volume tensors for action categorization and detection, IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (8) (2009) 1415–1428.

[37] B. Leibe, B. Schiele, Analyzing appearance and contour based methods for object categorization, in: Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on, Vol. 2, IEEE, 2003, pp. II–409.

[38] G. Kylberg, M. Uppström, I.-M. Sintorn, Virus texture analysis using local binary patterns and radial density profiles, in: Iberoamerican Congress on Pattern Recognition, Springer, 2011, pp. 573–580.

[39] N. Shroff, P. Turaga, R. Chellappa, Moving vistas: Exploiting motion for describing scenes, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE, 2010, pp. 1911–1918.

[40] H. Sun, X. Zhen, Y. Zheng, G. Yang, Y. Yin, S. Li, Learning deep match kernels for image-set classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3307–3316.

[41] C.-C. Chang, C.-J. Lin, Libsvm: a library for support vector machines, ACM transactions on intelligent systems and technology (TIST) 2 (3) (2011) 27.

[42] P. Li, Q. Wang, W. Zuo, L. Zhang, Log-euclidean kernels for sparse representation and dictionary learning, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 1601–1608.

[43] Z. Huang, R. Wang, S. Shan, X. Li, X. Chen, Log-euclidean metric learning on symmetric positive definite manifold with application to image set classification, in: International conference on machine learning, 2015, pp. 720–729.

[44] Z. Huang, L. Van Gool, A riemannian network for spd matrix learning, in: Thirty-First AAAI Conference on Artificial Intelligence, 2017.

[45] R. Wang, X.-J. Wu, K.-X. Chen, J. Kittler, Multiple manifolds metric learning with application to image set classification, in: 2018 24th International Conference on Pattern Recognition (ICPR), IEEE, 2018, pp. 627–632.