JOURNAL OF INDUSTRIAL AND
MANAGEMENT OPTIMIZATION
Volume **8**, Number **1**, February **2012**

doi:10.3934/jimo.2012.8.163

pp. **163–178**

A FAST ℓ_1-SOLVER AND ITS APPLICATIONS TO ROBUST FACE RECOGNITION

Huining Qiu and Xiaoming Chen

Sun Yat-sen Univeristy
NO.135 Xiguangxi Road, Guangzhou 510275, China

Wanquan Liu and Guanglu Zhou

Curtin University
GPO Box U1987, Perth, WA 6845, Australia

Yiju Wang

Qufu Normal University
Rizhao 276800, Shandong, China

Jianhuang Lai

Sun Yat-sen Univeristy
NO.135 Xiguangxi Road, Guangzhou 510275, China

Abstract. In this paper we apply a recently proposed Lagrange Dual Method (LDM) to design a new Sparse Representation-based Classification (LDM-SRC) algorithm for robust face recognition problem. The proposed approach improves the efficiency of the SRC algorithm significantly. The proposed algorithm has the following advantages: (1) it employs the LDM ℓ_1-solver to find solution of the ℓ_1-norm minimization problem, which is much faster than other state-of-the-art ℓ_1-solvers, e.g. ℓ_1-magic and ℓ_1_ℓ_s . (2) The LDM ℓ_1-solver utilizes a new Lagrange-dual reformulation of the original ℓ_1-norm minimization problem, not only reducing the problem size when the dimension of training image data is much less than the number of training samples, but also making the dual problem become smooth and convex. Therefore it converts the non-smooth ℓ_1-norm minimization problem into a sequence of smooth optimization problems. (3) The LDM-SRC algorithm can maintain good recognition accuracy whilst reducing the computational time dramatically. Experimental results are presented on some benchmark face databases.

1. Introduction.

Face recognition is an active research topic in the past two decades, and both the academic and industry communities have devoted much efforts to automatic face recognition development. The core problem of face recognition is to design algorithms which can perform automatic person identification or verification, when a digital image or a video frame sequence of that person is provided. For this purpose, numerous algorithms have been proposed by researchers with many significant results achieved [36, 26, 27].

2000 *Mathematics Subject Classification.* Primary: 65D18, 68U10; Secondary: 65K05.

Key words and phrases. Compressed sensing, ℓ_1 optimization, Lagrange dual, face recognition.

This paper was presented in the 8th ICOTA Conference held in Shanghai during 10-13 December, 2010. A brief version of this paper was published in the Conference Proceedings. The reviewing process of the paper was handled by Naihua Xin as the Guest Editor.

Many state-of-art face recognition algorithms are appearance-based methods, which use the features of the pixel intensity values in a digital face image. These pixel intensities are the measurements of light radiance emitted from a person along certain rays in space, and contain abundant information which can be used to determine identity from appearance. Some typical appearance-based methods include: subspace-based methods, Hidden Markov Model (HMM) methods, Bayesian methods, Support Vector Machine (SVM) methods, Kernel methods. In the literature of face recognition, the subspace-based algorithms have been extensively studied because of its effectiveness and meaningful explanations. To briefly review subspace-based algorithms we can find a number of classic algorithms: Eigenfaces and its variants [28, 25, 29], Fisherfaces and numerous modifications [3, 23], Laplacianfaces and its extensions [17, 35], ICA-based methods [1, 11], NMF-based methods [19, 20], unified subspace method [30], to name only a few. Most of these methods treat face images as points in a vector space, and adopt statistical or geometrical techniques to analyze the structure of the image vectors, trying to find effective representations in some other transformed vector spaces according to various objectives. The result of subspace analysis can generally lead to a projective subspace, in which the face image vectors can be projected so that more compact and effective features can be extracted. Once the subspace has been found, both the prototype training images and the test images are projected in the subspace, and further classification is performed based on adequate similarities measured in such subspace. The effectiveness for subspace-based methods relies on the applicability of approximating the face space with a particular subspace. Successful results have been reported widely in demonstrating the effectiveness of subspace-based methods for face recognition.

To date, high performance of face recognition algorithms has been achieved in laboratory controlled environments. However, the practical performance of face recognition techniques is still unsatisfactory in less controlled or uncontrolled environments, which is a barrier to their implementation in real world applications. Recent standardized vendor face technology tests have revealed that the main challenges for practical face recognition are the disruptive large intra-subject variations in human face appearance due to 3D head poses, illumination variations (e.g. indoor/outdoor environment), facial expressions, occlusions with other objects or accessories (e.g. sunglasses and scarfs), facial hair and aging [21, 22, 27, 10]. These interference factors can degenerate the performance of face recognition algorithms dramatically. As these difficulties exist in face recognition, more robust face recognition algorithms are still needed.

Recently, Wright *et al.* have developed a new face recognition framework for the robust face recognition problem, namely the Sparse Representation-based Classification (SRC) algorithm [32]. Their work is based on a newly developed compressed sensing theory, and tends to show its robust performance compared with traditional face recognition techniques. Compressed sensing is a technique firstly developed in signal processing community for reconstructing a sparse signal by utilizing the prior knowledge of its sparsity structure [7, 12]. The compressed sensing theory resolves to minimize the ℓ_0 norm, which is equivalently relaxed to minimizing the ℓ_1 norm under certain conditions. This transition of minimization object can yield surprisingly desirable solutions which manifest their robust property against a variety of noises in many problems. Due to its solid mathematics and attractive robustness, compressed sensing has already drawn immense attention in areas of mathematics, optimization, information theory, statistical signal processing, high-dimensional data analysis. A survey about compressed sensing and its broad applications can be found in [8].

The SRC algorithm for face recognition is robust in a sense that the sparse representation is less sensitive to variations in the face images, which could be caused by illuminations, occlusions, noises, or random pixel corruptions. The algorithm has demonstrated its robustness against noises and corruptions in experiments. Nevertheless, the major disadvantage of the SRC algorithm is its very expensive computational cost and limits its current applicability. Due to its vector representation of face images data, the SRC algorithm needs to solve an ℓ_1-regularized optimization problem whose size is the total number of pixels in the images, which can be extremely large when high resolution images are used.

Regarding the computational issue, some alternative approaches can be considered to overcome the problem. First, dimensionality reduction can be performed on the input images and then extracted feature vectors can be used instead of original pixel features. However, this approach might risk losing useful information of the original images. Second, some reformulations of the problem might be pursued to find the equivalent solution faster. In this paper, we follow the second approach and aim to reduce computational complexity of the SRC algorithm by reformulating the ℓ_1-regularized optimization problem in the SRC algorithm. The idea is based on our recent result in [31] and the ℓ_1-regularized optimization problem in the SRC algorithm is converted into a lower dimensional problem via Lagrange dual theorem.

The paper is organized as follows. In Section 2, we review the SRC algorithm and point out its high computational cost problem. Then, in Section 3 we propose a LDM-SRC algorithm which can solve the ℓ_1 minimization problems efficiently. In Section 4 we compare the proposed LDM-SRC algorithm with two other state-of-the-art algorithms (i.e. ℓ_1-magic and ℓ_1_ℓ_s) on some benchmark datasets, which reveals that the proposed algorithms can speed up without decreasing the recognition rate. Finally, we conclude the paper in Section 5.

2. The SRC problem.

In this section we briefly review the SRC algorithm and analyze the major high computational cost of the involved ℓ_1 minimization problem.

In face recognition research, it is generally conceived that there exists a “face subspace” which is formed by one person’s face images under different variations (e.g. pose, illumination, expression). As a result, linear models can be used to approximate the “face subspaces” by all the people in the training set. The recently proposed sparse representation-based face model is developed based on this hypothesis, and it uses all known training sample images to span a face subspace. For a test face image whose class label is unknown, one tries to reconstruct the test image sparsely from the training samples.

According to this model hypothesis, if given sufficient training samples of one person, then any new (test) sample for this person will approximately lie in the linear span of the training samples associated with this person. To be more precise, let us say, a database consists of k classes denoted as

$$\mathcal{A} = \{\mathbf{A}_{1,1}, \ldots, \mathbf{A}_{1,n_1}, \ldots, \mathbf{A}_{k,1}, \ldots, \mathbf{A}_{k,n_k}\},$$

where $\mathbf{A}_{i,l}$ is the l-th image belonging to class i, and n_i is number of samples for class i, $i = 1, \ldots, k$. By stacking pixels of each image $\mathbf{A}_{i,l}$ into a column vector $\mathbf{v}_{i,l}$, one can build up a matrix $\mathbf{A}$ to represent the training samples

$$\mathbf{A} = [\mathbf{v}_{1,1}, \ldots, \mathbf{v}_{1,n_1}, \ldots, \mathbf{v}_{k,1}, \ldots, \mathbf{v}_{k,n_k}] \in \mathcal{R}^{L\times N}, \tag{1}$$

where $L = hw$ is the number of pixels for an $h \times w$ image, and $N = n_1 + \ldots + n_k$ is the total number of samples for all classes.

For a new test image $\mathbf{y}$, we expect to represent it using linear combinations of samples from the database

$$\mathbf{y} = \mathbf{A}\mathbf{x}_0. \tag{2}$$

Most ideally, if $\mathbf{y}$ is known from person i, then based on the assumption that person i's face subspace is sufficient to represent itself, so the coefficients $\mathbf{x}_0$ should have a form of

$$\mathbf{x}_0 = [0, \ldots, 0, \alpha_{i,1}, \alpha_{i,2}, \ldots, \alpha_{i,n_i}, 0 \ldots, 0]\,. \tag{3}$$

In other words, the solution $\mathbf{x}_0$ in linear equation (2) should only have non-zero values at positions corresponding to the same person as the test image, therefore it should be very sparse. Thus, one can use "sparsity" as a heuristic principle for solving the linear equation (2), even though not knowing the true identity of the test image. For this purpose, one can set up an objective to measure the "sparsity" of the coefficients $\mathbf{x}$, that is

$$(\ell_0): \qquad \begin{aligned} &\hat{\mathbf{x}}_0 = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_0 \\ &\text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{y}. \end{aligned} \tag{4}$$

where $\|\cdot\|_0$ is the zero norm which simply counts the number of non-zero elements of $\mathbf{x}$. However such kind of optimization problem is NP hard and not easily tractable, therefore alternatives are needed. From the compressed sensing theory one knows that a restriction of ℓ_1-norm has an equivalent effect of producing sparse solutions. When the basis matrix $\mathbf{A}$ satisfies certain randomness conditions (e.g. the statistical restricted isometry property), (4) is equivalent to the following ℓ_1-norm minimization problem with high probability

$$(\ell_1): \qquad \begin{aligned} &\hat{\mathbf{x}}_1 = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 \\ &\text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{y}. \end{aligned} \tag{5}$$

Moreover, (4) is a model not taking account the existence of noise and can be enhanced by combining the ℓ_1-norm with an error tolerance constraint, thus leads to

$$(\ell_1^s): \qquad \begin{aligned} &\hat{\mathbf{x}}_1 = \arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 \\ &\text{s.t. } \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2 \le \epsilon. \end{aligned} \tag{6}$$

The two ℓ_1-norm based models (5) and (6) are both used in the SRC algorithm, and they are different because (5) is a noise-free model and (6) is a model tolerating the existence of noises. Nevertheless, they can be solved using the same optimization technique. Once the sparse coefficients $\mathbf{x}$ is solved, then they can be used for recognition by the SRC algorithm, which is summarized in Algorithm 1.

As can be seen, the most significant step in the SRC algorithm is how to solve the ℓ_1 minimization problem (5) or (6) effectively and efficiently. The authors in [32] use the ℓ_1-magic package [9] to find their solutions, and all their experimental results are obtained by downsampling the original images to smaller size images. In our experiments we find that the ℓ_1-magic solver has very high computational cost, and it even fails to run in some very large scale problems. This motivates us to find other alternative ℓ_1-solvers for practical use, such as $\ell_1\_\ell_s$ [18]. The $\ell_1\_\ell_s$ solver is

Algorithm 1 Sparse Representation-based Classification (SRC) (by Wright et al. [32])

```
Input: a matrix of training samples for $k$ classes
    $\mathbf{A} = [\mathbf{v}_{1,1}, \ldots, \mathbf{v}_{1,n_1}, \ldots, \mathbf{v}_{k,1}, \ldots, \mathbf{v}_{k,n_k}] \in \mathcal{R}^{(wh)\times N}$
(each column of $\mathbf{A}$ is a vectorization of training sample image $\mathbf{A}_{i,i_l}$); the class labels $class(p), p = 1, \ldots, k$; the corresponding class labels $label(i)$ of each training sample vector $A(:, i)$; a test sample $\mathbf{y} \in \mathcal{R}^{(wh)\times 1}$; and an optional error tolerance parameter $\epsilon > 0$.
Normalize the columns of $A$ to have unit $\ell^2$-norm.
Solve the $\ell_1$-norm minimization problem
    $\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{y},$
or alternatively solve the $\ell_1$-norm minimization problem
    $\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{s.t. } \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2 \leq \epsilon.$
Compute the per-class residuals
    $r_p(\mathbf{y}) = \|\mathbf{y} - \mathbf{A}\,\delta_p(\hat{\mathbf{x}})\|_2 \quad \text{for } p = 1, \ldots, k,$
where $\delta_p(\mathbf{x})$, for $p = 1, \ldots, k$ is a vector for the $p$-th class whose entries are defined as
    $\text{for } i = 1, \ldots, N, \quad \delta_p^{(i)}(\hat{\mathbf{x}}) = \begin{cases} \hat{x}_i, & \text{if } label(i) \text{ is } class(p) \\ 0, & \text{otherwise.} \end{cases}$
Output: $identity(\mathbf{y}) = class(p_*),\ p_* = \arg \min_p r_p(\mathbf{y}).$
```

a better algorithm for ℓ_1 optimization currently and it has been specially optimized for large scale problems. Our experiments show that in general it can run several times faster than the ℓ_1-magic solver and have good performance for face recognition problems. Even though, using the ℓ_1_ℓ_s solver in the SRC algorithm to recognize a 90×90 image with 1560 training images would take more than 1000 seconds. This is obviously impractical and the computation cost is too expensive. We have noted that some faster and memory economic algorithms have been proposed to solve the ℓ_1 minimization problem, such as GPSR [13], SPGL1 [4], SpaRSA [33] and YALL1 [34]. In particular, YALL1 is a new development of alternating direction methods and it can be applied to various forms of ℓ_1 minimization problems.

In next section, we will propose an LDM ℓ_1-solver as well as a modified LDM-SRC algorithm which can run faster whilst maintaining the face recognition performance.

3. **The proposed LDM-SRC algorithm.** In this section, we introduce the Lagrange Dual Method ℓ_1-solver for both the noise free problem (5) and the problem (6) with noise tolerance. We will consider the noise free case and the case with noise respectively, and reformulate these problems based on the perturbation technique and dual theorem. After that we will apply the LDM ℓ_1-solver to a proposed LDM-SRC algorithm.

3.1. **The LDM ℓ_1-solver for noise free problems.** In this case, we consider problem (5). First we note that the problem (5) is a non-smooth but convex optimization problem. We will convert it into a smooth optimization problem by using the Lagrange dual technique.

For $\mathbf{x} \in R^n$, let $\mathbf{x}^+$and $\mathbf{x}^-$respectively be the orthogonal projections of $\mathbf{x}$ and $-\mathbf{x}$ onto the nonnegative orthant, i.e, $\mathbf{x}^+ = max\{\mathbf{x}, 0\}$, $\mathbf{x}^- = max\{-\mathbf{x}, 0\}$ in componentwise. Then $\mathbf{x} = \mathbf{x}^+ - \mathbf{x}^-$ and $\|\mathbf{x}\|_1 = \|\mathbf{x}^+ + \mathbf{x}^-\|_1$ hold. In this case, the problem (5) becomes a standard linear optimization program

$$(\ell_1') : \quad \begin{aligned} &\hat{\mathbf{x}}_1 = \arg\min_{\mathbf{x}} \sum_{j=1}^n x_j^+ + \sum_{j=1}^n x_j^- \\ &\text{s.t.} \quad \mathbf{A}\mathbf{x}^+ - \mathbf{A}\mathbf{x}^- = \mathbf{y}, \\ &\qquad \mathbf{x}^+ \geq 0, \mathbf{x}^- \geq 0\ . \end{aligned} \tag{7}$$

For this linear program problem, Theorem 1 in [24] states that there exists a positive number $\overline{\tau}$ such that for any $\tau \in [0, \overline{\tau}]$, problem (7) can be equivalently transformed into the following problem

$$(\ell_1'') : \quad \begin{aligned} &\hat{\mathbf{x}}_1 = \arg\min_{\mathbf{x}} \quad \frac{\tau}{2}\sum_{j=1}^n (x_j^+)^2 + \frac{\tau}{2}\sum_{j=1}^n (x_j^-)^2 + \sum_{j=1}^n x_j^+ + \sum_{j=1}^n x_j^- \\ &\text{s.t.} \quad \mathbf{A}\mathbf{x}^+ - \mathbf{A}\mathbf{x}^- = \mathbf{y}, \\ &\qquad \mathbf{x}^+ \geq 0, \mathbf{x}^- \geq 0\ . \end{aligned} \tag{8}$$

Then we consider its dual program

$$\hat{\lambda} = \arg\max_{\lambda} \min_{\mathbf{x}^+, \mathbf{x}^- \geq 0} \quad L(\mathbf{x}^+, \mathbf{x}^-; \lambda), \tag{9}$$

where the Lagrange function is

$$\begin{aligned} L(\mathbf{x}^+, \mathbf{x}^-; \lambda) \quad &= \quad \frac{\tau}{2}\sum_{j=1}^n (x_j^+)^2 + \frac{\tau}{2}\sum_{j=1}^n (x_j^-)^2 + \sum_{j=1}^n x_j^+ + \sum_{j=1}^n x_j^- \\ &+ \quad \lambda^T \left(\mathbf{A}\mathbf{x}^+ - \mathbf{A}\mathbf{x}^- - \mathbf{y}\right). \end{aligned} \tag{10}$$

Further, if we denote the j-th column of $\mathbf{A}$ by $\mathbf{A}_{\cdot j}$ for $j = 1, \ldots, n$, then the inner minimization problem in the dual problem (9) can be simplified as

$$\begin{aligned} &\min_{\mathbf{x}^+, \mathbf{x}^- \geq 0} \quad L(\mathbf{x}^+, \mathbf{x}^-; \lambda) \qquad (11) \\ = \ &\sum_{j=1}^n \min_{\mathbf{x}^+, \mathbf{x}^- \geq 0} \frac{\tau}{2}\left(x_j^+ + \frac{1 + \lambda^T \mathbf{A}_{\cdot j}}{\tau}\right)^2 + \sum_{j=1}^n \min_{\mathbf{x}^+, \mathbf{x}^- \geq 0} \frac{\tau}{2}\left(x_j^- + \frac{1 - \lambda^T \mathbf{A}_{\cdot j}}{\tau}\right)^2 \\ &-\frac{n}{\tau} - \lambda^T \mathbf{y} - \frac{1}{\tau}\lambda^T \mathbf{A}\mathbf{A}^T \lambda. \end{aligned}$$

The last expression above implies that for any fixed λ the Lagrange function takes its minimum over the nonnegative orthant at

$$\hat{x}_j^+ = \max\{-\frac{1 + \lambda^T \mathbf{A}_{\cdot j}}{\tau}, 0\}, \ \hat{x}_j^- = \max\{-\frac{1 - \lambda^T \mathbf{A}_{\cdot j}}{\tau}, 0\}\ . \tag{12}$$

Thus

$$\begin{aligned}
& \min_{\mathbf{x}^+,\mathbf{x}^-\geq 0} \quad L(\mathbf{x}^+,\mathbf{x}^-;\lambda) \\
= \; & \frac{1}{2\tau}\sum_{j=1}^{n}\max{}^2\{1+\lambda^T\mathbf{A}_{\cdot j},0\}+\frac{1}{2\tau}\sum_{j=1}^{n}\max{}^2\{1-\lambda^T\mathbf{A}_{\cdot j},0\} \\
- \; & \frac{n}{\tau}-\lambda^T\mathbf{y}-\frac{1}{\tau}\lambda^T\mathbf{A}\mathbf{A}^T\lambda, \qquad (13)
\end{aligned}$$

so the dual problem (9) can be written as

$$\hat{\lambda} = \arg\max_{\lambda} \; -\frac{1}{2\tau}\theta(\lambda)-\frac{n}{\tau} = \arg\min_{\lambda}\theta(\lambda), \qquad (14)$$

$$\theta(\lambda) \overset{\triangle}{=} 2\lambda^T\mathbf{A}\mathbf{A}^T\lambda+2\tau\lambda^T\mathbf{y}-\sum_{j=1}^{n}\max{}^2\{1+\lambda^T\mathbf{A}_{\cdot j},0\}-\sum_{j=1}^{n}\max{}^2\{1-\lambda^T\mathbf{A}_{\cdot j},0\}.$$

Now, we give further analysis to problem (14). First, we note that the problem (8) is a convex program with linear constraints, and then the strong dual theorem holds [2]. Thus, if the optimal solution, say $\hat{\lambda}^*$, of the dual problem (14) is obtained, then the solution of the primal problem (8) can be obtained explicitly by

$$\hat{\mathbf{x}}^* = \max\{-\frac{1+\hat{\lambda}^{*T}\mathbf{A}_{\cdot j}}{\tau},0\}-\max\{-\frac{1-\hat{\lambda}^{*T}\mathbf{A}_{\cdot j}}{\tau},0\}. \qquad (15)$$

Second, although the maximum function $\max\{\cdot,0\}$ is non-differentiable, its square is smooth. This implies that the objective function in the dual program (14) is continuously differentiable with derivative given by

$$\begin{aligned}
\nabla\theta(\lambda) \; = \; & 4\mathbf{A}\mathbf{A}^T\lambda+2\tau\mathbf{y}-\sum_{j=1}^{n}2\max\{1+\lambda^T\mathbf{A}_{\cdot j},0\}\mathbf{A}_{\cdot j} \\
- \; & \sum_{j=1}^{n}2\max\{1-\lambda^T\mathbf{A}_{\cdot j},0\}\mathbf{A}_{\cdot j}. \qquad (16)
\end{aligned}$$

Therefore, one can use the gradient-based method to solve the dual problem (14). Observing that the objection function in (14) is convex, one can see that any stationary point of problem (14) is a global optimal solution, and this gives us more flexibility in solving the problem.

Based on above analysis, one can see that the primal problem (5) is equivalently transformed into a continuously differentiable and convex program with m variables. Usually, m is much smaller than the size of the original problem n, where m is the dimensionality of data and n is the number of training samples. For face recognition applications, it is usually held that $m \ll n$. This implies that the size of the dual problem is much smaller than that of the original problem (5), and this provides possibility to design more efficient algorithms for the original sparse representation problem.

3.2. **The LDM ℓ_1-solver for problems with noise.** Now we turn to consider problem (6) which is with noise tolerance. In the following, we will derive a Lagrange dual form of this problem.

According to Fuchs [14, 15, 16] and Boyd [5], the quadratic programming problem (6) can be equivalently transformed into the following ℓ_1-norm regularization problem

$$\hat{\mathbf{x}}_1 = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2 + \rho\|\mathbf{x}\|_1 \qquad \text{for some } \rho. \tag{17}$$

This transformation is equivalent provided the parameter ρ is properly chosen.

If we introduce a new variable $\mathbf{z}$ and let $\mathbf{z} = \mathbf{A}\mathbf{x}$, the problem (17) can be rewritten as a linearly constrained optimization problem

$$(\ell_1^s)': \quad \begin{aligned} &\hat{\mathbf{x}}_1 = \arg\min_{\mathbf{z},\,\mathbf{x}} \frac{1}{2}\|\mathbf{z}-\mathbf{y}\|_2^2 + \rho\|\mathbf{x}\|_1 \\ &\text{s.t.} \quad \mathbf{z} = \mathbf{A}\mathbf{x}. \end{aligned} \tag{18}$$

Similarly, by introducing two nonnegative variables $\mathbf{x}^+$ and $\mathbf{x}^-$ we can obtain the following quadratic minimization problem

$$(\ell_1^s)'': \quad \begin{aligned} &\hat{\mathbf{x}}_1 = \arg\min_{\mathbf{z},\,\mathbf{x}^+,\,\mathbf{x}^-} \frac{1}{2}\|\mathbf{z}-\mathbf{y}\|_2^2 + \rho\sum_{j=1}^n x_j^+ + \rho\sum_{j=1}^n x_j^- \\ &\text{s.t.} \quad \mathbf{z} = \mathbf{A}\mathbf{x}^+ - \mathbf{A}\mathbf{x}^-, \\ &\qquad\ \ \mathbf{x}^+ \geq 0,\, \mathbf{x}^- \geq 0\ . \end{aligned} \tag{19}$$

Again from Theorem 1 in [24], for any fixed $\mathbf{z} \in R^m$, there exists a parameter $\tilde{\tau}(\mathbf{z}) > 0$, such that for any $\tau \in [0, \tilde{\tau}(\mathbf{z})]$, the minimization problem (19) is equivalent to the following perturbed linear program

$$(\ell_1^s)''': \quad \begin{aligned} &\hat{\mathbf{x}}_1 = \arg\min_{\mathbf{z},\,\mathbf{x}^+,\,\mathbf{x}^-} \frac{1}{2}\|\mathbf{z}-\mathbf{y}\|_2^2 + \frac{\tau}{2}\sum_{j=1}^n (x_j^+)^2 + \frac{\tau}{2}\sum_{j=1}^n (x_j^-)^2 + \\ &\rho\sum_{j=1}^n x_j^+ + \rho\sum_{j=1}^n x_j^- \\ &\text{s.t.} \quad \mathbf{A}\mathbf{x}^+ - \mathbf{A}\mathbf{x}^- = \mathbf{z}, \\ &\qquad\ \ \mathbf{x}^+ \geq 0,\, \mathbf{x}^- \geq 0\ . \end{aligned} \tag{20}$$

For this problem, we again consider its Lagrange dual problem

$$\hat{\lambda} = \arg\max_{\lambda} \min_{\mathbf{x}^+ \geq 0, \mathbf{x}^- \geq 0, \mathbf{z}} L(\mathbf{x}^+, \mathbf{x}^-, \mathbf{z}; \lambda), \tag{21}$$

where the Lagrange function is as follows

$$\begin{aligned} L(\mathbf{x}^+, \mathbf{x}^-, \mathbf{z}; \lambda) \quad = \quad & \frac{1}{2}\|\mathbf{z}-\mathbf{y}\|_2^2 + \frac{\tau}{2}\sum_{j=1}^n (x_j^+)^2 + \frac{\tau}{2}\sum_{j=1}^n (x_j^-)^2 \\ & + \rho\sum_{j=1}^n x_j^+ + \rho\sum_{j=1}^n x_j^- + \lambda^T\left(\mathbf{A}\mathbf{x}^+ - \mathbf{A}\mathbf{x}^- - \mathbf{z}\right). \end{aligned} \tag{22}$$

A similar argumentation as that in last subsection yields the following simplified representation of the inner minimization problem in the dual problem (21)

$$\begin{aligned}\min_{\mathbf{x}^+\geq 0,\mathbf{x}^-\geq 0,\mathbf{z}} L(\mathbf{x}^+,\mathbf{x}^-,\mathbf{z};\lambda) &= \frac{1}{2\tau}\sum_{j=1}^{n}\max{}^2\{\rho+\lambda^T\mathbf{A}_{\cdot j},0\} \\ &\quad +\frac{1}{2\tau}\sum_{j=1}^{n}\max{}^2\{\rho-\lambda^T\mathbf{A}_{\cdot j},0\} \\ &\quad -\frac{1}{2}\lambda^T\lambda-\lambda^T\mathbf{y}-\frac{1}{\tau}\lambda^T\mathbf{A}\mathbf{A}^T\lambda-\frac{n\rho}{\tau}, \qquad (23)\end{aligned}$$

thus the dual problem (21) can be reduced to

$$\begin{aligned}\hat{\lambda} &= \arg\max_{\lambda} -\frac{1}{2\tau}\theta_{(\tau,\rho)}(\lambda)-\frac{n\rho}{\tau} = \arg\min_{\lambda}\theta_{(\tau,\rho)}(\lambda), \qquad (24)\\ \theta_{(\tau,\rho)}(\lambda) &\triangleq \lambda^T(\tau\mathbf{I}+2\mathbf{A}\mathbf{A}^T)\lambda+2\tau\lambda^T\mathbf{y}-\sum_{j=1}^{n}\max{}^2\{\rho+\lambda^T\mathbf{A}_{\cdot j},0\} \\ &\quad -\sum_{j=1}^{n}\max{}^2\{\rho-\lambda^T\mathbf{A}_{\cdot j},0\}.\end{aligned}$$

Since the primal problem (20) is a convex program with linear constraints, the strong dual program holds [5]. Thus, the solution of the primal problem (20) can be obtained via the optimal solution, say $\hat{\lambda}^*$, of the dual problem (24) with the following explicit formula

$$\hat{\mathbf{x}}^* = \max\{-\frac{\rho+\hat{\lambda}^{*T}\mathbf{A}_{\cdot j}}{\tau},0\}-\max\{-\frac{\rho-\hat{\lambda}^{*T}\mathbf{A}_{\cdot j}}{\tau},0\}. \qquad (25)$$

Then the task for solving problem (6) is turned into solving the dual problem (24), which has similar advantages as the dual problem (14) and can be solved more efficiently using smooth optimization techiniques.

3.3. The LDM-SRC algorithm.

From above discussions, we can summarize our main result in the following theorem.

Theorem 3.1. *In order to solve the ℓ_1 minimization problems in (5) and (6), we have:*

(1) There exists a constant $\tilde{\tau}>0$, such that for any $\tau\in(0,\tilde{\tau}]$, the solution of the noise free ℓ_1 minimization problem (5) can be obtained by

$$\hat{\mathbf{x}}^* = max\{-\frac{1+\hat{\lambda}^{*T}\mathbf{A}_{\cdot j}}{\tau},0\}-max\{-\frac{1-\hat{\lambda}^{*T}\mathbf{A}_{\cdot j}}{\tau},0\}, \qquad (26)$$

where $\hat{\lambda}^*$ is the solution of the Lagrange dual problem (14).

(2) There exists constants $\rho>0,\tilde{\tau}>0$, such that for any $\tau\in(0,\tilde{\tau}]$, the solution of the noisy ℓ_1 minimization problem (6) can be obtained by

$$\hat{\mathbf{x}}^* = max\{-\frac{\rho+\hat{\lambda}^{*T}\mathbf{A}_{\cdot j}}{\tau},0\}-max\{-\frac{\rho-\hat{\lambda}^{*T}\mathbf{A}_{\cdot j}}{\tau},0\}, \qquad (27)$$

where $\hat{\lambda}^*$ is the solution of the Lagrange dual problem (24).

Next we consider how to solve the Lagrange dual minimization problem (14) and (24). In fact, there are many algorithms in numerical optimization which can be used to solve (14) and (24). In this paper, we choose the limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) algorithm [6] for solving the problems because it can solve large scale unconstrained minimization problems efficiently due to its moderate memory requirement. The L-BFGS algorithm belongs to the family of quasi-Newton optimization methods, and it uses a limited memory variation of the BFGS update to approximate the inverse Hessian matrix. An important feature of L-BFGS is that it never explicitly forms or stores the Hessian matrix, but maintains a history of the past updates of the position and gradient.

Based on the BFGS algorithm, we design a practical version of algorithm for solving (5) and (6) and we call it the LDM ℓ_1-solver which is summarized as in Algorithm 2. For more detailed discussion on convergence of the LDM ℓ_1-solver algorithm, we refer the reader to our previous paper [31]. Note that the algorithm employs a decreasing sequence $\{\tau^k\}$ for searching an appropriate value of τ which can ensure the equivalence between (5) and (14) in the case without noise and that between (6) and (24) in the noisy scenario. In our practice, we choose the sequence $\{\tau^k\}$ as $\tau^0 = 0.005, \tau^1 = 0.001, \tau^2 = 0.0005, \tau^3 = 0.0001, \cdots$, which works well for our test problems. For the noisy problems, the parameters ε and ρ are of implicit correspondence and need to be carefully determined according to the problem. In our experiments, we first determine these parameters by making different trials for some testing samples and choose the values which yield the best performance, then fix them for other testing samples. Additionally, the L-BFGS method is dependent on a parameter – the number of history iterations kept in memory for computing Hessian, which we let it be 6 in our experiments.

With the LDM ℓ_1-solver algorithm, we now are able to propose an accelerated LDM-SRC algorithm for face recognition applications, as described in Algorithm 3.

Algorithm 2 LDM ℓ_1-solver

```
Input: a matrix $A \in R^{m\times n}$, a vector $y \in R^m$, parameter $\rho$ (only needed for
    problems with noise), and a sequence $\tau^{(1)} > \tau^{(2)} > \cdots > 0$ satisfying $\tau^{(k)} \to 0$
    as $k \to \infty$.
Set $k = 1$.
for $k = 1, 2, \cdots$ until a stopping criterion is satisfied do
    if the problem is the noise free problem (5), then compute $\lambda^{(k)}$ using the
        limited-memory BFGS algorithm to solve problem (14), and let
        $$\mathbf{x}_j^k = \max\{-\frac{1+\lambda^T\mathbf{A}_{\cdot j}}{\tau^{(k)}}, 0\} - \max\{-\frac{1-\lambda^T\mathbf{A}_{\cdot j}}{\tau^{(k)}}, 0\}, j = 1, 2, \ldots, n.$$
    else if the problem is the problem with noise (6), then compute $\lambda^{(k)}$ using
        the limited-memory BFGS algorithm to solve problem (24), and let
        $$\mathbf{x}_j^k = \max\{-\frac{\rho+\lambda^T\mathbf{A}_{\cdot j}}{\tau^{(k)}}, 0\} - \max\{-\frac{\rho-\lambda^T\mathbf{A}_{\cdot j}}{\tau^{(k)}}, 0\}, j = 1, 2, \ldots, n.$$
    end if
    set $k = k + 1$.
end for
Output: the sparse coefficients vector $\mathbf{x} = \mathbf{x}^k$.
```

4. **Experimental results.**

In order to show the efficiency of the proposed algorithm, we have tested the algorithm on three benchmark face databases, i.e. the

Algorithm 3 LDM-SRC for Face Recognition

```
Input: a matrix of training samples for $k$ classes
    $\mathbf{A} = [\mathbf{v}_{1,1}, \ldots, \mathbf{v}_{1,n_1}, \ldots, \mathbf{v}_{k,1}, \ldots, \mathbf{v}_{k,n_k}] \in \mathcal{R}^{(wh)\times N}$
    (each column of $\mathbf{A}$ is a vectorization of training sample image $\mathbf{A}_{i,i_l}$); the class labels $class(p), p = 1, \ldots, k$; the corresponding class labels $label(i)$ of each training sample vector $A(:, i)$; a test sample $\mathbf{y} \in \mathcal{R}^{(wh)\times 1}$; and an optional error tolerance parameter $\epsilon > 0$.
Normalize the columns of $A$ to have unit $\ell^2$-norm.
Use the LDM $\ell_1$-solver in Algorithm 2 to solve the noise free problem (5) or the problem with noise (6), and obtain the sparse coefficients vector $\mathbf{x}$.
Compute the per-class residuals
    $r_p(\mathbf{y}) = \|\mathbf{y} - \mathbf{A}\,\delta_p(\mathbf{x})\|_2 \quad \text{for } p = 1, \ldots, k,$
    where $\delta_p(\mathbf{x})$, for $p = 1, \ldots, k$ is a vector for the $p$-th class whose entries are defined as
    $\text{for } i = 1, \ldots, N, \quad \delta_p^{(i)}(\mathbf{x}) = \begin{cases} x_i, & \text{if } label(i) \text{ is } class(p) \\ 0, & \text{otherwise.} \end{cases}$
Output: $identity(\mathbf{y}) = class(p_*),\ p_* = \arg\min_p r_p(\mathbf{y})$.
```

ORL database, the Yale B plus Extended database and the AR database. For comparison, two other ℓ_1-solvers have been chosen to do the same face recognition test and they are ℓ_1-magic [9] and ℓ_1_ℓ_s [18].

Regarding the ORL face database, it contains 40 individuals, each with 10 face images, so totally 400 face images are available; the original image size of the database is 92 × 112, and we have cropped it into the size of 90 × 90.

Regarding the Yale B plus Extended face database, it contains 38 individuals, each with many variants of poses and illuminations. In our experiments we adopted these images with pose of frontal view (corresponding to those with filenames as '*_P00*') and all illuminations in subset 1 and 2. We crop and resize them into the size of 90 × 90.

Regarding the AR face database, we choose 120 individuals, each with 26 images available. We also crop and resize them into the size of 90 × 90.

All the computations are carried out using MATLAB on a server machine with Intel(R) Xeon(R) CPU 2.33GHz, 16GB of RAM, Windows Server 2003 Standard x64 Edition.

The performance of algorithms is evaluated by two criteria: the recognition accuracy R and the average computational time T, and they are defined as

$$R = \frac{n_c}{n}, \tag{28}$$

$$T = \frac{\sum_{i=1}^{n} t_i}{n}, \tag{29}$$

where n is the total number of testing samples, n_c is the number of correctly recognized testing samples, and t_i is the computational time used for recognizing the ith test sample.

The experiments are divided into two parts. In the first part we use the datasets without corruption both for training and testing images, and the purpose is to

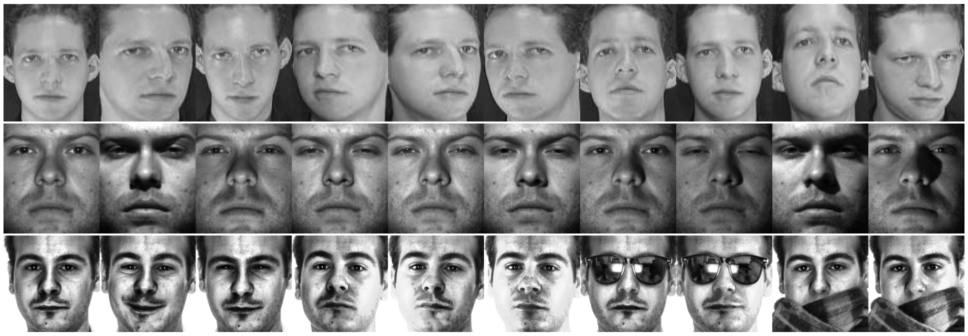


FIGURE 1. Example images from the face databases, from top to bottom are: ORL, Yale B and Extended, AR.

test the ℓ_1 solvers in the noise free case. In the second part we use corrupted images for testing and images without any corruption for training, and this is to test the ℓ_1 solvers in the case with noise tolerance. In each part, for each database, three ℓ_1 solvers employed by the SRC algorithm are executed separately and their performance (i.e. R and T) are recorded.

The experimental results are obtained on the ORL, the Yale B plus Extended (subset1+subset2), the AR face databases respectively. We show the results in Figures 2 - 7. In each figure, we show different recognition rates in a table, and draw the computational time across different image dimensionality in curves.

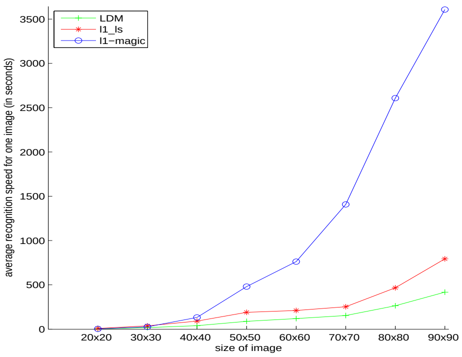
Figures 2, 3, 4 show the results when the databases are original and noise free. In this case, the three algorithms tend differently for their computational time when the image dimensionality increases. It is observed consistently on the three databases that: among the three ℓ_1 solvers, the LDM solver has the least computation time, following the ℓ_1_ℓ_s solver is second least, and the ℓ_1-magic solver consumes the most time. Regarding the recognition accuracy, it can be seen that all the three algorithm have similar recognition rates. To be more precise, the ℓ_1_ℓ_s solver has the best recognition rates generally, and the LDM solver has slightly lower recognition rates but keeps very close to the ℓ_1_ℓ_s solver, however the ℓ_1-magic solver usually has lower recognition rates than the other two algorithms.

Figures 5, 6, 7 show the results when the testing samples are corrupted by adding small block noises randomly, and the training images remain original during the test. From the experimental results, we conclude similarly as previous test: the LDM solver runs fastest, the ℓ_1_ℓ_s solver is the second fastest, and the ℓ_1-magic solver is the slowest; the trend of computational time is consistent on all three databases. Regarding the recognition accuracy, it can be seen that all three algorithms still maintain reasonable recognition rates, which can be expected. The ℓ_1_ℓ_s solver is the best generally, the LDM solver keeps close following it, but the ℓ_1-magic solver is relatively lower than the others. Overall, the recognition rates of all three algorithms on each databases are similar.

As a result, we conclude that the proposed LDM solver can run much faster while maintain a comparable recognition performance as other state-of-the-art ℓ_1 solvers.

image sizes	ℓ_1-magic	$\ell_1\_\ell_s$	LDM
$siz = 20 \times 20$	88.40	90.40	90.00
$siz = 30 \times 30$	88.60	89.10	89.00
$siz = 40 \times 40$	88.80	89.60	89.50
$siz = 50 \times 50$	88.70	89.50	89.50
$siz = 60 \times 60$	89.30	89.30	89.50
$siz = 70 \times 70$	87.90	89.00	89.00
$siz = 80 \times 80$	88.40	89.10	89.00
$siz = 90 \times 90$	88.90	89.50	89.10

(A) recognition rates

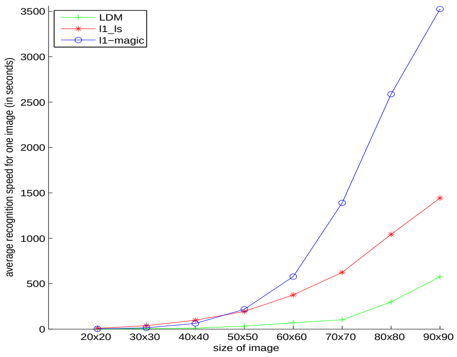


(B) speeds vs. dimensionality

FIGURE 2. Comparison of different ℓ_1-solvers on the ORL database

image sizes	ℓ_1-magic	$\ell_1\_\ell_s$	LDM
$siz = 20 \times 20$	100.00	100.00	100.00
$siz = 30 \times 30$	100.00	100.00	100.00
$siz = 40 \times 40$	100.00	100.00	100.00
$siz = 50 \times 50$	100.00	100.00	100.00
$siz = 60 \times 60$	100.00	100.00	100.00
$siz = 70 \times 70$	100.00	100.00	100.00
$siz = 80 \times 80$	100.00	100.00	100.00
$siz = 90 \times 90$	100.00	100.00	100.00

(A) recognition rates

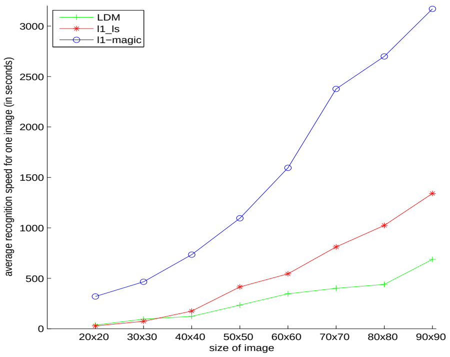


(B) speeds vs. dimensionality

FIGURE 3. Comparison of different ℓ_1-solvers on the Yale B plus Extended database

image sizes	ℓ_1-magic	$\ell_1\_\ell_s$	LDM
$siz = 20 \times 20$	86.50	90.00	89.90
$siz = 30 \times 30$	86.40	90.37	90.13
$siz = 40 \times 40$	86.90	90.38	89.96
$siz = 50 \times 50$	86.86	90.38	90.10
$siz = 60 \times 60$	89.90	90.85	90.25
$siz = 70 \times 70$	87.10	90.90	90.55
$siz = 80 \times 80$	87.50	91.28	91.09
$siz = 90 \times 90$	87.69	91.35	91.15

(A) recognition rates

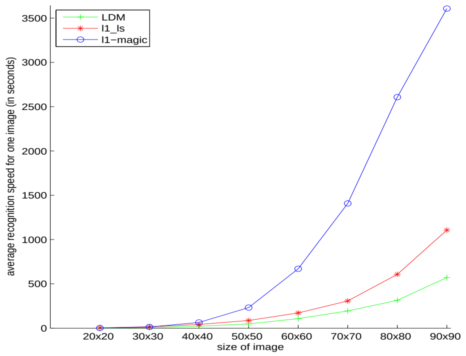


(B) speeds vs. dimensionality

FIGURE 4. Comparison of different ℓ_1-solvers on the AR database

image sizes	ℓ_1-magic	$\ell_1\_\ell_s$	LDM
$siz = 20 \times 20$	91.00	92.50	91.00
$siz = 30 \times 30$	91.00	90.50	89.50
$siz = 40 \times 40$	84.50	90.00	90.50
$siz = 50 \times 50$	89.00	89.00	89.00
$siz = 60 \times 60$	87.00	88.50	89.00
$siz = 70 \times 70$	85.50	89.00	89.50
$siz = 80 \times 80$	87.00	89.00	89.50
$siz = 90 \times 90$	87.00	89.00	89.50

(A) recognition rates

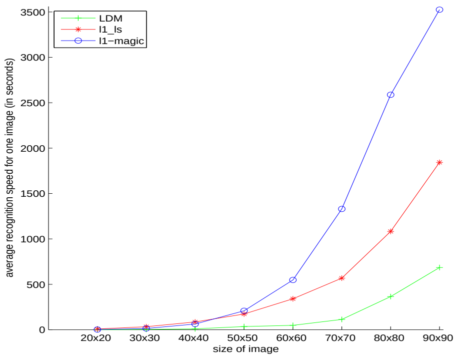


(B) speeds vs. dimensionality

FIGURE 5. Comparison of different ℓ_1-solvers on the ORL database(with corruptions)

image sizes	ℓ_1-magic	$\ell_1\_\ell_s$	LDM
$siz = 20 \times 20$	100.00	100.00	100.00
$siz = 30 \times 30$	100.00	100.00	100.00
$siz = 40 \times 40$	100.00	100.00	100.00
$siz = 50 \times 50$	100.00	100.00	100.00
$siz = 60 \times 60$	100.00	100.00	100.00
$siz = 70 \times 70$	100.00	100.00	100.00
$siz = 80 \times 80$	100.00	100.00	100.00
$siz = 90 \times 90$	100.00	100.00	100.00

(A) recognition rates

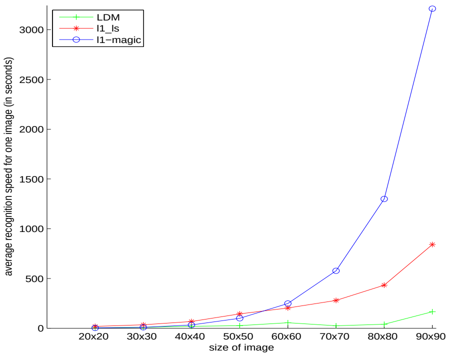


(B) speeds vs. dimensionality

FIGURE 6. Comparison of different ℓ_1-solvers on the Yale B+Extended (with corruptions)

image sizes	ℓ_1-magic	$\ell_1\_\ell_s$	LDM
$siz = 20 \times 20$	75.26	81.95	79.23
$siz = 30 \times 30$	75.17	81.96	79.62
$siz = 40 \times 40$	75.65	82.05	80.06
$siz = 50 \times 50$	76.10	82.26	80.58
$siz = 60 \times 60$	76.20	82.65	81.47
$siz = 70 \times 70$	76.45	82.90	81.79
$siz = 80 \times 80$	76.51	83.18	82.05
$siz = 90 \times 90$	77.05	83.33	82.50

(A) average recognition rates



(B) speeds vs. dimensionality

FIGURE 7. Comparison of different ℓ_1-solvers on the AR database (with corruptions)

5. **Conclusions.** In this paper, we have proposed an LDM-SRC algorithm for robust face recognition by using an alternative Lagrange-dual method for solving the ℓ_1 minimization problem. The proposed method is efficient when the number of data dimension is much larger than the number of training samples. Our experimental results show that the proposed algorithm runs much faster than other two algorithms while maintaining similar performance for face recognition tasks.

Acknowledgments. This project was partly supported by the China NSFC with grant numbers U0835005, 60803083 and 11171180, the Guangdong Program with grant number 2009A090100025 and the Australia Research Council.

REFERENCES

[1] M. S. Bartlett, J. R. Movellan and T. J. Sejnowski, *Face recognition by independent component analysis*, IEEE Transactions on Neural Networks, **6** (2002), 1450–1464.

[2] M. S. Bazaraa and C. M. Shetty, "Nonlinear Programming: Theory and Algorithms," John Wiley & Sons, New York-Chichester-Brisbane, 1979.

[3] P. N. Belhumeur, J. Hespanha and D. J. Kriegman, *Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **7** (1997), 711–720.

[4] E. van den Berg and M. P. Friedlander, *Probing the Pareto frontier for basis pursuit solutions*, SIAM Journal on Scientific Computing, **31** (2008/09), 890–912.

[5] S. Boyd and L. Vandenberghe, "Convex Optimization," Cambridge University Press, Cambridge, 2004.

[6] R. H. Byrd, P. Lu, J. Nocedal and C. Zhu, *A limited memory algorithm for bound constrained optimization*, SIAM Journal on Scientific Computing, **16** (1995), 1190–1208.

[7] E. J. Candès, J. Romberg and T. Tao, *Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information*, IEEE Transactions on Information Theory, **52** (2006), 489–509.

[8] E. J. Candès and M. Wakin, *An introduction to compressive sampling*, IEEE Signal Processing Magazine, **2** (2008), 21–30.

[9] E. J. Candès and J. Romberg, The code package ℓ_1-magic : `http://www.acm.caltech.edu/l1magic`.

[10] R. Chellappa, P. Sinha and P. J. Phillips, *Face recognition by computers and humans*, Computer, **2** (2010), 46–55.

[11] I. Dagher and R. Nachar, *Face recognition using IPCA-ICA algorithm*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **6** (2006), 996–1000.

[12] D. L. Donoho, *Compressed sensing*, IEEE Transactions on Information Theory, **52** (2006), 1289–1306.

[13] M. A. T. Figueiredo, R. D. Nowak and S. J. Wright, *Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems*, Selected Topics in IEEE Journal of Signal Processing, **4** (2007), 586–597.

[14] J.-J. Fuchs, *On sparse representations in arbitrary redundant bases*, IEEE Transactions on Information Theory, **50** (2004), 1341–1344.

[15] J.-J. Fuchs, *Recovery of exact sparse representations in the presence of noise*, Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'04), **2** (2004), II-533–II-536.

[16] J.-J. Fuchs, *Fast implementation of a ℓ_1-ℓ_1 regularized sparse representations algorithm*, Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'09), (2009), 3329–3332.

[17] X. He, S. Yan, Y. Hu, P. Niyogi and H. Zhang, *Face recognition using laplacianfaces*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **3** (2005), 328–340.

[18] K. Koh, S.-J. Kim and S. Boyd, The code package ℓ_1_ℓ_s : `http://www.stanford.edu/~boyd/l1_ls`.

[19] D. D. Lee and H. S. Seung, *Learning the parts of objects by non-negative matrix factorization*, Nature, **6755** (1999), 788–791.

[20] S. Z. Li, X. Hou, H. Zhang and Q. Cheng, *Learning spatially localized, parts-based representation*, Proceedings of 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01), **1** (2001), I-207–I-212.

[21] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min and W. Worek, *Overview of the face recognition grand challenge*, Proceedings of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), **1** (2005), 947–954.

[22] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer and W. Worek, *Preliminary face recognition grand challenge results*, The 7^{th} International Conference on Automatic Face and Gesture Recognition (AFGR'06), (2006), 15–24.

[23] J. Lu, K. N. Plataniotis and A. N. Venetsanopoulos, *Face recognition using LDA-based algorithms*, IEEE Transactions on Neural Networks, **1** (2003), 195–200.

[24] O. L. Mangasarian and R. R. Meyer, *Nonlinear perturbation of linear programs*, SIAM Journal on Control and Optimization, **17** (1979), 745–752.

[25] A. M. Martinez and A. C. Kak, *PCA versus LDA*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **2** (2001), 228–233.

[26] A. J. O'Toole, P. J. Phillips and A. Narvekar, FRVT 2002 evaluation report, NISTIR TR-6965, 2003.

[27] P. J. Phillips, W. T. Scruggs, A. J. O'Toole, P. J. Flynn, K. W. Bowyer, C. L. Schott and M. Sharpe, *FRVT 2006 and ICE 2006 large-scale experimental results*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **5** (2010), 831–846.

[28] M. Turk and A. Pentland, *Eigenfaces for recognition*, Journal of Cognitive Neuroscience, **1** (1991), 71–86.

[29] H.-Y. Wang and X.-J. Wu, *Weighted PCA space and its application in face recognition*, Proceedings of 2005 International Conference on Machine Learning and Cybernetics, **7** (2005), 4522–4527.

[30] X.-G. Wang and X.-O. Tang, *A unified framework for subspace face recognition*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **9** (2004), 1222–1228.

[31] Y. Wang, G. Zhou, L. Caccetta and W. Liu, *An alternative Lagrange-dual based algorithm for sparse signal reconstruction*, IEEE Transactions on Signal Processing, **4** (2011), 1895–1901.

[32] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry and Yi Ma, *Robust face recognition via sparse representation*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **2** (2009), 210–227.

[33] S. J. Wright, R. D. Nowak and M. Figueiredo, *Sparse reconstruction by separable approximation*, IEEE Transactions on Signal Processing, **57** (2009), 2479–2493.

[34] J. Yang and Y. Zhang, *Alternating direction algorithms for ℓ_1 problems in compressive sensing*, SIAM Journal on Scientific Computing, **33** (2011), 250–278.

[35] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang and S. Lin, *Graph embedding and extensions: A general framework for dimensionality reduction*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **1** (2007), 40–51.

[36] W. Zhao, R. Chellappa, P. J. Phillips and A. Rosenfeld, *Face recognition: A literature survey*, ACM Computing Surveys, **4** (2003), 399–458.

Received February 2011; 1st revision July 2011; final revision July 2011.

E-mail address: qiuhuining@ieee.org
E-mail address: cocochen669930@hotmail.com
E-mail address: w.liu@curtin.edu.au
E-mail address: g.zhou@curtin.edu.au
E-mail address: wyiju@hotmail.com
E-mail address: stsljh@mail.sysu.edu.cn