Robust Latent Subspace Learning for Image Classification

Xiaozhao Fang, *Member, IEEE*, Shaohua Teng, Zhihui Lai, Zhaoshui He, Shengli Xie, *Senior Member, IEEE*, and Wai Keung Wong

***Abstract*— This paper proposes a novel method, called robust latent subspace learning (RLSL), for image classification. We formulate an RLSL problem as a joint optimization problem over both the latent SL and classification model parameter predication, which simultaneously minimizes: 1) the regression loss between the learned data representation and objective outputs and 2) the reconstruction error between the learned data representation and original inputs. The latent subspace can be used as a bridge that is expected to seamlessly connect the origin visual features and their class labels and hence improve the overall prediction performance. RLSL combines feature learning with classification so that the learned data representation in the latent subspace is more discriminative for classification. To learn a robust latent subspace, we use a sparse item to compensate error, which helps suppress the interference of noise via weakening its response during regression. An efficient optimization algorithm is designed to solve the proposed optimization problem. To validate the effectiveness of the proposed RLSL method, we conduct experiments on diverse databases and encouraging recognition results are achieved compared with many state-of-the-arts methods.**

***Index Terms*— Classification, computer vision, data representation, linear regression (LR), subspace learning (SL).**

I. Introduction

IN THE fields of machine learning and computer vision, it is very important to choose an appropriate data representation for data analysis, since an appropriate representation can uncover the underlying explanatory factors behind the observed data and facilitate the sequent model learning [1]–[4]. For example, Li and Fu [1] proposed a supervised regularization-based robust subspace approach via low-rank learning. By introducing linear discriminant analysis (LDA), the new representation is more discriminative. Gu *e*t al. [2] proposed a projective dictionary pair learning method for dictionary learning in which the representation coefficients were used as the new data representation. Kim *e*t al. [4] proposed an elastic-net regularization-based low-rank matrix factorization method for new data representation. Although model learning from features extracted from raw inputs achieves encouraging performance for many classification tasks, it is still necessary to learn a more compact and discriminative data representation for more accurate classification. In this paper, we focus on learning an appropriate data representation by uncovering a latent subspace for image classification.

Subspace learning (SL) has been widely applied in data representation [5]–[7]. According to whether or not the label is available for training samples, SL methods can be categorized into three groups, supervised SL methods, semisupervised SL methods, and unsupervised SL methods. By using labels, supervised SL methods learn a discriminative projection for discriminant analysis. The representative methods include LDA [8]–[9], sparse tensor discriminant analysis [10], local LDA [11], locality sensitive discriminant analysis [12], and local discriminant embedding [13]. In these methods, Fisher's criterion [14] is used to seek a discriminative projection, which simultaneously maximizes the distances among the means of the classes and minimizes the distances among the samples from the same class. However, the cost of collecting high-quality labeled training samples for training is costly. In real-world applications, abundant unlabeled data are often easily accessible. And these unlabeled data are useful to enhance the algorithmic performance. Semisupervised SL methods can simultaneously use both the labeled and unlabeled samples to enhance algorithmic performance [15], [16]. Semisupervised discriminant analysis [17] was proposed to learn a discriminative projection for dimensionality reduction by using the labeled samples to maximize the margins between different classes and unlabeled samples to estimate the intrinsic geometric structure of the data. Flexible manifold embedding [18] uses the label information of labeled samples and the manifold structure of both labeled and unlabeled samples to perform semisupervised clustering and dimension reduction. By exploring the local relationship between data point and its neighbors, unsupervised SL methods can preserve the intrinsic manifold structure of the data. Representative works include locally linear embedding [19], Laplacian eigenmap [20], neighbor preserving embedding [21], and locality preserving projec-

Manuscript received March 29, 2016; revised August 23, 2016 and February 13, 2017; accepted April 4, 2017. Date of publication May 10, 2017; date of current version May 15, 2018. This work was supported in part by the Natural Science Foundation of China under Grant 61573248, Grant 61203376, Grant 61375012, and Grant 61573248, in part by the Hong Kong Polytechnic University under Project Code: 1-YW1C, and in part by the Shandong Provincial Natural Science Foundation, under Grant ZR2013FL016. *(Corresponding author: Zhihui Lai.)*

X. Fang and S. Teng are with the School of Computer Science and Technology, Guangdong University of Technology, Guangzhou 510006, China (e-mail: xzhfang168@126.com; shteng@gdut.edu.cn).

Z. Lai is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518055, China (e-mail: lai_zhi_hui@163.com).

Z. He and S. Xie are with the School of Automation, Guangdong University of Technology, Guangzhou 510006, China (e-mail: zhaoshuihe@gmail.com; shlxie@gdut.edu.cn).

W. K. Wong is with Institute of Textiles and Clothing, The Hong Kong Polytechnic University, Hong Kong, and The Hong Kong Polytechnic University Shenzhen Research Institute, Shenzhen 518055, China (e-mail: calvin.wong@polyu.edu.hk).

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

Digital Object Identifier 10.1109/TNNLS.2017.2693221

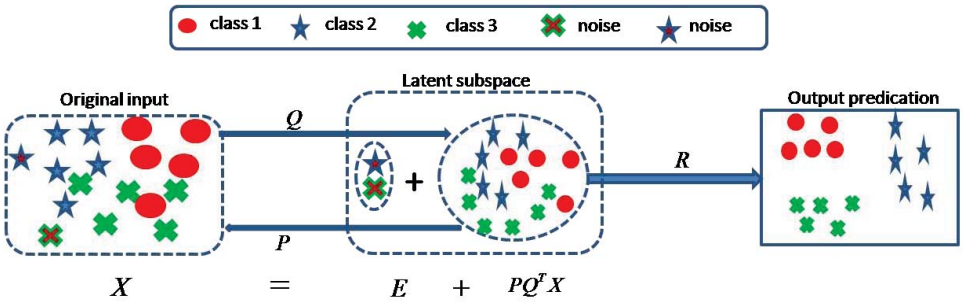


Fig. 1. Framework of our method. Different colors in the cell mean different classes. Matrix Q is used to learn the latent subspace where the original input X is transformed into two components: 1) a "clean" component PQ^TX and 2) noise E. Matrix P is used to perform a flexible data reconstruction. Matrix R is used to learn a classifier. Please note that only the "clear" data representation Q^TX is selected to learn the classifier parameter R.

tions [22]. The principal components analysis (PCA) [23] projects high-dimensional data into a lower dimensional space by seeking the direction of maximum variance for the optimal data reconstruction. Yan *et* al. [24] further reformulated some SL methods into a unified graph embedding framework, in which the desired geometric structures of data are encoded as the graph relationship.

Nonetheless, conventional SL methods only focus on the original visual features of data, which are independent of the follow-up tasks [25]. Thus, these methods may not obtain an overall optimality in algorithmic performance in some sense. A critical challenge is to build a bridge between the original visual features and objective outputs so that they are seamlessly connected. To this end, in this paper, a novel robust latent SL (RLSL) method is proposed to learn a robust latent subspace, which can be used as an intermediate between the original visual features and objective outputs. With the learned subspace, we choose an appropriate data representation for more accurate classification. Specifically, RLSL learns an appropriate data representation by simultaneously considering the minimizations of the regression loss and reconstruction error, which makes that the uncovered data representation will predict the labels and hold the main information of data, respectively. In our proposed method, the data representation and the classification are integrated into a unified optimization step, which makes that the data representation is closely related to classification so that the learned data representation is discriminative. Noise is ubiquitous in realistic data. Such noise may contaminate the data representation and consequently degrade the performance of classification. In order to learn a robust data representation, we decompose the original data into a "clean" component plus sparse noise component so that the negative influence of noise can be reduced. Fig. 1 shows the framework of the proposed method. We propose an effective and efficient iterative algorithm to solve the resulting optimization problem. And the effectiveness of RLSL is illustrated on different image classification tasks.

Our key contributions are summarized as follows.

1) We propose a novel data representation method by jointly optimizing the latent SL and classification model parameter prediction. In this way, the uncovered data representation is more compact and discriminative for the follow-up classification tasks.
2) Unlike conventional data reconstruction way which uses a single matrix to perform reconstruction such as PCA, our method uses a more flexible way (see Fig. 1), i.e., uses two different matrices for better reconstruction. In this way, one of the matrices can provide more freedom to guarantee that the uncovered data representation holds the main energy and thus is competent to perform classification. By introducing a sparse item to fit error, our algorithm is robust to different type noises.
3) We develop an efficient algorithm based on alternating direction method of multipliers (ADMM) to solve the resulting optimization problem. The theoretical and empirical analyses demonstrate the effectiveness of the designed optimization algorithm.

The remainder of this paper is arranged as follows. We begin with a review of the related works in Section II. In Section III, we elaborate our proposed formulation followed with its optimization algorithm and computation complexity and convergence analysis in Section IV. The effectiveness of the proposed method is experimentally verified in Section V. Finally, we summarize this paper and give the conclusion in Section VI.

II. Related Work

In this section, we briefly review the related works. We first give the objective function of linear regression (LR) method, since it is used to fit class labels in our method. For a collection of n training samples represented as a matrix $X = [x_1, x_2, \ldots, x_n] \in \Re^{m\times n}$, the objection function of LR is as follows:

$$\min_A \|Y - AX\|_F^2 + \lambda\|A\|_F^2 \tag{1}$$

where $Y = [y_1, y_2, \ldots, y_n] \in \Re^{c\times n}$ ($c \geq 2$ is the number of classes) is the corresponding binary label matrix and $A \in \Re^{c\times m}$ is the transformation matrix. Y is defined as follows: for each training sample x_i ($i = 1, 2, \ldots n$), $y_i \in \Re^c$ is its label vector. If x_i is from the kth class ($k = 1, 2, \ldots, c$), then only the kth entry of y_i is one and all the other entries are zero. For classification, margins between different classes are expected to be as large as possible after original data are transformed into their label space [i.e., Y in (1)]. To this end, many variants of LR have been proposed for classification by enlarging the margins [26]–[30].

Since our method aims to learn an appropriate data representation for image classification by uncovering a latent

subspace, we also view some representation-based methods. Representation-based methods have been shown enormous success in many recognition tasks. For example, LR classification (LRC) [31] and collaborative representation-based classification (CRC) with regularized least square [32] were proposed for face recognition. Robust regression that aims to find a "clean" data representation was proposed for classification and regression [33]. Sparse representation-based classification (SRC) finds the smallest number of training samples to represent a test sample and uses the representation results to perform classification [3]. It is reported that SRC obtains surprisingly experimental results in face recognition. Dictionary learning that transforms the original data representation into a discriminative and compact representation for recognition has been shown to produce state-of-the-art results [34]–[36]. Low-rank representation methods were also proposed to learn a robust data representation for data classification and clustering [37]–[41]. For example, robust PCA (RPCA) [38] decomposes original data into a low-rank component and a sparse component in which the low-rank component is commonly used as the "clean" data representation of original data and the sparse component is used as the error. Latent low-rank representation (LatLRR) [37] decomposes the original observation data into three components: principal feature representation, salient feature representation, and sparse noise in which the salient feature representation is discriminative, and thus, it can be used for discriminant analysis. The relations among many low-rank representation-based methods were discussed in [42]. Structured-constrained low-rank representation (SC-LRR) was proposed to find an appropriate data representation for subspace clustering [43].

Most these mentioned methods consist of two steps: 1) feature learning (data representation) and 2) classification is performed based on the learned features. Such two independent steps cannot guarantee an overall optimality in performance of a designed algorithm. In other words, the learned data representation may not be optimal for classification. Moreover, these mentioned methods only explore the original visual features of data to perform learning tasks, whereas the original visual features and the objective outputs are not linked. Due to the existing of so-called semantic gap, the original visual features cannot guarantee good classification results. Some works were proposed to uncover the latent discriminative subspace by learning a transformation [44], [45]. However, different from previous works, our method jointly optimizes problems of the latent SL and classification model parameter prediction, which can guarantee the overall optimality in algorithmic performance. A latent subspace is uncovered and used as an intermediate between the original visual features and objective outputs. In this way, we can uncover a more appropriate data representation for classification tasks. With two different matrices, our method can reveal data structure well and obtain good classification results.

III. Proposed Method

In this section, we introduce a novel latent SL method, which can learn an appropriate data representation and classification model parameter simultaneously.

Let us first introduce our notation. Let $X \in \Re^{m\times n}$ be a matrix containing n samples possibly corrupted by noise or outliers, where m is the dimensionality of samples. The goal of classification task is to accurately classify the samples to their respective classes. The definitions of binary label matrix Y and transformation matrix A are the same as those in (1). We consider the Frobenius norm of matrix A: $\|A\|_F^2 = \sum_{i=1}^{m}\sum_{j=1}^{n} A_{ij}^2 = Tr[A^T A]$. A^T denotes the transposed matrix of A. The ℓ_1-norm for A is defined as

$$\|A\|_1 = \sum_{i=1}^{m}\sum_{j=1}^{n} |A_{ij}|. \tag{2}$$

A. Motivation of RLSL

How to obtain a good data representation is an important issue in classification. However, image data contain much redundant information, and the discriminative information is not decided by all the pixels. As discussed in Section II, this paper aims to obtain a discriminative data representation by learning a latent space to enhance the classification performance and the robustness of algorithm. Recent studies [1], [33], [43], [45], [58] indicate that to obtain a good classification results always requires to find a compact and discriminative data representation. In [1], [33], [43], and [58], "clear" data representations were learned by using low-rank decomposition. However, these methods obtain the "clean" data by simple substraction, i.e., $X_{\text{clean}} = X - E$. In this way, the dimensionality of "clean" data representation is as the same as that of the original data. In general, the combinations of features are more discriminative than individual feature [41], [45]. Thus, obtaining a combination of features is a good choice for improving classification accuracy. To obtain a compact data representation, we expect that the dimensionality of "clean" data representation is lower than that of the original data representation. In [45], a latent subspace was learned to preserve the intrinsic geometric structure of data. However, the important information may be lost, since it does not perform the data reconstruction. Finding a compact and discriminant latent data representation and simultaneously avoiding the information loss seem to be a good choice to connect the original visual features and objective outputs. Thus, a natural idea is combine the original features to form an appropriate data representation and simultaneously learn the classification model parameters so that the learned data representation is discriminative.

B. Model of RLSL

To learn the new data representation, we present a formulation to learn an appropriate data representation $F \in \Re^{d\times n}$ in a latent subspace as follows:

$$F = Q^T X \tag{3}$$

where $Q \in \Re^{m\times d}$ (d denotes the dimensionality of latent space) is the linear transformation matrix which is used to link the latent and original input spaces.

During the training phase of classification, the obtained features $Q^T X$ are fed into a classifier $f(x, R)$ to learn the classification model parameter $R \in \Re^{c\times d}$, where c is the

number of classes. We aim at optimizing the transformation Q by minimizing the classification error. In this way, the obtained data representation $Q^T X$ is tightly coupled with classification. Our objective function for learning transformation Q and parameter R of classifier can be defined as

$$\min_{Q,R} \sum_{i=1}^{n} \psi(y_i, f(Q^T x_i, R)) + \lambda_3 \|R\|_F^2 \tag{4}$$

where $x_i \in \Re^m$ is the ith sample in $X \in \Re^{m\times n}$. R is the parameter of classifier $f(x, R)$. ψ is the regression loss function. y_i is the label vector of the ith sample. $\lambda_3 \geq 0$ is a regularization parameter. In this paper, a linear classifier $f(x, R) = Rx$ and a quadratic loss function, i.e., the LR, are used for model learning. We define the classification model as

$$\min_{Q,R} \|Y - RQ^T X\|_F^2 + \lambda_1 \|Q\|_F^2 + \lambda_3 \|R\|_F^2. \tag{5}$$

As discussed in Section III, some representation-based methods and SL methods ignore the information loss [35]–[38]. In this case, it is inevitable that the performance of classification is degraded. In our method, we consider the reconstructive ability of the obtained data representation. We therefore propose to impose the following constraint on the objective function:

$$\begin{aligned} &\min_{Q,R,P} \|Y - RQ^T X\|_F^2 + \lambda_1 \|Q\|_F^2 + \lambda_3 \|R\|_F^2 \\ &\text{s.t.} \quad X = PQ^T X, \quad P^T P = I \end{aligned} \tag{6}$$

where we impose the orthogonal constraint on $P \in \Re^{m\times d}$ to avoid a trivial solution. Different from the reconstruction ways of some conventional SL methods which use a single matrix to perform data reconstruction such as PCA, we use two different matrices P and Q to perform the data reconstruction (see Fig. 1). In doing so, Q has more freedom to fit labels so that a more accurate R can be obtained. P also has more freedom to make the representation $Q^T X$ hold the main energy of the "clean" data. Since outliers or noise-free component are unknown, so existing methods use X in the estimation of A. In presence of outliers or noise, the obtained result is a biased estimation of A. In our method, to obtain a "clean" data representation $Q^T X$ from the original input X, we explicitly factorize X into $PQ^T X + E$ and only compute R using the "clean" $Q^T X$. Thus, RLSL solves the following optimization problem:

$$\begin{aligned} &\min_{Q,R,P,E} \frac{1}{2}\|Y - RQ^T X\|_F^2 \\ &\qquad + \frac{1}{2}\lambda_1 \|Q\|_F^2 + \lambda_2 \|E\|_1 + \frac{1}{2}\lambda_3 \|R\|_F^2 \\ &\text{s.t.} \quad X = PQ^T X + E, \quad P^T P = I \end{aligned} \tag{7}$$

where $\lambda_1 \geq 0$ and $\lambda_2 \geq 0$ are scalars that weight the corresponding terms in (7). RLSL explicitly avoids transform the error term E into the output space by learning the regression parameter R only from the augmented "clean" data representation $Q^T X$. Observe that the first and third regularization terms control the complexity of this model. The second term encourages $E \in \Re^{m\times n}$ to be sparse. By jointly learning latent subspace Q and classification model parameter R, the obtained data representation $Q^T X$ is discriminative for classification.

IV. Optimization

In this section, we propose to use ADMM to solve problem (7).

A. Solving Problem (7)

To facilitate the optimization of (7), we rewrite the problem (7) as following by defining $A = RQ^T$:

$$\begin{aligned} &\min_{Q,R,E,A,P} \frac{1}{2}\|Y - AX\|_F^2 + \frac{1}{2}\lambda_1 \|Q\|_F^2 + \lambda_2 \|E\|_1 \\ &\qquad + \frac{1}{2}\lambda_3 \|R\|_F^2 \\ &\text{s.t.} \quad X = PQ^T X + E, \quad P^T P = I, \quad RQ^T = A. \end{aligned} \tag{8}$$

The augmented Lagrangian function of problem (8) is

$$\begin{aligned} \mathcal{J} = \arg \min_{Q,R,E,A,P,Y_1,Y_2} \; & \frac{1}{2}\|Y - AX\|_F^2 + \frac{1}{2}\lambda_1 \|Q\|_F^2 \\ & + \lambda_2 \|E\|_1 + \frac{1}{2}\lambda_3 \|R\|_F^2 \\ & + \langle Y_1, X - PQ^T X - E\rangle \\ & + \langle Y_2, RQ^T - A\rangle \\ & + \frac{\mu}{2}\big(\|X - PQ^T X - E\|_F^2 \\ & \quad + \|RQ^T - A\|_F^2\big) \\ \text{s.t.} \; & P^T P = I \end{aligned} \tag{9}$$

where Y_1 and Y_2 are Lagrange multipliers and $\mu > 0$ is a penalty parameter. The variables are updated alternately by minimizing the Lagrangian function, with other variables fixed. The iteration stops when the convergence conditions are met. We provide the details of solving (9) in the following.

Step 1 *(Update A):* Updating A by solving the following problem:

$$\mathcal{J} = \arg\min_A \frac{1}{2}\|Y - AX\|_F^2 + \frac{\mu}{2}\left\|RQ^T - A + \frac{Y_2}{\mu}\right\|_F^2. \tag{10}$$

By setting the derivative $(\partial \mathcal{J}/\partial A) = 0$, we obtain

$$\begin{aligned} &A(XX^T + \mu I) - \left(YX^T + \mu\left(RQ^T + \frac{Y_2}{\mu}\right)\right) = 0 \\ &\Rightarrow A = \left(YX^T + \mu\left(RQ^T + \frac{Y_2}{\mu}\right)\right)(XX^T + \mu I)^{-1}. \end{aligned} \tag{11}$$

Step 2 *(Update P):* Update P by solving the following problem:

$$\begin{aligned} &\mathcal{J} = \arg\min_P \left\|X - PQ^T X - E + \frac{Y_1}{\mu}\right\|_F^2 \\ &\text{s.t.} \quad P^T P = I. \end{aligned} \tag{12}$$

Let $M = X - E + (Y_1/\mu)$. The problem (12) is a classic orthogonal procrustes problem [41] which is solved as follows: first compute the singular value decomposition (SVD) of $m \times d$ matrix $MX^T Q$ as $MX^T Q = USV^T$ and then let $P = UV^T$.

Step 3 *(Update Q):* Update Q by solving the following problem:

$$\mathcal{J} = \arg\min_{Q} \frac{1}{2}\lambda_1 \|Q\|_F^2 + \frac{\mu}{2}\left\|X - PQ^TX - E + \frac{Y_1}{\mu}\right\|_F^2 + \frac{\mu}{2}\left\|RQ^T - A + \frac{Y_2}{\mu}\right\|_F^2. \quad (13)$$

By setting the derivative $(\partial\mathcal{J}/\partial Q) = 0$, we obtain

$$Q(\lambda_1 I + \mu R^T R) + \mu XX^T Q - (\mu XM^T P + \mu B^T R) = 0 \quad (14)$$

where $B = A - (Y_2)/(\mu)$. Q is essentially updated by solving a Sylvester equation.

Step 4 *(Update R):* Update R by solving the following problem:

$$\mathcal{J} = \arg\min_{R} \frac{1}{2}\lambda_3 \|R\|_F^2 + \frac{\mu}{2}\left\|RQ^T - A + \frac{Y_2}{\mu}\right\|_F^2. \quad (15)$$

By setting the derivative $(\partial\mathcal{J}/\partial R) = 0$, we obtain

$$\lambda_3 R + \mu RQ^TQ - \mu HQ = 0 \Rightarrow R = \mu HQ(\lambda_3 I + \mu Q^TQ)^{-1} \quad (16)$$

where $H = A - (Y_2/\mu)$.

Step 5 *(Update E):* Update E by solving the following problem:

$$\mathcal{J} = \arg\min_{E} \lambda_2 \|E\|_1 + \frac{\mu}{2}\left\|X - PQ^TX - E + \frac{Y_1}{\mu}\right\|_F^2 \quad (17)$$

which has the following closed-form solution [46], [47]:

$$E = \mathcal{S}_{\frac{\lambda_2}{\mu}}\left[X - PQ^TX + \frac{Y_1}{\mu}\right] \quad (18)$$

where $S_{(\lambda_2/\mu)}[t] = \text{sign}(t)\max(|t| - (\lambda_2/\mu), 0)$.

Step 6 *Update Y_1, Y_2 and μ*:

$$\begin{cases} Y_1 = Y_1 + \mu(X - PQ^TX - E) \\ Y_2 = Y_2 + \mu(RQ^T - A) \\ \mu = \min(\mu_{\max}, \rho\mu). \end{cases} \quad (19)$$

The complete algorithm is outlined in Algorithm 1.

Algorithm 1 Solving (7)

Input: Training samples matrix X; Label matrix Y; Parameter λ_1, λ_2, λ_3 and d;
Initialization: $A = \mathbf{0}$; $Q = \mathbf{0}$; $R = \mathbf{0}$; $E = \mathbf{0}$; $Y_1 = \mathbf{0}$; $Y_2 = \mathbf{0}$; $\mu_{\max} = 10^5$; $\rho = 1.01$; $\mu = 0.1$.
while not converged **do**
1. Update P by solving (12).
2. Update A by solving (10).
3. Update Q by solving (13).
4. Update R by solving (15).
5. Update E by solving (17).
6. Update Y_1, Y_2 and μ by
$$\begin{cases} Y_1 \leftarrow Y_1 + \mu(X - PQ^TX - E) \\ Y_2 \leftarrow Y_2 + \mu(RQ^T - A) \\ \mu \leftarrow \min\{\rho\mu, \mu_{\max}\} \end{cases}$$

end while
Output: Transformation matrix A

B. Computational Complexity Analysis

This section gives the computational complexity analysis for Algorithm 1.

For Algorithm 1, the major computation is SVD decomposition and solving Sylvester equation problem in steps (2) and (3), respectively. We will now discuss each part in detail. First, the complexity of computing A is $\mathcal{O}(cnm + cdm + m^2c + m^3)$. Second, the complexity of SVD decomposition of the $m \times d$ matrix is $\mathcal{O}(md\kappa)$, and thus, the complexity of computing P is $\mathcal{O}(\max(m^2n, m^2d) + md\kappa)$. The complexity of computing Q is about $\mathcal{O}(m^3 + \max(m^2n, m^2d) + m^2n + d^2c)$. Finally, the complexity of computing R is $\mathcal{O}(d^3 + d^2m + \max(mcd, d^2c))$. In practical, we find that the dimension of the latent space is equal to the number of classes. For simplicity, we assume that $m \geq n \gg \max(d, c)$. Thus, the total computational complexity of Algorithm 1 is about $\mathcal{O}\iota(m^2c + 2m^3 + 3m^2n)$, where ι is the number of iterations.

C. Convergence Analysis

In this section, we analyze the convergence property of the proposed optimization algorithm. Since the overall model in (7) is nonconvex, it is difficult to guarantee its convergence to a local minimum. However, an empirical evidence suggests that the proposed algorithm has a good convergence behavior (see Fig. 9). We propose a proof of weak convergence of the proposed algorithm by showing that under mild conditions, any limit point of the iteration sequence generated by the algorithm is a stationary point that satisfies the Karush–Kuhn–Tucker (KKT) conditions [4]. It is worth providing that any converging point must be a point that satisfies the KKT conditions, because they are necessary condition to be a local optimal solution. This result provides an assurance about the convergence behavior of the proposed algorithm.

Let us assume that the proposed algorithm reaches a stationary point. The KKT conditions for (8) are derived as follows (please note that the procedure of solving P does not involve in the Lagrange multipliers, and thus, we do not proof the KKT condition for it):

$$\begin{aligned} X - PQ^TX - E = 0, \quad RQ^T - A = 0 \\ \frac{\partial\mathcal{L}}{\partial Q} = \lambda_1 Q - XY_1^TP + Y_2^TR = 0 \\ \frac{\partial\mathcal{L}}{\partial R} = \lambda_3 R + Y_2 Q = 0 \\ \frac{\partial\mathcal{L}}{\partial A} = -YX^T + AXX^T - Y_2 = 0 \\ Y_1 \in \lambda_2 \partial_E ||E||_1. \end{aligned} \quad (20)$$

We can obtain the following equation from the last relationship in (20):

$$X - PQ^TX + \frac{Y_1}{\mu} \in X - PQ^TX + \lambda_2 \frac{\partial_E ||X - PQ^TX||_1}{\mu} \triangleq \mathcal{Q}_{\frac{\lambda_2}{\mu}}(X - PQ^TX) \quad (21)$$

where scalar function $\mathcal{Q}_{(\lambda_2/\mu)}(t) \triangleq t + (\lambda_2/\mu)\partial|t|$ is applied elementwise to $X - PQ^TX$. From [4], we obtain the following

−1.3496	−0.3015	0.5120	0.3821	0.5325	0.6280	−0.3285
−1.3902	−0.0642	−0.4506	0.1109	−0.1982	0.6221	0.0019
−1.0054	−0.1680	−0.3279	−0.2350	0.3079	−0.2816	−0.5620
1.0965	0.0940	0.1991	−0.2186	−0.5789	0.1173	−0.1243
0.9036	0.0905	1.1727	−1.0257	−0.0965	0.7737	0.4493
1.0272	0.1082	−0.4726	−0.0804	0.1431	0.0835	−0.6440
−0.0656	0.7922	0.8948	0.2654	0.3606	0.0158	0.5108
−0.5522	1.1189	0.2452	0.1323	0.0193	0.7094	−0.1921
0.0564	0.7034	−0.4465	−0.1660	0.1716	0.4460	−0.3487

(a)

−1.3451	−0.3012	0.5058	0.3820	0.5290	0.6210	−0.2267
−1.3910	−0.0643	−0.4229	0.1108	−0.1899	0.6300	0.0015
−1.0156	−0.1688	−0.3478	−0.2409	0.3078	−0.2789	−0.5672
1.0657	0.0944	0.1990	−0.2233	−0.5495	0.1179	−0.1235
0.9124	0.0973	1.1842	−1.0365	−0.0977	0.7763	0.4869
0.9975	0.1258	−0.4835	−0.1102	0.1530	0.0688	−0.5801
−0.0655	0.7967	0.8894	0.2630	0.3548	0.0136	0.4956
−0.5712	1.1155	0.2456	0.1432	0.0222	0.7182	−0.1921
0.0268	0.7005	−0.4885	−0.1837	0.1553	0.4502	−0.2376

(b)

−1.4331	−0.2208	0.3876	0.2293	0.5145	0.5801	−0.0477
−1.3088	−0.0643	−0.3912	0.1656	−0.0484	0.6309	−0.2959
−0.8515	−0.3283	−0.3724	0.2829	0.1439	0.0214	−0.1861
1.0233	0.1507	0.0812	−0.4602	−0.4141	0.0341	0.0881
0.9430	0.0963	1.2216	−0.9171	−0.1268	0.7813	0.3490
0.7172	0.0580	−0.3218	−0.1737	−0.3669	−0.2300	−0.0278
−0.0209	0.8028	0.8711	0.2130	0.4389	0.0781	0.2977
−0.6005	1.0464	0.2836	0.1479	−0.0329	0.6687	−0.0027
−0.0885	0.5635	−0.1907	0.0164	−0.2766	0.2781	−0.0878

(c)

Fig. 2. Original data and different reconstruction data. (a) Original data X. (b) Reconstruction data by PQ^TX. (c) Reconstruction data by QQ^TX.

relation:

$$E = \mathcal{Q}_{\frac{\lambda_2}{\mu}}^{-1}\left(X - PQ^TX + \frac{Y_1}{\mu}\right) \approxeq \mathcal{S}\left(X - PQ^TX + \frac{Y_1}{\mu}, \frac{\lambda_2}{\mu}\right) \tag{22}$$

where $\mathcal{S}(x,\tau) = \text{sign}(x)\max(|x| - \tau, 0)$. Therefore, the KKT condition is as follows:

$$\begin{aligned} &X - PQ^TX - E = 0, \quad RQ^T - A = 0 \\ &\lambda_1 Q - XY_1^TP + Y_2^TR = 0\lambda_3, \quad R + Y_2Q = 0 \\ &-YX^T + AXX^T - Y_2 = 0 \\ &E = \mathcal{S}\left(X - PQ^TX + \frac{Y_1}{\mu}, \frac{\lambda_2}{\mu}\right). \end{aligned} \tag{23}$$

Based on these conditions, we prove that algorithm converges to a point that satisfies the KKT condition.

Theorem 1: Let $\theta \triangleq (Q, R, A, E, P, Y_1, Y_2)$ and $\{\theta\}_{j=1}^{\infty}$ be generated by Algorithm 1 and suppose $\{\theta\}_{j=1}^{\infty}$ is bounded and $\lim_{j\to\infty}\{\theta^{j+1} - \theta^j\} = 0$. Then, every limit point of $\{\theta^j\}_{j=1}^{\infty}$ satisfies the KKT conditions. In particular, whenever $\{\theta\}_{j=1}^{\infty}$ converges, it converges to a KKT point.

The detailed proof of Theorem 1 is moved to the Appendix for the better flow of this paper.

D. Classification

When (7) is solved, we obtain the regression parameter A. Then, we directly use A to obtain the transformation results of training and testing samples, respectively. Suppose x_{test} is the test sample, its label is assigned as $j^* = \arg\max_j(Ax_{\text{test}})$ $(j = 1, 2\ldots, c)$. In practice, we find that the results obtained by using the above method and the nearest-neighbor (NN) classifier are generally similar. Thus, our method uses NN classifier to perform classification for the sake of simplicity.

V. Experiments

In this section, we present extensive experiments to validate the effectiveness of our method for different classification tasks. The first experiment, respectively, gives the visualizations of matrices P and Q. This is useful to shows that whether matrices P and Q work better in data reconstruction. The second experiment reports the comparisons of our method against the state-of-the-art classification methods on different types of databases. The third experiment is to test the algorithmic robustness to different types of noise. The fourth experiment is to study the algorithmic convergence and parameters sensitiveness. The code of this paper can be downloaded from http://www.scholat.com/laizhihui.

A. Example of Data Reconstruction

In this paper, we use two different matrices P and Q to reconstruct the data. In order to demonstrate this issue, we use a toy data set to test the reconstruction ability of different ways. The original data (nine samples in total) is shown in Fig. 2(a) which is from three classes and each class has three samples. The goal of our experiments is only to show that whether two different matrices P and Q work better than a single matrix Q in data reconstruction. Thus, the original data shown in Fig. 2(a) are relatively clean (the value of E is very small), which can provide a clear comparison between the original data and different reconstruction data. Please note that when we obtain the best classification accuracies, the reconstruction data are then calculated. The first experiment is to test the reconstruction ability of two different matrices P and Q (i.e., our method) in which the reconstruction data are calculated as PQ^TX. The reconstruction data are shown in Fig. 2(b). The second experiment first solves the following optimization problem:

$$\begin{aligned} &\min_{Q,R,E} \frac{1}{2}\|Y - RQ^TX\|_F^2 + \lambda_1\|E\|_1 + \frac{1}{2}\lambda_2\|R\|_F^2 \\ &\text{s.t.} \quad X = QQ^TX + E, \quad Q^TQ = I \end{aligned} \tag{24}$$

and then use the obtained Q to reconstruct data by using QQ^TX. The reconstruction result is shown in Fig. 2(c).

From the reconstruction results, we can see that the reconstruction result obtained by PQ^TX is better than that of QQ^TX. Thus, two different matrices form is better than a single matrix form in data reconstruction and they can guarantee that the obtained data representation Q^TX can hold the main energy of the data.

B. Experiments on the Different Data Sets

We first evaluate our method on three widely used face databases: 1) Extended Yale B [48], [37]; 2) CMU PIE [36], [49]; and 3) AR [35], [41], [42], [50]. Note that the difficulties of these face databases are not the same. As shown in Fig. 3, Extended Yale B is more simple than the other two databases. For each individual, it has about 64 near front images under different illuminations. For AR, it is more challenging to recognize them, since it contains different facial expressions, illuminations conditions, and occlusions of sun glass and scarf. The CMU PIE database is taken under different poses, expressions, and illuminations, which makes this database more difficult to classification. We also test our method on two more different types of databases: 1) Caltech 101 database for object recognition [35], [51] and 2) Fifteen Scene Categories for scene recognition [35], [36], [52].

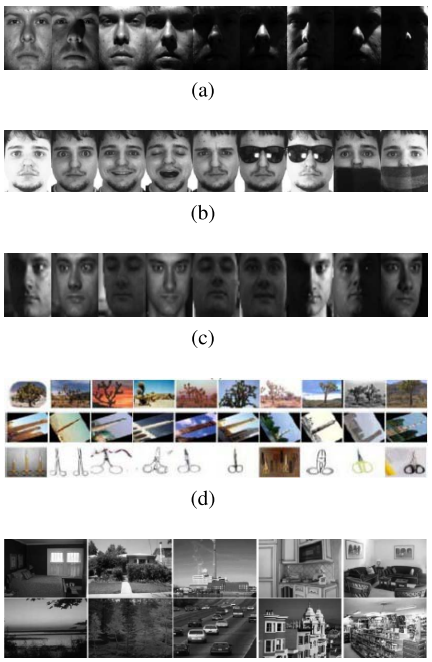


Fig. 3. Some images from (a) Extended YaleB, (b) AR, (c) CMU PIE, (d) Caltech101, and (e) Fifteen scene databases.

TABLE I

CLASSIFICATION ACCURACIES (%) OF DIFFERENT METHODS ON THE EXTENDED YALE B DATABASE

Alg.	10	15	20	25
SRC [3]	87.8±0.3	92.6±0.6	94.4±0.6	96.7±0.5
CRC [32]	86.1±0.5	90.7±0.3	93.0±0.2	94.1±0.3
LLC [25]	79.8±0.4	88.6±0.3	91.5±0.4	94.3±0.5
LRC [31]	83.3±0.4	89.4±0.5	92.4±0.2	93.6±0.3
LRSIC [29]	87.0±0.6	92.7±0.5	94.2±0.3	96.1±0.5
LRRC [28]	84.3±0.6	91.5±0.4	93.3±0.5	95.8±0.7
SLRRC [28]	85.5±0.4	91.4±0.6	94.0±0.5	95.6±0.7
TDDL [34]	84.3±0.2	88.9±0.3	92.5±0.4	95.0±0.6
Robust PCA [38]	86.1±0.2	90.5±0.4	93.5±0.6	95.4±0.3
LatLRR [37]	84.0±0.5	88.8±0.3	92.1±0.5	93.8±0.6
SVM [5]	81.5±1.4	89.2±1.0	92.6±0.7	94.5±0.6
ILRDFL [58]	86.8±0.7	91.3±0.6	93.9±0.8	95.5±0.6
SC-LRR [43]	85.6±0.4	88.7±0.8	92.8±0.3	94.5±0.8
LR	86.6±1.1	92.2±0.9	94.1±1.2	96.6±0.8
Our method	**89.0±1.3**	**93.7±0.8**	**95.2±0.6**	**97.4±0.5**

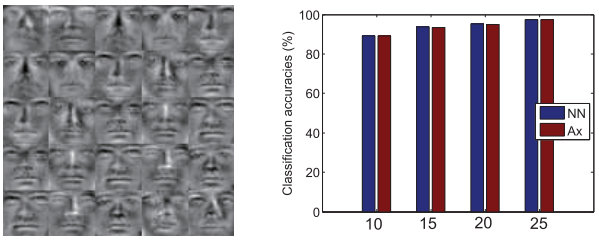
Fig. 4. Some examples of using our method to correct the errors in the Extended Yale B databases. Left: original data matrix X. Center: corrected data PQ^TX. Right: error E.

We compare our method with LR [see (1)], SRC [3], CRC [32], the locality constrained linear coding (LLC) [25], LRC [31], LRSIC [29], LRRC [28], SLRRC [28], TDDL [34], SVM [5], RPCA [38], LatLRR [37], SC-LRR [43], and ILRDFL [58]. For fairness, for RPCA and SC-LRR, first noises are removed by using respective methods for training samples and then uses the "clean" training samples to learn the model parameter of LR (1). Finally, the learned model parameters are used to extract the discriminative features for training and testing samples, respectively and use an NN classifier to classify testing samples. For LatLRR, the salient features are used to learn the model parameter of LR (1). Then, the classification process is as the same as that of RPCA. The platform is MATLAB 2010b under Window 7 on PC equipped with a 3.30-GHZ CPU and 8-GB memory.

1) Face Recognition: We compare our method with some state-of-the-arts methods on three different face image data sets.

a) Extended Yale B: The Extended Yale B contains 2414 images with 38 peoples and each people provides 59–64 images. Every image has 32×32 pixels. We randomly select 10, 15, 20, and 25 samples per person for training and the remaining samples are used for testing. Every experiment runs 30 times. When we evaluate SRC, CRC, LRC, and LRSIC, all training samples are used as the dictionary. The number of neighbors of LLC is set to 5, which is the same as that in [28]. Following [28], the dictionary size for LRRC, SLRRC, and TDDL is set to 140, i.e., each person has five atoms. The experimental results are shown in Table I. Note that we report the mean classification results (mean $\pm$ std) for all methods in this experiment. From the results in Table I, we can see that with a different number of training samples our method always achieve the best recognition results. An "abnormal" finding is that the classification results of RPCA are better than those of many dictionary learning methods, such as TDDL, LRRC, and SLRRC when we select few samples as training samples. The reason may be that: 1) when we remove noises, model (1) may be more effective than these classification models used in TDDL, LRRC, and SLRRC for improving classification accuracy and 2) when we use few training samples, we cannot learn a discriminative dictionary. Thus, the parameters of finally classification models used in these dictionary learning methods are not optimal. However, we also see that TDDL, LRRC, and SLRRC outperform RPCA when we use more training samples for model learning. This reason may be that when we use more training samples, these dictionaries learned by TDDL, LRRC, and SLRRC contain more abundant and discriminative information so that the learned coding coefficients are more discriminative. Similar cases happen in the subsequent experiments.

We randomly select some images from the Extended YaleB database to visualize RLSL's effectiveness in error correction. Fig. 4 shows some results produced by RLSL. It is worth to note that, for a given face image data matrix X, RLSL decomposes it into a "clean" part PQ^TX and a sparse error part E fitting noise. Seeing from the decomposable face images in Fig. 4, the shadow on these original face images can be effectively removed and used as the error part. It is also worth noting that the "error" term E contains the salient features that represents some key local parts of face images, e.g., the eyes and noses. This implies that it is possible to use RLSL to extract the salient regions, as done in face



(a) (b)

Fig. 5. (a) Visualization of basis vectors of matrix Q. (b) Classification results obtained by two different classifiers.

recognition and saliency detection [47]. Fig. 5(a) shows the basis vectors of matrix Q learned by RLSL on the Extended Yale B database. Each basis vector has the dimension of 1024. We plot these basis vectors as 32×32 gray-scale images. It is clear to see that the basis vectors of Q contain lots of details of face images. Fig. 5(b) shows the classification results by two different classifiers, i.e., NN and Ax [Ax is the classifier with the classification rule $j^* = \arg\max_j (Ax_{\text{test}})$], which indicates that the results obtained by these two classifiers are almost similar. Please note that the experiment is just only to verify that the use of NN classifier in our method is advisable.

b) CMU PIE: The CMU PIE database contains 41 368 images from 68 persons, each with 13 different poses, 43 different illuminations conditions, and 4 different expressions. In this experiment, we select a subset that contains five near front poses (C05, C07, C09, C27, and C29) and all the images are taken under different illuminations and expressions, to test different methods. Thus, there are 170 images for each person. Since LLC encodes the scale-invariant feature transform (SIFT), we should keep a certain amount of SIFT features. Thus, the face images are normalized to size of 64×64 pixels for LLC. In other methods, all images are simply cropped into 32×32. All the training samples are used as the dictionary for SRC, CRC, LRC, and LRSIC. We set the size of dictionary to 340 for LRRC, SLRRC, and TDDL. We also randomly select 10, 15, 20, and 25 samples per person for training and the remaining samples are used for testing. Every experiment runs 30 times and mean classification results (mean±std) are reported in Table II. We see that our method still obtains the best classification results. Especially, when the size of training samples is small, the improvement of classification accuracy of our method is more obvious than all the other methods.

c) AR: The AR database contains over 4000 color face images from 126 people (56 women and 70 men). Each person provides 26 images taken during two sessions. In each session, each person has 13 images, in which three images with sunglasses, another three with scarfs, and the remaining seven with facial expressions and illumination conditions. Following standard evaluation procedure [3] and [35], we use a subset consisting 2600 images from 50 male and 50 female. For each person, we randomly select 20 images for training and the remaining are used for testing. Each image is projected onto a 540-D vector with a randomly generated matrix [35]. The experiment results are shown in Table III. Please note

TABLE II

CLASSIFICATION ACCURACIES (%) OF DIFFERENT METHODS ON THE CMU PIE DATABASE

Alg.	10	15	20	25
SRC [3]	76.4±0.3	88.1±0.2	90.2±0.5	93.4±0.2
CRC [32]	83.8±0.4	88.3±0.3	91.0±0.6	93.2±0.3
LLC [25]	77.4±0.4	84.6±0.5	89.9±0.2	93.2±0.3
LRC [31]	75.9±0.7	85.0±0.4	90.5±0.5	92.3±0.4
LRSIC [29]	82.5±0.5	87.5±0.5	90.6±0.3	93.2±0.6
LRRC [28]	79.8±0.2	85.5±0.4	90.1±0.3	91.0±0.5
SLRRC [28]	80.8±0.3	86.7±0.1	89.6±0.3	91.8±0.2
TDDL [34]	78.8±0.5	85.5±0.4	88.7±0.4	91.4±0.5
Robust PCA [38]	81.1±0.3	84.2±0.4	88.1±0.2	91.0±0.1
LatLRR [37]	80.4±0.6	86.7±0.3	90.0±0.4	91.3±0.6
SVM [5]	79.1±0.9	86.9±0.6	90.5±0.4	92.8±0.5
LR	86.1±1.5	89.5±1.3	92.0±0.8	93.2±0.7
ILRDFL [58]	84.7±0.6	90.3±0.5	93.2±0.6	94.1±0.7
SC-LRR [43]	86.9±0.4	90.2±0.6	92.6±0.7	94.0±0.5
Our method	**88.7±0.4**	**92.7±0.3**	**94.5±0.2**	**95.5±0.3**

that some experiment results are directly cited from [35]. We report the mean classification results over 30 runs for our method. Our method achieves the best classification results and outperforms all the other methods.

As [28] and [29] did, we also consider the following three scenarios for three specific recognition tasks.

1) *Sunglasses:* For this specific recognition task, we select seven neutral images and one image with occlusion of sunglass from session 1 as the training set and select seven neutral images from session 2 and five images with sunglass images, in which two are the rest images with sunglass from session 1 and three from session 2 as the testing set.
2) *Scarf:* For this specific recognition task, we select seven unobscured images and one image with scarf from session 1 as the training set and use the remaining images (from sessions 1 and 2) as the testing set.
3) *Sunglasses and Scarf (Mixed):* For this specific recognition task, we select seven neutral images plus one with sunglasses and one with scarf from session 1 as the training set and choose the rest images in both sessions 1 and 2 as the testing set. Specifically, we use 9 images for training and 17 images for testing.

These experiments are repeated three times and we report the average experimental results for our method in Table IV. Our method again achieves the best classification results. For the scenario of sunglasses, scarf, and mixed, the classification accuracies of our method are best, about 0.6%, 1.2%, and 0.5% higher than their second best method ILRDFL, respectively. We also note that compared with other methods, RPCA earns a good position in classification accuracy, since it can effectively remove different types of noise.

2) Object Recognition: We use the Caltech 101 database to test our method for object recognition. The Caltech 101 is a widely used database for object recognition, which contains a total of 9146 images, split between 101 distinct objects (including faces, watches, pianos, and ants) and a background category. Therefore, this database has 102 categories in total. Each object category contains about 31–800 images. The size of each image is roughly 300 × 200 pixels.

TABLE III
CLASSIFICATION ACCURACIES (%) OF DIFFERENT METHODS ON THE AR DATABASE

Alg.	Accuracy	Alg.	Accuracy
LLC (30 local bases)[25]	69.5	SRC (all train. samp.)[3]	97.5
LLC* (70 local bases)[25]	88.7	SRC* (5 per person)[3]	66.5
K-SVD (5 per person)[54]	86.5	CRC [32]	97.3
D-SVD (5 per person)[55]	88.8	LRC [31]	94.5
LC-KSVD1 (5 per person)[35]	92.5	SVM [5]	96.7
LC-KSVD2 (5 per person)[35]	93.7	LatLRR [37]	97.6
LC-KSVD2 (all train.samp)[35]	97.8	Robust PCA [38]	97.8
LR	97.7	Our method	**98.8±0.5**

TABLE IV
CLASSIFICATION ACCURACIES (%) OF DIFFERENT METHODS ON THE AR DATABASE WITH THREE SPECIFIC RECOGNITION TASKS

Scenario.	Sunglasses	Scarf	Mixed
SRC [3]	88.6	85.6	83.2
CRC [32]	90.0	87.1	86.9
LLC [25]	87.1	85.8	84.1
LRC [31]	84.7	78.6	81.3
LRSIC [29]	87.2	79.5	83.5
LRRC [28]	86.1	83.4	82.7
SLRRC [28]	89.0	85.3	84.8
TDDL [34]	83.6	83.3	82.2
Robust PCA [38]	88.1	86.6	86.5
LatLRR [37]	87.1	86.3	84.3
SC-LRR [43]	89.4	90.5	91.5
ILRDFL [58]	93.6	92.9	93.3
Our method	**94.2±0.3**	**94.1±0.5**	**93.8±0.6**

TABLE V
CLASSIFICATION ACCURACIES (%) OF DIFFERENT METHODS ON THE CALTECH 101 DATABASE

Alg.	Accuracy	Alg.	Accuracy
SRC[3]	69.3	TDDL [34]	68.1
CRC [32]	71.5	LatLRR [37]	63.5
LLC [25]	64.8	Robust PCA [38]	68.4
LLC* [25]	70.8	Lazebnik [52]	64.6
LRC [31]	69.1	SVM [5]	72.9
LRSIC [29]	70.7	Yang [64]	73.2
LRRC [28]	70.1	Geusebroek [62]	64.1
SLRRC [28]	71.0	Y. Ng [60]	72.6
LC-KSVD1 [35]	73.4	Malik [53]	56.6
LC-KSVD2 [35]	73.6	ILRDFL [58]	73.8
Gemert [56]	64.1	Our method	**76.1±0.4**
SC-LRR [43]	72.4		

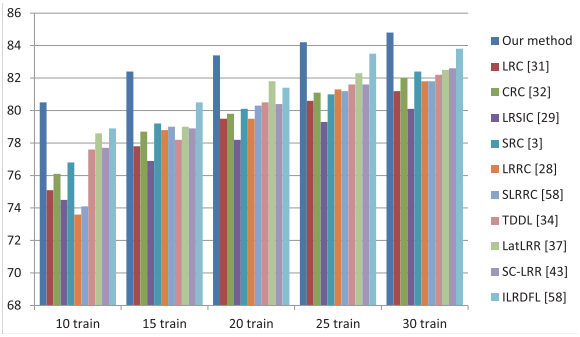


Fig. 6. Classification accuracies (%) on the deep learning features of Caltech 101 database, in which x-axis represents the different number of training samples and y-axis denotes the classification accuracy (%).

Following [28] and [35], we also test our method using spatial pyramid features. The process of generating the features is the same as that in [35]. Since the dimension of original features is too high, PCA is used to reduce the dimension to 1500. In this experiment, we randomly select 30 samples per category as training set and use the remaining samples as testing set. As [25] did, LLC is the original LLC that uses sparse coding to encode SIFT descriptors [35]. While LLC* uses sparse coding to encode the spatial pyramid features. For fairness, SRC, CRC, LRC, LRSIC, LRRC, SLRRC, LatLRR, and RPCA and our method all use the spatial pyramid features. The size of dictionaries of SRC, CRC, LRSIC, LRRC, SLRRC, TDDL, LC-KSVD1, and LC-KSVD2 is set to 3060, i.e., each category has 30 dictionary items. The neighborhood size of LLC and LLC* is set to 30. We also report the mean classification results (mean±std) over 30 runs for our method. The experiment results are shown in Table V. As the experimental results shows, our method performs the best among all the compared methods and at least 2.3% higher than the runner-up, ILRDFL. We also note that when we evaluate our method, the total of 17 classes in the Caltech 101 database achieve 100% classification accuracies, respectively.

In order to test our method better, we conduct the experiments on the deep learning feature. We select the features of decaf fc-6 of Caltech 101 database that are available at https://sites.google.com/site/crossdataset/home/files [59]. Since the dimension of original feature is very high, we use PCA as a preprocessing step to preserve 98% energy of the data. We randomly select 10, 15, 20, 25, and 30 samples per class for training and remaining samples for testing, and we report the mean classification results over 10 random splits. Fig. 6 shows the mean classification accuracy (%) of different methods in which our method obtains the best classification accuracies. This indicates that our method has good applicability to all kinds of features.

3) Scene Classification: We test our method on the Fifteen Scene Categories database, which contains 15 natural scene categories that expands on the 13-category database released in [35]. It contains 4485 images falling into 15 categories, such as kitchens, bedrooms, streets, and country scenes. Each category has 200–400 images. The feature data of this database can be available at http://www.umiacs.umd.edu/∼zhuolin/projectlcksvd.html. The features are computed as follows. First, computing a spatial pyramid feature with a four-level spatial pyramid and an SIFT-descriptor codebook with a size of 200, and then, PCA is used to reduce the features dimension to 3000. As [52] did, we randomly select 100 images per category as training samples and use the remaining as the testing samples. SRC, CRC, LRC, LRSIC, SLRRC, LRRC, DLSR, LatLRR, RPCA, and our method all use the spatial pyramid features provided in [40]. The dictionary size of SRC, CRC, LRSIC, LRRC, SLRRC, and

TABLE VI

CLASSIFICATION ACCURACIES (%) OF DIFFERENT METHODS ON THE FIFTEEN SCENE DATABASE

Alg.	Accuracy	Alg.	Accuracy
SRC[3]	91.8	TDDL [34]	92.1
CRC [32]	92.3	LatLRR [37]	91.5
LLC [25]	79.4	LRC [31]	92.3
LLC* [25]	89.2	Lazebnik [52]	81.4
LRSIC [29]	92.4	Lian [5]	86.4
LRRC [28]	90.1	Yang [57]	80.3
SLRRC [28]	91.3	Boureau [61]	84.3
LC-KSVD1 [35]	90.4	Gemert [56]	76.7
LC-KSVD2 [35]	92.9	ILRDFL [58]	97.2
SC-LRR [43]	96.8	Our method	**98.1±0.2**

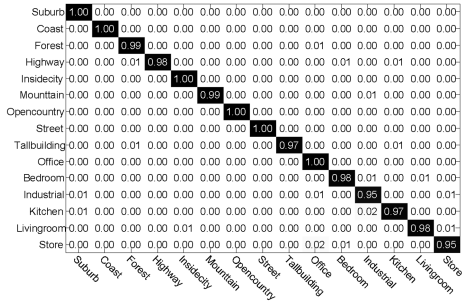


Fig. 7. Confusion matrix of our method on the Fifteen Scene Categories database.

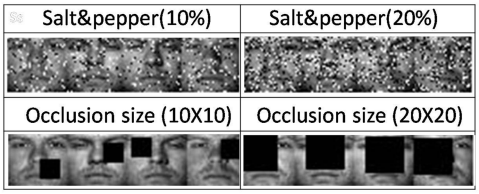


Fig. 8. Some corrupted images from Extended Yale B database.

TDDL are all 450. The number of neighborhoods of LLC* and LLC is set to 30. We also report the mean classification results (mean±std) over 30 runs for our method. The detailed comparison results are shown in Table VI. Again, our method performs the best among all the competitors. Our method outperforms the second and third best competitors ILRDFL and LC-KSVD2 by the margins of 0.9% and 5.2%, respectively. Fig. 7 shows the confusion matrix of our method on the Fifteen Scene Categories database, where the classification accuracy for each class is along the diagonal. All classes are classified well and the worst classification accuracy is as high as 95%.

C. Experiments on Contiguous and Random Pixel Corruptions

In this section, we randomly select 15 persons from the Extended Yale B database to test the robustness of our method to different types of corruptions. We, respectively, simulate various levels of contiguous occlusions and random pixel corruptions.

1) *Contiguous Occlusions:* The block occlusions are randomly added to different locations in the each image with a block size of 10×10 and 20×20, respectively.
2) *Random Pixel Corruptions:* We randomly choose pixels from each image and corrupt them by salt & pepper noise. The rates of corrupted pixels are 10% and 20%, respectively.

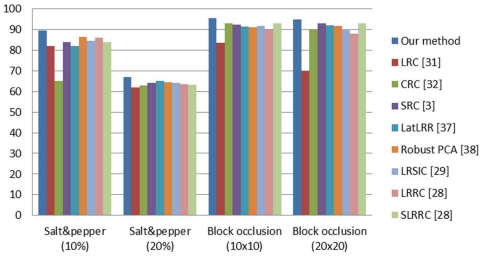
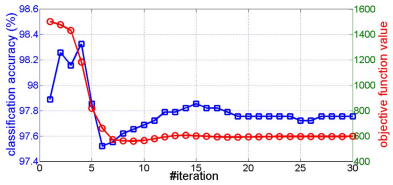
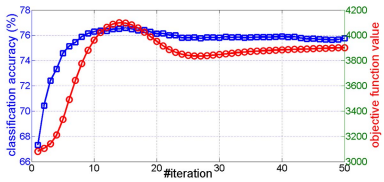


Fig. 9. Classification accuracies (%) on the Extended Yale B database, in which x-axis represents the different types of corruptions and y-axis denotes the classification accuracy (%).

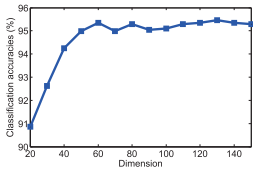


(a)

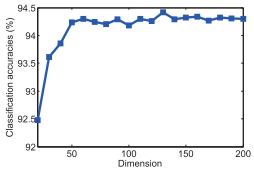


(b)

Fig. 10. Convergence curve and objective function versus iterations on (a) Fifteen scene and (b) Caltech101 databases.



(a)



(b)

Fig. 11. Classification accuracies (%) versus the dimension of latent subspace on (a) Extended Yale B and (b) CMU PIE databases, in which we randomly select 20 images per subject as training set and use the remaining as the testing set.

Fig. 8 shows some corrupted images from the Extended Yale B database. We randomly select 30 samples per subject as the training set and use the remaining as the testing set. Fig. 9 shows the classification results of different methods. Our method outperforms the others at all different types of corruption. We also note that all compared methods are more robust to contiguous occlusions than random pixel corruption.

D. Convergence and Parameters Sensitivity

In this section, we study the algorithmic convergence and the influence of parameters λ_1, λ_2, and λ_3 on classification accuracy by setting them at different values. In addition, we also give the range of value of d, i.e., the dimension of latent subspace.

1) Convergence: We show the convergence curve and objective function value versus the variation of iterations of our method in this section by running experiments on the Fifteen

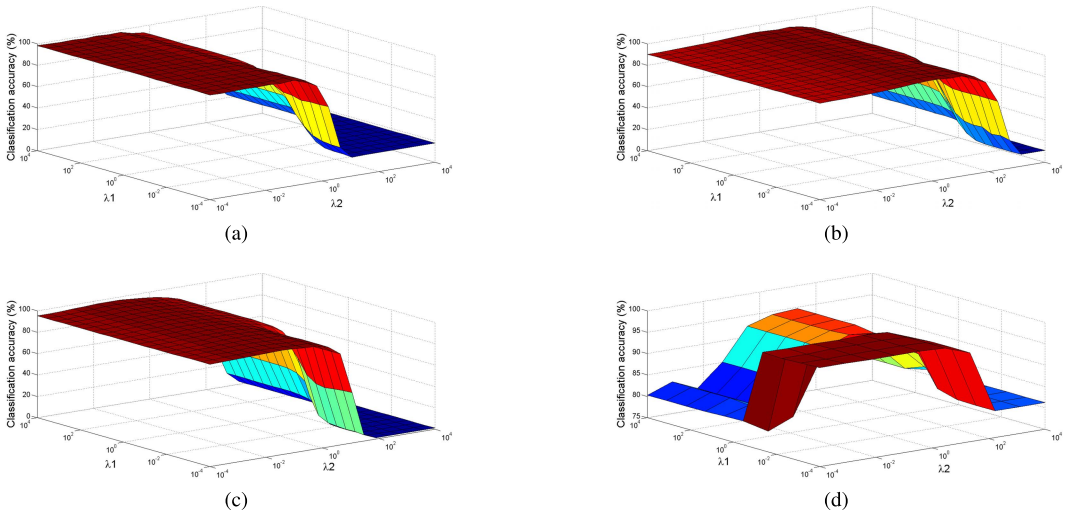


Fig. 12. Variations of classification accuracies (%) versus parameters λ_1 and λ_2 on (a) AR, (b) PIE, (c) Extended Yale B, and (d) Fifteen Scene Categories databases. # denotes that we randomly select the number of samples per subject as training set and use the remaining as the testing set.

Scene Categories and Caltech 101 databases. The experiment results are shown in Fig. 10. The objection function value on the Caltech 101 database increases in the first several iterations. This phenomenon can be interpreted as the consequence of the initial values for different variables. However, objective function values gradually decreases as the iteration number increases. Specifically, after 7 and 25 iterations for the Fifteen scene and Caltech 101 databases, respectively, they become stable. Meanwhile, the classification accuracy goes up in the first several iterations, and changes only a litter bit after ten iterations. The results in Fig. 10 demonstrate that the proposed optimization algorithm is effective and converges quickly, usually within 20 iterations. Similar results can be also found on the other databases.

2) Parameters Sensitivity: In practice, the value of d is difficult to tune, since its value could be from zero to infinite. In this section, we analyze the range of the value of d. First, we discard the constraint terms in problem (7). Equation (7) can be written as the following simple form:

$$\min_{Q,R} \|Y - RQ^T X\|_F^2 + \lambda \|RQ^T\|_F^2. \tag{25}$$

Proposition 1: Problem (24) is equivalent to doing the regularized regression (model parameter is R) in the regularized LDA subspace (projection matrix is Q).

The proof procedure is similar to that of [41, Th. 3] and interested readers can obtain more details from [41]. From Proposition 1, we know that d can be set to around c, where c is the number of classes. It is well known to all that the number of dimension for reconstruction is generally more than that of classes such as PCA. Therefore, if we consider the constraint terms, $d > c$. In our experiments, it is observed that the classification performance is not sensitive to the value of d if it is set a reasonable range. Fig. 11 shows the classification accuracies (%) versus the value of d in which we can see that when $d > c$, the classification accuracy reaches its peak and the changes are only a litter bit. In practical, d is tuned within $\{\min(5 \times \max\{\lfloor(c - 1/5)\rfloor, 1\}, c), \lfloor(m/100)\rfloor, \lfloor(m/50)\rfloor, \lfloor(m/15)\rfloor, \lfloor(m/10)\rfloor, \lfloor(m/5)\rfloor\}$, where $\lfloor k \rfloor$ denotes the largest integer not greater than k and m is the dimensionality of original data.

Fig. 12 shows the classification performance variation of our method with respect to different values of the parameters λ_1 and λ_2, which indicates that the performance of our method is not sensitive to the values of λ_1 and λ_2. Specifically, the classification performance is good when the values of parameters λ_2 and λ_1 are not large. The performance is bad when the parameter λ_2 is very large, which demonstrates that the sparse item cannot compensate noise well in this case. As a result, the noise degrades the performance. Our method is robust to different values of λ_1 on these databases when it is in the range of $[10^{-4}, 10^0]$, which indicates that a suitable λ_1 value can guarantee a better data representation. We also find that when the parameter λ_3 is tuned from $\{0.1, 0.5, 1, 3, 5, 10\}$, our method always obtain the best recognition results. How to identify the optimal values of these parameters are data dependent and still an open problem. In our experiments, λ_1 and λ_2 are first fixed due to the robustness and an attempt is made to find a candidate interval where the optimal parameters λ_3 and d may exist. Then, by fixed the value of λ_3 and d in the candidate interval, the candidate interval of λ_1 and λ_2 is determined. Finally, the optimal parameters in the 4-D candidate space (λ_1, λ_2, λ_3, and d) with a fixed step length are searched.

VI. Conclusion

In this paper, we propose a novel RLSL method for obtaining an appropriate data representation. Unlike conventional representation-based and SL methods which separate the feature learning and classification into two independent steps, our method learns data representation by integrating latent SL with LR to minimize the regression loss directly. In this way, the learned data representation can directly related to the classification performance and thus can greatly improve classification accuracy. We use two matrices to perform the

data reconstruction, such that the uncovered data representation is more accurate. We use a sparse item to model noise for the robustness of the uncovered data representation. Extensive experimental results show that our method achieves better recognition results than all the other methods in comparison. Our method is also much more robust than some state-of-the-art robust methods.

APPENDIX
PROOF OF THEOREM 1

First, we get the Lagrange multipliers Y_1 and Y_2 from Algorithm 1 as

$$\begin{aligned} Y_1^+ &\leftarrow Y_1 + \mu(X - PQ^TX - E) \\ Y_2^+ &\leftarrow Y_2 + \mu(RQ^T - A) \end{aligned} \tag{26}$$

where Y_i^+ is the next point of Y_i in a sequence $\{\theta_i^j\}_{j=1}^{\infty}$. If sequences of variables $\{Y_1^j\}_{j=1}^{\infty}$ and $\{Y_2^j\}_{j=1}^{\infty}$ converge to a stationary point, i.e., $(Y_1^+ - Y_1) \to 0$ and $(Y_2^+ - Y_2) \to 0$, then $(X - PQ^TX - E) \to 0$ and $(RQ^T - A) \to 0$. Therefore, the first two conditions in (23) are obtained.

For the third condition of the KKT condition, the following equation can be obtained by using the results obtained by Algorithm 1:

$$A^+ - A = \left(YX^T + \mu\left(RQ^T + \frac{Y_2}{\mu}\right)\right)(XX^T + \mu I)^{-1} - A \tag{27}$$

which is equivalent to

$$\begin{aligned} (A^+ - A)(XX^T + \mu I) &= \left(YX^T + \mu\left(RQ^T + \frac{Y_2}{\mu}\right)\right) \\ &\quad - A(XX^T + \mu I) \\ &= YX^T - AXX^T + Y_2 + \mu(RQ^T - A). \end{aligned} \tag{28}$$

Based on the second condition $RQ^T - A = 0$, we can infer that $(YX^T - AXX^T + Y_2) \to 0$, when $(A^+ - A) \to 0$.

Similar to the procedure of verifying the third condition, the fourth condition of the KKT condition can also be obtained by utilizing the result of Q in Algorithm 1. We first rewrite (14) as follows:

$$\frac{\lambda_1}{\mu}Q = XM^TP + B^TR - QR^TR - XX^TQ. \tag{29}$$

Then, we can obtain the following:

$$\begin{aligned} \frac{\lambda_1}{\mu}(Q^+ - Q) &= XM^TP + B^TR - QR^TR \\ &\quad - XX^TQ - \frac{\lambda_1}{\mu}Q \\ &= \frac{1}{\mu}\big(-\lambda_1 Q + XY_1^TP - Y_2^TR\big) \\ &\quad + \upsilon XX^TP - XE^TP \\ &\quad + A^TR - QR^TR - XX^TQ \\ &= \frac{1}{\mu}\big(-\lambda_1 Q + XY_1^TP - Y_2^TR\big) + (A - RQ^T)^TR \\ &\quad + X(PQ^TX)^TP - XX^TQ. \end{aligned} \tag{30}$$

Considering $P^TP = I$, (30) can be rewritten as

$$\begin{aligned} \frac{\lambda_1}{\mu}(Q^+ - Q) &= \frac{1}{\mu}\big(-\lambda_1 Q + XY_1^TP - Y_2^TR\big) \\ &\quad + (A - RQ^T)^TR + XX^TQP^TP - XX^TQ \\ &= \frac{1}{\mu}\big(-\lambda_1 Q + XY_1^TP - Y_2^TR\big) \\ &\quad + (A - RQ^T)^TR. \end{aligned} \tag{31}$$

If $(Q^+ - Q) \to 0$, then $(\lambda_1 Q - XY_1^TP + Y_2^TR) \to 0$ $(A - RQ^T = 0)$ as well.

Likewise, we can get the following equation using R from Algorithm 1:

$$(R^+ - R)(\lambda_3 I + \mu Q^TQ) = \mu(A - RQ^T)Q - (Y_2Q + \lambda_3 R). \tag{32}$$

Since $A - RQ^T$ converges to 0, we obtain $Y_2Q + \lambda_3 R = 0$ whenever $(R^+ - R) \to 0$.

For the last condition, from (23), we get the following equation:

$$E^+ - E = \mathcal{S}\left(X - PQ^TX + \frac{Y_1}{\mu}, \frac{\lambda_2}{\mu}\right) - E \tag{33}$$

when $(E^+ - E) \to 0$, we obtain the last condition.

Since $\{\theta^j\}_{j=1}^{\infty}$ is bound by the assumption in Theorem 1, $\{Q^{+^T}Q\}_{j=1}^{\infty}$ in (32) is bound. As a result, $\lim_{j\to\infty}(\theta^{j+1} - \theta^j) = 0$ can deduce that both sides of (28), (31), (32), and (33) are approximate to 0 when $j \to \infty$. Thus, the value of sequence $\{\theta\}_{j=1}^{\infty}$ asymptotically satisfies the KKT condition for objective function (7). □

REFERENCES

[1] S. Li and Y. Fu, "Robust subspace discovery through supervised low-rank constraints," in *Proc. SIAM Int. Conf. Data Mining*, 2014, pp. 163–171.

[2] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Projective dictionary pair learning for pattern classification," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 793–801.

[3] J. Wright, A. Y. Yang, A. Ganesh, S. 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.

[4] E. Kim, M. Lee, and S. Oh, "Elastic-net regularization of singular values for robust subspace learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 915–923.

[5] X.-C. Lian, Z. Li, B.-L. Lu, and L. Zhang, "Max-margin dictionary learning for multiclass image categorization," in *Proc. 11th Eur. Conf. Comput. Vis.*, Barcelona, Spain, Sep. 2010, pp. 157–170.

[6] Z. Ding, M. Shao, and Y. Fu, "Missing modality transfer learning via latent low-rank constraint," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4322–4334, Nov. 2015.

[7] Y. Fu, S. Yan, and T. S. Huang, "Correlation metric for generalized feature extraction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 12, pp. 2229–2235, Dec. 2008.

[8] A. M. Martínez and A. C. Kak, "PCA versus LDA," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 2, pp. 228–233, Feb. 2001.

[9] J. Yang, A. F. Frangi, J.-Y. Yang, D. Zhang, and Z. Jin, "KPCA plus LDA: A complete kernel Fisher discriminant framework for feature extraction and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 2, pp. 230–244, Feb. 2005.

[10] Z. Lai, Y. Xu, J. Yang, J. Tang, and D. Zhang, "Sparse tensor discriminant analysis," *IEEE Trans. Image Process.*, vol. 22, no. 10, pp. 3904–3915, Oct. 2013.

[11] Z. Fan, Y. Xu, and D. Zhang, "Local linear discriminant analysis framework using sample neighbors," *IEEE Trans. Neural Netw.*, vol. 22, no. 7, pp. 1119–1132, Jul. 2011.

[12] D. Cai, X. F. He, K. Zhou, J. W. Han, and H. J. Bo, "Locality sensitive discriminant analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 708–713.

[13] H.-T. Chen, H.-W. Chang, and T.-L. Liu, "Local discriminant embedding and its variants," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, San Diego, CA, USA, Jun. 2005, pp. 846–853.

[14] 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.

[15] Y. Han, Y. Yang, Y. Yan, Z. Ma, N. Sebe, and X. Zhou, "Semisupervised feature selection via spline regression for video semantic recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 2, pp. 252–264, Feb. 2015.

[16] H. Liu, Z. Wu, X. Li, D. Cai, and T. S. Huang, "Constrained nonnegative matrix factorization for image representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1299–1311, Jul. 2012.

[17] D. Cai, X. He, and J. Han, "Semi-supervised discriminant analysis," in *Proc. Int. Conf. Comput. Vis.*, 2007, pp. 1–7.

[18] F. Nie, D. Xu, I. W.-H. Tsang, and C. Zhang, "Flexible manifold embedding: A framework for semi-supervised and unsupervised dimension reduction," *IEEE Trans. Image Process.*, vol. 19, no. 7, pp. 1921–1931, Jul. 2010.

[19] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

[20] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Advances in Neural Information Processing System*, vol. 14. Cambridge, MA, USA: MIT Press, 2002, pp. 585–591.

[21] X. He, D. Cai, S. Yan, and H.-J. Zhang, "Neighborhood preserving embedding," in *Proc. Int. Conf. Comput. Vis.*, 2005, pp. 1208–1213.

[22] 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.

[23] J. Yang, D. Zhang, A. F. Frangi, and J.-Y. Yang, "Two-dimensional PCA: A new approach to appearance-based face representation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 131–137, Jan. 2004.

[24] 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.

[25] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proc. 23rd IEEE Conf. Comput. Vis. Pattern Recognit.*, San Francisco, CA, USA, Jun. 2010, pp. 3360–3367.

[26] J. Donahue *et al.*, "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 647–655.

[27] F. Nie, H. Wang, H. Huang, and C. Ding, "Adaptive loss minimization for semi-supervised elastic embedding," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, Beijing, China, 2013, pp. 1565–1571.

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

[29] C.-F. Chen, C.-P. Wei, and Y.-C. F. Wang, "Low-rank matrix recovery with structural incoherence for robust face recognition," in *Proc. 25th IEEE Conf. Comput. Vis. Pattern Recognit.*, Providence, RI, USA, Jun. 2012, pp. 2618–2625.

[30] A. Argyriou, T. Evgeniou, and M. Pontil, "Multi-task feature learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, pp. 41–48.

[31] I. Naseem, R. Togneri, and M. Bennamoun, "Linear regression for face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 11, pp. 2106–2112, Nov. 2010.

[32] L. Zhang, M. Yang, X. Feng, Y. Ma, and D. Zhang, "Collaborative representation based classification for face recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Jun. 2011, pp. 471–478.

[33] D. Huang, R. Cabral, and F. De la Torre, "Robust regression," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 363–375, Feb. 2016, doi: 10.1109/TPAMI.2015.2448091.

[34] 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.

[35] 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.

[36] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Sparse representation based Fisher discrimination dictionary learning for image classification," *Int. J. Comput. Vis.*, vol. 109, no. 3, pp. 209–232, Sep. 2014.

[37] G. Liu and S. Yan, "Latent low-rank representation for subspace segmentation and feature extraction," in *Proc. 13th IEEE Int. Conf. Comput. Vis.*, Barcelona, Spain, Nov. 2011, pp. 1615–1622.

[38] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 3, 2011, Art. no. 11.

[39] F. Bunea, Y. She, and M. Wegkamp, "Optimal selection of reduced rank estimators of high-dimensional matrices," *Ann. Statist.*, vol. 39, no. 2, pp. 1282–1309, 2011.

[40] S. Xiang, Y. Zhu, X. Shen, and J. Ye, "Optimal exact least squares rank minimization," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2012, pp. 480–488.

[41] X. Cai, C. Ding, F. Nie, and H. Huang, "On the equivalent of low-rank linear regressions and linear discriminant analysis based regressions," in *Proc. 19th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, 2013, pp. 1124–1132.

[42] H. Zhang, Z. Lin, C. Zhang, and J. Gao, "Relations among some low-rank subspace recovery models," *Neural Comput.*, vol. 27, no. 9, pp. 1915–1950, Sep. 2015.

[43] K. Tang, R. Liu, and J. Zhang, "Structure-constrained low-rank representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 12, pp. 2167–2179, Dec. 2014.

[44] G.-J. Qi, C. Aggarwal, Q. Tian, H. Ji, and T. S. Huang, "Exploring context and content links in social media: A latent space method," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 850–862, May 2011.

[45] Z. Li, J. Liu, J. Tang, and H. Lu, "Robust structured subspace learning for data representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 10, pp. 2085–2098, Oct. 2015.

[46] X. Fang, Y. Xu, X. Li, Z. Lai, and W. K. Wong, "Robust semi-supervised subspace clustering via non-negative low-rank representation," *IEEE Trans. Cybern.*, vol. 46, no. 8, pp. 1828–1838, Aug. 2016.

[47] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.

[48] A. S. Georghiades, P. N. Belhumeur, and D. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 643–660, Jun. 2001.

[49] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression database," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1615–1618, Dec. 2003.

[50] A. Martínez and R. Benavente, "The AR face database," Dept. Centre Visió Computador, Univ. Autònoma Barcelona, Barcelona, Spain, Tech. Rep. #24, Jun. 1998.

[51] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories," in *Proc. 17th IEEE Conf. Comput. Vis. Pattern Recognit.*, Washington, DC, USA, Jun. 2004, pp. 59–70.

[52] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, New York, NY, USA, Jun. 2006, pp. 2169–2178.

[53] 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.

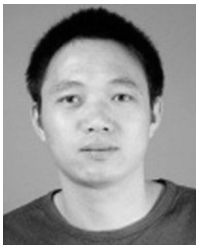
[54] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[55] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2691–2698.

[56] J. C. van Gemert, J.-M. Geusebroek, C. J. Veenman, and A. W. M. Smeulders, "Kernel codebooks for scene categorization," in *Proc. 10th Eur. Conf. Comput. Vis.*, Marseille, France, Oct. 2008, pp. 696–709.

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

[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 2015.
[59] T. Tommasi and T. Tuytelaars, "A testbed for cross-dataset analysis," in *Computer Vision—ECCV 2014 Workshops*. Zurich, Switzerland, 2014, pp. 18–31.
[60] A. Coates and A. Y. Ng, "The importance of encoding versus training with sparse coding and vector quantization," in *Proc. 28th Int. Conf. Mach. Learn.*, Washington, DC, USA, Jun. 2011, pp. 921–928.
[61] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," in *Proc. 23rd IEEE Conf. Comput. Vis. Pattern Recognit.*, San Francisco, CA, USA, Jun. 2010, pp. 2559–2566.
[62] J. C. van Gemert, C. J. Veenman, A. W. M. Smeulders, and J.-M. Geusebroek, "Visual word ambiguity," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 7, pp. 1271–1283, Jul. 2010.



Xiaozhao Fang (S'15–M'17) received the M.S. and Ph.D. degrees in computer science and technology with the Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen, China, in 2008 and 2016, respectively.

He is currently with the School of Computer Science and Technology, Guangdong University of Technology, Guangzhou, China. His current research interests include pattern recognition and machine learning.

Shaohua Teng was born in 1962. He received the Ph.D. degree in industry engineering with the Guangdong University of Technology, Guangzhou, China.

He is currently a Professor with the Guangdong University of Technology, Guangzhou, China. He is responsible for teaching data mining with the School of Computer Science and Technology. He is engaged in education and technology transfer on knowledge discovery issues. He has applied for six patents on his invention. He has authored 300 papers on computer magazines and international conferences and two books. His current research interests include big data, network security, cooperative work, machine learning, and statistical pattern recognition.

Dr. Teng received the Provincial Science and Technology Award, and the Guangdong Outstanding Teacher.

Zhihui Lai received the B.S. degree in mathematics from South China Normal University, Guangdong, China, the M.S. degree from Jinan University, Guangdong, and the Ph.D. degree in pattern recognition and intelligence system from the Nanjing University of Science and Technology, Nanjing, China, in 2002, 2007, and 2011, respectively.

He was a Research Associate and a Post-Doctoral Fellow with The Hong Kong Polytechnic University, Hong Kong, from 2010 to 2013. He is currently a Post-Doctoral Fellow with the Bio-Computing Research Center, Shenzhen Graduate School, Harbin Institute of Technology, Harbin, China. He has authored over 30 scientific papers in pattern recognition and computer vision. His current research interests include face recognition, image processing, and content-based image retrieval, pattern recognition, and compressive sense. For more information including all papers and related codes, the readers are referred to the website (http://www.scholat.com/laizhihui).

Zhaoshui He received the B.S. degree in applied mathematics from Hunan Normal University, Changsha, China, in 2000, and the Ph.D. degree in electronics and information engineering from the South China University of Technology, Guangzhou, China, in 2005.

He was a Research Scientist with the RIKEN Brain Science Institute, Tokyo, Japan. He is currently a Faculty Member of the Laboratory for Intelligent Information Processing, Guangdong University of Technology, Guangzhou. His current research interests include blind signal processing, sparse representation, model selection, tensor analysis, clustering, and their applications.

Shengli Xie (M'01–SM'02) received the M.S. degree in mathematics from Central China Normal University, Wuhan, China, in 1992, and the Ph.D. degree in automatic control from the South China University of Technology, Guangzhou, China, in 1997.

He was the Vice Dean of the School of Electronics and Information Engineering with the South China University of Technology from 2006 to 2010. He is currently the Director of the Institute of Intelligent Information Processing and the Guangdong Key Laboratory of Information Technology for the Internet of Things, and a Professor with the School of Automation, Guangdong University of Technology, Guangzhou. He has authored or co-authored four monographs and over 100 scientific papers in journals and conference proceedings. He holds over 30 patents. His current research interests include statistical signal processing and wireless communication, with an emphasis on blind signal processing and Internet of Things.

Wai Keung Wong received the Ph.D. degree from The Hong Kong Polytechnic University, Hong Kong.

He is currently with the Institute of Textiles and Clothing, The Hong Kong Polytechnic University, Hong Kong and The Hong Kong Polytechnic University Shenzhen Research Institute, Shenzhen, China. He has authored over 50 scientific articles in refereed journals, including the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the *Pattern Recognition*, the *International Journal of Production Economics*, the *European Journal of Operational Research*, the *International Journal of Production Research*, the *Computers in Industry*, and the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS. His current research interests include artificial intelligence, pattern recognition, and optimization of manufacturing scheduling, planning, and control.