Low-Rank Double Dictionary Learning from Corrupted Data for Robust Image Classification

Yi Rong¹,², Shengwu Xiong¹,† and Yongsheng Gao²,*

¹School of Computer Science and Technology, Wuhan University of Technology, China
²School of Engineering, Griffith University, Australia

Abstract

In this paper, we propose a novel low-rank double dictionary learning (LRD²L) method for robust image classification tasks, in which the training and testing samples are both corrupted. Unlike traditional dictionary learning methods, LRD²L simultaneously learns three components from corrupted training data: 1) a low-rank class-specific sub-dictionary for each class to capture the most discriminative class-specific features of each class, 2) a low-rank class-shared dictionary which models the common patterns shared in the data of different classes, and 3) a sparse error term to model the noise in data. Through low-rank class-shared dictionary and noise term, the proposed method can effectively separate the corruptions and noise in training samples from creating low-rank class-specific sub-dictionaries, which are employed for correctly reconstructing and classifying testing images. Comparative experiments are conducted on three public available databases. Experimental results are encouraging, demonstrating the effectiveness of the proposed method and its superiority in performance over the state-of-the-art dictionary learning methods.

Keywords: Low-rank dictionary learning; class-specific dictionary; class-shared dictionary; image classification; corrupted training samples; robustness.

† Joint Corresponding author. Tel: +86 027 87216780, E-mail: xiongsw@whut.edu.cn
* Corresponding author. Tel: +61 7 37353652, E-mail: yongsheng.gao@griffith.edu.au
1. Introduction

Image classification has been attracting much attention from researchers in the areas of pattern recognition and machine learning due to their importance in many real-world applications, such as computational forensics, face recognition and medical diagnosis [1],[2],[3]. By exploiting the label information of training images, supervised dictionary learning approaches [4],[5],[6],[7],[8],[9],[10],[54] have achieved impressive performances in learning the discriminative information from training samples for accurately classifying testing images, and are widely used in various applications [11],[12],[13],[14],[15]. However, in real-world applications, it is inevitable that the training and/or testing samples are corrupted (i.e., containing interference and noise components not relevant to the classes). These corruptions can appear in various forms (such as occlusions, facial expressions, lighting variations, pose variations, and noise in face recognition tasks) and make a dictionary learning algorithm fail to function due to significant performance drop.

Recent efforts have been made towards developing robust image classification algorithms to handle the corruptions in image samples [16],[17],[18],[19]. They developed new methods built on the strategy of simultaneously learning the reconstruction coefficients and weight matrix for each testing sample, which is calculated based on the reconstruction residual of an image sample. Such weight matrix can be employed to detect the error pixels in each testing sample to handle the corruptions in testing samples. Highly competitive performances are achieved when testing samples are corrupted. However, they cannot perform well when the training samples are corrupted as the corruptions in the training data will be carried into dictionaries, resulting in a poor reconstruction of testing samples, and thus lead to a failure of classification.
The robust image classification when both training and testing samples are corrupted remains an unsolved practical problem, because the training data obtained in real-world applications are often not clean. The challenge of such a problem lies in the fact that the intra-class variations in the corrupted image samples are larger than their inter-class differences. Inspired by [20], we find that an image in robust image classification tasks often contains three types of information: (1) Class-specific information, which are the discriminative features owned by one class. (2) Class-shared information, which are the common patterns shared by different classes. For example, in a face recognition task, face images from different classes often share common illumination, pose, and disguise (such as wearing glasses) variants. These variants are not helpful to distinguish different classes, but without them, images with similar variants cannot be well represented. (3) Sparse noise, which is useless for image classification and often enlarges the intra-class differences. If we can separate the class-specific information from the others and solely construct a dictionary to capture such information, this dictionary will be immune to the corruptions and noise in the training samples for correctly reconstructing and classifying testing samples.

Based on above observation, we propose a novel low-rank double dictionary learning (LRD2L) approach to separate these three types of information. More specifically, given a training sample set with labels, we propose to learn a low-rank class-specific sub-dictionary for each class, which captures the most discriminative features of the class. Simultaneously, a low-rank class-shared dictionary is constructed for representing the common patterns shared in images of different classes. An error term is introduced to approximate the sparse noise contained in image samples. Through the class-shared dictionary and noise term, the proposed method can effectively separate the corruptions and noise in training samples from affecting the construction of class-specific sub-
dictionaries, which will be utilised to correctly reconstruct and classify new testing samples. To the best of our knowledge, the proposed low-rank double dictionary (LRD$^2$L) method is the first attempt to integrate the low-rank matrix recovery technique with the class-specific and class-shared dictionary learning. By separating the corruptions and sparse noise in the training samples through the class-shared dictionary and sparse error term, the proposed method is invariant, in theory, to corruptions and noise in both training and testing data. A new alternative optimization algorithm is designed to effectively solve the optimization problem for the proposed method.

The rest of the paper is organized as follows: In Section 2, we briefly review the background and related work. The proposed low-rank double dictionary learning method and the design of its objective function are discussed in detail in Section 3. Section 4 presents our algorithms to practically solve the optimization problem for the proposed method. Experimental results on three public databases are reported in Section 5. Finally, the paper concludes in Section 6.

2. Background

Correctly learning a dictionary from training samples is of vital importance to effectively reconstructing and classifying testing samples for image classification tasks. Zhang and Li [15] proposed a discriminative K-SVD (DK-SVD) method for face recognition by integrating a classification error term into objective function. The classification accuracy of a linear classifier and the reconstructive power of dictionary are simultaneously considered, such that the learned dictionary is both reconstructive and discriminative. Jiang et al. [21] proposed a label consistent K-SVD (LC-KSVD) algorithm that associates the label information with the dictionary atoms. They introduced a binary code matrix to force the samples from the same classes to have similar
representations, which enhances the discriminability of their coding coefficients. Ramirez et al. [8] proposed a novel classification and clustering framework built on dictionary learning by introducing a structured incoherence regularization term to make sub-dictionaries as independent as possible. By imposing the Fisher criterion on the sparse coding coefficients, Yang et al. [22], [23] proposed a Fisher discrimination dictionary learning (FDDL) method to make the coding coefficients have small within-class scatters and large between-class scatters. Both the reconstruction residual and the distances between the representation coefficients of a testing sample and the mean representation coefficients of each class are taken into account to classify the testing samples, which achieved a higher level of accuracy. Gu et al. [9] proposed a projective dictionary pair learning (DPL) method for image classification, which learns an analysis dictionary and a synthesis dictionary for each class to make the training and testing processes more efficient.

To overcome the limitations of above algorithms that do not consider the common features shared by different classes, Zhou et al. [14] developed a novel joint dictionary learning (JDL) algorithm that learns one common dictionary and multiple category-specific dictionaries. By explicitly separating the common features from the category-specific patterns, the learned category-specific dictionaries can be more discriminative. Wang and Kong [20] proposed to learn a particular dictionary (called particularity) for each class, and a common pattern pool (called commonality) shared by all the classes. The commonality only complements the reconstruction of input images over the particular dictionary and does not contain any discriminative information. Therefore, the particularity captures the most discriminative features in each class and thus can be more compact and discriminative. Sun et al. [24] learned the common and class-specific dictionaries based on a weighted group sparse coding framework, and achieved
impressive performance on several public datasets. The group sparsity helps to capture
the structural information of data samples, which improves the discriminability of the
learned dictionary. Gao et al. [25] and Zhang et al. [53] applied the common and class-
specific dictionary learning strategy to solve a new fine-grained image categorization
problem. By imposing both the self-incoherence and the cross-incoherence constrains on
each dictionary, the performance of fine-grained classification is significantly improved.

The above algorithms work well on clean image data, but cannot generalize well
when the image samples are corrupted. To handle the image corruption problem, Yang et
al. [26] proposed a robust sparse coding (RSC) scheme to estimate the distribution of
corruptions and learn a distribution induced weight matrix for each testing sample. The
weight matrix is used to detect the error pixels and control the contribution of these pixels
in the reconstruction process. He et al. [27] developed a correntropy-based sparse model
to maximize the correntropy between a testing sample and its reconstructed image. The
corrupted pixels, which have small contributions to the correntropy, will be assigned
smaller weights, such that these corrupted pixels have less influence on the reconstruction
residual of the testing sample. Qian et al. [16] and Yang et al. [28] adopted a nuclear
norm regularization to describe the structural information of the errors in testing samples,
which increases the accuracy of detecting error pixels. They obtained promising results
in dealing with the structural corruptions in testing images. Following the scheme of RSC,
Wei and Wang [17],[19] proposed to learn a robust auxiliary dictionary for undersampled
face recognition, which can significantly enhance robustness to the occlusions in testing
samples. The maximum correntropy criterion [27] is also integrated into conventional
dictionary learning to construct the robust dictionary for dealing with outliers and noise
in testing samples [18],[29]. These algorithms achieved very promising performances
when the testing samples are corrupted. However, they still cannot solve the problem of
training sample corruption, because the corruptions in training data will be carried into their dictionaries, resulting in poor reconstructions and subsequently misclassifications of testing samples.

Low-rank matrix recovery theory [55], in principle, has the potential to handle data corruptions in dictionary learning. Let \( Y = [Y_1, Y_2, ..., Y_C] \in \mathbb{R}^{d \times N} \) be a training sample set, which consists of \( N \) training samples from \( C \) different classes. And let \( Y_i \in \mathbb{R}^{d \times N_i} \) denote a data set containing all the training samples from the \( i \)-th class, where \( d \) is the dimension of images and \( N_i \) is the number of training samples in the \( i \)-th class, which satisfies \( N = \sum_{i=1}^{C} N_i \). Suppose the training samples \( Y \) are contaminated by a corruption matrix \( E \in \mathbb{R}^{d \times N} \), and let the matrix \( Y_o \in \mathbb{R}^{d \times N} \) denote the underlying clean sample matrix, i.e., \( Y = Y_o + E \). The low-rank matrix recovery model aims to recover the clean sample matrix \( Y_o \) from the observed sample set \( Y \). This goal can be achieved by solving the following nuclear norm regularized problem:

\[
\min_{Y_o, E} \|Y_o\|_* + \alpha \|E\|_1 \quad \text{s. t. } Y = Y_o + E,
\]

where \( \alpha > 0 \) is a trade-off parameter to balance the influence of the nuclear norm minimization term \( \|Y_o\|_* \) and the error term \( \|E\|_1 \). \( \|\cdot\|_* \) is the nuclear norm of a matrix, i.e., the sum of the singular values of this matrix. The idea of solving the rank minimization problem in Eq. (1) was initially used in the robust principal component analysis (RPCA) [30], [31] to recover the underlying low-rank structure from corrupted data, in a single subspace. To reveal the membership of samples more accurately, a low-rank representation (LRR) [32] method is developed to generalize RPCA from single subspace to multi-subspaces setting. By integrating rank minimization into traditional sparse representation based dictionary learning framework, Ma et al. [33] and Li et al.
[34] proposed discriminative low-rank dictionary learning methods for robust image classification and achieved impressive classification performance.

3. Low-rank Double Dictionary Learning (LRD$^2$L)

This section presents a novel low-rank double dictionary learning (LRD$^2$L) to address the corruption problem in training samples. We first analyse the properties of class-specific information, class-shared information, and sparse noise in image samples. Based on the analysis, the objective function of LRD$^2$L is designed that will be used in the optimization algorithm presented in the next section.

3.1 The properties of the three components

The information contained in an image can be classified into three types, i.e., the class-specific information, the class-shared information and the sparse noise. Given a dataset $\mathbf{Y} = [\mathbf{Y}_1, \mathbf{Y}_2, ..., \mathbf{Y}_C]$ (see Section 2), we propose to decompose $\mathbf{Y}$ into three components, such that each component only represents one type of information:

$$Y_i = \overline{Y}_i + \tilde{Y}_i + E_i \quad \text{for } i = 1, 2, \ldots, C,$$  

(2)

where $\overline{Y}_i$ and $\tilde{Y}_i$ are the class-specific component and the class-shared component of the sample images from the $i$-th class, $E_i$ represents the sparse noise in the $i$-th class. To appropriately separate these components from an image, we analyse the properties of them as follows.

If images in the dataset are clean (i.e., only containing class-specific information of its class), the images belonging to the same class tend to be drawn from the same subspace, while the images from different classes are drawn from different subspaces. Therefore,
the column vectors in a class-specific component $\mathbf{Y}_i$, which correspond to images from the same class, are highly linearly correlated, whereas the column vectors in two different class-specific components $\mathbf{Y}_i$ and $\mathbf{Y}_j$ where $i \neq j$, which correspond to images from different classes, are independent. Thus, the class-specific component matrix $\mathbf{Y}_i$ for every class $i$, where $i = 1, 2, \ldots, C$, is expected to be low rank.

Since the class-shared components represent the common patterns across different classes, the class-shared components of different classes $\mathbf{Y}_i$ and $\mathbf{Y}_j$ are highly linearly correlated. Therefore, if we stack the class-shared components of all class into one matrix, i.e., $\mathbf{Y} = [\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_C]$, this matrix $\mathbf{Y}$ should be low rank. Note that the matrix $\mathbf{Y}_i$ for a single class, however, may not be low rank.

Because the noise contaminates a relatively small portion of the whole image and the noises in different images are statistically uncorrelated [35], the sparse noise component of each class $\mathbf{E}_i$ should be a sparse matrix.

### 3.2 Building objective function for LRD^2L

In this study, we propose to learn a class-specific dictionary $\mathbf{A} = [\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_C] \in \mathbb{R}^{d \times m_A}$ and a class-shared dictionary $\mathbf{B} \in \mathbb{R}^{d \times m_B}$ to represent the class-specific and the class-shared components of the data, respectively. $\mathbf{A}_i \in \mathbb{R}^{d \times m_{A_i}}$ is a sub-dictionary associated with the $i$-th class. According to the dictionary learning theory, Eq. (2) can be rewritten as:

$$\mathbf{Y}_i = \mathbf{A} \mathbf{X}_i + \mathbf{B} \mathbf{Z}_i + \mathbf{E}_i, \text{ for } i = 1, 2, \ldots, C, \quad (3)$$

where $\mathbf{X}_i$ is the representation coefficient matrix of $\mathbf{Y}_i$ over dictionary $\mathbf{A}$, and $\mathbf{Z}_i$ is the representation coefficient matrix of $\mathbf{Y}_i$ over dictionary $\mathbf{B}$.
Because both the class-specific component of each class $\bar{Y}_i$ and the class-shared component across all classes $\bar{Y} = [\bar{Y}_1, \bar{Y}_2, ..., \bar{Y}_C]$ have the low-rankness property, the class-specific sub-dictionary $A_i$ of each class and the class-shared dictionary $B$ should also be low-rank matrices. Hence we propose the following objective function for our low-rank double dictionary learning:

$$
\min_{A_i, X_i, B, Z_i, E_i} ||A_i||_* + ||B||_* + \alpha(||X_i||_1 + ||Z_i||_1) + \beta||E_i||_1
$$

s.t. $Y_i = AX_i + BZ_i + E_i$ for $i = 1, 2, ..., C,$

where $\alpha$ and $\beta$ are positive-valued parameters that balance the sparsity of the coefficient matrices and the weight of noise, respectively. Since the noise component $E_i$ of each class should be a sparse matrix, the $l_1$-norm regularization $||E_i||_1$ is used to model the sparse noise in each class.

We also adopt two terms to further enhance the discriminative power of the dictionaries. Firstly, since $A_i$ is the sub-dictionary associated with the $i$-th class, it is supposed to well represent the class-specific component of the $i$-th class. Therefore, by vertically partitioning the coefficient matrix $X_i$ as $X_i = [X_{i1}; X_{i2}; ...; X_{iC}]$, where $X_{ij}$ denotes the coefficients of $\bar{Y}_i$ corresponding to $A_j$, we can have the constraint $Y_i = A_iX_{ii} + BZ_i + E_i$. Secondly, the class-specific components of different classes should not be correlated, therefore for $\bar{Y}_j$, the coefficients $X_{ji} (i \neq j)$ are expected to be a zero matrices. To this end, an incoherence term $R(A_i) = \sum_{j=1, j\neq i}^C ||A_iX_{ji}||_F^2$ ($i = 1, 2, ..., C$) is introduced into the objective function. The minimization of such term makes the correlation between $\bar{Y}_j$ and $A_i (i \neq j)$ as small as possible.
Considering both the above two factors, the objective function of Eq. (4) is further improved for LRD\(^2\)L as follow:

\[
\min_{A_i, X_i, B_i, E_i} \|A_i\|_* + \|B\|_* + \alpha(\|X_i\|_1 + \|Z_i\|_1) + \beta\|E_i\|_1 + \lambda R(A_i)
\]

\[
s.t. \ Y_i = A_iX_i + B_iZ_i + E_i, \ Y_i = A_iX_{ii} + B_iZ_i + E_i,
\]

where \(\lambda > 0\) is a parameter that controls the contribution of the incoherence term. \(X_{ii}\) denotes the coefficients of \(\bar{Y}_i\) corresponding to \(A_i\). We will present the algorithm of solving this optimization problem in Section 4.

Different from [20] that decomposes the image data into only two components, our method captures three types of information from the images. By considering the sparse noise in both training and testing phases, the proposed LRD\(^2\)L method is more robust to noise. In addition, the multi-subspace structural information in data is ignored in [20], because the atoms in their dictionaries are processed independently. On the contrary, our method imposes the low-rank constrains on dictionaries to ensure that such structural information in the training images are correctly retained. Therefore, the dictionaries learned by LRD\(^2\)L can decompose each component more accurately, and consequently LRD\(^2\)L achieves higher levels of accuracy and robustness.

4. Optimization of LRD\(^2\)L

In this section, we propose an effective optimization algorithm to solve the problem (7), in which the entire optimization problem is divided into three sub-problems to be solved iteratively. 1) Updating the coefficients \(X_i\) and \(Z_i\) for class \(i (i = 1, 2, ..., C)\) by fixing the dictionaries \(A, B\) and other coefficients \(X_j\) and \(Z_j (i \neq j)\). 2) Updating the sub-
dictionary $A_i$ class-by-class by fixing other variables. The corresponding coefficients $X_{ii}$ should also be updated to satisfy the constraint $Y_i = A_iX_{ii} + BZ_i + E_i$. 3) Updating the class-shared dictionary $B$ and the corresponding coefficients $Z$ iteratively to satisfy the constraint $Y_i = AX_i + BZ_i + E_i$. Similar to [33], the sparse error $E_i$ is updated in each sub-problem. Because the weights for $E_i$ should be adjusted differently for the constraints $Y_i = A_iX_{ii} + BZ_i + E_i$ and $Y_i = AX_i + BZ_i + E_i$ respectively, the parameter $\beta$ in the second sub-problem is set differently from that in the other two sub-problems.

4.1 Updating coefficients $X_i$ and $Z_i$

Given the dictionaries $A$ and $B$, the coefficients $X_i$ and $Z_i$ are updated class by class by fixing all other coefficients $X_j$ and $Z_j$ ($i \neq j$). Thus, by ignoring the terms that are unrelated with $X_i$ and $Z_i$, the problem (5) is reduced to a sparse coding problem as follows:

$$\min_{X_i, Z_i, E_i} \alpha(\|X_i\|_1 + \|Z_i\|_1) + \beta_1\|E_i\|_1$$

s.t. $Y_i = AX_i + BZ_i + E_i$. (6)

Note that $\|\left[X_i; Z_i\right]\|_1 = \|X_i\|_1 + \|Z_i\|_1$ and the above constraint can be rewritten as $Y_i = [A, B][X_i; Z_i] + E_i$. Therefore, by defining $P_i = [X_i; Z_i]$, $D = [A, B]$ and introducing an auxiliary variable $H$, the problem (6) is converted to the equivalent problem:

$$\min_{P_i, E_i, H} \alpha\|H\|_1 + \beta_1\|E_i\|_1$$

s.t. $Y_i = DP_i + E_i$, $P_i = H$. (7)

The above problem can be solved efficiently by the Augmented Lagrange Multipliers
(ALM) [36] method, which minimizes the augmented Lagrange function of problem (7) as

$$
\min_{P_i, E_i, H} \alpha \|H\|_1 + \beta_1 \|E_i\|_1 + \langle T_1, Y_i - DP_i - E_i \rangle + \langle T_2, P_i - H \rangle \\
+ \frac{\mu}{2} (\|Y_i - DP_i - E_i\|_F^2 + \|P_i - H\|_F^2),
$$

where $T_1$ and $T_2$ are Lagrange multipliers and $\mu > 0$ is a positive penalty parameter. $\langle A, B \rangle = Tr(A^T B)$ is the sum of the diagonal elements of the matrix $A^T B$, and $A^T$ is the transpose of the matrix $A$. The optimization procedure of problem (8) is shown in Algorithm 1, where $I$ is an identity matrix. The optimization problem in steps 1 and 3 can be solved by the soft-thresholding or shrinkage operator.
Algorithm 1. Solving Problem (8) via ALM

\textbf{Input:} Data matrix $Y_\iota$, initial dictionary matrix $D = [A, B]$, parameters $\alpha$ and $\beta_1$

\textbf{Initialization:} $E_\iota = T_1 = 0$, $P_\iota = H = T_2 = 0$, $\mu = 10^{-6}$, $\rho = 1.1$, $\epsilon = 10^{-8}$, $\max \mu = 10^{30}$.

\textbf{while} not converged and the maximal iteration number is not reached \textbf{do}

1: Fix the others and update $H$ by:
\[
H = \text{argmin}_H \alpha \|H\|_1 + \langle T_2, P_\iota - H \rangle + \frac{\mu}{2} \|P_\iota - H\|_F^2.
\]

2: Fix the others and update $P_\iota$ by:
\[
P_\iota = (I + D^T D)^{-1} (H + D^T Y_\iota - D^T E_\iota + (D^T T_1 - T_2))/\mu.
\]

3: Fix the others and update $E_\iota$ by:
\[
E_\iota = \text{argmin}_{E_\iota} \beta_1 \|E_\iota\|_1 + \langle T_1, Y_\iota - D P_\iota - E_\iota \rangle + \frac{\mu}{2} \|Y_\iota - D P_\iota - E_\iota\|_F^2.
\]

4: Update the multipliers $T_1$ and $T_2$ by:
\[
T_1 = T_1 + \mu (Y_\iota - D P_\iota - E_\iota), T_2 = T_2 + \mu (P_\iota - H).
\]

5: Update the parameter $\mu$ by:
\[
\mu = \min (\rho \mu, \max \mu).
\]

6: Check the convergence conditions:
\[
\|Y_\iota - D P_\iota - E_\iota\|_\infty < \epsilon, \|P_\iota - H\|_\infty < \epsilon
\]

\textbf{end}

\textbf{Output:} The coefficient matrix $P_\iota = [X_\iota; Z_\iota]$ and the error matrix $E_\iota$.

4.2 Updating class-specific sub-dictionary $A_\iota$

With the learned coefficients $X_\iota$, the sub-dictionary $A_\iota$ is updated class by class, and the corresponding coefficients $X_{\iota i}$ is also updated to meet the constraint. Then, the second sub-problem is reduced to the following optimization problem:
\[
\min_{A_i, X_{ii}, E_i} \|A_i\|_* + \alpha\|X_{ii}\|_1 + \beta_2\|E_i\|_1 + \lambda R(A_i)
\]
\[s.t. \ Y_i = A_i X_{ii} + B Z_i + E_i. \tag{9}\]

For mathematical brevity, we define \(Y_A = Y_i - B Z_i\). Two auxiliary variables \(J\) and \(S\) are introduced and the problem (9) becomes the following equivalent optimization problem:

\[
\min_{A_i, X_{ii}, S, E_i} \|J\|_* + \alpha\|S\|_1 + \beta_2\|E_i\|_1 + \lambda R(A_i)
\]
\[s.t. \ Y_A = A_i X_{ii} + E_i, \ A_i = J, \ X_{ii} = S. \tag{10}\]

Problem (10) can be solved by solving the following Augmented Lagrange Multiplier problem through the ALM method:

\[
\min_{A_i, J, X_{ii}, S, E_i} \|J\|_* + \alpha\|S\|_1 + \beta_2\|E_i\|_1 + \lambda R(A_i)
\]
\[+ \langle T_1, Y_A - A_i X_{ii} - E_i \rangle + \langle T_2, A_i - J \rangle + \langle T_3, X_{ii} - S \rangle \tag{11}\]
\[+ \frac{\mu}{2} (\|Y_A - A_i X_{ii} - E_i\|_F^2 + \|A_i - J\|_F^2 + \|X_{ii} - S\|_F^2), \]

where \(T_1, T_2\) and \(T_3\) are Lagrange multipliers and \(\mu > 0\) is a positive penalty parameter. The derivative of the incoherence term \(R(A_i) = \sum_{j=1,j\neq i}^C \|A_i X_{ji}\|_F^2\) with respect to \(A_i\) is \(\frac{dR(A_i)}{dA_i} = 2 \sum_{j=1,j\neq i}^C A_i X_{ji} X_{ji}^T\). Hence by defining the matrix \(V = \frac{2\lambda}{\mu} \sum_{j=1,j\neq i}^C X_{ji} X_{ji}^T\), the optimization process of problem (11) is presented in Algorithm 2. The optimization problem in step 1 can be solved by using the singular value thresholding (SVT) operator [37]. Same as in [23], [33], normalization operations are used after updating \(J\) and \(A_i\) respectively to make the columns of the learned dictionary be unit vectors.
Algorithm 2. Solving Problem (11) via ALM

Input: Data matrix $Y_A = Y_i - BZ_i$, coefficient matrix $X$, parameters $\alpha, \beta, \lambda$.

Initialization: $E_i = T_1 = 0, A_i = J = T_2 = 0, S = T_3 = 0, \mu = 10^{-6}, \rho = 1.1, 
\varepsilon = 10^{-8}, \max_{\mu} = 10^{30}$, incoherent matrix $V = \frac{2\lambda}{\mu} \sum_{j=1, j\neq i}^c X_{ji}^T X_{ji}$.

while not converged and the maximal iteration number is not reached do

1: Fix the others and update $J$ by

$$J = \arg \min_J \alpha \|J\|_* + \langle T_2, A_i - J \rangle + \frac{\mu}{2} \|A_i - J\|_F^2.$$

normalize the columns in $J$ to unit vectors.

2: Fix the others and update $A_i$ by:

$$A_i = (J + Y_A X_{ii}^T - E_i X_{ii}^T + (T_1 X_{ii}^T - T_2) / \mu) (I + X_{ii} X_{ii}^T + V)^{-1},$$

normalize the columns in $A_i$ to unit vectors.

3: Fix the others and update $S$ by:

$$S = \arg \min_S \alpha \|S\|_1 + \langle T_3, X_{ii} - S \rangle + \frac{\mu}{2} \|X_{ii} - S\|_F^2.$$

4: Fix the others and update $X_{ii}$ by:

$$X_{ii} = (I + A_i^T A_i)^{-1} (S + A_i^T Y_A - A_i^T E_i + (A_i^T T_1 - T_3) / \mu).$$

5: Fix the others and update $E_i$ by:

$$E_i = \arg \min_{E_i} \beta_2 \|E_i\|_1 + \langle T_1, Y_A - A_i X_{ii} - E_i \rangle + \frac{\mu}{2} \|Y_A - A_i X_{ii} - E_i\|_F^2.$$

6: Update the multipliers $T_1, T_2$ and $T_3$ by:

$$T_1 = T_1 + \mu (Y_A - A_i X_{ii} - E_i), T_2 = T_2 + \mu (A_i - J), 
T_3 = T_3 + \mu (X_{ii} - S).$$

7: Update the parameter $\mu$ by:

$$\mu = \min (\rho \mu, \max_{\mu}).$$

8: Check the convergence conditions:

$$\|Y_A - A_i X_{ii} - E_i\|_\infty < \varepsilon, \|A_i - J\|_\infty < \varepsilon, \|X_{ii} - S\|_\infty < \varepsilon$$

end

Output: Sub-dictionary matrix $A_i$, coefficient matrix $X_{ii}$ and error matrix $E_i$. 

16
4.3 Updating class-shared dictionary $B$

When all the class-specific sub-dictionaries are updated, we learn the class-shared dictionary $B$ with all the other variables fixed. Hence, the objective function in (5) is reduced to

$$\min_{B, Z_i, E_i} \|B\|_* + \alpha \|Z_i\|_1 + \beta_1 \|E_i\|_1$$

$$s.t. \ Y_i = AX_i + BZ_i + E_i. \quad (12)$$

Different from the class-specific sub-dictionary $A_i$ that is associated only with an individual class, the class-shared dictionary $B$ is related to image samples across all the classes. Therefore, when updating $B$, we need to consider all the relationships of image samples across different classes altogether. By summing up the objective function in (12) of all the classes, the problem (12) can be converted to the following optimization problem because $\sum_{i=1}^{C} \|Z_i\|_1 = \|[Z_1, Z_2, ..., Z_C]\|_1$:

$$\min_{B, [Z_1, Z_2, ..., Z_C], [E_1, E_2, ..., E_C]} \|B\|_* + \alpha \|[Z_1, Z_2, ..., Z_C]\|_1 + \beta_1 \|[E_1, E_2, ..., E_C]\|_1$$

$$s.t. \ [Y_1, Y_2, ..., Y_C] = A[X_1, X_2, ..., X_C] + B[Z_1, Z_2, ..., Z_C] + [E_1, E_2, ..., E_C]. \quad (13)$$

Since $X = [X_1, X_2, ..., X_C], Z = [Z_1, Z_2, ..., Z_C]$ and $E = [E_1, E_2, ..., E_C]$, the Eq. (13) can be reformulated as

$$\min_{B, Z, E} \|B\|_* + \alpha \|Z\|_1 + \beta_1 \|E\|_1$$

$$s.t. \ Y = AX + BZ + E. \quad (14)$$
Similar to the second sub-problem, we define \( Y_B = Y - AX \) and introduce two auxiliary variables \( Q \) and \( L \), such that the problem (14) is converted to the following equivalent optimization problem:

\[
\min_{B,Q,Z,L} \|Q\|_* + \alpha \|L\|_1 + \beta \|E\|_1
\]

\[
s.t. \quad Y_B = BZ + E, \quad B = Q, \quad Z = L.
\]

The above problem can be solved by solving its Augmented Lagrange Multiplier problem through the ALM method:

\[
\min_{B,Q,Z,L} \|Q\|_* + \alpha \|L\|_1 + \beta \|E\|_1
\]

\[
+ \langle T_1, Y_B - BZ - E \rangle + \langle T_2, B - Q \rangle + \langle T_3, Z - L \rangle
\]

\[
+ \mu \left( \|Y_B - BZ - E\|_F^2 + \|B - Q\|_F^2 + \|Z - L\|_F^2 \right).
\]

We can observe that the objective function of problem (16) has the same form as that of problem (11), except without the incoherence term \( R(A_i) \). Therefore, by setting the parameter \( \lambda \) to zero, the problem (16) can also be solved by Algorithm 2.

4.4 Complete algorithm of LRD\(^2\)L

With the algorithms for the above three sub-problems, the optimization procedures of (8), (11) and (16) need to be iterated a few times to reach the solution of LRD\(^2\)L. The complete algorithm of LRD\(^2\)L is summarized in Algorithm 3.
Similar to [39],[40],[41],[52], the proposed objective function (5) of LRD$^2$L is non-convex to the unknown variables. Since the proposed LRD$^2$L algorithm can only converge to a local minimum, the initialization of each dictionary is important for achieving a desirable solution. As discussed in Section 3, we construct the sub-dictionary $A_i$ to represent the class-specific information of each class, while the class-shared information and the sparse noise should not be captured by it. Therefore, similar to [39], by applying the singular value decomposition (SVD) on each training samples set $Y_i$, we can initialize the sub-dictionary $A_i \in \mathbb{R}^{d \times m_{A_i}}$ as follows:

$$A_i = U_i (1:d,1) S_i (1,1) V_i (1:m_{A_i},1)^T$$

$$A_i(:,i) = A_i(:,i)/\text{norm}(A_i(:,i)) \text{ for } i = 1,2,\ldots,m_{A_i}.$$  \hfill (17)

where $U_i$, $S_i$ and $V_i$ are the singular value decomposition matrices of $Y_i$, i.e., $Y_i = U_i S_i V_i^T$, and the normalization operation makes the columns of $A_i$ be unit vectors. Such an initialization scheme makes sub-dictionary $A_i$ capture the most significant

---

**Algorithm 3.** The Complete Algorithm of LRD$^2$L

**Input:** Data matrix $Y$, initial dictionary matrix $D = [A, B]$, parameters $\alpha, \beta_1, \beta_2, \lambda$.  

**while the maximal iteration number is not reached do**  

1: Fix the others and update $X_i$ and $Z_i$ class by class, by using the Algorithm 1;  
2: Fix the others and update $A_i$ class by class, by using the Algorithm 2;  
3: Fix the others and update $B$ by solving the problem (16);  

end  

**Output:** Class-specific dictionary $A$, class-shared dictionary $B$, coefficient matrix $X$, coefficient matrix $Z$ and error matrix $E$.  

class-specific information with the lowest rank (only rank 1), and maximally removes the
effect of the class-shared information and the noise from $A_i$. For the class-shared
dictionary $B$, same as [23], we initialize the atoms of $B$ with the eigenvectors of $Y$, which
are associated with the largest $m_B$ eigenvalues. Our experiments show that this
initialization strategy can lead to a desirable result. In addition, in each iteration,
Algorithm 1 and Algorithm 2 reinitialize the associated variables and the outputs are the
local minima optimized from the zero matrices. Therefore, similar to the results in
[49],[51],[52], the value of the objective function shown in Fig. 1 slightly fluctuates with
the increase of iteration number. However, after a few iterations, the objective function
value tends to be stable and only varies in a very small range (see Fig. 1). It can be seen that only 10 to 15 iterations can yield stable results. Therefore, for the rest of the experiments in this work, we set the maximal iteration number to 15.

4.5 Computational complexity analysis

In this section, we analyze the computational complexity of our approach. For updating the coefficients $X_i$ and $Z_i$, the computations involve the matrix inversion and the matrix multiplications in Algorithm 1. Since the dictionary $D = [A, B]$ is a $d \times (m_A+m_B)$ matrix, the computational complexity of the matrix inversion calculation in the step 2 is $O((m_A+m_B)^3)$, and the matrix multiplications cost $O(dN_i(m_A+m_B))$.

For updating the sub-dictionary $A_i$ using Algorithm 2, the step 1 requires computing the SVD of a $d \times m_{A_i}$ matrix, whose complexity is $O(dm_{A_i}^2)$. Since $A_i \in \mathbb{R}^{d \times m_{A_i}}$ and $X_{ii} \in \mathbb{R}^{m_{A_i} \times N_i}$, the matrix inversions in the steps 2 and 4 both cost $O(m_{A_i}^3)$, and the matrix multiplication operations have the complexity of $O(dN_i m_{A_i})$. Similarly, for updating the class-shared dictionary $B$ using Algorithm 2, the SVT operator in the step 1 has the complexity of $O(dm_{B}^2)$. The time cost of the matrix inversions and the matrix multiplications are $O(m_{B}^3)$ and $O(dNm_B)$, respectively.

Considering the number of classes and the number of iterations, the overall complexity of the proposed algorithm is $Ct_1O(dN_i(m_A+m_B) + (m_A+m_B)^3) + Ct_2O(dN_i m_{A_i} + dm_{A_i}^2 + m_{A_i}^3) + t_3O(dNm_B + dm_B^2 + m_B^3)$, where $t_1, t_2$ and $t_3$ are the numbers of iterations needed for solving each sub-problem, respectively. $C$ is the number of classes. The entire optimization algorithm can be fast provided that the dictionary sizes $m_A$ and $m_B$ are not too large.

4.6 Classification based on LRD$^2$L
In the previous sections, we have addressed the corruption and noise problem in the training samples at the dictionary learning stage. To handle the corruptions and the noise in testing samples at the testing stage, after learning the class-specific dictionary \( A = [A_1, A_2, \ldots, A_C] \) and the class-shared dictionary \( B \), we encode a testing image \( y \in \mathbb{R}^d \) over the learned dictionaries by solving the following problem:

\[
\begin{equation}
\min_{x, z, e} \|x\|_1 + \gamma \|z\|_1 + \omega \|e\|_1 \\
\text{s.t.} \quad y = Ax + Bz + e, 
\end{equation}
\]

where \( x \) and \( z \) are the coding coefficient vectors of \( y \) in terms of \( A \) and \( B \), respectively. \( e \) is the error vector to approximate the sparse noise in \( y \). This problem has the similar form with the problem (6) and can be solved by ALM algorithm as well. Then, for each class, the testing image \( y \) can be approximated as

\[
y'_i = A\delta_i(x) + Bz + e.
\]

Denote \( x = [x_1; x_2; \ldots; x_c] \), where \( x_i \) is the coding coefficient sub-vector associated with the sub-dictionary \( A_i \). \( \delta_i(x) \) selects the coefficients corresponding to the \( i \)-th class, which is defined as \( \delta_i(x) = [0; 0; \ldots; x_i; \ldots; 0] \). Finally, \( y \) is classified to the class with the smallest reconstruction residual, \( i.e., \text{identity}(y) = \arg\min_i \|y'_i - y\|_2^2 \).

5. Experimental results

To evaluate the effectiveness of the proposed LRD\(^2\)L method, we conducted experiments on three public available databases for face and object recognitions. The
performance of our approach is compared with six state-of-the-art methods, including sparse representation classifier (SRC) [35], fisher discrimination dictionary learning (FDDL) [23], label consistent KSVD [21] version 1 (LC-KSVD1) and version 2 (LC-KSVD2), DL-COPAR [20] and discriminative low-rank dictionary for sparse representation (DLRD_SR) [33]. We implemented SRC and DLRD_SR algorithms. The codes of the other competing methods were downloaded from the authors’ homepages. Same as [40],[41],[42], the image samples used in this paper are normalized to unit length (i.e., the \( \ell_2 \)-norm of each image vector equals to one). As the accuracy of our method is not sensitive to \( \alpha \) and \( \lambda \), we fix \( \alpha = 0.1, \lambda = 1 \) for all the experiments in this paper. \( \beta_1, \beta_2 \) and the parameters in the benchmark algorithms are tuned manually for each experimental setting to the best possible performance of each competing method.

5.1 Experiments on AR dataset

The AR dataset [43] contains more than 4000 frontal view face images of 126 subjects. The face images in AR dataset are corrupted by different types of corruptions, such as illuminations, facial expressions and disguises (sunglasses and scarf). For each subject, there are 26 images captured in two different sessions with a two-week interval. Each session consists of 13 images, with 3 images under different illumination conditions (right light on, left light on and both lights on), 3 images under different expression changes (smiling, angry and screaming) and 6 images with disguises (sunglasses and scarf in combination of illumination changes). Fig. 2 shows example images of one subject from the two sessions.
5.1.1 Tests on data without occlusion

In this experiment, we use a subset of the dataset, which contains 2600 images from the first 50 male individuals and the first 50 female individuals. For each individual, 7 images from the session 1 with different expressions (including neutral) and illuminations are used as training samples, and the 7 images from the session 2 under the same conditions are used as testing samples.

In both of the above training and testing images, a varying percentage (from 10% to 40%) of randomly selected pixels are replaced with noise uniformly distributed over \([0, V_{\text{max}}]\) as used in [33], [38], where \(V_{\text{max}}\) is the maximal pixel value in the image. All the images are cropped and normalized to the size of 60×43 pixels. Fig. 3 shows the average classification accuracies of 5 repeated experiments. Table 1 lists the classification results of different methods without added noise.

It can be seen that, without added noise (i.e., \(\xi = 0\)), our approach achieves a classification accuracy of 97.29%, which is 1.15% higher than the DLRD_SR method and over 4.43% than other five competing methods. When the noise level becomes \(\xi = 20\), the accuracy improvements of the proposed LRD^2L over the DLRD_SR method and the other competing methods increase up to 4.60% and more than 21.02% respectively. The improvements can go up to 17.29% and more than 39.15% when \(\xi = 40\). These
results demonstrate the effectiveness and superiority of the proposed LRD$^2$L over the state-of-the-art methods in recognizing faces from corrupted training and testing data.

Fig. 3. Classification results of different methods on AR dataset under different percentage of random pixel corruption ($\xi$).

Table 1. Classification results of different methods on AR dataset with $\xi = 0$ ($\beta_1 = 0.1, \beta_2 = 0.015$).

<table>
<thead>
<tr>
<th>Methods (Dictionary Size)</th>
<th>Classification Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRC ($7 \times 100 = 700$)</td>
<td>92.14 %</td>
</tr>
<tr>
<td>FDDL ($7 \times 100 = 700$)</td>
<td>92.29 %</td>
</tr>
<tr>
<td>LC-KSVD1 ($7 \times 100 = 700$)</td>
<td>92.00 %</td>
</tr>
<tr>
<td>LC-KSVD2 ($7 \times 100 = 700$)</td>
<td>92.86 %</td>
</tr>
<tr>
<td>DL-COPAR ($7 \times 100 + 7 = 707$)</td>
<td>92.86 %</td>
</tr>
<tr>
<td>DLRD_SR ($7 \times 100 = 700$)</td>
<td>96.14 %</td>
</tr>
<tr>
<td>Proposed LRD$^2$L ($5 \times 100 + 200 = 700$)</td>
<td>97.29 %</td>
</tr>
</tbody>
</table>

Fig. 4 visualizes image decomposition results obtained by DLRD_SR (shown in red box) and the proposed LRD$^2$L (shown in blue box) on AR dataset, under 20% of random pixel corruption. It can be observed by comparing Fig. 4 (b) and Fig. 4 (d) that
the images recovered by $\text{LRD}^2\text{L}$ have much less noise, and preserve more class-shared information, such as facial expression and illumination variations. This results in a much stronger representative capability of the dictionary learned by $\text{LRD}^2\text{L}$ than $\text{DLRD}_\text{SR}$. Furthermore, because the proposed $\text{LRD}^2\text{L}$ separates the class-specific information and the class-shared information through learning different dictionaries, the class-specific components (see Fig.4 (f)) only contain the most discriminative features of each class, i.e. the identity information, and the common patterns (such as the illuminations and the expressions) are represented by the class-shared components (see Fig.4 (g)). Hence, the class-specific sub-dictionaries learned by $\text{LRD}^2\text{L}$ can be more discriminative and robust than the dictionaries learned by $\text{DLRD}_\text{SR}$.

Fig. 4. Images decomposition examples of $\text{DLRD}_\text{SR}$ [33] and $\text{LRD}^2\text{L}$ on AR dataset, under 20% of random pixel noises.
5.1.2 Tests on data with occlusion

In addition to the random noise corruption, we further evaluate the effectiveness of the proposed approach in dealing with real-world disguises in both training and testing samples. Following the experimental setting as in [38], [44], [45], we conduct comparative experiments under three different scenarios:

**Sunglasses:** In this scenario, all seven undisguised images and one randomly selected image with sunglasses from the session 1 of each person are used to construct the training set. The seven undisguised images from the session 2 and all remaining images with sunglasses (the two remaining images from the session 1 and all the three images from the session 2) of each person are used as testing samples. Thus, there are 8 training samples and 12 testing samples for each individual. The sunglasses cover about 20% of the face image.

**Scarf:** In this scenario, all seven undisguised images and one randomly selected image with scarf from the session 1 of each person are chosen to construct the training set. The seven undisguised images from the session 2 and all the remaining images with scarf are used as testing samples. Thus, there are also 8 training samples and 12 testing samples for each individual. The scarf covers about 40% of the face image.

<table>
<thead>
<tr>
<th>Methods (dictionary size)</th>
<th>Scenarios</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>sunglasses</td>
</tr>
<tr>
<td>SRC (= number of training samples)</td>
<td>89.83 %</td>
</tr>
<tr>
<td>FDDL (8 × 100 = 800)</td>
<td>90.08 %</td>
</tr>
<tr>
<td>LC-KSVD1 (8 × 100 = 800)</td>
<td>87.75 %</td>
</tr>
<tr>
<td>LC-KSVD2 (8 × 100 = 800)</td>
<td>89.36 %</td>
</tr>
<tr>
<td>DL-COPAR (8 × 100 + 8 = 808)</td>
<td>90.08 %</td>
</tr>
<tr>
<td>DLRD_SR (8 × 100 = 800)</td>
<td>94.08 %</td>
</tr>
<tr>
<td>Proposed LRD^2L (5 × 100 + 200 = 700)</td>
<td><strong>95.28 %</strong></td>
</tr>
</tbody>
</table>
Mixed: In this scenario, both the sunglasses and scarf occlusions are both considered. For each subject, all seven undisguised images, one randomly selected image with sunglasses, and one randomly selected image with scarf from the session 1 are chosen as training samples. The remaining images are used as testing samples. Thus, there are 9 training samples and 17 testing samples for each individual.

All the experiments are repeated 5 times. Table 2 reports the average classification accuracies of different approaches under different scenarios. Consistently, the proposed LRD$^2$L obtains the best accuracies across all scenarios. More specifically, the proposed LRD$^2$L outperforms DLRD_SR by 1.20% for the sunglasses scenario, 1.65% for the scarf scenario and 2.06% for the mixed scenario. Compared with the other benchmark methods, LRD$^2$L achieves at least 5.20%, 4.65% and 4.12% improvements for the three scenarios, respectively. This demonstrates the effectiveness of the proposed low-rank double dictionary learning method.

5.2 Experiments on Extended YaleB dataset

The Extended YaleB dataset [46] consists of 2414 near frontal face images from 38 individuals, with each individual having around 59-64 images. The images are captured under various laboratory-controlled illumination conditions (some example images are shown in Fig. 5). For each subject, we randomly select 30 face images to build the training set, and the remaining images are used as testing samples. All the images are manually cropped and normalized to the size of 32×32 pixels. The experiments are repeated 5 times and the average classification accuracies are reported in Table 3.
From the experimental results shown in Table 3, it is observed that, with sufficient training samples representing the illumination variants in the testing data, SRC achieves better classification result than LC-KSVD. It also can be seen that FDDL, DL-COPAR and DLRD_SR obtain comparable results with SRC on smaller size dictionaries. This demonstrates that dictionary learning methods can usually learn a more compact and discriminative dictionary than directly using training samples as dictionary. The proposed LRD²L obtains the highest classification accuracy of 99.43%, which improves by 1.31% over DLRD_SR and at least 1.50% over the other benchmark methods.

To further evaluate the robustness to different levels of pixel corruption, 10% to 40% of randomly selected pixels in all training and testing images are corrupted by the random noise as in Section 5.1.1. For each class, 30 images are randomly selected as training samples and the rest are used as testing samples. Each experiment is repeated 5 times and
the average classification accuracies under different levels of pixel corruptions are shown in Fig. 6.

From the Fig. 6, it can be seen that the proposed LRD^2L consistently obtains the highest classification accuracies under varying noise corruption. When $\xi = 20$, the proposed LRD^2L outperforms the DLRD_SR and the other five methods by 11.17% and at least 23.25%, respectively. Such improvement increases to 17.49% and at least 30.92% respectively, when the noise level increases to 40% (i.e., $\xi = 40$). This demonstrates the robustness and the effectiveness of the proposed LRD^2L method comparing to the other competing methods.

5.3 Experiments on COIL-20 dataset
The COIL-20 object dataset [47] consists of 1440 grayscale images from 20 objects with various poses. Each object contains 72 images, which are captured in equally spaced views, i.e., one image is taken for every 5 degree interval. Fig. 7 shows an example set of images from one object with different viewpoints. In this experiment, same as in [48], [49], 10 images per class are randomly selected as training samples, while the remaining images are used as testing samples. All the images are manually cropped and resized to 32×32 pixels. All experiments are repeated 5 times and the average classification accuracies of different approaches are reported in Table 4.

![Example images of the COIL-20 dataset.](image)

Table 4. Classification results of different methods on COIL-20 dataset with $\xi = 0$ ($\beta_1 = 0.07, \beta_2 = 0.04$).

<table>
<thead>
<tr>
<th>Methods (dictionary size)</th>
<th>Classification Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRC ($10 \times 20 = 200$)</td>
<td>87.50%</td>
</tr>
<tr>
<td>FDDL ($10 \times 20 = 200$)</td>
<td>87.98%</td>
</tr>
<tr>
<td>LC-KSVD1 ($10 \times 20 = 200$)</td>
<td>90.68%</td>
</tr>
<tr>
<td>LC-KSVD2 ($10 \times 20 = 200$)</td>
<td>90.87%</td>
</tr>
<tr>
<td>DL-COPAR ($10 \times 20 + 10 = 210$)</td>
<td>90.42%</td>
</tr>
<tr>
<td>DLRD_SR ($10 \times 20 = 200$)</td>
<td>89.63%</td>
</tr>
<tr>
<td>Proposed LRD^2L ($6 \times 20 + 80 = 200$)</td>
<td>92.05%</td>
</tr>
</tbody>
</table>

As shown in Table 4, DLRD_SR does not work as well as in the previous two experiments and only achieves a classification rate of 89.63%, which becomes lower than those of DL-COPAR and LC-KSVD. This may be because that, without enough training data, the correlations between some training samples from the same class can be non-
linear due to the pose variations existing in the dataset. This makes the low-rankness property of each class less obvious. And due to the low-rank regularization on each sub-dictionary, a portion of class-shared information, i.e. pose variants information, are removed with sparse noise. Consequently, DLRD_SR cannot well represent the testing samples with similar pose variants, which results in a degraded classification accuracy. Different from DLRD_SR, by learning a class-shared dictionary to capture the pose variants, the proposed LRD²L achieves the highest classification accuracy of 92.05%, still outperforming the competing methods with margins between 1.18% and 4.55%.

![Classification results of different methods on COIL-20 dataset under different percentage of random pixel corruption (ξ).](image)

To evaluate the robustness of LRD²L on COIL-20 object dataset, we replace 10% to 40% of randomly selected pixels in all training and testing images with pixel value of 255 using the same strategy as in [49], [50]. All experiments are repeated 5 times and the
average classification accuracies of different methods are shown in Fig. 8. Again, the proposed LRD\(^2\)L consistently outperforms the competing methods under different levels of pixel corruptions. With 10% or more corruptions, our approach performs better than DLRD\(_{SR}\) and DL-COPAR by over 3.82% improvements, and obtains at least 6% improvement over the other competing methods.

It is very encouraging to see that the proposed LRD\(^2\)L consistently outperforms the state-of-the-art approaches in all the experiments, showing that the dictionaries learned by our approach are more discriminative and more robust to the corruptions in both training and testing samples.

5.4 Effect of \(\beta_1\) and \(\beta_2\)

To analyze the influence of the parameters \(\beta_1\) and \(\beta_2\) on the performance of the proposed LRD\(^2\)L, experiments are conducted by varying their values from 0.001 to 1. The classification accuracies on all the three datasets are shown in Fig. 9. Note that, for each of the three databases, the classification accuracy of the proposed LRD\(^2\)L consistently reaches its peak value and remains flat when \(\beta_1\) and \(\beta_2\) are in the range of [0.01, 0.1], which indicates that our method is relatively insensitive to \(\beta_1\) and \(\beta_2\). When \(\beta_1\) and \(\beta_2\) become too small (less than 0.01), the accuracy decreases because some of the class-specific and class-shared information is incorrectly balanced to the sparse noise component reducing the representation ability of each dictionary. On the other hand, the accuracy also drops when \(\beta_1\) and \(\beta_2\) become greater than 0.1, because the error matrix \(E\) is too sparse to correctly capture the noise in the images.
Effect of the incoherence term $R(A_i)$

To evaluate the effect of the incoherence term $R(A_i)$, experiments are conducted to compare the performances of LRD$_{2}$L with and without $R(A_i)$ on the three datasets. Their classification results are reported in Table 5. It can be seen that the incoherence term $R(A_i)$ consistently improves the classification accuracy of LRD$_{2}$L on all datasets. With $R(A_i)$, the classification accuracy of LRD$_{2}$L increases by 0.58%, 0.30% and 0.49% respectively on the three datasets. This demonstrates that reducing the coherence between
the class-specific sub-dictionaries of different classes by \( R(A_i) \) can effectively increase the discriminability of LRD\(^2\)L.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Classification Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LRD(^2)L with ( R(A_i) )</td>
</tr>
<tr>
<td>AR</td>
<td>97.29%</td>
</tr>
<tr>
<td>Extended YaleB</td>
<td>99.43%</td>
</tr>
<tr>
<td>COIL-20</td>
<td>92.05%</td>
</tr>
</tbody>
</table>

5.6 Evaluation of classification time

We conducted experiments on the three datasets to compare the computational time of the proposed LRD\(^2\)L algorithm with the benchmark methods. Table 6 summarizes the average classification time of all the methods on the three datasets. All the experiments are conducted on a 64-bit computer with Inter i7-4710MQ 2.5GHz CPU and 16GB RAM under the MATLAB R2014a programming environment. As shown in Table 6, LC-KSVD costs much less time than the other methods, due to the efficiency of its classification scheme. The classification process of LRD\(^2\)L is much faster than SRC and FDDL. And the time cost of LRD\(^2\)L is close to that of DLRD_SR, because they use similar ALM schemes for classification.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Classification Time (second)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AR</td>
</tr>
<tr>
<td>SRC</td>
<td>4058.47</td>
</tr>
<tr>
<td>FDDL</td>
<td>4045.83</td>
</tr>
<tr>
<td>LC-KSVD1</td>
<td>9.88</td>
</tr>
<tr>
<td>LC-KSVD2</td>
<td>9.86</td>
</tr>
<tr>
<td>DL-COPAR</td>
<td>26.35</td>
</tr>
<tr>
<td>DLRD_SR</td>
<td>122.74</td>
</tr>
<tr>
<td>Proposed LRD(^2)L</td>
<td>178.13</td>
</tr>
</tbody>
</table>

6. Conclusion
In this paper, we presented a novel low-rank double dictionary learning ($\text{LRD}^2\text{L}$) approach that can effectively handle corruptions in both training and testing data, to achieve robust image classification. The proposed $\text{LRD}^2\text{L}$ method incorporates the low-rank matrix recovery technique with the class-specific and class-shared dictionary learning. More specifically, our method simultaneously learns a low-rank class-specific dictionary for each class to capture the class-specific information owned by each class, a low-rank class-shared dictionary to represent the common features shared by all classes, and a sparse error term to approximate the sparse noise in image data. Based on the analysis on the properties of the three types of information, by imposing the low-rank regularizations on each class-specific sub-dictionary and class-shared dictionary together with the sparse noise term, the proposed method can better separate the corruptions and noise in training samples from affecting the construction of class-specific sub-dictionaries. In this way, the low-rank class-specific dictionary only captures the discriminative features of each class for correctly reconstructing and classifying new testing samples. A new alternative optimization algorithm is developed to solve the optimization problem for the proposed method. Experiments on face and object recognition tasks demonstrate the effectiveness and superiority of the proposed $\text{LRD}^2\text{L}$ approach compared to the state-of-the-art benchmark methods, especially when the data are corrupted.

Acknowledgements

This work was supported in part by National Key Research & Development Program of China (2016YFD0101903).
References


