Approximate Low-Rank Projection Learning for Feature Extraction

Xiaozhao Fang, *Member, IEEE*, Na Han, Jigang Wu, *Member, IEEE*, Yong Xu, *Senior Member, IEEE*, Jian Yang, *Member, IEEE*, Wai Keung Wong, and Xuelong Li, *Fellow, IEEE*

***Abstract*— Feature extraction plays a significant role in pattern recognition. Recently, many representation-based feature extraction methods have been proposed and achieved successes in many applications. As an excellent unsupervised feature extraction method, latent low-rank representation (LatLRR) has shown its power in extracting salient features. However, LatLRR has the following three disadvantages: 1) the dimension of features obtained using LatLRR cannot be reduced, which is not preferred in feature extraction; 2) two low-rank matrices are separately learned so that the overall optimality may not be guaranteed; and 3) LatLRR is an unsupervised method, which by far has not been extended to the supervised scenario. To this end, in this paper, we first propose to use two different matrices to approximate the low-rank projection in LatLRR so that the dimension of obtained features can be reduced, which is more flexible than original LatLRR. Then, we treat the two low-rank matrices in LatLRR as a whole in the process of learning. In this way, they can be boosted mutually so that the obtained projection can extract more discriminative features. Finally, we extend LatLRR to the supervised scenario by integrating feature extraction with the ridge regression. Thus, the process of feature extraction is closely related to the classification so that the extracted features are discriminative. Extensive experiments are conducted on different databases for unsupervised and supervised feature extraction, and very encouraging results are achieved in comparison with many state-of-the-arts methods.**

***Index Terms*— Computer vision, feature extraction, low-rank representation (LRR), pattern recognition, ridge regression.**

I. Introduction

FEATURE extraction is a critical step for data representation and has been widely applied in pattern recognition, data mining, and computer vision to name just a few. Many works [1]–[6] have been proposed for feature extraction. For example, principle component analysis (PCA) [7] is an unsupervised feature extraction method, which projects high-dimensional data into a lower dimensional subspace by seeking the direction of maximum variance for the optimal data reconstruction. Neighbor preserving embedding (NPE) [8] and locality preserving projections (LPPs) [9] exploit the local relationship between data points and its neighbors to perform feature extraction. Nonnegative matrix factorization (NMF) [10] has been proposed for multivariate data analysis with nonnegative constraints. NMF has nonnegative and local characteristics, and thus, the obtained nonnegative components can be used as new features of original data. Linear discriminant analysis [11] is a supervised feature extraction method, which projects data into a lower dimensional subspace based on Fisher's linear discriminant and produces well-separated features. Wang *et al.* [12] projected each descriptor into a local-coordinate system and used these local coordinates as new features for the follow-up learning tasks.

Recently, representation-based feature extraction methods [13]–[15] have drawn great attention. Sparse representation (SR) and low-rank representation (LRR) are the most famous two representation-based feature extraction methods. SR classification (SRC) has shown its excellent power in face recognition [5]. SRC uses the smallest number of training samples to represent the test sample and then uses the representation results to perform classification. In other words, SRC adopts the representation coefficients as the new representation of data to perform final classification. Unfortunately, when the training samples are widely corrupted, e.g., unreasonable expression, pose, and illumination, the performance of SRC may be degraded. To address this problem, a series of methods has been proposed [3], [6], [16], [17] and among them, the LRR-based methods have recently attracted a great deal of attention due to the pleasing efficacy in recovering data and

Manuscript received April 10, 2017; revised August 2, 2017 and November 15, 2017; accepted January 16, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61573248, Grant 61772141, Grant 61375012, Grant 61773328, Grant 61703283, and Grant 61761130079, in part by The Hong Kong Polytechnic University under Project 1-YW1C, in part by the Research Grant of The Hong Kong Polytechnic University under Project G-YBD9, in part by the Key Research Program of Frontier Sciences, CAS, under Grant QYZDY-SSW-JSC044, in part by the Guangdong Provincial Natural Science Foundation under Grant 17ZK0422, in part by the Guangdong Science and Technology Planning Project under Grant 2013B091300009 and Grant 2016B030308001, and in part by the Guangzhou Science and Technology Planning Project under Grant 201604046017 and Grant 201604030034. *(Corresponding author: Na Han.)*

X. Fang, N. Han, and J. Wu are with the School of Computer Science and Technology, Guangdong University of Technology, Guangzhou 510006, China (e-mail: xzhfang168@126.com; hannagdut@126.com; asjgwu@gmail.com).

Y. Xu is with the Bio-Computing Research Center, Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen 518055, China, and also with the Key Laboratory of Network Oriented Intelligent Computation, Shenzhen 518055, China (e-mail: yongxu@ymail.com).

J. Yang is with the School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: csjyang@mail.njust.edu.cn).

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

X. Li is with the Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: xuelong_li@opt.ac.cn).

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.2018.2796133

removing errors. These LRR-based methods focus on low-rank data representation based on the hypothesis that data approximately jointly span several low-dimensional subspaces [16]. Since the dimension of subspace corresponds to the rank of representation coefficient matrix, these LRR-based methods impose a low-rank constraint on the representation matrix to enhance the correlation among the representation coefficient vectors. Thus, these LRR-based models are easy to capture the global structure of data. To exploit the local structure of data, Zhuang *et al.* [18] proposed to impose joint low-rank and sparse constraints on the representation coefficient matrix so that the global and local structures of data can be simultaneously captured. Latent low-rank representation (LatLRR) [3] is a recently proposed feature learning method, which considers two views of sample matrix. In other words, LatLRR recovers the column and row space information of data by learning two different low-rank matrices. In LatLRR, one of these two low-rank matrices is used as the projection matrix for extracting salient features. The experimental results in [3] shown that the extracted salient features are discriminative. However, LatLRR has the following three disadvantages, which may degrade the performance. First, LatLRR separately learns these two low-rank matrices so that they cannot be boosted mutually during learning. Second, the dimension of features learned by LatLRR is as the same as that of the original data (or the dimension of features learned by LatLRR cannot be reduced), which is not preferred in many applications, such as dimensionality reduction and feature extraction. Finally, as far as we know, LatLRR has not been extended to the supervised scenario. In addition, in the supervised scenario, many representation-based feature extraction methods commonly consist of two separate steps. They first extract the discriminative features by label information. Then, they use the extracted features to train a specific classifier such as support vector machine. In other words, these representation-based methods minimize the rankness and sparsity of some solution related to feature learning, which is not directly connected to the subsequent recognition tasks. Thus, it is evident that these two independent steps may limit the overall optimally in recognition in some sense.

To address these problems mentioned above, in this paper, we first propose an approximate low-rank projection learning (ALPL) for feature extraction in which two different matrices are introduced to replace the single low-rank projection matrix in LatLRR. In this way, the dimension of features learned by ALPL can be reduced, which is more flexible than LatLRR. To learn an optimal low-rank projection for extracting discriminant features, we further propose an extended approximate low-rank projection matrix learning (EALPL) method that treats two different low-rank matrices as a whole instead of separately learning them as in LatLRR. Therefore, these matrixes can be boosted mutually. In this way, EALPL can effectively encode salient features of data by considering the recovery of row and column space information simultaneously. In order to extend EALPL to the supervised scenario, we propose to integrate feature extraction with the rigid regression so that the process of feature extraction is closely related to classification. Thus, the extracted features are discriminative for recognition. Extensive experiments of unsupervised and supervised feature extraction are conducted on different databases that verify the advantages of our methods.

Our key contributions are summarized as follows.

1) We address the problem that the dimension of features learned by LatLRR cannot be reduced by using two different matrices to replace the single low-rank matrix in LatLRR.
2) We further propose a simple yet effective method for simultaneously recovering row and column space information. In doing so, we can learn the optimal projection by treating two different low-rank matrices used in LatLRR as a whole in the learning process. The experiments show that the proposed method can extract more discriminative features.
3) We extend LatLRR to the supervised scenario by integrating feature extraction with rigid regression. In this way, the extracted features are discriminative and thus are competent to recognition tasks.
4) We develop efficient algorithms based on the alternating direction method of multipliers (ADMM) to solve the proposed formulations. The theoretical and empirical analysis demonstrates that the designed optimization algorithms are efficient and effective.

The remainder of this paper is arranged as follows. We introduce the related works in Section II. Then, we elaborate our methods in Section III followed with its optimization algorithm. The experiments and analyses are represented in Section IV. Section V concludes this paper with future work.

II. Related Works

In this section, we briefly review the related works of representation-based feature extraction methods. Please note that since our methods are related to SR and LRR, we mainly introduce many SR-based and LRR-based feature extraction methods.

A. Sparse Representation-Based Feature Extraction

SRC [5] is a classic representation-based method, which represents a test sample $y \in \Re^m$ by a linear combination of a few atoms in an over complete dictionary $D = [D_1, \ldots, D_c] \in \Re^{m\times\kappa}$ with an SR coefficient vector $\alpha \in \Re^{\kappa}$, where c denotes the number of classes and $D_i \in \Re^{m\times\kappa_i}$ is the subdictionary associated with the ith class and $\kappa = \sum_{i=1}^{c} \kappa_i$. The objective function of SRC is as follows:

$$\min_{\alpha} \|y - D\alpha\|_2^2 + \beta\|\alpha\|_1 \tag{1}$$

where β is the relative weight between the two terms and $\|\alpha\|_1$ is the ℓ_1 norm of α, which is defined as $\|\alpha\|_1 = \sum_i |\alpha_i|$, where α_i represents the ith element of α. Suppose that $\alpha = [\alpha_1^T, \ldots, \alpha_c^T]^T$, and α_i is the subvector associated with the dictionary D_i of the ith class. Test sample y is then classified to class j^* if class j^* produces the smallest reconstruction error

$$j^* = \arg\min_{j} \|y - D_j\alpha_j\|_2^2. \tag{2}$$

It is reported that SRC achieves the surprised recognition results on face recognition [5]. A limitation of SRC is that the atoms in the dictionary are required to be well aligned. However, the strict requirement is impractical in many real-word applications. To address the problem, Wagner *et al.* [19] proposed and extended SRC to deal with variations of face in illumination, alignment, occlusion, and pose. Peng *et al.* [20] proposed a robust alignment method via sparse and low-rank decomposition to seek an optimal set of image domain transformations for linearly correlated images. Zhuang *et al.* [21] proposed a single-sample face recognition method by introducing the sparse illumination learning and transfer technique for the image corruption and misalignment. In addition to SRC, collaborative representation-based classification (CRC) [14] and linear representation-based classification (LRC) [13] have also achieved good recognition results for face recognition. Xu *et al.* [22] further proposed a two-step SR method for face recognition. Dictionary learning (DL) is also a representation-based method for feature learning. The reconstruction coefficients in DL can be used as the new features of training data, which should be discriminative for learning a discriminative dictionary. Mairal *et al.* [23] proposed a task-driven DL (TDDL) method, which incorporates the empirical error into the DL, and thus TDDL can learn a discriminative dictionary. Jiang *et al.* [24] proposed a label consistent DL method (LC-KSVD) by integrating a linear classifier into DL.

B. Low-Rank Representation-Based Feature Extraction

We begin with a review of LRR [3], [16]. Then, we introduce LatLRR [3].

The LRR model is based on the assumption that data are approximately sampled from a union of multiple low-dimensional subspaces. Given a set of data samples $X \in \Re^{m\times n}$ (m and n are the dimension and the number of samples, respectively) that are drawn from a union of multiple subspaces given by $\bigcup_{i=1}^{\pi} \mathcal{S}$, where $\mathcal{S}_1, \mathcal{S}_2,\ldots,\mathcal{S}_\pi$ are low-dimensional subspaces, LRR aims at finding the lowest rank representation of all samples jointly. The objective function of LRR is as follows:

$$\min_{Z,E} \operatorname{rank}(Z) + \lambda\|E\|_0, \quad \text{s.t. } X = AZ + E \tag{3}$$

where the columns of A are a set of known bases or dictionary items, E denotes the error components, $rank(\cdot)$ denotes the rank of matrix, $\|\cdot\|_0$ is the ℓ_0 pseudonorm, and λ is a penalty parameter for balancing the low-rank term and the reconstruction fidelity. Since the rank function is discrete and ℓ_0 is a pseudonorm, it is difficult to solve problem (3). A convex relaxation of the above optimization problem was proposed as

$$\min_{Z,E} \|Z\|_* + \lambda\|E\|_1, \quad \text{s.t. } X = AZ + E \tag{4}$$

where $\|Z\|_*$ is the nuclear norm of Z, which is the sum of all the singular values of Z.

Once obtaining a low-rank solution (Z^*, E^*), we can recover the "clean" representation AZ^* or $X - E^*$ for data X. We also directly recover the "clean" data, whose optimization problem is the formulation of robust PCA (RPCA) [25]

$$\min_{Y,E} \|Y\|_* + \lambda\|E\|_1, \quad \text{s.t. } X = Y + E \tag{5}$$

where Y is the "clean" representation of X. RPCA only aims to recover the low-rank "clean" data from given noisy data, but LRR can reveal the implicit data membership. However, when the training samples are not sufficient, the performance of LRR may be degraded. LatLRR [3] is a recently proposed LRR-based subspace learning method, which can exploit many unobserved samples to represent the observed samples well. Its model is formulated as

$$\min_{Z} \|Z\|_*, \quad \text{s.t. } X = [X, X_H]Z \tag{6}$$

where X is the observed data and X_H is the unobserved hidden data. With Bayesian inference [26], X can be represented by $X = XZ + LX$, where $Z \in \Re^{n\times n}$ is the low-rank reconstruction matrix and $L \in \Re^{m\times m}$ is the low-rank projection matrix. The objective function of LatLRR is formulated as

$$\min_{Z,L} \|Z\|_* + \lambda\|L\|_1, \quad \text{s.t. } X = XZ + LX. \tag{7}$$

To reduce the influence of noise, LatLRR uses a sparse matrix $E \in \Re^{m\times n}$ to model noise

$$\min_{Z,L,E} \|Z\|_* + \|L\|_* + \lambda\|E\|_1, \quad \text{s.t. } X = XZ + LX + E. \tag{8}$$

The experimental results in [3] show that the features represented by XZ are visually similar to PCA features, i.e., principle features. The features represented by LX are the salient features, which correspond to the key object parts such as the eyes in a face image. From [3], we can see that LatLRR separately uses two low-rank matrices for two views, i.e., column and row views. In this way, there is no response between these two low-rank matrices so that the learned projection matrix may not be optimal. The dimension of features learned by LX is as the same as that as X. However, the goal of feature extraction is to extract the more flexible and discriminative feature for the follow-up classification task and thus LatLRR is not a perfect feature extraction method from this point.

III. Approximate Low-Rank Projection Learning

In this section, we introduce our methods for feature extraction in unsupervised and supervised scenarios.

A. Problem Formulation for Feature Extraction in Unsupervised Scenario

As previously discussed, the dimension of features obtained using LatLRR is fixed, which is disadvantageous for real feature extraction. To address this problem, we use two different matrices P and Q to replace the single low-rank projection matrix $L \in \Re^{m\times m}$ in LatLRR. In other words, we set $L = PQ^T$. Thus, we propose the following formulation:

$$\begin{aligned}\min_{P,Q,Z,E} \quad & \|Z\|_* + \frac{1}{2}\lambda_1\|Q\|_F^2 + \lambda_2\|E\|_1 \\ \text{s.t. } & X = XZ + PQ^T X + E\end{aligned} \tag{9}$$

where $P \in \Re^{m\times d}$ and $Q \in \Re^{m\times d}$, $d < m$. Thus, the rank of $L = PQ^T$ is d (where d is the dimensionality of the derived new subspace). λ_1 and λ_2 are two parameters that weight the importance of these three items. Matrix Q can be used as a projection matrix to extract salient features by the linear transformation $Q^T X$ and the dimension of features represented by $Q^T X$ can be chosen as one expectation. In other words, we can project the original data into a subspace with a random dimension. In order to avoid the trivial solution for the problem of (9), we impose the orthogonal constraint on matrix P. Thus, (9) can be rewritten as

$$\min_{P,Q,Z,E} \|Z\|_* + \frac{1}{2}\lambda_1\|Q\|_F^2 + \lambda_2\|E\|_1$$
$$\text{s.t. } X = XZ + PQ^T X + E, \quad P^T P = I. \tag{10}$$

We call the formulation in (10) as the approximate low-rank projection learning (ALPL) method. If we abandon XZ, the constraint term becomes $X = PQ^T X + E$. This is somewhat a PCA-like term, whose purpose is to ensure that $Q^T X$ can hold the main energy of data so as to guarantee a better recognition result [2], [4]. It is evident that ALPL separately recovers the column and row space information of data [3]. Specifically, ALPL uses XZ and $PQ^T X$ to recover row space and column space information of data, respectively. However, a disadvantage is that there is no interaction among P, Q, and Z in the process of learning so that they cannot be boosted mutually. To accurately recover the column and row space information of data, we further propose to simultaneously calculate P, Q, and Z, i.e., we treat $PQ^T XZ$ as a whole instead of separately calculating them. In doing so, we can learn the optimal P, Q, and Z by exploiting the mutual boosting among them. Thus, we propose the following objective function, which aims to simultaneously recover the column space and row space information of data by using $PQ^T XZ$

$$\min_{P,Q,Z,E} \|Z\|_* + \frac{1}{2}\lambda_1\|Q\|_F^2 + \lambda_2\|E\|_1$$
$$\text{s.t. } X = PQ^T XZ + E, \quad P^T P = I. \tag{11}$$

We call the formulation in (11) as the EALPL. In the subsequent experiments, we will verify that EALPL usually obtains higher recognition rate than ALPL. In Fig.3, we experimentally show that the Projection matrix Q contains main details information of data. Thus, the features represented by $Q^T X$ are very competent to recognition tasks.

1) ADMM for Solving ALPL and EALPL: In this section, we use the ADMM to solve problem (10). First, we introduce an auxiliary variable H in order to make the objective function of (10) separable. Thus, the optimization problem can be rewritten as follows:

$$\min_{P,Q,Z,E,H} \|H\|_* + \frac{1}{2}\lambda_1\|Q\|_F^2 + \lambda_2\|E\|_1$$
$$\text{s.t. } X = XZ + PQ^T X + E, \quad P^T P = I, \; Z = H. \tag{12}$$

The augmented Lagrangian function of problem (12) is

$$\begin{aligned}
&\mathcal{L}(P, Z, Q, E, H)\\
&\quad = \|H\|_* + \frac{1}{2}\lambda_1\|Q\|_F^2 + \lambda_2\|E\|_1\\
&\qquad + \langle Y_1, X - XZ - PQ^T X - E\rangle + \langle Y_2, Z - H\rangle\\
&\qquad + \frac{\mu}{2}\big(\|X - XZ - PQ^T X - E\|_F^2 + \|Z - H\|_F^2\big)\\
&\qquad \text{s.t. } P^T P = I
\end{aligned} \tag{13}$$

where Y_1 and Y_2 are Lagrange multipliers and $\mu > 0$ is a penalty parameter. The variables are updated alternately by minimizing the augmented Lagrangian function, with other variables fixed. We provide details of solving (13) with ADMM in the following:

$$\min_H \|H\|_* + \frac{\mu}{2}\left\|Z - H + \frac{Y_2}{\mu}\right\|_F^2 \tag{14}$$

$$\min_E \lambda_2\|E\|_1 + \frac{\mu}{2}\|G_1 - E\|_F^2 \tag{15}$$

$$\min_Q \frac{1}{2}\lambda_1\|Q\|_F^2 + \frac{\mu}{2}\|G_2 - PQ^T X\|_F^2 \tag{16}$$

$$\min_P \frac{\mu}{2}\|G_2 - PQ^T X\|_F^2, \quad \text{s.t. } P^t P = I \tag{17}$$

$$\min_Z \frac{\mu}{2}\left(\|G_3 - XZ\|_F^2 + \left\|Z - H + \frac{Y_2}{\mu}\right\|_F^2\right) \tag{18}$$

where $G_1 = X - XZ - PQ^T X + (Y_1/\mu)$, $G_2 = X - XZ - E + (Y_1/\mu)$, and $G_3 = X - PQ^T X - E + (Y_1/\mu)$. The solutions of H and E are $\Theta_{(1/\mu)}(Z + (Y_2/\mu))$ and $\Omega_{(\lambda_2/\mu)}(G_1 + (Y_1/\mu))$, where Θ and Ω are the singular value thresholding shrinkage [3] and the ℓ_1 minimization operator [4], respectively. By setting the derivative of (16) with respect to Q and setting to zero, we obtain $Q = (\lambda_1\ I + \mu XX^T)^{-1}(\mu X G_2^T P)$ for problem (16). Problem (17) is a classic orthogonal ProCrustes problem [27], which is solved as follows: first compute the singular-value decomposition (SVD) of matrix $G_2 X^T Q$ as $G_2 X^T Q = USV^T$ and then let $P = UV^T$. By setting the derivative of (18) with respect to Z and setting to zero, we obtain $Z = (I + X^T X)^{-1}(H - (Y_2/\mu) + X^T G_3)$ for problem (18). The complete algorithm is outlined in Algorithm 1. Please note that we initialize orthogonal matrix P by using the PCA, which can speed up algorithm speed.

In Section III-B, we will introduce the supervised ALPL (SALPL) method and corresponding optimization algorithm. Please note that the optimization procedure of problem (11) is similar to that of SALPL, and thus, we do not describe the optimization procedure of problem (11) detailedly in this section.

B. Problem Formulation for Feature Extraction in Supervised Scenario

To the best of our knowledge, LatLRR is originally designed for feature extraction in the unsupervised scenario. In this section, we extend LatLRR to the supervised scenario by introducing the label information. We first define a binary label matrix $Y = [y_1, \ldots, y_n] \in \Re^{c\times n}$, where c is the number of classes. For each training sample $x_i \in \Re^m$ $(i = 1, \ldots, n)$,

Algorithm 1 ALPL

```
Input: Training samples matrix $X$; Parameters $\lambda_1$, $\lambda_2$; Dimensionality $d$.
Initialization: $H = \mathbf{0}$; $Q = \mathbf{0}$; $E = \mathbf{0}$; $Z = \mathbf{0}$; $P^* = arg \min_P Tr(P^T(-\Sigma)P),\ s.t.\ P^T P = I$; where $\Sigma$ is the data covariance; $Y_1 = \mathbf{0}$; $Y_2 = \mathbf{0}$; $\mu_{max} = 10^5$; $\rho = 1.01$; $\mu = 0.1$.
while not converged do
  1. Update $P$ by solving (17);
  2. Update $Z$ by solving (18);
  3. Update $Q$ by solving (16);
  4. Update $E$ by solving (15);
  5. Update $H$ by solving (14);
  6. Update $Y_1$, $Y_2$ and $\mu$ by
     $\begin{cases} Y_1 \leftarrow Y_1 + \mu(X - XZ - PQ^T X - E) \\ Y_2 \leftarrow Y_2 + \mu(Z - H) \\ \mu \leftarrow \min\{\rho\mu, \mu_{max}\} \end{cases}$
end while
Output: Projection matrix $Q$
```

$y_i = [0, 0, \ldots, 1, \ldots, 0, 0]^T \in \Re^c$ is its label vector, where the position of 1 indicates the class of x_i.

Our basic idea is to utilize the label information, i.e., the binary label matrix Y, to learn discriminative features $Q^T X$ resulting from the EALPL. During the training process, the extracted features are fed into a classifier $f(x, Q)$ to learn its model parameter Q. To extract discriminative features, we thus aim at optimizing Q by minimizing the classification error so that the process of feature extraction is tightly related to classification performance. Thus, our objective function is as follows:

$$\min_Q \sum_{i=1}^{n} \phi(y_i, f(x_i, Q)) + \frac{1}{2}\|Q\|_F^2 \tag{19}$$

where $x_i \in \Re^m$ is the ith sample of $X \in \Re^{m \times n}$. In this paper, we use a linear classifier $f(x, Q) = Q^T x$, i.e., adopt the multivariate rigid regression. The optimization can still be performed for other classifiers but is more involved. The final objective function can be written as

$$\begin{aligned} &\min_{P,Q,Z,E} \frac{1}{2}\|Y - Q^T X\|_F^2 + \frac{1}{2}\lambda_1\|Q\|_F^2 + \lambda_2\|E\|_1 + \lambda_3\|Z\|_* \\ &\text{s.t. } X = PQ^T XZ + E, \quad P^T P = I \end{aligned} \tag{20}$$

where $\lambda_1 \geq 0$, $\lambda_2 \geq 0$, and $\lambda_3 \geq 0$ are the nonnegative parameters. We call the formulation in (20) as the SALPL. It is evident that Q values are both the projection matrix and the parameter of the classification model in (20). The term $\|Y - Q^T X\|_F^2$ represents the classification error. By using Q, we can project the original samples from the original feature space with a dimension of m into a discriminative space with a dimension of c. Accordingly, discriminative features $Q^T X$ can be obtained.

1) ADMM for Solving SALPL: We also use ADMM to solve problem (20). We first convert (20) into the following augmented Lagrange function by defining $Z = H$:

$$\begin{aligned} &\mathcal{F}(P, Q, Z, E, H) \\ &= \|Y - Q^T X\|_F^2 + \frac{1}{2}\lambda_1\|Q\|_F^2 + \lambda_2\|E\|_1 \\ &\quad + \lambda_3\|H\|_* + \langle Y_1, X - PQ^T XZ - E\rangle + \langle Y_2, Z - H\rangle \\ &\quad + \frac{\mu}{2}\left(\|X - PQ^T XZ - E\|_F^2 + \|Z - H\|_F^2\right) \\ &\quad \text{s.t. } P^T P = I \end{aligned} \tag{21}$$

where Y_1 and Y_2 are the Lagrange multipliers and $\mu > 0$ is a penalty parameter. We iteratively update variables until the convergence of the algorithm. The details of updating variables are as follows:

$$\min_Q \frac{1}{2}\|Y - Q^T X\|_F^2 + \frac{1}{2}\|Q\|_F^2 + \frac{\mu}{2}\left\|X - PQ^T XZ - E + \frac{Y_1}{\mu}\right\|_F^2 \tag{22}$$

$$\min_Z \frac{\mu}{2}\left(\left\|X - PQ^T XZ - E + \frac{Y_1}{\mu}\right\|_F^2 + \left\|Z - H + \frac{Y_2}{\mu}\right\|_F^2\right) \tag{23}$$

$$\min_P \frac{\mu}{2}\left\|X - PQ^T XZ - E + \frac{Y_1}{\mu}\right\|_F^2, \text{ s.t. } P^T P = I \tag{24}$$

$$\min_H \lambda_3\|H\|_* + \frac{\mu}{2}\left\|Z - H + \frac{Y_2}{\mu}\right\|_F^2 \tag{25}$$

$$\min_E \lambda_2\|E\|_1 + \frac{\mu}{2}\left\|X - PQ^T XZ - E + \frac{Y_1}{\mu}\right\|_F^2. \tag{26}$$

By setting the derivative of (22)and (23) with respect to Q and Z and setting them to zero, we obtain $Q = (XX^T + \lambda_1 I + \mu XZZ^T X^T)^{-1}(\mu XZK_1^T P + XY^T)$ and $Z = (X^T QQ^T X + I)^{-1}(K_3 + X^T QP^T K_2)$, respectively, where $K_1 = X - E + (Y_1/\mu)$, $K_2 = X - E + (Y_1/\mu)$, and $K_3 = H - (Y_2/\mu)$. The solution of P is also a classic orthogonal ProCrustes problem. The solutions of H and E are $\Theta_{(Y_3/\mu)}(Z + (Y_2/\mu))$ and $\Omega_{(\lambda_2/\mu)}(X - PQ^T XZ + (Y_1/\mu))$, where Θ and Ω are the singular value thresholding shrinkage and the ℓ_1 minimization operators, respectively. The algorithm framework of solving problem (21) is shown in Algorithm 2.

Algorithm 2 SALPL

```
Input: Training samples matrix $X$; Binary label matrix $Y$; Parameters $\lambda_1$, $\lambda_2$ and $\lambda_3$.
Initialization: $H = \mathbf{0}$; $Q = \mathbf{0}$; $E = \mathbf{0}$; $Z = \mathbf{0}$; $P^* = arg \min_P Tr(P^T(-\Sigma)P),\ s.t.\ P^T P = I$; where $\Sigma$ is the data covariance; $Y_1 = \mathbf{0}$; $Y_2 = \mathbf{0}$; $\mu_{max} = 10^5$; $\rho = 1.01$; $\mu = 0.1$.
while not converged do
  1. Update the variables as (22)-(26);
  2. Update $Y_1$, $Y_2$ and $\mu$ by
     $\begin{cases} Y_1 \leftarrow Y_1 + \mu(X - PQ^T XZ - E) \\ Y_2 \leftarrow Y_2 + \mu(Z - H) \\ \mu \leftarrow \min\{\rho\mu, \mu_{max}\} \end{cases}$
end while
Output: Projection matrix $Q$
```

C. Complexity Analysis

In Algorithms 1 and 2, the main time-consuming components are the following steps:

1) SVD computation in solving H and P;
2) matrix multiplication and inverse in solving Q and Z.

Since the computation complexities of Algorithms 1 and 2 are similar, we take the computation complexity analysis of Algorithm 1, for example, to demonstrate the analysis procedure. Specifically, in Algorithm 1, the computation complexity of solving H and P is about $\mathcal{O}(n^3)$ and $\mathcal{O}(m^2k)$ (k is the number of multiplication in SVD); the computation complexity of solving Q is about $\mathcal{O}(m^3)$. Please note that the computation complexity of solving Z is negligible, since $(I + XX^T)^{-1}$ can be precalculated before going to the loop. Since there are α multiplications, the computation complexity of these operations is about $\alpha\mathcal{O}(m^3)$. Therefore, the total computation complexity of Algorithms 1 and 2 are about $\mathcal{O}\eta(n^3 + m^2k + (\alpha + 1)\ m^3)$, where η is the number of iterations. The computation complexity of LatLRR is about $\mathcal{O}(n^3+m^3)$ [3]. When $m \leq n$, the computation complexities of Algorithm 1 and 2 are approximately equal to that of LatLRR, since the value of k is small.

D. Convergence Analysis

The convergence of ADMM was proved for two blocks [16]. However, Algorithms 1 and 2 are designed for five blocks, and the objective function of augmented Lagrange functions is not convex and thus the convergence properties of algorithms cannot be theoretically guaranteed. Although it is difficult to obtain a strong convergence property of the proposed optimization algorithms, we present a week convergence property of the proposed Algorithms 1 and 2 by showing that under mild conditions, any limit points of the iteration sequence generated by Algorithms 1 and 2 are the stationary points that satisfy the Karush–Kuhn–Tucker (KKT) conditions. 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 algorithms.

Next, we take Algorithm 1 as an example to proof that any limit point of the iteration sequence generated by Algorithm 1 is the stationary point that satisfies the KKT conditions. The proof procedure in Algorithm 2 is similar to that of Algorithm 1.

Let us assume that the proposed algorithm reaches a stationary point. The KKT conditions for (12) 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):

$$\begin{aligned}
&X - XZ - PQ^TX - E = 0, \quad Z - H = 0\\
&\frac{\partial\mathcal{L}}{\partial Q} = \lambda_1 Q - XY_1^TP = 0, \quad \frac{\partial\mathcal{L}}{\partial Z} = Y_2 - X^TY_1 = 0\\
&Y_1 \in \lambda_2\partial_E||E||_1, \quad Y_2 \in \partial_H||H||_*.
\end{aligned} \tag{27}$$

We can obtain the following equation from the second to last one relationship in (27):

$$\begin{aligned}
X - XZ - PQ^TX + \frac{Y_1}{\mu}&\\
&\in X - XZ - PQ^TX + \lambda_2\frac{\partial||X - XZ - PQ^TX||_1}{\mu}\\
&\triangleq \mathcal{Q}_{\frac{\lambda_2}{\mu}}(X - XZ - PQ^TX)
\end{aligned} \tag{28}$$

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

$$E = \mathcal{Q}_{\frac{\lambda_2}{\mu}}^{-1}\left(X - XZ - PQ^TX + \frac{Y_1}{\mu}\right) \tag{29}$$

where $\mathcal{Q}_\beta^{-1}(t) \triangleq \mathcal{S}(t, \beta)$ and $\mathcal{S}$ is the ℓ_1 minimization operation $\mathcal{S}(x, \tau) := \operatorname{sgn}(x)\max(|x| - \tau, 0)$ [29]. Similarly, we can obtain the following equation from the last one relationship in (27):

$$Z + \frac{Y_2}{\mu} = Z + \frac{\partial_H(||H||_*)}{\mu} = Z + \frac{\partial_Z(||Z||_*)}{\mu} \triangleq \Omega_{\frac{1}{\mu}}(Z) \tag{30}$$

where scalar function $\Omega_{(1/\mu)}(t) \triangleq t + (1/\mu)\partial||t||_*$. From [4], we can obtain the following relation:

$$H = \Omega_{\frac{1}{\mu}}^{-1}\left(Z + \frac{Y_2}{\mu}\right) \tag{31}$$

where $\Omega_\beta^{-1}(t) \triangleq \Psi(t, \beta)$ and Ψ is the singular value thresholding, which is computed as

$$\Psi(t, \beta) = US_\beta(\Sigma)V^T \tag{32}$$

where $S_\beta(\Sigma_{ii}) = \operatorname{sgn}(\Sigma_{ii})\max(0, |\Sigma_{ii} - \beta|)$ is the soft-thresholding operator and $t = U\Sigma V^T$ is the SVD of t [29].

Therefore, the KKT condition is as follows:

$$\begin{aligned}
X - XZ - PQ^TX - E = 0,& \quad Z - H = 0\\
\lambda_1 Q - XY_1^TP = 0,& \quad Y_2 - X^TY_1 = 0\\
E = \mathcal{S}&\left(X - XZ - PQ^TX + \frac{Y_1}{\mu}, \frac{\lambda_2}{\mu}\right)\\
H = \Psi&\left(Z + \frac{Y_2}{\mu}, \frac{1}{\mu}\right).
\end{aligned} \tag{33}$$

We can prove that algorithm converges to a point that satisfies the KKT condition.

Theorem 1: Let $\theta \triangleq (H, Q, E, P, Z, Y_1, Y_2)$ 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 proofs of Theorem 1 can be found in the Supplementary Material.

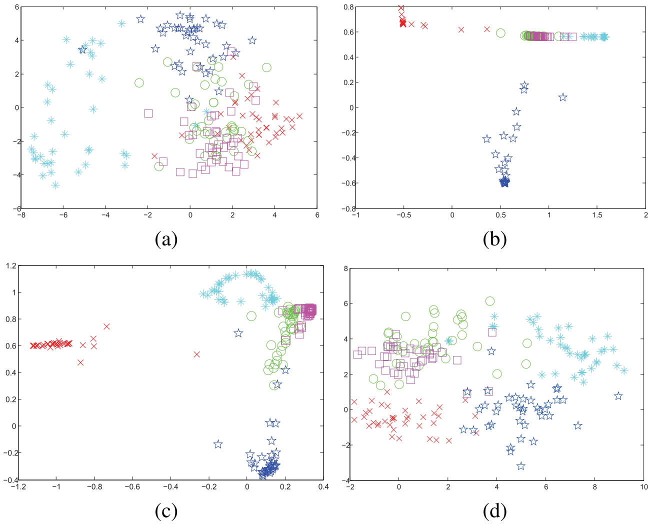


Fig. 1. 2-D projection of digits by using (a) PCA, (b) NPE, (c) LPP, and (d) EALPL in which “x” denotes 0, “*” denotes 1, “o” denotes 2, “□” denotes 3, and “☆” denotes 4.

E. Classification

When problems (10), (11), and (20) are solved, we obtain the projection matrix Q. Then, we directly use Q to obtain the projection results of the training and test data, respectively. Finally, we apply the nearest neighbor (NN) classifier to classify the projection results of test data. We may also use other classifiers to perform classification but is more involved. So we leave it for the future work.

IV. Experiments

In this section, we will first validate the performance of our methods on the digit visualization. Afterward, we will experimentally present the effectiveness of our methods in unsupervised and supervised scenarios. Our code and related data will be released online (http://www.yongxu.org/lunwen.html) if this paper is accepted.

A. Digit Visualization

The experiment involves digit visualization [30]. We use 20×16 images of handwritten digits, which are publicly available at http://www.cs.toronto.edu/ roweis/data. The data set contains 39 samples from each class (digits from “0” to “9”). Each digit image sample is represented lexicographically as a high-dimensional vector of length 320. We use different methods to project the data set in the 2-D space for comparison purpose, and the results are shown in Fig. 1 in which we only use the subset of digits “0”∼“4” to illustrate the experimental results owing to the space limitation. Please note that since the dimension learned by LatLRR is the same as that of the original data, we do not give its digit visualization in this section. For NPE and LPP, we use $k = 6$ to construct the affinity graphs.

Observe that the projection results of PCA are spread out, since the purpose of PCA is to maximize the variance and thus the different classes seem to heavily overlap. On the other hand, NPE and LPP methods obtain more meaningful results, e.g., samples sharing the same class labels are mapped close to each other. This can be explained as follows: NPE and LPP aim at preserving locality. EALPL seems to provide more better result than PCA, NPE, and LPP, since its clusters appear more cohesive, which means that EALPL extracts more discriminant/salient features.

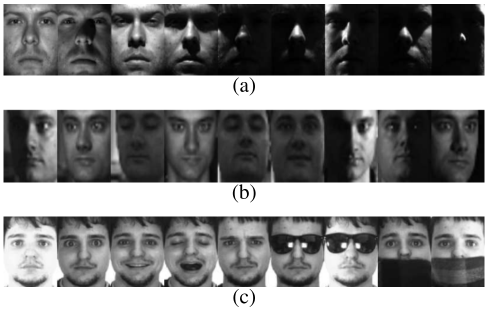


Fig. 2. Some images from (a) YaleB, (b) PIE, and (c) AR databases.

B. Experiments on Real Benchmark Databases for Unsupervised Scenario

In this section, we evaluate our ALPL and EALPL methods on three widely used face databases: Extended YaleB (YaleB) [31], CMU PIE (PIE) [32], and AR [5], [33]. It should be pointed out that the difficulty of face images in these three databases is different. As shown in Fig. 2, YaleB is relatively simple. Each person has about 64 near frontal images under different illuminations. The PIE database is taken under different poses, expressions, and illuminations, and thus it is more difficult for recognition. The challenge of AR database is that it contains different facial expressions, illumination conditions, and occlusions (sun glasses and scarf). We compare our ALPL and EALPL methods with PCA [7], NPE [8], LPP [9], low-rank preserving projections (LRPPs) [34], NMF [10], and LatLRR [3].

1) *YaleB:* YaleB database contains 2414 human face images of 38 subjects. Each subject contains about 64 images taken under different illuminations. Half of the images are corrupted by shadows or reflection. Each image is cropped and resized to 32×32 pixels. We randomly select 10, 20, and 30 training images from each person and the rest for testing.

2) *PIE:* PIE database contains 41 368 face images of 68 persons, each being under 13 different poses, 43 different illumination conditions, and with four different expressions. We select a subset of this database for this experiment, which contains five frontal poses (C05, C07, C09, C27, and C29) and all the images under different illuminations and expressions. Thus, there are 11 554 face images in total and about 170 images for each person. The size of each image is 32×32 pixels. We also randomly select 10, 20, and 30 training samples from each person and the rest for testing.

3) *AR:* AR database contains over 4000 color images corresponding to 126 people’s faces (70 men and 56 women). Each person has 26 face images taken during two sessions. In each session, each person has 13 images, where three images with

TABLE I

Recognition Rate (%) of Different Methods on These Three Databases

	# Tr	PCA	NPE	LPP	NMF	LRPP	LatLRR	ALPL	EALPL
YaleB	10	54.47±0.64(320)	73.40±0.30(190)	62.38±0.26(150)	69.12±0.88(300)	72.20±0.88(160)	68.50±1.16	81.02±1.06(115)	**86.99±0.70(150)**
	20	69.84±0.55(320)	75.09±0.37(370)	78.58±0.46(270)	78.63±0.80(280)	84.76±0.79(160)	82.70±0.56	90.74±0.72(115)	**92.34±0.60(150)**
	30	76.92±0.68(400)	78.10±0.41(390)	86.49±0.50(350)	85.16±0.89(320)	88.10±0.82(160)	87.60±0.72	92.81±0.79(115)	**94.01±0.71(150)**
PIE	10	43.91±0.56(380)	60.72±0.36(280)	52.67±0.45(260)	72.36±0.68(360)	66.67±0.73(320)	63.76±0.70	74.85±0.94(400)	**77.03±0.83(235)**
	20	61.29±0.42(380)	79.04±0.63(320)	76.80±0.65(350)	83.39±0.81(320)	80.84±0.85(320)	81.93±0.40	86.70±0.48(400)	**89.12±0.61(235)**
	30	71.93±0.47(390)	84.47±0.54(350)	88.23±0.37(340)	85.23±0.73(300)	87.60±0.79(320)	88.34±0.35	90.69±0.39(400)	**91.69±0.36(235)**
AR	10	43.87±0.42(260)	71.69±0.68(280)	83.55±0.89(320)	69.86±3.02(110)	62.25±0.76(360)	53.65±1.13	72.53±0.86(270)	**87.64±0.82(330)**
	15	49.09±0.43(210)	86.53±0.78(310)	88.61±0.59(310)	80.02±1.55(110)	80.31±1.12(360)	69.86±1.16	81.76±0.81(270)	**92.71±0.75(330)**
	20	58.33±0.64(230)	91.49±0.81(300)	92.75±0.76(300)	86.24±1.50(110)	88.90±0.78(360)	79.73±1.62	87.65±1.67(270)	**95.38±0.98(330)**

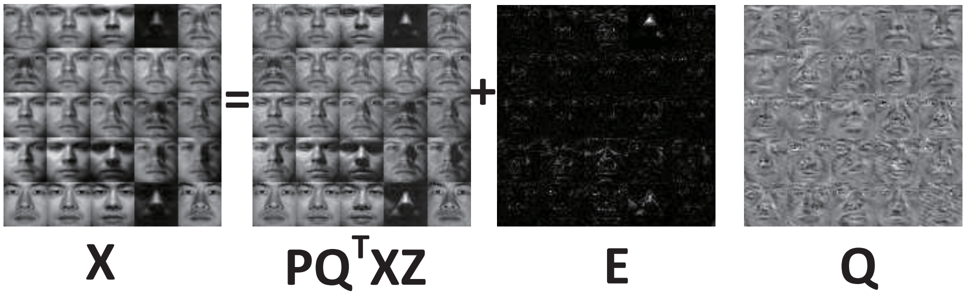


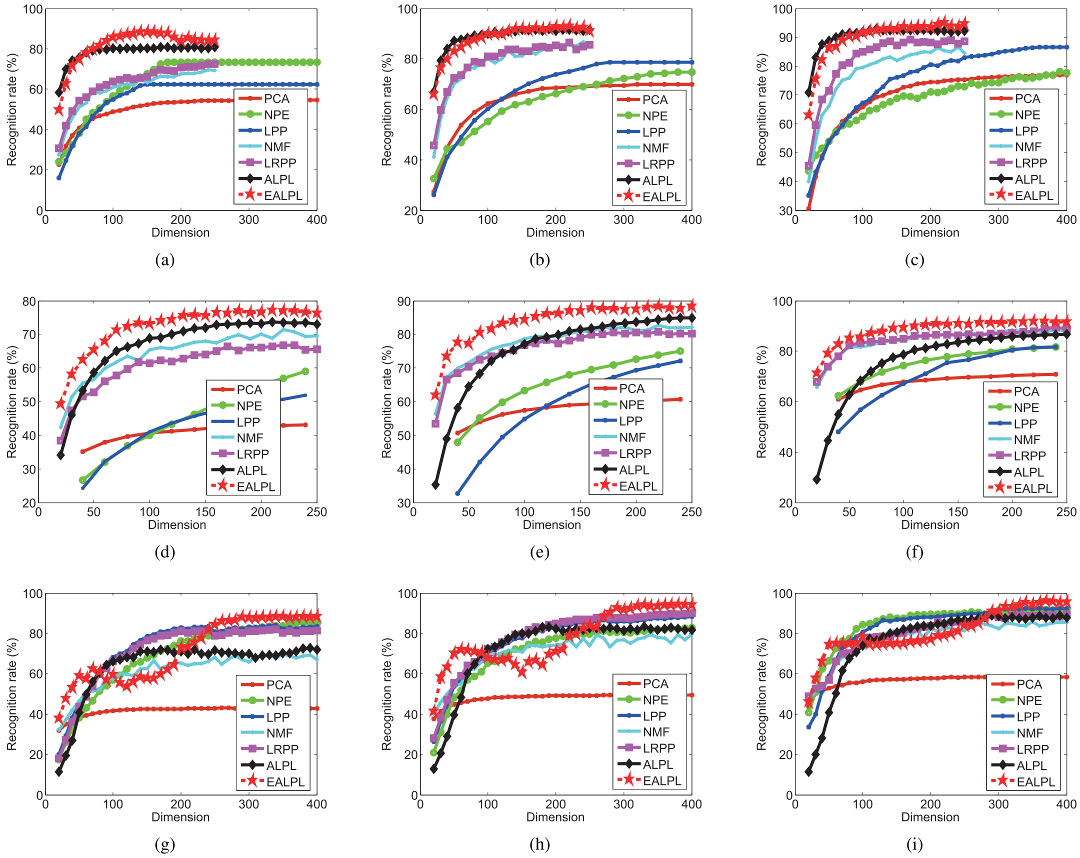
Fig. 3. Some samples of using EALPL to correct the errors in the Extended YaleB data set. Left: original data matrix X. Middle: corrected data PQ^TXZ. Right: error E. Right-most: basis vectors of matrix Q.

sunglasses, another three with scarfs, and the rest seven are different facial expressions and illumination conditions. The original images are of 165×120 pixels. As [5] did, we crop and resize each image to 540 pixels. In this experiment, we select a subset of the database consisting of 2600 images from 50 female and 50 male subjects. We randomly select 10, 15, and 20 training samples from each person and the rest for testing.

Every experiment runs ten times and then the mean recognition rate and standard deviation (%) are reported. Table I shows the recognition results of different methods on these three databases (the last number in parentheses are the optimal dimensions). For NPE and LPP, we tune the number of neighbors from [5, 10]. In our experiments, we define the adjacency graph for LPP by using the heat kernel and tune the heat kernel parameter by using the grid search technique.

From the results in Table I, we obtain three interesting observations. First, recognition rate in the projected space of our ALPL and EALPL tends to outperform PCA, NPE, LPP, and NMF under different experimental settings, and the another superiority is that when our ALPL and EALPL obtain the best recognition results, the projected dimension is generally lower than NPE, LPP, and NMF. For example, the EALPL method is able to project the original data to a subspace with quite low dimensions on the Extended YaleB and AR databases. Such low-dimensional subspace projected by EALPL gains an advantage over that obtained by the other methods with higher dimensions. Second, we also observe that ALPL and EALPL outperform LatLRR with more small number of dimensions on these three data sets (please note that the subspace dimension of LatLRR is as the same as that of the original data). This is because although LatLRR aims to extract the salient features, ALPL and EALPL can select the more flexible dimensions for the extracted salient features and thus the extracted features are more effective than that learned by LatLRR. In addition, EALPL emphasizes to treat the PQ^TXZ as a whole which encourages them to boost mutually during the feature extraction and thus, the dimension reduction process is more efficient and effective. Finally, since our method is somewhat similar to LRPP, we give the comparison of our methods and LRPP. We can see that EALPL consistency beats LRPP and the improvement of recognition rate is evident. The reason may be that LRPP only emphasis the low-rank embedding, whereas EALPL not only emphasis low-rank embedding but also extracts the salient features and thus is competent to perform recognition.

EALPL decomposes original face images into the "clean" parts PQ^TXZ and a sparse error part E fitting noise. Fig. 3 shows the performance of images recovered by EALPL, in which these images with shadow are approximately recovered. In other words, the shadow can be slightly removed and used as the error part. The basis vectors of matrix Q learned by EALPL are shown in Fig. 3 (right-most) in which each basis vector has the dimension of 1024. We plot these basis vectors as 32 × 32 gray scale images. We can see that the basis vectors of matrix Q contain lots of details of face images, which urges EALPL to extract more effective salient and discriminative features. To clearly show the performance of different methods with different dimensions, we run them only once and plot the recognition rate versus the variations of the dimension in Fig. 4 in which the dimension (i.e., the horizontal axis) means the number of the column vectors in the projection matrix Q used for feature extraction. As can be seen from Fig. 4, EALPL obtains the best recognition results in the experiment. Especially, in YaleB and PIE databases, our EALPL is able to project the original data into a subspace with

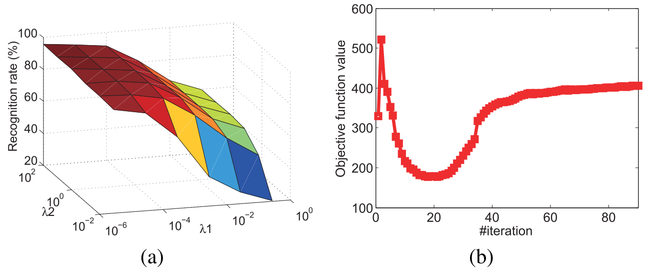


(a) (b) (c) (d) (e) (f) (g) (h) (i)

Fig. 4. Recognition rate (%) versus the dimensions on these three databases. (a) YaleB (# Tr = 10). (b) YaleB (# Tr = 20). (c) YaleB (# Tr = 30). (d) PIE (# Tr = 10). (e) PIE (# Tr = 20). (f) PIE (# Tr = 30). (g) AR (# Tr = 10). (h) AR (# Tr = 15). (i) AR (# Tr = 20).

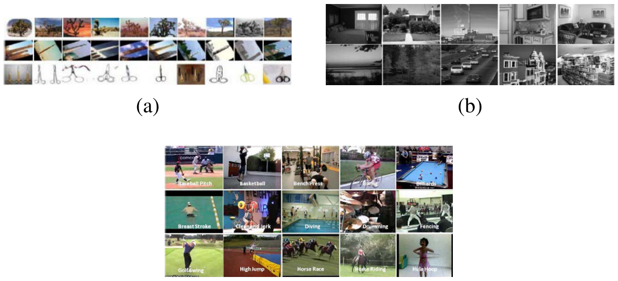
quite lower dimensions and such lower dimensions projected by EALPL would even outperform that obtained by the other methods with high dimensions. We also observe that EALPL obtains dissatisfied recognition results when the number of dimensions is small in the AR database. This is because more noisy data occur in this database such as occlusion and illumination, and thus, the extracted salient features may be inaccuracy in such low dimension. As dimensions increase, EALPL can extract more discriminative features. As a result, EALPL obtains the best recognition results in this database with somewhat high dimension. The ALPL method also obtains the better recognition results in most cases. However, in the AR database, the superiority of ALPL is not evident. The reason may be that ALPL separately learns XZ and PQ^TX such that ALPL cannot extract the effective salient features in this database.

1) Parameters Sensitivity and Algorithm Convergence of EALPL: The variations of the parameters versus the recognition rates of EALPL in the AR data set based on a single run are shown in Fig. 5(a), which shows EALPL is very robust to the value of λ_2 in large range, i.e., $\lambda_2 \in [10^{-2}, 10^2]$, whereas EALPL obtains the best performance when the range of $\lambda 1$ is small, i.e., $\lambda_1 \in [10^{-6}, 10^{-4}]$. In other words, EALPL effectively uses the sparse error part to compensate noise when $\lambda_2 \in [10^{-2}, 10^2]$. Unfortunately, EALPL is not robust to λ_1. How to identify the optimal values for parameters is data-dependent and still an open problem. In our experiments, we adopt the grid search technique to seek the optimal values for λ_1 and λ_2. We also empirically show the convergence curve of objective function values versus iterations in Fig. 5(b). We can see that the objective function value has a violent vibration in the first few iterations. This phenomenon can be interpreted as the consequence of the inexact solutions of Q and Z. (In practical, the exact solution is permutated a little in our method by adding a Tikhonov regularization λI to the inverse of the matrices for the stable solutions. For example, in Algorithm 1, we set $Q = (\lambda I + \lambda_1\ I + \mu X X^T)^{-1}(\mu X G_2^T P)$, where $\lambda = 0.01$.) When λ is larger, the vibration is more violent. The curve of objective function value finally reaches stabilization after several iterations, which indicates that EALPL has a good convergence property.



(a) (b)

Fig. 5. Parameters sensibility and algorithmic convergence of EALPL in the AR data set. (a) Variations of recognition rate versus the parameters $\lambda1$ and $\lambda2$ (# Tr = 20). (b) Convergence curve versus iterations (# Tr = 20).



(a) (b)

(c)

Fig. 6. Some images from (a) Caltech101, (b) 15-Scene, and (c) UCF50 databases.

Similar parameters' sensibility and algorithmic convergence can also be found in other databases used in this paper.

Here, we should give a clarification about the sensitivity of parameter λ_1. In our experiments, we always set $\lambda = 0.01$ so as to obtain the stable solution for Q. However, when we set $\lambda_1 = 0.01$, we find that the value of λ is dominative. Thus, the value of λ_1 in our experiments may be a biased estimation, i.e., $\lambda_1 \notin [10^{-6}, 10^{-4}]$. It is well known that seeking the optimal parameter for algorithm is very time-consuming and expensive. Therefore, we first fix the value of λ ($\lambda = 0.01$) and then seek the optimal value of λ_1, which is less time-consuming than the way of seeking the optimal values for λ and λ_1 simultaneously.

C. Experiments on Real Benchmark Databases for Supervised Scenario

In this section, we conduct extensive experiments on face, objective, scene, and action recognitions to validate the effectiveness of features extracted by SALPL. We first evaluate SALPL on YaleB and PIE face databases. We then test SALPL on three more different types of databases: Caltech 101 database for object recognition [24], 15-Scene Categories for scene recognition [1], and UCF50 action database for action recognition [24], [35]. Fig. 6 shows some images from the last three databases. We compare SALPL with many SR- and LRR-based methods: SRC [5], CRC [14], locality-constrained linear coding (LLC) [36], LRC [13], LRSIC [37], LRRC [38], SLRRC [38], TDDL [23], LatLRR [3], LC-KSVD [24], and ILRDFL [26]. For LatLRR, the features of LX are feed into the multivariate rigid regression for model parameter prediction. Then, the learned model parameter is used to project the data and the recognition results are obtained by using the NN classifier.

1) Face Recognition: For YaleB and PIE face databases, we randomly select 10, 15, 20, and 25 images per person as training set and remaining samples are used for testing and this process is repeated ten times and then mean recognition rates are reported in Tables II and III, respectively. In Table II, for these methods of 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 [38]. Following [38], the dictionary size for LRRC, SLRRC, and TDDL is set to 140, i.e., each person has five atoms, respectively. In Table III, 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.

TABLE II
RECOGNITION RATES (%) OF DIFFERENT METHODS ON THE YALEB DATABASE

Method	10	15	20	25
SRC [5]	87.8	92.6	94.4	96.7
CRC [14]	86.1	90.7	93.0	94.1
LLC [36]	79.8	88.6	91.5	94.3
LRC [13]	83.3	89.4	92.4	93.6
LRSIC [37]	87.0	92.7	94.2	96.1
LRRC [38]	84.3	91.5	93.3	95.8
SLRRC [38]	85.5	91.4	94.0	95.6
TDDL [23]	84.3	88.9	92.5	95.0
ILRDFL [26]	86.8	91.3	94.1	95.5
LatLRR [3]	84.0	88.8	92.1	93.8
SALPL	**88.3**	**93.3**	**95.8**	**97.1**

TABLE III
RECOGNITION RATES (%) OF DIFFERENT METHODS ON THE PIE DATABASE

Method	10	15	20	25
SRC [5]	76.4	88.1	90.2	93.4
CRC [14]	83.8	88.3	91.0	93.2
LLC [36]	77.4	84.6	89.9	93.2
LRC [13]	75.9	85.0	90.5	92.3
LRSIC [37]	82.5	87.5	90.6	93.2
LRRC [38]	79.8	85.5	90.1	91.0
LRRC [38]	80.8	86.7	89.6	91.8
TDDL [23]	78.8	85.5	88.7	91.4
ILRDFL [26]	86.4	90.2	93.0	94.1
LatLRR [3]	80.4	86.7	90.0	91.3
SALPL	**88.0**	**92.1**	**94.1**	**95.2**

From the results in these two tables, we can see that SALPL achieves the best recognition results and outperforms the compared methods. The superiority is very obvious in Table III. For example, when the number of training samples of each person is 10, 15, and 20, SALPL makes about 1.6%, 1.9%, and 1.1% improvement compared with the second best methods, respectively.

2) Object Recognition: We use the Caltech 101 database to test SALPL for object recognition. The Caltech 101 database contains over 9144 images from 102 classes. 101 distinct classes are of animals, flowers, trees, and so on and there is a background class. Each class contains about 31–800 images. The size of each image is roughly 300×200 pixels. As [24] did, the spatial pyramid feature is used in our experiment. Since the feature dimension is too high, PCA is used to reduce the feature dimension to 1500. In this experiment, we randomly select 5, 10, 15, 20, 25, and 30 samples per class as the training set and the others for testing. Every experiment runs ten times and then the mean recognition rates (%) are reported in Table IV. For fairness, all methods use the spatial pyramid features. The dictionary size of SRC, LRSIC, and LRRC is set to 3060, i.e., for 30 dictionary items per class.

TABLE IV

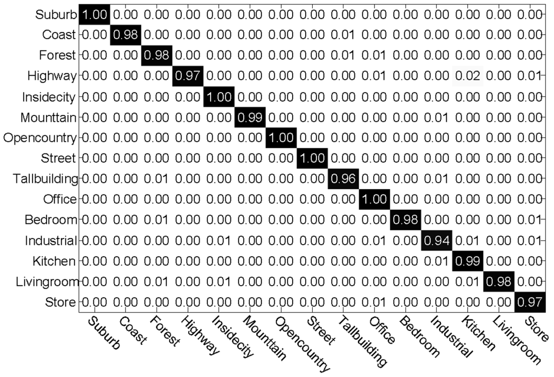
RECOGNITION RATES (%) OF DIFFERENT METHODS ON THE CALTECH 101 DATABASE ($LRRC^2$ IS THE METHOD OF LRRC WITHOUT Q)

Number of train	5	10	15	20	25	30
Malik [39]	46.6	55.8	59.1	62.0	-	66.20
Lazebnik [1]	-	-	56.4	-	-	64.6
Griffin [40]	44.2	54.5	59.0	63.3	65.8	67.60
Irani [41]	-	-	65.0	-	-	70.40
Grauman [42]	-	-	61.0	-	-	69.10
Pham [43]	-	-	42.0	-	-	-
Getmert [44]	-	-	-	-	-	64.16
Yang [45]	-	-	67.0	-	-	73.20
LLC [36]	51.15	59.77	65.43	67.74	70.16	73.44
SRC [5]	48.8	60.1	64.9	67.7	69.2	70.7
K-SVD [46]	49.8	59.8	65.2	68.7	71.0	73.2
D-KSVD [47]	49.6	59.5	65.1	68.6	71.1	73.0
LC-KSVD1 [24]	53.5	61.9	66.8	70.3	72.1	73.4
LC-KSVD2 [24]	54.0	63.1	67.7	70.5	72.3	73.6
LCLE-DL [48]	-	61.6	-	-	-	-
ILRDFL [26]	-	-	-	-	-	72.6
LRRC [38]	-	-	66.1	-	-	73.6
$LRRC^2$ [38]	-	-	65.5	-	-	73.3
LRSIC [37]	-	-	58.3	-	-	65.7
SALPL	**56.3**	**64.4**	**68.3**	**72.0**	**74.1**	**76.0**

TABLE V

RECOGNITION RATES (%) OF DIFFERENT METHODS ON THE 15-SCENE CATEGORIES DATABASE

Method	R. rate	Method	R. rate
SRC [5]	91.8	TDDL [23]	92.1
CRC [14]	92.3	LatLRR [3]	91.5
LLC [36]	79.4	Lazebnik [1]	81.4
LLC* [36]	89.2	Yang [45]	80.3
LRC [13]	92.3	Lian [50]	86.4
LRSIC [37]	92.4	Boureau [51]	84.3
LRRC [38]	90.1	Gemert [44]	76.7
SLRRC [38]	91.3	LC-KSVD2 [24]	92.9
LC-KSVD1 [24]	90.4	SALPL	**98.3**



	Suburb	Coast	Forest	Highway	Insidecity	Mountain	Opencountry	Street	Tallbuilding	Office	Bedroom	Industrial	Kitchen	Livingroom	Store
Suburb	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Coast	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00
Forest	0.00	0.00	0.98	0.00	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00
Highway	0.00	0.00	0.00	0.97	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.02	0.00	0.01
Insidecity	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Mounttain	0.00	0.00	0.00	0.00	0.00	0.99	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00
Opencountry	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Street	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Tallbuilding	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.96	0.00	0.00	0.01	0.00	0.00	0.00
Office	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00
Bedroom	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.98	0.00	0.00	0.00	0.01
Industrial	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.01	0.00	0.94	0.01	0.00	0.01
Kitchen	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.99	0.00	0.00
Livingroom	0.00	0.00	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.98	0.00
Store	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.97

Fig. 7. Confusion matrix for the 15-Scene Categories database.

TABLE VI

RECOGNITION RATES (%) OF DIFFERENT METHODS ON THE UCF50 DATABASE

Method	R. rate	Method	R. rate
SRC [5]	61.4	LRRC [38]	54.3
JDL [53]	53.5	SLRRC [38]	56.9
LLC* [36]	57.0	Oliva and Torralba [55]	38.8
LRC [13]	57.6	Laptev and Wang. [56], [57]	47.9
LRSIC [37]	58.6	Sadanand and Corso [52]	57.9
LC-KSVD [24]	53.6	FDDL [58]	61.1
COPAR [54]	52.5	SALPL	**63.0**

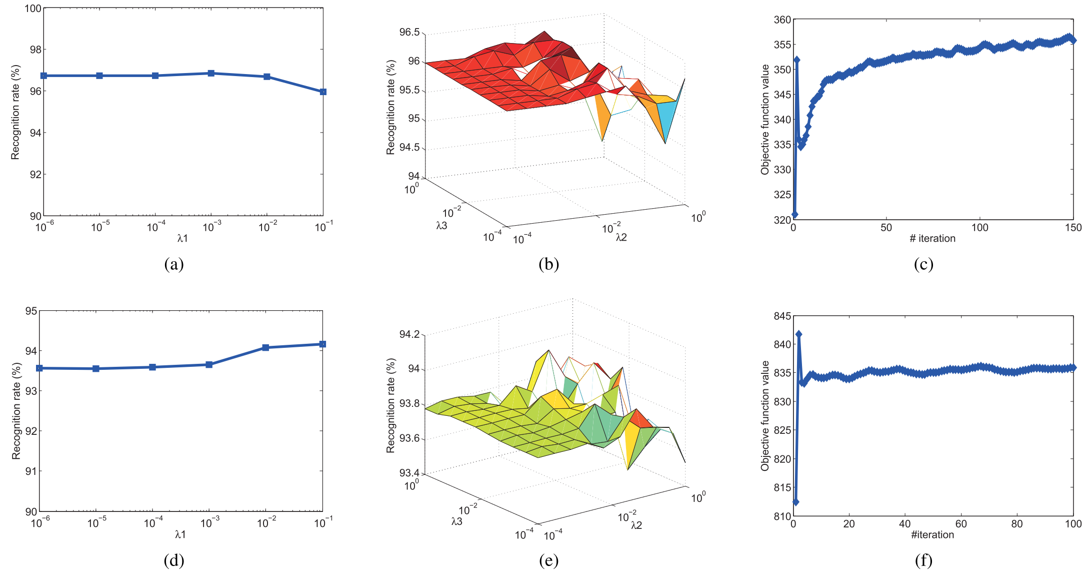
As can be seen from Table IV, SALPL performs the best among all the compared methods and has about (2%–4%) improvement over the runner-up under different cases. We also note that a total of 17 classes achieve 100% recognition rate when we select 30 images per class as training set.

3) Scene Recognition: We test our method on the 15-Scene Categories database for scene recognition. This database contains 15 natural scene categories that expand on the 13 category database released in [49]. It contains 4485 images falling into 15 categories, such as bedrooms, kitchens, streets, and country scenes. Each category has 200–400 images.

In this experiment, we use the feature data of the 15-Scene Categories database provided in [24]. As [1] did, we randomly select 100 images as training samples and use the remaining as testing samples. It should be pointed out that, as [24] did, LLC is also the original LLC, which uses sparse coding to encode SIFT descriptors, while LLC* uses sparse coding to encode the spatial pyramid feature. The dictionary size of SRC, CRC, LRC, LRSIC, LRRC, and SLRRC is all 450. LLC and LLC* both have 30 neighborhoods. We report the mean recognition rate for our method over 10 runs in Table V. Again, SALPL performs the best among all the competitors. Specifically, SALPL outperforms the second best competitor LC-KSVD2 by margin of 5.5%. Fig. 7 gives the confusion matrix of our method on this database, where the recognition rate for each class is among the diagonal. All classes are classified well and the worst recognition rate is as high as 94%.

4) Action Recognition: In this section, we evaluate our method on the UCF50 database for action recognition [35]. The UCF50 database is one of the largest action recognition databases consisting of realistic taken from Youtube. It contains 50 action categories with a total of 6617 action videos and the categories are of basketball shooting, baseball pitch, diving, biking, tennis swing, and so on. In this experiment, we use the action feature representation presented in [52], whose code and feature data can be downloaded from http://web.eecs.umich.edu/~jjcorso/. We use PCA to reduce the feature dimension to 3000 for computational efficiency. Following the common experiment settings, we test different methods using fivefold groupwise cross-validation methodology. The dictionary size of SRC, CRC, LRC, and LRSIC is set to 1500, i.e., 30 dictionary atoms for each category. LLC* uses the original LLC method to encode the action feature and the neighborhood number is 30. Table VI gives the comparison results. It can be seen that our method outperforms other methods and makes about 1.6% improvement over the follow-up SRC.

5) Parameters Sensitivity and Algorithm Convergence of SALPL: There are several regularization parameters affecting the performance of SALPL. In the following, we study the influence of parameters λ_1, λ_2, and λ_3 by examining the variability of SALPL recognition performance with different values of these parameters. We choose the YaleB and PIE databases as the test data set. The results are visualized



(a) (b) (c)

(d) (e) (f)

Fig. 8. Recognition rate (%) versus the dimensions and objective function versus the number of iterations on (a)–(c) YabeB and (d)–(f) PIE databases in which we select 20 images per subject for training and remaining are used for testing.

in Fig. 8. As can be seen, SALPL is very robust to the values of λ_1. When $\lambda_3 \in [10^{-4}, 10^{0}]$ and $\lambda_2 \in [10^{-4}, 10^{-2}]$, SALPL always obtains the best recognition results. In SALPL, λ_3 controls the value of the rank of Z. We observe that the value of λ_3 is small. The reason may be that when $Q^T X$ preserves the main information of data for recognition, $PQ^T X$ may not respect the structure of original data well. In this way, the block-diagonal structure of matrix Z is not obvious and thus the value of the rank of Z is not small. Similarly, when SALPL obtains the best recognition results, the value of λ_2 is small, which is reasonable, since E should not be very sparse when it can effectively compensate noise.

The KKT conditions of problem (12) are given by (33). Based on (33), we check the following criterion for algorithmic stopping:

$$\text{norm}(X - PQ^T XZ - E, \text{Inf}) \le \epsilon \ \&\& \ \text{norm}(Z - H, \text{Inf}) \le \epsilon$$

for an appropriate tolerance, e.g., $\epsilon = 10^{-6}$. We plot the convergence curves of objective function values with respect to the number of iterations in Fig. 8. Although the objective function values have a violent vibration in the first few iterations, they eventually reach steadily as the iteration goes on. This indicates that SALPL eventually converges a point that satisfies the KKT condition.

D. Limitation

The limitation of the proposed methods is that the mathematical programming formulation is nonconvex. Although we provide an ADMM-style algorithm for solving this math program, weak convergence properties are presented. However, we note that convergence curves of EALPL on the AR data set [see Fig. 5(b)], SALPL on the Yale B data set [see Fig. 8(c)], and SALPL on the PIE data set [see Fig. 8(f)] still slowly ascend or fluctuate when the number of iterations is large. This may indicate that standard ADMM cannot guarantee that the objective value of ALPL, EALPL, and SALPL converges or the sequences of $\{H, Q, E, P, Z, Y_1, Y_2\}$ have limit value when there are more than two variables in ALPL, EALPL, and SALPL. In practice, the conditions of Algorithm 1 are somewhat strong, and thus, it may not be always satisfied in many practical cases.

To the best of our knowledge, a formal theory proof of convergence behavior for such optimization is still missing. Thus, under mild conditions, Theorem 1 is still efficient, i.e., limit points of the iteration sequence generated by Algorithms 1 and 2 are the stationary points that satisfy the KKT conditions. The experimental results show that the proposed methods can achieve good recognition results in practical by using the ADMM-style algorithm. Exploring other convex relaxations for our objective function is our future work.

V. Conclusion

This paper proposes three novel feature extraction algorithms based on the LatLRR, including ALPL, EALPL, and SALPL. ALPL and EALPL can address some intrinsic problems of LatLRR, and SALPL expects to handle supervised problem by combining with rigid regression. These proposed algorithms are examined on different data sets, and experimental results indicate that our proposed algorithms perform better than the existing algorithms. Although the experimental results are remarkable, the computationally more efficient algorithms are still required for their real-world applications. In addition, the question of how to choose the optimal parameters combination needs further investigation.

References

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

[2] X. Fang, S. Teng, Z. He, S. Xie, and W. Wong, "Robust latent subspace learning for image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2017.2693221.

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

[4] H. Zou, T. Hastie, and R. Tibshirani, "Sparse principal component analysis," *J. Comput. Graph. Stat.*, vol. 15, no. 2, pp. 265–286, Jun. 2006.

[5] 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.

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

[7] X. Ma, F. Zhang, Y. Li, and J. Feng, "Robust sparse representation based face recognition in an adaptive weighted spatial pyramid structure," *Sci. China Inf. Sci.*, vol. 61, pp. 012101-1–012101-13, Jan. 2018.

[8] X. He, D. Cai, S. Yan, and H. Zhang, "Neighborhood preserving embedding," in *Proc. Int. Conf. Comput. Vis.*, Oct. 2005, pp. 17–21.

[9] 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.

[10] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1548–1560, Aug. 2011.

[11] R. Fisher, "The use of multiple measures in taxonomic problems," *Ann. Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[12] D. Wang, S. Hoi, Y. He, J. Zhu, T. Mei, and J. Luo, "Retrieval-based face annotation by weak label regularized local coordinate coding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 550–563, Mar. 2014.

[13] 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.

[14] L. Zhang, M. Yang, and X. C. Feng, "Sparse representation or collaborative representation: Which helps face recognition?" in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 471–478.

[15] H. Liu, J. Qin, F. Sun, and D. Guo, "Extreme kernel sparse learning for tactile object recognition," *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4509–4520, Dec. 2017, doi: 10.1109/TCYB.2016.2614809.

[16] 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.

[17] K. Tang, R. Liu, Z. Su, and J. Zhang, "Structure-constrained low-rank representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 22, no. 7, pp. 1149–1161, Dec. 2014.

[18] L. Zhuang, H. Gao, J. Tang, Z. Lin, Y. Ma, and N. Yu, "Constructing a nonnegative low-rank and sparse graph with data-adaptive features," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3717–3728, Nov. 2015.

[19] A. Wagner, J. Wright, A. Ganesh, Z. Zhou, and Y. Ma, "Towards a practical face recognition system: Robust registration and illumination by sparse representation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 597–604.

[20] Y. Peng, A. Ganesh, J. Wright, W. Xu, and Y. Ma, "RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2233–2246, Nov. 2012.

[21] L. Zhuang, T.-H. Chan, A. Y. Yang, S. S. Sastry, and Y. Ma, "Sparse illumination learning and transfer for single-sample face recognition with image corruption and misalignment," *Int. J. Comput. Vis.*, vol. 114, no. 2, pp. 272–287, 2014.

[22] Y. Xu, D. Zhang, J. Yang, and J.-Y. Yang, "A two-phase test sample sparse representation method for use with face recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 9, pp. 1255–1262, Sep. 2011.

[23] 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.

[24] 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.

[25] E. J. Cand0és, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 3, p. 11, 2011.

[26] 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.

[27] H. Zhou, T. Hastie, and R. Tibshirani, "Sparse principle component analysis," *J. Comput. Graph. Stat.*, vol. 15, no. 2, pp. 265–286, 2006.

[28] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Scientific, 1999.

[29] M. Yin, J. Gao, Z. Lin, Q. Shi, and Y. Guo, "Dual graph regularized latent low-rank representation for subspace clustering," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 4918–4933, Dec. 2015.

[30] E. Kokiopoulou and Y. Saad, "Orthogonal neighborhood preserving projections: A projection-based dimensionality reduction technique," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 12, pp. 2143–2156, Dec. 2007.

[31] A. S. Georghiades, P. N. Belhumeur, and D. J. 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.

[32] 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.

[33] A. M. Martinez and R. Benavente, "The AR face database," CVC, Tech. Rep. 24, Jun. 1998.

[34] Y. Lu, Z. Lai, Y. Xu, X. Li, D. Zhang, and C. Yuan, "Low-rank preserving projections," *IEEE Trans. Cybern.*, vol. 46, no. 8, pp. 1900–1913, Aug. 2016.

[35] K. Reddy and M. Shah, "Recognitizing 50 human action categories of Web videos," *Mach. Vis. Appl.*, vol. 24, no. 5, pp. 971–981, 2013.

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

[37] C. Chen, C. Wei, and Y. Wang, "low-rank matrix recovery with structural incoherence for robust face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2618–2625.

[38] Y. Zhang, Z. Jing, and L. S. Davis, "Learning structured low-rank representations for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 281–288.

[39] H. Zhang, A. C. Berg, M. Maire, and J. Malik, "SVM-KNN: Discriminative nearest neighbor classification for visual category recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2006, pp. 2126–2136.

[40] G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category dataset," California Inst. Technol., Pasadena, CA, USA, Tech. Rep. 7694, 2007.

[41] O. Boiman, E. Shechtman, and M. Irani, "In defense of nearest-neighbor based image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[42] P. Jain, B. Kullis, and K. Grauman, "Fast image search for learned metrics," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[43] D. Pham and S. Venkatesh, "Joint learning and dictionary construction for pattern recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[44] J. C. Van Gemert, J. M. Geusebroek, C. J. Veenman, and A. W. M. Smeulders, "Kernel codebooks for scene categorization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Oct. 2008, pp. 696–709.

[45] 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.*, Jun. 2009, pp. 1794–1801.

[46] 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.

[47] 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.

[48] Z. Li, Z. Lai, Y. Xu, J. Yang, and D. Zhang, "A locality-constrained and label embedding dictionary learning algorithm for image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 2, pp. 278–293, Feb. 2017.

[49] F. F. Li and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 524–531.

[50] X. Lian, Z. Lin, B. Lu, and L. Zhang, "Max-margin dictionary learning for multiclass image categorization," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 157–170.

[51] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2559–2566.

[52] 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.

[53] M. Zhou *et al.*, "'Nonparametric Bayesian dictionary learning for analysis of noisy and incomplete images," *IEEE Trans. Image Process.*, vol. 21, no. 1, pp. 130–144, Jan. 2012.

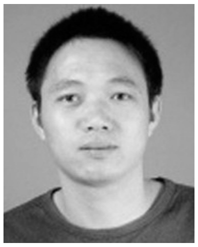
[54] S. Kong and D. Wang, "A dictionary learning approach for classification: Separating the particularity and the commonality," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 186–199.

[55] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the sparial envelope," *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, 2001.

[56] I. Laptev, "On space-time interest points," *Int. J. Comput. Vis.*, vol. 64, nos. 2–3, pp. 107–123, 2005.

[57] H. Wang, M. M. UIIah, A. Klaser, I. Laptev, and C. Schmid, "Evaluation of local spatio-temporal features for action recognition," in *Proc. Brit. Mach. Vis. Conf.*, 2009, pp. 1–11.

[58] 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.



Xiaozhao Fang (S'15–M'17) received the M.S. degree and the Ph.D. degree in computer science and technology from 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, data mining, and machine learning.

Na Han received the B.S. degree in computer science and technology from the Harbin Institute of Technology, Shenzhen, China, in 2004. She is currently pursuing the Ph.D. degree in computer science and technology with the School of Computer Science and Technology, Guangdong University of Technology, Guangzhou, China.

Her current research interests include pattern recognition and machine learning.

Jigang Wu (M'10) received the B.Sc. degree from Lanzhou University, Lanzhou, China, in 1983, and the Ph.D. degree from the University of Science and Technology of China, Hefei, China, in 2000.

He was a Research Fellow with the Center for High Performance Embedded Systems, Nanyang Technological University, Singapore, from 2000 to 2010. He was a Dean and Tianjin Distinguished Professor with the School of Computer Science and Sofware, Tianjin Polytechnic University, Tianjin, China, from 2010 to 2015. He is currently a Distinguished Professor with the School of Computer Science and Technology, Guangdong University of Technology, Guangzhou, China. He has authored over 200 papers in the IEEE TRANSACTIONS ON COMPUTERS, the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION SYSTEMS, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, the *Journal of Parallel and Distributed Computing*, *Parallel Computing*, the *Journal of Scientific Achievements*, and international conferences. His current research interests include network computing, cloud computing, machine intelligence, and reconfigurable architecture.

Dr. Wu serves in the China Computer Federation as a Technical Committee Member in the branch committees, high-performance computing, theoretical computer science, and fault-tolerant computing.

Yong Xu (M'06–SM'15) was born in Sichuan, China, in 1972. He received the B.S., M.S. and Ph.D. degrees from the Nanjing University of Science and Technology, Nanjing, China, in 1994, 1997, and 2005, respectively, all in pattern recognition and intelligence systems.

He is currently with the Shenzhen Graduate School, Harbin Institute of Technology, Harbin, China. His current research interests include pattern recognition, biometrics, machine learning, and video analysis.

Jian Yang (M'08) received the B.S. degree in mathematics from Xuzhou Normal University, Xuzhou, China, in 1995, the M.S. degree in applied mathematics from Changsha Railway University, Changsha, China, in 1998, and the Ph.D. degree in pattern recognition and intelligence systems from the Nanjing University of Science and Technology (NUST), Nanjing, China, in 2002.

In 2003, he was a Post-Doctoral Researcher with the University of Zaragoza, Zaragoza, Spain. From 2004 to 2006, he was a Post-Doctoral Fellow with the Biometrics Centre, The Hong Kong Polytechnic University, Hong Kong. From 2006 to 2007, he was a Post-Doctoral Fellow with the Department of Computer Science, New Jersey Institute of Technology, Newark, NJ, USA. He is currently a Professor with the School of Computer Science and Technology, NUST. He has authored over 80 scientific papers in pattern recognition and computer vision. He has over 2000 ISI Web of Science and 4000 Google Scholar citations. His current research interests include pattern recognition, computer vision, and machine learning.

Dr. Yang was a recipient of the RyC Program Research Fellowship sponsored by the Spanish Ministry of Science and Technology in 2003.

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, 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*, *Pattern Recognition*, the *International Journal of Production Economics*, the *European Journal of Operational Research*, the *International Journal of Production Research*, *Computers in Industry*, and the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, among others. His current research interests include artificial intelligence, pattern recognition, and optimization of manufacturing scheduling, planning, and control.

Xuelong Li (M'02–SM'07–F'12) is currently a Full Professor with the Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, China, and with the University of Chinese Academy of Sciences, Beijing, China.