Information **2015**, *6*, 287-299; doi:10.3390/info6030287

OPEN ACCESS

information

ISSN 2078-2489

www.mdpi.com/journal/information

Article

Robust Sparse Representation for Incomplete and Noisy Data

Jiarong Shi *, Xiuyun Zheng and Wei Yang

School of Science, Xi'an University of Architecture and Technology, Xi'an 710055, China;
E-Mails: xyzhengzzf@sohu.com (X.Z.); yangweipyf@163.com (W.Y.)

* Author to whom correspondence should be addressed; E-Mail: shijiarong@xauat.edu.cn;
Tel./Fax: +86-29-8220-5670.

Academic Editor: Willy Susilo

Received: 12 June 2015 / Accepted: 16 June 2015 / Published: 24 June 2015

Abstract: Owing to the robustness of large sparse corruptions and the discrimination of class labels, sparse signal representation has been one of the most advanced techniques in the fields of pattern classification, computer vision, machine learning and so on. This paper investigates the problem of robust face classification when a test sample has missing values. Firstly, we propose a classification method based on the incomplete sparse representation. This representation is boiled down to an l_1 minimization problem and an alternating direction method of multipliers is employed to solve it. Then, we provide a convergent analysis and a model extension on incomplete sparse representation. Finally, we conduct experiments on two real-world face datasets and compare the proposed method with the nearest neighbor classifier and the sparse representation-based classification. The experimental results demonstrate that the proposed method has the superiority in classification accuracy, completion of the missing entries and recovery of noise.

Keyword: sparse representation; robust; face classification; alternating direction method of multipliers; incomplete; l_1 minimization

1. Introduction

As a parsimony method, the sparse signal representation means that we desire to represent a signal by the linear combination of a few basis elements in an over-complete dictionary. The emerging theory of sparse representation and compressed sensing [1,2] has made exciting breakthroughs and received a great deal of attention in the past decade. Nowadays, the sparse representation has already been a powerful technique for efficiently acquiring, compressing and reconstructing a signal. Besides, the sparse representation also has two powerful functions, that is, it is robust to large sparse corruptions and discriminative to class labels. These two distinguished functions promote its extensive and successful applications in areas such as pattern classification [3–5], computer vision [6] and machine learning [7].

It is worth mentioning that Wright *et al.* [3] proposed a novel method for robust face classification. They applied the idea of sparse representation to pattern classification and demonstrated that this unorthodox method can obtain significant improvements in classification accuracy over traditional methods. Subsequently, Yin *et al.* [8] extended the aforementioned classification method to the kernel version. Moreover, Huang *et al.* [9] and Qiao *et al.* [4] performed respectively face classification and signal classification by combining the discriminative methods with sparse representation. In [7], Cheng *et al.*, constructed a robust and datum-adaptive l_1-graph on the basis of sparse representation. Compared with k-nearest graph and ε-ball graph, the l_1-graph is more robust to large sparse noise and more discriminative to neighbors. Zhang *et al.* [10] presented a robust seminonnegative graph embedding framework, and Chen *et al.* [11] applied non-negative sparse coding to facial expression classification. Elhamifar *et al.* [12] proposed a framework of sparse subspace clustering, which harnessed the sparse representation to cluster data drawn from multiple low dimensional subspaces.

In the community of pattern classification and machine learning, we mainly restrict our attention to the situation that all samples do not have any missing entries. However, the datasets with missing values are ubiquitous in many practical applications such as image in-painting, video encoding and collaborative filtering. A commonly-used modeling assumption in data analysis is that the investigated dataset is (approximately) low-rank. Based on this assumption, Candès *et al.* [13] proposed a technique of matrix completion via convex optimization and showed that most low-rank matrices can be exactly completed under certain conditions. If we make a further clustering analysis on these datasets, Shi *et al.* [14] proposed the method of incomplete low-rank representation, which is validated to be very robust to missing values.

For the task of pattern classification, we usually stipulate that all samples from the same class lie in a low dimensional subspace. If the training samples have missing entries, the matrix completion or the incomplete low-rank representation can be employed to complete or recover all missing values. This paper considers the pattern classification problem that the test samples have missing values while the training samples are complete. To address it, we propose a method of incomplete sparse representation. This method treats each incomplete test sample as the linear combination of all training samples and searches the sparest representation.

The remainder of this paper is organized as follows. Section 2 reviews the classification problem based on sparse representation. In Section 3, we propose a model of incomplete sparse representation and develop an alternating direction method of multipliers (ADMM) [15] to solve it. Convergence analysis and model extension are made in Section 4. In Section 5, we carry out experiments on two well-known face datasets and validate the superiority of the proposed method by comparing with other techniques. The last section draws the conclusions.

2. Sparse Representation for Classification

A fundamental problem in pattern classification is how to determine the class label of a test sample according to labeled training samples from distinct classes. Given a training set collected by C classes, we express all samples from the i-th class as $\left\{\mathbf{a}_{ij} \in \mathbb{R}^d\right\}_{j=1}^{N_i}$, where d is the dimensionality of each sample, and N_i is the sample number of the i-th class, $i = 1, 2, ..., C$. Denote $N = \sum_{i=1}^{C} N_i$ and $\mathbf{A}_i = \left(\mathbf{a}_{i1}, \mathbf{a}_{i2}, \ldots, \mathbf{a}_{iN_i}\right)$. Thus, the entire training samples can be concatenated into a $d \times N$ matrix $\mathbf{A} = \left(\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_C\right)$.

We assume that the samples from the same class lie in a low-dimensional linear subspace and there are sufficient training samples for each class. Given a new test sample $\mathbf{y} = (y_1, y_2, ..., y_d)^T \in \mathbb{R}^d$ with label i, we can express it as the linear combination of the training samples from the i-th class:

$$\mathbf{y} = w_{i1}\mathbf{a}_{i1} + w_{i2}\mathbf{a}_{i2} + \ldots + w_{iN_i}\mathbf{a}_{iN_i} \tag{1}$$

for some real scalars w_{ij}, $j = 1, 2, ..., N_i$. Set $\mathbf{w} = (0, ..., 0, w_{i1}, w_{i2}, ..., w_{iN_i}, 0, ..., 0)^T \in \mathbb{R}^N$. Then the linear representation of $\mathbf{y}$ can be re-expressed in terms of N training samples as

$$\mathbf{y} = \mathbf{Aw}. \tag{2}$$

In view of the assumption that the training samples are enough, the coefficient vector $\mathbf{w}$ satisfying Equation (2) is not unique. Moreover, the vector $\mathbf{w}$ is also sparse if N_i / N is very small.

If the class label of $\mathbf{y}$ is unknown, we still desire to obtain its label on the basis of the linear representation coefficients $\mathbf{w}$. The sparse representation is essentially discriminative to classification and robust to large sparse noise or outliers. Considering these advantages, we will construct a sparse representation model to perform classification. For the given dictionary matrix $\mathbf{A}$, the sparse representation of $\mathbf{y}$ can be reached by solving an l_1-minimization problem

$$\min_{\mathbf{w}} \|\mathbf{w}\|_1, \quad \text{s.t. } \mathbf{y} = \mathbf{Aw} \tag{3}$$

or its stable version

$$\min_{\mathbf{w}} \|\mathbf{w}\|_1, \quad \text{s.t. } \|\mathbf{y} - \mathbf{Aw}\|_2 \le \delta \tag{4}$$

where the error bound $\delta > 0$, $\|\bullet\|_1$ and $\|\bullet\|_2$ are the l_1-norm and the l_2-norm of vectors respectively. Sparse representation is a global method and it has superiority in determining the class label over other local methods such as nearest neighbor (NN) and nearest subspace (NS) [3].

Denote the optimal solution of Problem (3) or (4) by $\hat{\mathbf{w}}$. Sparse representation-based classification (SRC) [3] utilizes $\hat{\mathbf{w}}$ to judge which class $\mathbf{y}$ belongs to. The following will present the detailed implementation process. We first introduce C characteristic functions $\delta_i : \mathbb{R}^N \to \mathbb{R}^N$ as below

$$\left(\delta_i(\mathbf{x})\right)_j = \begin{cases} x_j, & \text{if } \sum_{k=0}^{i-1} N_k < j \le \sum_{k=0}^{i} N_k \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

for arbitrary $\mathbf{x} = (x_1, x_2, ..., x_N)^T \in \mathbb{R}^N$, where $N_0 = 0$, $i = 1, 2, ..., C$. Then we computed C residuals $r_i(\mathbf{y}) = \|\mathbf{y} - \mathbf{A}\delta_i(\hat{\mathbf{w}})\|_2$, $i = 1, 2, ..., C$. Finally, the class of $\mathbf{y}$ is labeled as $\arg\min_i r_i(\mathbf{y})$.

3. Incomplete Sparse Representation for Classification

As the second orders generalization of the compressed sensing and sparse representation theory, the low-rank matrix completion is a technique to complete all missing or unknown entries of a matrix by means of the low-rank structure. If all training samples are complete and the test sample has missing entries, we can not effectively recover the missing values through the use of the low rank property. In other words, the available matrix completion methods become invalid. To solve this problem, we propose a method of incomplete sparse representation for classification. The proposed method not only completes effectively the missing entries but also obtains better classification performance.

3.1. Model of Incomplete Sparse Representation

Considering the existence of noise, we decompose a given test sample $\mathbf{y}$ into the sum of two terms: $\mathbf{y} = \mathbf{A}\mathbf{w} + \mathbf{e}$, where $\mathbf{e} \in \mathbb{R}^d$ is the noise vector. Let $\Omega \subset \{1, 2, ..., d\}$ be an index set, then y_k is missing if and only if $k \notin \Omega$. For the convenience of description, we define an orthogonal projector operator $P_\Omega(\bullet): \mathbb{R}^d \to \mathbb{R}^d$ as follows

$$\left(P_\Omega(\mathbf{y})\right)_k = \begin{cases} y_k, & \text{if } k \in \Omega, \\ 0, & \text{otherwise}. \end{cases} \tag{6}$$

Thus, the known entries of $\mathbf{y}$ can be written as $\mathbf{y}_0 = P_\Omega(\mathbf{y})$.

For the incomplete test sample $\mathbf{y}_0$, we hope to complete all missing entries and obtain the sparsest linear representation on the basis of $\mathbf{A}$ and Ω. To this end, we construct an l_1-minimization problem

$$\begin{aligned} &\min_{\mathbf{w},\mathbf{e},\mathbf{y}} \ \|\mathbf{w}\|_1 + \tau \|\mathbf{e}\|_2^2 \\ &\text{s.t. } \mathbf{y} = \mathbf{A}\mathbf{w} + \mathbf{e}, P_\Omega(\mathbf{y}) = \mathbf{y}_0 \end{aligned} \tag{7}$$

where the tradeoff factor $\tau > 0$. As a matter of fact, the objective function in the above problem can be replaced by $\|\mathbf{w}\|_1 + \tau \|P_\Omega(\mathbf{e})\|_2^2$. This conclusion means that it is impossible to recover the noise corresponding to the missing positions. Problem (7) is a convex and non-smooth minimization with equality constraints. We will employ the alternating direction method of multipliers (ADMM) to solve this problem.

3.2. Generic Formulation of ADMM

The ADMM [15] is a simple and easily implemented optimization method proposed in 1970s. It is well suited to distributed convex optimization, and, in particular, to large scale optimization problems with multiple non-smooth terms in the objective functions. Hence, the method has received a lot of attention in recent years.

Generally, ADMM solves the constrained optimization problem taking the following generic form:

$$\min_{\mathbf{x}\in\mathbb{R}^m, \mathbf{y}\in\mathbb{R}^n} f(\mathbf{x}) + g(\mathbf{y}), \text{ s.t. } \mathbf{B}\mathbf{x} + \mathbf{C}\mathbf{y} = \mathbf{d} \tag{8}$$

where $f: \mathbb{R}^m \to \mathbb{R}$, $g: \mathbb{R}^n \to \mathbb{R}$, and they are proper and convex. The augmented Lagrange function of Problem (8) is defined as follows

$$L_\mu(\mathbf{x},\mathbf{y},\lambda)=f(\mathbf{x})+g(\mathbf{y})+\langle\lambda,\mathbf{Bx}-\mathbf{Cy}-\mathbf{d}\rangle+\mu\|\mathbf{Bx}-\mathbf{Cy}-\mathbf{d}\|_2^2/2 \tag{9}$$

where μ is a positive scalar, λ is the Lagrange multiplier vector and $\langle\cdot,\cdot\rangle$ is the inner operator of vectors.

ADMM updates alternatively each block of variables. The iterative formulations of blocks of variables are outlined as follows

$$\begin{cases}\mathbf{x}:=\arg\min\limits_{x} L_\mu(\mathbf{x},\mathbf{y},\lambda)\\ \mathbf{y}:=\arg\min\limits_{\mathbf{y}} L_\mu(\mathbf{x},\mathbf{y},\lambda)\\ \lambda:=\arg\max\limits_{\lambda} L_\mu(\mathbf{x},\mathbf{y},\lambda).\end{cases} \tag{10}$$

Moreover, the values of μ will be increased during the procedure of iterations.

3.3. Algorithm for Incomplete Sparse Representation

We adopt the method of ADMM with multiple blocks of variables to solve the problem of incomplete sparse representation. By introducing two auxiliary variables $\mathbf{z}\in\mathbb{R}^d$ and $\mathbf{u}\in\mathbb{R}^N$, we reformulate Problem (7) as follows:

$$\begin{aligned}&\min_{\mathbf{w},\mathbf{e},\mathbf{y},\mathbf{z},\mathbf{u}} \|\mathbf{w}\|_1+\tau\|\mathbf{e}\|_2^2,\\ &\text{s.t.}\ \ \mathbf{z}=\mathbf{Au}+\mathbf{e}, \mathbf{w}=\mathbf{u}, \mathbf{z}=\mathbf{y}, P_\Omega(\mathbf{y})=\mathbf{y}_0.\end{aligned} \tag{11}$$

Without considering the constraint $P_\Omega(\mathbf{y})=\mathbf{y}_0$, the augmented Lagrange function of the above optimization problem is constructed as

$$\begin{aligned}&L_\rho(\mathbf{w},\mathbf{e},\mathbf{y},\mathbf{z},\mathbf{u},\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2,\boldsymbol{\lambda}_3)\\ &=\|\mathbf{w}\|_1+\tau\|\mathbf{e}\|_2^2+\frac{\rho}{2}\left(\|\mathbf{z}-\mathbf{Au}-\mathbf{e}\|_2^2+\|\mathbf{w}-\mathbf{u}\|_2^2+\|\mathbf{z}-\mathbf{y}\|_2^2\right)\\ &\quad+\langle\boldsymbol{\lambda}_1,\mathbf{z}-\mathbf{Au}-\mathbf{e}\rangle+\langle\boldsymbol{\lambda}_2,\mathbf{w}-\mathbf{u}\rangle+\langle\boldsymbol{\lambda}_3,\mathbf{z}-\mathbf{y}\rangle\end{aligned} \tag{12}$$

where the penalty coefficient $\rho>0$, $\boldsymbol{\lambda}_1\in\mathbb{R}^d$, $\boldsymbol{\lambda}_2\in\mathbb{R}^N$ and $\boldsymbol{\lambda}_3\in\mathbb{R}^d$ are three Lagrange multipliers vectors. Let $\boldsymbol{\lambda}=\{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2,\boldsymbol{\lambda}_3\}$, then Equation (12) is equivalent to

$$\begin{aligned}&L_\rho(\mathbf{w},\mathbf{e},\mathbf{y},\mathbf{z},\mathbf{u},\boldsymbol{\lambda})\\ &=\|\mathbf{w}\|_1+\tau\|\mathbf{e}\|_2^2+\frac{\rho}{2}\left(\|\mathbf{z}-\mathbf{Au}-\mathbf{e}+\boldsymbol{\lambda}_1/\rho\|_2^2+\|\mathbf{w}-\mathbf{u}+\boldsymbol{\lambda}_2/\rho\|_2^2+\|\mathbf{z}-\mathbf{y}+\boldsymbol{\lambda}_3/\rho\|_2^2\right).\end{aligned} \tag{13}$$

ADMM updates alternatively each block of variables by minimizing or maximizing L_ρ. We subsequently give the detailed iterative procedure for Problem (11).

Computing $\mathbf{w}$. When $\mathbf{w}$ is unknown and other blocks of variables are fixed, the calculation formulation of $\mathbf{w}$ is as follows:

$$\begin{aligned}\mathbf{w}&:=\arg\min_{\mathbf{w}} L_\rho\\ &=\arg\min_{\mathbf{w}}\frac{1}{\rho}\|\mathbf{w}\|_1+\frac{1}{2}\|\mathbf{w}-(\mathbf{u}-\boldsymbol{\lambda}_2/\rho)\|_2^2\\ &=S_{1/\rho}(\mathbf{u}-\boldsymbol{\lambda}_2/\rho)\end{aligned} \tag{14}$$

where $S_{1/\rho}(\bullet):\mathbb{R}^N\to\mathbb{R}^N$ is the absolute shrinkage operator [16] defined by

$$\left(S_{1/\rho}(\mathbf{x})\right)_j=\max\left(\left|x_j\right|-1/\rho,0\right)\operatorname{sgn}(x_j) \tag{15}$$

for arbitrary $\mathbf{x}\in\mathbb{R}^N$.

Computing $\mathbf{e}$. If $\mathbf{e}$ is unknown and other variables are given, $\mathbf{e}$ is updated by minimizing L_ρ:

$$\begin{aligned}\mathbf{e}&:=\arg\min_{\mathbf{e}} L_\rho\\&=\arg\min_{\mathbf{e}}\tau\|\mathbf{e}\|_2^2+\frac{\rho}{2}\left\|\left(\mathbf{z}-\mathbf{Au}+\boldsymbol{\lambda}_1/\rho\right)-\mathbf{e}\right\|_2^2\\&=\frac{\rho}{2\tau+\rho}\left(\mathbf{z}-\mathbf{Au}+\boldsymbol{\lambda}_1/\rho\right).\end{aligned} \tag{16}$$

Computing $\mathbf{u}$. The update formulation of $\mathbf{u}$ is calculated as follows:

$$\mathbf{u}:=\arg\min_{\mathbf{u}} L_\rho=\arg\min_{\mathbf{u}} f(\mathbf{u}) \tag{17}$$

where $f(\mathbf{u})=\left\|\left(\mathbf{z}-\mathbf{e}+\boldsymbol{\lambda}_1/\rho\right)-\mathbf{Au}\right\|_2^2+\left\|\left(\mathbf{w}+\boldsymbol{\lambda}_2/\rho\right)-\mathbf{u}\right\|_2^2$. By setting the derivative of $f(\mathbf{u})$ to zeros, we have

$$\mathbf{A}^T\mathbf{Au}-\mathbf{A}^T\left(\mathbf{z}-\mathbf{e}+\boldsymbol{\lambda}_1/\rho\right)+\mathbf{u}-\left(\mathbf{w}+\boldsymbol{\lambda}_2/\rho\right)=0 \tag{18}$$

or, equivalently,

$$\mathbf{u}=\left(\mathbf{A}^T\mathbf{A}+\mathbf{I}_N\right)^{-1}\left(\mathbf{A}^T\left(\mathbf{z}-\mathbf{e}+\boldsymbol{\lambda}_1/\rho\right)+\left(\mathbf{w}+\boldsymbol{\lambda}_2/\rho\right)\right) \tag{19}$$

where $\mathbf{I}_N$ is an *N*-order identity matrix.

Computing $\mathbf{z}$. Fix $\mathbf{w},\mathbf{e},\mathbf{y},\mathbf{u}$ and $\boldsymbol{\lambda}$, and minimize L_ρ with respect to $\mathbf{z}$:

$$\begin{aligned}\mathbf{z}&:=\arg\min_{\mathbf{z}} L_\rho\\&=\arg\min_{z}\left\|\mathbf{z}-\left(\mathbf{Au}+\mathbf{e}-\boldsymbol{\lambda}_1/\rho\right)\right\|_2^2+\left\|\mathbf{z}-\left(\mathbf{y}-\boldsymbol{\lambda}_3/\rho\right)\right\|_2^2\\&=\frac{1}{2}\left(\mathbf{Au}+\mathbf{e}+\mathbf{y}-\left(\boldsymbol{\lambda}_1+\boldsymbol{\lambda}_3\right)/\rho\right).\end{aligned} \tag{20}$$

Computing $\mathbf{y}$. Given $\mathbf{w},\mathbf{e},\mathbf{z},\mathbf{u}$ and $\boldsymbol{\lambda}$, we calculate $\mathbf{y}$ as follows

$$\begin{aligned}\mathbf{y}&:=\arg\min_{\mathbf{y}} L_\rho\\&=\arg\min_{\mathbf{y}}\left\|\left(\mathbf{z}+\boldsymbol{\lambda}_3/\rho\right)-\mathbf{y}\right\|_2^2\\&=\mathbf{z}+\boldsymbol{\lambda}_3/\rho.\end{aligned} \tag{21}$$

Considering the constraint $P_\Omega(\mathbf{y})=\mathbf{y}_0$, we further obtain the iterative formulation of $\mathbf{y}$:

$$\mathbf{y}:=\mathbf{y}_0+P_{\bar{\Omega}}\left(\mathbf{z}+\boldsymbol{\lambda}_3/\rho\right) \tag{22}$$

where $\bar{\Omega}$ is the complementary set of Ω.

Computing $\boldsymbol{\lambda}$. Given $\mathbf{w},\mathbf{e},\mathbf{y},\mathbf{z},\mathbf{u}$, we update $\boldsymbol{\lambda}$ as follows

$$\begin{cases} \boldsymbol{\lambda}_1 := \boldsymbol{\lambda}_1 + \rho(\mathbf{z} - \mathbf{Au} - \mathbf{e}) \\ \boldsymbol{\lambda}_2 := \boldsymbol{\lambda}_2 + \rho(\mathbf{w} - \mathbf{u}) \\ \boldsymbol{\lambda}_3 := \boldsymbol{\lambda}_3 + \rho(\mathbf{z} - \mathbf{y}) \end{cases} \tag{23}$$

The whole iterative process for solving Problem (11) is outlined in Algorithm 1. In the initialization step, the blocks of variables can be chosen as follows: $\mathbf{e} = 0$, $\mathbf{y} = \mathbf{y}_0$, $\mathbf{z} = 0$, $\mathbf{u} = 0$, $\boldsymbol{\lambda}_1 = 0$, $\boldsymbol{\lambda}_2 = 0$, $\boldsymbol{\lambda}_3 = 0$. We set the stopping condition of Algorithm 1 to be

$$\max\left(\|\mathbf{z} - \mathbf{Au} - \mathbf{e}\|_2, \|\mathbf{w} - \mathbf{u}\|_2, \|\mathbf{z} - \mathbf{y}\|_2\right) < \varepsilon \tag{24}$$

where ε is a sufficiently small positive number. The inverse matrix $\left(\mathbf{A}^T\mathbf{A} + \mathbf{I}\right)^{-1}$ is computed only once and the corresponding computational complexity is $O(N^2 d + N^3)$. In addition, the complexity of Algorithm 1 is also $O(N^2 d + N^3)$ for each outer loop.

Algorithm 1. Solving Problem (11) via ADMM.

```
Input: the dictionary matrix $\mathbf{A}$ constructed by all training samples, an incomplete test sample
    $\mathbf{y}_0$ and the sampling index set $\Omega$.
Output: $\mathbf{y}$, $\mathbf{w}$ and $\mathbf{e}$.
Initialize: $\mathbf{e}, \mathbf{y}, \mathbf{z}, \mathbf{u}, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \boldsymbol{\lambda}_3$, $\rho, \bar{\rho}, \tau$, $\mu > 1$.
While not converged do
    Update $\mathbf{w}$ according to (14).
    Update $\mathbf{e}$ according to (16).
    Update $\mathbf{u}$ according to (19).
    Update $\mathbf{z}$ according to (20).
    Update $\mathbf{y}$ according to (22).
    Update $\boldsymbol{\lambda}$ according to (23).
    Update $\rho$ as $\min(\mu\rho, \bar{\rho})$.
End while
```

Let $\hat{\mathbf{y}}$, $\hat{\mathbf{w}}$ and $\hat{\mathbf{e}}$ be the output variables of Algorithm 1. The vector $\hat{\mathbf{y}}$ denotes the completed sample of $\mathbf{y}_0$ and $\hat{\mathbf{w}}$ indicates the sparse representation of $\hat{\mathbf{y}}$ over the basis matrix $\boldsymbol{A}$. In view of the discriminative performance of $\hat{\mathbf{w}}$, this sparse vector can be employed to obtain the class label of $\mathbf{y}_0$. More specially, we first compute C residuals $r_i(\mathbf{y}_0) = \|\mathbf{y}_0 - P_\Omega(\mathbf{A}\delta_i(\hat{\mathbf{w}}))\|_2$ and then label the class of $\mathbf{y}_0$ to be $\arg\min_i r_i(\mathbf{y}_0)$. The above method is called incomplete SRC (ISRC), a variant of SRC.

4. Convergence Analysis and Model Extension

Although the minimization Problem (11) is convex and continuous, it is still difficult to straightly prove the convergence of Algorithm 1. The main reason for this difficulty is that the number of blocks of variables is more than two. If there is no missing value, we can design an exact ADMM for solving Problem (11). The following theorem shows the convergence of the modified method.

Theorem 1. If $\Omega = \{1,2,...,d\}$ and $L_0(\mathbf{w},\mathbf{e},\mathbf{y}_0,\mathbf{z},\mathbf{u},\boldsymbol{\lambda})$ has a saddle point, then the iterative formulations on the basis of an exact ADMM

$$\begin{cases} (\mathbf{w}^{k+1},\mathbf{z}^{k+1}) := \arg\min\limits_{\mathbf{z},\mathbf{w}} L_\rho(\mathbf{w},\mathbf{e}^k,\mathbf{y}_0,\mathbf{z},\mathbf{u}^k,\boldsymbol{\lambda}^k) \\ (\mathbf{u}^{k+1},\mathbf{e}^{k+1}) := \arg\min\limits_{\mathbf{u},e} L_\rho(\mathbf{w}^{k+1},\mathbf{e},\mathbf{y}_0,\mathbf{z}^{k+1},\mathbf{u},\boldsymbol{\lambda}^k) \\ \boldsymbol{\lambda}_1^{k+1} := \boldsymbol{\lambda}_1^k + \rho\left(\mathbf{z}^{k+1} - \mathbf{A}\mathbf{u}^{k+1} - \mathbf{e}^{k+1}\right) \\ \boldsymbol{\lambda}_2^{k+1} := \boldsymbol{\lambda}_2^k + \rho\left(\mathbf{w}^{k+1} - \mathbf{u}^{k+1}\right) \\ \boldsymbol{\lambda}_3^{k+1} := \boldsymbol{\lambda}_3^k + \rho\left(\mathbf{z}^{k+1} - \mathbf{y}_0\right) \end{cases} \tag{25}$$

satisfy that $L_\rho(\mathbf{w},\mathbf{e},\mathbf{y}_0,\mathbf{z},\mathbf{u},\boldsymbol{\lambda})$ converges to the optimal value.

Proof. The objective function of Problem (11) can be rewritten as $f(\mathbf{z},\mathbf{w}) + g(\mathbf{u},\mathbf{e})$, where $f(\mathbf{z},\mathbf{w}) = \|\mathbf{w}\|_1$ and $g(\mathbf{u},\mathbf{e}) = \tau\|\mathbf{e}\|_2^2$. It is obvious that $f(\mathbf{z},\mathbf{w})$ and $g(\mathbf{u},\mathbf{e})$ are two closed, proper and convex functions.

Since $\Omega = \{1,2,...,d\}$, we have $\mathbf{y} = \mathbf{y}_0$, which means it is not necessary to consider the update of $\mathbf{y}$. Under this circumstance, the constraints in Problem (11) are equivalent to

$$\begin{pmatrix} \mathbf{I}_d & \mathbf{O}_{d\times N} \\ \mathbf{I}_d & \mathbf{O}_{d\times N} \\ \mathbf{O}_{N\times d} & \mathbf{I}_N \end{pmatrix} \begin{pmatrix} \mathbf{z} \\ \mathbf{w} \end{pmatrix} - \begin{pmatrix} \mathbf{A} & \mathbf{I}_d \\ \mathbf{O}_{d\times N} & \mathbf{O}_{d\times d} \\ \mathbf{I}_N & \mathbf{O}_{N\times d} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{e} \end{pmatrix} = \begin{pmatrix} 0 \\ \mathbf{y}_0 \\ 0 \end{pmatrix} \tag{26}$$

where $\mathbf{O}_{d\times N}$ is a zero matrix with size of $d\times N$.

In consideration of the characteristics of the objective function $L_\rho(\mathbf{w},\mathbf{e},\mathbf{y}_0,\mathbf{z},\mathbf{u},\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2,\boldsymbol{\lambda}_3)$ and the constraints (26), we have the following results for the iterative formulations (25): $\lim\limits_{k\to\infty}(\mathbf{z}^k - \mathbf{A}\mathbf{u}^k - \mathbf{e}^k) = 0$, $\lim\limits_{k\to\infty}(\mathbf{w}^k - \mathbf{u}^k) = 0$, $\lim\limits_{k\to\infty}(\mathbf{z}^k - \mathbf{y}_0) = 0$ and $L_\rho(\mathbf{w},\mathbf{e},\mathbf{y}_0,\mathbf{z},\mathbf{u},\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2,\boldsymbol{\lambda}_3)$ converges to the optimal value [15]. This completes the proof. □

For the aforementioned ISRC, we only consider one incomplete test sample. In the following, we will extent it to the case of a batch of test samples with missing values. Given a set of *m* incomplete test samples $\left\{\mathbf{y}_{0i} \in \mathbb{R}^d\right\}_{i=1}^m$, we can construct a matrix $\mathbf{Y}_0 = \left(\mathbf{y}_{01},\mathbf{y}_{02},...,\mathbf{y}_{0m}\right) \in \mathbb{R}^{d\times m}$ and a two-dimensional index set $\Omega' \subset \{1,2,...,d\}\times\{1,2,...,m\}$, where Ω' indicates the positions of missing values for $\mathbf{Y}_0$.

We establish the batch learning model of incomplete sparse representation:

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{E},\mathbf{Y},\mathbf{Z},\mathbf{U}} \|\mathbf{W}\|_1 + \tau\|\mathbf{E}\|_F^2, \\ &\text{s.t. } \mathbf{Z} = \mathbf{A}\mathbf{U} + \mathbf{E}, \mathbf{W} = \mathbf{U}, \mathbf{Z} = \mathbf{Y}, P_{\Omega'}(\mathbf{Y}) = \mathbf{Y}_0 \end{aligned} \tag{27}$$

where $\mathbf{W} \in \mathbb{R}^{N\times m}, \mathbf{U} \in \mathbb{R}^{N\times m}, \mathbf{E} \in \mathbb{R}^{d\times m}, \mathbf{Y} \in \mathbb{R}^{d\times m}, \mathbf{Z} \in \mathbb{R}^{d\times m}$, $P_{\Omega'}(\bullet): \mathbb{R}^{d\times m} \to \mathbb{R}^{d\times m}$ is a two-dimensional generalization of $P_\Omega(\bullet)$, $\|\bullet\|_1$ and $\|\bullet\|_F$ are the component-wise *l*1-norm and the Frobenius norm of matrices respectively. Without considering the constraints $P_{\Omega'}(\mathbf{Y}) = \mathbf{Y}_0$, the augmented Lagrange function for Problem (27) is

$$L_\rho(\mathbf{W},\mathbf{E},\mathbf{Y},\mathbf{Z},\mathbf{U},\Lambda)$$
$$= \|\mathbf{W}\|_1 + \tau\|\mathbf{E}\|_F^2 + \frac{\rho}{2}\left(\|\mathbf{Z}-\mathbf{AU}-\mathbf{E}+\Lambda_1/\rho\|_F^2 + \|\mathbf{W}-\mathbf{U}+\Lambda_2/\rho\|_F^2 + \|\mathbf{Z}-\mathbf{Y}+\Lambda_3/\rho\|_F^2\right) \quad (28)$$

where $\rho > 0$, $\Lambda_1 \in \mathbb{R}^{d\times m}, \Lambda_2 \in \mathbb{R}^{N\times m}, \Lambda_3 \in \mathbb{R}^{d\times m}$ and $\Lambda = \{\Lambda_1, \Lambda_2, \Lambda_3\}$. When solving Problem (27), we adopt the similar iterative procedure with Problem (11) by minimizing or maximizing alternatively $L_\rho(\mathbf{W},\mathbf{E},\mathbf{Y},\mathbf{Z},\mathbf{U},\Lambda)$.

5. Experiments

This section demonstrates the effectiveness and the efficiency of ISRC by conducting experiments on Olivetti Research Laboratory (ORL) and Yale face datasets. We compare the results of the proposed method with that of NN and SRC.

5.1. Datasets Description and Experimental Setting

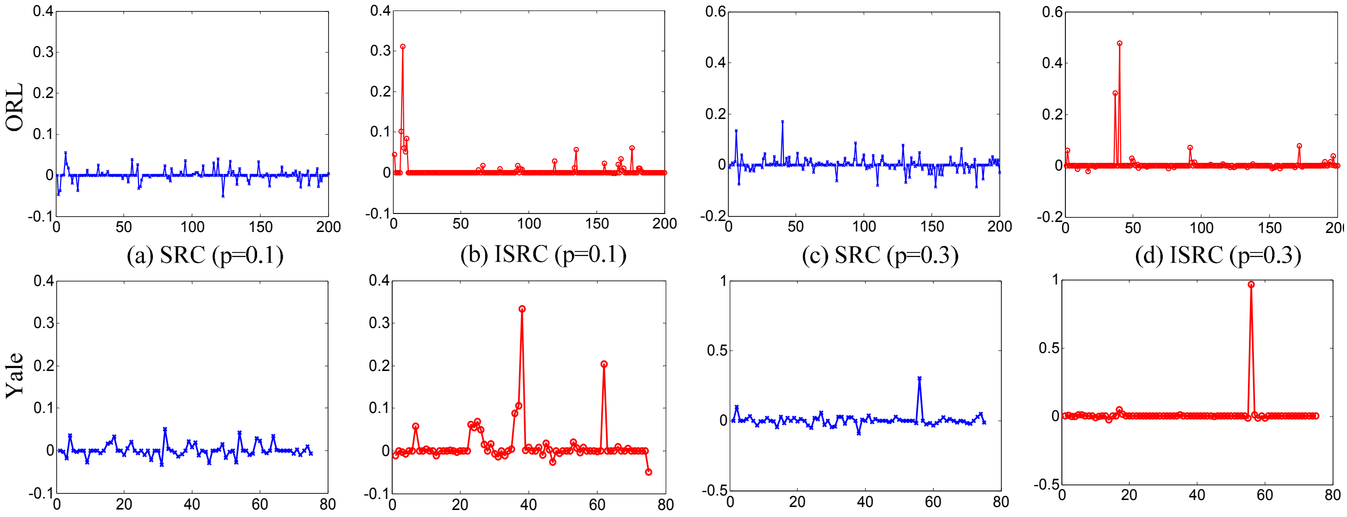
ORL dataset contains 10 different face images of each of 40 individuals [17]. These 400 images were captured at different time with different illuminations and varying facial details. The Yale face dataset consists of 165 images from 15 persons and there are 11 images for each person [18]. The images were taken with different illuminations, varying facial expressions and details. All images in both datasets are in grayscale and resized to be 64 × 64 for computational convenience. Hence, the dimensionality of each sample is $d = 4096$. Moreover, each sample is normalized to a unit vector in the sense of the l_2-norm due to the existence of variable illumination conditions and poses.

In ORL dataset, five images per person are randomly selected for training and the remaining five images for testing. In Yale dataset, we randomly choose six images per person as the training samples and the other images as the testing samples. For any sample $\mathbf{y}$ from the testing set, we generate randomly an index set Ω according to the Bernoulli distribution, *i.e.*, the probability of $i \in \Omega$ is stipulated as p for arbitrary $i \in \{1,2,...,d\}$, where $p \in (0,1]$. The probability p is also named the sampling probability and $p = 1$ means that no entry is missing. Thus, an incomplete sample of $\mathbf{y}$ is expressed as $\mathbf{y}_0 = P_\Omega(\mathbf{y})$. The generating manner of Ω indicates that the number of missing entries is approximately pd.

In Algorithm 1, the parameters are set as $\tau = 10^3, \rho = 10^{-8}, \bar{\rho} = 10^{10}, \mu = 1.1, \varepsilon = 10^{-8}$. For each dataset, we consider different values of p. For fixed p, the experiments are repeated 10 times and the average classification accuracies are reported. When carrying out NN, we compute the distance between $\mathbf{y}_0$ and $\mathbf{a}_{ij}$ as follows: $\|\mathbf{y}_0 - P_\Omega(\mathbf{a}_{ij})\|_2$. In addition, all missing values are replaced with zeros in implementing SRC.

5.2. Experimental Analysis

We first compare the sparsity of the coefficient vectors obtained by SRC and ISRC respectively. Two different sampling probabilities are considered, that is, p = 0.1 and p = 0.3. The comparison results are partially shown in Figure 1. From this figure, we can see that each linear representation vector of ISRC has only a few relatively large components in the sense of absolute values and other values are close to zero. Compared with ISRC, SRC has worse sparsity performance due to the fact that its amplitude is relatively small. These observations show that ISRC has superiority over SRC in obtaining sparse representations.

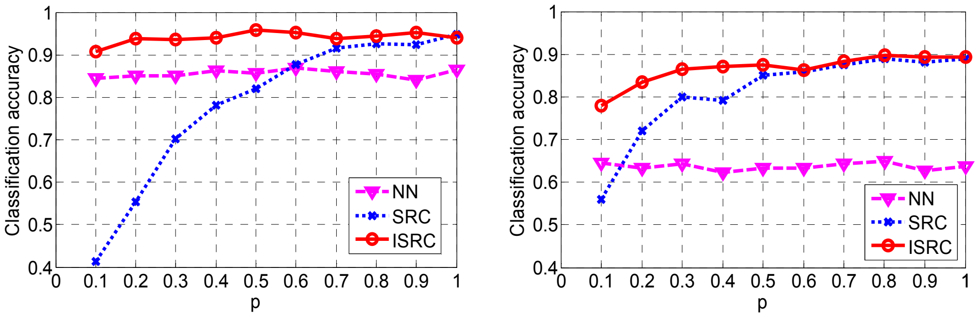


(a) SRC (p=0.1) (b) ISRC (p=0.1) (c) SRC (p=0.3) (d) ISRC (p=0.3)

(e) SRC (p=0.1) (f) ISRC (p=0.1) (g) SRC (p=0.3) (h) ISRC (p=0.3)

Figure 1. Sparsity comparisons of the linear representations between SRC and ISRC.

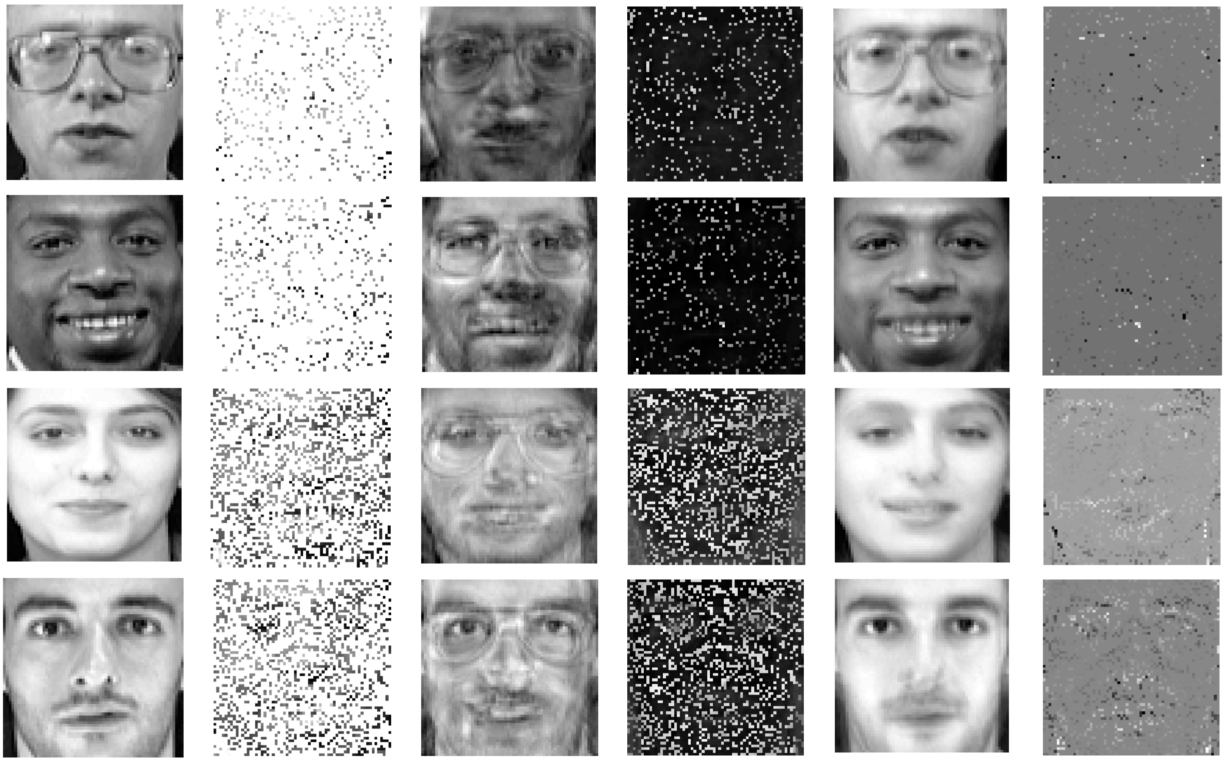
Then, we compare the classification performance of ISRC with that of SRC and NN on the two given datasets. To this end, we vary the values of *p* from 0.1 to 1 with an interval of 0.1. When *p* = 1, ISRC becomes SRC. Figure 2 shows the comparison results of classification accuracy, where (a) and (b) represent the results of ORL and Yale respectively. It can be seen from this figure that ISRC achieves the best classification accuracy compared with SRC and NN, and it is relatively stable for different values of *p*. SRC is very sensitive to the choice of *p* and its classification accuracy degenerates steeply with the decreasing of *p*. In addition, NN has lower classification accuracy although it is stable. To sum it up, ISRC is the most robust method and has the best classification performance.



(a) ORL (b) Yale

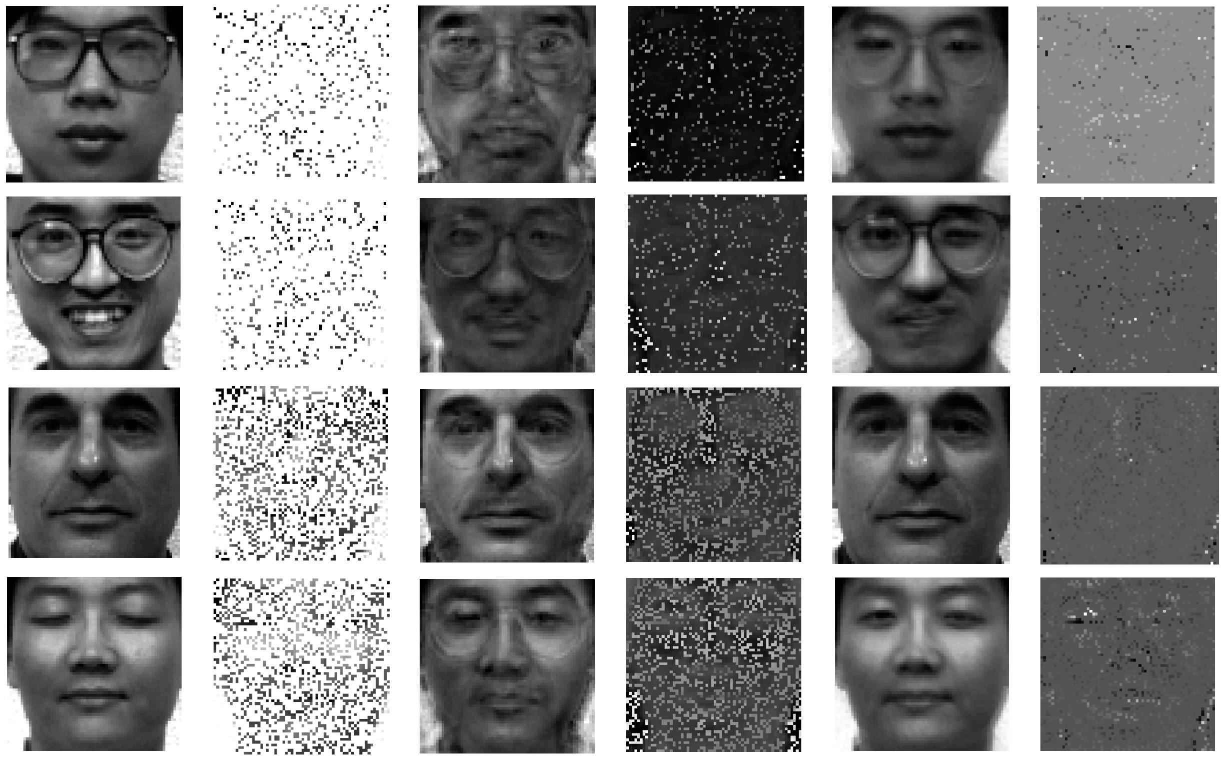
Figure 2. Classification performance comparisons among NN, SRC and ISRC.

For a test sample with missing values, both SRC and ISRC can recover all missing entries and noise to some extent. Finally, we compare their performance in completing missing entries and recovering the sparse noise. Here, we only consider two sampling probabilities, *i.e.*, *p* = 0.1 and *p* = 0.3. For these two given probabilities, we compare the completed images and the recovered noise images obtained by SRC and ISRC respectively, as shown partially in Figures 3 and 4.



(a) Original images (b) Incomplete images (c) Images completed by SRC (d) Noises recovered by SRC (e) Images completed by ISRC (f) Noises recovered by ISRC

Figure 3. Completion and recovery performance comparisons on ORL.



(a) Original images (b) Incomplete images (c) Images completed by SRC (d) Noises recovered by SRC (e) Images completed by ISRC (f) Noises recovered by ISRC

Figure 4. Completion and recovery performance comparisons on Yale.

In the above two figures, the sample probability is set to 0.1 in the first two lines of images and 0.3 in the latter two lines. For each figure, the first two columns of images display the original and the incomplete images respectively, where the positions with missing entries are shown in white. The third and the fifth columns of images give the completed images by SRC and ISRC respectively. The fourth and the last columns of images show the noise recovered by SRC and ISRC respectively. By comparing the completed images with the original images, we can see that ISRC not only has the better completion performance, but also automatically corrects the corruptions to a certain extent. Moreover, ISRC is more efficient in recovering noise than SRC. In summary, ISRC has better recovery performance than SRC.

6. Conclusions

This paper studies the problem of robust face classification with incomplete test samples. To address this problem, we propose a classification model based on incomplete sparse representation, which can be regarded as the generalization of sparse representation-based classification. Firstly, the incomplete sparse representation is described as an l_1-minimization and the alternating direction method of multipliers is employed to solve this optimization problem. Then, we analyze the convergence of the proposed algorithm and extend the model to the case of a batch of test samples. Finally, the experimental results on two well-known face datasets demonstrate that the proposed classification method is very effective in improving classification performance and recovering the missing entries and noise. It still needs further research on the model and algorithm of incomplete sparse representation. In the future, we will consider the sparse representation-based classification problem that both training and testing samples have missing values.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (No. 61403298, and No. 11401457), by the Natural Science Basic Research Plan in Shaanxi Province of China (No. 2014JQ8323), by the Shaanxi Provincial Education Department (No. 2013JK0587).

Author Contributions

Jiarong Shi constructed the model, developed the algorithm and wrote the manuscript. Xiuyun Zheng designed the experiments. Wei Yang implemented all experiments. All three authors were involved in organizing and refining the manuscript. All authors have read and approved the final manuscript.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Donoho, D. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306.
2. Candès, E.J.; Michael, W. An introduction to compressive sampling. *IEEE Signal Process. Mag.* **2008**, *25*, 21–30.

3. Wright, J.; Yang, A.Y.; Ganesh, A.; Sastry, S.S.; Ma, Y. Robust face recognition via sparse representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 210–227.
4. Qiao, L.; Chen, S.; Tan, X. Sparsity preserving discriminant analysis for single training image face recognition. *Pattern Recogn. Lett*. **2010**, *31*, 422–429.
5. Zhang, S.; Zhao, X.; Lei, B. Robust facial expression recognition via compressive sensing. *Sensors* **2012**, *12*, 3747–3761.
6. Wright, J.; Ma, Y.; Mairal, J.; Sapiro, G.; Huang, T.; Yan, S. Sparse representation for computer vision and pattern recognition. *Proc. IEEE* **2010**, *98*, 1031–1044.
7. Cheng, B.; Yang, J.; Yan, S.; Fu, Y.; Huang, T. Learning with L1-graph for image analysis. *IEEE Trans. Image Process.* **2010**, *19*, 858–866.
8. Yin, J.; Liu, Z.; Jin, Z.; Yang, W. Kernel sparse representation based classification. *Neurocomputing* **2012**, *77*, 120–128.
9. Huang, K.; Aviyente, S. Sparse representation for signal classification. *Neural Inf. Proc. Syst*. **2006**, *19*, 609–616.
10. Zhang, H.; Zha, Z.J.; Yang, Y.; Yan, S.; Chua, T.S. Robust (semi) nonnegative graph embedding. *IEEE Trans. Image Process*. **2014**, *23*, 2996–3012.
11. Chen, Y.; Zhang, S.; Zhao, X. Facial expression recognition via non-negative least-squares sparse coding. *Information* **2014**, *5*, 305–318.
12. Elhamifar, E.; Vidal, R. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2765–2781.
13. Candès, E.J.; Recht, B. Exact matrix completion via convex optimization. *Found. Comput. Math*. **2009**, *9*, 717–772.
14. Shi, J.; Yang, W.; Yong, L.; Zheng, X. Low-rank representation for incomplete data. *Math. Probl. Eng*. **2014**, *2014*, doi:10.1155/2014/439417.
15. Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; Eckstein, J. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn*. **2011**, *3*, 1–122.
16. Candès, E.J.; Li, X.; Ma, Y.; Wright, J. Robust principal component analysis? *J. ACM* **2011**, *58*, doi:10.1145/1970392.1970395.
17. ORL Database of Faces. Available online: http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html (accessed on 23 June 2015).
18. Yale Face Database. Available online: http://vision.ucsd.edu/content/yale-face-database (accessed on 23 June 2015).