Two-dimensional relaxed representation

Qiulei Dong*
National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences, P.O. Box 2728, Beijing 100190, PR China

A R T I C L E   I N F O
Article history:
Received 8 October 2012
Received in revised form 1 March 2013
Accepted 26 April 2013
Communicated by Jinhui Tang
Available online 22 June 2013

Keywords:
Relaxed representation
Sparsity
Image classification

A B S T R A C T
In this paper, a novel classification framework called two-dimensional relaxed representation (2DRR) is proposed for image classification. Different from recent popular vector-based representations with/without sparsity which encode a vector signal as a sparse/nonsparse linear combination of elementary vector signals, 2DRR is based on 2D image matrices, where each column of the input matrix signal is represented by a combination of the corresponding columns of the elementary matrices. In order to preserve the global linear coding relationship between the input matrix and these elementary matrices, the proposed 2DRR constrains the coding coefficients corresponding to each column of the input matrix to be locally close. Then two algorithms are derived from the 2DRR framework under the $l_2$ norm and the $l_1$ norm respectively. Extensive experimental results show the effectiveness of the proposed algorithms in comparison to three existing algorithms.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

Inspired by the finding in low-level human vision [1,2] that natural images can be generally coded by a sparse combination of structural primitives involving both additive and subtractive interactions, sparse representation has been investigated extensively in pattern recognition and image processing [3–8].

Sparse representation is a kind of representation that accounts for an unknown signal with a linear combination of a small number of elementary signals. Recently, there has been an increasing interest in the sparse representation classification problem. Huang and Aviyente [4] proposed a sparse-representation-based algorithm for signal classification which incorporated reconstruction properties, discrimination power and sparsity. Wright et al. [6] proposed a sparse representation classification (SRC) algorithm for face recognition, where they calculated the sparse representation of an input face image in terms of a set of training images, and performed classification by checking which class yields the least residual. And an extended version of the SRC algorithm was proposed to handle face misalignment in [9].


It is worthy to point out that there exist arguments on whether or not the ‘sparsity’ helps classification recently [11–14]. As opposed to the SRC algorithm [6], Shi et al. [12] considered that the sparsity assumption is not supported by facial data, and proposed an $l_2$-norm-based classification algorithm (denoted as $l_2C$ in this paper) for the face recognition problem. Furthermore, Zhang et al. [13] analyzed the working mechanism of SRC, and proposed a collaborative representation based classification algorithm with regularized least square (CRC_RLS), which can be considered as a regularization version of the $l_2C$ algorithm [12]. Their experiments shown that the CRC_RLS algorithm has very competitive accuracy for face recognition but with significantly lower complexity compared with the SRC algorithm.

Addressing Shi’s criticism [12] on SRC, Wright et al. [14] gave a discussion on the discrepancy between [6,12] within the context of a richer set of experimental results which demonstrate the advantage of the sparse representation.

Here, we do not give a direct evaluation on which is better between SRC and CRC_RLS (or $l_2C$), as well as whether it is necessary to impose the $l_1$-norm sparsity constraint on the coding coefficients or not, but propose a classification framework based on image matrices by relaxing the sparsity/nonsparse constraint with respect to the training sample vectors [6,13,12] to the constraint with respect to the training sample matrices, called two-dimensional relaxed representation (2DRR). Then two algorithms are derived from the 2DRR framework under the $l_2$ norm and the $l_1$ norm respectively, which can be considered respectively as a generalized version of the SRC algorithm and a generalized version of the CRC_RLS algorithm.

The remainder of this work is organized as follows: Section 2 introduces related work. Section 3 formulates the 2DRR classification framework, and proposes two algorithms from this framework under the $l_2$ norm and the $l_1$ norm respectively. Extensive experimental results show the effectiveness of the proposed algorithms in comparison to three existing algorithms.
results are reported in Section 4. Finally, some concluding remarks are listed in Section 5.

2. Related work

Here is a brief review of the SRC algorithm [6] which uses the $l_1$-norm to regularize the coding coefficient vector, as well as the CRC_RLS algorithm [13], which uses the $l_2$-norm to regularize the coding coefficient vector.

For ease of reading, the main notations in the following sections are defined here: Let $Y = [y_1, y_2, ..., y_n] \in \mathbb{R}^{m \times n}$ denote a test image of size $m \times n$, $y(i = 1, 2, ..., n)$ the $i$-th column of $Y$, $y = [y_1^T, y_2^T, ..., y_n^T]$ the column-wise vectorization of $Y$. Let $D = \{D_1, ..., D_l, ..., D_k\}$ denote a given training sample set consisting of $k$ subsets, where $k$ is the class number and $D_i(i = 1, 2, ..., k)$ is the subset containing the samples associated with the $i$-th class.

2.1. Sparse representation for classification

The SRC algorithm [6] employed sparse coding for face recognition based on the assumption that a test face image can be well represented by a small number of the training images. It consists of two main steps: First, the test image is coded sparsely in terms represented by a small number of the training images. It consists of all the training samples under the $l_2$-norm-regularizer, i.e. the designed representation in the $l_2$-norm-regularizer. And their difference is the norm of the coding coefficient vector: the $l_1$ norm is used in SRC for constraining the coefficient vector to be sparse, while the $l_2$ norm is used in CRC for making its solution stable and introducing a certain amount of ‘sparsity’ [13].

However, in many cases, a datum is in a matrix form, e.g. image data, and it may lose its spatial information to transform it into a vector. In addition, many applications in pattern recognition including face recognition are typical small-sample-size problems. Taking the face recognition as an example, even although these two algorithms have tackled such a small-sample-size problem by encoding the testing sample with respect to the dictionary consisting of all the training samples under the $l_1$-norm constraint and the $l_2$-norm constraint respectively as (3) and (5), their reconstruction residuals may be still large so that the consequent classification becomes unstable, especially when the number of the training samples is sufficiently smaller than the sample's dimension, which is a common phenomena in many real applications.

Addressing the above discussions, we propose a representation framework for encoding images more accurately based on image matrices, rather than vectors as in SRC and CRC_RLS, called 2D relaxed representation (2DRR). In the proposed 2DRR, a test image is coded column by column in terms of the corresponding columns of the training images in order to improve the representation accuracy, and simultaneously the coding coefficient vectors associated with neighboring columns of the test image are constrained to be locally close so that the global coding relationship between the test data and the training samples is preserved to a certain extent. More concretely, assuming that the training set is composed of $N$ images of size $m \times n$, $n$ dictionaries $H_i \in \mathbb{R}^{m \times N}$ ($i = 1, ..., n$) are constructed respectively by the $i$-th columns of all the training samples. $H_i(i = 1, ..., n)$ is used to code the $i$-th column of the test sample, and then these obtained coding coefficient vectors corresponding to neighboring columns are required to be as close as possible. Therefore, the 2DRR framework is

3. Two-dimensional relaxed representation

3.1. Motivation and formulation

According to the review of SRC and CRC_RLS in Section 2, it is noted that they are both vector-based algorithms, which represent an input vector with a combination of elementary vectors. And their difference is the norm of the coding coefficient vector: the $l_1$ norm is used in SRC for constraining the coefficient vector to be sparse, while the $l_2$ norm is used in CRC for making its solution stable and introducing a certain amount of ‘sparsity’ [13].

It is necessary to point out that the $l_2$C algorithm [12] is highly similar to the CRC_RLS algorithm in essence. There are only two slight differences between them: (i) the $l_2$C algorithm only calculate the least squares solution to the first item in (5), but does not introduce any $l_2$-norm-regularizer, i.e. the designed representation in the $l_2$C algorithm is a special case of collaborative representation with $\lambda = 0$. (ii) The basic reconstruction residual (4) is used as the classification criterion in the $l_2$C algorithm, while the regularized residual (7) is used in the CRC_RLS algorithm.

The following regularized least square problem:

$$\hat{a} = \arg \min ||y - D\alpha||_2^2 + \lambda ||\alpha||_2^2$$

where $\lambda$ is the tuning parameter. Obviously, the solution to (5) is

$$\hat{a} = (D^T D + \lambda I)^{-1} D^T y$$

where $(\cdot)^*$ denotes the pseudo-inverse, $I$ denotes the identity matrix.

Finally, some concluding remark are listed in Section 5.
formulated as the following optimization problem:
\[
\min \lambda \sum_{i=1}^{n} \mathcal{R}_q(a_i) + \eta \sum_{i=1}^{n} \sum_{j \in \mathcal{N}(i)} \|a_i - a_j\|_2^2 W_{ij}
\]
subject to:
\[
\|Y - [H_1 a_1, H_2 a_2, \ldots, H_n a_n]\|_2^2 \leq \epsilon
\]
where \(\mathcal{R}_q(\cdot)\) is an \(\ell_q\)-norm-regularizer, i.e. \(\mathcal{R}_1(\cdot) = \|\cdot\|_1\) and \(\mathcal{R}_2(\cdot) = \|\cdot\|_2\). \(\mathcal{N}(i)\) denotes the \(k\)-neighborhood of \(i\). \(W_{ij}\) is a sparse symmetric weight matrix, where each element \(W_{ij}\) indicates the weight of the penalty item for the coding coefficients corresponding to the neighboring columns. \(W_{ij}\) can be set in the 0/1 manner or in the heat kernel manner.

Based on the Lagrange multiplier method, we can rewrite the formulation (8) of 2DRR as:
\[
\min \|Y - [H_1 a_1, H_2 a_2, \ldots, H_n a_n]\|_2^2 + \frac{\lambda}{n} \sum_{i=1}^{n} \mathcal{R}_q(a_i)
\]
\[
+ \eta \sum_{i=1}^{n} \sum_{j \in \mathcal{N}(i)} \|a_i - a_j\|_2^2 W_{ij}
\]
According to the designed 2DRR, two algorithms are derived from the formulation (9) under the \(l_2\) norm and the \(l_1\) norm respectively, which are described in detail in the following two subsections.

3.2. 2DRR\(_l^2\) algorithm

When \(q = 2\), i.e. \(\mathcal{R}_2(a_i) = \|a_i\|_2^2\) in (9), the original optimization problem becomes:
\[
\min \|Y - [H_1 a_1, H_2 a_2, \ldots, H_n a_n]\|_2^2 + \frac{\lambda}{n} \sum_{i=1}^{n} \|a_i\|_2^2
\]
\[
+ \eta \sum_{i=1}^{n} \sum_{j \in \mathcal{N}(i)} \|a_i - a_j\|_2^2 W_{ij}
\]
(10)

Let \(a_i = [a_{i1}, a_{i2}, \ldots, a_{ip}]^T\), then (10) is reformulated as:
\[
\min \|Y - H a_i\|_2^2 + \frac{\lambda}{n} \|a_i\|_2^2 + 2\eta \|a_i\|_2^2 S_{a_i}
\]
where
\[
H = \begin{bmatrix}
H_1 & 0 & \cdots & 0 \\
0 & H_2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & H_n
\end{bmatrix}
\]
\(S = (Y^T W Y)_{NN}\), \(Y\) is a diagonal matrix with \(Y_{ii} = \sum_{j} W_{ij}\). \(I_N\) is the \(N\)-order identity matrix.

Obviously, (11) is a least-squares problem, whose least-squares solution is:
\[
\hat{a}_i = \left(H^T H + \frac{\lambda}{n} I_N + 2\eta S_{a_i}\right)^{-\dagger} H^T Y
\]
(12)
where \(\dagger\) denotes the pseudo-inverse.

Then similar to (7), the test image \(Y\) is recognized as the class which gives the minimum reconstruction residual according to the calculated coefficient vectors \(\{\hat{a}_i\}_{i=1}^{n}\)
\[
c = \arg\min_{i} \|H_1 \hat{a}_i(\hat{a}_1), H_2 \hat{a}_i(\hat{a}_2), \ldots, H_n \hat{a}_i(\hat{a}_n)\|_2/\|\hat{a}_i\|_2
\]
(13)

The complete recognition procedure is summarized in Algorithm 1, denoted as 2DRR\(_l^2\).

**Remark.** When the number \(N\) of the training samples is too large, the sizes of the matrices \(H\) and \(S\) in (12) would be large accordingly and it would be of high computational cost to calculate the matrix inverse in (12). Therefore, in this case, the alternating least squares (ALSs) strategy is used to calculate the solution to (10), i.e. at each alternation step only \(a_i\) is updated by fixing the rest variables \(a_1, a_{i-1}, a_{i+1}, \ldots, a_n\).

### Algorithm 1. 2DRR\(_l^2\)

**Input:** A training image set \(D\) for \(k\) classes, a test image \(Y\)

**Output:** \(Y\)s label \(c\)

1. Normalize the training images to have unit Frobenius norm;
2. Construct \(\{H_i\}_{i=1}^{n}\) according to \(D\);
3. Code \(Y\) over \(\{H_i\}_{i=1}^{n}\) according to (12);
4. Compute the regularized residuals:
5. \(e_i(Y) = \|H_1 \hat{a}_i(\hat{a}_1), H_2 \hat{a}_i(\hat{a}_2), \ldots, H_n \hat{a}_i(\hat{a}_n)\|_2/\|\hat{a}_i\|_2\)
6. Identity of \(Y\): \(c = \arg\min_i(e_i(Y))\)

3.3. 2DRR\(_l^1\) algorithm

When \(q = 1\), i.e., \(\mathcal{R}_1(a_i) = \|a_i\|_1\) in (9), the original optimization problem becomes:
\[
\min \|Y - [H_1 a_1, H_2 a_2, \ldots, H_n a_n]\|_2^2 + \frac{\lambda}{n} \sum_{i=1}^{n} \|a_i\|_1
\]
\[
+ \eta \sum_{i=1}^{n} \sum_{j \in \mathcal{N}(i)} \|a_i - a_j\|_2^2 W_{ij}
\]
(14)

Similar to (10), (14) is reformulated as:
\[
\min_{a_i} \|Y - H a_i\|_2^2 + \frac{\lambda}{n} \|a_i\|_1 + 2\eta \|a_i\|_2 S_{a_i}
\]
(15)

The second-order cone programs (SOCPs) [19] can be used to solve this \(l_1\)-regularized problem (15) effectively.

Then similar to (4), the test image \(Y\) is recognized as the class which gives the minimum reconstruction residual according to the calculated coefficient vectors \(\{\hat{a}_i\}_{i=1}^{n}\)
\[
c = \arg\min_i \|H_1 \hat{a}_i(\hat{a}_1), H_2 \hat{a}_i(\hat{a}_2), \ldots, H_n \hat{a}_i(\hat{a}_n)\|_2
\]
(16)

The complete recognition procedure is summarized in Algorithm 2, denoted as 2DRR\(_l^1\).

**Algorithm 2. 2DRR\(_l^1\)**

**Input:** A training image set \(D\) for \(k\) classes, a test image \(Y\)

**Output:** \(Y\)s label \(c\)

1. Normalize the training images to have unit Frobenius norm;
2. Construct \(\{H_i\}_{i=1}^{n}\) according to \(D\);
3. Code \(Y\) over \(\{H_i\}_{i=1}^{n}\) by solving the minimization problem (15);
4. Compute the residuals:
5. \(e_i(Y) = \|H_1 \hat{a}_i(\hat{a}_1), H_2 \hat{a}_i(\hat{a}_2), \ldots, H_n \hat{a}_i(\hat{a}_n)\|_2\)
6. Identity of \(Y\): \(c = \arg\min_i(e_i(Y))\)

3.4. Algorithmic analysis

#### 3.4.1. Parameter selection

As seen from (9) that the introduced parameter \(\eta\) is used to tune the tradeoff between the reconstruction residual and the distribution of the coding coefficient vectors. When \(\eta \to 0\), the constraint on the pairwise coding coefficient vectors corresponding to neighboring columns of the test image are weakened, and then the reconstruction residual becomes lower, but the coding vectors may be distributed more dispersively due to the loss of the global coding relationship between the test image and the training images, consequently more columns of the test image may be represented mainly by a combination of the corresponding columns of interclass training images. When \(\eta \to \infty\), the constraint
on the pairwise coding coefficient vectors corresponding to neighboring columns of the test image are reinforced, but the reconstruction residual is prone to be larger, consequently the classification becomes unstable, especially when the number of the training samples is sufficiently smaller than the sample's dimension. In the extreme case when $\eta = \infty$, all the coding vectors are required to be equivalent, then the 2DRR under the $l_1/l_2$ norm is in essence equivalent to SRC/CRC_RLS.

In addition, as seen from (9), the parameter $\eta$ tunes the tradeoff between the reconstruction residual and the sparsity of the solution to (9). (The $l_2$-norm introduces a certain amount of sparsity to the solution although this sparsity is weaker than that by the $l_1$-norm [13].) With the increase of $\eta$, the sparsity of the solution is reinforced, and the reconstruction residual increases so that the representation accuracy decreases. Therefore, in order to represent an input image effectively, $\eta$ is generally set to a relatively small/moderate value. When $\lambda = 0$ and $\eta = 0$, the 2DRR with only the first item in (9) is in essence equivalent to $l_2C$.

Summarizing the above discussions, learning these two tuning parameters is necessary for obtaining a good classification performance. Here, the leave-one-out cross-validation is used to determine these two parameters.

### 3.4.2. Relationship between 2DRRL, 2DRRl, and SRC/CRC_RLS/$l_2C$

From the above discussions, it is noted that 2DRR is constructed by relaxing the constraint on the coding coefficient vector with respect to the training vector samples [6,13,12] to the constraint on a set of coding coefficient vectors in terms of the columns of the training matrix samples, which can be considered as a generalization of the sparse representation (SR) [6], the collaborative representation (CR) [13], and the $l_2$-norm-based representation [12], i.e. all these three representations are special cases of 2DRR under different constraints. More concretely, when $\eta = \infty$, 2DRRL is in essence equivalent to SRC/CRC_RLS. When $\lambda = 0$ and $\eta = \infty$, the 2DRRL is equivalent to $l_2C$. It can be concluded further that, when 2DRRL/2DRRl and SRC/CRC_RLS/$l_2C$ are used for classification with a given set of training samples, the lowest value of the misclassification rates computed by 2DRRL/2DRRl with different combinations of $\lambda$ and $\eta$ should be no larger than the lowest value of the misclassification rates computed by SRC (CRC_RLS) with different choices of $\lambda$. Extensive experimental results in Table 4 confirm this conclusion.

### 3.4.3. Reconstruction residual analysis

Reconstruction residual can reflect the representation accuracy of an algorithm effectively. Here, the reconstruction residuals with respect to all the training samples by our algorithms, SRC, CRC_RLS, and $l_2C$ are analyzed. According to the first item in (3) and (5), the reconstruction residual by the vector-based algorithms (SRC, CRC_RLS, and $l_2C$) is defined as

$$ RV(a) = \|y-\hat{D}a\|_2 $$

Similarly, according to the first item in (9), the reconstruction residual by our matrix-based algorithms is defined as:

$$ RM(\{\hat{a}_i\}_{i=1}^n) = \|Y[H_1\hat{a}_1, H_2\hat{a}_2, \ldots, H_n\hat{a}_n]\|_2 $$

Obviously, both $RV(a)$ and $RM(\{\hat{a}_i\}_{i=1}^n)$ are used for measuring the difference between the input image and the corresponding reconstructed images under the $l_2$ norm. We introduce the following proposition on the reconstruction residuals by 2DRRL and $l_2C$, and give a brief proof.

**Proposition.** Let $\hat{a}$ be the optimal solution to the minimization problem (5) with $\lambda = 0$, and let $\{\hat{a}_i\}_{i=1}^n$ be the optimal solution to the minimization problem (10) with $\lambda = 0$. For an arbitrarily selected parameter $\eta$, $RM(\{\hat{a}_i\}_{i=1}^n) \leq RV(\hat{a})$.

**Proof.** Since $\{\hat{a}_i\}_{i=1}^n$ is the optimal solution to the minimization problem (10) with $\lambda = 0$, then

$$ \|Y[H_1\hat{a}_1, H_2\hat{a}_2, \ldots, H_n\hat{a}_n]\|_2 + \eta \sum_{i=1}^n \|\hat{a}_i-u_i\|_2^2 \leq \|Y[H_1\hat{a}_1, H_2\hat{a}_2, \ldots, H_n\hat{a}_n]\|_2 + \|u-a\|_2^2 $$

$$ = \|Y[H_1\hat{a}_1, H_2\hat{a}_2, \ldots, H_n\hat{a}_n]\|_2 + \eta \sum_{i=1}^n \|\hat{a}_i-u_i\|_2^2 $$

$$ \leq \|y-\hat{D}a\|_2^2 $$

Therefore, according to the definitions (17) and (18), we have

$$ RM(\{\hat{a}_i\}_{i=1}^n) \leq RV(\hat{a}) \leq RV(\hat{a}) $$

This proposition shows that the proposed 2DRRL is able to represent an input image more accurately compared with $l_2C$. Furthermore, when $\lambda$ is set to a small/moderate value but not zero, we speculate from the proof of the introduced proposition that the similar conclusion on the reconstruction residuals by 2DRRL/2DRRl and CRC_RLS (SRC) is tenable, i.e. the computed reconstruction residual by 2DRRL/2DRRl is smaller than CRC_RLS ($l_2C$) under a same small/moderate $\lambda$. Currently, we are not able to give a theoretical proof for the speculation yet. However, extensive experimental results in Section 4 show that the speculation holds true.

### 4. Experiments

The proposed 2DRRL and 2DRRl are all implemented on a Core 2 Duo 2.53 GHz PC. To further evaluate the proposed algorithms, they are compared with three state-of-the-art algorithms, SRC [6], $l_2C$ [12], and CRC_RLS [13].

#### 4.1. Databases

All these algorithms are tested on the following three face databases where all the images are normalized to have unit Frobenius norm:

1. Yale database [20]: The Yale database consists of 165 gray images of 15 persons. The images contain variations in lighting condition (left-light, center-light, right-light), facial expression (normal, happy, sad, sleepy, surprised and wink), and with/without glasses. The original face images in Yale database are aligned by fixing the locations of two eyes first, and then they are cropped and resized to 32 × 32 pixels.

2. ORL database [21]: The ORL database contains 400 images of 40 individuals. The images are captured at different times and with different variations including expression (open or closed eyes, smiling or non-smiling) and facial details (glasses or no-glasses). Each image is manually cropped and normalized to the size of 32 × 32 pixels.

3. AR database: A subset which contains 50 male subjects and 50 female subjects, is chosen from the AR database [22] in this experiments. For each subject, 14 images with only illumination variations (normal, happy, sad, sleepy, surprised and wink), and with/without glasses, are used for measuring the similar conclu-

#### 4.2. Numerical results

At first, the images in the Yale database are used to test the influence of the parameter $p$ which is used to describe the size of the neighborhood $Ne(\cdot)$ in (8), and the weight matrix $W$ is defined
in the 0/1 manner. A random subset with \( m(=3, 4, 5) \) images for each person in the Yale database is selected for training and the rest for testing. For each given \( m \), both 2DRR\(_1\) with \( p=2, 4, 6, 8, 10 \) and 2DRR\(_2\) with \( p=2, 4, 6, 8, 10 \) are performed independently 10 times, and the corresponding mean value of the error rates (MVERs) and the standard deviation of the error rates (SDERs) are shown respectively in Tables 1 and 2. It can be noted from these two tables that the computed MVER by 2DRR\(_1\)/2DRR\(_2\) varies slightly with \( p \), probably because \( p \) controls the size of the neighborhood of each coding coefficient vector and indirectly influences the closeness between the coding coefficient vectors associated with neighboring columns of a test image to some extent. In the above experiments, 2DRR\(_1\) with \( p=4 \) achieves comparably good performance in most cases, while 2DRR\(_2\) with \( p=6 \) achieves comparably good performance in most cases. Therefore, in the following experiments, the parameter \( p \) for 2DRR\(_1\) is set to 4, and the parameter \( p \) for 2DRR\(_2\) is set to 6.

To further test the performances of all these algorithms, in each of the referred databases in Section 4.1, a random subset with \( m(=3, 4, 5) \) images for each person is selected for training and the rest for testing. For each given \( m \), all the algorithms are performed independently 10 times, and the MVER and the SDER in these databases are shown respectively in Tables 3–5.

In addition, to compare the reconstruction accuracy of all these algorithms, for each given \( m(=3, 4, 5) \), Table 6 lists respectively the mean value of the reconstruction residuals (MVRRs) and the standard deviation of the reconstruction residuals (SDRRs) according to (17) and (18) in the Yale database by all these algorithms, and Fig. 1 shows a set of the corresponding reconstructed images. Here are some points revealed in Tables 3–6, and Fig. 1:

1. As seen from Tables 3–5, in all the referred databases, 2DRR\(_2\) performs better than CRC\(_R\)_L and \( l_2 \)C, while 2DRR\(_1\) performs better than SRC, which confirms the conclusion in Section 3.4.2.

2. Compared with 2DRR\(_1\), the error rates computed by 2DRR\(_2\) are lower in the Yale database and the AR database, but higher in the ORL database. Compared with SRC, the error rates computed by CRC\(_R\)_L are higher in the Yale database, but quite close in the ORL database and the AR database. The above observations show that the sparsity assumption cannot necessarily guarantee improving the classification accuracy for the face recognition problem which is a typical small-sample-size problem. In addition, it is noted that both CRC\(_R\)_L and SRC outperform \( l_2 \)C, probably due to the fact that the introduced regularizers on the coding coefficient vectors in both CRC\(_R\)_L and SRC make their solutions more stable.

3. As seen from Table 6 and Fig. 1, the reconstruction residuals computed by 2DRR\(_2\) and 2DRR\(_1\) are lower than those by the other referred algorithms, and the corresponding recovered images by 2DRR\(_2\) and 2DRR\(_1\) have better reconstruction quality than those by those referred algorithms, probably because the proposed 2DRR is based on 2D image matrices and codes an image column by column in terms of the corresponding columns of the training images, which further confirms our speculation in Section 3.4.3.

5. Conclusions

In this paper, we propose a novel classification framework based on 2D image matrices, called two-dimensional relaxed representation (2DRR). In the 2DRR framework, the input matrix is coded

### Table 1
MVER(SDER) comparison with different values of \( p \) by 2DRR\(_1\) in the Yale database.

<table>
<thead>
<tr>
<th>( m )</th>
<th>MVER(SDER)</th>
<th>( p=2 )</th>
<th>( p=4 )</th>
<th>( p=6 )</th>
<th>( p=8 )</th>
<th>( p=10 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-Train</td>
<td>27.5(2.36)</td>
<td>27.3(2.62)</td>
<td>28.1(3.10)</td>
<td>28.8(2.28)</td>
<td>29.9(1.78)</td>
<td></td>
</tr>
<tr>
<td>4-Train</td>
<td>22.9(2.56)</td>
<td>22.6(1.91)</td>
<td>22.4(3.52)</td>
<td>23.3(2.96)</td>
<td>24.3(2.96)</td>
<td></td>
</tr>
<tr>
<td>5-Train</td>
<td>19.7(3.67)</td>
<td>18.7(2.07)</td>
<td>19.4(2.39)</td>
<td>21.1(3.01)</td>
<td>21.7(2.79)</td>
<td></td>
</tr>
</tbody>
</table>

### Table 2
MVER(SDER) comparison with different values of \( p \) by 2DRR\(_2\) in the Yale database.

<table>
<thead>
<tr>
<th>( m )</th>
<th>MVER(SDER)</th>
<th>( p=2 )</th>
<th>( p=4 )</th>
<th>( p=6 )</th>
<th>( p=8 )</th>
<th>( p=10 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-Train</td>
<td>27.5(2.79)</td>
<td>27.3(2.84)</td>
<td>26.7(2.36)</td>
<td>27.2(2.56)</td>
<td>27.4(2.95)</td>
<td></td>
</tr>
<tr>
<td>4-Train</td>
<td>22.2(2.60)</td>
<td>21.7(2.44)</td>
<td>21.4(3.34)</td>
<td>21.4(2.74)</td>
<td>21.6(1.56)</td>
<td></td>
</tr>
<tr>
<td>5-Train</td>
<td>18.5(2.36)</td>
<td>18.6(3.06)</td>
<td>18.1(3.19)</td>
<td>18.6(3.56)</td>
<td>18.3(3.45)</td>
<td></td>
</tr>
</tbody>
</table>

### Table 3
MVER(SDER) comparison on the Yale database.

<table>
<thead>
<tr>
<th>Methods</th>
<th>MVER(SDER)</th>
<th>3-Train</th>
<th>4-Train</th>
<th>5-Train</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRC</td>
<td>30.7(0.63)</td>
<td>27.1(5.41)</td>
<td>23.8(3.90)</td>
<td></td>
</tr>
<tr>
<td>2DRR(_1)</td>
<td>27.1(2.62)</td>
<td>22.6(1.91)</td>
<td>18.7(2.07)</td>
<td></td>
</tr>
<tr>
<td>( l_2 )C</td>
<td>36.4(2.82)</td>
<td>32.1(4.84)</td>
<td>26.9(6.20)</td>
<td></td>
</tr>
<tr>
<td>CRC(_R)_L</td>
<td>36.3(2.67)</td>
<td>31.6(4.40)</td>
<td>26.9(6.11)</td>
<td></td>
</tr>
<tr>
<td>2DRR(_2)</td>
<td>26.7(2.36)</td>
<td>21.4(3.34)</td>
<td>18.1(3.19)</td>
<td></td>
</tr>
</tbody>
</table>

### Table 4
MVER(SDER) comparison on the ORL database.

<table>
<thead>
<tr>
<th>Methods</th>
<th>MVER(SDER)</th>
<th>3-Train</th>
<th>4-Train</th>
<th>5-Train</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRC</td>
<td>14.2(0.93)</td>
<td>10.0(2.04)</td>
<td>6.6(1.88)</td>
<td></td>
</tr>
<tr>
<td>2DRR(_1)</td>
<td>11.3(1.58)</td>
<td>7.7(2.27)</td>
<td>4.2(0.84)</td>
<td></td>
</tr>
<tr>
<td>( l_2 )C</td>
<td>16.2(0.90)</td>
<td>13.8(3.08)</td>
<td>8.9(2.07)</td>
<td></td>
</tr>
<tr>
<td>CRC(_R)_L</td>
<td>14.6(1.21)</td>
<td>11.1(2.22)</td>
<td>6.9(0.27)</td>
<td></td>
</tr>
<tr>
<td>2DRR(_2)</td>
<td>13.4(1.46)</td>
<td>9.9(2.36)</td>
<td>5.2(1.22)</td>
<td></td>
</tr>
</tbody>
</table>

### Table 5
MVER(SDER) comparison on the AR database.

<table>
<thead>
<tr>
<th>Methods</th>
<th>MVER(SDER)</th>
<th>3-Train</th>
<th>4-Train</th>
<th>5-Train</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRC</td>
<td>0.2491(0.0563)</td>
<td>0.2230(0.0503)</td>
<td>0.2105(0.0514)</td>
<td></td>
</tr>
<tr>
<td>2DRR(_1)</td>
<td>0.2160(0.0456)</td>
<td>0.2080(0.0372)</td>
<td>0.1850(0.0336)</td>
<td></td>
</tr>
<tr>
<td>( l_2 )C</td>
<td>0.2486(0.0564)</td>
<td>0.2223(0.0505)</td>
<td>0.2094(0.0516)</td>
<td></td>
</tr>
<tr>
<td>CRC(_R)_L</td>
<td>0.2494(0.0566)</td>
<td>0.2229(0.0505)</td>
<td>0.2103(0.0517)</td>
<td></td>
</tr>
<tr>
<td>2DRR(_2)</td>
<td>0.2104(0.0410)</td>
<td>0.1998(0.0390)</td>
<td>0.1826(0.0381)</td>
<td></td>
</tr>
</tbody>
</table>

### Table 6
Reconstruction residual comparison in the Yale database.

<table>
<thead>
<tr>
<th>Methods</th>
<th>MVRR(SDRR)</th>
<th>3-Train</th>
<th>4-Train</th>
<th>5-Train</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRC</td>
<td>0.2491(0.0563)</td>
<td>0.2230(0.0503)</td>
<td>0.2105(0.0514)</td>
<td></td>
</tr>
<tr>
<td>2DRR(_1)</td>
<td>0.2160(0.0456)</td>
<td>0.2080(0.0372)</td>
<td>0.1850(0.0336)</td>
<td></td>
</tr>
<tr>
<td>( l_2 )C</td>
<td>0.2486(0.0564)</td>
<td>0.2223(0.0505)</td>
<td>0.2094(0.0516)</td>
<td></td>
</tr>
<tr>
<td>CRC(_R)_L</td>
<td>0.2494(0.0566)</td>
<td>0.2229(0.0505)</td>
<td>0.2103(0.0517)</td>
<td></td>
</tr>
<tr>
<td>2DRR(_2)</td>
<td>0.2104(0.0410)</td>
<td>0.1998(0.0390)</td>
<td>0.1826(0.0381)</td>
<td></td>
</tr>
</tbody>
</table>
column by column with respect to the corresponding columns of the elementary matrices, and simultaneously the coding coefficients corresponding to each column of the input matrix are constrained to be locally close so that the global linear relationship between the input matrix and these elementary matrices is preserved to some extent. Then two algorithms are derived from the 2DRR framework under the $l_2$ norm and the $l_1$ norm respectively.

The advantages of the proposed 2DRR are: (1) It can represent a matrix signal more accurately compared with these vector-representation-based classification algorithms. (2) 2DRR$\ell_2$ and 2DRR$\ell_1$ derived from the 2DRR framework have better performances for face recognition than these vector-representation-based classification algorithms.

Acknowledgment

This work was supported by National Natural Science Foundation of China under Grant nos. (60905020, 61173098).

References


