Hindawi
Complexity
Volume 2019, Article ID 7835797, 17 pages
https://doi.org/10.1155/2019/7835797

Research Article

CoLR: Classification-Oriented Local Representation for Image Recognition

Tan Guo,[1] **Lei Zhang**,[2] **Xiaoheng Tan**,[2] **Liu Yang**,[1] **Zhiwei Guo**,[3] **and Fupeng Wei**[4]

[1]*School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*
[2]*School of Microelectronics and Communication Engineering, Chongqing University, Chongqing 400044, China*
[3]*Chongqing Engineering Laboratory for Detection, Control and Integrated System, Chongqing Technology and Business University, Chongqing 400067, China*
[4]*School of Instrumentation Science and Engineering, Harbin Institute of Technology, Harbin 150001, China*

Correspondence should be addressed to Zhiwei Guo; zwguo@ctbu.edu.cn and Fupeng Wei; weifupeng@yeah.net

Received 28 February 2019; Revised 20 May 2019; Accepted 29 May 2019; Published 20 June 2019

Academic Editor: Michele Scarpiniti

Naïve sparse representation has stability problem due to its unsupervised nature, which is not preferred for classification tasks. For this problem, this paper presents a novel representation learning method named classification-oriented local representation (CoLR) for image recognition. The core idea of CoLR is to find the most relevant training classes and samples with test sample by taking the merits of class-wise sparseness weighting, sample locality, and label prior. The proposed representation strategy can not only promote a classification-oriented representation, but also boost a locality adaptive representation within the selected training classes. The CoLR model is efficiently solved by Augmented Lagrange Multiplier (ALM) scheme based on a variable splitting strategy. Then, the performance of the proposed model is evaluated on benchmark face datasets and deep object features. Specifically, the deep features of the object dataset are obtained by a well-trained convolutional neural network (CNN) with five convolutional layers and three fully connected layers on the challenging ImageNet. Extensive experiments verify the superiority of CoLR in comparison with some state-of-the-art models.

1. Introduction

Image recognition is a fundamental problem in pattern recognition community. Industrial prospects and research interests in image recognition have been motivated by a wide range of real-world applications [1–3]. There are mainly two components for a common image recognition system: (1) robust and discriminant feature learning [4–7], such as Gabor wavelet-based features [8] and local binary pattern (LBP) [9]. The linear holistic appearance-based approaches, such as principal component analysis (PCA) [10] and linear discriminative analysis (LDA) [11], have greatly advanced image recognition technology. In addition, nonlinear subspace methods, such as Kernel PCA [12], use kernel tricks to map original data into a high-dimensional space to make data separable. The second component is (2) classifier construction, for example, Nearest Neighbor (NN) [13], Nearest Feature Line (NFL) [14], and Nearest Feature Plane (NFP) [15]. These classifiers, known as representation-based models, concern how to identify the query image based on the linear combination of training samples [16].

According to the labels of the training samples used to represent test sample, representation-based approaches could be divided into within-class representation-based methods and across-class representation-based methods [16]. Within-class representation-based methods evaluate the relation between query sample and the training samples of each individual class separately by class. NN is the simplest non-parametric within-class representation-based classification method. It classified the test sample by searching for its NN in training dataset. As a simple extension of NN, NFL classifies the test sample according to the nearest feature line of every

two samples in each training class. NFP further uses three independent training samples to represent the test sample. Classifiers using more training samples to represent test image have also been proposed, such as the nearest subspace (NS) classifiers [17], which represent the test sample by all the training samples of each class.

Recent research has demonstrated that sparse coding (or sparse representation) is a powerful image representation model. The most typical across-class representation-based method is sparse representation-based classification (SRC) method [16], where test samples are first sparsely coded over all the training samples based on the l_1-norm minimization of representation coefficients, and then performs classification according to the representation results. SRC is robust to occlusion, illumination, and noise. Many related works have been developed to improve SRC, such as kernel based SRC [18], robust sparse coding [19], Gabor feature based SRC [20], and sparse dense hybrid representation [21]. SRC assumes that samples from a single subject lie on a subspace and restricts the representation of test sample to be sparse by the regularization of sparsity-inducing l_1-norm. This endows SRC with discriminative ability. Although SRC has shown excellent results, its working mechanism has been heat-debated. Zhang *et al.* [22] argued that all the samples should give a contribution to represent the test sample and proposed a collaborative representation (CR) based method for classification with l_2-norm constraint. Yang *et al.* [23] indicated that l_0-optimizer can achieve sparsity only, whereas l_1-optimizer in SRC can achieve closeness as well as sparsity. It is closeness that guarantees the effectiveness of the l_1-optimizer based SRC. Akhtar *et al.* [24] suggested that sparseness explicitly contributes to improved classification; hence it should not be completely ignored for computational gains. More recently, the discriminant nature of collaborative representation was analyzed in [25].

Ideally, the representation of a test example should focus on the training data from the identical class with the assumption of independence of sample subspaces, and such a representation is also discriminative for classification. Nevertheless, the assumption cannot always be guaranteed due to the similarity of patterns between different classes. That is, samples from different classes may have high correlation, and SRC tends to randomly select a single representative sample from the high-correlation training samples [26–28]. Meanwhile, SRC might select quite different training sample with test sample to favor sparsity. This trait is undesirable as it will produce different representations for similar images and thus destroy the classification performance.

One feasible way for the limitation is to go beyond sparsity and take into account additional information about the underlying structure of the solution. A desired representation for a test sample should have a group structure with the significant representation coefficients focusing on the correct subject [29]. For this purpose, Huang *et al.* [30] proposed a class specific sparse representation for classification (CSSRC) to seek certain group sparsity structure by harnessing the label information of training data. CSSRC could be solved as a classical group lasso (GL) problem attaining the purpose of sparsity at the group level by minimizing the l_1-norm across classes and l_2-norm within each class. The minimization of l_1-norm across classes can help find the correct subject. However, the dense representation within the selected class may prevent the attained result from the desired solution as the optimal representation of a test sample over training samples of the correct subject may not necessarily be dense [31]. The class-wise sparse representation (CSR) method is proposed in [31] to seek an optimum representation of the test image by minimizing the class-wise sparsity of the training data. However, the class-wise sparsity regularization treats different training classes equally and does not take the relation between test sample and each training class into consideration, which may restrict the discrimination of obtained representation.

Furthermore, Yu *et al.* [32] empirically observed that sparse coding results tend to be local, and the nonzero coefficients are often assigned to training samples nearby the represented sample. They theoretically pointed out that under certain assumption locality is more essential than sparsity, as locality must lead to sparsity but not necessarily vice versa. Based on the observation, Wang *et al.* [33] developed a locality-constrained linear coding (LLC) scheme to find a locality-constrained representation of test sample by its nearest neighbors. Lu *et al.* [34] proposed a weighted sparse representation-based classification method, where both data locality and linearity were considered. Several two-phase sparse representation-based methods have been proposed to conduct coarse-to-fine classification and achieved good performance [35–38]. These methods actually illustrate the effect of "locality" in representation-based classification methods and illustrate the principal that samples closer to the test sample should have more significance in representing it. By considering both data locality and label information of training data, Chao *et al.* [39] proposed a locality and group sensitive sparse representation (LGSR). Nevertheless, the group sparsity penalized term was directly added to the data locality constraint term, which may disarrange the group sparse structure of the solution. In [40], a weighted group sparse representation classification (WGSRC) is developed by minimizing the weighted mixed-norm ($l_{2,1}$-norm) regularized reconstruction error with respect to training samples. However, the l_2-norm reconstruction error measurement might not handle real world contamination well [31]. Table 1 tabulates a brief comparison of several related state-of-the-art methods. Besides, some dictionary learning methods have been developed to learn a dictionary from the training samples to replace the original training samples for representation learning [41–49]. The K-SVD algorithm is one of the typical dictionary learning algorithms [41]. K-SVD is a generalized k-means clustering algorithm. However, the K-SVD algorithm is not suitable for classification tasks, because it only requires that the learned dictionary should well reconstruct the training samples. To enhance the discriminative ability of learned dictionary, Zhang and Li studied a discriminative K-SVD (D-KSVD) algorithm [42]. Jiang *et al.* [43] proposed a label consistent K-SVD (LC-KSVD) dictionary learning algorithm, which associated the label information with the atoms to improve the classification performance. Li *et al.*

Table 1: A comparison between several representation-based learning methods.

Methods	Sparse			Sample local	Representation residual measure
	Sample sparse	Group sparse	Class-wise sparse		
SRC [16]	√	×	×	×	l_2-norm
LLC [32]	×	×	×	√	l_2-norm
LGSR [39]	×	√	×	√	l_2-norm
CSSRC [30]	×	√	×	×	l_2-norm
WGSRC [40]	×	√	×	√	l_2-norm
CSR [31]	×	×	√	×	l_1-norm
CoLR	×	×	√	√	l_1-norm

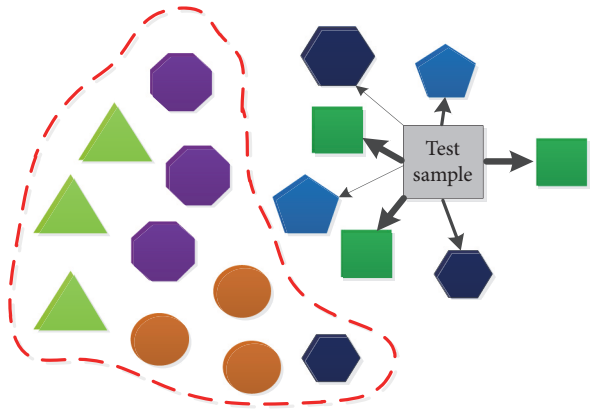


Figure 1: Overview of the proposed method. CoLR adopts a coarse-to-fine strategy to search for the most relevant training samples and classes with test sample. Nonneighbor training classes and samples are implicitly excluded by imposing bigger punishment (in red dotted line). This way, the test sample is guided to be represented by its nearby training classes and samples for a more discriminative classification-oriented local representation. This is the discriminability origin of CoLR.

[44] developed a discriminative dictionary learning algorithm termed as locality constrained and label embedding dictionary learning (LCLE-DL) algorithm. More recently, a mechanism-based structured analysis discriminative dictionary learning (ADDL) framework is developed to seamlessly integrate analysis discriminative dictionary learning, analysis representation, and analysis classifier training into a unified model [45]. A discriminative block-diagonal low-rank representation (BDLRR) method is proposed to learn discriminative data representation by imposing an effective structure in the low rank representation framework [50].

One of the key problems for representation-based classification method is to fully utilize the prior information of data distribution. This paper aims to learn discriminative classification-oriented local representation for image recognition with the guiding of prior information induced from the observed data. The overview of the proposed method is shown in Figure 1. The idea is that the training samples and classes near the test sample should make more contribution in representing it. In view of this, different training class is distinguished with different weights according to the distance between the test sample and each training class. Furthermore, weights are introduced into each class by utilizing a locality adaptor penalizing the training samples far away from the test sample. In sum, we consider three principals, i.e., class-wise sparseness, data locality, and class weighting. Class information is utilized to seek the minimum number of training classes in representing test sample. An l_1-norm based loss function is utilized, by which the obtained representation can less over-fit the outliers. With these principals, the developed model can not only encourage class-wise sparseness, but also boost locality sensibility within the selected training classes. Thus, the obtained representation is more discriminative with both the most relevant training classes and samples highlighted, which is desired for classification. Several contributions of the paper are listed as follows.

(1) By taking the advantages of data locality and label priors, the proposed CoLR model can learn discriminative classification-oriented local representation uncovering the underlying membership of test samples for classification.

(2) An efficient optimization algorithm is developed to solve CoLR model based on a variable splitting strategy and the Augmented Lagrangian Multiplier (ALM) method.

(3) The performance of CoLR is verified on various benchmark face databases and deep CNN features, and promising results have been achieved in comparison with some state-of-the-art models.

The remainder of this paper is organized as follows. Section 2 introduces some typical across-class representation-based classification algorithms. The CoLR model is proposed and described in Section 3. Experiments on benchmark databases are presented to evaluate the proposed method in comparison with other popular methods in Section 4. Finally, the conclusion is made in Section 5.

2. Related Works

In closed-universe image recognition scenario, the training samples and their class labels are usually provided. Assume there are n training face samples from C distinguished classes with n_i training samples in class i, the training sample matrix $\mathbf{A} = [\mathbf{A}_1, \mathbf{A}_2 \dots \mathbf{A}_C] \in \Re^{m \times n}$, $\mathbf{A}_i = [\boldsymbol{a}_{i,1}, \boldsymbol{a}_{i,2} \dots \boldsymbol{a}_{i,n_i}] \in$

$\mathfrak{R}^{m\times n_i}(i = 1, 2 \dots C)$. Each column in $\mathbf{A}$ is an m-dimensional vector through column concatenation of the training sample. $n = \sum_{i=1}^{C} n_i$ is the total number of training samples. The task is to assign the given test sample $\mathbf{y} \in \mathfrak{R}^m$ with the correct class label. In this section, we will review two typical works.

2.1. Sparse Representation for Classification. Sparse representation based classification (SRC) is based on the concept that patterns from the same class lie on a linear subspace [54]. The method parsimoniously searches for the most similar training sample in training set to represent the test sample. If a new coming test sample $\mathbf{y}$ comes from the class i, it will approximately lie on the linear span of the training samples of this class as

$$\mathbf{y} \approx \mathbf{A}_i \boldsymbol{x}_i \tag{1}$$

And the other samples in the training set cannot reconstruct the test sample as faithful as the ones from the identical class of the test sample. In practice, the test sample is represented using all the samples of training set $\mathbf{A}$.

$$\mathbf{y} \approx \mathbf{A}\boldsymbol{x} \tag{2}$$

where $\boldsymbol{x} = [0 \dots 0, x_{i,1}, x_{i,2} \dots x_{i,n_i}, 0 \dots 0]^{\mathrm{T}}$ is a coefficient vector whose entries are ideally zero except those associated with i-th class. For a sparse and discriminative solution, SRC aims to solve the l_0-minimization problem as

$$\begin{aligned} \boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} \quad & \|\boldsymbol{x}\|_0 \\ \text{s.t.} \quad & \left\|\mathbf{y} - \mathbf{A}\boldsymbol{x}\right\|_2 \le \varepsilon \end{aligned} \tag{3}$$

where ε is a given tolerance and $\| \bullet \|_0$ denotes the l_0-norm, which indicates the number of nonzero entries in the representation vector. However, the minimization of the l_0-norm is an NP-hard problem. Donoho proved that "for most large underdetermined systems of linear equations the minimal l_1-norm solution is also the sparsest solution" [55]. Thus, the solution of (3) is equivalent to the following l_1-norm minimization problem:

$$\begin{aligned} \boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} \quad & \|\boldsymbol{x}\|_1 \\ \text{s.t.} \quad & \left\|\mathbf{y} - \mathbf{A}\boldsymbol{x}\right\|_2 \le \varepsilon \end{aligned} \tag{4}$$

Once the coding vector is obtained, the identification of the test sample is performed by checking which training class leads to the minimal reconstruction residual as follows:

$$\text{identity}\left(\mathbf{y}\right) = \arg\min_i \left\{\left\|\mathbf{y} - \mathbf{A}_i \boldsymbol{x}_i\right\|_2\right\} \tag{5}$$

where $\boldsymbol{x}_i$ is the coding vector corresponding to training class $\mathbf{A}_i$. A sparse representation $\mathbf{x}$ of $\mathbf{y}$ over $\mathbf{A}$ is naturally discriminative to indicate the identity of $\mathbf{y}$.

2.2. Group LASSO for Classification. Using the l_1-norm to regularize the representation coefficient as in (4) may lead to an unstable solution because the training samples exhibit strong correlations. In the circumstance, SRC is known to have stability problem. It might represent a test sample by training data from distinct subjects, which is undesirable for classification. As a remedy for this problem, structured representation methods are proposed to take advantage of label priors. The training samples with the same label are defined as a group. An $l_{2,1}$-norm regularization term is used as the group sparsity constraint, which is also known as group lasso [56]. The formulation of group lasso is defined as

$$\begin{aligned} \boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} \quad & \sum_{i=1}^{C} \left\|\boldsymbol{x}_i\right\|_2 \\ \text{s.t.} \quad & \left\|\mathbf{y} - \mathbf{A}\boldsymbol{x}\right\|_2 \le \varepsilon \end{aligned} \tag{6}$$

where $\boldsymbol{x}_i$ is the coding vector corresponding to training class $\mathbf{A}_i$. This constraint enforces nonzeros coefficients to occur at a few specific groups, while those within the same group can be dense once that group is selected. However, the dense representation within the class may adversely affect the correct class selection during the minimization process of group lasso [30].

3. The Proposed Classification-Oriented Local Representation Model

3.1. Model Formulation. Searching classes instead of individual samples can alleviate the problem of random selection of highly correlated data in SRC. By taking both weighted class-wise sparseness across classes and sample locality within each class into consideration, the following optimization problem is proposed.

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \frac{1}{2}\left\|\mathbf{y} - \mathbf{A}\boldsymbol{x}\right\|_1 + \lambda \sum_{i=1}^{C} \mathrm{w}_i \mathrm{z}_i \\ \text{s.t.} \quad & \mathrm{z}_i = \left\|\boldsymbol{d}_i \bigodot \boldsymbol{x}_i\right\|_2 \end{aligned} \tag{7}$$

where $\boldsymbol{x} \in \mathfrak{R}^n$ is representation coefficient of test sample $\mathbf{y}$ over training dataset $\mathbf{A}$. The first term $\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_1$ is the representation residual of the test sample $\mathbf{y}$ measured by l_1-norm, and the second term indicates the weighted class-wise sparseness regularization term. Specifically, the l_2-norm of locality sensitive coefficients within each class is weighted to search for the minimal number of classes out of C training classes by introducing a new class indicator variable $\mathbf{z} = [\mathrm{z}_1, \mathrm{z}_2, \dots, \mathrm{z}_C]^{\mathrm{T}}$. w_i $(i = 1, 2 \dots C)$ is the weight of samples of class i in representing test sample $\mathbf{y}$. $\mathbf{d}_i$ is a vector that gives different freedom for each training sample associated with class i. $\lambda > 0$ is a tradeoff parameter between the representation residual and representation coefficient.

We aim to represent test sample $\mathbf{y}$ not only by its neighbors but also by highly relevant training classes. As illustrated in (7), unlike pursuing a group sparse representation in $\mathfrak{R}^n$ like group lasso, we directly search for a weighted class-wise sparse representation in $\mathfrak{R}^{\mathrm{C}}$ by taking the relation between test sample and each training class into consideration. Afterwards, the dissimilar training samples

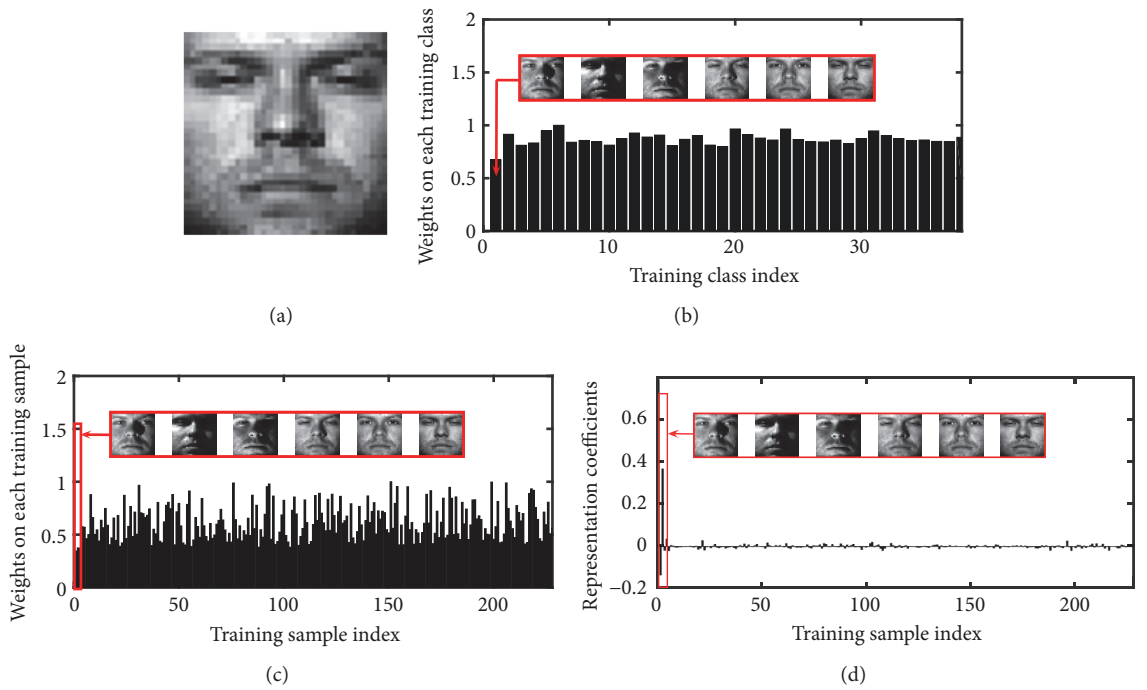


FIGURE 2: An example for proposed method on 228 training samples of 38 subjects from Extended Yale B database [51]. (a) A test sample from Class 1; (b) the weight on each training class; (c) the weight on each training sample; (d) the representation coefficients obtained by our method.

are penalized by incorporating a locality adaptor. w_is vary from class to class to boost or inhibit the corresponding training class in representing $\mathbf{y}$, and a large w_i would make the corresponding z_i shrink to be small. l_1-norm is used to measure the representation residual, which is proved to be robust to handle outliers, and the obtained representation of test sample would less over-fit the outliers. In model (7), $\odot$ denotes the element-wise multiplication, and $\mathbf{d}_i = [d_{i,1}, d_{i,2} \dots d_{i,n_i}]$ $(i = 1, 2 \dots C)$ performs as a locality adaptor that gives different freedom for each training sample $\boldsymbol{a}_{ij}$ (the j-th training sample of class i) proportional to its similarity with test sample $\mathbf{y}$.

$$d_{ij} = \exp\left(\frac{\text{dist}\left(\mathbf{y}, \boldsymbol{a}_{ij}\right)}{\sigma_1}\right) \tag{8}$$

where $\text{dist}(\mathbf{y}, \boldsymbol{a}_{ij})$ is the Euclidean distance between test sample $\mathbf{y}$ and training sample $\boldsymbol{a}_{i,j}$. σ_1 is used to adjust the weight decay speed for the locality adaptor. A larger d_{ij} indicates a farther distance between $\mathbf{y}$ and $\boldsymbol{a}_{ij}$. In (7), w_i is used to evaluate the distance between the test sample with training class i. The linear regression model is utilized to measure the distance between test sample and each training class. The test sample is firstly represented as a linear combination of the training samples in each class.

$$\mathbf{y} \approx \mathbf{A}_i \boldsymbol{x}_i \tag{9}$$

The reconstructed test sample $\tilde{\boldsymbol{y}}_i$ over class $\mathbf{A}_i$ is

$$\tilde{\mathbf{y}}_i = \mathbf{A}_i \left(\mathbf{A}_i^{\mathrm{T}} \mathbf{A}_i\right)^{-1} \mathbf{A}_i^{\mathrm{T}} \mathbf{y} \tag{10}$$

The following metric is utilized to measure the distance between test sample $\mathbf{y}$ and training class i:

$$w_i = \exp\left(\frac{\left\|\mathbf{y} - \tilde{\mathbf{y}}_i\right\|^2}{\sigma_2}\right) \tag{11}$$

In (11), w_i indicates the distance from $\mathbf{y}$ to the subspace generated by $\mathbf{A}_i$. A larger w_i means class i is further from $\mathbf{y}$ and should make less contribution to represent it. σ_2 is a bandwidth parameter and used for adjusting the weight decay. Figure 2 intuitively illustrates the motivation. Figure 2(a) is a test sample from Class 1. Figure 2(b) shows the normalized distance, i.e., w_i $(i = 1, 2 \dots C)$ in (7), between test sample and each training class. Figure 2(c) shows the normalized distance, i.e., $\mathbf{d}_i$ $(i = 1, 2 \dots n)$ in (7), between test sample and each training sample. Figure 2(d) shows the final representation of test sample (a) over the training set. As the test sample is from Class 1, so the punishment on the representation coefficient of training samples in Class 1 is relatively small. For other training samples and classes, the punishment is bigger. With these constrains, the test sample is guided be represented by the nearby training samples from Class 1 as shown in Figure 2(d), which is quite discriminative for classification.

3.2. Model Optimization. As presented above, the optimization problem of developed model is expressed as follows:

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \frac{1}{2}\left\|\mathbf{y}-\mathbf{A}\boldsymbol{x}\right\|_1+\lambda\sum_{i=1}^{C}\mathrm{w}_i\mathrm{z}_i \\ \text{s.t} \quad & \mathrm{z}_i=\left\|\mathbf{d}_i\bigodot\boldsymbol{x}_i\right\|_2 \end{aligned} \tag{12}$$

The second term in (12) is the representation coefficient regularization. $\sum_{i=1}^{C}\mathrm{w}_i\mathrm{z}_i = \mathrm{w}_1\mathrm{z}_1+\ldots+\mathrm{w}_C\mathrm{z}_C$. The term can be written in the form of l_1-norm with a new variable $\boldsymbol{v}$. $\|\boldsymbol{v}\|_1=\mathrm{w}_1\|\mathbf{d}_1\bigodot\boldsymbol{x}_1\|_2+\ldots+\mathrm{w}_C\|\mathbf{d}_C\bigodot\boldsymbol{x}_C\|_2$ with $v_i=\mathrm{w}_i\|\mathbf{d}_i\bigodot\boldsymbol{x}_i\|_2$ $(i=1,2\ldots\mathrm{C})$. Model (12) can be rewritten as

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \frac{1}{2}\left\|\mathbf{y}-\mathbf{A}\boldsymbol{x}\right\|_1+\lambda\left\|\boldsymbol{v}\right\|_1 \\ \text{s.t.} \quad & v_i=\mathrm{w}_i\left\|\mathbf{d}_i\bigodot\boldsymbol{x}_i\right\|_2 \end{aligned} \tag{13}$$

where $\boldsymbol{v}=[v_1,v_2,\ldots,v_C]^{\mathrm{T}}$. We introduce a vector $\mathbf{r}_i=[\mathrm{r}_{i1},\mathrm{r}_{i2},\ldots,\mathrm{r}_{in_i}]^{\mathrm{T}}$ $(i=1,2\ldots C)$, $\mathrm{r}_{ij}=\mathrm{w}_i\mathrm{d}_{ij}$, $j=1,2\ldots,n_i$, and then we have

$$v_i=\left\|\mathbf{r}_i\bigodot\boldsymbol{x}_i\right\|_2 \tag{14}$$

Furthermore,

$$v_i=\left\|\mathbf{R}_i\boldsymbol{x}_i\right\|_2 \tag{15}$$

where $\mathbf{R}_i\in\Re^{n_i\times n_i}$ is a diagonal matrix for the i-th class with $\mathbf{r}_i$ as its diagonal elements, and $\mathbf{R}$ is given by

$$\mathbf{R}\coloneqq\begin{bmatrix}\mathbf{R}_1 & \cdots & \\ \vdots & \ddots & \vdots \\ & \cdots & \mathbf{R}_C\end{bmatrix} \tag{16}$$

Let $\boldsymbol{e}=\mathbf{y}-\mathbf{A}\boldsymbol{x}$. We obtain the following equivalent optimization problem with the diagonal weight matrix $\mathbf{R}$ by introducing auxiliary variable $\boldsymbol{u}\in\Re^{C}$:

$$\begin{aligned} \min_{\boldsymbol{e},\boldsymbol{x},\boldsymbol{z},\boldsymbol{u}} \quad & \frac{1}{2}\left\|\boldsymbol{e}\right\|_1+\lambda\left\|\boldsymbol{v}\right\|_1 \\ \text{s.t.} \quad & \boldsymbol{e}=\mathbf{y}-\mathbf{A}\boldsymbol{x}, \\ & \boldsymbol{u}=\mathbf{R}\boldsymbol{x}, \\ & v_i=\left\|\boldsymbol{u}_i\right\|_2 \end{aligned} \tag{17}$$

where $\boldsymbol{v}=[v_1,v_2,\ldots,v_C]^{\mathrm{T}}$. Denote $\widetilde{\boldsymbol{u}}=[\|\boldsymbol{u}_1\|_2,\|\boldsymbol{u}_2\|_2,\ldots,\|\boldsymbol{u}_C\|_2]^{\mathrm{T}}\in\Re^{C}$, and $\boldsymbol{u}_i=\mathbf{R}_i\boldsymbol{x}_i\in\Re^{n_i}$ is a subvector of $\boldsymbol{u}$ with elements associated with class i. (17) is a constrained optimization problem which could be solved by the Augmented Lagrangian Multiplier (ALM) method [57, 58]. Its corresponding ALM function is given as

$$\begin{aligned} & L_{\mu}(\boldsymbol{e},\boldsymbol{x},\boldsymbol{v},\boldsymbol{u}) \\ &=\frac{1}{2}\left\|\boldsymbol{e}\right\|_1+\lambda\left\|\boldsymbol{v}\right\|_1+\boldsymbol{\alpha}^{\mathrm{T}}(\mathbf{y}-\mathbf{A}\boldsymbol{x}-\boldsymbol{e}) \\ &\quad+\boldsymbol{\beta}^{\mathrm{T}}(\mathbf{R}\boldsymbol{x}-\boldsymbol{u})+\boldsymbol{\gamma}^{\mathrm{T}}(\widetilde{\boldsymbol{u}}-\boldsymbol{v}) \\ &\quad+\frac{\mu}{2}\left(\left\|\mathbf{y}-\mathbf{A}\boldsymbol{x}-\boldsymbol{e}\right\|_2^2+\left\|\mathbf{R}\boldsymbol{x}-\boldsymbol{u}\right\|_2^2+\left\|\widetilde{\boldsymbol{u}}-\boldsymbol{v}\right\|_2^2\right) \end{aligned} \tag{18}$$

where $\boldsymbol{\alpha}\in\Re^{m}$, $\boldsymbol{\beta}\in\Re^{n}$, $\boldsymbol{\gamma}\in\Re^{C}$ are vectors of Lagrange multipliers and $\mu>0$ is the penalty parameter. Instead of optimizing all arguments simultaneously, as $\boldsymbol{e},\boldsymbol{x},\boldsymbol{v},\boldsymbol{u}$ are separable, we solve them individually and iteratively. In the $(k+1)$-th iteration, the updating schemes are as follows.

Step 1 (update $\boldsymbol{e}$). We update $\boldsymbol{e}$ by solving the following subproblem with $\boldsymbol{x}$, $\boldsymbol{z}$, and $\boldsymbol{u}$ fixed:

$$\begin{aligned} \boldsymbol{e}_{k+1} &= \arg\min_{\boldsymbol{e}}\frac{1}{2}\left\|\boldsymbol{e}\right\|_1+\boldsymbol{\alpha}_k^{\mathrm{T}}(\mathbf{y}-\mathbf{A}\boldsymbol{x}_k-\boldsymbol{e}) \\ &\quad+\frac{\mu_k}{2}\left\|\mathbf{y}-\mathbf{A}\boldsymbol{x}_k-\boldsymbol{e}\right\|_2^2 \\ &= S_{1/(2\mu_k)}\left[\mathbf{y}-\mathbf{A}\boldsymbol{x}_k+\frac{\boldsymbol{\alpha}_k}{\mu_k}\right] \end{aligned} \tag{19}$$

where $S_{\epsilon}(\bullet)$, $\epsilon>0$, is the soft-thresholding (shrinkage) operator defined component-wise as

$$\left[S_{\epsilon}(\boldsymbol{\theta})\right]_i=\mathrm{sign}(\boldsymbol{\theta}_i)\bullet\max\left\{\left|\boldsymbol{\theta}_i\right|-\epsilon,0\right\} \tag{20}$$

Step 2 (update $\boldsymbol{x}$). We update $\boldsymbol{x}$ by solving the following subproblem with $\boldsymbol{e}$, $\boldsymbol{v}$, and $\boldsymbol{u}$ fixed:

$$\begin{aligned} & \boldsymbol{x}_{k+1} \\ &=\arg\min_{\boldsymbol{x}}\boldsymbol{\alpha}_k^{\mathrm{T}}(\mathbf{y}-\mathbf{A}\boldsymbol{x}-\boldsymbol{e}_{k+1})+\boldsymbol{\beta}_k^{\mathrm{T}}(\mathbf{R}\boldsymbol{x}-\boldsymbol{u}_k) \\ &\quad+\frac{\mu_k}{2}\left(\left\|\mathbf{y}-\mathbf{A}\boldsymbol{x}-\boldsymbol{e}_{k+1}\right\|_2^2+\left\|\mathbf{R}\boldsymbol{x}-\boldsymbol{u}_k\right\|_2^2\right) \\ &=\arg\min_{\boldsymbol{x}}\frac{\mu_k}{2}\boldsymbol{x}^{\mathrm{T}}\left(\mathbf{A}^{\mathrm{T}}\mathbf{A}+\mathbf{R}^{\mathrm{T}}\mathbf{R}\right)\boldsymbol{x} \\ &\quad-\mu_k\left(\left(\mathbf{y}-\boldsymbol{e}_{k+1}+\frac{\boldsymbol{\alpha}_k}{\mu_k}\right)^{\mathrm{T}}\mathbf{A}+\left(\boldsymbol{u}_k-\frac{\boldsymbol{\beta}_k}{\mu_k}\right)^{\mathrm{T}}\mathbf{R}\right)\boldsymbol{x} \end{aligned} \tag{21}$$

This is a convex quadratic problem. Hence the problem reduces to solving the following linear system:

$$\begin{aligned} & \mu_k\left(\mathbf{A}^{\mathrm{T}}\mathbf{A}+\mathbf{R}^{\mathrm{T}}\mathbf{R}\right)\boldsymbol{x} \\ &\quad=\mu_k\left(\mathbf{A}^{\mathbf{T}}\left(\mathbf{y}-\boldsymbol{e}_{k+1}+\frac{\boldsymbol{\alpha}_k}{\mu_k}\right)+\mathbf{R}^{\mathrm{T}}\left(\boldsymbol{u}_k-\frac{\boldsymbol{\beta}_k}{\mu_k}\right)\right) \end{aligned} \tag{22}$$

The close-form solution of $\boldsymbol{x}$ is obtained as

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \left(\mathbf{A}^{\mathrm{T}}\mathbf{A}+\mathbf{R}^{\mathrm{T}}\mathbf{R}\right)^{-1} \\ &\quad\cdot\left(\mathbf{A}^{\mathrm{T}}\left(\mathbf{y}-\boldsymbol{e}_{k+1}+\frac{\boldsymbol{\alpha}_k}{\mu_k}\right)+\mathbf{R}^{\mathrm{T}}\left(\boldsymbol{u}_k-\frac{\boldsymbol{\beta}_k}{\mu_k}\right)\right) \end{aligned} \tag{23}$$

Step 3 (update $\boldsymbol{v}$). We update $\boldsymbol{v}$ by solving the following subproblem with $\boldsymbol{x}$, $\boldsymbol{e}$, and $\boldsymbol{u}$ fixed:

$$\begin{aligned} \boldsymbol{v}_{k+1} &= \arg\min_{\boldsymbol{v}}\lambda\left\|\boldsymbol{v}\right\|_1+\boldsymbol{\gamma}_{\boldsymbol{k}}^{\boldsymbol{T}}(\widetilde{\boldsymbol{u}}_k-\boldsymbol{v})+\frac{\mu_k}{2}\left\|\widetilde{\boldsymbol{u}}_k-\boldsymbol{v}\right\|_2^2 \\ &=\arg\min_{\boldsymbol{v}}\frac{\lambda}{\mu_k}\left\|\boldsymbol{v}\right\|_1+\frac{1}{2}\left\|\boldsymbol{v}-\left(\widetilde{\boldsymbol{u}}_k+\frac{\boldsymbol{\gamma}_k}{\mu_k}\right)\right\|_2^2 \end{aligned} \tag{24}$$

which can also be solved via the soft-thresholding operator as in (20).

Input: training set $\mathbf{A}$, test sample $\mathbf{y}$, diagonal matrix $\mathbf{R}$, parameter λ.
1: **Initialize:** $\boldsymbol{x}_0 = \mathbf{0}$, $\boldsymbol{z}_0 = \mathbf{0}$, $\boldsymbol{u}_0 = \mathbf{0}$, $\boldsymbol{\alpha}_0 = \mathbf{0}$, $\boldsymbol{\beta}_0 = \mathbf{0}$, $\boldsymbol{\gamma}_0 = \mathbf{0}$, $\mu_0 = 10^{-3}$, $\mu_{max} = 10^8$, $\rho = 1.1$, $\varepsilon = 10^{-6}$.
2: **While** *not converged* **do**
3: Fix other variable and update $\mathbf{e}$ by solving problem (19).
4: Fix other variable and update $\mathbf{x}$ by solving problem (21).
5: Fix other variable and update $\mathbf{v}$ by solving problem (24).
6: Fix other variable and update $\mathbf{u}$ by solving problem (25).
7: Update the multipliers and parameters by (28)
8: **Check the convergence conditions**:
$\|\boldsymbol{y} - \mathbf{A}\boldsymbol{x} - \boldsymbol{e}\|_\infty < \varepsilon$, $\|\mathbf{R}\boldsymbol{x} - \boldsymbol{u}\|_\infty < \varepsilon$, and $\|\boldsymbol{v} - \widetilde{\boldsymbol{u}}\|_\infty < \varepsilon$
9: **End While**
Output: $\boldsymbol{x}$, $\boldsymbol{e}$

Algorithm 1: SolvingCoLR model based on ALM.

Step 4 (update $\boldsymbol{u}$). We update $\boldsymbol{u}$ by solving the following subproblem with $\boldsymbol{e}$, $\boldsymbol{v}$, and $\boldsymbol{x}$ fixed:

$$\begin{aligned}\boldsymbol{u}_{k+1} = \arg\min_{\boldsymbol{u}} \boldsymbol{\beta}_k^T \left(\mathbf{R}\boldsymbol{x}_{k+1} - \boldsymbol{u}\right) + \boldsymbol{\gamma}_k^T \left(\widetilde{\boldsymbol{u}} - \boldsymbol{v}_{k+1}\right) \\ + \frac{\mu_k}{2}\left(\left\|\mathbf{R}\boldsymbol{x}_{k+1} - \boldsymbol{u}\right\|_2^2 + \left\|\widetilde{\boldsymbol{u}} - \boldsymbol{v}_{k+1}\right\|_2^2\right)\end{aligned} \tag{25}$$

With some manipulation, we have

$$\begin{aligned}\boldsymbol{u}_{k+1} = \arg\min_{\boldsymbol{u}} \sum_{i=1}^{C} \Bigg[\left(\boldsymbol{\gamma}_{ki} - \mu_k \boldsymbol{v}_{(k+1)i}\right) \left\|\boldsymbol{u}_i\right\|_2 \\ + \mu_k \left\| \boldsymbol{u}_i - \frac{\left(\mathbf{R}_i \boldsymbol{x}_{(k+1)i} + \boldsymbol{\beta}_{ki}/\mu_k\right)}{2} \right\|_2^2 \Bigg]\end{aligned} \tag{26}$$

which has a closed form solution by one-dimensional shrinkage (or soft-thresholding) formula [59]

$$\begin{aligned}\boldsymbol{u}_{(k+1)i} = \max\left(\left\|\boldsymbol{q}_{ki}\right\|_2 - \frac{\left(\boldsymbol{\gamma}_{ki} - \mu_k \boldsymbol{v}_{(k+1)i}\right)}{2\mu_k}, 0 \right) \frac{\boldsymbol{q}_{ki}}{\left\|\boldsymbol{q}_{ki}\right\|_2}. \\ i = 1, 2 \ldots C.\end{aligned} \tag{27}$$

where $\boldsymbol{q}_{ki} \coloneqq (\mathbf{R}_i \boldsymbol{x}_{(k+1)i} + \boldsymbol{\beta}_{ki}/\mu_k)/2$

Step 5. Update the Lagrangian multipliers $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$ and the parameter μ as follows:

$$\begin{aligned}\boldsymbol{\alpha}_{k+1} &= \boldsymbol{\alpha}_k + \mu_k \left(\mathbf{y} - \mathbf{A}\boldsymbol{x}_{k+1} - \boldsymbol{e}_{k+1}\right) \\ \boldsymbol{\beta}_{k+1} &= \boldsymbol{\beta}_k + \mu_k \left(\mathbf{R}\boldsymbol{x}_{k+1} - \boldsymbol{u}_{k+1}\right) \\ \boldsymbol{\gamma}_{k+1} &= \boldsymbol{\gamma}_k + \mu_k \left(\widetilde{\boldsymbol{u}}_{k+1} - \boldsymbol{v}_{k+1}\right) \\ \mu_{k+1} &= \min\left(\rho\mu_k, \mu_{max}\right)\end{aligned} \tag{28}$$

Note that the subproblems for $\boldsymbol{e}$, $\boldsymbol{x}$, $\boldsymbol{u}$, $\boldsymbol{v}$ are all convex problems. They both have closed-form solutions. For the competence of presentation, the detailed optimization procedure is outlined in Algorithm 1. The convergence property of Algorithm 1 can be guaranteed by the existing ADM theory [56]. From the experimental perspective, the developed optimization algorithm exhibits good convergence property as shown in Figure 3. In our experiments, the iteration number is empirically less than 100 under the given settings. Once the optimal solution $\boldsymbol{x}$ is achieved, the label of test sample $\mathbf{y}$ can be obtained as

$$i^* = \arg\min_i \left\{ \frac{\left\|\mathbf{y} - \mathbf{A}_i \mathbf{x}_i\right\|_1}{\left\|\mathbf{x}_i\right\|_1} \right\}, \quad i = 1, 2 \ldots C \tag{29}$$

where $\boldsymbol{x}_i$ is the coding vector associated with training class $\mathbf{A}_i$.

4. Experiments

4.1. Parameter Setting. In this section, we will study the key parameters in proposed CoLR model and give some parameter setting suggestions to use it for classification tasks. There are 3 parameters, i.e., λ, σ_1, and σ_2 in proposed CoLR model. Among them, λ is used to keep balance between the reconstruction error and the level of weighted class-wise sparseness of the representation coefficient. σ_1 and σ_2 are the bandwidth parameters in sample and class distance metrics. When λ is larger, fewer training samples would be selected. Empirically speaking, a relatively small λ is preferred to keep the balance between representation residual and the representation coefficient. For CoLR, we search for the optimal λ in the range of $[10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10]$ to obtain the best recognition accuracy. The optimal σ_1 and σ_2 are searched in the set of $[1/32, 1/16, 1/8, 1/4, 1/2, 1, 10]$. Figure 4 shows recognition accuracy versus different parameters on CMU PIE database [52]. The parameters influence each other. When the value of λ is assigned to be a small positive value, our method could achieve promising performance. Figure 4(a) shows that when the value of parameter λ varies between 10^{-5} and 10^{-3}, CoLR performs relatively well and the curve of recognition rate tends to be smooth for different values of σ_1 and σ_2. It can be seen that our method is robust to the values of parameters. When λ is set as 10^{-4}, Figure 4(b) shows that the recognition rates are generally high when the value of σ_2 is in the range of $[1/4, 10]$. When the value of σ_1 is constant, the recognition rate tends to

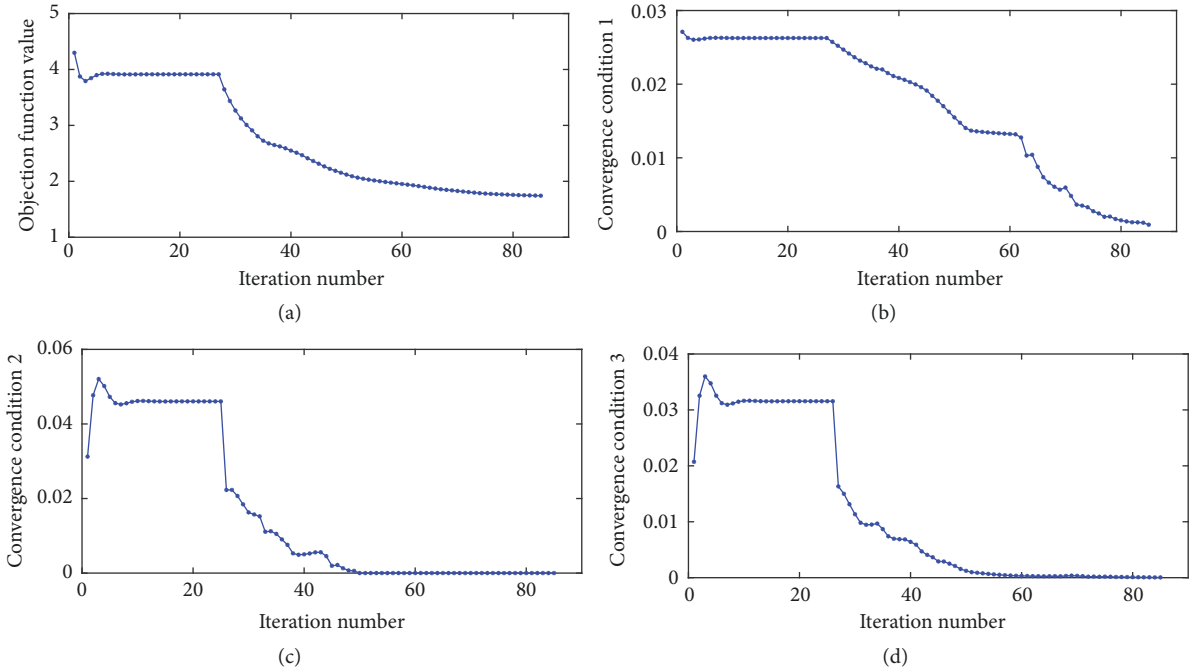


FIGURE 3: An illustration for the convergence property of Algorithm 1. (a) is the curve of objective function value versus iteration number. (b)-(d) are the curves of 3 convergence conditions in optimization versus iteration number: (b) $\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x} - \boldsymbol{e}\|_\infty$; (c) $\|\boldsymbol{v} - \tilde{\boldsymbol{u}}\|_\infty$; (d) $\|\boldsymbol{R}\boldsymbol{x} - \boldsymbol{u}\|_\infty$.

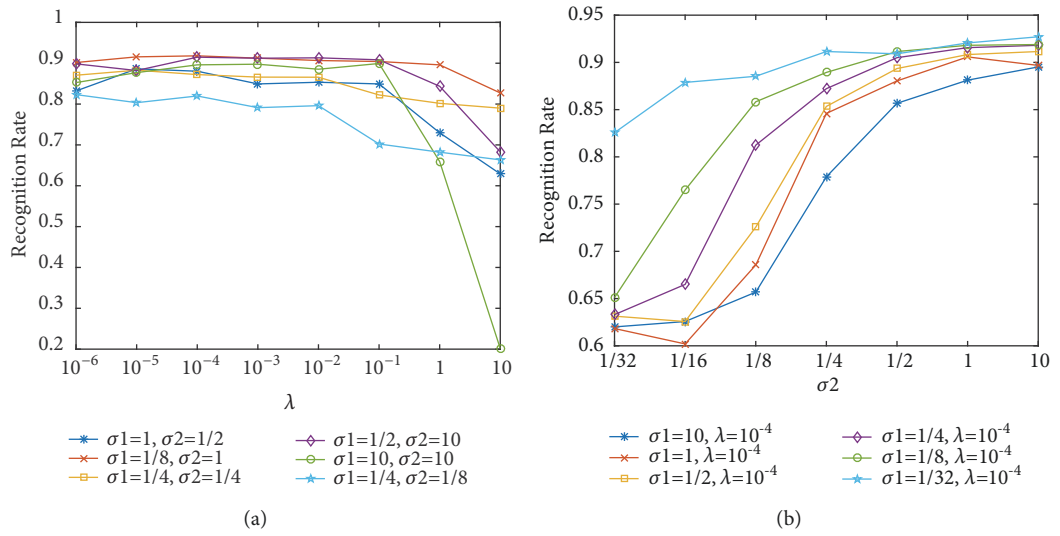
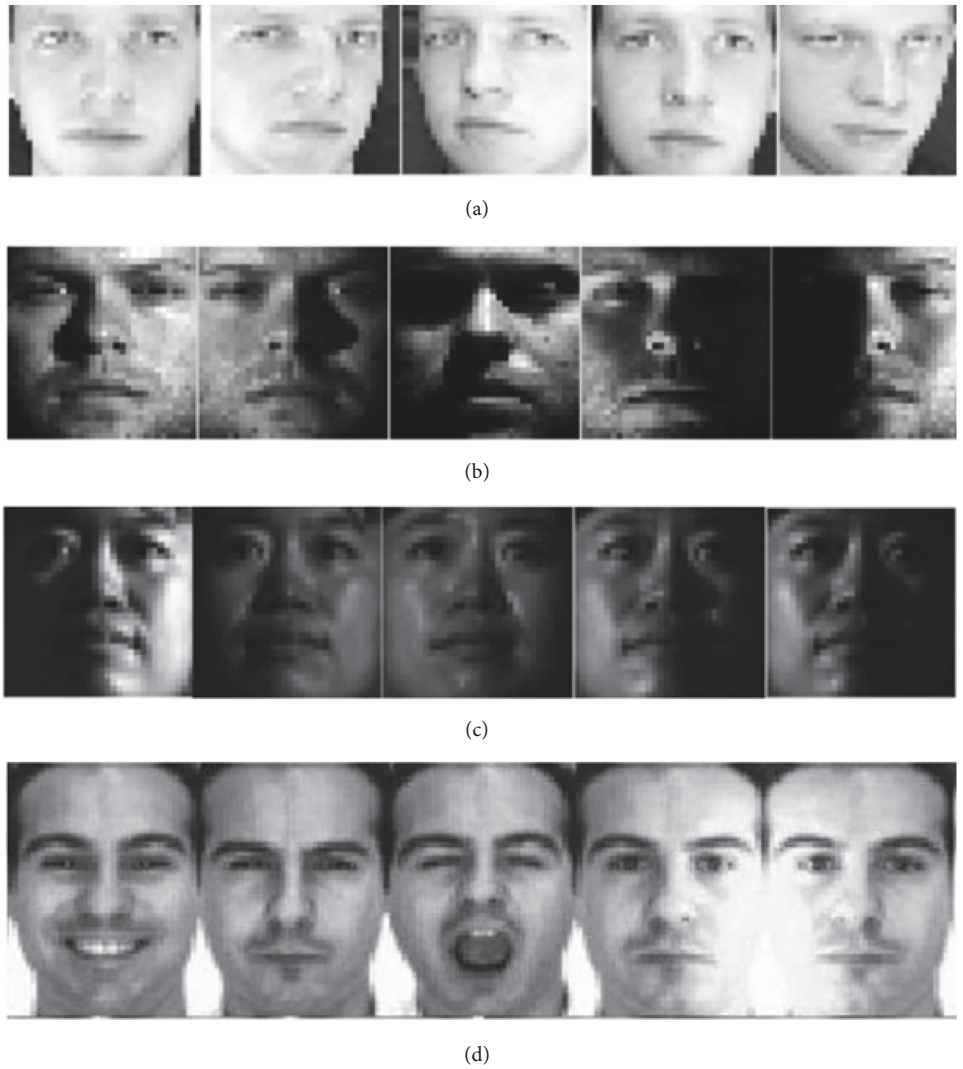


FIGURE 4: Face recognition rates of CoLR under different parameter settings on CMU PIE dataset [52]. (a) λ versus σ_1 and σ_2; (b) σ_1 versus σ_2 with $\lambda = 0.0001$.

increase as σ_2 increases. And when the value of σ_2 is constant, the recognition rate tends to increase as σ_1 decreases. On the basis of the analysis above, the optimal values of λ, σ_1, and σ_2 are suggested to be assigned from the range of $[10^{-5}, 10^{-3}]$, $[1/32, 1/8]$, and $[1/4, 10]$, respectively. With these parameters, stable and satisfactory performance of CoLR model can be expected.

4.2. Experimental Results on Face Recognition. In this section, our approach is compared with several related state-of-the-art approaches, including nearest neighbor (NN), linear regression classification (LRC) [17], collaborative representation based classification (CRC) [22], sparse representation for classification (SRC) [16], locality constrained linear coding for classification (LLC) [32], weighted group sparse



(a)

(b)

(c)

(d)

Figure 5: Some facial images used in our experiments: (a) the ORL dataset; (b) the Extended Yale B dataset; (c) the CMU PIE dataset; (d) the AR dataset.

representation classification (WGSRC) [40], class-wise sparse representation (CSR) [31], and discriminative block-diagonal low-rank representation (BDLRR) [50]. Some dictionary learning-based classification methods are also compared, including LC-KSVD [43], LCLE-DL [44], and dictionary pair learning (DPL) [48]. LLC can be seen as an extended version of SRC exploiting data locality instead of sparsity constraint for improved representation coding and adopts a reconstruction-based classification rule. WGSRC classify a test sample by minimizing the weighted $l_{2,1}$-norm regularized reconstruction error with respect to training images. CSR seeks an optimum representation of the query image by minimizing the class-wise sparsity of the training data. The parameter settings of other methods follow the references. Experiments were conducted on 5 face datasets, including the ORL [3], Extended Yale B [51], CMU PIE [52], and AR [53]. A description of the 4 datasets is shown in Table 2.

Table 2: Description of the datasets used in the experiments.

Database	# Samples	# Dimension	# Classes
ORL [3]	400	1,024	40
Extended Yale B [51]	2,414	1,024	38
CMU PIE [52]	1,680	1,024	68
AR [53]	1,400	1,260	100

4.2.1. Experiments on the ORL Database. The ORL face database consists of 400 face images from 40 individuals with 10 images per person. The images were taken at different times, lighting variation, facial expressions (open/closed eyes, smiling/not smiling), and facial details (glassed/no glassed) against a dark homogeneous background. In the experiments, each image in ORL database is manually cropped and resized to 32×32. Figure 5(a) shows some sample images from one

Table 3: Recognition rates on the ORL dataset with different number of training samples.

Methods	# training samples per subject on ORL dataset			
	3	4	5	6
NN	0.758±0.031	0.819±0.024	0.863±0.014	0.875±0.020
LRC [17]	0.810±0.034	0.885±0.027	0.916±0.020	0.946±0.018
CRC [22]	0.851±0.025	0.916±0.015	0.936±0.011	0.942±0.025
SRC [16]	0.882±0.030	0.933±0.016	0.943±0.011	0.954±0.016
LLC [32]	0.847±0.027	0.893±0.024	0.899±0.020	0.909±0.021
WGSRC [40]	0.881±0.025	0.932±0.020	0.946±0.012	0.958±0.016
LC-KSVD2 [43]	0.857±0.026	0.902±0.015	0.913±0.015	0.932±0.010
LCLE-DL [44]	0.870±0.017	0.913±0.015	0.927±0.020	0.939±0.011
CSR [31]	0.859±0.024	0.924±0.020	0.942±0.011	0.955±0.019
DPL [48]	0.870±0.025	0.912±0.024	0.926±0.022	0.939±0.023
BDLRR [50]	0.879±0.017	0.921±0.021	0.932±0.021	0.949±0.016
CoLR	*0.882±0.034*	*0.940±0.018*	*0.952±0.011*	*0.965±0.015*

Table 4: Recognition rates on the ORL dataset under different feature dimensions.

Methods	Feature dimensions (5 samples per subject)		
	30	60	90
NN	0.845±0.020	0.853±0.040	0.859±0.009
LRC [17]	0.901±0.018	0.911±0.027	0.927±0.018
CRC [22]	0.857±0.021	0.914±0.025	0.939±0.012
SRC [16]	0.898±0.019	0.932±0.018	0.949±0.015
LLC [32]	0.844±0.020	0.885±0.026	0.913±0.020
WGSRC [40]	0.894±0.016	0.932±0.029	0.943±0.015
LC-KSVD2 [43]	0.851±0.026	0.876±0.025	0.924±0.020
LCLE-DL [44]	0.858±0.029	0.876±0.032	0.926±0.020
CSR [31]	0.894±0.011	0.929±0.022	0.943±0.015
DPL [48]	0.896±0.033	0.908±0.041	0.926±0.029
BDLRR [50]	0.917±0.019	0.926±0.026	0.934±0.023
CoLR	*0.923±0.015*	*0.937±0.019*	*0.956±0.016*

subject. Two experimental settings are considered. Firstly, a random subset with p (= 3, 4, 5, 6) images of each individual is selected for training and the rest for testing. For each experimental scenario, we first apply PCA as preprocessing step to reduce the dimension of original data to 100. We run the programs 10 times and calculate the recognition rates as well as the standard deviations, which are reported in Table 3. Besides, we evaluate the performance of different methods with different feature dimensions, i.e., 30, 60, 90, and 120. The experimental results are listed in Table 4. The best results are highlighted in the italic face font.

4.2.2. Experiments on the Extended Yale B Database. Extended Yale B face database contains about 2414 frontal face images of 38 persons and around 64 near frontal images under different illuminations per person. In this experiment, we simply use the cropped images and resize them to 32×32 pixels. Figure 5(b) shows some example images of one subject. A random subset with p (=6, 8, 12, 16) images per individual is taken with labels to form the training set, and the remaining samples are used for testing. The experiment is repeated 10 times, and the comparison results are shown in Table 5. We then evaluate the performance of different algorithms under different feature dimensions with 16 samples per subject as the training set. The FR recognition rates under different dimensions are shown in Table 6, where the best results are highlighted in the italic face font.

4.2.3. Experiments on the CMU PIE Database. The CMU PIE database contains over 40, 000 face images of 68 individuals. Images of each individual were acquired across 13 different poses, under 43 different illumination conditions, and with 4 different expressions. Here we use a near frontal pose subset, namely, C07, for experiments, which contains 1629 images of 68 individuals. Each individual has about 24 images. All images are manually cropped and resized to 32×32 pixel in our experiment. Several sample images from the dataset are shown in Figure 5(c). A random subset with p (= 2, 4, 6, 8) images of each individual is selected as training dataset, and the rest is used for testing. For each given p, we

Table 5: Recognition rates on the Extended Yale B dataset with different number of training samples.

Methods	# training samples per subject			
	6	8	12	16
NN	0.417±0.012	0.482±0.011	0.582±0.014	0.655±0.008
LRC [17]	0.686±0.015	0.774±0.011	0.858±0.006	0.901±0.007
CRC [22]	0.847±0.012	0.892±0.009	0.930±0.005	0.949±0.004
SRC [16]	0.811±0.011	0.863±0.010	0.915±0.009	0.943±0.006
LLC [32]	0.822±0.009	0.871±0.008	0.918±0.006	0.943±0.007
WGSRC [40]	0.820±0.009	0.871±0.010	0.921±0.005	0.946±0.007
LC-KSVD2 [43]	0.796±0.014	0.857±0.009	0.897±0.008	0.922±0.008
LCLE-DL [44]	0.802±0.015	0.855±0.009	0.897±0.010	0.921±0.007
CSR [31]	0.828±0.009	0.877±0.007	0.917±0.008	0.937±0.007
DPL [48]	0.810±0.012	0.857±0.006	0.909±0.008	0.930±0.004
BDLRR [50]	0.792±0.014	0.843±0.007	0.909±0.009	0.935±0.007
CoLR	*0.860±0.008*	*0.906±0.009*	*0.942±0.005*	*0.957±0.005*

Table 6: Recognition rates on the Extended Yale Bdataset under different feature dimensions.

Methods	Feature dimensions (16 samples per subject)		
	50	100	150
NN	0.497±0.009	0.593±0.009	0.624±0.013
LRC [17]	0.866±0.007	0.885±0.009	0.898±0.007
CRC [22]	0.794±0.015	0.893±0.013	0.923±0.007
SRC [16]	0.860±0.008	0.907±0.011	0.923±0.007
LLC [32]	0.741±0.018	0.864±0.013	0.902±0.007
WGSRC [40]	0.822±0.009	0.896±0.013	0.920±0.006
LC-KSVD2 [43]	0.656±0.010	0.865±0.006	0.907±0.007
LCLE-DL [44]	0.671±0.010	0.873±0.006	0.909±0.008
CSR [31]	0.839±0.009	0.900±0.013	0.922±0.008
DPL [48]	0.832±0.015	0.901±0.009	0.921±0.005
BDLRR [50]	*0.871±0.011*	0.904±0.009	*0.930±0.007*
CoLR	0.847±0.007	*0.908±0.012*	0.929±0.006

independently run all the methods 10 times and report the recognition rates as well as the standard deviations in Table 7. Table 8 lists the recognition rates and corresponding standard deviations of different comparing methods with different dimensions of features. In the experiment scenarios, 6 images were randomly chosen from each subject for training set, and the remaining samples were used for testing. Similarly, PCA is utilized to calculate a low dimensional subspace to reduce the dimensionality of original face data, and the reduced dimensionalities are set as 50, 100, 150, and 200.

4.2.4. Experiments on the AR Database. In this experiment, a subset of AR database that contains 50 males and 50 females with 6 illumination and 8 expression variations in two sessions is used. 7 images with only illumination and expression changes from session 1 are used as training set, and 7 images from session 2 are used as testing set. Figure 5(d) shows some sample images from one person. The recognition rates of different methods on this recognition task are shown in Table 9.

4.2.5. Experiments on the Deep CNN Features. Deep learning with a convolutional neural network has been proved to be very effective in feature extraction and representation of images. In this experiment, we would like to test the performance of different representation-based classifiers on deep CNN features. The structures of CNN for training on the ImageNet with 1000 categories are the same as the proposed CNN in [60]. The basic structure of the adopted is illustrated in Figure 6, which includes 5 convolutional layers and 3 fully connected layers. For further details of the CNN training architecture and features one can refer to [60, 61]. The CNN outputs of the 6-th (f_6) and 7-th (f_7) fully connected layers are used as inputs of different representation-based classifiers for classification, respectively. We adopt the deep convolutional activated features (DeCAF) from [61] for experiments.

Table 7: Recognition rates on the CMU PIE dataset with different number of training samples.

Methods	# training samples per subject			
	2	4	6	8
NN	0.363±0.015	0.550±0.013	0.678±0.013	0.759±0.014
LRC [17]	0.513±0.025	0.815±0.013	0.900±0.014	0.925±0.010
CRC [22]	0.810±0.018	0.916±0.007	0.933±0.005	0.939±0.005
SRC [16]	0.782±0.019	0.906±0.009	0.938±0.005	0.948±0.008
LLC [32]	0.789±0.021	0.912±0.008	0.935±0.006	0.945±0.006
WGSRC [40]	0.791±0.019	0.913±0.008	0.939±0.006	*0.951±0.007*
LC-KSVD2 [43]	0.764±0.022	0.892±0.009	0.921±0.007	0.936±0.007
LCLE-DL [44]	0.773±0.023	0.894±0.008	0.926±0.007	0.937±0.007
CSR [31]	0.806±0.019	0.915±0.008	0.938±0.006	0.946±0.007
DPL [48]	0.760±0.016	0.893±0.009	0.930±0.008	0.941±0.006
BDLRR [50]	0.737±0.015	0.885±0.010	0.932±0.005	0.947±0.008
CoLR	*0.820±0.020*	*0.920±0.007*	*0.942±0.006*	0.947±0.006

Table 8: Recognition rates on the CMU PIE dataset under different feature dimensions.

Methods	Feature dimensions (6 samples per subject)			
	50	100	150	200
NN	0.604±0.019	0.649±0.018	0.669±0.016	0.681±0.018
LRC [17]	0.893±0.014	0.899±0.009	0.898±0.009	0.903±0.008
CRC [22]	0.868±0.014	0.925±0.006	0.930±0.007	0.937±0.004
SRC [16]	0.903±0.013	0.930±0.005	0.931±0.006	0.939±0.005
LLC [32]	0.838±0.015	0.917±0.009	0.929±0.006	0.938±0.005
WGSRC [40]	0.892±0.015	0.930±0.006	0.934±0.007	*0.942±0.004*
LC-KSVD2 [43]	0.828±0.011	0.918±0.010	0.930±0.005	0.934±0.007
LCLE-DL [44]	0.834±0.012	0.922±0.006	0.930±0.008	0.937±0.005
CSR [31]	0.891±0.014	0.916±0.008	0.920±0.008	0.929±0.005
DPL [48]	0.886±0.007	0.922±0.006	0.934±0.009	0.937±0.006
BDLRR [50]	*0.907±0.007*	0.928±0.005	0.933±0.011	0.941±0.009
CoLR	*0.907±0.013*	*0.931±0.004*	*0.935±0.007*	0.935±0.004

Table 9: Recognition rates on the AR dataset under different feature dimensions.

Methods	Feature dimensions (7 samples per subject)			
	50	100	150	200
NN	0.651	0.689	0.694	0.694
LRC [17]	0.683	0.720	0.734	0.736
CRC [22]	0.789	0.876	0.904	*0.930*
SRC [16]	0.816	0.887	0.899	0.919
LLC [32]	0.784	0.869	0.889	0.906
WGSRC [40]	0.814	0.883	0.903	0.916
LC-KSVD2 [43]	0.669	0.793	0.829	0.846
LCLE-DL [44]	0.694	0.817	0.863	0.883
CSR [31]	0.824	0.884	0.897	0.917
DPL [48]	0.786	0.854	0.877	0.890
BDLRR [50]	*0.831*	0.881	0.886	0.915
CoLR	0.824	*0.897*	*0.906*	0.923

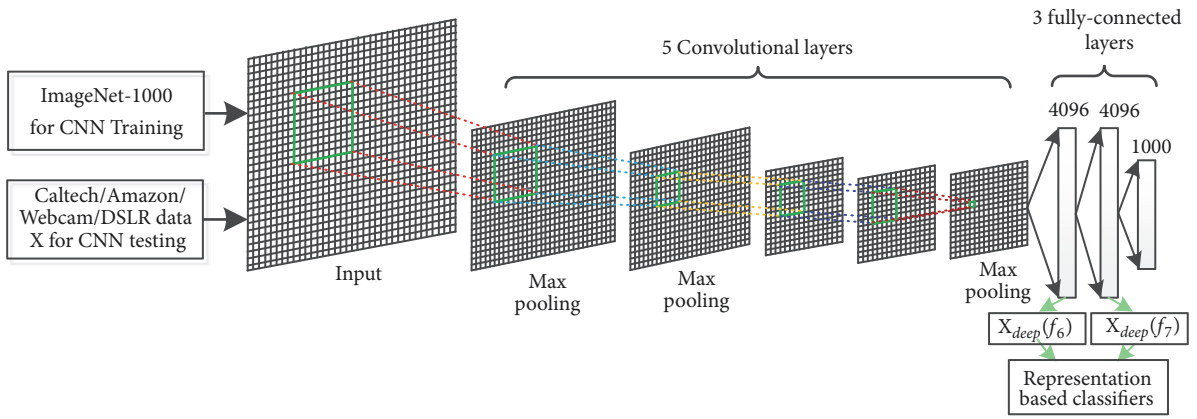


Figure 6: Diagram of the training and testing protocol in the experiment.

Figure 7: Examples of object images from 4 sources: Amazon (1st row), DSLR (2nd row), Webcam (3rd row), and Caltech (4th row).

The well-trained network parameters shown in Figure 6 are used for deep representation of the 4DA (domain adaptation) dataset. The 4DA dataset includes four domains such as Caltech 256 (C), Amazon (A), Webcam (W), and DSLR (D) sampled from different sources, in which 10 object classes are selected. Example images are shown in Figure 7.

Single-domain recognition task is considered in this experiment. For each dataset with different CNN-layer features, 20, 8, 8, and 8 samples per class are randomly selected for training from Amazon, DSLR, Webcam, and Caltech domains, respectively, and the remaining are used as test samples for each domain. PCA is further applied to both training and test sets to preserve 99% principle components of data for computational efficiency. The average recognition accuracy for each method is reported. Recognition rates of each method on 4DA datasets using f_6 and f_7 CNN-layer features are shown in Tables 10 and 11, respectively.

Based on the experimental results above, some conclusions can be reached as follows.

(1) In general, across-class representation-based methods such as SRC and CRC perform better than within-class representation-based methods, e.g., NN and LRC. The reason

Table 10: Recognition rates of each method on 4DA datasets using f_6 CNN-layer features.

Methods	Datasets			
	Amazon	DLSR	Webcam	Caltech
NN	0.923±0.012	0.979±0.021	0.972±0.020	0.817±0.014
LRC [17]	*0.936±0.009*	0.977±0.019	0.977±0.018	0.856±0.010
CRC [22]	0.896±0.021	0.978±0.018	0.976±0.016	0.810±0.038
SRC [16]	0.935±0.006	0.975±0.022	0.977±0.018	0.852±0.012
LLC [32]	0.936±0.007	0.975±0.022	0.978±0.018	0.853±0.013
WGSRC [40]	0.936±0.007	0.975±0.022	0.977±0.018	0.852±0.012
LC-KSVD2 [43]	0.933±0.007	0.975±0.022	0.974±0.013	0.848±0.012
LCLE-DL [44]	0.933±0.006	0.977±0.020	0.975±0.012	0.847±0.012
CSR [31]	0.924±0.006	0.975±0.021	0.972±0.013	0.659±0.012
DPL [48]	*0.936±0.007*	0.981±0.009	0.978±0.020	0.851±0.020
BDLRR [50]	*0.936±0.006*	0.973±0.011	0.972±0.023	*0.859±0.017*
CoLR	0.933±0.009	*0.982±0.020*	*0.979±0.014*	0.831±0.014

Table 11: Recognition rates of each method on 4DA datasets using f_7 CNN-layer features.

Methods	Datasets			
	Amazon	DLSR	Webcam	Caltech
NN	0.928±0.011	0.971±0.017	0.964±0.010	0.828±0.012
LRC [17]	*0.937±0.011*	0.973±0.014	*0.976±0.014*	*0.863±0.008*
CRC [22]	0.914±0.011	0.958±0.018	0.972±0.016	0.825±0.011
SRC [16]	*0.937±0.010*	0.971±0.015	0.975±0.013	0.860±0.008
LLC [32]	0.936±0.010	0.971±0.015	0.974±0.013	0.861±0.009
WGSRC [40]	0.936±0.010	0.971±0.015	0.975±0.013	0.861±0.009
LC-KSVD2 [43]	0.935±0.009	0.978±0.013	0.975±0.014	0.854±0.008
LCLE-DL [44]	0.936±0.009	0.973±0.013	0.973±0.015	0.856±0.006
CSR [31]	0.933±0.009	0.971±0.016	0.974±0.011	0.817±0.013
DPL [48]	*0.937±0.008*	0.971±0.021	0.964±0.018	0.850±0.019
BDLRR [50]	0.933±0.005	0.969±0.020	0.963±0.017	0.858±0.014
CoLR	0.933±0.011	*0.975±0.011*	*0.976±0.014*	0.848±0.006

can be attributed to the collaborative representation characteristic of the across-class representation-based methods. This is due to the fact that face recognition is often an undersampled classification problem. As a result, within-class representation-based methods might be incapable of providing enough representation ability for query sample because of limited within-class samples, which will restrict their performance.

(2) As for across-class representation-based methods, training samples compete with each other to win their share in representation learning with sparseness constraint, which will make the learned representation more discriminative for classification. As a result, sparseness-based methods tend to outperform nonsparse methods, such as CRC. CRC has closed-form solution with lower computational cost. One possible future research direction is to further improve its discriminant ability for classification with computational advantage inherited.

(3) Supervised representation learning methods, such as WGSRC, GSC, and CSR, utilize label information for representation learning, which will make the obtained representation more suitable and discriminative for classification tasks. Comparatively speaking, the proposed CoLR model outperforms comparing methods in most of the cases under considered experimental scenarios. This is partly because CoLR imposes bigger punishment on these training classes and samples that are likely to be far away from the query sample. The method actually adopts a coarse-to-fine strategy to search for the most relevant training samples and classes with query sample and implicitly excludes the nonneighbor training classes and samples. With the strategy, more discriminative classification-oriented and locality adaptive representation can be learnt, which tends to be more efficient and adaptive for classification tasks as the experimental results show. As a classification method, CoLR can be combined with different kinds of feature for classification. The performance of CoLR can be further enhanced using discriminative deep CNN features.

(4) The emphasis of this paper is on learning a discriminative representation for classification on given dictionary.

Thus, CoLR model directly adopts the original training samples as the dictionary to learn classification-oriented representation. Experimental results show that CoLR could yield excellent performance in comparison with dictionary learning methods, which show that the representation learning strategy of CoLR is efficient in exploring the inherent discrimination information of training samples for classification. Many references have suggested that a dictionary learnt from the original training data seems to be more discriminative and compact than original training data. The experimental results have shown the effectiveness and efficiency of CoLR for image recognition. Future work will consider the possible applications of CoLR by unifying dictionary learning methods.

5. Conclusions

In this paper, we have proposed a novel CoLR model for image recognition. Specifically, an informative and discriminative classification-oriented local representation could be learned in terms of l_1-norm loss function by taking both the weighted class-wise sparseness and data locality within each class into consideration. The developed representation strategy can encourage classification steered representation and boost locality sensitivity within the selected training classes highlighting the test sample's most relevant training classes and samples. Also, an efficient optimization algorithm is devised to solve CoLR model based on a variable splitting strategy and the ALM scheme. Experimental results on several face databases and the deep CNN features show that CoLR can yield promising performance by comparing with many representative models.

Data Availability

The source codes and datasets used to support the findings of this study are available from the submitting author upon request via email: guot@cqupt.edu.cn.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by National Natural Science Foundation of China (No. 61771079, 61571069, 61801072), the Science and Technology Research Program of Chongqing Municipal Education Commission (No. KJQN201800632, KJQN201800617), and Foundation and Frontier Research Project of Chongqing Municipal Science and Technology Commission (No. cstc2018jcyjAX0344, cstc2018jcyjAX0549, cstc2017zdcy-zdzxX0002).

References

[1] Z. Zhang, L. Shao, Y. Xu, L. Liu, and J. Yang, "Marginal representation learning with graph structure self-adaptation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 10, pp. 4645–4659, 2018.

[2] X. Fang, N. Han, J. Wu et al., "Approximate low-rank projection learning for feature extraction," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 11, pp. 5228–5241, 2018.

[3] D. Cai, X. He, J. Han, and H.-J. Zhang, "Orthogonal laplacianfaces for face recognition," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3608–3614, 2006.

[4] Z. Zhang, M. Zhao, and T. W. S. Chow, "Binary-and multi-class group sparse canonical correlation analysis for feature extraction and classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 10, pp. 2192–2205, 2013.

[5] Z. Zhang, S. Yan, and M. Zhao, "Pairwise sparsity preserving embedding for unsupervised subspace learning and classification," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 4640–4651, 2013.

[6] F. Luo, B. Du, L. Zhang, L. Zhang, and D. Tao, "Feature learning using spatial-spectral hypergraph discriminant analysis for hyperspectral image," *IEEE Transactions on Cybernetics*, vol. 49, no. 7, pp. 2406–2419, 2019.

[7] F. Luo, H. Huang, Y. Duan, J. Liu, and Y. Liao, "Local geometric structure feature for dimensionality reduction of hyperspectral imagery," *Remote Sensing*, vol. 9, no. 8, p. 790, 2017.

[8] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition," *IEEE Transactions on Image Processing*, vol. 11, no. 4, pp. 467–476, 2002.

[9] T. Ahonen, A. Hadid, and M. Pietikainen, "Face recognition with local binary patterns," in *Proceedings of the 8th European Conference on Computer Vision (ECCV '04)*, vol. 3021 of *Lecture Notes in Computer Science*, pp. 469–481, Springer, Prague, Czech Republic, May 2004.

[10] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[11] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.

[12] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, no. 5, pp. 1299–1319, 1998.

[13] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[14] S. Z. Li and J. Lu, "Face recognition using the nearest feature line method," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 10, no. 2, pp. 439–443, 1999.

[15] J. Chien and C. Wu, "Discriminant waveletfaces and nearest feature classifiers for face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 12, pp. 1644–1649, 2002.

[16] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[17] I. Naseem, R. Togneri, and M. Bennamoun, "Linear regression for face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 11, pp. 2106–2112, 2010.

[18] S. Gao, I. Tsang, and L.-T. Chia, "Kernel sparse representation for image classification and face recognition," in *Computer Vision ECCV 2010*, K. Daniilidis, P. Maragos, and N. Paragios, Eds., pp. 1–14, Springer, Berlin, Germany, 2010.

[19] M. Yang, L. Zhang, J. Yang, and D. Zhang, "Robust sparse coding for face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 625–632, IEEE, Colorado Springs, Colo, USA, June 2011.

[20] M. Yang, L. Zhang, S. C. K. Shiu, and D. Zhang, "Gabor feature based robust representation and classification for face recognition with Gabor occlusion dictionary," *Pattern Recognition*, vol. 46, no. 7, pp. 1865–1878, 2013.

[21] X. Jiang and J. Lai, "Sparse and dense hybrid representation via dictionary decomposition for face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 5, pp. 1067–1079, 2015.

[22] L. Zhang, M. Yang, and X. Feng, "Sparse representation or collaborative representation: Which helps face recognition?" in *Proceedings of the IEEE International Conference on Computer Vision (ICCV '11)*, pp. 471–478, IEEE, Barcelona, Spain, November 2011.

[23] J. Yang, L. Zhang, Y. Xu, and J.-Y. Yang, "Beyond sparsity: the role of L1-optimizer in pattern classification," *Pattern Recognition*, vol. 45, no. 3, pp. 1104–1118, 2012.

[24] N. Akhtar, F. Shafait, and A. Mian, "Efficient classification with sparsity augmented collaborative representation," *Pattern Recognition*, vol. 65, pp. 136–145, 2017.

[25] W. Deng, J. Hu, and J. Guo, "Face recognition via collaborative representation: its discriminant nature and superposed representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 10, pp. 2513–2521, 2018.

[26] T. Guo, L. Zhang, X. Tan, L. Yang, and Z. Liang, "Data induced masking representation learning for face data analysis," *Knowledge-Based Systems*, vol. 177, pp. 82–93, 2019.

[27] T. Guo, L. Zhang, and X. Tan, "Neuron pruning-based discriminative extreme learning machine for pattern classification," *Cognitive Computation*, vol. 9, no. 4, pp. 581–595, 2017.

[28] T. Guo, X. Tan, L. Zhang, Q. Liu, L. Deng, and C. Xie, "Learning robust weighted group sparse graph for discriminant visual analysis," *Neural Processing Letters*, vol. 49, no. 1, pp. 203–226, 2019.

[29] A. Majumdar and R. K. Ward, "Classification via group sparsity promoting regularization," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '09)*, pp. 861–864, IEEE, Taiwan, April 2009.

[30] S. Huang, Y. Yang, D. Yang, L. Huangfu, and X. Zhang, "Class specific sparse representation for classification," *Signal Processing*, vol. 116, pp. 38–42, 2015.

[31] J. Lai and X. Jiang, "Classwise sparse and collaborative patch representation for face recognition," *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 3261–3272, 2016.

[32] K. Yu, T. Zhang, and Y. Gong, "Nonlinear learning using local coordinate coding," in *Proceedings of the 23rd Annual Conference on Neural Information Processing Systems*, pp. 2223–2231, British Columbia, Canada, December 2009.

[33] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 3360–3367, IEEE, San Francisco, Calif, USA, June 2010.

[34] C.-Y. Lu, H. Min, J. Gui, L. Zhu, and Y.-K. Lei, "Face recognition via weighted sparse representation," *Journal of Visual Communication and Image Representation*, vol. 24, no. 2, pp. 111–116, 2013.

[35] Y. Xu, D. Zhang, J. Yang, and J. Y. Yang, "A two-phase test sample sparse representation method for use with face recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 9, pp. 1255–1262, 2011.

[36] Z. Liu, J. Pu, M. Xu, and Y. Qiu, "Face recognition via weighted two phase test sample sparse representation," *Neural Processing Letters*, vol. 41, no. 1, pp. 43–53, 2015.

[37] E. G. Ortiz and B. C. Becker, "Face recognition for web-scale datasets," *Computer Vision and Image Understanding*, vol. 118, pp. 153–170, 2014.

[38] N. Zhang and J. Yang, "K nearest neighbor based local sparse representation classifier," in *Proceedings of the Chinese Conference on Pattern Recognition (CCPR '10)*, pp. 1–5, IEEE, Chongqing, China, October 2010.

[39] Y. Chao, Y. Yeh, Y. Chen, Y. Lee, and Y. F. Wang, "Locality-constrained group sparse representation for robust face recognition," in *Proceedings of the 18th IEEE International Conference on Image Processing (ICIP '11)*, pp. 11–14, IEEE, Brussels, Belgium, September 2011.

[40] X. Tang, G. Feng, and J. Cai, "Weighted group sparse representation for undersampled face recognition," *Neurocomputing*, vol. 145, pp. 402–415, 2014.

[41] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[42] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 2691–2698, IEEE, San Francisco, Claif, USA, June 2010.

[43] Z. L. Jiang, Z. Lin, and L. S. Davis, "Learning a discriminative dictionary for sparse coding via label consistent K-SVD," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 1697–1704, IEEE, Colorado Springs, Colo, USA, June 2011.

[44] Z. Li, Z. Lai, Y. Xu, J. Yang, and D. Zhang, "A locality-constrained and label embedding dictionary learning algorithm for image classification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 2, pp. 278–293, 2017.

[45] Z. Zhang, W. Jiang, J. Qin et al., "Jointly learning structured analysis discriminative dictionary and analysis multiclass classifier," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3798–3814, 2018.

[46] Z. Li, Z. Zhang, Z. Fan, and J. Wen, "An interactively constrained discriminative dictionary learning algorithm for image classification," *Engineering Applications of Artificial Intelligence*, vol. 72, no. 8, pp. 241–252, 2018.

[47] Z. Zhang, W. Jiang, F. Li, M. Zhao, B. Li, and L. Zhang, "Structured latent label consistent dictionary learning for salient machine faults representation-based robust classification," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 644–656, 2017.

[48] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Projective dictionary pair learning for pattern classification," in *Proceedings of the 28th Annual Conference on Neural Information Processing Systems (NIPS '14)*, pp. 793–801, December 2014.

[49] Z. Zhang, F. Li, T. W. Chow, L. Zhang, and S. Yan, "Sparse codes auto-extractor for classification: a joint embedding and dictionary learning framework for representation," *IEEE Transactions on Signal Processing*, vol. 64, no. 14, pp. 3790–3805, 2016.

[50] Z. Zhang, Y. Xu, L. Shao, and J. Yang, "Discriminative block-diagonal representation learning for image recognition," *IEEE*

Transactions on Neural Networks and Learning Systems, vol. 29, no. 7, pp. 3111–3125, 2018.

[51] 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 Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 643–660, 2001.

[52] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression database," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1615–1618, 2003.

[53] A. Martinez and R. Benavente, "The AR face database," CVC Technical Report 24, 1998.

[54] R. Barsi and D. Jacobs, "Lambertian reflection and linear subspaces," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 25, no. 2, pp. 218–233, 2003.

[55] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal l_1-norm solution is also the sparsest solution," *Communications on Pure and Applied Mathematics*, vol. 59, no. 6, pp. 797–829, 2006.

[56] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2006.

[57] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, NY, USA, 1996.

[58] Z. Lin, M. Chen, L. Wu, and Y. Ma, "The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices," UIUC UILU-ENG-09-2215, 2009.

[59] W. Deng, W. Yin, and Y. Zhang, "Group sparse optimization by alternating direction method," Defense Technical Information Center TR11-06, Department of Computational and Applied Mathematics Rice University, Houston, Tex, USA, 2011.

[60] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS '12)*, pp. 1097–1105, Lake Tahoe, Nev, USA, December 2012.

[61] J. Donahue, Y. Jia, O. Vinyals et al., "DeCAF: a deep convolutional activation feature for generic visual recognition," in *Proceedings of the 31st International Conference on Machine Learning (ICML '14)*, pp. 988–996, Beijing, China, June 2014.