Optimal Discriminative Projection for Sparse Representation-Based Classification via Bilevel Optimization

Guoqing Zhang, Huaijiang Sun, Yuhui Zheng, Guiyu Xia, Lei Feng, and Quansen Sun

***Abstract*— Recently, sparse representation-based classification (SRC) has been widely studied and has produced state-of-the-art results in various classification tasks. Learning useful and computationally convenient representations from complex redundant and highly variable visual data is crucial for the success of SRC. However, how to find the best feature representation to work with SRC remains an open question. In this paper, we present a novel discriminative projection learning approach with the objective of seeking a projection matrix such that the learned low-dimensional representation can fit SRC well and that it has well discriminant ability. More specifically, we formulate the learning algorithm as a bilevel optimization problem, where the optimization includes an ℓ_1-norm minimization problem in its constraints. Through the bilevel optimization model, the relationship between sparse representation and the desired feature projection can be explicitly exploited during the learning process. Therefore, SRC can achieve a better performance in the transformed subspace. The optimization model can be solved by using a stochastic gradient ascent algorithm, and the desired gradient is computed using implicit differentiation. Furthermore, our method can be easily extended to learn a dictionary. The extensive experimental results on a series of benchmark databases show that our method outperforms many state-of-the-art algorithms.**

***Index Terms*— Sparse representation, discriminative projection, bilevel optimization, dictionary learning.**

I. Introduction

OVER the past years, as a promising technique for representing high-dimensional data efficiently and providing resilience against noise, sparse coding has been successfully applied in various computer vision and pattern recognition tasks, such as face recognition [1], [2], image restoration [3], object classification [4], [5] and texture classification [6]. Sparse coding aims to represent each input signal as a linear combination of a few atoms in a dictionary that is usually over-complete. The coefficients of the linear combination are called sparse codes. Using sparsity as a prior has led to state-of-the-art results in many fields [7], [8]. Wright *et al.* [7] presented a sparse representation-based classification (SRC) method for face recognition and achieved promising performance. Zhang *et al.* [8] presented a discriminative, structured low-rank framework for image classification.

Because of its effective and robust to pose, illumination and expression as well as occlusion and disguise, SRC has been widely used in many recognition tasks and has achieved impressive performance in recent years [9]–[13]. However, when the training images per subject are insufficient or the dimension of images is far greater than the training sample size, the performance of SRC degrades significantly. Therefore, it is necessary to perform a dimensionality reduction (feature extraction) that can find low-dimensional and compact representations before implementing SRC. A number of dimensionality reduction algorithms have been proposed to reduce the dimensions of the data and improve the discriminativeness of the features [14]–[17]. However, some researchers [7] have claimed that the choice of features is not important for SRC as long as the sparse representation is correctly computed and the number of features is large enough. However when the dimension of the sample is relatively small, the classification performances using different feature representation methods are significantly different. Zhang *et al.* [18] presented an unsupervised dimensionality reduction method for SRC that improves performance. Therefore, a well-designed dimensionality reduction method can improve the performance of SRC.

Recently, some sparse representation-based discriminative projection methods have been proposed [19]–[24]. Qiao *et al.* [19] gave a sparsity preserving projection (SPP) algorithm to preserve the sparse reconstruction relationship of original data in a new subspace. Gui *et al.* [20] presented a discriminant sparse embedding algorithm by adding the discriminant information into SPP. Zhou and Tao [23] presented a double shrinking model to build a sparse projection matrix for dimensionality reduction. To further enhance discrimination,

Manuscript received November 17, 2017; revised October 18, 2018 and January 27, 2019; accepted February 23, 2019. Date of publication March 4, 2019; date of current version April 3, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61772272, Grant 61673220, Grant 61801199, and Grant 61806099, in part by the Natural Science Foundation of Jiangsu Province, China, under Grant BK20180790, and in part by the University Natural Science Research Foundation of Jiangsu Province under Grant 18KJB520033F. This paper was recommended by Associate Editor G. Cheung. *(Corresponding author: Huaijiang Sun.)*

G. Zhang, Y. Zheng, and G. Xia are with the School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China (e-mail: xiayang14551@163.com; zhengyh@vip.126.com; xiaguiyu1989@sina.com).

H. Sun and Q. Sun are with the School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: sunhuaijiang@njust.edu.cn; sunquansen@njust.edu.cn).

L. Feng is with the School of Computer Engineering, Jinling Institute of Technology, Nanjing 211169, China (e-mail: fenglei492327278@126.com).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2019.2902672

Feng *et al.* [24] jointly learned a dimensionality reduction matrix and a discriminative dictionary for face recognition. Although these methods achieved very competitive performance, none of them have a direct connection to SRC. Thus, the extracted features may not be optimal for the final classification. How to extract the most discriminative and robust features that can best work with SRC is a key issue and a challenging problem.

Since SRC [7] predicts the class label of a given testing sample based on the representation residual, Yang *et al.* [25] proposed an SRC-steered discriminative projection (SRC-DP) method. Until now, SRC-DP is the most closely connected feature representation algorithm with SRC and this idea has been widely applied in many studies in recent years [26]–[31]. Zhang *et al.* [27] proposed a multiple kernel learning-based orthogonal discriminative projection method for image classification. Yang *et al.* [28] devised a feature extraction method based on collaborative representation. Gao *et al.* [29] proposed a discriminative sparsity preserving projections (DSPP) method. Yan and Yang [30] presented a sparse discriminative feature selection method. Although improved performance has been reported in SRC-DP, it still has several drawbacks. Firstly, the model of SRC-DP is a trace ratio problem. For computational convenience, Yang *et al.* [25] converted this problem into a more tractable ratio trace problem, which is solved using the iterated generalized eigenvalue decomposition algorithm. Its solution may deviate from the original objectives and lead to uncertainty in subsequent classification task. Secondly, SRC-DP does not consider the relationship between the sparse representation and the desired projection matrix, which is important for improving the performance of SRC.

Therefore, in order to enhance the recognition performance of SRC, we propose a novel feature representation method, namely optimal discriminative projection for sparse representation-based classification via bilevel optimization (ODP-SRC). Our method aims to learn a feature projection matrix such that the extracted features in the low dimensional subspace can fit SRC well and simultaneously characterize the discriminant structure embedded in high-dimensional data well. More specifically, we model our learning algorithm as a bilevel optimization problem [32], [33] in which the relationship between sparse representation and the desired discriminative projection can be expressed explicitly in the objective function. The optimization model can be solved efficiently using the stochastic gradient ascent procedure, and implicit differentiation is employed to calculate the desired gradient [32], [34]–[36]. Furthermore, the proposed approach can be easily extended to learn a dictionary simultaneously with the projection matrix using the same optimization method. Thus, the performance of SRC can be further improved.

II. Related Work

A. Sparse Representation-Based Classification

Sparse coding has recently attracted much attention in vision signal and image processing research [1], [37]. Suppose that we have c different classes of subjects, denote the column-arranged training samples of class i as $\boldsymbol{A}_i = [\boldsymbol{y}_{i,1}, \boldsymbol{y}_{i,2}, \cdots, \boldsymbol{y}_{i,n_i}] \in R^{d\times n_i}$, where d is the dimension of the training data. The entire training set is defined as $\boldsymbol{A} = [\boldsymbol{A}_1, \boldsymbol{A}_2, \cdots, \boldsymbol{A}_c] \in R^{d\times n}$, where $n = \sum_{i=1}^{c} n_i$. Given a test sample $\boldsymbol{y}$, we represent $\boldsymbol{y}$ in an overcomplete dictionary whose basis vectors are training samples themselves, i.e., $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{\alpha}$, where $\boldsymbol{\alpha} = [\boldsymbol{\alpha}_1; \boldsymbol{\alpha}_2; \cdots; \boldsymbol{\alpha}_c]$ and $\boldsymbol{\alpha}_i$ is the sparse coding vector over $\boldsymbol{A}_i$. If $\boldsymbol{y}$ is from the i-th class, usually $\boldsymbol{y} = \boldsymbol{A}_i\boldsymbol{\alpha}_i$ holds well. This implies that most coefficients in $\boldsymbol{\alpha}$ are nearly zeros and only $\boldsymbol{\alpha}_i$ has significant nonzero entries. Then the classic sparse coding problem can be formulated as follows

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1, \tag{1}$$

where λ is the regularization parameter for controlling the sparsity of $\boldsymbol{\alpha}$.

Once Eq. (1) is solved, the classification can be performed using minimum class-wise reconstruction error. The reconstruction error for each class is computed by

$$r_i = \|\boldsymbol{y} - \boldsymbol{A}\delta_i(\hat{\boldsymbol{\alpha}})\|_2^2, \tag{2}$$

where $\delta_i(\cdot) : R^n \to R^n$ is the characteristic function which selects the coefficients associated with class i. The classification is made by $identify(\boldsymbol{y}) = \arg\min_i\{r_i\}$.

B. Sparse Representation Classifier-Steered Discriminative Projection

Let $\boldsymbol{X} = [\boldsymbol{X}_1, \boldsymbol{X}_2, \cdots, \boldsymbol{X}_c] \in R^{m\times n}$ be the training data matrix in the original input space, where $\boldsymbol{X}_i = [\boldsymbol{x}_{i,1}, \boldsymbol{x}_{i,2}, \cdots, \boldsymbol{x}_{i,n_i}] \in R^{m\times n_i}$ is the matrix formed by the training samples of class i. Under a linear transformation $\boldsymbol{y} = \boldsymbol{P}^T\boldsymbol{x}$, each data sample $\boldsymbol{x}_{ij}$ in input space R^m is mapped into a d-dimensional space R^d. As a result, the data matrix in the input space is converted into the one in R^d, that is $\boldsymbol{A} = \boldsymbol{P}^T\boldsymbol{X}$.

In the reduced d-dimensional space, consider the classification rule of SRC. For each training sample $\boldsymbol{y}_{ij}$, leave it out from the training set and use the remaining training samples to linearly represent it. By solving the ℓ_1 optimization problem in Eq. (1), a sparse representation coefficient vector $\boldsymbol{\alpha}_{ij}$ can be obtained. Then, the within-class and between-class reconstruction residuals in the mapped d-dimensional space can be defined as:

$$\begin{aligned} J_w &= tr\left(\sum_{i,j} \boldsymbol{P}^T(\boldsymbol{x}_{ij} - \boldsymbol{X}\delta_i(\boldsymbol{\alpha}_{ij}))(\boldsymbol{x}_{ij} - \boldsymbol{X}\delta_i(\boldsymbol{\alpha}_{ij}))^T\boldsymbol{P}\right) \\ &= tr(\boldsymbol{P}^T\boldsymbol{S}_w^L\boldsymbol{P}), \end{aligned} \tag{3}$$

and

$$\begin{aligned} J_b &= tr\left(\sum_{i,j}\sum_{s\neq i} \boldsymbol{P}^T(\boldsymbol{x}_{ij} - \boldsymbol{X}\delta_s(\boldsymbol{\alpha}_{ij}))(\boldsymbol{x}_{ij} - \boldsymbol{X}\delta_s(\boldsymbol{\alpha}_{ij}))^T\boldsymbol{P}\right) \\ &= tr(\boldsymbol{P}^T\boldsymbol{S}_b^L\boldsymbol{P}), \end{aligned} \tag{4}$$

where $tr(\cdot)$ is the trace operator, $\delta_i(\boldsymbol{\alpha}_{ij})$ and $\delta_s(\boldsymbol{\alpha}_{ij})$ are vectors whose only nonzero entries are the entries in $\boldsymbol{\alpha}_{ij}$ associated with classes i and $s (s \neq i)$, respectively. $\boldsymbol{S}_w^L = \sum_{i,j}(\boldsymbol{x}_{ij} - \boldsymbol{X}\delta_i(\boldsymbol{\alpha}_{ij}))(\boldsymbol{x}_{ij} - \boldsymbol{X}\delta_i(\boldsymbol{\alpha}_{ij}))^T$ and $\boldsymbol{S}_b^L = \sum_{i,j}\sum_{s\neq i}(\boldsymbol{x}_{ij} - \boldsymbol{X}\delta_s(\boldsymbol{\alpha}_{ij}))(\boldsymbol{x}_{ij} - \boldsymbol{X}\delta_s(\boldsymbol{\alpha}_{ij}))^T$ are called the within-class and between-class sparse scatter matrices, respectively.

The criterion of SRC-DP is to maximize the between-class reconstruction residual and minimize the within-class reconstruction residual and can be expressed as follows:

$$J(\boldsymbol{P}) = \max_{\boldsymbol{P}} \frac{tr(\boldsymbol{P}^T \boldsymbol{S}_b^L \boldsymbol{P})}{tr(\boldsymbol{P}^T \boldsymbol{S}_w^L \boldsymbol{P})}. \tag{5}$$

Then, the solution of the optimal projection matrix can be chosen as the generalized eigenvectors of $\boldsymbol{S}_b^L \varphi = \lambda \boldsymbol{S}_w^L \varphi$ corresponding to d largest eigenvalues [25]. However, when $\boldsymbol{S}_w^L$ is singular, it is difficult to solve the criterion in Eq. (5) using the generalized eigenvalue decomposition algorithm. In addition, it does not considers the relationship between the sparse representation and the desired projection during the learning process, thus, the learned features may not be optimal for SRC.

III. Optimal Discriminative Projection for Sparse Representation-Based Classification

A. Formulation

Our proposed ODP-SRC aims to learn a projection matrix such that SRC achieves optimum performance in the transformed low-dimensional space. For each training sample $\boldsymbol{y}_i$, similar to SRC-DP, represent it using the remaining training samples. Once the sparse representation coefficient $\boldsymbol{\alpha}_i$ is obtained, we define the within-class reconstruction residual in the projected space J_w as follows:

$$\begin{aligned} J_w &= tr\left(\sum_{i=1}^{n}(\boldsymbol{y}_i - \boldsymbol{A}\delta_{\ell_i}(\boldsymbol{\alpha}_i))(\boldsymbol{y}_i - \boldsymbol{A}\delta_{\ell_i}(\boldsymbol{\alpha}_i))^T\right) \\ &= tr\left(\sum_{i=1}^{n}(\boldsymbol{P}^T\boldsymbol{x}_i - \boldsymbol{P}^T\boldsymbol{X}\delta_{\ell_i}(\boldsymbol{\alpha}_i))(\boldsymbol{P}^T\boldsymbol{x}_i - \boldsymbol{P}^T\boldsymbol{X}\delta_{\ell_i}(\boldsymbol{\alpha}_i))^T\right) \\ &= tr\left(\sum_{i=1}^{n}\boldsymbol{P}^T(\boldsymbol{x}_i - \boldsymbol{X}\boldsymbol{I}_{\ell_i}\boldsymbol{\alpha}_i)(\boldsymbol{x}_i - \boldsymbol{X}\boldsymbol{I}_{\ell_i}\boldsymbol{\alpha}_i)^T\boldsymbol{P}\right) \\ &= \sum_{i=1}^{n} tr\left(\boldsymbol{P}^T\boldsymbol{S}_w^i\boldsymbol{P}\right), \end{aligned} \tag{6}$$

where $\boldsymbol{S}_w^i = (\boldsymbol{x}_i - \boldsymbol{X}\boldsymbol{I}_{\ell_i}\boldsymbol{\alpha}_i)(\boldsymbol{x}_i - \boldsymbol{X}\boldsymbol{I}_{\ell_i}\boldsymbol{\alpha}_i)^T$ is the within-class scatter matrix with respect to the i-th sample, and $\boldsymbol{I}_{\ell_i} \in R^{n\times n}$ is a diagonal matrix whose diagonal elements are 1 only when associated with the non-zero entries of $\delta_{\ell_i}(\boldsymbol{\alpha}_i)$, 0 otherwise. ℓ_i is the label of $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$.

Similarly, we can define J_b to evaluate the between-class reconstruction residual as follows:

$$\begin{aligned} J_b &= tr\left(\sum_{i=1}^{n}\sum_{s\neq\ell_i}^{c}(\boldsymbol{y}_i - \boldsymbol{A}\delta_s(\boldsymbol{\alpha}_i))(\boldsymbol{y}_i - \boldsymbol{A}\delta_s(\boldsymbol{\alpha}_i))^T\right) \\ &= tr\left(\sum_{i=1}^{n}\sum_{s\neq\ell_i}^{c}(\boldsymbol{P}^T\boldsymbol{x}_i - \boldsymbol{P}^T\boldsymbol{X}\delta_s(\boldsymbol{\alpha}_i))(\boldsymbol{P}^T\boldsymbol{x}_i - \boldsymbol{P}^T\boldsymbol{X}\delta_s(\boldsymbol{\alpha}_i))^T\right) \\ &= tr\left(\sum_{i=1}^{n}\boldsymbol{P}^T\sum_{s\neq\ell_i}^{c}(\boldsymbol{x}_i - \boldsymbol{X}\boldsymbol{I}_s\boldsymbol{\alpha}_i)(\boldsymbol{x}_i - \boldsymbol{X}\boldsymbol{I}_s\boldsymbol{\alpha}_i)^T\boldsymbol{P}\right) \\ &= \sum_{i=1}^{n} tr\left(\boldsymbol{P}^T\boldsymbol{S}_b^i\boldsymbol{P}\right), \end{aligned} \tag{7}$$

where $\boldsymbol{S}_b^i = \sum_{s\neq\ell_i}^{c}(\boldsymbol{x}_i - \boldsymbol{X}\boldsymbol{I}_s\boldsymbol{\alpha}_i)(\boldsymbol{x}_i - \boldsymbol{X}\boldsymbol{I}_s\boldsymbol{\alpha}_i)^T$ is the between-class scatter matrix with respect to the i-th sample, $\boldsymbol{I}_s \in R^{n\times n}$ is a diagonal matrix whose diagonal elements are 1 only associated when with the non-zero entries of $\delta_s(\boldsymbol{\alpha}_i)$, 0 otherwise.

$\boldsymbol{\alpha}_i$ is calculated in the transformed d-dimensional space by using the following ℓ_1 optimization problem:

$$\begin{aligned} \boldsymbol{\alpha}_i &= \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{y}_i - \boldsymbol{A}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1 \\ &= \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{P}^T\boldsymbol{x}_i - \boldsymbol{P}^T\boldsymbol{X}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1 \end{aligned} \tag{8}$$

where $\boldsymbol{\alpha}_i$ is the sparse representation of $\boldsymbol{P}^T\boldsymbol{x}_i$ with respect to $\boldsymbol{P}^T\boldsymbol{X}$. From Eq. (8) we can see that $\boldsymbol{\alpha}_i$ is dependent on $\boldsymbol{P}$; therefore, we should exploit the relationship between $\boldsymbol{\alpha}_i$ and the desired projection matrix during the learning process.

Thus, we can learn $\boldsymbol{P}$ by maximizing the following objective function with the sparsity constraint using ℓ_1-norm regularization:

$$\begin{aligned} J(\boldsymbol{P}) &= \sum_{i=1}^{n}\max_{\boldsymbol{P}} J_i(\boldsymbol{P}) = \sum_{i=1}^{n}\max_{\boldsymbol{P}} \frac{tr(\boldsymbol{P}^T\boldsymbol{S}_b^i\boldsymbol{P})}{tr(\boldsymbol{P}^T\boldsymbol{S}_w^i\boldsymbol{P})} \\ s.t.\boldsymbol{\alpha}_i &= \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{P}^T\boldsymbol{x}_i - \boldsymbol{P}^T\boldsymbol{X}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1, \\ &\quad \boldsymbol{A} = \boldsymbol{P}^T\boldsymbol{X}, \\ &\quad \|\boldsymbol{A}(:,n)\|_2 \leq 1, \quad i = 1, 2, \cdots, n. \end{aligned} \tag{9}$$

We can see that the objective function in Eq. (9) is a bilevel optimization model [33], where optimization problems (ℓ_1-norm minimization in this case) appear in the constraints. In our method, the upper-level problem J selects the projection $\boldsymbol{P}$, and the lower-level ℓ_1-norm minimization returns the sparse codes $\boldsymbol{\alpha}_i$ to the upper-level J in order to evaluate the objective function value. Through the bilevel optimization model, the relationship between $\boldsymbol{\alpha}_i$ and $\boldsymbol{P}$ can be expressed explicitly in the objective function. The desired projection can be obtained by using a stochastic gradient ascent algorithm, as provided in [32], [34] and [35], where backpropagation and implicit differentiation [38] are adopted to compute the gradient of Eq. (9) with respect to the projection matrix.

B. Optimization Method

The objective function in Eq. (9) is highly non-convex. To solve this problem, we can use a stochastic gradient ascent method for optimization. In Eq. (9), the upper-level optimization depends on the variable $\boldsymbol{\alpha}_i$, which is the output of the lower level ℓ_1-minimization problem. We assume that we can define $\boldsymbol{\alpha}_i$ as an implicit function $\boldsymbol{\alpha}_i(\boldsymbol{P})$ of $\boldsymbol{P}$ depending on the input $\boldsymbol{x}_i$. The problem in Eq. (9) may be viewed solely in terms of the upper-level variable $\boldsymbol{P}$. Given a feasible point for $\boldsymbol{P}$, the ascent method makes an attempt to find an ascent direction along which the upper-level objective value increases. The major issue concerning the ascent method is the availability of the gradient of the upper-level objective, $\nabla_{\boldsymbol{P}} J_i$, at a feasible point. Applying the chain rule, we have

$$\nabla_{\mathbf{P}} J_i = \frac{\partial J_i}{\partial \mathbf{P}} + \frac{\partial J_i}{\partial \mathbf{S}_b^i}\frac{\partial \mathbf{S}_b^i}{\partial \boldsymbol{\alpha}_i}\frac{\partial \boldsymbol{\alpha}_i}{\partial \mathbf{P}} + \frac{\partial J_i}{\partial \mathbf{S}_w^i}\frac{\partial \mathbf{S}_w^i}{\partial \boldsymbol{\alpha}_i}\frac{\partial \boldsymbol{\alpha}_i}{\partial \mathbf{P}} \tag{10}$$

where the function is evaluated at the current iteration. The problem is reduced to computing the gradients of the sparse representation coefficient $\boldsymbol{\alpha}_i$ with respect to the projection matrix $\boldsymbol{P}$. Once $\partial \boldsymbol{\alpha}_i / \partial \boldsymbol{P}$ is computed, we can get $\nabla_{\boldsymbol{P}} J_i$.

For ease of presentation, we drop the subscripts of $\boldsymbol{\alpha}_i$ in the following. Denote α_j is the j-th element of $\boldsymbol{\alpha}$, and denote $\boldsymbol{\Lambda}$ as the index set of nonzero sparse coefficients of $\boldsymbol{\alpha}$, i.e., $\boldsymbol{\Lambda} = \{j : \alpha_j \neq 0\}$. Let $\widetilde{\boldsymbol{\alpha}}$ denote the vector built with the elements $\{\alpha_j\}_{j \in \boldsymbol{\Lambda}}$ and $\widetilde{\boldsymbol{X}}$ being the corresponding columns (the supports selected by $\widetilde{\boldsymbol{\alpha}}$). It is easy to find that

$$\frac{\partial J_i}{\partial \boldsymbol{P}} = \frac{2\boldsymbol{S}_b^i \boldsymbol{P} tr\left(\boldsymbol{P}^T \boldsymbol{S}_w^i \boldsymbol{P}\right) - 2\boldsymbol{S}_w^i \boldsymbol{P} tr\left(\boldsymbol{P}^T \boldsymbol{S}_b^i \boldsymbol{P}\right)}{tr\left(\boldsymbol{P}^T \boldsymbol{S}_w^i \boldsymbol{P}\right)^2}, \tag{11}$$

$$\frac{\partial J_i}{\partial \boldsymbol{S}_b^i} = \frac{\boldsymbol{P}\boldsymbol{P}^T}{tr\left(\boldsymbol{P}^T \boldsymbol{S}_w^i \boldsymbol{P}\right)}, \quad \frac{\partial J_i}{\partial \boldsymbol{S}_w^i} = \frac{-tr\left(\boldsymbol{P}^T \boldsymbol{S}_b^i \boldsymbol{P}\right) \boldsymbol{P}\boldsymbol{P}^T}{tr\left(\boldsymbol{P}^T \boldsymbol{S}_w^i \boldsymbol{P}\right)^2} \tag{12}$$

$$\frac{\partial \boldsymbol{S}_b^i}{\partial \boldsymbol{\alpha}_i} = \frac{\partial \boldsymbol{S}_b^i}{\partial \widetilde{\boldsymbol{\alpha}}_i} = \sum_{s \neq \ell_i} 2(\widetilde{\boldsymbol{I}}_s \widetilde{\boldsymbol{X}}^T \widetilde{\boldsymbol{X}} \widetilde{\boldsymbol{I}}_s \widetilde{\boldsymbol{\alpha}} - \widetilde{\boldsymbol{I}}_s \widetilde{\boldsymbol{X}}^T \boldsymbol{x}_i), \tag{13}$$

$$\frac{\partial \boldsymbol{S}_w^i}{\partial \boldsymbol{\alpha}_i} = \frac{\partial \boldsymbol{S}_w^i}{\partial \widetilde{\boldsymbol{\alpha}}_i} = 2(\widetilde{\boldsymbol{I}}_{\ell_i} \widetilde{\boldsymbol{X}}^T \widetilde{\boldsymbol{X}} \widetilde{\boldsymbol{I}}_{\ell_i} \widetilde{\boldsymbol{\alpha}} - \widetilde{\boldsymbol{I}}_{\ell_i} \widetilde{\boldsymbol{X}}^T \boldsymbol{x}_i). \tag{14}$$

where $\widetilde{\boldsymbol{I}}_{\ell_i}$ and $\widetilde{\boldsymbol{I}}_s$ consist of the columns of $\boldsymbol{I}_{\ell_i}$ and $\boldsymbol{I}_s$ corresponding to the index set $\boldsymbol{\Lambda}$. To evaluate the gradient in Eq. (10), we still need to find the derivative $\partial \boldsymbol{\alpha}_i / \partial \boldsymbol{P}$. However, there is no analytical link between $\boldsymbol{\alpha}$ and the projection $\boldsymbol{P}$. Following [32] and [34], we overcome this problem by using implicit differentiation to find the derivative.

For the lasso problem in Eq. (8), we have the following condition for the optimum $\boldsymbol{\alpha}$ [35], [39].

$$\frac{\partial \|\boldsymbol{P}^T \boldsymbol{x} - \boldsymbol{P}^T \boldsymbol{X} \boldsymbol{\alpha}\|_2^2}{\partial \alpha_j} + \lambda \cdot sign(\alpha_j) = 0, \quad for\ j \in \boldsymbol{\Lambda}, \tag{15}$$

$$\left| \frac{\partial \|\boldsymbol{P}^T \boldsymbol{x} - \boldsymbol{P}^T \boldsymbol{X} \boldsymbol{\alpha}\|_2^2}{\partial \alpha_j} \right| < \lambda, \quad for\ j \notin \boldsymbol{\Lambda}. \tag{16}$$

Eq. (15) is the stationary condition for $\boldsymbol{\alpha}$ to be optimal [32], which links $\boldsymbol{\alpha}$ and $\boldsymbol{P}$ analytically on the index set $\boldsymbol{\Lambda}$. We rewrite this condition as:

$$\widetilde{\boldsymbol{X}}^T \boldsymbol{P}\boldsymbol{P}^T \widetilde{\boldsymbol{X}} \widetilde{\boldsymbol{\alpha}} - \widetilde{\boldsymbol{X}}^T \boldsymbol{P}\boldsymbol{P}^T \boldsymbol{x} + \lambda \cdot sign(\widetilde{\boldsymbol{\alpha}}) = 0. \tag{17}$$

It is clear that $\widetilde{\boldsymbol{\alpha}}$ is a continuous function of $\boldsymbol{P}$ [40]. Therefore, a small perturbation on $\boldsymbol{P}$ will not change the signs of the elements in $\widetilde{\boldsymbol{\alpha}}$. As a result, we can apply the implicit differentiation on Eq. (17) to obtain:

$$\frac{\partial \{\widetilde{\boldsymbol{X}}^T \boldsymbol{P}\boldsymbol{P}^T \widetilde{\boldsymbol{X}} \widetilde{\boldsymbol{\alpha}} - \widetilde{\boldsymbol{X}}^T \boldsymbol{P}\boldsymbol{P}^T \boldsymbol{x}\}}{\partial \boldsymbol{P}} = \frac{\partial \{-\lambda \cdot sign(\widetilde{\boldsymbol{\alpha}})\}}{\partial \boldsymbol{P}}, \tag{18}$$

which gives:

$$\frac{\partial \widetilde{\boldsymbol{X}}^T \boldsymbol{P}\boldsymbol{P}^T \widetilde{\boldsymbol{X}}}{\partial \boldsymbol{P}} \widetilde{\boldsymbol{\alpha}} + \widetilde{\boldsymbol{X}}^T \boldsymbol{P}\boldsymbol{P}^T \widetilde{\boldsymbol{X}} \frac{\partial \widetilde{\boldsymbol{\alpha}}}{\partial \boldsymbol{P}} - \frac{\partial \widetilde{\boldsymbol{X}}^T \boldsymbol{P}\boldsymbol{P}^T \boldsymbol{x}}{\partial \boldsymbol{P}} = 0. \tag{19}$$

Then, the desired gradient can be solved by:

$$\frac{\partial \widetilde{\boldsymbol{\alpha}}}{\partial \boldsymbol{P}} = (\widetilde{\boldsymbol{X}}^T \boldsymbol{P}\boldsymbol{P}^T \widetilde{\boldsymbol{X}})^{-1} \left(\frac{\partial \widetilde{\boldsymbol{X}}^T \boldsymbol{P}\boldsymbol{P}^T \boldsymbol{x}}{\partial \boldsymbol{P}} - \frac{\partial \widetilde{\boldsymbol{X}}^T \boldsymbol{P}\boldsymbol{P}^T \widetilde{\boldsymbol{X}}}{\partial \boldsymbol{P}} \widetilde{\boldsymbol{\alpha}} \right). \tag{20}$$

Algorithm 1 ODP-SRC

Input: training samples $\{\boldsymbol{x}_i\}_{i=1}^n$, sparsity regularization λ.
Initial: initialize $\boldsymbol{P}^{(0)}$, the number of iterations $t = 1$,
Repeat
  For $i = 1, 2, \cdots, n$ **do**
    Computer gradient $\nabla_{\boldsymbol{P}} J_i$ according to Eq. (10);
    Update $\boldsymbol{P}^{(t)} = \boldsymbol{P}^{(t)} + \eta \cdot \nabla_{\boldsymbol{P}} J_i$, where η is the step size for stochastic gradient ascent. $\eta = min(\rho, \rho i_0 / i)$, where ρ is a constant, $i_0 = t/10$.
  end for
  Update $\boldsymbol{P}^{(t+1)} = \boldsymbol{P}^{(t)}$;
  $t = t + 1$;
Until convergence
Output: Projection matrix **P**

Since the number of non-zero coefficients is generally far smaller than the dimension m, the inverse $(\widetilde{\boldsymbol{X}}^T \boldsymbol{P}\boldsymbol{P}^T \widetilde{\boldsymbol{X}})^{-1}$ is well-conditioned. Eq. (20) only gives us the derivative function of $\widetilde{\boldsymbol{\alpha}}$ with respect to $\boldsymbol{P}$, which builds only on the index set $\boldsymbol{\Lambda}$. To evaluate Eq. (10), we can set the remaining gradient elements of $\partial \boldsymbol{\alpha} / \partial \boldsymbol{P}$ to zero. From a practical point of view, as long as the approximate derivative given by Eq. (10) is a feasible ascent direction for the optimization, the ascent method guarantees that the objective function will always increase for a feasible step along that direction [35].

With the gradient in Eq. (10) calculated, we employ a stochastic gradient ascent procedure for updating $\boldsymbol{P}$.

$$\boldsymbol{P}^{(t)} = \boldsymbol{P}^{(t)} + \eta \cdot \nabla_{\boldsymbol{P}} J_i \tag{21}$$

where η is the step size. The overall optimization procedure is summarized in Algorithm 1.

C. Dictionary Learning

The dictionary also plays an important role in SRC as it is expected to faithfully and discriminatively represent the query image. A dictionary can be learnt with the projection matrix $\boldsymbol{P}$ fixed, using the same optimization algorithm to iterative joint learn the projection matrix and dictionary. Denote the dictionary in the input space as $\boldsymbol{D} = [\mathbf{D}_1, \boldsymbol{D}_2, \cdots, \boldsymbol{D}_c] \in R^{m \times K}$, and in the reduced space as $\boldsymbol{B} = [\boldsymbol{B}_1, \boldsymbol{B}_2, \cdots, \boldsymbol{B}_c] \in R^{d \times K}$, where K is the dictionary size. Under a linear transformation, the dictionary in the reduced subspace is rewritten as $\boldsymbol{B} = \boldsymbol{P}^T \boldsymbol{D}$.

Thus, we can learn $\boldsymbol{P}$ and $\boldsymbol{D}$ by maximizing the following objective function where the ℓ_1-norm regularization is the sparsity constraint:

$$\begin{aligned} J(\boldsymbol{P}, \boldsymbol{D}) &= \sum_{i=1}^{n} \max_{\boldsymbol{P}, \boldsymbol{D}} J_i(\boldsymbol{P}, \boldsymbol{D}) \\ &= \sum_{i=1}^{n} \max_{\boldsymbol{P}, \boldsymbol{D}} \frac{tr(\boldsymbol{P}^T \boldsymbol{S}_b^i \boldsymbol{P})}{tr(\boldsymbol{P}^T \boldsymbol{S}_w^i \boldsymbol{P})} \\ s.t. \boldsymbol{\alpha}_i &= \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{P}^T \boldsymbol{x}_i - \boldsymbol{P}^T \boldsymbol{D} \boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_1, \\ & \|\boldsymbol{P}^T \boldsymbol{D}(:, k)\|_2 \leq 1, \quad \forall k \in \{1, 2, \cdots, k\}, \end{aligned} \tag{22}$$

where $\boldsymbol{S}_b^i = \sum_{s\neq\ell_i}^{c}(\boldsymbol{x}_i - \boldsymbol{D}\boldsymbol{I}_s\boldsymbol{\alpha}_i)(\boldsymbol{x}_i - \boldsymbol{D}\boldsymbol{I}_s\boldsymbol{\alpha}_i)^T$ and $\boldsymbol{S}_w^i = (\boldsymbol{x}_i - \boldsymbol{D}\boldsymbol{I}_{\ell_i}\boldsymbol{\alpha}_i)(\boldsymbol{x}_i - \boldsymbol{D}\boldsymbol{I}_{\ell_i}\boldsymbol{\alpha}_i)^T$. $\boldsymbol{I}_s \in R^{K\times K}$ and $\boldsymbol{I}_{\ell_i} \in R^{K\times K}$ are diagonal matrices whose elements are 1 only associated with the s-th and ℓ_i classes, $s \neq \ell_i$, 0 otherwise.

In this case, $\boldsymbol{\alpha}_i$ is the output of the low-level ℓ_1-norm minimization based on $\boldsymbol{D}$. We can assume that $\boldsymbol{\alpha}_i$ is an implicit function $\boldsymbol{\alpha}_i(\boldsymbol{D})$ depending on the input $\boldsymbol{x}_i$. Similar to the projection update, using the chain rule, no matter when $\partial\boldsymbol{\alpha}_i/\partial\boldsymbol{D}$ is well defined, we have

$$\nabla_{\boldsymbol{D}} J_i = \frac{\partial J_i}{\partial \boldsymbol{S}_w^i}\frac{\partial \boldsymbol{S}_w^i}{\partial \boldsymbol{D}} + \frac{\partial J_i}{\partial \boldsymbol{S}_w^i}\frac{\partial \boldsymbol{S}_w^i}{\partial \boldsymbol{\alpha}_i}\frac{\partial \boldsymbol{\alpha}_i}{\partial \boldsymbol{D}} + \frac{\partial J_i}{\partial \boldsymbol{S}_b^i}\frac{\partial \boldsymbol{S}_b^i}{\partial \boldsymbol{D}} + \frac{\partial J_i}{\partial \boldsymbol{S}_b^i}\frac{\partial \boldsymbol{S}_b^i}{\partial \boldsymbol{\alpha}_i}\frac{\partial \boldsymbol{\alpha}_i}{\partial \boldsymbol{D}} \quad (23)$$

The problem is reduced to computing the gradients as $\partial\boldsymbol{\alpha}_i/\partial\boldsymbol{D}$. It is clear that:

$$\frac{\partial J_i}{\partial \boldsymbol{S}_w^i} = \frac{-tr\left(\boldsymbol{P}^T\boldsymbol{S}_b^i\boldsymbol{P}\right)\boldsymbol{P}\boldsymbol{P}^T}{tr\left(\boldsymbol{P}^T\boldsymbol{S}_w^i\boldsymbol{P}\right)^2}, \quad (24)$$

$$\frac{\partial \boldsymbol{S}_w^i}{\partial \boldsymbol{D}} = 2(\boldsymbol{D}\boldsymbol{I}_{\ell_i}\boldsymbol{\alpha}_i - \boldsymbol{x}_i)(\boldsymbol{I}_{\ell_i}\boldsymbol{\alpha}_i)^T \quad (25)$$

$$\frac{\partial \boldsymbol{S}_w^i}{\partial \boldsymbol{\alpha}_i} = \frac{\partial \boldsymbol{S}_w^i}{\partial \widetilde{\boldsymbol{\alpha}}} = 2\widetilde{\boldsymbol{I}}_{\ell_i}^T\widetilde{\boldsymbol{D}}^T(\boldsymbol{D}\boldsymbol{I}_{\ell_i}\boldsymbol{\alpha}_i - \boldsymbol{x}_i) \quad (26)$$

$$\frac{\partial J_i}{\partial \boldsymbol{S}_b^i} = \frac{\boldsymbol{P}\boldsymbol{P}^T}{tr\left(\boldsymbol{P}^T\boldsymbol{S}_w^i\boldsymbol{P}\right)}, \quad (27)$$

$$\frac{\partial \boldsymbol{S}_b^i}{\partial \boldsymbol{D}} = \sum_{s\neq\ell_i} 2(\boldsymbol{D}\boldsymbol{I}_s\boldsymbol{\alpha}_i - \boldsymbol{x}_i)(\boldsymbol{I}_s\boldsymbol{\alpha}_i)^T \quad (28)$$

$$\frac{\partial \boldsymbol{S}_b^i}{\partial \boldsymbol{\alpha}_i} = \frac{\partial \boldsymbol{S}_b^i}{\partial \widetilde{\boldsymbol{\alpha}}} = \sum_{s\neq\ell_i} 2\widetilde{\boldsymbol{I}}_s^T\widetilde{\boldsymbol{D}}^T(\boldsymbol{D}\boldsymbol{I}_s\boldsymbol{\alpha}_i - \boldsymbol{x}_i) \quad (29)$$

where $\widetilde{\boldsymbol{D}}$ consists of the atoms of $\boldsymbol{D}$ in the index set $\boldsymbol{\Lambda}$. To calculate $\nabla_{\boldsymbol{D}} J_i$, it is still important to compute the derivative $\partial\boldsymbol{\alpha}_i/\partial\boldsymbol{D}$.

Exploiting the same optimization method, the desired gradient can be solved by:

$$\frac{\partial\widetilde{\boldsymbol{\alpha}}}{\partial\widetilde{\boldsymbol{D}}} = (\widetilde{\boldsymbol{D}}^T\boldsymbol{P}\boldsymbol{P}^T\widetilde{\boldsymbol{D}})^{-1}\left(\frac{\partial\widetilde{\boldsymbol{D}}^T\boldsymbol{P}\boldsymbol{P}^T\boldsymbol{x}}{\partial\widetilde{\boldsymbol{D}}} - \frac{\partial\widetilde{\boldsymbol{D}}^T\boldsymbol{P}\boldsymbol{P}^T\widetilde{\boldsymbol{D}}}{\partial\widetilde{\boldsymbol{D}}}\widetilde{\boldsymbol{\alpha}}\right). \quad (30)$$

with the gradient in Eq. (9) calculated, the dictionary update rule is simply:

$$\boldsymbol{D}^{(t)} = \boldsymbol{D}^{(t)} + \gamma\,\nabla_{\boldsymbol{D}} J_i \quad (31)$$

where γ is the step size set as η.

We adopt a standard iterative learning method to jointly learn dictionary $\boldsymbol{D}$ and projection $\boldsymbol{P}$ until the algorithm is convergent. Algorithm 2 summarizes the detailed steps of the proposed method. Fix $\boldsymbol{D}$ to update $\boldsymbol{P}$ by using Algorithm 1. When updating $\boldsymbol{D}$, we fix $\boldsymbol{P}$, and the algorithm is shown in Algorithm 2. Since the optimization problem is non-convex, we can only expect this stochastic gradient procedure to find a local maximum. Still, numerical simulations have shown that the algorithm usually converges to a local maximum in a few iterations. To empirically show the convergence of our method, we take some examples to display the convergence behaviors of ODP-SRC-DL on the AR, Extended Yale B [43], LFWa [44] and UCF 50 action [45] databases.

Algorithm 2 ODP-SRC-DL

Input: Training set $\{\boldsymbol{x}_i\}_{i=1}^n$, parameters λ, η, γ, and initial dictionary $\boldsymbol{D}^{(0)}$

Output: Projection matrix $\boldsymbol{P}$, learned dictionary $\boldsymbol{D}$.

Step 1 (Initialization)

Initialize projection $\boldsymbol{P}^0$, iteration times $t = 1$.

Step 2 (Optimization)

Repeat

Solve $\boldsymbol{D}^{(t)}$ with fixed $\boldsymbol{P}^{(t-1)}$ via Eq. (31)

$\boldsymbol{D}^{(t)} = \boldsymbol{D}^{(t)} + \gamma\,\nabla_{\boldsymbol{D}} J_i$

Project the columns of $\boldsymbol{D}^{(t)}$ onto the unit circle;

Solve $\boldsymbol{P}^{(t)}$ with fixed $\boldsymbol{D}^{(t)}$ by

$\boldsymbol{P}^{(t)} = \boldsymbol{P}^{(t)} + \eta\nabla_{\boldsymbol{P}} J_i$

$t = t + 1$;

until convergence

Step 3 (Output)

Output $\boldsymbol{P} = \boldsymbol{P}^{(t)}$, $\boldsymbol{D} = \boldsymbol{D}^{(t)}$

The curves of the objective function value and the corresponding recognition rates are shown in Fig. 1. It seems that in all experiments our method can achieve stable performance in a few iterations. When the objective function value varies in a flat region, we think the proposed method can obtain a proper and reliable solution and stop the iteration. Also the recognition rate varies only within a small range after several iterations. We note that on the AR and Extended Yale B databases, our method achieves high recognition rate at the initialized iteration. The main reason is that we select a well initial projection matrix for our method, i.e., we use the initial SRC-DP algorithm [25] to initialize the projection matrix $\boldsymbol{P}^{(0)}$. Let us take an example to display the performance of ODP-SRC-DL by using different initializations of $\boldsymbol{P}^{(0)}$. Fig. 2 shows the recognition rates of ODP-SRC-DL under different initialization matrices. We observe that selecting different $\boldsymbol{P}^{(0)}$ result in different performances in the initial iteration. Using the initial SRC-DP algorithm to initialize the projection matrix can achieve high recognition rate. However, after sever iterations, the final recognition results tend to be consistent.

Note That: We need to initialize the dictionary $\boldsymbol{D}^{(0)}$ to learn the sparse coding coefficient $\boldsymbol{\alpha}$ for every training sample. For $\boldsymbol{D}^{(0)}$, we employ the proposed method in [41] to learn the dictionary class by class and then combine the specific-class dictionary as the final dictionary, $\boldsymbol{D}^{(0)} = [\boldsymbol{D}_1^{(0)}, \boldsymbol{D}_2^{(0)}, \cdots, \boldsymbol{D}_c^{(0)}]$. Thus, each dictionary atom is initialized and labeled based on the corresponding class. Furthermore, during the entire learning process, the label of each atom remains fixed, although $\boldsymbol{d}_k$, $k = 1, \cdots, K$ is updated in each iteration [36], where $\boldsymbol{d}_k$ are the dictionary atoms. The main reason for this is that in each iteration step, we only select the dictionary atoms corresponding to the indices $\boldsymbol{\Lambda}$, which as the index set of the nonzero sparse coefficient of $\boldsymbol{\alpha}$. According to the sparse representation theory, most coefficients in $\boldsymbol{\alpha}$

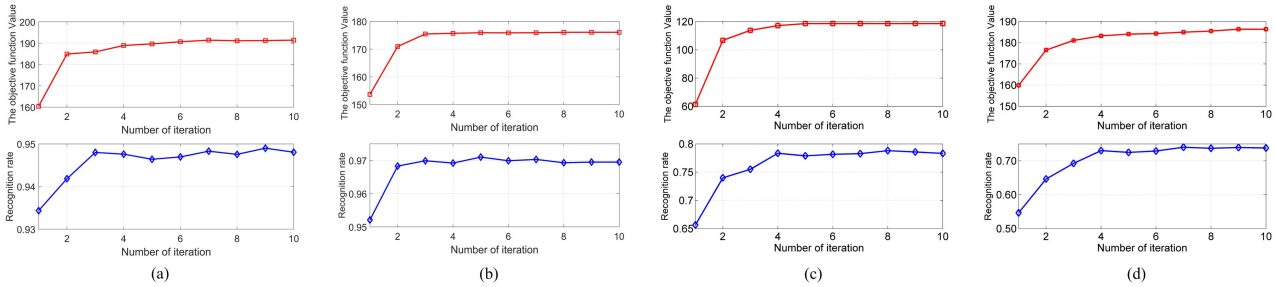


Fig. 1. The top row shows the objective function value vs iterations; the bottom row shows the recognition rate vs iterations in the (a) AR, (b) Extended Yale B, (c) LFWa and (d) UCF 50 databases.

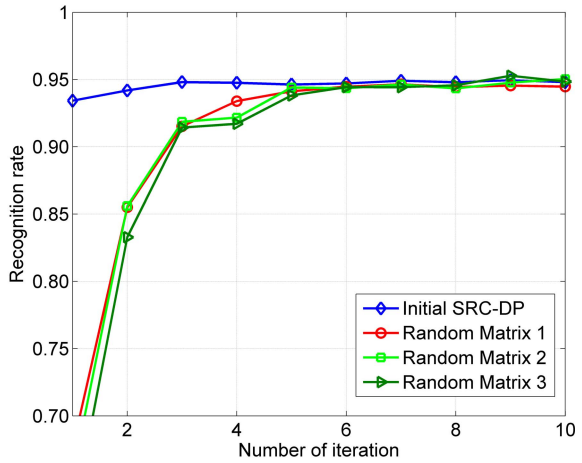


Fig. 2. Recognition rate vs iterations under different initializations of $\boldsymbol{P}^{(0)}$.

are nearly zero, and only the coefficients associated with its corresponding class have significant nonzero entries. Thus, in each iteration, the selected dictionary atoms have the same labels as the selected nonzero coefficients.

On the other hand, our method aims to maximize between-class sparse reconstruction residual and simultaneous minimize the within-class sparse reconstruction residual in the transformed space. This encourages the result that dictionaries associated with the corresponding class can well represent the training samples of this class, and cannot represent training samples of other class well. Hence, the learned dictionary has better discrimination capability.

IV. Experimental Results

In this section, we verify the performance of ODP-SRC in various classification tasks, including face recognition, object recognition, action classification, and scene classification. For face recognition, we adopt three commonly used face databases, including AR [42], Extended Yale B [43] and LFWa [44]. For action recognition, we use the UCF 50 [45] database to evaluation our method. Finally, the Caltech 101 [46] and 15 Scene Categories [48] databases are used for object recognition and scene recognition, respectively. For each database, the average recognition rate is used as the criterion for comparing the performance of different state-of-the-art algorithms.

A. Parameter Setting

There are three parameters in the proposed model: λ, η, and γ. To achieve the best performance, in all the experiments, the sparsity regularization parameter λ in the training and testing phases is determined via five-fold cross-validation. For face recognition, we set $\lambda_{train} = \lambda_{test} = 0.005$ for the AR and Extended Yale B databases, and $\lambda_{train} = 0.05$, $\lambda_{test} = 0.001$ for the LFWa database. For action recognition, we set $\lambda_{train} = 0.001$, $\lambda_{test} = 0.005$ for the UCF50 database. For object recognition, we set $\lambda_{train} = \lambda_{test} = 0.001$ for the Caltech 101 database, and $\lambda_{train} = \lambda_{test} = 0.001$ for the scene experiment. η is the learning rate [36], [49] for updating projection $\boldsymbol{P}$, and we set η as $min(\rho, \rho i_0/i)$, where i represents the subscript of each training sample. In the example $\boldsymbol{x}_i$, $i = 1, 2, \cdots, n$, where n is the total number of training samples. ρ is a constant, selected from $\{10^{-6}, 10^{-5}, \cdots, 10\}$, $i_0 = t/10$, and t is the number of iterations. Similarly, γ is the step size for updating dictionary $\boldsymbol{D}$, and we set $\gamma = \eta$.

B. Face Recognition

1) AR Dataset: We choose a non-occluded subset (14 images per subject) from the AR database, which consists of 1680 face images of 120 subjects (65 males and 55 females). The face portion of each image is manually cropped to 50×45 pixels. To clearly illustrate the advantage of our method, some representative feature extraction algorithms, including SRC-steered discriminative projection (SRC-DP) [25], trace ratio optimization-based SRC-DP (TR-SRC-DP) [27], sparsity preserving projection (SPP) [19], locality preserving projection (LPP) [16], and linear discriminant analysis (LDA) [15] are used for comparison.

We randomly select $n_i = 3, 4, 6$, and 7 samples per subject for training, and test on the rest. Therefore, the total sizes of the training samples are 360, 480, 720, and 840. To make a fair comparison with SRC-DP [25] and TR-SRC-DP [27], and following the experiment settings in [25] and [27], we first perform PCA to reduce the dimension to 200 before implementing SRC-DP, SPP, LPP, LDA, TR-SRC-DP, and the proposed ODP-SRC. For LPP, we set the number of nearest neighbors as $n_i - 1$, where n_i refers to the training samples per subject. Finally, SRC is employed for classification. Each split is repeated 10 times, and we evaluate the performance of these methods according to different dimensions, which vary from 5 to 160 with increments of 5.

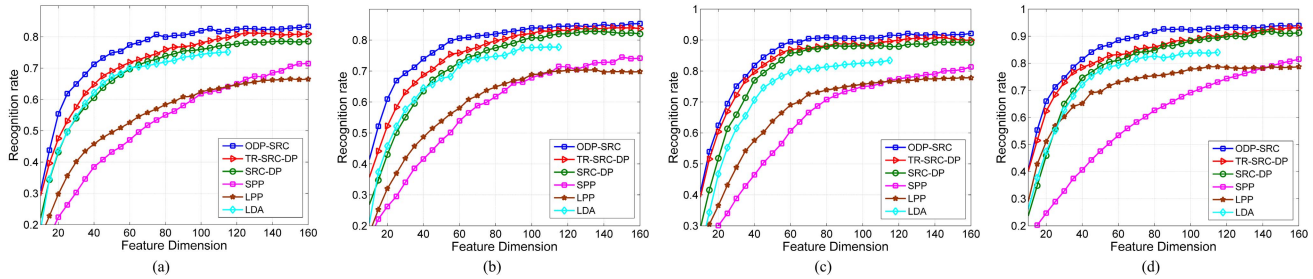


Fig. 3. Recognition rate versus different feature dimensions in the AR database. The number of training samples per class are (a) 3, (b) 4, (c) 6, (d) 7.

TABLE I

RECOGNITION RATES (%) WITH DIFFERENT NUMBER OF TRAINING SAMPLES ON THE AR DATABASE

Methods	$n_i = 3$	$n_i = 4$	$n_i = 6$	$n_i = 7$
ODP-SRC	**83.4(160)**	**85.8(160)**	**92.2(125)**	**94.1(160)**
TR-SRC-DP [27]	81.2(125)	84.0(145)	90.8(145)	93.2(155)
SRC-DP [25]	78.1(145)	82.9(120)	89.3(145)	91.7(145)
SPP [19]	71.5(160)	74.5(150)	81.3(160)	81.6(160)
LPP [16]	66.5(150)	70.4(135)	77.7(150)	79.0(150)
LDA [15]	75.5(119)	78.8(110)	83.6(119)	85.3(119)

Table I lists the maximum recognition rate of each method and the corresponding dimensions. We can see that ODP-SRC obtains the best recognition rate and is consistently better than other feature extraction methods, irrespective of variations in training sample sizes. Consequently the learned features fit SRC well in the transformed subspace, which can improve the performance of SRC.

We also evaluate the performance of related approaches according to different dimensions. Fig. 3 shows the experimental results. It is clear that with the increase of the feature dimensions, the performance of our method and that of other methods also increase, and ODP-SRC always performs the best. Furthermore, when the projected samples are in a relatively low subspace, ODP-SRC performs significantly better than SRC-DP and TR-SRC-DP [27]. This is because by using the bilevel optimization model, the relationship between the sparse representation and the projection is considered in our learning algorithm. Thus, the learned low-dimensional representations can characterize the discriminant structure embedded in high-dimensional data well and have strong discriminant ability.

In order to comprehensively analyze the advantages of the bilevel optimization model, we introduce dictionary learning into our framework, denoted as ODP-SRC-DL. We randomly select 7 samples per class as training samples and evaluate ODP-SRC-DL using different dictionary sizes. ODP-SRC uses the original samples as the dictionary. The recognition results are plotted in Fig. 4. We can see that in all cases, ODP-SRC-DL outperforms ODP-SRC. It is noteworthy that even with the atom number set as 3, the performance of ODP-SRC-DL improves on ODP-SRC by at least 5%.

2) Extended Yale B Dataset: The Extended Yale B face database contains 2414 frontal face images of 38 people. There are about 64 images of each person. The cropped and normalized face images are captured under various

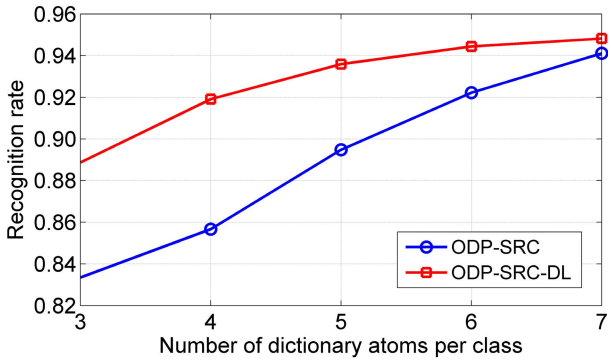


Fig. 4. Recognition rate versus the different numbers of dictionary atoms in the AR database.

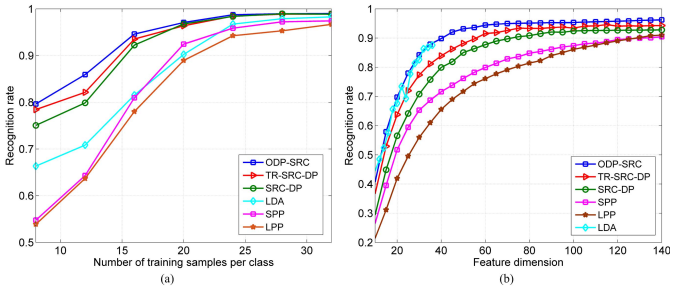


Fig. 5. Recognition rate versus different (a) training sample size, and (b) feature dimensions (16 samples per class).

laboratory-controlled lighting conditions. For computational convenience, these images are resized as 48×42.

Following the experimental set ups of [25], we evaluate our method by using three setups. In the first setup, we use the first 8, 12, 16, 20, 24, 28, 32 images per subject for training, and we test on the rest. LDA, LPP, SPP, SRC-DP, TR-SRC-DP, and the proposed ODP-SRC methods are used for feature extraction. Before implementing the evaluating algorithms, we first use PCA to project all the samples respectively into 100, 120, 140, 160, 180, 200, and 220 dimension spaces according to the number of training samples per class. Finally, SRC is used for classification. Fig. 5(a) illustrates the recognition rate of each method with respect to different samples. It can be seen that ODP-SRC achieves the best result and consistently outperforms TR-SRC-DP, SRC-DP, and other methods, especially when the number of training samples per class is relative small. With the increasing training sample sizes, the performance

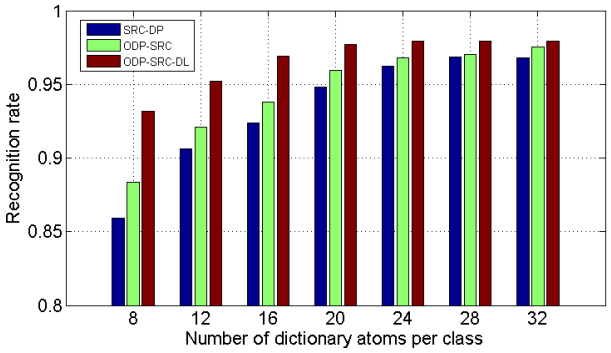


Fig. 6. Recognition rate (%) versus different number of training samples on Extended Yale B database.

of all methods also increase. Overall, our method is still the best. These experimental results are consistent with the results obtained for the AR database.

In the second set up, we evaluate the effect of ODP-SRC using different dimensions when the number of training samples per class is 16. We perform ODP-SRC in the PCA-transformed space. The dimension varies from 5 to 140 with increments of 5. From Fig. 5(b) we can see that our ODP-SRC achieves high recognition rate under different feature subspaces. This indicates that our method is able to find an appropriate low-dimensional representation for SRC.

Finally, we verify the effect of our ODP-SRC-DL method versus different dictionary atoms. We randomly select 32 samples per subject as training data, the remaining samples being the testing data. The dictionary size varies from 8 to 32 with increments of 4. ODP-SRC uses the original training samples as the dictionary, and we randomly select 8, 12, 16, 20, 24, 28, 32 training samples as the dictionary atoms. The average recognition rates of each algorithm are shown in Fig. 6. It can be seen that the ODP-SRC-DL always outperforms ODP-SRC, especially when the dictionary size is small. This means that simultaneously learning the dictionary and projection matrix can further improve classification accuracy.

3) LFW Dataset: LFW [47] is a large-scale database, which contains variations in pose, illumination, expression, misalignment, and occlusion. The aligned labeled face in the wild (LFWa) [44] dataset is an aligned version of LFW. We use 143 subjects with no less than 11 samples per subject (4,174 images in total) in LFWa dataset to perform the experiment. For each person, the first 10 samples are selected as the training samples and the rest are used for testing. A histogram of uniform-LBP is extracted by partitioning a face image into 10×8 patches.

From the experimental results of the AR and Extended Yale B databases, we can see that our feature extraction algorithm is very effective for improving the performance of SRC, and ODP-SRC-DL always outperforms ODP-SRC. In this subsection, in order to further evaluate the performance of ODP-SRC-DL, we compare our method with several of the latest dictionary learning methods, such as Discriminative K-SVD (D-KSVD) [50], Label Consistent K-SVD (LC-KSVD) [36], Fisher discrimination dictionary learning (FDDL) [51], latent dictionary learning (LDL) [52], and the latest bilevel model-based discriminative dictionary learning method (BMDDL) [53], which directly minimizes the classification error in the upper level and uses the sparsity term and the Laplacian term to characterize the intrinsic data structure in the lower level. Furthermore, we also compare our method with SRC [7] as the baseline. The number of dictionary atoms is set as the number of training samples in the same class. We reduce the feature dimension to 1000, and other methods are performed in the PCA-transformed space.

TABLE II
RECOGNITION RATE (%) ON THE LFWA DATABASE

Method	Accuracy	Method	Accuracy
SRC [8]	72.7	LDL [52]	77.2
D-KSVD [50]	65.9	BMDDL [53]	77.7
LC-KSVD [36]	66.0	ODP-SRC	76.2
FDDL [51]	74.8	ODP-SRC-DL	**78.7**

The experimental results are summarized in Table II. We can observe that ODP-SRC achieves at least a 3% improvement compared with SRC. This demonstrates that considering the relationship between sparse representation and the desired projection is crucial for improving the performance of SRC. ODP-SRC-DL also achieves a 2.5% improvement compared with ODP-SRC, indicating that learning a dictionary is meaningful for improving the classification performance. In addition, ODP-SRC-DP outperforms all the dictionary learning methods and results in a 1.0% improvement compared with BMDDL. Although BMDDL also uses the bilevel model to learn a dictionary, it is designed based on a simple linear predictive classifier that ignores the within-class and between-class discriminative information during the learning process. In comparison, our method is designed based on the decision rule of the SRC, not only considering the within-class information, but also considering the between-class relationship. Secondly, BMDDL ignores the importance of feature learning, while our method formulates the discriminative projection and dictionary learning into an optimization framework. Thus the learned features and dictionary are complementary to each other and can capture more discriminative information in the reduced space, which is meaningful in classification.

C. Action Recognition

We use the UCF 50 dataset [45] for action recognition, one of the largest action recognition databases, with 50 action categories, consisting of 6617 realistic videos taken from YouTube, such as Baseball Pitch, Basketball Drumming, Biking, Diving, Tennis Swing, etc.

We directly use the action bank feature vector provided in [59] to evaluate our method and the related methods. Following the common experiment settings in [53] and [59], we reduce the feature dimension to 1500, and the dictionary size is 1500. We take the ODP-SRC-DL through five-fold group-wise cross validation, and D-KSVD [50], LC-KSVD [36], FDDL [51], task-driven dictionary learning (TDDL) [49], BMDDL [53], SRC [8], and other related methods are used for comparison. The comparison results are

TABLE III
RECOGNITION RATE (%) ON THE UCF50 ACTION DATABASE

Method	Accuracy	Method	Accuracy
Sadanand [59]	57.9	FDDL [51]	68.1
Zhang [60]	60.9	LC-KSVD [36]	67.6
Liu [61]	62.7	TDDL [49]	64.8
SRC [8]	62.9	BMDDL [53]	73.2
D-KSVD [50]	65.9	ODP-SRC-DL	**73.8**

TABLE IV
RECOGNITION ACCURACY (%) ON CALTECH 101 DATASET

Method	Accuracy	Method	Accuracy
Lazebink [48]	64.6	SRC [8]	69.3
Germert [54]	64.2	D-KSVD [50]	71.2
Y. Ng [55]	72.6	FDDL [51]	72.6
Malik [56]	56.6	TDDL [49]	71.5
Zhang [57]	73.5	LC-KSVD [36]	72.0
Quan [11]	68.4	BMDDL [53]	75.5
Zhou [58]	75.2	**OPD-SRC-DL**	**76.9**

summarized in Table III. We can see that ODP-SRC-DL outperforms the competing dictionary learning methods. This validates the superiority of ODP-SRC-DL in classifying actions.

D. Object Recognition

The Caltech 101 database [46] includes images of 102 classes (101 common object classes and a background class) collected randomly from the Internet. The number of images in each class varies from 31 to 800 (9,144 images in total). The size of each image is roughly 300 × 300 pixels.

Following the common experimental settings in [25] and [53], we randomly select 30 samples per category for training and test on the rest. For fair comparison, we test our method with spatial pyramid features, as used in [25] and [53]. Since the feature dimension is too high, we reduce the feature dimension to 1500, and other methods use the PCA-reduced features. The dictionary size is 3060.

Table IV summarizes the comparison results, SRC [7], D-KSVD [50], LC-KSVD [36], FDDL [51], BMDDL [53], TDDL [49], and some state-of-the-art algorithms are used for comparison. As Table IV shows, OPD-SRC-DL obtains the best results, improves on the second-best method by around 1.4%. There are two reasons for this improvement. The first reason is that the learned projection matrix and dictionary are jointly learnt, and the former leads to features being able to capture more discriminative information for objects, while the later can encode a compactness coding coefficient in the projected space. In addition, the relationships between the sparse representation and the desired projection and dictionary are considered during the overall learning process; thus, the learned projection and dictionary are suitable for SRC and simultaneously improve its performance.

To further analyze our experiments, Fig. 7 shows the confusion matrix between all classes for the best results (i.e., 76.9%). The classification accuracies of each class are displayed in Fig. 8, where recognition accuracies are sorted in ascending order. There are 9 categories in total, which achieve

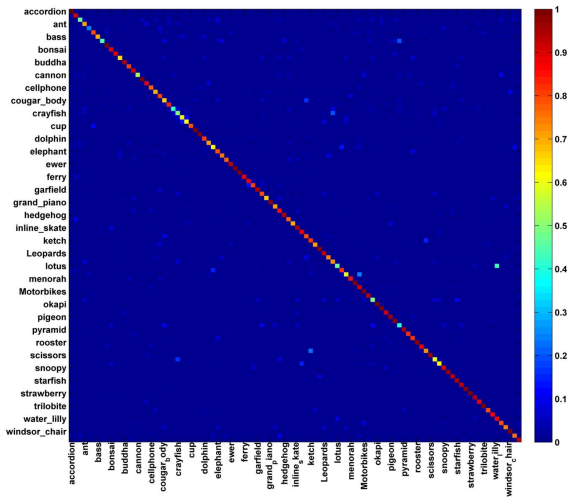


Fig. 7. Confusion matrix on the Caltech 101 dataset.

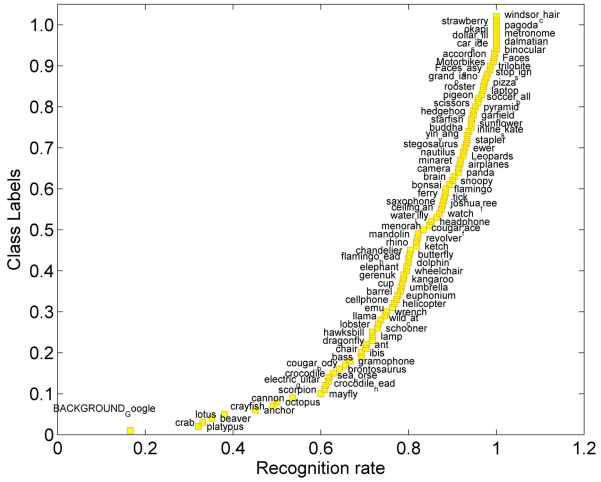


Fig. 8. Recognition rate (%) of each class on the Caltech 101 dataset.

100% accuracy. Fig. 9 shows a few of the easiest and hardest object subjects.

E. Scene Classification

The fifteen scene categories database was first introduced in [48]. Each category contains around 200 to 400 images, and the average image size is around 250 × 400 pixels. This database contains 15 scenes, including kitchen, bedroom, and country scenes. Following the common experimental settings, we use the extracted features provided by [36], and the dimension is reduced to 3000. D-KSVD [50], LC-KSVD [36], FDDL [51], TDDL [49], BMDDL [53], and SRC [8] all use the PCA-transformed features. We randomly select 100 images per category for training and use the remaining samples for testing. The dictionary size is set as 450.

The comparison results are reported in Table V. Our method outperforms all the competing dictionary learning methods and other state-of-the-art methods, and it improves on BMDDL by around 1.3%.

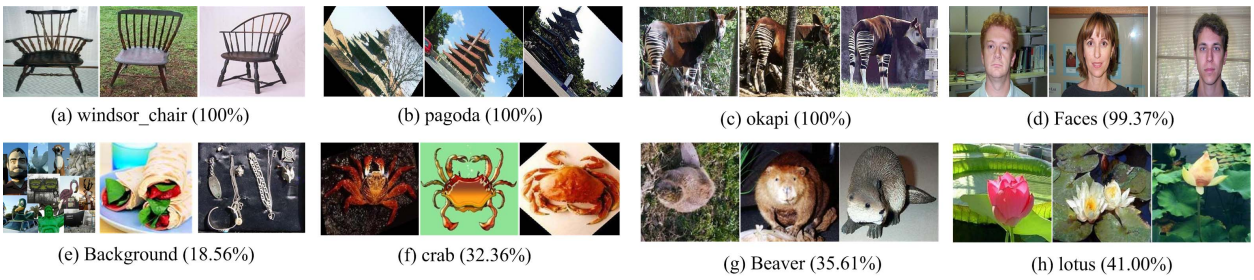


Fig. 9. The top and bottom rows show the classes in which our method performed the best and the worst.

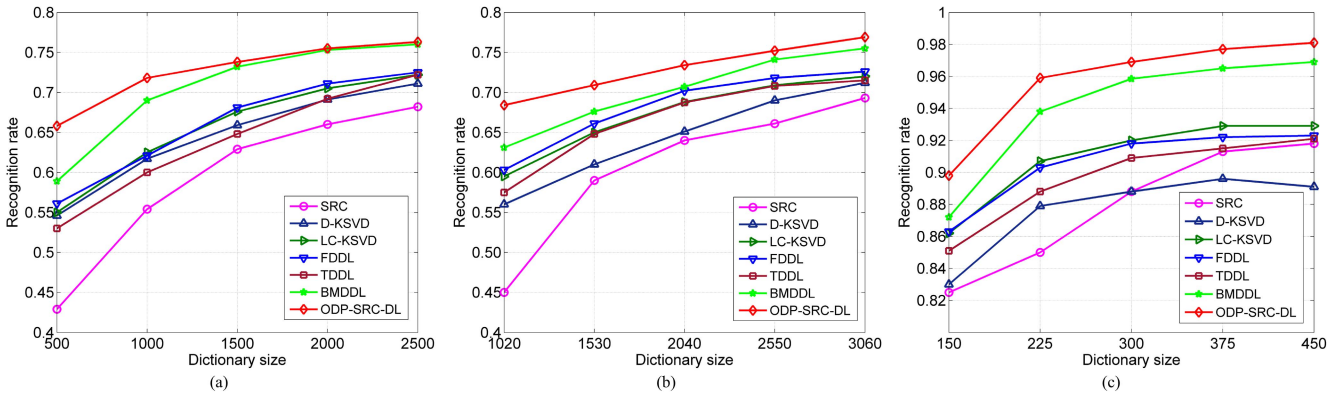


Fig. 10. Recognition rates (%) using different dictionary sizes. (a) UCF50. (b) Caltech 101. (c) 15 Scene Categories.

TABLE V
RECOGNITION RATE (%) ON THE 15 SCENE CATEGORIES DATABASE

Method	Accuracy	Method	Accuracy
Lazebink [48]	81.4	SRC [8]	91.8
Gemert [54]	76.7	D-KSVD [50]	89.1
Yang [5]	80.3	FDDL [51]	92.3
Lian [62]	86.4	LC-KSVD [36]	92.9
Yang [63]	92.9	TDDL [49]	92.1
Wei [64]	91.8	BMDDL [53]	96.9
Song [65]	85.7	ODP-SRC-DL	**98.2**

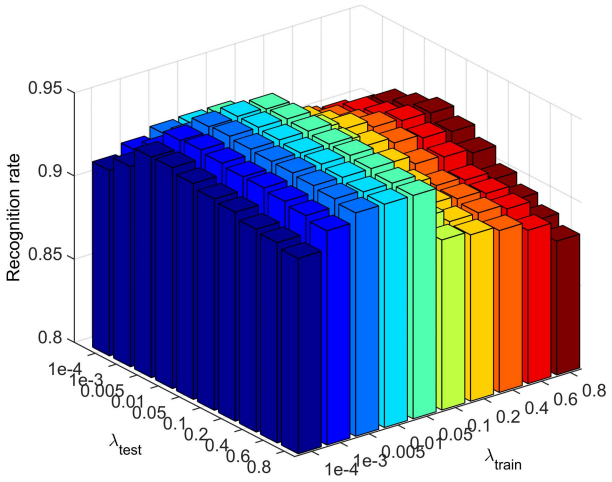


Fig. 11. Recognition rates (%) of ODP-SRC versus different values of λ for the AR database.

F. Parameter Analysis

In this subsection, we take the AR database as an example and conduct experiments to evaluate the effect of our method compared with different sparsity regularization parameters λ. The parameter λ contains two parts: λ_{train} and λ_{test}. When evaluating one parameter, the other is fixed to the values used in the AR recognition experiment. Fig. 11 shows the performance of ODP-SRC versus different λs. We can see that our proposed ODP-SRC achieves a stable performance when λ is set as a suitable range.

We also conduct experiments on the UCF50, Caltech 101, and 15 scene categories databases to evaluate the performance of our method with different dictionary sizes. The experimental results are summarized in Fig. 10. We can see that with different dictionary sizes, our method consistently outperforms the other six competing methods on all three databases. These results clearly demonstrate that OPD-SRC-DL is able to learn a more discriminative dictionary. This indicates that considering the relationship between the sparse representation and the desired dictionary is crucial for improving the performance of SRC.

G. Comparison of Computation Time

In this subsection, we compare the computing time of our method with that of D-KSVD [30], LC-KSVD [36], TDDL [49], BMDDL [53], and SRC [8] for the Extended Yale B and 15 Scene Categories databases. Our hardware configuration is a 3.30 GHz CPU and 8GB of RAM. It should be pointed out that the experimental settings in this subsection are as described in the above subsections. Table VI reports the

TABLE VI
RECOGNITION RATE (%) ON THE 15 SCENE CATEGORIES DATABASE

Method	Extended YaleB		15 Scene Categories	
	Training	Testing	Training	Testing
SRC [8]	-	0.140	-	0.263
D-KSVD [30]	676.4	0.057	7.79×10^3	0.069
LC-KSVD [36]	222.9	0.024	446.5	0.064
TDDL [49]	3.17×10^3	0.028	5.24×10^3	0.054
BMDDL [53]	304.6	0.021	416.7	0.058
ODP-SRC-DL	1.94×10^3	0.027	2.83×10^3	0.061

computing times of the different methods. Note that SRC has no training time, only testing time, and the reported testing time only refers to the time for classifying one sample.

From Table VI, we can see that the proposed method is time-consuming in its training stage (primarily the computing time for learning a projection matrix and a dictionary). However, as for classifying a testing sample, the computing time of ODP-SRC-DL is comparable to other state-of-the-art methods. The testing time of our method is four times faster than SRC. The main reason is that SRC uses all training samples as the dictionary, while ODP-SRC-DL uses the learned dictionary $\boldsymbol{D}$ to calculate the spares representation coefficient vector $\boldsymbol{\alpha}$. Since the number of dictionary atoms is less than the number of training samples, the testing time of our method is faster than SRC. Note that the proposed algorithm ODP-SRC-DL consists of two parts: offline and online computation. The projection and dictionary can be learned offline, and classification is performed online. Thus, the training time does not affect the practical application of our method.

V. CONCLUSION

In this paper, we proposed a bilevel optimization model based discriminative projection learning method (ODP-SRC). By learning a discriminative projection matrix, the extracted features in the transformed space can characterize the discriminant structure of data well and simultaneously fit SRC. The most important point is that by using the bilevel optimization model, the relationship between sparse representation and the desired projection can be expressed explicitly in the objective function. Thus, SRC can achieve optimum performance in a reduced low-dimensional space. Furthermore, our method can be easily extended to learn a dictionary jointly with the projection matrix. This will further improve the classification performance of SRC. Experimental results for many benchmark databases also verify these conclusions.

REFERENCES

[1] M. Yang, L. Zhang, J. Yang, and D. Zhang, "Robust sparse coding for face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 625–632.

[2] C. Bao, Y. Quan, and H. Ji, "A convergent incoherent dictionary learning algorithm for sparse coding," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 302–316.

[3] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep./Oct. 2009, pp. 2272–2279.

[4] N. Kulkarni and B. Li, "Discriminative affine sparse codes for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 1609–1616.

[5] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Aug. 2009, pp. 1794–1801.

[6] Y. Quan, Y. Huang, and H. Ji, "Dynamic texture recognition via orthogonal tensor dictionary learning," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 73–81.

[7] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[8] Y. Zhang, Z. Jiang, and L. S. Davis, "Learning structured low-rank representation for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 676–683.

[9] M. Yang and L. Chen, "Discriminative semi-supervised dictionary learning with entropy regularization for pattern classification," in *Proc. AAAI Conf. Artif. Intell.*, Feb. 2017, pp. 1623–1626.

[10] J. Guo, Y. Guo, X. Kong, M. Zhang, and R. He, "Discriminative analysis dictionary learning," in *Proc. AAAI Conf. Artif. Intell.*, Feb. 2016, pp. 1617–1623.

[11] Y. Quan, Y. Xu, Y. Sun, Y. Huang, and H. Ji, "Sparse coding for classification via discrimination ensemble," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 5839–5847.

[12] W. Huang, F. Sun, L. Cao, D. Zhao, H. Liu, and M. Harandi, "Sparse coding and dictionary learning with linear dynamical systems," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3938–3947.

[13] G. Zhang, H. Sun, F. Porikli, Y. Liu, and Q. Sun, "Optimal couple projections for domain adaptive sparse representation-based classification," *IEEE Trans. Image Process.*, vol. 26, no. 12, pp. 5922–5935, Dec. 2017.

[14] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognit. Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.

[15] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

[16] X. He, S. Yan, Y. Hu, P. Niyogi, and H.-J. Zhang, "Face recognition using Laplacianfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 3, pp. 328–340, Mar. 2005.

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

[18] L. Zhang, M. Yang, Z. Feng, and D. Zhang, "On the dimensionality reduction for sparse representation based face recogniton," in *Proc. Int. Conf. Pattern Recognit.*, 2010, pp. 1237–1240.

[19] L. Qiao, S. Chen, and X. Tan, "Sparsity preserving projections with applications to face recognition," *Pattern Recognit.*, vol. 43, no. 1, pp. 331–341, Jan. 2010.

[20] J. Gui, Z. Sun, W. Jia, R. Hu, Y. Lei, and S. Ji, "Discriminant sparse neighborhood preserving embedding for face recognition," *Pattern Recognit.*, vol. 45, no. 8, pp. 2884–2893, 2012.

[21] G. Zhang, H. Sun, Z. Ji, G. Xia, L. Feng, and Q. Sun, "Kernel dictionary learning based discriminant analysis," *J. Vis. Commun. Image Represent.*, vol. 40, pp. 470–484, Oct. 2016.

[22] G. Zhang, H. Sun, G. Xia, and Q. Sun, "Kernel collaborative representation based dictionary learning and discriminative projection," *Neurocomputing*, vol. 207, pp. 300–309, Sep. 2016.

[23] T. Zhou and D. Tao, "Double shrinking sparse dimension reduction," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 244–257, Jan. 2013.

[24] Z. Feng, M. Yang, L. Zhang, Y. Liu, and D. Zhang, "Joint discriminative dimensionality reduction and dictionary learning for face recognition," *Pattern Recognit.*, vol. 46, no. 8, pp. 2134–2143, 2013.

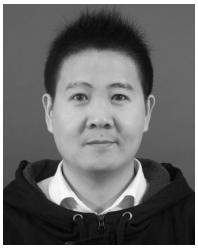
[25] J. Yang, D. Chu, L. Zhang, Y. Xu, and J. Yang, "Sparse representation classifier steered discriminative projection with applications to face recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 7, pp. 1023–1035, Jul. 2013.

[26] Q. Gao, Q. Wang, Y. Huang, X. Gao, X. Hong, and H. Zhang, "Dimensionality reduction by integrating sparse representation and Fisher criterion and its applications," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5684–5695, Dec. 2015.

[27] G. Zhang, H. Sun, G. Xia, and Q. Sun, "Multiple kernel sparse representation based orthogonal discriminative projection and its cost-sensitive extension," *IEEE Trans. Image Process.*, vol. 25, no. 9, pp. 4271–4285, Sep. 2016.

[28] W. Yang, Z. Wang, and C. Sun, "A collaborative representation based projections method for feature extraction," *Pattern Recognit.*, vol. 48, no. 1, pp. 20–27, 2015.

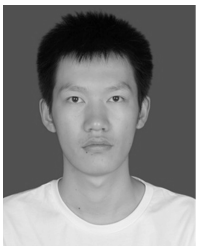
[29] Q. Gao, Y. Huang, H. Zhang, X. Hong, K. Li, and Y. Wang, "Discriminative sparsity preserving projections for image recognition," *Pattern Recognit.*, vol. 48, no. 5, pp. 2543–2553, 2015.
[30] H. Yan and J. Yang, "Sparse discriminative feature selection," *Pattern Recognit.*, vol. 48, no. 5, pp. 1827–1835, 2015.
[31] Y. Xu, Z. Zhong, J. Yang, J. You, and D. Zhang, "A new discriminative sparse representation method for robust face recognition via ℓ_2 regularization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2233–2242, Oct. 2017.
[32] J. Yang, Z. Wang, Z. Lin, X. Shu, and T. Huang, "Bilevel sparse coding for coupled feature spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2360–2367.
[33] B. Colson, P. Marcotte, and G. Savard, "An overview of bilevel optimization," *Ann. Oper. Res.*, vol. 153, no. 1, pp. 235–256, 2007.
[34] J. Yang, K. Yu, and T. Huang, "Supervised translation-invariant sparse coding," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 3517–3524.
[35] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, "Coupled dictionary training for image super-resolution," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3467–3478, Aug. 2012.
[36] Z. Jiang, Z. Lin, and L. S. Davis, "Label consistent K-SVD: Learning a discriminative dictionary for recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2651–2664, Nov. 2013.
[37] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proc. IEEE*, vol. 98, no. 6, pp. 1031–1044, Jun. 2010.
[38] J. A. Bagnell and D. M. Bradley, "Differentiable sparse coding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 113–120.
[39] B. Efron, T. Hastie, I. Johnstone and R. Tibshirani, "Least angle regression," *Annu. Statist.*, vol. 32, no. 2, pp. 407–499, 2004.
[40] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *J. Mach. Learn. Res.*, vol. 11, pp. 19–60, Jan. 2010.
[41] M. Yang, L. Zhang, J. Yang, and D. Zhang, "Metaface learning for sparse representation based face recognition," in *Proc. IEEE Conf. Image Process.*, Sep. 2010, pp. 1601–1604.
[42] A. M. Martinez and R. Benavente, "The AR face database," Univ. Barcelona, Barcelona, Spain, CVC Tech. Rep. 24, Jun. 1998.
[43] K.-C. Lee, J. Ho, and D. J. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 684–698, May 2005.
[44] L. Wolf, T. Hassner, and Y. Taigman, "Similarity scores based on background samples," in *Proc. Asian Conf. Comput. Vis.*, Sep. 2009, pp. 88–97.
[45] K.-K. Reddy and M. Shah, "Recognizing 50 human action categories of web videos," *Mach. Vis. Appl.*, vol. 24, no. 5, pp. 971–981, 2013.
[46] F. F. Li, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories," *Comput. Vis. Image Understand.*, vol. 106, no. 1, pp. 59–70, 2007.
[47] G. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Comput. Sci., Univ. Massachusettss, Amherst, Amherst, MA, USA, Tech. Rep. 07–49, 2007.
[48] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognition natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2006, pp. 2169–2178.
[49] J. Mairal, F. Bach, and J. Ponce, "Task-driven dictionary learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 791–804, Apr. 2012.
[50] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2691–2698.
[51] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Fisher discrimination dictionary learning for sparse representation," in *Proc. 13th Int. Conf. Comput. Vis.*, Nov. 2011, pp. 543–550.
[52] M. Yang, D. Dai, L. Shen, and L. V. Gool, "Latent dictionary learning for sparse representation based classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 4138–4145.
[53] P. Zhou, C. Zhang, and Z. Lin, "Bilevel model-based discriminative dictionary learning for recognition," *IEEE Trans. Image Process.*, vol. 26, no. 3, pp. 1173–1187, Mar. 2017.
[54] J. Gemert, J.-M. Geusebroek, C. J. Veenman, and A. W. M. Smeulder, "Kernel codebooks for scene categorization," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 696–709.
[55] C. Adam and A. Y. Ng, "The importance of encoding versus training with sparse coding and vector quantization," in *Proc. Int. Conf. Mach. Learn.*, 2011, pp. 921–928.
[56] S. Maji, A. C. Berg, and J. Malik, "Efficient classification for additive kernel SVMs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 66–77, Jan. 2013.
[57] C. Zhang, J. Cheng, J. Liu, J. Pang, Q. Huang, and Q. Tian, "Beyond explicit codebook generation: Visual representation using implicitly transferred codebooks," in *Proc. IEEE Conf. Image Process.*, Dec. 2015, vol. 24, no. 12, pp. 5777–5788.
[58] P. Zhou, Z. Lin, and C. Zhang, "Integrated low-rank-based discriminative feature learning for recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 5, pp. 1080–1093, May 2016.
[59] S. Sadanand and J. J. Corso, "Action bank: A high-level representation of activity in video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1234–1241.
[60] B. Zhang, A. Perina, V. Murino, and A. D. Bue, "Sparse representation classification with manifold constraints transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4557–4565.
[61] Q. Liu and C. Liu, "A novel locally linear KNN model for visual recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1329–1337.
[62] X.-C. Lian, Z. Li, B.-L. Lu, and L. Zhang, "Max-margin dictionary learning for multiclass image categorization," in *Proc. Eur. Conf. Comput. Vis. Pattern Recognit.*, Sep. 2010, pp. 157–170.
[63] S. Yang and D. Ramanan, "Multi-scale recognition with DAG-CNNs," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2015, pp. 1215–1223.
[64] X.-S. Wei, B.-B. Gao, and J. Wu, "Deep spatial pyramid ensemble for cultural event recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop (ICCVW)*, Dec. 2015, pp. 280–286.
[65] X. Song, S. Jiang, and L. Herranz, "Joint multi-feature spatial context for scene recognition on the semantic manifold," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1312–1320.



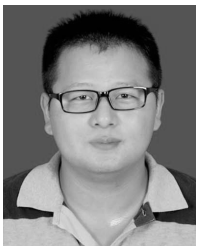
Guoqing Zhang received the B.Sc. and M.Sc. degrees in information engineering from Yangzhou University, Yangzhou, China, in 2009 and 2012, respectively, and the Ph.D. degree in pattern recognition and intelligent system from the Nanjing University of Science and Technology, Nanjing, China, in 2017. He is currently an Assistant Professor with the College of Computer and Software, Nanjing University of Information Science and Technology, Nanjing. His main research fields are image processing and computer vision, multimedia analysis, and pattern recognition.

Huaijiang Sun received the B.S. and Ph.D. degrees from the School of Marine Engineering, Northwestern Polytechnical University, Xian, China, in 1990 and 1995, respectively. He is currently a Professor with the Department of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include computer vision and pattern recognition, image and video processing, and intelligent information processing.

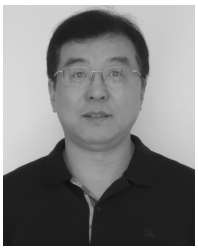
Yuhui Zheng received the B.Sc. degree in pharmacy engineering and the Ph.D. degree in pattern recognition and intelligent system from the Nanjing University of Science and Technology, Nanjing, China, in 2004 and 2009, respectively. He is currently an Associate Professor with the College of Computer and Software, Nanjing University of Information Science and Technology, Nanjing. His research interests are multimedia data analysis and processing, image and video segmentation, and computer vision.



Guiyu Xia received the B.S. and Ph.D. degrees in software engineering from the Nanjing University of Science and Technology, Nanjing, China, in 2012 and 2017, respectively. He is currently a Faculty Member with the School of Automation, Nanjing University of Information Science and Technology, Nanjing. His research interests include pattern recognition, machine learning, and human motion capture data reusing.



Lei Feng received the Ph.D. degree from the Nanjing University of Science and Technology, in 2017. He is currently a Lecturer with the Jinling Institute of Technology. His research interests include pattern recognition and compressive sensing.



Quansen Sun received the Ph.D. degree in pattern recognition and intelligence system from the Nanjing University of Science and Technology, Nanjing, China, in 2006, where he is currently a Professor with the School of Computer Science and Engineering. His research interests include pattern recognition, image processing, computer vision, and data fusion.