This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2392472, IEEE Transactions on Circuits and Systems for Video Technology

Discriminative Analysis for Symmetric Positive Definite Matrices on Lie Groups

Chunyan Xu, Canyi Lu, Junbin Gao, Wei Zheng, Tianjiang Wang, Shuicheng Yan

***Abstract*—In this paper, we study Discriminative Analysis for symmetric positive definite (SPD) matrices on Lie Groups, namely transforming a Lie Group (LG) into a dimension-reduced one by optimizing data separability. Particularly, we take the space of SPD matrices, e.g. covariance matrices, as a concrete example of Lie Groups, which has proved to be a powerful tool for high-order image feature representation. The discriminative transformation of a Lie Group is achieved by optimizing the within-class compactness as well as the between-class separability based on the popular graph embedding framework [1]. A new kernel based on the geodesic distance between two samples in the dimension-reduced Lie group is then defined, and fed into classical kernel-based classifiers, e.g. Support Vector Machine, for various visual classification tasks. Extensive experiments on five public datasets, i.e., Scene-15, Caltech101, UIUC-Sport, MIT-Indoor and VOC07, well demonstrate the effectiveness of discriminative analysis for SPD matrices on Lie groups, and state-of-the-art performances are reported.**

***Index Terms*—Discriminative analysis, Lie group, graph embedding, visual classification.**

I. Introduction

The past few decades have witnessed significant growth in the utilization of structured data in various computer vision and machine learning applications [2], [3], [4], where richer representations of data, such as matrices, tensors or graphs, are utilized instead of typical vector spaces. One of the interesting data structures that have gained much attention recently is Symmetric Positive Definite (SPD) matrices, the space of which is known to form a Lie group [5], [6], [7]. A Lie group is a group with the structure of a differentiable manifold such that the group operations, multiplication and inverse, are differentiable maps. The SPD matrices provide a powerful platform for analyzing visual signals, such as images and videos.

Due to their great importances in computer vision literature, SPD matrices have been much researched. A popular kind of SPD image descriptors, namely region covariance [8], is a powerful tool for encoding second-order image features, and has been applied to object detection and texture classification. To address the semantic segmentation problem, Carreira *et al.* [9] took advantage of the manifold structure of SPD matrices to analyze a certain kind of second-order statistics of image features.

The SPD matrices are mapped to a high dimension Hilbert space by kernels, and applied to pedestrian detection, image segmentation, and object categorization [6]. In order to employ the geometric structure of SPD matrices, Li *et al.* [7] also proposed a kernel based method for sparse representation and dictionary learning on Lie groups, and applied it to scene categorization, texture classification and face recognition.

Although SPD matrices have been proved to be a powerful tool for visual feature representation, successful applications still suffer from two limitations: 1) without considering the local structure of data on Lie groups, the discrimination may be adversely affected by some outliers and multi-modal classes may adversely affect the discrimination; 2) due to the redundancy of the manifold-valued data, its discriminative power may also be limited. To overcome the above limitation, a possible way is to perform discriminative analysis to reduce the dimension of the SPD matrices as in the Euclidean subspace.

To address these issues, we propose a novel method in this work, called discriminative analysis for SPD matrices on Lie groups, namely transforming a Lie group to a dimension-reduced one by optimizing data separability. In particular, we learn a discriminative transformation between two Lie groups based on the popular graph embedding framework [1]. First, two graphs are defined, i.e. an intrinsic graph and a penalty graph. The intrinsic graph characterizes the within-class compactness and connects points of the same class, while the penalty graph characterizes the between-class separability and connects the marginal points of different classes. Then a discriminative transformation is learned by enhancing the within-class compactness as well as maximizing the between-class separability. As a by-product, our proposed algorithm also reduces the dimensionality of SPD matrices, which will reduce the cost of model training and test in pattern analysis. Finally, a new kernel based on the geodesic distance between two points in the dimension-reduced Lie group is then defined and fed into classical kernel-based classifiers, e.g. Support Vector Machine (SVM), for various visual classification tasks. Taking the UIUC-Sport event classification as an example, Fig. 1 illustrates the framework of the discriminative analysis for SPD matrices on Lie groups.

The remainder of this paper is organized as follows. In Section II, we will review some related work. Section III introduces the background knowledge about Lie group. Section

Corresponding author: Tianjiang Wang, tjwang@hust.edu.cn.

Email address: (xuchunyan01, canyilu) @gmail.com (C. Xu, C. Lu), jbgao@csu.edu.au (J. Gao), w0209.zheng@samsung.com (W. Zheng) and eleyans@nus.edu.sg (S. Yan).

C. Xu and T. Wang are with School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, Hubei, 430074, P.R.China. C. Xu is also a visiting student in National University of Singapore.

S. Yan and C. Lu are with Department of Electrical and Computer Engineering, National University of Singapore, 117583, Singapore.

J. Gao is with School of Computing and Mathematics, Charles Sturt University, Bathurst, NSW 2795, Australia.

W. Zheng is with Beijing Samsung Telecom R&D Center, China.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2392472, IEEE Transactions on Circuits and Systems for Video Technology

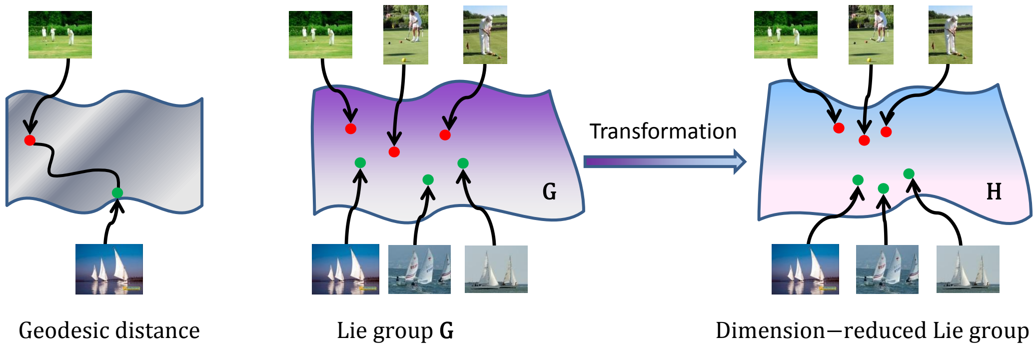


Fig. 1. An illustration of the discriminative analysis for SPD matrices on Lie groups. Symmetric positive definite (SPD) matrices are embedded as points on the Lie group, and the geometric distance between two points can be computed. Our proposed algorithm transforms the data points from a Lie Group $\mathbf{G}$ into a dimension-reduced one $\mathbf{H}$ by optimizing data separability. Specifically, the connected points with the same label stay as close together as possible, while connected points sharing different labels stay as far away as possible. Example images are from the UIUC-Sport database [10].

IV describes the discriminative analysis on Lie groups and also presents how to solve the induced optimization problem. In Section V, we apply our approach to visual classification, followed by experimental results and algorithm analysis in Section VI. The main findings and possible future research are summarized in Section VII.

II. Related work

We recall some standard similarity meatures for SPD matrices. The distance or similarity measure over the space of SPD matrices is crucial for discriminative analysis. A simple method is to represent $n \times n$ SPD matrices as vectors in $\mathbb{R}^{\frac{d\times(d+1)}{2}}$, and then the distance in the Euclidean space can be used [9]. However, the space of SPD matrices does not conform with the Euclidean geometry, but forms a Lie group which is a differentiable manifold. Vectorizing the covariances into the Euclidean space ignores the manifold structure which leads to poor performance [11].

A better choice is to take the curvature of the Riemannian manifold into account and use the corresponding geodesic length along the manifold surface as the distance. Several different similarity measures or metrics of SPD matrices have been proposed. For X_i, X_j in S_n^+, the Affine Invariant Riemannian Metric (AIRM) [12] is defined: $D_{AIRM} = ||\log(X_i^{-1/2} X_j X_i^{-1/2})||_F$. This metric enjoys several useful theoretical properties, but its computational complexity for lager matrices causes significant slowdowns. The Jensen-Bregman LogDet Divergence (JBLD) [13] is defined as $D_{JBLD} = \log \mid \frac{X_i+X_j}{2} \mid - \frac{1}{2}\log \mid X_i X_j \mid$, where $\mid \cdot \mid$ denotes the determinant. Although the computational speed of JBLD metric is fast, the structure of the manifold space may not be preserved well. Wang *et al.* [14] proposed the J-divergence as the distance measure for SPD matrices, $D_{Jdiv}(X_i, X_j) = \sqrt{\frac{1}{2}\mathrm{tr}(X_i^{-1}X_j + X_j^{-1}X_i) - n)}$, where $\mathrm{tr}(\cdot)$ is the matrix trace operator, and n is the size of square matrix X_i and X_j. And the symmetrized KL-Divergence Metric (KLDM) [2] is defined, $D_{\mathrm{KLDM}}(X_i, X_j) = \frac{1}{2}\mathrm{tr}(X_i^{-1}X_j + X_j^{-1}X_i - 2I)$. The J-divergence [14] and KLDM metric [2] require the inversion of the SPD matrices, which can be slow, and may lead to poor accuracy with overestimating the Riemannian metric.

The Log-Euclidean Riemannian Metric (LERM) is defined as follows,

$$D_{\mathrm{LE}}(X_i, X_j) = ||\log(X_i) - \log(X_j)||_F, \tag{1}$$

where $X_i, X_j \in \mathcal{S}_n^+$, $\mathcal{S}_n^+$ denotes the set of SPD matrices with size $n \times n$, $\log(\cdot)$ is the principal matrix logarithm, and $||\cdot||_F$ denotes the Frobenius norm of a matrix. The Log-Euclidean mapping $\log(X)$ maps the SPD matrix to a flat Riemannian space and thus the Euclidean distances can be used in LERM, which are widely used due to their easy computation and several important properties, e.g. invariance to inversion and similarity transforms. For the pedestrian detection problem, Tuzel *et al.* [15] utilized SPD matrices as object descriptors and implemented the boosting method on the Riemannian manifold. Vemulapalli *et al.* [16] performed classification by mapping SPD matrices from Riemannian manifolds to Euclidean spaces using the kernel learning approach. We also used the Log-Euclidean Riemannian Metric in this work.

The conventional discriminative analysis methods, e.g. Linear Discriminant Analysis (LDA) [17] and Marginal Fisher analysis (MFA) [1], do not take into account the Lie group geometric structure of the data, which may result in the loss of important discriminative information. Moreover, it is difficult to carry out the discriminative analysis on Lie groups as the conventional vector-based discriminative analysis methods.

There have also been considerable research efforts [18] devoted to discriminative analysis for Riemannian manifolds such as the Grassmann manifold. Hamm *et al.* [19] performed the discriminant analysis by using kernel linear discriminant analysis (LDA) with the Grassmann kernels, but only focused on manifold kernel LDA to classification problems. Harandi *et al.* [20] also proposed a graph embedding discriminative analysis on the Grassmann manifold, which maps the manifold space into reproduced kernel Hilbert spaces.

Recently, Li *et al.* [21] embedded the space of Gaussian into the space of SPD matrices, which are analyzed from the Lie group manifold point of view. In order to get a discriminative distance, they attempted to find a linear transformation in the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2392472, IEEE Transactions on Circuits and Systems for Video Technology

logarithm domain, which is defined as

$$\begin{aligned} D(X_i, X_j) &= \text{tr}((\log(X_i) - \log(X_j))^T M(\log(X_i) - \log(X_j))) \\ &= ||A(\log(X_i) - \log(X_j))||_F \end{aligned} \tag{2}$$

where X_i, X_j and M are three SPD matrices with size $n \times n$, and $M = A^T A$. Wang *et al.* [22] proposed a discriminative learning approach by modeling the image set with SPD matrices, and also explored the Log-Euclidean distance metric for SPD matrices. With this metric function, they can map the SPD matrix from the Riemannian manifold to a Euclidean space, and then some learning methods (such as linear discriminant analysis and partial least squares) with respect to the vector space can be exploited in linear/kernel formulation.

From the perspective of distance metric, our proposed method is also based on the same Log-Euclidean Riemannian Metric (LERM) with [21], and can also be seen as a kind of metric learning method on the Lie group manifold space. However, these manifold discriminative analysis methods, which are based on a metric learning view of the problem, 1) do not tackle subspace learning between the manifold spaces, but transform the manifold space to the vector space; 2) and they also suffer from the redundancy of the manifold-valued data, namely not transforming a Lie group to a dimension-reduced one.

III. Background Knowledge of Lie Group

A. Lie Groups and Lie Algebra

A Lie group $\mathbf{G}$ is a smooth manifold with a group structure, in which the group operations of multiplication and inversion are smooth maps [23]. In particular, the group is characterized by a unique identity element $I \in \mathbf{G}$ and two group operations

$$\text{multiplication } g_1 g_2 : \mathbf{G} \times \mathbf{G} \to \mathbf{G}, \quad \text{inversion } g^{-1} : \mathbf{G} \to \mathbf{G}, \tag{3}$$

which are smooth mappings.

The tangent space of the Lie group $\mathbf{G}$ to its identity element I forms a Lie algebra $\mathfrak{g}$. We can map between the Lie group and its tangent space from the identity element I using the exp and log maps,

$$\bar{X} = \log(X), \quad X = \exp(\bar{X}), \tag{4}$$

where $X \in \mathbf{G}$ and $\bar{X} \in \mathfrak{g}$ are elements of Lie group and Lie algebra, respectively. In this paper, we only focus on matrix Lie groups. The exponential and logarithm maps of a matrix are given by

$$\log(X) = \sum_{i=1}^{\infty} \frac{(-1)^{i-1}}{i}(X - I)^i, \quad \exp(\bar{X}) = \sum_{i=0}^{\infty} \frac{1}{i!}\bar{X}^i. \tag{5}$$

Let $\mathbf{G}$ and $\mathbf{H}$ be Lie groups with their corresponding Lie algebras $\mathfrak{g}$ and $\mathfrak{h}$. A transformation $\phi : \mathbf{G} \to \mathbf{H}$ from a Lie group $\mathbf{G}$ to $\mathbf{H}$ is called a smooth map if it is a group homomorphism [24]. That is to say, the Lie group homomorphism is a map between Lie groups $\phi : \mathbf{G} \to \mathbf{H}$, which is both a group homomorphism and a smooth map. ϕ_* is a map between the corresponding Lie algebras $\mathfrak{g}$ and $\mathfrak{h}$: $\phi_* : \mathfrak{g} \to \mathfrak{h}$. The maps ϕ and ϕ_* are related by the exponential maps. For any $\bar{X} \in \mathfrak{g}$, we have

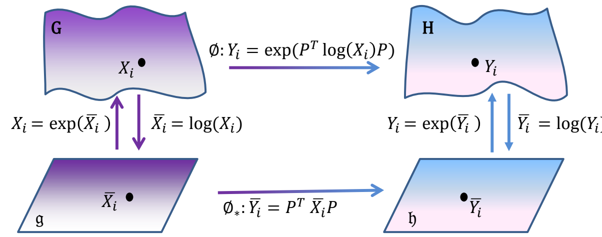


Fig. 2. An illustration of the transformation between Lie groups. The tangent spaces of Lie groups $\mathbf{G}$ and $\mathbf{H}$ to the identity element I form Lie algebras $\mathfrak{g}$ and $\mathfrak{h}$, respectively. The Lie group and the corresponding Lie algebra space can be mapped using the exp and log maps. The map between Lie algebras ($\mathfrak{g} \to \mathfrak{h}$) is denoted by ϕ_*: $\bar{Y}_i = P^T \bar{X}_i P$. Then we can get the transformation from one Lie group $\mathbf{G}$ to another Lie group $\mathbf{H}$, ϕ: $Y_i = \exp(P^T \log(X_i)P)$.

$$\phi(\exp(\bar{X})) = \exp(\phi_*(\bar{X})). \tag{6}$$

B. $\mathcal{S}_n^+$ as a Lie Group

$\mathcal{S}_n^+$ is a Lie group with the identity element being the identity matrix I and its inverse operation follows the regular matrix inversion. In the Log-Euclidean framework, the logarithmic multiplication $\odot : \mathcal{S}_n^+ \times \mathcal{S}_n^+ \mapsto \mathcal{S}_n^+$ and the inversion are defined as [5]:

$$\begin{aligned} X_i \odot X_j &:= \exp(\log(X_i) + \log(X_j)), \\ X_i^{-1} &:= \exp(-\log(X_i)). \end{aligned} \tag{7}$$

It can be seen that the multiplication operation and inverse operation are smooth mappings in the Log-Euclidean framework. Therefore, the space of $\mathcal{S}_n^+$ forms a Lie group.

The distance of two points on a Lie group can be measured by the shortest length of the curve connecting them. The minimum length curve between two points is called the geodesic. With the above logarithm map and the group operation, the geodesic distance [23], [5] between two elements on a Lie group can be computed by

$$D_{\text{LE}}(X_i, X_j) = ||\log(X_i) - \log(X_j)||_F, \tag{8}$$

where $X_i, X_j \in \mathcal{S}_n^+$ and $||\cdot||_F$ denotes the Frobenius norm of a matrix.

IV. Discriminative Analysis for SPD Matrices

In this section, we introduce discriminative analysis for SPD matrices on Lie groups, namely transforming a Lie group $\mathbf{G}$ into a dimension-reduced one $\mathbf{H}$ by optimizing data separability. Under the popular graph embedding framework [1], we learn the discriminative transformation between Lie groups, by introducing intrinsic and penalty graphs to respectively characterize within-class compactness and between-class separability.

A. Transformation between Lie Groups

Let $X_i \in \mathcal{S}_m^+$ and $Y_i \in \mathcal{S}_n^+$ (usually $n < m$) be two points on two Lie groups $\mathbf{G}$ and $\mathbf{H}$, respectively. According to the Lie group homomorphism theory [24], we introduce

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2392472, IEEE Transactions on Circuits and Systems for Video Technology

a transformation from a Lie group $\mathbf{G}$ to another Lie group $\mathbf{H}$, i.e., $\phi : X_i \rightarrow Y_i$. The tangent spaces of Lie groups $\mathbf{G}$ and $\mathbf{H}$ to the identity element I form Lie algebras $\mathfrak{g}$ and $\mathfrak{h}$, respectively. The Lie group and the corresponding Lie algebra space can be mapped using the exp and log map, i.e., $\mathbf{G} = \exp(\mathfrak{g})$ and $\mathfrak{g} = \log(\mathbf{G})$.

In the Lie algebraical space, $\bar{X}_i \in \mathfrak{h}$, $\bar{Y}_i \in \mathfrak{g}$ are respectively two points corresponding to $X_i \in \mathbf{G}$ and $Y_i \in \mathbf{H}$ by the exp and log maps, i.e.

$$\begin{aligned} \bar{X}_i &= \log(X_i), \qquad X_i = \exp(\bar{X}_i), \\ \bar{Y}_i &= \log(Y_i), \qquad Y_i = \exp(\bar{Y}_i). \end{aligned} \tag{9}$$

It can be seen from (9) that $\bar{X}_i$ is symmetric and with the same size as X_i. $\bar{Y}_i$ is also symmetric and with the same size as Y_i. Instead of defining the transformation $\phi : X_i \rightarrow Y_i$ between the Lie groups $\mathbf{G}$ and $\mathbf{H}$ directly, we define the transformation $\phi_* : \bar{X}_i \rightarrow \bar{Y}_i$ between their corresponding Lie algebras $\mathfrak{h}$ and $\mathfrak{g}$ as follows

$$\phi_* : \quad \bar{Y}_i = P^T \bar{X}_i P, \tag{10}$$

where $P \in \mathbb{R}^{m\times n}$. Multiplying P^T and P on both sides of $\bar{X}_i$ preserves the symmetric structure of the data.

Based on the mapping (9) between the Lie groups and their corresponding Lie algebras using the exp and log map, the corresponding transformation from the Lie group $\mathbf{G}$ to $\mathbf{H}$ is:

$$\begin{aligned} \phi : \quad Y_i &= \exp(\bar{Y}_i) \\ &= \exp(P^T \bar{X}_i P) \\ &= \exp(P^T \log(X_i) P). \end{aligned} \tag{11}$$

According to the above Lie group transformation ϕ, all such points X_i on the Lie group $\mathbf{G}$ can be mapped to other points Y_i in the set $\mathbf{H}$. It can be proved that $\mathbf{H}$ also forms a Lie group by the definition directly. Actually, the multiplication and inversion operations of the elements (Y_i, $Y_j \in \mathbf{H}$) are:

$$\begin{aligned} Y_i \odot Y_j :&= \exp(\log(Y_i) + \log(Y_j)) \\ &= \exp(P^T \log(X_i) P + P^T \log(X_j) P) \\ &= \exp(P^T (\log(X_i) + \log(X_j)) P), \\ Y_i^{-1} :&= \exp(-\log(Y_i)) \\ &= \exp(-P^T \log(X_i) P). \end{aligned} \tag{12}$$

Since X_i and X_j are SPD matrices, it is easy to be verified that both $Y_i \odot Y_j$ and Y_i^{-1} are SPD matrices. By the definition of the Lie group, all the projected points Y_i form a Lie group $\mathbf{H}$. Fig. 2 illustrates how to construct the transformation between Lie groups.

B. Discriminative Learning for SPD matrices on Lie Groups

In this subsection, we present the discriminative analysis for SPD matrices on Lie groups to improve the discriminative power of the data. Specifically, we aim to learn a discriminative transformation from a Lie group into a dimension-reduced one by optimizing data separability. The discriminative transformation of the Lie Group is achieved by enhancing the within-class compactness as well as maximizing the between-class separability based on the popular graph embedding framework in [1].

Based on the Marginal Fisher Analysis (MFA) method [1], we design the intrinsic graph and the penalty graph for our proposed algorithm. The intrinsic graph W^w characterizes the within-class compactness and connects each data point with its neighboring points of the same class, while the penalty graph W^b characterizes the between-class separability and connects the marginal point pairs of different classes.

Suppose we are given N labeled points $\{X_i, l_i\}_{i=1}^N$ from the underlying Lie group $\mathbf{G}$, where $X_i \in \mathcal{S}_n^+$ and $l_i \in \{1, 2, .., C\}$ with C denoting the number of classes. The local space structure of the Lie group can be modeled by building the intrinsic graph W^w and the penalty graph W^b. Based on the within-class compactness and the between-class separability, W^w and W^b are respectively defined by:

$$\begin{aligned} W_{ij}^w &= \begin{cases} 1, \text{if } X_i \in N_{k_1}^+(X_j) \text{ or } X_j \in N_{k_1}^+(X_i) \\ 0, \text{otherwise}, \end{cases} \\ W_{ij}^b &= \begin{cases} 1, \text{if } (X_i, X_j) \in P_{k_2}(c_i) \text{ or } (X_i, X_j) \in P_{k_2}(c_j) \\ 0, \text{otherwise}. \end{cases} \end{aligned} \tag{13}$$

Here, $N_{k_1}^+(X_i)$ indicates the index set of the k_1 nearest neighbors of the sample X_i in the same class, π_c denotes the index set of samples belonging to the cth class, $P_{k_2}(c)$ is the set of the k_2 nearest data pairs among the set $\{(X_i, X_j), X_i \in \pi_c, X_j \notin \pi_c\}$, and the nearest neighbors of the samples are computed by the distance metric in Eqn. (8).

To further improve the discriminative power on Lie groups and preserve the geometrical structure of the data, we perform discriminative analysis by simultaneously characterizing the within-class compactness and the between-class separability. In other words, the connected points of W^w stay as close together as possible, while connected points of W^b stay as distant as possible. Then we can describe the above analysis by optimizing the following two objective functions:

$$\begin{aligned} \min_P f_1 &= \sum_{i,j} D_{LE}(Y_i, Y_j)^2 W_{ij}^w \\ &= \sum_{i,j} ||\log(Y_i) - \log(Y_j)||_F^2 W_{ij}^w \end{aligned} \tag{14}$$

$$\begin{aligned} \max_P f_2 &= \sum_{i,j} D_{LE}(Y_i, Y_j)^2 W_{ij}^b \\ &= \sum_{i,j} ||\log(Y_i) - \log(Y_j)||_F^2 W_{ij}^b, \end{aligned} \tag{15}$$

where $Y_i = \exp(P^T \log(X_i) P)$ and $\log(Y_i) = P^T \log(X_i) P$. Eqn. (14) punishes the neighbours in the same class if they are mapped far away on the new Lie group $\mathbf{H}$, while Eqn. (15) punishes the points of different classes if they are mapped close together on the new Lie group $\mathbf{H}$. By converting both problems into minimization, the overall optimization problem is [1]

$$P^* = \arg\min_P (f_1 - f_2). \tag{16}$$

The whole procedure of our proposed algorithm is outlined in Algorithm 1.

[1]One may use other objective functions to learn the transformation, e.g. $\min \frac{f_1}{f_2}$. For the convenience of the optimization, we simply use (16).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2392472, IEEE Transactions on Circuits and Systems for Video Technology

Algorithm 1 Discriminative analysis for SPD matrices on Lie groups

Input: Training set $\{X_i, l_i\}_{i=1}^N$ from underlying Lie group $\mathbf{G}$, where $X_i \in \mathcal{S}_n^+$, and $l_i \in \{1, 2, ..., C\}$, with C denoting the number of classes.

1. Construct intrinsic graph W^w and penalty graph W^b with Eqn. (13).
2. Solve problem (16) for learning a discriminative transformation.
3. Transform all points from a Lie group $\mathbf{G}$ to another dimension-reduced Lie group $\mathbf{H}$ by Eqn. (11).

Differentiating $f_1 - f_2$ with respect to the transformation matrix P yields a gradient rule which will be used for optimization:

$$\frac{\partial(f_1 - f_2)}{\partial P} = 8\sum_{i,j}(\log(X_i)PP^T\log(X_i)P - \log(X_i)PP^T\log(X_j)P)(W_{ij}^w - W_{ij}^b). \quad (17)$$

In minimizing the criterion in (16), we can calculate (17) and update the transformation matrix P by a conjugate gradients optimizer (like the neighbourhood components analysis (NCA) method [25]), i.e.

$$P_{t+1} = P_t - \epsilon\frac{\partial(f_1 - f_2)}{\partial P_t}, \quad (18)$$

where ϵ is the step-size in the gradient descent. Furthermore, by restricting P to be a non-square matrix of $m \times n$ $(n < m)$, the data dimension is reduced after the transformation by the discriminative analysis for SPD matrices. Furthermore, to optimize the transformation matrix P in Eqn. (16), we employ a conjugate gradient optimizer which is a standard optimization method [26], and as a result, the transformation matrix P can only obtain a local minimum in the sense of (16), which will again be shown in the experimental parts.

We finally perform a rigorous theoretical complexity analysis of the proposed algorithm. For the per-iteration with Eqn. (17), the computational complexity is about $\mathcal{O}(N^2m^2n)$, where N is the number of image samples in the training set, m and n are the size of the square matrix X_i and Y_i respectively. For mapping each SPD matrix from Lie group manifold to the Euclidean space, the complexity of computing $\log(X_i)$ is $\mathcal{O}(n^3)$, where n is the size of a square matrix X_i. Therefore, for computing similarity matrix with Eqn. (8), the computational complexity is $\mathcal{O}(Nn^3)$.

V. Application for Visual Classification

To evaluate our proposed algorithm, we apply it for visual classification, and introduce how to extract SPD descriptors for visual images. Recently, Carreira *et al.* [9] mapped SPD local descriptors to the tangent space using the theory of Log-Euclidean metrics, but they just constructed the feature vector from the upper triangle of $\log(X_i)$, and then obtained the distance by the inner product between feature vectors. Inspired by the second-order feature pooling algorithm [9], we utilize this SPD descriptor for the visual classification problem.

We use the second-order image feature pooling algorithm [9] to extract the second-order feature with a spatial pyramid scheme [27]. For an image i, the SPD descriptor of an image region R_k can be defined as:

$$X_{ik} = \frac{1}{|\mathrm{f}_{R_k}|}\sum_{o:(\mathrm{f}_o \in R_k)} \mathrm{f}_o\mathrm{f}_o^T \quad (19)$$

where $\mathrm{f}_o \in \mathbb{R}^m$ are all descriptors of an image i, $\mathrm{f}_o \in R_k$ and $|\mathrm{f}_{R_k}|$ denote the descriptors and the corresponding number in the image region R_k, respectively.

A weighted sum of the distance between two images I_i and I_j is:

$$D_{LE}(I_i, I_j) = \sqrt{\sum_{k=1}^{K} w_k(D_{LE}(X_{i,k}, X_{j,k}))^2}, \quad (20)$$

where K is the total number of image regions, $D_{LE}(X_{i,k}, X_{j,k})$ is the distance between the respective tth image region of I_i and I_j, and w_k is the weight of the kth region.

The above SPD image descriptors on the Lie group $\mathbf{G}$ can be transformed to another Lie group $\mathbf{H}$ by the our proposed algorithm. Then a kernel based on the geodesic distance between two samples in the dimension-reduced Lie group is defined as follows:

$$K_{LE}(\mathrm{I}_i, \mathrm{I}_j) = \exp(-\gamma(D_{\mathrm{LE}}(I_i, I_j))), \quad (21)$$

where the parameter γ is directly related to scaling. It can be easily proved, the same as in [28], that the newly defined Lie group kernel is a valid Mercer's kernel. The Lie group kernel can be employed in classification methods such as Nearest Neighbour or Support Vector Machines.

VI. Experiments

In this section we evaluate our proposed algorithm by comparing it with several existing state-of-the-arts. We first introduce the experimental setups, and then report and analyze the experimental results, after which a further discussion about the effectiveness of the proposed algorithm is given.

A. Experimental Setups

The experiments are conducted on four commonly used datasets: Scene15 [27], Caltech101 [29], UIUC-Sport [10], MIT-Indoor [30] and PASCAL VOC2007 [31].

The Scene15 [27] database consists of 4,485 images with 15 categories, each category containing 200 to 400 images. Following the same experimental setting as in [27], we take 100 images per category for training and the rest images are used for test, and report the averaged classification accuracies over 10 trials.

The Caltech101 [29] contains 8,677 images in total, with 102 categories (including one background category). Following the experimental protocol stated by the designers of this dataset, we randomly choose 15 (for the first round), 30 (for the second round) images per category for training, and use the rest images for test. Then we conduct the experiment with this random split for 10 times and report the average classification accuracy over these 10 trials for comparison.

The UIUC-Sport dataset [10] has 8 complex event classes. Following the sample experiment setting used in [32], [33],

TABLE V
THE MEAN AVERAGE PRECISIONS ON PASCAL VOC 2007 DATABASE. PLEASE SEE THE TEXT FOR DETAILS ABOUT EACH METHOD.

Method	mAP (in %)
IFK (SIFT) [44]	58.3
Best of VOC07 [31]	59.4
BoF+HOG [36]	59.82
FLH [38]	60.4
IFK (SIFT+Color) [44]	61.7
FLH+BOW [38]	62.8
FK+VC [35]	62.9
LG	61.83
Ours	**63.37**

70 images from each class are randomly sampled for training and 60 images are sampled for test. We run the experiment for 10 trials, and report the average classification accuracy.

The MIT-Indoor dataset [30] consists of 67 clustered indoor scene categories, and we adopt the fixed training/testing splits as in [30].

We also use PASCAL VOC2007 dataset [31] to analyze the performances of the proposed method from various aspects. The dataset contains objects of 20 categories and it poses a challenging task of object recognition due to significant variations in terms of appearances and poses even with occlusions. There are 5,011 training images and 4,952 test images. The performance is evaluated by the standard PASCAL protocol which computes average precision (AP) based on the precision/recall curve; we report the mean Average Precision (mAP) across the 20 categories.

All experiments are conducted based on the following experimental setups:

- To construct the SPD image features (with no dictionary learning), we extract the 128-dimensional SIFT descriptors, as well as the additional 17-dimensional features including RGB color values, location, gradient, and Harris features, via the VLFeat library [45]. Thus the size of the SPD descriptor $X_i \in \mathcal{S}_m^+$ is 145×145, i.e. $m = 145$.
- To construct the SPD descriptors, the image is divided into 1×1, 2×2 and 4×4 grids, so that totally 21 spatial pooling regions are obtained, assigning the same weight w at the same layer.
- We empirically set the parameters k_1 and k_2 of the intrinsic graph W_w and the penalty graph W_b in all visual classification experiments, as described in [1]. Specifically, we sample five values $\{2, 3, 5, 7, 9\}$ of k_1 and choose the value with the best performance. We similarly choose the best k_2 in the set $\{20, 40, 60, 80\}$.
- A one-versus-all scheme is used to tackle the multi-class problem, and the SVM training and testing are performed using the libsvm software package [46]. The parameter γ of the LG kernel is set to 0.001 in our experiment.
- The dimension n of dimension-reduced data $Y_i \in \mathcal{S}_m^+$ is selected in the set n={145, 135, 125, 115, 105}. In our experiment, the best results are reported with $n = 125$.
- In order to evaluate the effectiveness of our proposed algorithm, we design a Lie group (LG) kernel method over the original data space without using the proposed algorithm for preprocessing, and we denote this method as LG (without discriminative analysis). And the dimension m of the original SPD data X_i is 145.
- The SPD matrices in the matrix log operation should meet some conditions. Following the same setting defined in [9], we also added a small constant on their diagonal (0.001 in all experiments) for numerical stability.

B. Performance Comparison

Table I shows the classification accuracy on the Scene15 database. It can be seen that the discriminative analysis for SPD matrices on Lie groups significantly outperforms the others including the Log-Euclidean kernel method with sparse representation and dictionary learning of SPD matrices [7], Spatial Pyramid Matching (SPM) [27], SPM on the semantic manifold [34], mid-level visual concepts [35], Discriminative part detectors learning method [37], etc. The LG and our proposed methods are both better than the other methods, and the discriminative analysis for SPD matrices on Lie groups performs the best when the dimension n of the transformed data $\mathcal{S}_+^n$ is 125. Furthermore, the proposed method improves the performance by 1.86% over the Lie group kernel method, which does not perform transformation between Lie groups.

The experimental results on the Caltech101 database are shown in Table II. We compare the proposed method with the exiting algorithms such as Low-rank sparse coding [32], Sparse embedding [41], Kernel sparse representation [42], etc. The results indicate that our algorithm is significantly better than the other methods. The classification accuracy of the proposed method is 1.6% and 2.28% higher than the result of the LG method for 15 and 30 training images, respectively.

Table III and IV show the comparison results on the UIUC-Sport and MIT-Indoor datasets, respectively. Our proposed method effectively works compared to the other methods (e.g. Kernel sparse representation [42], Discriminative part detectors learning method [37] and the LG method). The classification accuracies of UIUC-Sport and MIT-Indoor datasets are substantially improved from 88.4% and 52.3% (the best reported results [35]) to 90.90 and 55.57, respectively.

Finally, we compare in Table V the result of our proposed algorithm with some results in the literature [44], [31], [36], [38], [35] on the PASCAL VOC 2007 database. The best method during the PASCAL VOC 2007 competition (by INRIA) [31] reported 59.4% mAP with multiple channels and costly non-linear SVMs. Fernando *et al.* [38] obtained an mAP of 60.4% with the method of Frequent Local Histograms (FLH) alone, and got an mAP of 62.8% after combining FLH with bag-of-visual-words (FLH+BOW) of SIFT-128 and 5K visual word vocabulary. In [44], Improved Fisher Kernel (IFK) obtained two results 58.3% and 61.7% with SIFT features only and with SIFT and color information, respectively. T. Kobayashi [36] reported the 59.83 % mAP by combining Bag-of-Feature with Histogram of Oriented Gradients (BoF+HOG). The method combining the improved fisher kernel with our visual concepts (FK+VC) [35] got an mAP of 62.9%. The proposed method is comparable to some exiting methods, and thus we can say that the method effectively works for the problem of multi-label image classification.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2392472, IEEE Transactions on Circuits and Systems for Video Technology

TABLE I
PERFORMANCE COMPARISON ON THE SCENE15 DATABASE.

Method	Accuracy (%)
Li.P *et al.* [7]	80.92±0.44
Lazebnik *et al.* [27]	81.4±0.5
Kwitt *et al.* [34]	82.3
Li *et al.* [35]	85.4
Kobayashi *et al.* [36]	85.63±0.67
Sun *et al.* [37]	86.0±0.8
Fernando *et al.* [38]	86.2±0.4
Zheng *et al.* [39]	86.3
LG	88.02±0.47
Ours	**89.88±0.46**

TABLE II
PERFORMANCE COMPARISON ON THE CALTECH101 DATABASE.

Method	Accuracy (%)	
	15 tr.	30 tr.
McCann *et al.* [40]	66.1±1.1	71.9±0.6
Zhang *et al.* [32]	-	75.02±0.74
Sun *et al.* [37]	-	78.8±0.5
Nguyen *et al.* [41]	69.5	77.3
Goh *et al.* [42]	71.1±1.3	78.9±1.1
Duchenne *et al.* [43]	75.3±0.7	80.3±1.2
Feng *et al.* [33]	70.3	82.6
LG	75.83±0.7	81.41±0.9
Ours	**77.42±0.9**	**83.69±0.8**

TABLE III
PERFORMANCE COMPARISON ON THE UIUC-SPORT DATABASE.

Method	Accuracy (%)
LiJia *et al.* [10]	76.3
Kwitt *et al.* [34]	83.0
Sun *et al.* [37]	86.4±0.88
Zheng *et al.* [39]	87.2
Zhang *et al.* [32]	88.17±0.85
Li *et al.* [35]	88.4
LG	89.0±1.2
Ours	**90.91±0.9**

TABLE IV
PERFORMANCE COMPARISON ON THE MIT-INDOOR DATABASE.

Method	Accuracy (%)
Quattoni *et al.* [30]	26
LiJia *et al.* [10]	37.6
Kwitt *et al.* [34]	44.0
Zheng *et al.* [39]	47.2
Sun *et al.* [37]	51.4
Li *et al.* [35]	52.3
LG	53.46
Ours	**55.58**

C. Algorithm Analysis

Our proposed algorithm has a fast convergence of the iterations when learning a discriminative transformation in Eqn. (16). In Fig. 3, we show the relationship between the objective function values $f_1 - f_2$ and the number of iterations on the Scene15 database, namely how the objective function $f_1 - f_2$ changes with respect to the number of iterations on the Scene15 database.

We then analyze the sensitivity of our ptoposed algorithm to the different dimension n of the transformed SPD descriptors. As shown in Figure 4, different n values have different impact on the classification rates. Due to the redundancy of the manifold-valued data, the discriminative power is limited in the original space of the SPD data. It is noticed that the classification accuracy of the 125-dimensional data is better than the original space of the SPD data ($n = 145$), but the accuracy rate is reduced when the dimension n of the SPD data is 105. If the dimension of the SPD data is very low, the discriminative information may be not sufficient. Therefore, the discriminative power of the data is better only if the dimension of the SPD data is appropriate.

Here we also analyze the effectiveness of the proposed algorithm with respect to the within-class compactness and the between-class separability on the Scene15 database. We randomly select 10 images per category with scene labels, then order them according to their labels, and test our algorithm on this subset. The derived affinity matrices by the LG method and our proposed method are illustrated in Fig. 5. We can see that the discriminative analysis for SPD matrices on Lie groups obtains an affinity matrix which is closer to block diagonal by a discriminative transformation.

Fig. 6 presents the relationship between the discriminative power and the parameter γ on the Scene15 database. The

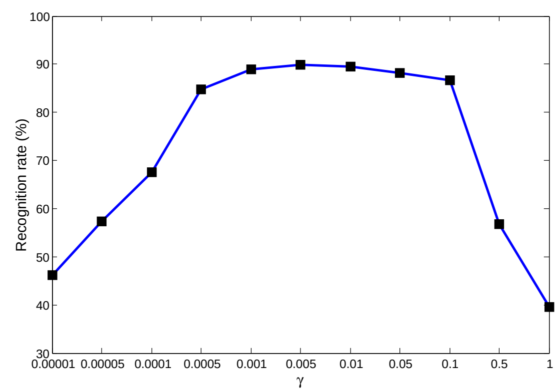


Fig. 6. Performance on the Scene15 database for various parameter γ values of the Lie group kernel.

recognition rate is robust as long as the value of the parameter γ falls in the range of approximately from 0.0005 to 0.1, however the Lie group kernel with a smaller value of the parameter γ from 0.1 to 1 can significantly deteriorate the recognition rates.

VII. CONCLUSIONS AND FUTURE WORK

In this work, we proposed discriminative analysis for SPD matrices on Lie groups by transforming a Lie group into a dimension-reduced one. Within the graph embedding framework, a discriminative transformation is learned by optimizing the data separability. This will reduce the cost of model training and testing in pattern analysis. Experimental results show that the proposed method achieve superior performances by comparing with state-of-the-art methods.

The main shortcoming of the proposed approach is costly computation time. Take the UIUC-Sport dataset as an example, the training stage takes around 8hrs, while the testing time is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2392472, IEEE Transactions on Circuits and Systems for Video Technology

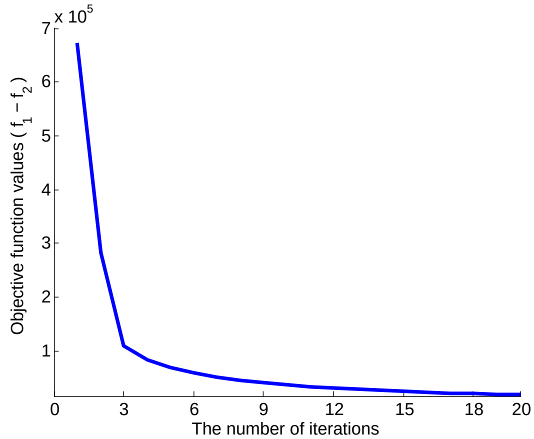


Fig. 3. The relationship between number of iterations and objective function values $f_1 - f_2$ on the Scene15 database.

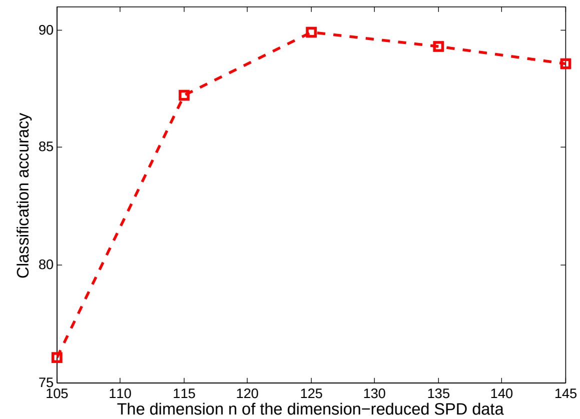
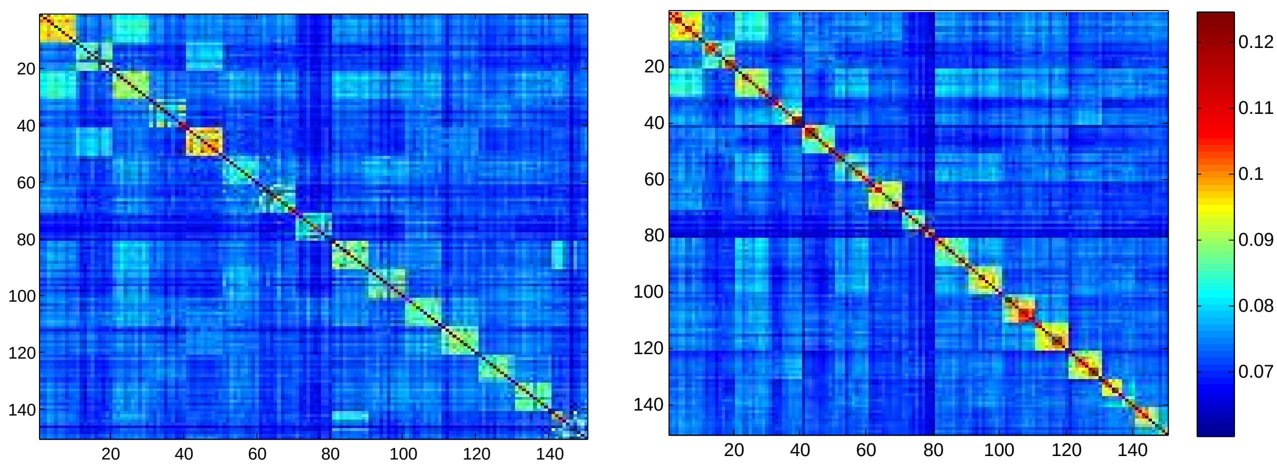


Fig. 4. The relationship between feature dimension and recognition rate on the Scene15 database.



(a) LG

(b) Ours

Fig. 5. The affinity matrices derived by (a) LG with no discriminative analysis, and (b) Our method on the Scene15 database.

about 1hr on an Intel Core 2 Quad processor with 2.83GHz CPU and 8.00GB RAM. Therefore, we shall study how to speed up our proposed manifold learning algorithm for SPD matrices in future.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (Grant 61073094 and U1233119). This research was partly supported under Australian Research Council Discovery Projects funding scheme (project DP140102270).

REFERENCES

[1] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, “Graph embedding and extensions: A general framework for dimensionality reduction,” *TPAMI*, vol. 29, no. 1, pp. 40–51, Jan 2007. 1, 2, 3, 4, 6

[2] M. Moakher and P. G. Batchelor, “Symmetric positive-definite matrices: From geometry to applications and visualization,” in *Visualization and Processing of Tensor Fields*, 2006, pp. 285–298. 1, 2

[3] Y. M. Lui, “Tangent bundles on special manifolds for action recognition,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 6, pp. 930–942, 2012. 1

[4] C.-Y. Lu, H. Min, Z.-Q. Zhao, L. Zhu, D.-S. Huang, and S. Yan, “Robust and efficient subspace segmentation via least squares regression.” in *ECCV*, vol. 7578, 2012, pp. 347–360. 1

[5] 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 Analysis Applications*, vol. 29, no. 1, pp. 328–347, 2006. 1, 3

[6] S. Jayasumana, R. Hartley, M. Salzmann, H. Li, and M. Harandi, “Kernel methods on the riemannian manifold of symmetric positive definite matrices,” in *CVPR*, 2013, pp. 73–80. 1

[7] P. Li, Q. Wang, W. Zuo, and L. Zhang, “Log-euclidean kernels for sparse representation and dictionary learning,” in *ICCV*, 2013, pp. 1601–1608. 1, 6, 7

[8] O. Tuzel, F. Porikli, and P. Meer, “Region covariance: A fast descriptor for detection and classification,” in *ECCV*, 2006, pp. 589–600. 1

[9] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu, “Semantic segmentation with second-order pooling,” in *ECCV*, 2012, pp. 430–443. 1, 2, 5, 6

[10] E. P. X. Li-Jia Li, Hao Su and L. Fei-Fei, “Object bank: A high-level image representation for scene classification & semantic feature sparsification,” in *NIPS*, 2010, pp. 1378–1386. 2, 5, 7

[11] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache, “Log-Euclidean metrics for fast and simple calculus on diffusion tensors,” *Magnetic Resonance in Medicine*, vol. 56, no. 2, pp. 411–421, 2006. 2

[12] X. Pennec, P. Fillard, and N. Ayache, “A Riemannian framework for tensor computing,” *IJCV*, vol. 66, pp. 41–66, 2006. 2

[13] A. Cherian, S. Sra, A. Banerjee, and N. Papanikolopoulos, “Jensen-

Bregman logdet divergence with application to efficient similarity search for covariance matrices," *TPAMI*, vol. 35, no. 9, pp. 2161–2174, 2013. 2

[14] Z. Wang and B. Vemuri, "An affine invariant tensor dissimilarity measure and its applications to tensor-valued image segmentation," in *CVPR*, vol. 1, June 2004, pp. 228–233. 2

[15] O. Tuzel, F. Porikli, and P. Meer, "Pedestrian detection via classification on riemannian manifolds," *PAMI*, vol. 30, no. 10, pp. 1713–1727, Oct 2008. 2

[16] R. Vemulapalli, J. Pillai, and R. Chellappa, "Kernel learning for extrinsic classification of manifold features," in *CVPR*, June 2013, pp. 1782–1789. 2

[17] J. Ye, R. Janardan, S. Member, C. H. Park, and H. Park, "An optimization criterion for generalized discriminant analysis on undersampled problems," *TPAMI*, vol. 26, pp. 982–994, 2004. 2

[18] S. Yan, Y. Hu, D. Xu, H.-J. Zhang, B. Zhang, and Q. Cheng, "Nonlinear discriminant analysis on embedded manifold," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 4, pp. 468–477, 2007. 2

[19] J. Hamm and D. D. Lee, "Grassmann discriminant analysis: A unifying view on subspace-based learning," in *ICML*, 2008, pp. 376–383. 2

[20] M. T. Harandi, C. Sanderson, S. A. Shirazi, and B. C. Lovell, "Graph embedding discriminant analysis on Grassmannian manifolds for improved image set matching," in *CVPR*, 2011, pp. 2705–2712. 2

[21] P. Li, Q. Wang, and L. Zhang, "A novel earth mover's distance methodology for image matching with gaussian mixture models," in *ICCV*, 2013, pp. 1689–1696. 2, 3

[22] R. Wang, H. Guo, L. S. Davis, and Q. Dai, "Covariance discriminative learning: A natural and efficient approach to image set classification," in *CVPR*, 2012, pp. 2496–2503. 3

[23] O. Tuzel, F. M. Porikli, and P. Meer, "Learning on Lie groups for invariant detection and tracking," in *CVPR*, 2008, pp. 1–8. 3

[24] C. Chevalley, *Theory of Lie Groups*. Princeton University Press, 1999. 3

[25] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov, "Neighbourhood components analysis," in *NIPS*, 2004, pp. 513–520. 5

[26] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004. 5

[27] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *CVPR*, 2006, pp. 2169–2178. 5, 6, 7

[28] M. Yang, L. Zhang, S.-K. Shiu, and D. Zhang, "Robust kernel representation with statistical local features for face recognition," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 6, pp. 900–912, 2013. 5

[29] R. F. Li Fei-fei and P. Perona, "Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories," in *CVPRW*, 2004. 5

[30] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *CVPR*, 2009, pp. 413–420. 5, 6, 7

[31] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results," http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html. 5, 6

[32] T. Zhang, B. Ghanem, S. Liu, C. Xu, and N. Ahuja, "Low-rank sparse coding for image classification," in *ICCV*, 2013, pp. 281–288. 5, 6, 7

[33] J. Feng, B. Ni, Q. Tian, and S. Yan, "Geometric ℓ_p-norm feature pooling for image classification," in *CVPR*, 2011, pp. 2697–2704. 5, 7

[34] R. Kwitt, N. Vasconcelos, and N. Rasiwasia, "Scene recognition on the semantic manifold," in *ECCV*, 2012, pp. 359–372. 6, 7

[35] Q. Li, J. Wu, and Z. Tu, "Harvesting mid-level visual concepts from large-scale internet images," in *CVPR*, 2013, pp. 851–858. 6, 7

[36] T. Kobayashi, "Bof meets hog: Feature extraction based on histograms of oriented p.d.f. gradients for image classification," in *CVPR*, 2013, pp. 747–754. 6, 7

[37] J. Sun and J. Ponce, "Learning discriminative part detectors for image classification and cosegmentation," in *ICCV*, 2013, pp. 3400–3407. 6, 7

[38] T. T. Basura Fernando, Elisa Fromont, "Effective use of frequent itemset mining for image classification," in *ECCV*, 2012, pp. 214–227. 6, 7

[39] Y. Zheng, Y.-G. Jiang, and X. Xue, "Learning hybrid part filters for scene recognition," in *ECCV*, 2012, pp. 172–185. 7

[40] S. McCann and D. G. Lowe, "Local naive Bayes nearest neighbor for image classification," in *CVPR*, 2012, pp. 3650–3656. 7

[41] H. Nguyen, V. Patel, N. Nasrabadi, and R. Chellappa, "Sparse embedding: A framework for sparsity promoting dimensionality reduction," in *ECCV*, 2012, pp. 414–427. 6, 7

[42] S. Gao, I. W.-H. Tsang, and L.-T. Chia, "Kernel sparse representation for image classification and face recognition," in *ECCV*, 2010, pp. 1–14. 6, 7

[43] O. Duchenne, A. Joulin, and J. Ponce, "A graph-matching kernel for object categorization," in *ICCV*, 2011, pp. 1792–1799. 7

[44] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *ECCV*, 2010, pp. 143–156. 6

[45] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org/, 2008. 6

[46] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 27:1–27:27, 2011. 6

APPENDIX

Proof of Eqn. (14) and (15)

By using the transformation function $\log(Y_i) = P^T \log(X_i)P$, we have

$$
\begin{aligned}
f_1 =& \sum_{i,j} ||\log(Y_i) - \log(Y_j)||_F^2 W_{ij}^w \\
=& \sum_{i,j} ||P^T \log(X_i)P - P^T \log(X_j)P||_F^2 W_{ij}^w \\
=& \sum_{i,j} \mathrm{tr}((P^T \log(X_i)P - P^T \log(X_j)P)^T \\
& (P^T \log(X_i)P - P^T \log(X_j)P))W_{ij}^w \\
=& \sum_{i,j} \mathrm{tr}((P^T \log(X_i)P - P^T \log(X_j)P) \\
& (P^T \log(X_i)P - P^T \log(X_j)P))W_{ij}^w \\
=& \sum_{i,j} \mathrm{tr}(P^T \log(X_i)PP^T \log(X_i)P - \\
& P^T \log(X_i)PP^T \log(X_j)P - \\
& P^T \log(X_j)PP^T \log(X_i)P + \\
& P^T \log(X_j)PP^T \log(X_j)P)W_{ij}^w \\
=& 2\sum_{i,j} \mathrm{tr}(P^T \log(X_i)PP^T \log(X_i)P \\
& - P^T \log(X_i)PP^T \log(X_j)P)W_{ij}^w.
\end{aligned}
$$

Similarly,

$$
\begin{aligned}
f_2 =& \sum_{i,j} ||\log(Y_i) - \log(Y_j)||_F^2 W_{ij}^b \\
=& 2\sum_{i,j} \mathrm{tr}(P^T \log(X_i)PP^T \log(X_i)P \\
& - P^T \log(X_i)PP^T \log(X_j)P)W_{ij}^b.
\end{aligned}
$$

Proof of Eqn. (17)

Differentiating the objective function f_1 with respect to the transformation matrix P is:

$$
\begin{aligned}
\frac{\partial \widetilde{f_1}}{\partial P} =& 2\sum_{i,j} \mathrm{tr}(4\log(X_i)PP^T \log(X_i)P - 2\log(X_i) \\
& PP^T \log(X_j)P - 2\log(X_j)PP^T \log(X_i)P)W_{ij}^w \\
=& 8\sum_{i,j} (\log(X_i)PP^T \log(X_i)P \\
& - \log(X_i)PP^T \log(X_j)P)W_{ij}^w.
\end{aligned}
\tag{22}
$$

Similarly,

$$
\begin{aligned}
\frac{\partial \widetilde{f_2}}{\partial P} =& 8\sum_{i,j} (\log(X_i)PP^T \log(X_i)P \\
& - \log(X_i)PP^T \log(X_j)P)W_{ij}^b.
\end{aligned}
\tag{23}
$$

Now, combining (22) and (23) gives

$$
\begin{aligned}
\frac{\partial (f_1 - f_2)}{\partial P} =& 8\sum_{i,j} (\log(X_i)PP^T \log(X_i)P \\
& - \log(X_i)PP^T \log(X_j)P)(W_{ij}^w - W_{ij}^b).
\end{aligned}
\tag{24}
$$

Chunyan Xu received the B.Sc. degree from Shandong Normal University in 2007 and the M.Sc. degree from Huazhong Normal University in 2010. She is currently working toward the Ph.D. degree in the School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, and also in Department of Electrical and Computer Engineering, National University of Singapore. Her research interests include computer vision, manifold learning and kernel methods.

Canyi Lu received the bachelor of mathematics from Fuzhou University in 2009, and the master degree in the pattern recognition and intelligent system in 2012. From August 2013, he was a phd student with the Department of Electrical and Computer Engineering at National University of Singapore. His research interests include computer vision and machine learning. His homepage is https://sites.google.com/site/canyilu.

Junbin Gao graduated from Huazhong University of Science and Technology (HUST), China in 1982 with BSc. degree in Computational Mathematics and obtained PhD from Dalian University of Technology, China in 1991. He is a Professor in Computing Science in the School of Computing and Mathematatics at Charles Sturt University, Australia. He was a senior lecturer, a lecturer in Computer Science from 2001 to 2005 at University of New England, Australia. From 1982 to 2001 he was an associate lecturer, lecturer, associate professor and professor in Department of Mathematics at HUST. His main research interests include machine learning, data mining, Bayesian learning and inference, and image analysis.

Wei Zheng received the Bachelor's degree from Tsinghua University, Beijing, China, in 2006. He got the Ph.D. degree at Institute of Computing Technology, Chinese Academy of Sciences, Beijing, in 2013. Currently, he is a researcher at Beijing Samsung Telecom R&D Center, Beijing. His research interests include image categorization, object detection, and scene analysis.

Tianjiang Wang received the B. Sc. degree in computational mathematics in 1982 and the PhD degree in computer science in 1999 from Huazhong University of Science and Technology (HUST), Wuhan, China. He is currently a Professor with the School of Computer Science, Huazhong University of Science and Technology, Wuhan, China. He has finished some related projects and is the author of more than 20 related papers. His research interests include machine learning, computer vision, and data mining.

Shuicheng Yan is currently an Associate Professor at the Department of Electrical and Computer Engineering at National University of Singapore, and the founding lead of the Learning and Vision Research Group (http://www.lv-nus.org). Dr. Yan's research areas include machine learning, computer vision and multimedia, and he has authored/co-authored hundreds of technical papers over a wide range of research topics, with Google Scholar citation $>$ 15,000 times and H-index 51. He is ISI Highly-cited Researcher, 2014 and IAPR Fellow 2014. He has been serving as an associate editor of IEEE TKDE, TCSVT and ACM Transactions on Intelligent Systems and Technology (ACM TIST). He received the Best Paper Awards from ACM MM'13 (Best Paper and Best Student Paper), ACM MM12 (Best Demo), PCM'11, ACM MM10, ICME10 and ICIMCS'09, the runner-up prize of ILSVRC'13, the winner prize of ILSVRC14 detection task, the winner prizes of the classification task in PASCAL VOC 2010-2012, the winner prize of the segmentation task in PASCAL VOC 2012, the honourable mention prize of the detection task in PASCAL VOC'10, 2010 TCSVT Best Associate Editor (BAE) Award, 2010 Young Faculty Research Award, 2011 Singapore Young Scientist Award, and 2012 NUS Young Researcher Award.