Probability Distribution-Based Dimensionality Reduction on Riemannian Manifold of SPD Matrices

JIEYI REN AND XIAO-JUN WU, (Member, IEEE)
School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China
Corresponding author: Xiao-Jun Wu (wu_xiaojun@jiangnan.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61672265 and Grant U1836218, and in part by the 111 Project of Ministry of Education of China under Grant B12018.

ABSTRACT Representing images and videos with Symmetric Positive Definite (SPD) matrices and utilizing the intrinsic Riemannian geometry of the resulting manifold has proved successful in many computer vision tasks. Since SPD matrices lie in a nonlinear space known as a Riemannian manifold, researchers have recently shown a growing interest in learning discriminative SPD matrices with appropriate Riemannian metrics. However, the computational complexity of analyzing high-dimensional SPD matrices is nonnegligible in practical applications. Inspired by the theory of nonparametric estimation, we propose a probability distribution-based approach to overcome this drawback by learning a mapping from the manifold of high-dimensional SPD matrices to a manifold of lower-dimension, which can be expressed as an optimization problem on the Grassmann manifold. Specifically, we perform the dimensionality reduction for high-dimensional SPD matrices with popular Riemannian metrics and an affinity matrix constructed using an estimated probability distribution function (PDF) to achieve maximum class separability. The evaluation of several classification tasks shows the competitiveness of the proposed approach compared with state-of-the-art methods.

INDEX TERMS Probability distribution, Riemannian geometry, SPD matrices.

I. INTRODUCTION
In recent years, Symmetric Positive Definite (SPD) matrices have received significant attention in the field of computer vision, due to their capacity to characterize data variations. As an effective and powerful representation of images and videos, SPD matrices have been employed in diverse applications. Successful examples include covariance descriptors for recognizing faces [1]–[3], actions [4], textures [5], gestures [6], and pedestrians [7] as well as diffusion tensors for segmenting images [8], [9].

While offering advantages and promising properties for various computer vision tasks, the handling of SPD matrices on the basis of their intrinsic geometry becomes a challenge. As shown in several studies [10], [11], the space of SPD matrices is not a vector space but a non-linear Riemannian manifold. Though it is feasible to apply the Euclidean geometry to SPD matrices and employ matrix Frobenius norm to measure the similarity between them, poor performance and undesirable side-effects would be the end result [8], [12]. In other words, conventional learning methods based on the Euclidean structure are inadequate for analyzing SPD matrices owing to their neglect of the manifold geometry. In an attempt to generalize algorithms designed for Euclidean spaces to Riemannian manifolds, several researchers [2], [3], [13] have used popular Riemannian metrics [8], [9], [14] to account for the geometry of SPD matrices. Briefly, these works can be categorized into two major types: The first aims to obtain a locally flattened approximation of the manifold in the tangent space where linear geometry applies [6], [7], [15]; the second category of methods embed the manifold into a high-dimensional Reproducing Kernel Hilbert Space (RKHS) with an implicit map, and then use kernel-based methods [2], [3], [13], [16]. However, mapping SPD matrices either into a tangent space or into a high-dimensional RKHS will inevitably distort the geometry of the original manifold and the computational cost increases dramatically with the size of the SPD matrices.
FIGURE 1. Conceptual illustration of typical learning methods on the manifold of SPD matrices. Methods in type (a) flatten the original manifold $S_+^{n}$ with the tangent space $S^n_+$ (the space of symmetric matrices) and then apply Euclidean geometry. The type (b) methods embed the original manifold $S_+^{n}$ into an RKHS $H$ with an implicit kernel map, which makes it possible to exploit various kernel-based methods. The approach proposed in this article, ( type (c)), learns a mapping from the original manifold $S_+^{n}$ to a lower-dimensional one $S_+^{m}$ ($m < n$) directly, with the additional objective to enhance the discriminative power of the representation.

Since high-dimensional SPD matrices are often encountered in computer vision, another type of learning method [17]–[19] has been proposed to seek a column full-rank projection matrix mapping from the manifold of high-dimensional SPD matrices to a lower-dimensional and more discriminative one. The work in [18] aims to learn a tangent map that can directly transform the logarithm of SPD matrices from the original tangent space to a more discriminative one, and then formulates the learning as an optimization problem of seeking a Mahalanobis-like matrix. While a mapping from the original manifold to the target manifold can be derived from this tangent map, it distorts the geometry of SPD matrices to a certain extent. In contrast, the work [17] directly transforms the original manifold to a lower-dimensional, more discriminative manifold. It restricts the projection matrix to be orthonormal, and thus implements the learning as an optimization problem on the Grassmann manifold. It is noteworthy that the Riemannian metrics used in this framework should be affine-invariant, and the solution space of the optimal projection matrix is constrained. Thankfully, the work in [19] developed an adaptation to provide support for affine-invariant Riemannian metrics. The key concepts of typical learning methods on the manifold of SPD matrices are illustrated in Fig. 1.

In this article, we propose an improved manifold to manifold dimensionality reduction approach for SPD matrices that shares the same scheme as [17]–[19], yet we enhance the discriminative power by exploiting the probability distribution of SPD matrices to define a weighted affinity measure for the optimization. Compared with [17], [19], the main contribution of this work is twofold:

- By exploiting the probability distribution of SPD matrices, we provide a new solution to the problem of constructing the affinity matrix. As the probability distribution indicates the membership grade of a sample to a class, the resulting affinity matrix will be more accurate than the one based on a binary nearest neighbor selection.
- To reflect the notion of between-class dispersion more accurately, we employ a ‘sample-to-class’ membership to represent the dissimilarities of inter-class samples instead of the simplistic ‘sample-to-sample’ distance. Besides, we introduce a new strategy that focuses on the points from other classes that are closest to a given point. The number of samples drawn from such selected classes is determined by a control parameter.

II. PRELIMINARIES

This section provides an overview of Riemannian geometry on the manifold of SPD matrices and several popular Riemannian metrics.

A. RIEMANNIAN MANIFOLD OF SPD MATRICES

In mathematics, a differentiable manifold $M$ is a topological space locally similar to Euclidean space with a globally defined differential structure. For a point $P$ on the manifold, its tangent space $T_P M$ is defined by tangent vectors of all possible curves passing through $P$.

SPD matrices are symmetric matrices with the property that all their eigenvalues are positive. As mentioned above, the space of $n \times n$ SPD matrices (denoted by $S_+^{n}$) is not a subspace in Euclidean space but a specific Riemannian manifold with nonpositive curvature. More precisely, $S_+^{n}$ is a differentiable manifold endowed with an inner product on each tangent space. For a point $X \in S_+^{n}$, it is possible to define the derivatives of curves in its tangent space $T_X S_+^{n}$ by

\[ T_X S_+^{n} = \{ X + \lambda Y : X, Y \in S_+^{n}, \lambda \in \mathbb{R} \} \]
using the logarithm map \( \log_X : S^n_{++} \to T_X S^n_{++} \), which has an inner product \( \langle , \rangle_X \). A smoothly-varying family of inner products on all tangent spaces is known as a Riemannian metric. It enables us to define various geometric notions on the manifold, such as angles and lengths of curves. The geodesic distance between two points on the manifold specifies the length of the shortest curves connecting the two points. Such shortest curves are known as geodesics. They are similar to straight lines in \( \mathbb{R}^n \). Obviously, the geodesic distance induced by a Riemannian metric is a more appropriate measurement for comparing two SPD matrices than the Euclidean distance.

### B. RIEMANNIAN METRICS

A number of metrics on \( S^n_{++} \) have been recently proposed to utilize its nonlinear geometry, but not all of them define a true geodesic distance on the manifold. There are two most popular ones: Affine-Invariant Metric (AIM) [8] and Log-Euclidean Metric (LEM) [12].

Although AIM has high computational costs, in practice, its invariance to affine transformations is a desirable property for computer vision tasks. Given a point \( X \in S^n_{++} \), AIM is defined as

\[
(V, W)_X = \langle X^{-1/2}VX^{-1/2}, X^{-1/2}WX^{-1/2} \rangle = \text{Tr} \left( X^{-1}VX^{-1}W \right),
\]

where \( V, W \in T_X S^n_{++} \) and \( T_X S^n_{++} \) denotes the tangent space of \( S^n_{++} \) at point \( X \). The geodesic distance between two points \( X_i, X_j \in S^n_{++} \) is induced by AIM as

\[
\delta_i (X_i, X_j) = \| \log \left( X_i^{-1/2}X_jX_i^{-1/2} \right) \|_F,
\]

where \( \log (\cdot) \) is the matrix logarithm operator and \( \| \cdot \|_F \) denotes the matrix Frobenius norm.

Another widely used Riemannian metric LEM is expressed by classical Euclidean computation as

\[
\delta_l (X_i, X_j) = \| \log (X_i) - \log (X_j) \|_F.
\]

LEM also defines a true geodesic distance on \( S^n_{++} \) and is simple to use compared with AIM.

Besides, the Stein divergence [14] derived from the Bregman matrix divergence also approximates geodesic induced by AIM while being less expensive to compute and preserving invariance to affine transformations. The approximated geodesic distance is defined as

\[
\delta^2 (X_i, X_j) = \ln \det \left( \frac{X_i + X_j}{2} \right) - \frac{\ln \det (X_iX_j)}{2},
\]

where \( \det (\cdot) \) is the matrix determinant operator.

### III. PROPOSED APPROACH

In this section, we describe the proposed dimensionality reduction approach for SPD matrices. Firstly, we formulate the problem of dimensionality reduction for SPD matrices and then introduce the process of constructing an affinity matrix from the estimated probability distribution function (PDF) to yield maximum discriminability. Finally, we provide a detailed solution to the corresponding optimization problem.

#### A. DIMENSIONALITY REDUCTION FOR SPD MATRICES

Suppose we have a set of SPD matrices \( X = \{ X_1, \ldots, X_N \} \), \( X_i \in S^n_{++} \). To achieve the goal of dimensionality reduction, a map \( F : S^n_{++} \times \mathbb{R}^{n\times m} \rightarrow S^m_{++}, m < n \) from the original manifold \( S^n_{++} \) to the target manifold \( S^m_{++} \) of lower dimensionality with a parameter \( W \in \mathbb{R}^{n\times m} \) is defined as

\[
F (X_i, W) = W^TX_iW.
\]

Thus, we seek to find a transformation \( W \) to ensure the resulting space preserves a similar structure to that of the original data and forms a valid manifold of SPD matrices \( S^m_{++} \ni W^TX_iW > 0 \). Obviously, \( W \) is required to be a full-rank matrix. Analogous to [17], [19], we construct a \( N \times N \) real matrix \( G \) as an affinity matrix. Its element \( G(i, j) \) encodes a measure of similarity or dissimilarity of SPD matrices \( X_i \) and \( X_j \). The detailed information about this affinity matrix will be given in Sect. III-B.

With the affinity matrix \( G \), we define a cost function to describe the overall distribution of SPD matrices in the target manifold as

\[
L (W) = \sum_{i,j=1}^{N} G(i, j) \delta^2 \left( W^TX_iW, W^TX_jW \right),
\]

where \( \delta \) can be one of the geodesic distances induced by Riemannian metrics in (2), (3), or (4). To avoid potential degeneracies and following the common practice in dimensionality reduction, the solution is enforced to be orthogonal, i.e., \( W^TW = I_m \) and \( I_m \) is the identity matrix of size \( m \). It has been proven that this additional requirement entails no loss of generality for Riemannian metrics mentioned in Sect. II-B [17], [19]. Formally, the procedure of dimensionality reduction for SPD matrices can be expressed as a minimization problem

\[
W^* = \arg \min_W L (W), \text{ s.t. } W^TW = I_m.
\]

#### B. AFFINITY MATRIX

Although different criteria can be applied to construct the affinity matrix, we choose the supervised scenario since discriminative information is critical to classification. Suppose there are \( C \) different classes here and each point \( X_i \in S^n_{++} \) belongs to one of them. In this article, the affinity matrix \( G \) used by the cost function in (6) is defined as

\[
G = G_w - G_b,
\]

which resembles the Maximum Margin Criterion (MMC) [20]. \( G_w \) and \( G_b \) are affinity matrices for encoding within-class similarity and between-class dissimilarity respectively. The theory of Graph Embedding was exploited
by [17], [19] to define $G_w$ and $G_b$ in the form of symmetric binary matrices, constructed from nearest neighbor graphs as

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

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

where $N_w (X_i)$ is the set of $\lambda_w$ nearest intra-class neighbors of $X_i$, and $N_b (X_i)$ is the set of $\lambda_b$ nearest inter-class neighbors of $X_i$. $\lambda_w$ and $\lambda_b$ are used to reflect the trade-off between within-class compactness and between-class dispersion.

With this affinity matrix $G$, the mapping $f$ defined in (5) with an optimal $W$ learned from (7) is considered as a discriminative mapping for not only minimizing the intra-class distances but also maximizing the inter-class distances. Nevertheless, there are two problems to overcome. First, the binary nearest neighbor selection is too simplistic. It does not differentiate the degrees of (dis)similarity between two points, which makes the mapping sensitive to scattered samples (see case (a) in Fig. 2). Second, the between-class affinity matrix $G_b$ defined in (10) encodes the between-class dispersion via a few selected ‘sample-to-sample’ pair-wise distances rather than ‘sample-to-class’ membership (see case (b) in Fig. 2).

To overcome these problems, we introduce a novel definition for the affinity matrix based on the probability distribution of SPD matrices. Let $X$ be a set. A probability density function on $X$ is a function $f : X \rightarrow \mathbb{R}^+$ that satisfies $\int_X f (x) dx = 1$. We use Kernel Density Estimation (KDE) to obtain a fine-grained estimate $\hat{f}_p (x)$ of the distribution $f_p (x)$ about a set $X_p$. It is defined as

$$\hat{f}_p (x) = \frac{1}{N_p h} \sum_{i=1}^{N_p} K \left( \frac{x - x_i}{h} \right),$$

where $N_p$ is the number of points $x_i \in X_p$, $K$ is a kernel satisfying $K (x) \geq 0, \forall x \in X$, $\int K (x) dx = 1$, and $h$ is a bandwidth parameter (also known as the smoothing parameter). By utilizing the Gaussian kernel

$$K (x) = \frac{1}{\sqrt{2\pi \Sigma}} \exp \left( -\frac{x^T \Sigma^{-1} x}{2} \right),$$

we can combine the bandwidth parameter $h$ with the covariance $\Sigma$ (i.e., $\Sigma = I$ and $H = \text{diag}(h)$) and express the estimate as

$$\hat{f}_p (x) = \frac{1}{N_p \sqrt{2\pi \Sigma}} \exp \left( -\frac{(x - x_i)^T H^{-1} (x - x_i)}{2} \right).$$

As introduced in Section I, we propose to rely on the probability distribution of SPD matrices to define a new affinity matrix $\hat{G}$. However, a multi-dimensional KDE defined in (11) is based on a Euclidean structure and would not work for SPD matrices. If one attempts to interpret SPD matrices as living in a Euclidean space and use the multi-dimensional KDE for them, the underlying PDF would involve a ‘delta function’ which vanishes when one moves away from the manifold, and ‘becomes infinite’ on the manifold in order to have a proper normalization [21]. More specifically, the estimator will ‘try to converge’ to a singular PDF. In such circumstances, the PDF will be infinite on the manifold and zero elsewhere. In order to estimate the PDF on the manifold of SPD matrices, we can map them to the tangent space and transform them into vectors. Since KDE may produce less reliable and very sparse PDFs [22], we can learn a low-dimensional representation of these vectors through PCA. However, this solution is inconsistent with the aim of performing discriminative manifold-to-manifold dimensionality reduction for SPD matrices.

Recently, [23] analyzed the KDE in spaces of Gaussian distributions endowed with different metrics. Explicit expressions for the estimator were provided based on the Fisher metric defined in terms of multivariate centered Gaussian distributions $N_c$. Under the Fisher metric, the space of multivariate centered Gaussian distributions is isometric to the space of SPD matrices under the affine-invariant metric, and the explicit expression of the distance between two centered Gaussians is [8]:

$$\delta_{\text{Fisher}} (N_{\Sigma_1}, N_{\Sigma_2}) = \| \log (\Sigma_1^{-1/2} \Sigma_2 \Sigma_1^{-1/2}) \|,$$

where $\| \|$ is the norm associated with the scalar product $tr \{A B^T\}$. Suppose $\{(X_i, c_i)\}_{i=1}^M$ is the set of $n \times n$ SPD matrices with corresponding labels and $c_i \in \{1, \ldots, C\}$. The estimator given in [23] is defined as:

$$\hat{f}_c (X) = \frac{1}{N_c} \sum_{i} \frac{2^{-n(n+1)/2}}{h^n} \prod_{p<q} \frac{|\lambda_p - \lambda_q|}{\sinh \left( \frac{|\lambda_p - \lambda_q|}{2h} \right)} K \left( \frac{\delta_{\text{Fisher}} (X, X_i)}{h} \right),$$

where $\lambda_p$ and $\lambda_q$ are the eigenvalues of $\log (X_i^{-1/2} XX_i^{-1/2})$. 

![FIGURE 2. Two instances illustrating the deficiencies of the symmetric binary affinity matrix $G$. In case (a), the elements of $X_1$ with its nearest intra-class neighbours (including $X_1$ itself) in $G_w$ are set to 1 when $\lambda_w = 5$. However, distances between $X_1$ and other points vary dramatically, and this distribution cannot be uniquely reflected in $G_w$ with those binary elements. In case (b), the nearest inter-class neighbours of $X_1$ in class $c_1 (\lambda_b = 2)$ only include samples from class $c_2$. While the distance between $X_1$ in class $c_1$ to a sample from class $c_3$ is almost the same as the distances between $X_1$ in class $c_1$ to samples from class $c_2$. In another word, the affinity matrix $G_b$ could be unreliable since it encodes the inter-class dissimilarity with just a few pairs of samples, especially when $\lambda_b$ is small.](image-url)
Naturally, \( \hat{f}_c (X) \) indicates the membership grade of a sample \( X \) to a class \( c \), which describes the local geometrical structure of the data. We define the new within-class affinity matrix \( \hat{G}_w \) as
\[
\hat{G}_w (i, j) = \begin{cases} 
(\lambda_w - 1) \left| \hat{f}_c (X_i) - \hat{f}_c (X_j) \right|, & X_j \in \hat{N}_w (X_i) \\
1, & i = j \\
0, & \text{otherwise}
\end{cases}
\]
(16)
where \( \lambda_w \) is the set of \( \lambda_w \) 'furthest' intra-class neighbors of \( X_i \) that are selected on the basis of their proximities to \( X_i \) in the descending order.

Note that the definition of \( \hat{N}_w (X_i) \) here is different from \( N_w (X_i) \) in (9), where the latter ones are selected on the basis of their ascending order distance to \( X_i \) measured by a Riemannian metric. The reason for this change is that we intend to construct an improved \( \hat{G}_w \) by giving a priority to the weights of samples from the same class that are far away from each other. The proximity of two samples is computed by the absolute value of the difference between their membership grade to class \( c \) (i.e., \( \hat{f}_c (X) \)). For two intra-class samples, a large difference between their membership grade means that bringing them close to each other would contribute to achieving the objective of within-class compactness and vice versa. In other words, we mainly focus on keeping samples in overall close proximity instead of equally reducing their relative distances. As a consequence, \( \hat{G}_w \) encodes the within-class similarities by selecting pairs of intra-class samples with large variances and marks the corresponding elements with real-valued (rather than binary) coefficients and \( \sum \hat{G}_w (i, j) = \sum G_w (i, j) \). Analogously to (16), a new within-class affinity matrix \( \hat{G}_b \) is defined as
\[
\hat{G}_b (i, j) = \begin{cases} 
\lambda_b \hat{f}_b (X_i), & c_j \in \hat{C}_b (X_i) \text{ and } X_j \in \hat{T}_b (c_j) \\
\lambda_b \sum_{k=1}^{\lambda_b} \hat{f}_b (X_k), & \text{otherwise}
\end{cases}
\]
(17)
where \( \hat{C}_b (X_i) \) is the set of \( \lambda_b \) most confounded classes of \( X_i \) that are selected on the basis of the membership grade of \( X_i \) in these classes arranged in a descending order (i.e., \( \hat{f}_b (X_k) \)). \( \hat{T}_b (c_j) = \{ X_k \}_{k=1}^{\lambda_b} \) is the set of \( \lambda_b \) most representative samples in class \( c_j \) that are selected on the basis of the membership grade of samples in class \( c_j \) to itself in descending order (i.e., \( \hat{f}_b (X_k) \)). \( \lambda_b \) is used to control the number of samples selected from a class \( c \) to describe the between-class dissimilarity between \( c \) and another inter-class sample \( X \) and \( \sum \hat{G}_b (i, j) = \sum G_b (i, j) \). Finally, the affinity matrix \( G \) is calculated by \( G = \hat{G}_w - \hat{G}_b \).

### C. Optimization

As discussed in [17], the minimization problem in (7) is regarded as an optimization problem on a Grassmann manifold and can be solved by the Riemannian Conjugate Gradient (RCG) method [24]. Therefore, we need to compute the Jacobian matrices of \( \delta^2 \) with respect to \( W \). For AIM and the Stein divergence, the corresponding Jacobian matrices are derived as
\[
D_W \delta^2 \left( W^T X W, W^T X W \right) = 4 \left( X W \left( W^T X W \right)^{-1} - X W \left( W^T X W \right)^{-1} \right) \\
\times \log \left( W^T X W \left( W^T X W \right)^{-1} \right) .
\]
(18)
\[
D_W \delta^2 \left( W^T X W, W^T X W \right) = -X W \left( W^T X W \right)^{-1} \left( W^T X W \right)^{-1} + \left( X + X \right) W \left( W^T \left( \frac{X + X}{2} \right) W \right)^{-1} .
\]
(19)
Despite the fact that there is no analytical form for the gradient of LEM with respect to \( W \), an approximation \( \log \left( W^T X W \right) \approx W^T \log (X) W \) was given by [19] and the required gradient is given by
\[
D_W \delta^2 \left( W^T X W, W^T X W \right) \approx 4 \left( \log (X) - \log (X) \right) W \times W^T \left( \log (X) - \log (X) \right) W
\]
(20)

The procedure of our probability distribution-based dimensionality reduction approach is represented in Algorithm 1, where \( V_W L (W) \) is the gradient on Grassmann manifold and \( \tau (H, W_0, W_1) \) is the parallel transport of the tangent vector \( H \) from \( W_0 \) to \( W_1 \). The main computational cost of this algorithm is an iterative Riemannian optimization, which includes three major steps: evaluating the objective, computing the gradient, and performing RCG. A detailed analysis of the complexity of these three major steps can be found in [19].

### D. Covariance Descriptors

When we use SPD matrices as the input of our approach, we refer to the covariance matrices computed from feature vectors. Let \( S = [s_1, s_2, \ldots, s_r] \) be the data matrix of a set with \( r \) samples, where \( s_i \in \mathbb{R}^n \) denotes the \( n \)-dimensional vector representation of the \( i \)-th sample. A sample-based covariance matrix \( C \) is defined as
\[
C = \frac{1}{r - 1} \sum_{i=1}^{r} \left( s_i - \bar{s} \right) (s_i - \bar{s})^T ,
\]
(21)
where \( \bar{s} = \frac{1}{r} \sum s_i \) is the mean of all samples.

Except for being used as a fundamental concept in various fields of computer vision, the covariance matrix was initially introduced as a region descriptor by [5] over ten years ago. Given an \( w \times l \) image region, a feature vector \( s_i \in \mathbb{R}^n \)
Algorithm 1: Optimization Algorithm

**Input:**
A set of SPD matrices and their corresponding labels \(\{(X_i, c_i)\}_{i=1}^M, X_i \in S_{++}^n, c_i \in \{1, \ldots, C\}\).

Parameters \(m, h, \lambda_w, \lambda_b, \text{ and } \lambda_s\).

**Output:**
The optimal transform \(W\).

Construct affinity matrix using (16) and (17).

\[
W_{\text{old}} \leftarrow W_0 \quad \text{(i.e., the truncated identity matrix } I_{n \times m})
\]

\[
H_{\text{old}} \leftarrow 0.
\]

repeat

\[
H \leftarrow -\nabla_W L (W) + \eta \tau (H_{\text{old}}, W_{\text{old}}, W).
\]

Line search along the geodesic in the direction \(H\) from \(W\) to find \(W^* = \arg \min W L (W)\).

\[
H_{\text{old}} \leftarrow H, W_{\text{old}} \leftarrow W, W \leftarrow W^*.
\]

until Convergence

is extracted from each pixel of this region to describe its attributes such as location, color, gradient, filter response, etc. With these feature vectors, an \(n \times n\) covariance matrix \(C\) can be computed to represent this image region. There are several merits of choosing the covariance matrix as a region descriptor. First, it provides a natural manner to fuse various feature types. Second, it is robust against noise and outliers due to its averaging operation. Third, it allows a comparison between regions of different sizes to be carried out, which means a region covariance descriptor is independent of the size of the region. In addition to a region descriptor, the covariance matrix has recently been utilized as a form of general representation in an increasing number of computer vision tasks such as image set classification and frame-based video classification [15]. In this scenario, each individual image/frame in a given set will produce a corresponding feature vector, and the covariance matrix \(C\) is then computed from these image/frame-based vectors to represent this set.

In order to exploit the Riemannian geometry, the nonsingularity of \(C\) will be indispensable. When \(C\) is used as a region descriptor, its size \((n \times n)\) is determined by the dimensionality of the feature vectors. The total number of these feature vectors extracted from an image region is usually much greater than their dimensionality \((w \cdot l \gg n)\). This situation ensures the covariance matrix \(C\) is positive definite and allows it to be computed with a complexity of \(O(n^3)\) very fast. In contrast, using the covariance matrix as a general representation for image sets/videos usually suffers from a sufficient sample problem. This is because the feature vectors are no longer pixel-based and have a relatively high dimension, while the number of available images per set in most tasks is likely to be low. \(C\) is a valid SPD matrix only if \(r \approx n\), otherwise \(C\) would have zero eigenvalues and become indefinite. To avoid the latter case, a popular choice is to append a small-scaled identity matrix to the covariance matrix \(C\) as \(X = C + \lambda I\), \(\lambda = 0.001\). Although the covariance matrix \(C\) becomes strictly positive definite, the impact of such a regularization deepens as the size of the perturbation matrix grows [25]. As a consequence, expressive features of compact dimensionality are highly desired for the covariance representation. Feasible measures include, but are not limited to, reducing image resolution, mapping data to a low-dimensional space, performing proper feature extraction, applying suitable regularization methods, etc.

**IV. EVALUATIONS**

In this section, we study the effectiveness of the proposed probability distribution-based SPD dimensionality reduction (PDSPDDR) approach by presenting the results of major experiments involving three different computer vision tasks: object classification, material categorization, face recognition, and scene classification. To demonstrate the effectiveness of our approach, we compare it with several representative SPD manifold learning algorithms. We first briefly describe these methods and the datasets used in the experiments and then discuss the results.

**A. IMPLEMENTATION DETAILS**

For the benchmarking, the proposed PDSPDDR approach and three types of SPD manifold learning methods, including Riemannian metric-based methods, Riemannian kernel-based methods, and dimensionality reduction methods are evaluated with their corresponding classifiers. The Riemannian metric-based methods employ a simple NN classifier (the number of neighbors used in the NN classifiers is set to 1). These methods provide clear evidence of the benefits of the discriminative manifold mapping. The Riemannian kernel-based methods make use of kernels on the Riemannian manifold to exploit some classic vectorial algorithms (PLS, Sparse Representation). The dimensionality reduction methods seek a nonlinear mapping from the original manifold to another space with more discriminative power and usually formulate the learning procedure of this manifold mapping as an optimization problem. The algorithms evaluated in the experiments are listed as follows:

- **Riemannian metric-based methods:**
  - AIM: AIM-based NN classifier;
  - LEM: LEM-based NN classifier;
  - Stein: Stein divergence-based NN classifier.

- **Riemannian kernel-based methods:**
  - CDL: PLS-based Covariance Discriminative Learning [2];
  - RSR: Stein divergence-based Riemannian Sparse Representation [3];
  - LEKSR: Log-Euclidean Kernels for Sparse Representation [16].

- **Dimensionality reduction methods:**
  - LEML: Log-Euclidean Metric Learning with LEM-based NN classifier [18];
We also keep the target dimensionality \( m \) and the maximum iteration number of the RCG is set to 50, and the Gaussian kernel. For DRSPD and PDDRSPD, the maximum number of small objects is repeated 10 times with randomly selected gallery/probe combinations and the average recognition accuracies are reported.

### B. OBJECT CLASSIFICATION

For the first task, we used the ETH80 dataset [27]. This dataset contains images of 8 different categories and each category has 10 objects from everyday life: pears, tomatoes, cucumbers, cows, dogs, horses, cups, and cars. Each object includes 41 images captured under different views and some of them are shown in Fig. 3. We resize all the images to \( 20 \times 20 \) and treat each object as an image set. Therefore, the set-based covariance descriptors of size \( 400 \times 400 \) are generated from intensity images. Note that we use 4 randomly selected objects for the gallery and the other 6 objects for the probe, which is different from the setting in [2] to increase the difficulty of classification on this dataset. We use \( m = 30, \lambda_b = 3 \) on this dataset.

Table 1 shows the performance of our approach and other methods on this dataset. There is no doubt that kernel-based learning methods (i.e., CDL and RSR) are superb, while the Riemannian metrics based NN methods obtain lower performance. However, with the help of discriminative dimensionality reduction, the performance of NN methods is notably improved. Especially, our PDDRSPD not only outperforms DRSPD with a lead of over 2%, but also outperforms the kernel-based methods including CDL, RSR, and LEKSR. This demonstrates the benefits of encoding (dis)similarity between SPD matrices with their PDFs. The highest accuracy of 94.39% is obtained by RSR on the learned manifold with PDDRSPD, which is slightly better than the performance of MMDML on this dataset.

### C. MATERIAL CATEGORIZATION

For experiments in material categorization, we used the UIUC dataset [28]. The UIUC material dataset contains 18 classes of complex materials taken in the wild from four general categories: bark, fabric, construction materials, and fur of animals. Each class has 12 images with various geometric fine-scale details (see Fig. 4 for sample images). Following the same protocol as in [19], we randomly assigned half of the images of each class as gallery data and used the rest as probe data. To obtain robust and discriminative Region Covariance Matrices (RCMs) [5] for material categorization, we extract SIFT features [29] from grayscale images. Specifically, every image is resized to \( 400 \times 400 \) and then the dense SIFT descriptors are computed on a grid (the spacing is 4 pixels and the size of a bin is 4 pixels). Therefore, each grid point produces one 128-dimensional SIFT feature vector used for constructing RCMs of size \( 128 \times 128 \). We adopt the same parameters from [19] for the dimensionality reduction on this dataset.

As listed in Table 1, we can see the proposed PDDRSPD improves the most related method DRSPD by 1% ∼ 2% with NN and also achieves a comparable performance to the kernel-based learning methods (our PDDRSPD-AIM and PDDRSPD-Stein is better than RSR and LEKSR, but not as good as CDL). Note that the gap between the dimensionality reduction methods and the NN methods with the LEM solution is quite small, which is probably caused by deviations from the approximation of LEM. However, the low-dimensional SPD matrices with kernel sparse coding (PDDRSPD-RSR and PDDRSPD-LEKSR) outperform CDL.

MMDML: Multi-Manifold Deep Metric Learning for Image Set Classification [26];
DRSPD-AIM: Dimensionality reduction on SPD manifolds with AIM-based NN classifier [19];
DRSPD-LEM: DRSPD with LEM-based NN classifier [19];
DRSPD-Stein: DRSPD with Stein divergence-based NN classifier [19];
DRSPD-RSR: DRSPD with Stein divergence-based Riemannian Sparse Representation [19];
DRSPD-LEM: DRSPD with Stein divergence-based Riemannian Sparse Representation [19];
PDDRSPD-AIM: PDDRSPD with AIM-based NN classifier.
PDDRSPD-LEM: PDDRSPD with LEM-based NN classifier.
PDDRSPD-Stein: PDDRSPD with Stein divergence-based Riemannian Sparse Representation [19].
PDDRSPD-RSR: PDDRSPD with Stein divergence-based Riemannian Sparse Representation.
PDDRSPD-LEKSR: PDDRSPD with Log-Euclidean Kernels for Sparse Representation [19].

For a fair comparison, the main parameters of each algorithm are empirically tuned according to the original settings. For LEKSR, we use the implementation based on the Gaussian kernel. For DRSPD and PDDRSPD, the maximum iteration number of the RCG is set to 50, and the \( \lambda_w \) is set to the minimum number of samples in each class, while \( \lambda_b \) was determined by cross-validation (\( \lambda_b \leq \lambda_w \)). We also keep the target dimensionality \( m \) in PDDRSPD to be consistent with that in DRSPD for each dataset. In the pursuit of good performance, the \( \lambda_s \) of our PDDSPDDR is set to be the same as \( \lambda_w \). For each task, the experiments are repeated 10 times with randomly selected gallery/probe combinations and the average recognition accuracies are reported.

**FIGURE 3.** Samples from the ETH80 dataset.
TABLE 1. Recognition accuracies (%) with standard deviation of the three types of learning methods and the proposed PDDRSPD on three different datasets.

| Method           | ETH-80   | UIUCM    | YTC       | MDSD       |
|------------------|----------|----------|-----------|------------|
| AIM              | 74.58±4.04 | 47.31±2.33 | 56.23±2.16 | 51.75±7.08 |
| LEM              | 80.06±4.41 | 47.62±2.27 | 52.96±2.01 | 59.25±6.39 |
| Stein           | 75.00±4.82 | 44.67±2.61 | 55.49±2.67 | 37.91±7.25 |
| CDL            | 91.90±2.48 | 55.12±3.73 | 78.83±2.32 | 85.04±4.88 |
| RSR            | 92.02±3.13 | 52.41±2.97 | 76.94±3.24 | 84.82±6.35 |
| LEKSR         | 90.91±3.63 | 51.77±3.48 | 70.36±3.76 | 83.13±5.60 |
| LEML           | 86.48±3.09 | 50.35±2.80 | 71.65±2.72 | 72.01±6.79 |
| MMDML          | 94.12±3.45 | -         | 80.57±2.84 | 86.15±5.81 |
| DRSPD-AIM      | 90.08±3.32 | 51.39±2.73 | 73.54±2.06 | 72.48±6.26 |
| DRSPD-LEM      | 86.20±3.27 | 49.27±3.59 | 67.82±3.88 | 66.67±6.15 |
| DRSPD-Stein    | 88.32±3.54 | 50.03±3.16 | 72.01±3.15 | 73.21±6.37 |
| DRSPD-RSR      | 93.81±3.43 | 56.56±4.26 | 78.33±3.71 | 86.38±6.02 |
| DRSPD-LEKSR    | 92.94±3.54 | 54.59±3.51 | 73.51±3.66 | 85.49±5.86 |
| PDDDRSPD-AIM   | 92.75±3.04 | 53.11±3.07 | 75.23±2.95 | 76.73±6.51 |
| PDDDRSPD-LEM   | 89.13±3.20 | 50.74±3.23 | 68.82±3.10 | 71.90±6.20 |
| PDDDRSPD-Stein | 90.42±2.75 | 53.29±2.96 | 74.37±3.31 | 76.52±6.38 |
| PDDDRSPD-RSR   | 94.39±3.56 | 57.78±3.55 | 79.42±3.87 | 88.75±5.99 |
| PDDDRSPD-LEKSR | 93.71±3.60 | 55.29±3.85 | 74.07±3.49 | 87.48±5.67 |

FIGURE 4. Samples from the UIUC material dataset.

FIGURE 5. Samples from the YouTube Celebrities dataset.

D. FACE RECOGNITION
For the last task, we used the YouTube Celebrities (YTC) dataset [30]. It is a quite challenging and widely used video face dataset which has 1910 video sequences of 47 different persons collected from the YouTube website, such as actors, actresses, and politicians (see Fig. 5 for examples). Most video sequences are low resolution and recorded at a high compression ratio with different numbers of frames (vary from 8 to 400). The face region in each frame in a video sequence is first extracted and then resized into a 20 × 20 intensity image. Histogram equalization is applied to eliminate the illumination effect. Hence, each video sequence generated an image set of faces. Similar to the strategy in [19], we introduce the mean face in each image set to improve the set-based covariance descriptor by concatenating it with the mean to yield a 401-dimensional SPD matrix as

\[
X = \begin{bmatrix} C + \mu \mu^T & \mu \\ \mu^T & 1 \end{bmatrix},
\]

where \(\mu\) and \(C\) are the mean and covariance of one image set. Following the standard practice [31], the entire dataset is equally divided into five folds (with a minimal overlap). In each fold, one person has 3 randomly chosen image sets for the gallery and the other 6 image sets for the probe.

As can be seen from Table 1, our approach achieves a performance that is comparable to the state-of-the-art methods, with PDDDRSPD-AIM yielding the maximum performance of 75.23%. It is clear that our approach is better than some dimensionality reduction based methods (i.e., LEML and DRSPD) on this dataset and the performance of NN based on AIM and the Stein divergence is dramatically improved after being combined with our PDDDRSPD (up to 20%). With the superb capability of deep learning to model the nonlinearity of samples, MMDML achieves the best result of 80.57%. It is worth noting that our approach also boosts the performance of RSR to a similar level with MMDML (from 76.94% to 79.42% in the case of Stein divergence-based metric learning).

E. SCENE CLASSIFICATION
Scene classification is an important task in computer vision. In our work, we formulate it as an image set classification problem and evaluate it on the MDSD dataset. MDSD includes 13 classes with 10 small clips per class. These videos have large variations in terms of illumination, view, scale, and resolution (see Fig. 6 for example). Recently, deep neural networks have become the most popular and powerful approach for image classification, and the direct use of raw images is not able to provide robust and discriminative features for this task. Therefore, we use the output of the last pooling layer of ResNet [32] (pretrained on ImageNet [33]) as the descriptor
for each frame. The original-sized color images are used as the input to CNN and the dimensionality of the CNN output is reduced by PCA to 400. Following the same settings as in [22], we adopt a seventy-thirty-ratio (STR) protocol which randomly selects 7 videos for training and the remaining 3 videos for testing. We set \( m = 40 \) and \( \lambda_b = 7 \) on this dataset.

We report the experimental results on this challenging dataset in Table.1. Our approach succeeds in boosting the performance of the covariance descriptors computed from deep features with different classifiers. We can see that the PDDRSBD-LEM improves on the closest method, DRSPD-LEM, by 5\%, while PDDRSBD-RSR and PDDRSBD-LEKSR also outperforms the deep metric learning method MMDML with a lead up to 2\%. In addition, the performance of the Stein divergence-based NN is doubled by applying the proposed PDSPADD approach.

**F. PARAMETER SENSITIVITY**

Note that there are 5 major parameters in the proposed approach, and they were all set in our experiments in a simple, yet principled manner. Here we introduce the setting of these parameters and analyze the influence of some parameters on the overall performance.

There are 3 shared parameters in the proposed PDDRSBD and the most related DRSPD: the number \( \lambda_w \) of intra-class neighbors, the number \( \lambda_b \) of inter-class neighbors and the dimensionality \( m \) of the induced manifold. As observed in [19], the performance of learned low-dimensional SPD matrices monotonically increases when the value of \( \lambda_w \) is gradually raised from one to its upper bound. Therefore, it is intuitively clear to set \( \lambda_w \) to the minimum number of gallery samples in each class (maximize the notion of within-class compactness). Different from \( \lambda_w \), [19] also found that either too high or too low value of \( \lambda_b \) would result in performance degradation. Since \( \lambda_b \) is designed to keep a balance between with-class affinity matrix and between-class affinity matrix, we set \( \lambda_b \) to its optimum by empirical optimisation using a cross-validation performed in a limited range (1 \( \leq \lambda_b \leq \lambda_w \)). We also tested the sensitivity of both algorithms to \( m \). According to our observation on the UIUC material dataset, the final result is very insensitive to the variation of \( m \).

For the two new parameters \( \lambda_s \) and \( h \) introduced in the proposed approach, we investigated their influence on the overall performance. For this purpose, we applied our PDDRSBD-AIM on the UIUC material dataset with varying values of \( \lambda_s \) (from 1 to 6) and \( h \) (with \( \lambda_s = 1 \)). The results are depicted in Fig. 7. As described in Sect. III-B, \( \lambda_s \) is the number of samples selected from a certain class \( c_j \) to describe the inter-class dissimilarity between the given sample \( X_j \) and class \( c_j \), so it naturally has the same upper bound with \( \lambda_w \): the minimum number of gallery samples in each class. When \( \lambda_s = 1 \), the number of effective elements in \( G_i \) is equivalent to those in \( G_h \). We can see that the accuracy of PDSPADD-AIM is generally improved when the value of \( \lambda_s \) increases, and the maximum performance is achieved for \( \lambda_s = 6 \). Based on this, by simply letting \( \lambda_s = \lambda_w \) for the proposed PDDRSBD, good performance can be expected in all experiments.

In order to obtain the probability distribution of SPD matrices on the Riemannian manifold, \( h \) is an essential smoothing parameter for producing reliable estimates. As shown in Fig. 7, the sensitivity of our approach to \( h \) is relatively low. In any case, the performance can be improved by tuning \( h \).

In addition, we have also analyzed the efficiency of our approach. In particular, Fig. 8 shows the run time of 50 iterations of DRSPD and PDDRSBD (with \( \lambda_s = 1 \) and \( \lambda_s = 6 \)). We performed this comparison on the UIUC material dataset with a fixed gallery/probe combination. When the value of \( \lambda_s \) increases, the running time of PDSPDDR also increases, but the optimal cost becomes lower too. However, when \( \lambda_s \) is set to 1, our approach still gains a better result than DRSPD within comparable time, thanks to the affinity matrix constructed from PDFs.

Overall, our approach does require one extra parameter \( h \), compared with DRSPD, but we consider it not burdensome and its impact on the performance is very positive.

**V. CONCLUSION**

We have proposed a probability distribution-based dimensionality reduction approach for SPD matrices to map the manifold of high-dimensional SPD matrices into a manifold of lower-dimensional discriminative ones. By exploiting the PDFs of SPD matrices estimated with a KDE, we defined a novel affinity matrix to enhance the discriminative power of popular Riemannian metrics on the target manifold and formulated the learning of manifold-to-manifold transformation as an optimization problem on the Grassmann manifold.
Extensive evaluations have shown that the competitiveness and efficiency of our proposed approach compared with the state-of-the-art methods on three challenging datasets. In the future, we intend to extend our probability distribution-based approach to the unsupervised scenario.

REFERENCES

[1] Y. Pang, Y. Yuan, and X. Li, “Gabor-based region covariance matrices for face recognition,” IEEE Trans. Circuits Syst. Video Technol., vol. 18, no. 7, pp. 989–993, Jul. 2008.

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

[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 Computer Vision—ECCV. Berlin, Germany: Springer, 2012, pp. 216–229.

[4] 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. ICAL, vol. 13, 2013, pp. 2466–2472.

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

[6] A. Sanin, C. Sanderson, M. T. Harandi, and B. C. Lovell, “Spatio-temporal covariance descriptors for action and gesture recognition,” in Proc. IEEE Workshop Appl. Comput. Vis. (WACV), Jan. 2013, pp. 103–110.

[7] 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.

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

[9] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, “Geometric means in symmetric positive semi-definite matrices,” SIAM J. Matrix Anal. Appl., vol. 29, no. 1, pp. 328–347, Jan. 2007.

[10] Y. Xie, B. C. Vemuri, and J. Ho, “Statistical analysis of tensor fields,” in Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent. Berlin, Germany: Springer, 2010, pp. 682–689.

[11] 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. 149–152.

[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. Jayasumana, R. Hartley, M. Salzmann, H. Li, and M. 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.

[14] A. Cherian, S. Sra, A. Banerjee, and N. Papanikolopoulos, “Jensen-Bregman LogDet divergence with application to efficient similarity search for covariance matrices,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 9, pp. 2161–2174, Sep. 2013.

[15] D. Tosato, M. Farenzena, M. Spera, V. Murino, and M. Cristani, “Multiclass classification on Riemannian manifolds for video surveillance,” in Proc. Eur. Conf. Comput. Vis. Berlin, Germany: Springer, 2010, pp. 378–391.

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

[17] 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. Cham, Switzerland: Springer, 2014, pp. 17–32.

[18] 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. ICML, 2015, pp. 720–729.

[19] M. Harandi, M. Salzmann, and R. Hartley, “Dimensionality reduction on SPD manifolds: The emergence of geometry-aware methods,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 1, pp. 48–62, Jan. 2018.

[20] H. Li, T. Jiang, and K. Zhang, “Efficient and robust feature extraction by maximum margin criterion,” IEEE Trans. Neural Netw., vol. 17, no. 1, pp. 157–165, Jan. 2006.

[21] A. Ozakin and A. G. Gray, “Submanifold density estimation,” in Proc. Adv. Neural Inf. Process. Syst., 2009, pp. 1375–1382.

[22] M. Harandi, M. Salzmann, and M. Baktashmotlagh, “Beyond gauss: Image-set matching on the Riemannian manifold of PDFs,” in Proc. IEEE Int. Conf. Comput. Vis., Dec. 2015, pp. 4112–4120.

[23] E. Chevallier, E. Kalunga, and J. Angulo, “Kernel density estimation on spaces of Gaussian distributions and symmetric positive definite matrices,” SIAM J. Imag. Sci., vol. 10, no. 1, pp. 191–215, Jan. 2017.

[24] P.-A. Absil, R. Mahony, and R. Sepulchre, Optimization Algorithms on Matrix Manifolds. Princeton, NJ, USA: Princeton Univ. Press, 2009.

[25] M. Faraki, M. T. Harandi, and F. Porikli, “Image set classification by symmetric positive semi-definite matrices,” in Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV), Mar. 2016, pp. 1–8.

[26] J. Lu, G. Wang, W. Deng, P. Moulin, and J. Zhou, “Multi-manifold deep metric learning for image set classification,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2015, pp. 1137–1145.

[27] B. Leibe and B. Schiele, “Analyzing appearance and contour based methods for object categorization,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2, Jun. 2003, p. I-409.

[28] Z. Liao, J. Rock, Y. Wang, and D. Forsyth, “Non-parametric filtering for geometric detail extraction and material representation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2013, pp. 963–970.

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

[30] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley, “Face tracking and recognition with visual constraints in real-world videos,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2008, pp. 1–8.

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

[32] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2016, pp. 770–778.

[33] J. Deng, W. Dong, R. Socher, L.-J. Li, L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2009, pp. 248–255.

JIEYI REN received the B.E. and M.Eng degrees from Southwest Jiaotong University, China, in 2009 and 2012, respectively, and the Ph.D. degree with the Jiangsu Provincial Engineering Laboratory of Pattern Recognition and Computational Intelligence, Jiangnan University. His research interests include image classification and manifold learning.

XIAO-JUN WU (Member, IEEE) received the B.S. degree in mathematics from Nanjing Normal University, Nanjing, China, in 1991, and the M.S. and Ph.D. degrees in pattern recognition and intelligent system from the Nanjing University of Science and Technology, Nanjing, in 1996 and 2002, respectively. From 1996 to 2006, he taught at the School of Electronics and Information, Jiangsu University of Science and Technology, where he was promoted to a Professor. He has been with the School of Information Engineering, Jiangnan University, since 2006, where he is a Professor of computer science and technology. He was a Visiting Researcher with the Centre for Vision, Speech, and Signal Processing, University of Surrey, UK, from 2003 to 2004. He has published more than 200 articles. His current research interests include pattern recognition, computer vision, and computational intelligence. He was a Fellow of the International Institute for Software Technology, United Nations University, from 1999 to 2000. He was a recipient of the Most Outstanding Postgraduate Award from the Nanjing University of Science and Technology.