Geometry-Aware Similarity Learning on SPD Manifolds for Visual Recognition

Zhiwu Huang, *Member, IEEE*, Ruiping Wang, *Member, IEEE*, Xianqiu Li, Wenxian Liu,
Shiguang Shan, *Senior Member, IEEE*, Luc Van Gool, *Member, IEEE*,
and Xilin Chen, *Fellow, IEEE*

Abstract—Symmetric positive definite (SPD) matrices have been employed for data representation in many visual recognition tasks. The success is mainly attributed to learning discriminative SPD matrices encoding the Riemannian geometry of the underlying SPD manifolds. In this paper, we propose a geometry-aware SPD similarity learning (SPDSL) framework to learn discriminative SPD features by directly pursuing a manifold-manifold transformation matrix of full column rank. Specifically, by exploiting the Riemannian geometry of the manifolds of fixed-rank positive semidefinite (PSD) matrices, we present a new solution to reduce optimization over the space of column full-rank transformation matrices to optimization on the PSD manifold, which has a well-established Riemannian structure. Under this solution, we exploit a new supervised SPDSL technique to learn the manifold–manifold transformation by regressing the similarities of selected SPD data pairs to their ground-truth similarities on the target SPD manifold. To optimize the proposed objective function, we further derive an optimization algorithm on the PSD manifold. Evaluations on three visual classification tasks show the advantages of the proposed approach over the existing SPD-based discriminant learning methods.

Index Terms—Discriminative SPD matrices, Riemannian geometry, SPD manifold, geometry-aware SPD similarity learning, PSD manifold.

I. INTRODUCTION

RECENTLY, symmetric positive definite (SPD) matrices of real numbers have appeared in many branches of computer vision. Examples include region covariance matrices for pedestrian detection [1], [2] and texture categorization [3]–[5], joint covariance descriptors for action
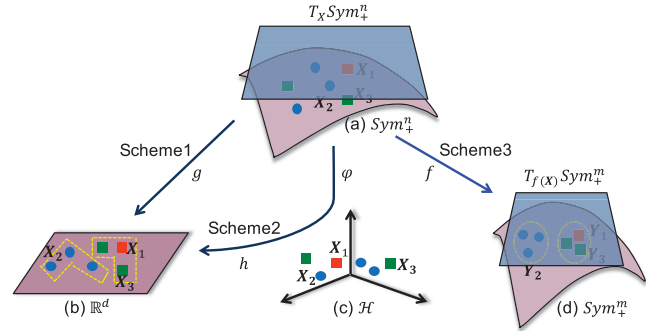

Fig. 1. Three different learning schemes on SPD manifolds. The first (a)→(b) flattens the original manifold Sym^n_+ by tangent space approximation and then learns a map g to discriminative Euclidean space $\mathbb{R}^d$. The second (a)→(c)→(b) is designed to first embed Sym^n_+ with an implicit map φ into RKHS $\mathcal{H}$ and then learn mapping h to discriminative Euclidean space $\mathbb{R}^d$. The last (a)→(d) aims to learn a map f from the original SPD manifold Sym^n_+ to a more discriminative SPD manifold Sym^m_+. Here, $X \in Sym^n_+$ and $f(X) \in Sym^m_+$ are the SPD matrices and $T_X Sym^n_+$ and $T_{f(X)} Sym^m_+$ are the tangent spaces.

recognition [5], [6], diffusion tensors for medical image segmentation [4], [7], [8] and image set based covariance matrices for video face recognition [9]–[11]. Due to the effectiveness of measuring the useful second-order information of processed data, such SPD matrix features have been shown to provide powerful representations for still images and dynamic videos in the field of computer vision.

However, such advantages of the SPD matrices are often accompanied by the challenges of the non-Euclidean data structure that underlies a specific Riemannian manifold [7], [8]. Applying Euclidean geometry directly to SPD matrices often results in poor performance and undesirable effects, such as the swelling of diffusion tensors in the case of SPD matrices [7], [12]. To overcome the drawbacks of the Euclidean representation, researchers [7], [8], [13] have conducted extensive studies on Riemannian metrics, e.g., the affine-invariant metric [7] and Log-Euclidean metric [8], to encode the Riemannian geometry of SPD manifolds so that the manifold-valued data can be treated appropriately.

By applying these classic Riemannian metrics, several studies extend traditional Euclidean algorithms to work on the manifolds of SPD matrices to learn more discriminative SPD matrices or their vector forms. To this end, these studies exploit effective methods on an SPD manifold by either flattening it via tangent space approximation [2], [14]–[16], [51] (See Fig.1 (a)→(b)) or mapping it into a

high-dimensional reproducing kernel Hilbert space (RKHS) [3], [4], [9], [17]–[20], [53] (See Fig.1 (a)→(c)→(b)). Obviously, both families of methods inevitably distort the geometric structure of the original SPD manifold by mapping the manifold into flat Euclidean space or high-dimensional RKHS. Thus, they often achieve sub-optimal solutions for the problem of learning the discriminative features of SPD manifolds. Furthermore, both learning schemes are computationally expensive due to the increased dimensions of the processed SPD matrices.

Several techniques were introduced for dimensionality reduction on Riemannian manifolds in [21]–[28]. For example, in [22]–[25], traditional nonlinear techniques were extended to their Riemannian counterparts by introducing various Riemannian geometry concepts, such as the Karcher mean, tangent spaces and geodesics, in locally linear embedding (LLE), Hessian LLE and Laplacian eigenmaps. As these methods do not define parametric mapping to low-dimensional space, they are limited to the transformation setting. In contrast, some studies, such as [26] and [27], encode the parametric mapping when pursuing the target low-dimensional manifold. However, they are originally designed for different types of Riemannian manifolds (such as Grassmann manifolds), not SPD manifolds.

For SPD data, the existing dimensionality reduction methods [5], [29], [52] aim to pursue a column full-rank transformation matrix to map the original SPD manifold to lower-dimensional discriminative SPD manifold, as shown in Fig.1 (a)→(d). However, since directly learning the manifold-manifold transformation matrix is difficult, [5] decomposed it to the product of an orthonormal matrix with a matrix in a general linear group and required the employed Riemannian metrics to be affine invariant. In this process, optimizing the manifold-manifold transformations is equivalent to optimizing over orthonormal projections. Although the additional requirement simplifies the optimization of the transformation, it does not only reduce the original solution space but inevitably excludes all non-affine-invariant Riemannian metrics,[1] such as the well-known Log-Euclidean metric, which has proved to be much more efficient than the affine-invariant metric [8]. While the work [29] exploited the Log-Euclidean metric under the same scheme, it attempted to learn a tangent map, which implicitly approximates the tangent space and hence introduces some distortions of the true geometry of SPD manifolds.

In this paper, under the last scheme (see Fig.1 (a)→(d)), we propose a new geometry-aware SPD similarity learning (SPDSL) framework to expand the problem domain of learning discriminative SPD features by exploiting either affine-invariant or non-affine-invariant Riemannian metrics on SPD manifolds. There are three main contributions in this work to realize the SPDSL framework:

- By exploiting the Riemannian geometry of the manifolds of fixed-rank positive semidefinite (PSD) matrices, our SPDSL framework provides a new solution to directly

learn the manifold-manifold transformation matrix. As no additional constraint is required, the optimal transformation is pursued in a favorable solution space, enabling a wide range of well-established Riemannian metrics to work.
- To fulfill the solution, a new supervised SPD similarity learning technique is proposed to learn the transformation by regressing the similarities of selected SPD pairs to the target similarities on the resulting SPD manifold.
- We derive an optimization approach that exploits the classic Riemannian conjugate gradient (RCG) algorithm on the PSD manifold to optimize the proposed objective function.

II. BACKGROUND

Let $Sym_n = \{H : H^T = H\}$ be a set of real, symmetric matrices of size $n \times n$ and $Sym_n^+ = \{X \in Sym_n : \omega^T X \omega \succ 0, \forall \omega \in \mathbb{R}^n, \omega \neq 0\}$ be a set of SPD matrices. The mapping space Sym_n is endowed with the usual Euclidean metric (i.e., inner product) $\langle H_1, H_2 \rangle = Tr(H_2^T H_1)$. As noted in [7] and [8], the set of SPD matrices Sym_n^+ is an open convex subset of Sym_n. Thus, the tangent space to Sym_n^+ at any SPD matrix in it can be identified with the set Sym_n. The smoothly varying family of inner products on each tangent space is known as the Riemannian metric, allowing the space of SPD matrices Sym_n^+ to yield a Riemannian manifold. Based on this Riemannian metric, the geodesic distance between two elements X_1, X_2 on the SPD manifold is generally measured by $\langle \log_{X_1}(X_2), \log_{X_1}(X_2) \rangle_{X_1}$. Several Riemannian metrics and divergences have been proposed to equip SPD manifolds. For example, the affine-invariant metric [7], Stein divergence [31], and Jeffereys divergence [17] are designed to be invariant to affine transformation of SPD manifolds; that is, for any $M \in GL(n)$ (i.e., the group of real invertible $n \times n$ matrices), the metric function δ_A has the property $\delta_A^2(X_1, X_2) = \delta_A^2(MX_1M^T, MX_2M^T)$. In contrast, the Log-Euclidean metric [8], Cholesky distance [32] and Power-Euclidean metric [32] are not affine invariant Riemannian metrics. Among these metrics, only the affine-invariant metric [7] and Log-Euclidean metric [8] define a true geodesic distance on SPD manifolds [4]. In addition, Stein divergence is widely used due to its favorable properties and strong performance in visual recognition tasks [31]. Therefore, this paper focuses on studying these three representative Riemannian metrics for the manifolds of SPD matrices.

Definition 1: By defining the inner product in the tangent space at one anchor point X_1 on the manifold of SPD matrices as $\langle H_1, H_2 \rangle_{X_1} = \langle X_1^{-1/2} H_1 X_1^{-1/2}, X_1^{-1/2} H_2 X_1^{-1/2} \rangle$ and the logarithm map as $\log_{X_1}(X_2) = X_1^{1/2} \log(X_1^{-1/2} X_2 X_1^{-1/2}) X_1^{1/2}$, the geodesic distance between two SPD matrices X_1, X_2 on the manifold is induced by the affine-invariant metric (AIM) as

$$\delta_a^2(X_1, X_2) = \| \log(X_1^{-1/2} X_2 X_1^{-1/2}) \|_{\mathcal{F}}^2. \tag{1}$$

Definition 2: The approximated geodesic distance between two SPD matrices X_1, X_2 on the SPD manifold is defined

---

[1]Although the recent extension work [30] studied this limitation, it merely introduced an approximated adaption of the Log-Euclidean metric.

using Stein divergence as

$$\delta_a^2(X_1, X_2) = \ln \det\left(\frac{X_1 + X_2}{2}\right) - \frac{1}{2}\ln \det(X_1 X_2). \quad (2)$$

Definition 3: By defining the inner product in the tangent space at the SPD point X_1 on the SPD manifold as $\langle H_1, H_2 \rangle_{X_1} = \langle D\log(X_1)[H_1], D\log(X_1)[H_2]\rangle$ ($D\log(X_1)[H]$ denotes the directional derivative) and the logarithm map as $\log_{X_1}(X_2) = D^{-1}\log(X_1)[\log(X_2) - \log(X_1)]$, the geodesic distance between two SPD matrices X_1, X_2 is derived by the Log-Euclidean metric (LEM) as

$$\delta_l^2(X_1, X_2) = \|\log(X_1) - \log(X_2)\|_{\mathcal{F}}^2. \quad (3)$$

III. Proposed Approach

In this section, we propose a new solution of Riemannian geometry-aware dimensionality reduction for SPD matrices and then present our supervised SPD similarity learning method under the solution. Finally, we give a detailed description of our optimization algorithm.

A. Riemannian Geometry-Aware Dimensionality Reduction on SPD Manifolds

Given a set of SPD matrices $X = \{X_1, \ldots, X_N\}$, where each matrix $X_i \in Sym_n^+$, and a transformation $W \in \mathbb{R}^{n \times m}$ ($m < n$) is pursued for mapping the original SPD manifold Sym_n^+ to a lower-dimensional SPD manifold Sym_m^+. Formally, this procedure attempts to learn the parameter W of a mapping in the form $f : Sym_n^+ \times \mathbb{R}^{n \times m} \to Sym_m^+$, which is defined as:

$$f(X_i, W) = W^T X_i W. \quad (4)$$

To ensure the resulting mapping yields a valid SPD manifold $Sym_m^+ \ni W^T X_i W \succ 0$, the manifold-manifold transformation W is required to be a column full-rank matrix $W \in \mathbb{R}_*^{n \times m}$.

Since the solution space is a non-compact Stiefel manifold $\mathbb{R}_*^{n \times m}$, where the distance function has no upper bound, directly optimizing on the manifold is infeasible. Fortunately, the conjugates (taking the form of WW^T) of column full-rank matrices span a compact manifold $Sym_n^+(m)$ of positive semidefinite (PSD) matrices, which is a quotient space of $\mathbb{R}_*^{n \times m}$ and has a well-established Riemannian structure. In contrast, by additionally assuming the transformation W to be orthogonal, as done in [5], Eqn.4 can be optimized on a compact Stiefel manifold, which is a subset of the non-compact Stiefel manifold $\mathbb{R}_*^{n \times m}$. Further, for affine-invariant metrics (e.g., AIM), optimizing on a Stiefel manifold can be reduced to optimizing over Grassmannian [5]. However, such an orthogonal solution space is smaller than the original solution space $\mathbb{R}_*^{n \times m}$, making the optimization theoretically yield a suboptimal solution of W. Thus, we perform optimization on the PSD manifold to search for the optimal solution of W. We now study the geometry of the PSD manifolds $Sym_n^+(m)$.

For all orthogonal matrices O of size $m \times m$, the map $W \to WO$ leaves WW^T unchanged. This property of W results in the equivalence class of the form $[W] = \{WO | O \in \mathbb{R}^{m \times m}, O^T O = I_m\}$, and yields a one-to-one correspondence with the rank-m PSD matrix $Q = WW^T \in Sym_n^+(m)$.

By quotienting this equivalence relation out, the set of rank-m PSD matrices $Sym_n^+(m)$ is reduced to the quotient of the manifold $\mathbb{R}_*^{n \times m}$ by the orthogonal group $\mathcal{O}(m) = \{O \in \mathbb{R}^{m \times m} | O^T O = I_m\}$, i.e., $Sym_n^+(m) = \mathbb{R}_*^{n \times m}/\mathcal{O}(m)$. With the studied relationship between $Sym_n^+(m)$ and $\mathbb{R}_*^{n \times m}$, the function $\phi : Sym_n^+(m) \to \mathbb{R} : Q \mapsto \phi(Q)$ is able to derive the function $g : \mathbb{R}_*^{n \times m} \to \mathbb{R} : W \mapsto g(W)$ defined as $g(W) = \phi(WW^T)$. Here, g is defined in the total space $\mathbb{R}_*^{n \times m}$ and descends as a well-defined function in the quotient manifold $Sym_n^+(m)$. Therefore, optimizing over the total space $\mathbb{R}_*^{n \times m}$ is reduced to optimizing on the PSD manifold $Sym_n^+(m)$, which has been thoroughly studied in several works [33]–[36]. Since a PSD manifold is a quotient space, each element on the PSD manifold is actually an equivalence class $[W]$. Thus, optimizing on the PSD manifold addresses W directly rather than WW^T. To more easily understand this point, one can take the well-known Grassmann manifold, where each element can also be represented by the equivalence class $[W]$ and the optimization on it seeks the solution of W directly, as an analogy. Similarly the optimization over the PSD manifold is actually direct optimization of W rather than $Q = WW^T$. Thus, we design an objective function directly on W instead of Q, as done in the existing works [33], [35] and the popular manifold optimization (manopt) toolbox.[2]

It can be further proved that the quotient space $Sym_n^+(m)$ presents the structure of a Riemannian manifold [33]. As a result, by endowing the total space $\mathbb{R}_*^{n \times m}$ with the usual Riemannian structure of Euclidean space (i.e., the inner product $\langle H_1, H_2 \rangle = Tr(H_2^T H_1)$), the quotient space $Sym_n^+(m)$ follows a Riemannian structure. The inner product occurs in the tangent space T_W of the manifold $\mathbb{R}^{n \times m}$. In the case of the manifold $Sym_n^+(m)$, the corresponding tangent space is decomposed into two orthogonal subspaces, the vertical space $\mathcal{V}_W = \{W\Omega | \Omega \in \mathbb{R}^{n \times m}, \Omega^T = -\Omega\}$ and the horizontal space $\mathcal{H}_W = \{H \in T_W | H^T W = W^T H\}$, to achieve the inner product $\langle H_1, H_2 \rangle$. This Riemannian metric allows several classic optimization techniques, such as the Riemannian conjugate gradient (RCG) algorithm [33], to work on the PSD manifold $Sym_n^+(m)$. For more detailed background on the Riemannian geometry of PSD manifolds, please refer to [33] and [35].

By exploiting the Riemannian geometry of the fixed-rank PSD manifold $Sym_n^+(m)$, we create the possibility of directly pursuing an optimal column full-rank manifold-manifold transformation matrix to solve the problem of dimensionality reduction of SPD features.

B. Supervised SPD Similarity Learning

As previously studied, under the proposed framework of dimensionality reduction of SPD features, a target SPD manifold Sym_m^+ of lower dimensionality can be derived. On the new SPD manifold Sym_m^+, the geodesic distance between the two original SPD points X_i, X_j is obtained by:

$$\hat{\delta}^2(X_i, X_j) = \delta^2(f(X_i, W), f(X_j, W)), \quad (5)$$

where $f(X_i, W)$ is the manifold-manifold transformation computed by Eqn.4, and δ is the geodesic distance induced

[2]The manopt toolbox is available at http://www.manopt.org.

by the commonly used affine-invariant or non-affine-invariant Riemannian metrics in Eqn.1, Eqn.2 and Eqn.3.

In this paper, we focus on the problem of supervised SPD similarity learning for more robust visual classification tasks, where SPD features have shown great power. Formally, for each SPD matrix $X_i \in Sym_n^+$, we define its class indicator vector: $y_i = [0, \ldots, 1, \ldots, 0] \in \mathbb{R}^c$, where the k-th entry is 1 and the other entries are 0, indicating that X_i belongs to the k-th class of c total classes. As discriminant learning techniques developed in Euclidean space, we assume that prior knowledge is known regarding the distances between pairs of SPD points on the new SPD manifold Sym_m^+. Let us consider the similarity or dissimilarity between pairs of SPD points: two SPD points are similar if the similarity based on the geodesic distance between them on the new manifold is larger, while two SPD points are dissimilar if their similarity is smaller.

Given a set of similarity constraints, our goal is to learn the manifold-manifold transformation matrix W that parameterizes the similarities of SPD points on the target SPD manifold Sym_m^+. For this purpose, we exploit the supervised criterion of centered kernel target alignment [37]–[39] to learn discriminative features on the SPD manifold by regressing the similarities of selected sample pairs to the target similarities. Formally, our supervised SPD similarity learning (SPDSL) approach is based on maximizing the following objective function:

$$\mathcal{J}(W) = \frac{\langle UG \circ k(W)U, G \circ (YY^T)\rangle_{\mathcal{F}}}{\|UG \circ k(W)U\|_{\mathcal{F}}}, \quad s.t. \ W \in \mathbb{R}_*^{n \times m}, \tag{6}$$

where $\langle \cdot \rangle_{\mathcal{F}}$ and $\|\cdot\|_{\mathcal{F}}$ are the Frobenius inner product and the Frobenius norm, respectively. The elements of matrix $k(W)$ encode the similarities of the SPD data, while the elements of YY^T present the ground-truth similarities of the involved SPD points. The matrix G is used to select the pairs of SPD points whose corresponding elements are 1. The matrix $U = I_N - \frac{\mathbf{1}_N \mathbf{1}_N^T}{N}$ is employed to center the data similarity matrix $k(W)$ and the similarity matrix YY^T on the labels. N is the number of samples, I_N is an identity matrix of size $N \times N$, $\mathbf{1}_N$ is a vector of size N with all entries being ones, $Y = [y_1, \ldots, y_N]^T$ is assumed to be centered, i.e., $U(YY^T)U \to YY^T$, for simplicity. In the following, we present the formulations of the matrices $k(W)$ and G in greater detail.

More specifically, the matrix $k(W)$ in Eqn.6 encodes the similarity between each pair of SPD points (X_i, X_j) on the SPD manifold Sym_n^+ and takes the form:

$$k_{ij}(W) = \exp(-\beta \hat{\delta}^2(X_i, X_j)), \tag{7}$$

where $\hat{\delta}^2(X_i, X_j)$ is computed using Eqn.5, β is typically fixed as $\beta = \frac{1}{\sigma^2}$, and σ is empirically set to the mean distance of the original training sample pairs. Eqn.7 respects a form of the Gaussian kernel function. Nevertheless, as the objective function Eqn.6 can be expressed as the sum of the similarity regression results of selected sample pairs, Eqn.7 serves as a tool to encode the similarities and is thus not necessarily positive definite (PD).

In practical applications, the computational burden of handling the full kernel matrix $k(W)$ on the SPD manifold scales quadratically with the size of the SPD training data. To address this problem, we exploit graph embedding [40] to select a limited number of data pairs to construct a sparse kernel matrix (non PD) with a large number of elements being zero. The matrix G is defined to select the pairs of SPD points for SPD similarity learning. By using this matrix, $G \circ k(W)$ can be regarded as the sparse kernel matrix, where the operation $\circ$ denotes the Hadamard product and the matrix $G = G_w + G_b$. Here, G_w and G_b are defined as:

$$G_w(i, j) = \begin{cases} 1, & \text{if } X_i \in N_w(X_j) \text{ or } X_j \in N_w(X_i) \\ 0, & \text{otherwise,} \end{cases} \tag{8}$$

$$G_b(i, j) = \begin{cases} 1, & \text{if } X_i \in N_b(X_j) \text{ or } X_j \in N_b(X_i) \\ 0, & \text{otherwise,} \end{cases} \tag{9}$$

where $N_w(X_i)$ is the set of v_w nearest neighbors of X_i that share the same class label as y_i, and $N_b(X_i)$ is the set of v_b nearest neighbors of X_i with different class labels from y_i. According to the theory of graph embedding [40], the within-class similarity graph G_w and the between-class dissimilarity graph G_b, respectively defined in Eqn.8 and Eqn.9, can encode the local geometric structure of the processing data space. Consequently, in addition to accelerating the discriminant similarity learning on the SPD manifolds, graph embedding can learn the discriminative information of SPD data and characterize the local Riemannian geometry of the underlying SPD manifold. The efficiency and effectiveness of the proposed discriminant learning approach on SPD manifolds is further studied in the experimental section.

C. Riemannian Conjugate Gradient Optimization

As discussed above, optimization in the solution space $\mathbb{R}_*^{n \times m}$ of the column full-rank transformation matrices in our objective function can be reduced to optimization on the Riemannian manifold of rank-m PSD matrices, $Sym_n^+(m)$. Therefore, in this section, we exploit the RCG algorithm [33] to optimize our objective function $\mathcal{J}(W)$ in Eqn.6 by deriving its corresponding gradient on the PSD manifold $Sym_n^+(m)$.

Algorithm 1 Optimization Algorithm

Input: The initial matrix W_0
1. $H_0 \leftarrow 0$, $W \leftarrow W_0$.
2. **Repeat**
3. $\quad H_k \leftarrow -\nabla_W J(W_k) + \eta \tau (H_{k-1}, W_{k-1}, W_k)$.
4. $\quad$ Line search along the geodesic γ in the direction H_k from $W_{k-1} = \gamma(k-1)$ to find $W_k = \arg\min_W \mathcal{J}(W)$.
5. $\quad H_{k-1} \leftarrow H_k$, $W_{k-1} \leftarrow W_k$.
8. **Until** convergence
Output: The optimized matrix W

Similar to the traditional conjugate gradient (CG) algorithm developed in Euclidean space, our employed RCG algorithm is an iterative procedure. As given in Algorithm1, an outline for the iterative part of the RCG algorithm is as follows: at the k-th iteration, we find W_k by searching

the minimum of $\mathcal{J}$ along the geodesic in direction $\boldsymbol{H}_{k-1}$ from $\boldsymbol{W}_{k-1}$, compute the Riemannian gradient $\nabla_W \mathcal{J}(\boldsymbol{W}_k)$ at this point, and then choose the new search direction $\boldsymbol{H}_k = -\nabla_W \mathcal{J}(\boldsymbol{W}_k) + \eta \tau(\boldsymbol{H}_{k-1}, \boldsymbol{W}_{k-1}, \boldsymbol{W}_k)$. In the iterative procedure, the Riemannian gradient $\nabla_W \mathcal{J}(\boldsymbol{W}_k)$ can be easily approximated from its corresponding Euclidean gradient $D_W \mathcal{J}(\boldsymbol{W}_k)$ by $\nabla_W \mathcal{J}(\boldsymbol{W}_k) = D_W \mathcal{J}(\boldsymbol{W}_k) - \boldsymbol{W}_k \boldsymbol{W}_k^T D_W \mathcal{J}(\boldsymbol{W}_k)$, and the operation $\tau(\boldsymbol{H}_{k-1}, \boldsymbol{W}_{k-1}, \boldsymbol{W}_k)$ is the parallel transport of tangent vector $\boldsymbol{H}_{k-1}$ from $\boldsymbol{W}_{k-1}$ to $\boldsymbol{W}_k$. For more details about the RCG algorithm, we refer readers to [5] and [33].

For now, we only need to compute the Euclidean gradient $D_W \mathcal{J}(\boldsymbol{W})$ for the proposed objective function $\mathcal{J}(\boldsymbol{W})$ in Eqn.6. To obtain the Euclidean gradient $D_W \mathcal{J}(\boldsymbol{W})$, we express its corresponding directional derivative $D_W \mathcal{J}(\boldsymbol{W})[\dot{\boldsymbol{W}}]$ in direction $\dot{\boldsymbol{W}}$. Formally, they are related by the following equality:

$$D_W \mathcal{J}(\boldsymbol{W})[\dot{\boldsymbol{W}}] = \langle D_W \mathcal{J}(\boldsymbol{W}), \dot{\boldsymbol{W}} \rangle. \qquad (10)$$

We compute the adjoint of the directional derivative to obtain the Euclidean gradient. By employing the standard properties of directional derivatives, $D_W \mathcal{J}(\boldsymbol{W})[\dot{\boldsymbol{W}}]$ can be derived by:

$$\begin{aligned} &D_W \mathcal{J}(\boldsymbol{W})[\dot{\boldsymbol{W}}] \\ &= \frac{\langle \boldsymbol{U}\boldsymbol{G} \circ D_W k(\boldsymbol{W})[\dot{\boldsymbol{W}}]\boldsymbol{U}, \boldsymbol{G} \circ (\boldsymbol{Y}\boldsymbol{Y}^T) \rangle_{\mathcal{F}} \|\mathcal{L}\|_{\mathcal{F}}}{\|\mathcal{L}\|_{\mathcal{F}}^2} \\ &\quad - \frac{\langle \mathcal{L}, \boldsymbol{G} \circ (\boldsymbol{Y}\boldsymbol{Y}^T) \rangle_{\mathcal{F}} \langle \frac{\mathcal{L}}{\|\mathcal{L}\|_{\mathcal{F}}}, \boldsymbol{U}\boldsymbol{G} \circ D_W k(\boldsymbol{W})[\dot{\boldsymbol{W}}]\boldsymbol{U} \rangle_{\mathcal{F}}}{\|\mathcal{L}\|_{\mathcal{F}}^2} \\ &= \langle D_W k(\boldsymbol{W})[\dot{\boldsymbol{W}}], \boldsymbol{U}\left(\frac{\boldsymbol{G} \circ (\boldsymbol{Y}\boldsymbol{Y}^T)}{\|\mathcal{L}\|_{\mathcal{F}}} - \frac{\mathcal{J}(\boldsymbol{W})\mathcal{L}}{\|\mathcal{L}\|_{\mathcal{F}}^2} \right)\boldsymbol{U} \rangle_{\mathcal{F}}, \quad (11) \end{aligned}$$

where $k(\boldsymbol{W})$ is formulated by Eqn.7, $\mathcal{L} = \boldsymbol{U}\boldsymbol{G} \circ k(\boldsymbol{W})\boldsymbol{U}$, $\langle \cdot \rangle_{\mathcal{F}}$ indicates the Frobenius inner product, and $\| \cdot \|_{\mathcal{F}}$ denotes the Frobenius norm.

Accordingly, the key issue in Eqn.11 is to estimate $D_W k(\boldsymbol{W})$. When δ in Eqn.5 is the geodesic distance of AIM defined in Eqn.1, the Euclidean gradient of $k(\boldsymbol{W})$ can be derived as:

$$\begin{aligned} &D_W k_{ij}(\boldsymbol{W}) \\ &= -4\beta k_{ij}(\boldsymbol{W})(\boldsymbol{B}_i \hat{\boldsymbol{X}}_i^{-1} - \boldsymbol{B}_j \hat{\boldsymbol{X}}_j^{-1}) \log(\hat{\boldsymbol{X}}_j^{-\frac{1}{2}} \hat{\boldsymbol{X}}_i \hat{\boldsymbol{X}}_j^{-\frac{1}{2}}), \quad (12) \end{aligned}$$

where $\boldsymbol{B}_i = \boldsymbol{X}_i \boldsymbol{W}$, $\hat{\boldsymbol{X}}_i = \boldsymbol{W}^T \boldsymbol{X}_i \boldsymbol{W} \in Sym_m^+$.

For other affine-invariant metrics, such as Stein divergence [31], the corresponding Euclidean gradient of $k(\boldsymbol{W})$ with the geodesic distance function δ defined in Eqn.2 can be computed by:

$$\begin{aligned} &D_W k_{ij}(\boldsymbol{W}) \\ &= -\beta k_{ij}(\boldsymbol{W})((\boldsymbol{B}_i + \boldsymbol{B}_j)\boldsymbol{A}_{ij}^{-1} - \boldsymbol{B}_i \hat{\boldsymbol{X}}_i^{-1} - \boldsymbol{B}_j \hat{\boldsymbol{X}}_j^{-1}), \quad (13) \end{aligned}$$

where $\boldsymbol{A}_{ij} = \boldsymbol{W}^T \frac{\boldsymbol{X}_i + \boldsymbol{X}_j}{2} \boldsymbol{W}$ and is therefore applicable in our proposed framework.

By endowing the SPD manifold with the non-affine invariant metric LEM, it seems difficult to calculate the Euclidean gradient of $D_W k(\boldsymbol{W})$ due to the matrix logarithms it contains. Thus, we study the computation of the Euclidean gradient for the LEM case in the following.

First, we decompose the derivative of LEM w.r.t. $\boldsymbol{W}$ into three derivatives with the trace form $Tr(\cdot)$:

$$\begin{aligned} &D_W(\|\log(\boldsymbol{W}^T \boldsymbol{X}_i \boldsymbol{W}) - \log(\boldsymbol{W}^T \boldsymbol{X}_j \boldsymbol{W})\|_F^2) \\ &= D_W(Tr(\log^2(\boldsymbol{W}^T \boldsymbol{X}_i \boldsymbol{W})) + D_W(Tr(\log^2(\boldsymbol{W}^T \boldsymbol{X}_j \boldsymbol{W})) \\ &\quad - 2D_W(Tr(\log(\boldsymbol{W}^T \boldsymbol{X}_i \boldsymbol{W})\log(\boldsymbol{W}^T \boldsymbol{X}_j \boldsymbol{W}))). \qquad (14) \end{aligned}$$

Proposition 1: The derivatives of the three trace forms $Tr(\cdot)$ in Eqn.14 can be computed by (Here, $\boldsymbol{B}_i = \boldsymbol{X}_i \boldsymbol{W}$, $\hat{\boldsymbol{X}}_i = \boldsymbol{W}^T \boldsymbol{X}_i \boldsymbol{W}$):

$$\begin{cases} D_W(Tr(\log^2(\hat{\boldsymbol{X}}_i)) = 4\boldsymbol{B}_i \mathrm{D}\log(\hat{\boldsymbol{X}}_i)[\log(\hat{\boldsymbol{X}}_i)]. & (15) \\ D_W(Tr(\log^2(\hat{\boldsymbol{X}}_j)) = 4\boldsymbol{B}_j \mathrm{D}\log(\hat{\boldsymbol{X}}_j)[\log(\hat{\boldsymbol{X}}_j)]. & (16) \\ D_W(Tr(\log(\hat{\boldsymbol{X}}_i)\log(\hat{\boldsymbol{X}}_j)) = 2\boldsymbol{B}_i \mathrm{D}\log(\hat{\boldsymbol{X}}_i)[\log(\hat{\boldsymbol{X}}_j)] \\ \qquad + 2\boldsymbol{B}_j \mathrm{D}\log(\hat{\boldsymbol{X}}_j)[\log(\hat{\boldsymbol{X}}_i)]. & (17) \end{cases}$$

Proof: The three formulas for the gradients with the matrix logarithm correspond to the three formulas with rotation matrices in [41, Sec. 5.3], where a detailed proof is given.

By using **Proposition 1.** (i.e., Eqn.15, Eqn.16, Eqn.17) and the sum rule of directional derivatives, we derive $D_W k(\boldsymbol{W})$, where δ is the geodesic distance of the LEM in Eqn.5 as:

$$\begin{aligned} D_W k_{ij}(\boldsymbol{W}) &= -4(\boldsymbol{B}_i \mathrm{D}\log(\hat{\boldsymbol{X}}_i)[\log(\hat{\boldsymbol{X}}_i) - \log(\hat{\boldsymbol{X}}_j)] \\ &\quad + \boldsymbol{B}_j \mathrm{D}\log(\hat{\boldsymbol{X}}_j)[\log(\hat{\boldsymbol{X}}_j) - \log(\hat{\boldsymbol{X}}_i)])\beta k_{ij}(\boldsymbol{W}). \end{aligned}$$

$$(18)$$

To calculate Eqn.18, we apply the block triangular matrix function developed in [42] to compute the form of $\mathrm{D}\log(\hat{\boldsymbol{X}})[\boldsymbol{H}]$, which is the directional (Fréchet) derivative of log at $\hat{\boldsymbol{X}} \in Sym_m^+$ along $\boldsymbol{H} \in Sym_n$. The following theorem shows that the directional derivative appears as the $(1, 2)$ block of the resulting big matrix when $f : \hat{\boldsymbol{X}} \mapsto \log(\hat{\boldsymbol{X}})$ is evaluated at a certain block triangular matrix.

Theorem 1: Let $f : \hat{\boldsymbol{X}} \mapsto \log(\hat{\boldsymbol{X}})$ be $2n - 1$ times continuously differentiable on $\mathbb{G}$ and let the spectrum of $\hat{\boldsymbol{X}}$ lie in $\mathbb{G}$, where $\mathbb{G}$ is an open subset of $\mathbb{R}$. Then

$$f\left(\begin{bmatrix} \hat{\boldsymbol{X}} & \boldsymbol{H} \\ 0 & \hat{\boldsymbol{X}} \end{bmatrix}\right) = \begin{bmatrix} f(\hat{\boldsymbol{X}}) & \mathrm{D}\log(\hat{\boldsymbol{X}})[\boldsymbol{H}] \\ 0 & f(\hat{\boldsymbol{X}}) \end{bmatrix}. \qquad (19)$$

Proof: The result was proved by Najfeld and Havel [43, Th. 4.11] under the assumption that f is analytic.

The directional derivative of the matrix logarithm can be easily computed using **Theorem 1**. The pseudo matlab code for computing $\mathrm{D}\log(\hat{\boldsymbol{X}})[\boldsymbol{H}]$ is: n = size(X, 1); Z = zeros(n); A = log([X, H; Z, X]); D = A(1:n, (n+1):end), where $D = \mathrm{D}\log(\hat{\boldsymbol{X}})[\boldsymbol{H}]$.

With the gradient formulas derived in Eqn.12, Eqn.13 and Eqn.18, the Euclidean gradient Eqn.11 of the objective function Eqn.6 for these cases can be computed to feed into the exploited RCG algorithm on the PSD manifold. Since the global convergence of the RCG algorithm has been thoroughly studied [44], we do not investigate it any further. The main time cost of the algorithm is computing the gradient in Eqn.11, $O(lk_0 n^2 m + lk_1 nm^2)$ (l is the iteration number, k_0 and k_1 denote the number of selected samples and pairs, respectively, and n and m indicate the dimensions of the original and target manifolds, respectively) in the LEM case. In the experiment,

we also study the running time of each iteration of the algorithm when varying the number of selected between-class pairs for each SPD sample.

IV. EXPERIMENTS

In this section, we study the effectiveness of the proposed geometry-aware SPD similarity learning (SPDSL) approach by conducting experimental evaluations of three visual classification tasks: face recognition, material categorization and action recognition.

In these three tasks, the SPD features have been shown to provide powerful representations for images and videos via set-based covariance [9]–[11], region covariance [1], [2] and joint covariance descriptors [5], [6]. Therefore, they are natural choices to evaluate the proposed SPDSL exploiting the AIM, Stein divergence and LEM.

To evaluate the effectiveness of the proposed SPDSL approach, we compare three categories of SPD-based learning methods: basic Riemannian metric baseline methods, kernel learning based SPD discriminant learning methods and dimensionality reduction based SPD discriminant learning methods:

1) Basic Riemannian metrics on SPD manifolds: Affine-invariant metric (AIM) [7], Stein divergence [31], Log-Euclidean metric (LEM) [8]
2) Kernel learning based SPD matrix learning methods: PLS-based covariance discriminative learning (CDL) [9], Riemannian sparse representation (RSR) [3] and Log-Euclidean kernels (LEK) [45]
3) Dimensionality reduction based SPD matrix learning methods: Log-Euclidean metric learning (LEML) [29] and SPD manifold learning (SPDML-AIM and SPDML-Stein) [5] with AIM and Stein divergence

Note that the proposed SPDSL belongs to the last category of SPD discriminant learning methods. As this paper focuses on the problem of supervised SPD discriminant learning, we report the performance of the original discriminant learning methods, such as SPDML, rather than performance when further coupling them with other classifiers, as done in [5]. In addition, to study the discriminant learning power of our proposed framework, we replace its supervised learning scheme with that of SPDML but still perform optimization on the exploited solution space. The adaptions of the proposed SPDSL are denoted SPDSL-AIM*, SPDSL-Stein* and SPDSL-LEM*.

For RSR, the parameter β is densely sampled around the order of the mean distance, and the parameter λ is sampled in the range of $[0.0001, 0.001, 0.01, 0.1]$. For LEK, there are three implementations based on polynomial, exponential and radial basis kernels, which are, respectively, denoted LEK-κ_{p_n}, LEK-κ_{e_n} and LEK-κ_g. For LEK-κ_{p_n} and LEK-κ_{e_n}, we densely sample the parameter n from 1 to 50. The parameters β in LEK-κ_g and λ in the three LEK versions are all tuned in the same way as RSR. For LEML, the parameter η is tuned in the range of $[0.1, 1, 10]$, and ζ is tuned from 0.1 to 0.5. For SPDML and our SPDSL method, the maximum iteration number of the optimization algorithm is set to 50, the parameter v_w is fixed as the minimum number of samples

Fig. 2. Video frames from the YTC video dataset [46].

in one class, and the dimensionality of the lower-dimensional SPD manifold and v_b are tuned by cross-validation. The parameter β in our method is set to $\beta = \frac{1}{\sigma^2}$, where σ is equal to the mean distance of all pairs of training data.

A. Face Recognition

In the first experiment, we use the YouTube Celebrities (YTC) video face database [46] to perform video face recognition. The dataset is challenging and widely used in video face recognition research. It has 1,910 video clips of 47 subjects collected from YouTube. Most of the clips contain hundreds of frames, as shown in Fig. 2, which include noise and are often low resolution, highly compressed, and low quality.

For the testing protocol, following [9], [10], [29], the dataset is randomly split into the gallery and the probe, which have 3 image sets and 6 image sets, respectively, for each subject. The process of random testing is repeated 10 times to evaluate the video face recognition.

In our experiment, each face image in a video is cropped into a 20×20 intensity image and is then histogram-equalized to eliminate lighting effects. Following [9], [29], we extract the set-based covariance matrix for each video sequence of frames in this dataset. To avoid matrix singularity, we add a small ridge δI to each covariance matrix Σ, where $\delta = 10^{-3} \times trace(\Sigma)$ and I is the identity matrix. In the literature, the mean face in each video has been proved to benefit video face recognition. Therefore, we improve the set-based covariance matrix feature by concatenating it with the mean to yield a $(d + 1)$-dimensional SPD matrix as $\begin{bmatrix} \Sigma + \mu\mu^T & \mu \\ \mu^T & 1 \end{bmatrix}$, where $\mu \in \mathbb{R}^d$ and $\Sigma \in Sym_+^d$ represent the mean and the covariance matrix of one image set. Note that the dimensions of the target manifolds for the dimensionality reduction methods are all set to 40 for the YTC database.

As can be seen from Table I, the baseline method LEM outperforms the other two baselines, AIM and Stein, in most cases, which demonstrates that LEM is more effective than the other two Riemannian metrics in the evaluation. The results in Table I also show that most of the kernel learning (Category2) and dimensionality reduction (Category3) methods improve the accuracy of the baselines: AIM, Stein and LEM. This demonstrates that learning discriminative SPD features in these methods is beneficial to visual recognition tasks.

Compared with the state-of-the-art kernel learning based methods CDL and RSR, the dimensionality reduction based methods LEML and SPDML perform worse in the face recognition task. In contrast, the proposed SPDSL improves LEML and SPDML by approximately 2% and 7%, respectively,

TABLE I

AVERAGE RANK-1 FACE RECOGNITION RATES (%) WITH THE STANDARD DEVIATION OF THE THREE CATEGORIES OF
COMPETING METHODS, INCLUDING THE PROPOSED SPDSL, ON THE YTC DATABASE

| Category1	AIM	Stein	LEM		
Accuracy	62.85 ± 3.46	61.46 ± 3.52	63.91 ± 3.25		
Category2	CDL [9]	RSR [3]	LEK-κ_{p_n} [45]	LEK-κ_{e_n} [45]	LEK-κ_g [45]
Accuracy	$\mathbf{72.67 \pm 2.47}$	$\mathbf{72.77 \pm 2.69}$	61.85 ± 3.24	62.17 ± 3.52	56.30 ± 3.62
Category3	LEML [29]	SPDML-AIM [5]	SPDML-Stein [5]		
Accuracy	70.53 ± 2.95	64.66 ± 2.92	61.57 ± 3.43		
The proposed	SPDSL-AIM*	SPDSL-Stein*	SPDSL-LEM*	SPDSL-AIM	SPDSL-Stein	SPDSL-LEM
Accuracy	64.27 ± 2.84	62.31 ± 3.48	69.32 ± 2.04	71.60 ± 2.45	71.03 ± 2.39	$\mathbf{72.29 \pm 1.58}$
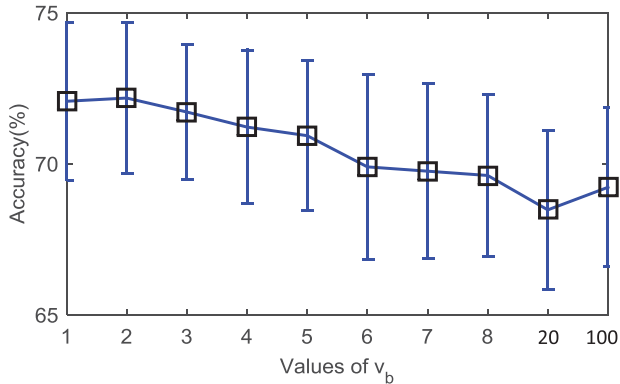
Fig. 3. Recognition accuracy of the proposed SPDSL-LEM on the YTC dataset for varying values of v_b (i.e., different sparse degrees of the involved kernel matrix $k(\boldsymbol{W})$).

Fig. 4. Running time of the proposed SPDSL-LEM on the YTC dataset for varying values of v_b (i.e., different sparse degrees of the involved kernel matrix $k(\boldsymbol{W})$).

and achieves comparable performance with CDL and RSR. By comparison with SPDML, the performances of the adaption of our proposed SPD similarity learning framework SPDSL-AIM* and SPDSL-Stein* are close to those of SPDML-AIM and SPDML-Stein. These results indicate that the former solution can be approximated by the latter solution when the involved Riemannian metric is affine invariant. Nevertheless, after using the proposed similarity learning technique, both SPDSL-AIM and SPDSL-Stein can clearly outperform the SPDML method. In addition, our SPDSL method can address cases where the SPD manifold is equipped with the non-affine Riemannian metric LEM, and we can observe that SPDSL-LEM* and SPDSL-LEM achieve higher accuracies in most cases. The big improvement of the proposed SPDSL over the adapted SPDSL* stems from the proposed similarity learning technique, which can learn more robust representations over the original SPD data of the YTC database.

Additionally, we study the effectiveness of the proposed SPDSL when varying its key parameter v_b. As shown in Fig.3, we present the behavior of the sparse (non PD) kernel cases on the YTC database for different values of v_b in the interval $[1, 8]$ and the values of 20 and 100 while fixing the parameter $v_w = 3$. When $k(\boldsymbol{W})$ achieves a full kernel matrix, the performance reaches 72.57%, which is close to the highest performance (72.29%) reached in the sparse kernel matrix cases (see Fig.3).

The efficiency of the proposed SPDSL technique is also studied. As shown in Fig.4, the running time is the average training time of each iteration of the optimization algorithm, which typically iterates 50 times. Specifically, we perform the
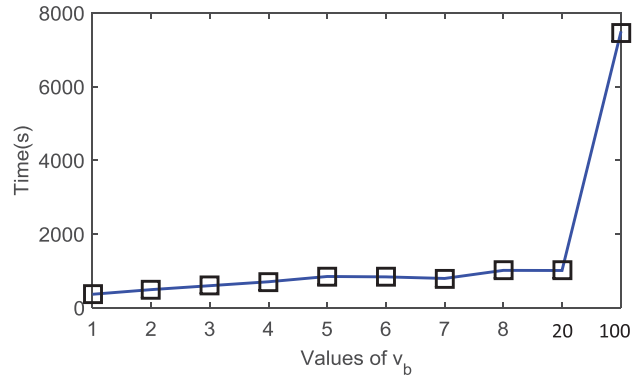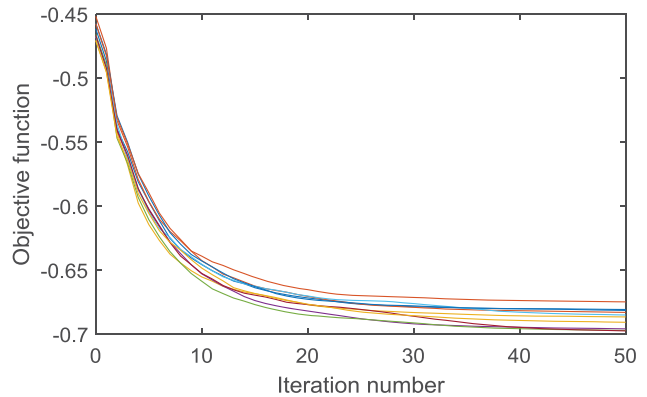

Fig. 5. Convergence behavior of the generalized RCG algorithm for the proposed SPDSL-LEM in 10 random testings of the YTC dataset with the parameter $v_b = 2$.

test on the YTC dataset and employ an Intel(R) Core(TM) i5-2400 (3.10GHz) PC. As the value of v_b increases, the running time increases substantially, especially when $k(\boldsymbol{W})$ is full, with a running time of approximately 13,975 seconds (i.e., approximately 30 times that of the case where $v_b = 2$ at each iteration, and extremely expensive when the algorithm iterates 50 times) on YTC. Hence, when huge datasets are involved, the sparse kernel case scales much better than the full (PD) kernel case with very slight gain/loss of accuracy.

We also investigate the convergence behavior of the exploited RCG algorithm for our SPDSL approach. As seen from the results in Fig.5, the optimization algorithm exploited on the PSD manifold converges to a favorable solution after several tens of iterations.

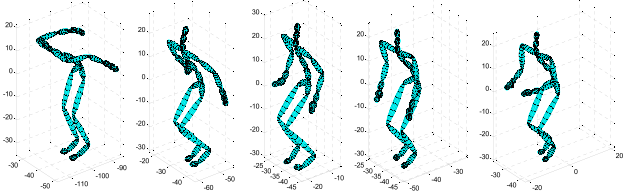Fig. 6.　Samples from the UIUC material dataset [47].

Fig. 7.　Hopping action from the HDM05 Motion Capture database [49].

B. Material Categorization

For the task of material categorization, we conduct experiments on the UIUC material dataset [47]. This dataset includes 18 subcategories of materials from four general categories: *bark, fabric, construction materials*, and *outer coat of animals*. Each subcategory contains 12 images taken at different scales. Several samples from this database are shown in Fig.6.

Region covariance matrices (RCMs) [1] and SIFT features [48] have been shown to be robust and discriminative for material categorization [47]. As in [5], we extract RCMs of size 128×128 using 128-dimensional SIFT features from grayscale images. Specifically, we resize each image to 400×400 and compute the dense SIFT descriptors on a grid with 4-pixel spacing (each patch size is 16x16, the number of angles is 8, and the number of bins is 4). One 128-dimensional SIFT feature is thus yielded in each grid point. For the dimensionality reduction methods, the dimensions of the target manifolds are all set to 40.

Following [5], we randomly select half of the images from each subcategory of the UIUC dataset as training data, and the remaining images are used as the testing data. The evaluation process is conducted 10 times in our experiment.

In Table I, for the competing methods, we report the average accuracies and the standard deviations of 10 random testings on the UIUC dataset. As concluded in the last evaluation, the proposed dimensionality reduction technique SPDSL improves the most-related method SPDML by 2%–4% and achieves comparable performance to the state-of-the-art methods. Compared to the last evaluation on the YTC database, the gains of SPDSL over SPDSL* are small, possibly because the UIUC dataset does contain large SPD data for each class, which reduces the discriminant power of the proposed similarity learning technique.

C. Action Recognition

To address the problem of human action recognition, we make use of the HDM05 motion capture database [49]. As shown in Fig.7, the dataset contains 2 337 sequences of 130 motion classes, e.g.,'clap above head', 'lie down floor',

'rotate arms' and 'throw basket ball', in 10 to 50 realizations executed by various actors.

The 3D locations of 31 joints of the subjects are provided over time, acquired at 120 frames per second. Following the previous works [5], [6], we represent an action of a K-joints skeleton observed over m frames by its joint covariance descriptor. This descriptor is an SPD matrix of size $3K \times 3K$, which is computed using the second-order statistics of 93-dimensional vectors concatenating the 3D coordinates of the 31 joints in each frame.

Similar to the evaluation protocol for the UIUC dataset, we conduct random evaluations 10 times, in which half of the sequences (approximately 1 100 sequences) are randomly selected for training data, and the rest are used for testing. On the HDM05 database, [5] only used 14 motion classes for evaluation while we consider 130 action classes. Thus, our reported recognition rates are slightly lower than those published in [5].

Table III summarizes the performances of the algorithms on the HDM05 dataset. In the evaluation, the dimensions of resulting manifolds achieved by the dimensionality reduction methods are all set to 30. In contrast to the results in the last two evaluations, CDL and RSR behave worse than the other methods because the testing and training data of the HDM05 database are more diverse. In this case, the kernel learning methods tend to overfit the training data and perform worse on the distinct testing data. The proposed SPDSL again improves the existing dimensionality reduction methods LEML and SPDML by 1%–3% and achieves state-of-the-art performance on the HDM05 database.

D. Discussion

In contrast to existing kernel learning based SPD discriminant analysis methods, such as CDL and RSR, our SPDSL is proposed for dimensionality reduction on SPD manifolds. In theory, our method overcomes the general drawbacks of these kernel learning methods, which are limited by the requirement of Mercer kernels and the high complexity that scales with the square of the training data size.

While our SPDSL and the existing SPDML and LEML are all designed for dimensionality reduction on SPD manifolds, there are some differences between them as follows.

First, compared with the related manifold learning method SPDML, our SPDSL framework proposes a more general solution and a more favorable objective function. This point is validated by the three evaluations. As shown in Table I, Table II and Table III, there are two key conclusions from the three visual recognition tasks:

a) With respect to the proposed new solution, the main benefits result from enlarging the search domain and enabling the use of non-affine-invariant metrics (e.g., LEM). While SPDSL* for the affine-invariant metrics AIM and Stein slightly improves SPDML (this may depend on the data), the gains of SPDSL*-LEM over the AIM and Stein cases are relatively obvious, i.e., 1.65%, 2.15%, and 6.21% on average for the three datasets.

b) The proposed objective function (working for similarity regression) is quite different from that (working for graph

TABLE II

AVERAGE RECOGNITION ACCURACIES (%) AND STANDARD DEVIATIONS OF THE THREE CATEGORIES OF COMPETING METHODS, INCLUDING THE PROPOSED SPDSL, ON THE UIUC DATABASE

Category1	AIM	Stein	LEM			
Accuracy	46.30 ± 2.90	42.87 ± 2.27	46.30 ± 2.86			
Category2	CDL [9]	RSR [3]	LEK-κ_{p_n} [45]	LEK-κ_{e_n} [45]	LEK-κ_g [45]	
Accuracy	$\mathbf{54.91 \pm 4.72}$	52.41 ± 4.03	48.89 ± 3.29	49.54 ± 3.67	49.63 ± 3.03	
Category3	LEML [29]	SPDML-AIM [5]	SPDML-Stein [5]			
Accuracy	52.53 ± 2.13	48.09 ± 1.82	49.17 ± 2.37			
The proposed	SPDSL-AIM*	SPDSL-Stein*	SPDSL-LEM*	SPDSL-AIM	SPDSL-Stein	SPDSL-LEM
Accuracy	50.00 ± 3.60	49.35 ± 2.47	50.28 ± 3.78	$\mathbf{52.31 \pm 3.55}$	51.57 ± 4.16	52.13 ± 3.49

TABLE III

AVERAGE RECOGNITION ACCURACIES (%) AND STANDARD DEVIATIONS OF THE THREE CATEGORIES OF COMPETING METHODS, INCLUDING THE PROPOSED SPDSL, ON THE HDM05 DATABASE

Category1	AIM	Stein	LEM			
Accuracy	42.70 ± 1.74	42.13 ± 2.63	43.98 ± 2.13			
Category2	CDL [9]	RSR [3]	LEK-κ_{p_n} [45]	LEK-κ_{e_n} [45]	LEK-κ_g [45]	
Accuracy	41.74 ± 1.92	41.12 ± 2.53	47.22 ± 1.62	46.87 ± 1.72	48.72 ± 3.00	
Category3	LEML [29]	SPDML-AIM [5]	SPDML-Stein [5]			
Accuracy	46.87 ± 2.19	47.25 ± 2.78	46.21 ± 2.65			
The proposed	SPDSL-AIM*	SPDSL-Stein*	SPDSL-LEM*	SPDSL-AIM	SPDSL-Stein	SPDSL-LEM
Accuracy	47.93 ± 2.62	46.35 ± 2.45	48.88 ± 3.18	48.09 ± 2.49	49.02 ± 2.93	$\mathbf{49.13 \pm 2.74}$

embedding) used in [5]. While it is difficult to theoretically prove the gains, we have empirically studied its superiority in the above evaluations. When comparing SPDSL with SPDSL*, the improvements are 2.13%, 1.03%, and 6.34% on average for the three evaluated databases.

Second, in contrast to LEML, which focuses on metric learning, our SPDSL learns discriminative similarities on SPD manifolds. Moreover, while LEML performs metric learning on the tangent space of SPD manifolds, the proposed SPDSL learns similarity directly on the SPD manifolds. Intuitively, the proposed SPDSL learning scheme more faithfully respects the Riemannian geometry of the data space and therefore is able to learn more appropriate SPD features for visual classification tasks. The evaluations on the three used databases have demonstrated some improvements of the proposed SPDSL over LEML.

V. CONCLUSIONS

We have proposed a geometry-aware SPD similarity learning (SPDSL) framework for more robust visual classification tasks. Under this framework, by exploiting the Riemannian geometry of PSD manifolds, we enable direct learning of the manifold-manifold transformation matrix. To achieve discriminant learning on the SPD features, this work devises a new SPDSL technique for SPD manifolds. With the objective of the proposed SPDSL, we derive an optimization algorithm on PSD manifolds to pursue the transformation matrix. Extensive evaluations demonstrate both the effectiveness and the efficiency of our SPDSL on three challenging datasets.

For future work, research on the relationship between the selected Riemannian metrics of PSD manifolds and SPD manifolds would be interesting for the problem of supervised SPD similarity learning. Furthermore, if the designed discriminant function on SPD features is neglected, learning the transformation on SPD features for object sets is equal to learning the projection on single object features. Thus, we can follow [50] to extend this work to learn hierarchical representations on object features by leveraging the current powerful deep learning techniques.

REFERENCES

[1] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 589–600.

[2] O. Tuzel, F. Porikli, and P. Meer, "Pedestrian detection via classification on Riemannian manifolds," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 10, pp. 1713–1727, Oct. 2008.

[3] M. T. Harandi, C. Sanderson, R. Hartley, and B. C. Lovell, "Sparse coding and dictionary learning for symmetric positive definite matrices: A kernel approach," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 216–229.

[4] S. Jayasumana, R. Hartley, M. Salzmann, H. Li, and M. T. Harandi, "Kernel methods on the Riemannian manifold of symmetric positive definite matrices," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 73–80.

[5] M. T. Harandi, M. Salzmann, and R. Hartley, "From manifold to manifold: Geometry-aware dimensionality reduction for SPD matrices," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 17–32.

[6] M. E. Hussein, M. Torki, M. A. Gowayyed, and M. El-Saban, "Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations," in *Proc. Int. Joint Conf. Artif. Intell.*, 2013, pp. 2466–2472.

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

[8] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, "Geometric means in a novel vector space structure on symmetric positive-definite matrices," *SIAM J. Matrix Anal. Appl.*, vol. 29, no. 1, pp. 328–347, 2007.

[9] R. Wang, H. Guo, L. S. Davis, and Q. Dai, "Covariance discriminative learning: A natural and efficient approach to image set classification," in *Proc. Comput. Vis. Pattern Recognit.*, 2012, pp. 2496–2503.

[10] R. Vemulapalli, J. Pillai, and R. Chellappa, "Kernel learning for extrinsic classification of manifold features," in *Proc. Comput. Vis. Pattern Recognit.*, 2013, pp. 1782–1789.

[11] J. Lu, G. Wang, and P. Moulin, "Image set classification using holistic multiple order statistics features and localized multi-kernel metric learning," in *Proc. Int. Conf. Comput. Vis.*, 2013, pp. 329–336.

[12] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, "Log-Euclidean metrics for fast and simple calculus on diffusion tensors," *Magn. Reson. Med.*, vol. 56, no. 2, pp. 411–421, 2006.

[13] S. Sra. (Oct. 2011). "Positive definite matrices and the S-divergence." [Online]. Available: https://arxiv.org/abs/1110.1773

[14] D. Tosato, M. Farenzena, M. Cristani, M. Spera, and V. Murino, "Multiclass classification on Riemannian manifolds for video surveillance," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 378–391.

[15] J. Carreira, R. Caseiro, J. Caseiro, and C. Sminchisescu, "Semantic segmentation with second-order pooling," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 430–443.

[16] A. Sanin, C. Sanderson, M. T. Harandi, and B. C. Lovell, "Spatiotemporal covariance descriptors for action and gesture recognition," in *Proc. WACV Workshop*, 2013, pp. 103–110.

[17] M. T. Harandi, M. Salzmann, and F. Porikli, "Bregman divergences for infinite dimensional covariance matrices," in *Proc. Comput. Vis. Pattern Recognit.*, 2014, pp. 1003–1010.

[18] M. Faraki, M. T. Harandi, and F. Porikli, "Approximate infinitedimensional region covariance descriptors for image classification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2015, pp. 1364–1368.

[19] M. T. Harandi, R. Hartley, B. Lovell, and C. Sanderson, "Sparse coding on symmetric positive definite manifolds using Bregman divergences," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 6, pp. 1294–1306, Jun. 2016.

[20] H. Minh, M. Biagio, and V. Murino, "Log-Hilbert-Schmidt metric between positive definite operators on Hilbert spaces," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 388–396.

[21] X. Liu, A. Srivastava, and K. Gallivan, "Optimal linear representations of images for object recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2003, pp. I-229–I-234.

[22] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment," *J. Shanghai Univ.*, vol. 8, no. 4, pp. 406–424, 2004.

[23] T. Zhang, J. Yang, D. Zhao, and X. Ge, "Linear local tangent space alignment and application to face recognition," *Neurocomputing*, vol. 70, nos. 7–9, pp. 1547–1553, 2007.

[24] A. Goh and R. Vidal, "Clustering and dimensionality reduction on riemannian manifolds," in *Proc. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–7.

[25] Z. Lai, Z. Jin, and W. K. Wong, "Tangent space discriminant analysis for feature extraction," in *Proc. IEEE 17th Int. Conf. Image Process.*, Sep. 2010, pp. 3793–3796.

[26] S. Jung, I. L. Dryden, and J. S. Marron, "Analysis of principal nested spheres," *Biometrika*, vol. 99, no. 3, pp. 551–568, 2012.

[27] Z. Huang, R. Wang, S. Shan, and X. Chen, "Projection metric learning on Grassmann manifold with application to video based face recognition," in *Proc. Comput. Vis. Pattern Recognit.*, 2015, pp. 140–149.

[28] Z. Lai, W. K. Wong, Y. Xu, J. Yang, and D. Zhang, "Approximate orthogonal sparse embedding for dimensionality reduction," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 4, pp. 723–735, Apr. 2016.

[29] Z. Huang, R. Wang, S. Shan, X. Li, and X. Chen, "Log-Euclidean metric learning on symmetric positive definite manifold with application to image set classification," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 720–729.

[30] M. Harandi, M. Salzmann, and R. Hartley, "Dimensionality reduction on SPD manifolds: The emergence of geometry-aware methods," *IEEE Trans. Pattern Anal. Mach. Intell.* [Online.] Available: http://ieeexplore.ieee.org/document/7822908/

[31] S. Sra, "A new metric on the manifold of kernel matrices with application to matrix geometric means," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 144–152.

[32] I. L. Dryden, A. Koloydenko, and D. Zhou, "Non-Euclidean statistics for covariance matrices, with applications to diffusion tensor imaging," *Ann. Appl. Statist.*, vol. 3, no. 3, pp. 1102–1123, 2009.

[33] P. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. Princeton, NJ, USA: Princeton Univ. Press, 2008.

[34] S. Bonnabel and R. Sepulchre, "Riemannian metric and geometric mean for positive semidefinite matrices of fixed rank," *SIAM J. Matrix Anal. Appl.*, vol. 31, no. 3, pp. 1055–1070, 2009.

[35] M. Journee, F. Bach, P. Absil, and R. Sepulchre, "Low-rank optimization on the cone of positive semidefinite matrices," *SIAM J. Optim.*, vol. 20, no. 5, pp. 2327–2351, 2010.

[36] G. Meyer, S. Bonnabel, and R. Sepulchre, "Regression on fixed-rank positive semidefinite matrices: A Riemannian approach," *J. Mach. Learn. Res.*, vol. 12, pp. 593–625, Feb. 2011.

[37] N. Cristianini, J. Shawe-Taylor, A. Elisseeff, and J. S. Kandola, "On kernel target alignment," in *Proc. Adv. Neural Inf. Process. Syst.*, 2001, pp. 367–373.

[38] C. Cortes, M. Mohri, and A. Rostamizadeh, "Algorithms for learning kernels based on centered alignment," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 795–828, Jan. 2012.

[39] F. Yger and M. Sugiyama. (Feb. 2015). "Supervised LogEuclidean metric learning for symmetric positive definite matrices." [Online]. Available: https://arxiv.org/abs/1502.03505

[40] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, Jan. 2007.

[41] N. Boumal and P. Absil, "A discrete regression method on manifolds and its application to data on SO(n)," *IFAC Proc. Volumes*, vol. 44, no. 1, pp. 2284–2289, 2011.

[42] A. Al-Mohy and N. Higham, "Computing the Fréchet derivative of the matrix exponential, with an application to condition number estimation," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 4, pp. 1639–1657, 2009.

[43] I. Najfeld and T. F. Havel, "Derivatives of the matrix exponential and their computation," *Adv. Appl. Math.*, vol. 16, no. 3, pp. 321–375, 1995.

[44] W. W. Hager and H. Zhang, "A survey of nonlinear conjugate gradient methods," *Pacific J. Optim.*, vol. 2, no. 1, pp. 35–58, 2006.

[45] P. Li, Q. Wang, W. Zuo, and L. Zhang, "Log-Euclidean kernels for sparse representation and dictionary learning," in *Proc. Int. Conf. Comput. Vis.*, 2013, pp. 1601–1608.

[46] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley, "Face tracking and recognition with visual constraints in real-world videos," in *Proc. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[47] Z. Liao, J. Rock, Y. Wang, and D. Forsyth, "Non-parametric filtering for geometric detail extraction and material representation," in *Proc. Comput. Vis. Pattern Recognit.*, 2013, pp. 963–970.

[48] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[49] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber, "Documentation: Mocap database HDM05," Univ. Bonn, Bonn, Germany, Tech. Rep. CG-2007-2, 2007.

[50] Z. Huang and L. Van Gool, "A Riemannian network for SPD matrix learning," *Assoc. Adv. Artif. Intell.*, vol. 2, no. 4, p. 6, 2017.

[51] M. E. Fathy, A. Alavi, and R. Chellappa, "Discriminative Log-Euclidean feature learning for sparse representation-based recognition of faces from videos," in *Proc. Int. Joint Conf. Artif. Intell.*, 2016, pp. 3359–3367.

[52] T. Matsuzawa, R. Relator, J. Sese, and T. Kato, "Stochastic dykstra algorithms for metric learning with positive definite covariance descriptors," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 786–799.

[53] J. Zhang, L. Wang, L. Zhou, and W. Li, "Learning discriminative Stein kernel for SPD matrices and its applications," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 5, pp. 1020–1033, May 2016.
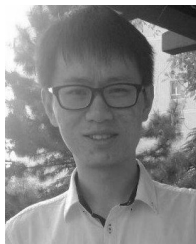
Zhiwu Huang received the B.S. degree in computer science and technology from Huaqiao University, Quanzhou, China, in 2007, the M.S. degree in computer software and theory from Xiamen University, Xiamen, China, in 2010, and the Ph.D. degree in computer science and technology from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2015. He has been a Post-Doctoral Researcher with the Computer Vision Laboratory, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, since 2015. His research interests include computer vision, Riemannian computing, metric learning, and deep learning.
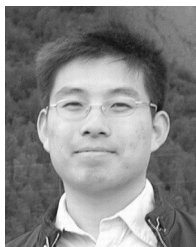
Ruiping Wang received the B.S. degree in applied mathematics from Beijing Jiaotong University, Beijing, China, in 2003, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, in 2010. He was a Post-Doctoral Researcher with the Department of Automation, Tsinghua University, Beijing, from 2010 to 2012. He also spent one year as a Research Associate with the Computer Vision Laboratory, Institute for Advanced Computer Studies, University of Maryland, College Park, from 2010 to 2011. He has been with the Faculty of the Institute of Computing Technology, Chinese Academy of Sciences, since 2012, where he is currently an Associate Professor. His research interests include computer vision, pattern recognition, and machine learning.
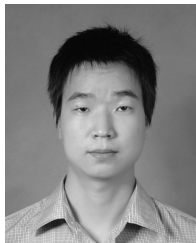
Xianqiu Li received the B.S. degree in statistics from the Huazhong University of Science and Technology, Wuhan, China, in 2013, and the M.S. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, in 2016. His interests are computer vision, pattern recognition, and machine learning.

Wenxian Liu received the B.S. degree in computer science from Xidian University, Xi'an, China, in 2012, and the M.S. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, in 2015. His interests cover computer vision, pattern recognition, and machine learning.

Shiguang Shan received the M.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 1999, and the Ph.D. degree in computer science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2004. He joined ICT, CAS in 2002 and has been a Professor since 2010. He is currently the Deputy Director of the Key Laboratory of Intelligent Information Processing, CAS. He has published over 200 papers in refereed journals and proceedings in the areas of computer vision and pattern recognition. His research interests cover computer vision, pattern recognition, and machine learning. He especially focuses on face recognition related research topics. He received the China's State Scientific and Technological Progress Awards in 2005 for his work on face recognition technologies. He has served as the Area Chair for many international conferences, including ICCV'11, ICPR'12, ACCV'12, FG'13, ICPR'14, and ICASSP'14. He served as a Workshop Co-Chair of ACCV14 and a Website Co-Chair of ICCV15. He is an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING, *Neurocomputing*, and the *EURASIP Journal of Image and Video Processing*.

Luc Van Gool received the degree in electro-mechanical engineering from Katholieke Universiteit Leuven in 1981. He is currently a Full Professor of computer vision with Swiss Federal Institute of Technology (ETH), Zurich, and Katholieke Universiteit Leuven, Belgium. He leads research and teaches at both places. He has authored over 300 papers in his field. He is also a co-founder of about ten spin-off companies. His main interests include 3D reconstruction and modeling, object recognition, and tracking and gesture analysis. He has been a program committee member of several, major computer vision conferences (e.g., the Program Chair of ICCV'05, Beijing, and the General Chair of ICCV'11, Barcelona, and ECCV'14, Zurich). He received several best paper awards (e.g., the David Marr Prize '98, Best Paper CVPR'07, the Tsuji Outstanding Paper Award ACCV'09, and Best Vision Paper ICRA'09). In 2015, he received the 5-yearly Excellence Prize for Applied Sciences from the Flemish Institute for Scientific Research (FWO).

Xilin Chen received the B.S., M.S., and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 1988, 1991, and 1994, respectively. He was a Professor with the Harbin Institute of Technology from 1999 to 2005. He was a Visiting Scholar with Carnegie Mellon University, Pittsburgh, PA, from 2001 to 2004. He has been a Professor with the Institute of Computing Technology, Chinese Academy of Sciences (CAS), since 2004. He is currently the Director of the Key Laboratory of Intelligent Information Processing, CAS. He has published one book and over 200 papers in refereed journals and proceedings in the areas of computer vision, pattern recognition, image processing, and multimodal interfaces. He served as an organizing committee/program committee member for more than 50 conferences. He is a fellow of the China Computer Federation. He was a recipient of several awards, including the China's State Scientific and Technological Progress Award in 2000, 2003, 2005, and 2012 for his research work. He is a leading Editor of the *Journal of Computer Science and Technology*, and an Associate Editor-in-Chief of the *Chinese Journal of Computers*.