SIViP (2015) 9:1601–1612
DOI 10.1007/s11760-014-0617-8

ORIGINAL PAPER

Fast linearized alternating direction method of multipliers for the augmented ℓ_1-regularized problem

Zhen-Zhen Yang · Zhen Yang

Received: 25 June 2013 / Revised: 10 January 2014 / Accepted: 15 January 2014 / Published online: 29 January 2014

Abstract The problem of reconstructing a sparse signal from an underdetermined linear system captures many applications. And it has been shown that this NP-hard problem can be well approached via heuristically solving a convex relaxation problem where the ℓ_1-norm is used to induce sparse structure. However, this convex relaxation problem is nonsmooth and thus not tractable in general, besides, this problem has a lower convergence rate. For these reasons, a signal reconstruction algorithm for solving the augmented ℓ_1-regularized problem is proposed in this paper. The separable structure of the new model enables us to solve the involved subproblems more efficiently by splitting the augmented Lagrangian function. Hence, an implementable numerical algorithm which called fast linearized alternating direction method of multipliers (FLADMM) is proposed to solve this novel model. Our experimental results show that the FLADMM method yields a higher peak signal-to-noise ratio reconstructed signal as well as a faster convergence rate at the same sampling rate as compared to the linearized Bregman method (LBM), the fast linearized Bregman iteration (FLBI) algorithm and the fast alternating direction method of multipliers (FADMM). Besides, this method is more robust than the LBM, FLBI and the FADMM algorithms at the same noise level.

Z.-Z. Yang (✉)
College of Communication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
e-mail: 2011010101@njupt.edu.cn

Z.-Z. Yang · Z. Yang
Key Lab of Broadband Wireless Communication and Sensor Network Technology, Ministry of Education, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

Keywords Fast linearized alternating direction method of multipliers · Alternating direction method of multipliers · The augmented ℓ_1 -regularized problem · Compressed sensing · Signal reconstruction

1 Introduction

Recently, the problem of reconstructing a sparse signal has received tremendous attention from researchers and engineers, particularly those in the areas of compressed sensing (CS) [1–5], machine learning [6–8] and statistics [9–11]. The fundamental problem of sparse signal reconstruction is to find the signal with (nearly) fewest nonzero entries from an underdetermined linear system.

In CS, let $\mathbf{x} \in \mathbb{R}^N$ be a signal vector, and it can be represented with an orthogonal basis $\mathbf{\Psi} = \{\varphi_i | i = 1, 2, \cdots, N\}$; that is,

$$\mathbf{x} = \sum_{i=1}^{N} \theta_i \varphi_i = \mathbf{\Psi\Theta} \tag{1}$$

where $\theta_i = \langle \mathbf{x}, \varphi_i \rangle$ denotes the sparse coefficients. The signal $\mathbf{x}$ has a K-sparse representation if $\|\mathbf{\Theta}\|_0 = K$ (where $\|\mathbf{\Theta}\|_0$ denotes the number of nonzero entries of $\mathbf{\Theta}$, and $K \ll N$) and $\mathbf{x}$ is a compressible signal if many coefficients in the vector $\mathbf{\Theta}$ are small and can be neglected without seriously degrading the signal [4].

We measure $\mathbf{\Theta}$ directly through a matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ with $M < N$, and then, we can get an underdetermined linear system as follows,

$$\mathbf{y} = \mathbf{A}\Theta \tag{2}$$

where $\mathbf{y} \in \mathbb{R}^M$ is the measurement and $\mathbf{A}$ is the measurement matrix. In this paper, the random Gaussian matrix is chosen as the measurement matrix $\mathbf{A}$.

The aim of a basis pursuit (BP) problem [12] is to find a sparse signal $\boldsymbol{\Theta} \in \mathbb{R}^N$ by solving the constrained minimization problem as follows,

$$\min_{\boldsymbol{\Theta}} \|\boldsymbol{\Theta}\|_1 \text{ s.t. } \mathbf{y} = \mathbf{A}\boldsymbol{\Theta} \tag{3}$$

and then, the signal $\mathbf{x}$ can be reconstructed through the orthogonal basis $\boldsymbol{\Psi}$ by the Eq. (1). The BP problem (3) can be transformed into a linear problem and then solved by conventional linear programming (LP) methods [12]. Another way to solve this problem is by using an entropy functional to approximate the ℓ_1 penalty function [13]. However, this approximated method only can obtain the approximated solution of the BP problem (3).

For a signal with noise, a variant of (3) is the following ℓ_1-regularized problem [14],

$$\min_{\boldsymbol{\Theta}} \frac{1}{2} \|\mathbf{A}\Theta - \mathbf{y}\|_2^2 + \tau \|\boldsymbol{\Theta}\|_1 \tag{4}$$

where $\tau \in (0, \infty)$ is the regularized variable, $\|\bullet\|_1$ denotes the ℓ_1-norm, and $\|\bullet\|_2$ denotes the ℓ_2-norm, $\|\boldsymbol{\Theta}\|_1$ is the regularization. In most signal recovery problems, nonsmooth regularizations such as the total variation (TV) regularization [15], the filtered variation (FV) regularization [16] and the hybrid regularization [17] are also popular and powerful choices. Problem in (4) can be solved by the alternating direction method of multipliers (ADMM) [18,19] and the fast ADMM (FADMM) [20].

The nonsmooth objective functions in BP problem (3) pose numerical challenges. Lai et al. in [21] augmented them by adding $\frac{1}{2\rho} \|\boldsymbol{\Theta}\|_2^2$, where $\rho \in (0, \infty)$ is the variable. They argued that minimizing the augmented objective $\|\boldsymbol{\Theta}\|_1 + \frac{1}{2\rho} \|\boldsymbol{\Theta}\|_2^2$ leads to fast numerical algorithms because not only accurate solutions can be obtained by using a sufficiently large, yet not excessive large, value of ρ, but the Lagrange dual problems are also continuously differentiable and subject to gradient-based acceleration techniques.

To find a sparse signal $\boldsymbol{\Theta}$, Lai et al. in [21] and Cai et al. in [22] solved the augmented BP problem as follows,

$$\min_{\boldsymbol{\Theta}} \|\boldsymbol{\Theta}\|_1 + \frac{1}{2\rho} \|\boldsymbol{\Theta}\|_2^2 \text{ s.t. } \mathbf{y} = \mathbf{A}\boldsymbol{\Theta} \tag{5}$$

This problem is the augmented model of the problem (3) and can be solved by the linearized Bregman method (LBM) [22], the accelerated linearized Bregman method (ALBM) [23] and the fast linearized Bregman iteration (FLBI) algorithm [24]. For a signal with noise, inspired by the idea of the ℓ_1-regularized model (4) and the augmented BP problem (5), we propose to find a sparse vector by solving the augmented ℓ_1-regularized problem as follows,

$$\min_{\boldsymbol{\Theta}} \frac{1}{2} \|\mathbf{A}\Theta - \mathbf{y}\|_2^2 + \gamma \|\boldsymbol{\Theta}\|_2^2 + \tau \|\boldsymbol{\Theta}\|_1 \tag{6}$$

where $\tau, \gamma \in (0, \infty)$ are the variables. This augmented ℓ_1-regularized problem (6) is also known as the elastic net problem, which is a convex combination of the ℓ_1-regularized problem ($\gamma = 0$) (4) and the ridge regression problem ($\tau = 0$) [21,25]. When $\tau, \gamma \in (0, \infty)$, this problem has the characteristics of both the ℓ_1-regularized problem and the ridge regression problem. It has been shown that the augmented ℓ_1-regularized model (6) outperforms the ℓ_1-regularized problem (4) on reported real-world regression problems [25].

Two contributions are made in this paper: (i) a novel sparse signal reconstruction model which called the augmented ℓ_1-regularized model for the incomplete and noisy measurement is proposed to extend the ℓ_1-regularized problem (4) and the augmented BP problem (5) into a more general setting; (ii) the FLADMM (fast linearized ADMM) which is inspired by the idea of the fast method [26] and the linearized method [27] is proposed to solve this new reconstruction model efficiently. The LBM and the FLBI algorithms are equivalent to solve the Lagrangian function of the augmented BP problem (5) and then update the multiplier [22], and the FADMM is equivalent to solve the Lagrangian function of the ℓ_1-regularized problem (4) and then update the multiplier [20], while the FLADMM is actually to solve the augmented Lagrangian function of the equality constrained optimization model of the problem (6, and then update the multiplier. Nocedal et al. in [28] introduced that the augmented Lagrangian function differs from the Lagrangian function by the presence of the squared terms, while it differs from the quadratic penalty function in the presence of the summation term involving the multiplier. In this sense, augmented Lagrangian function is a combination of the Lagrangian and quadratic penalty functions, and it generally has a more accurate solution than that of the Lagrangian function method [28]. Moreover, the augmented ℓ_1-regularized model (6) outperforms the ℓ_1-regularized problem (4) and the augmented BP problem (5) on reported real-world regression problems [22,25]. Therefore, the FLADMM has a better performance than that of the LBM, FLBI and FADMM algorithms. Besides, the fast method and the linearized method are used to accelerate the convergence rate, so the FLADMM has a faster convergence rate than the LBM, FLBI and FADMM algorithms do. Our experimental results demonstrate that the FLADMM yields a higher peak signal-to-noise ratio (PSNR) reconstructed signal as well as a faster convergence rate at the same sampling rate as compared to the LBM, FLBI and FADMM algorithms. Besides, this method is more robust than the LBM, FLBI and FADMM algorithms at the same noise level.

The rest of this paper is organized as follows. In Sect. 2, we propose the FLADMM algorithm to reconstruct the sparse

coefficients from the novel augmented ℓ_1-regularized problem. Section 3 presents extensive numerical results to evaluate the performance of the proposed reconstruction algorithm in comparison with LBM, FLBI and FADMM algorithms. Finally, concluding remarks are provided in Sect. 4.

2 Proposed reconstruction algorithm

In this section, a new reconstruction algorithm called FLADMM algorithm is proposed to solve the augmented ℓ_1-regularized problem. To this end, we first briefly review the known ADMM algorithm.

2.1 Alternating direction method of multipliers

Eckstein et al. in [29] combined the method of multipliers (MM) and the alternating direction method (ADM) to develop the alternating direction method of multipliers (ADMM), which thus has the virtue of both methods. The ADMM is based on a variable-splitting technique and a method of multipliers framework. Besides, it is preferable to other methods, because it decouples many difficult problems into simple subproblems that can be easily solved.

We consider a constrained optimization problem as described by

$$\begin{aligned} &\min_{\mathbf{u},\mathbf{v}} f(\mathbf{u}) + g(\mathbf{v}) \\ &s.t. \mathbf{Cu} + \mathbf{Dv} = \mathbf{b} \end{aligned} \tag{7}$$

where $f(\mathbf{u})$ and $g(\mathbf{v})$ are convex functions, $\mathbf{C}$ and $\mathbf{D}$ are given matrices, $\mathbf{b}$ is a given vector, and $\mathbf{u}$ and $\mathbf{v}$ are unknown variables.

The augmented Lagrangian function based on the constrained optimization problem (7) is given as follows,

$$\begin{aligned} \Gamma(\mathbf{u}, \mathbf{v}, \boldsymbol{\lambda}) = \mathrm{f}(\mathbf{u}) + g(\mathbf{v}) - \boldsymbol{\lambda}^T (\mathbf{Cu} + \mathbf{Dv} - \mathbf{b}) \\ + \frac{\mu}{2} \|\mathbf{Cu} + \mathbf{Dv} - \mathbf{b}\|_2^2 \end{aligned} \tag{8}$$

where $\mu \in (0, \infty)$ is a penalty variable, and $\boldsymbol{\lambda} \in \mathbb{R}^M$ is a Lagrangian multiplier. For given $\boldsymbol{\lambda}_k$, we can get

$$(\mathbf{u}_{k+1}, \mathbf{v}_{k+1}) = \arg\min_{\mathbf{u},\mathbf{v}} \Gamma(\mathbf{u}, \mathbf{v}, \boldsymbol{\lambda}_k) \tag{9}$$

Then, the Lagrange multiplier can be obtained from the following update,

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k - \mu (\mathbf{Cu}_{k+1} + \mathbf{Dv}_{k+1} - \mathbf{b}) \tag{10}$$

However, an accurate and joint optimization of $\mathbf{u}$ and $\mathbf{v}$ can be costly. In contrast, ADMM utilizes a separable structure in optimization problem (7) and replaces the joint least squares by two simpler subproblems. Specifically, ADMM minimizes $\Gamma(\mathbf{u}, \mathbf{v}, \boldsymbol{\lambda}_k)$ with respect to $\mathbf{u}$ and $\mathbf{v}$ separately via a nonlinear Gauss-Seidel type iteration. After just one alternating least squares with respect to $\mathbf{u}$ and $\mathbf{v}$, the multiplier $\boldsymbol{\lambda}$ is updated immediately. In short, given $(\mathbf{v}_k, \boldsymbol{\lambda}_k)$, ADMM iterates as follows,

$$\begin{cases} \mathbf{u}_{k+1} = \arg\min_{\mathbf{u}} \Gamma(\mathbf{u}, \mathbf{v}_k, \boldsymbol{\lambda}_k) \\ \mathbf{v}_{k+1} = \arg\min_{\mathbf{v}} \Gamma(\mathbf{u}_{k+1}, \mathbf{v}, \boldsymbol{\lambda}_k) \\ \boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k - \mu (\mathbf{Cu}_{k+1} + \mathbf{Dv}_{k+1} - \mathbf{b}) \end{cases} \tag{11}$$

The above ADMM proposed to solve the constrained optimization problem (7) can be summarized as follows.

Algorithm 1: ADMM algorithm

(1) Initialization: Given $\mu > 0$, starting points $\mathbf{v}_0$ and $\boldsymbol{\lambda}_0$, and iteration index $k = 0$;
(2) Update $\mathbf{u}$: $\mathbf{u}_{k+1} = \arg\min_{\mathbf{u}} \Gamma(\mathbf{u}, \mathbf{v}_k, \boldsymbol{\lambda}_k)$;
(3) Update $\mathbf{v}$: $\mathbf{v}_{k+1} = \arg\min_{\mathbf{v}} \Gamma(\mathbf{u}_{k+1}, \mathbf{v}, \boldsymbol{\lambda}_k)$;
(4) Update $\boldsymbol{\lambda}$: $\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k - \mu (\mathbf{Cu}_{k+1} + \mathbf{Dv}_{k+1} - \mathbf{b})$;
(5) The iteration is terminated if the termination condition is satisfied; otherwise, set $k = k + 1$ and return to step 2).

We terminate the ADMM algorithm when the relative change in the $\mathbf{u}$ vector between two consecutive iterations becomes small enough.

2.2 Fast linearized alternating direction method of multipliers

Yang et al. in [18] studied the use of alternating direction algorithms for the ℓ_1-regularized problem arising from sparse solution reconstruction in CS. Inspired by the work of Yang et al. in [18], we will present the FLADMM for solving the augmented ℓ_1-regularized problem in this section.

Variable splitting is a very simple procedure that involves creating a new auxiliary variable $\boldsymbol{\Xi} \in \mathbb{R}^N$ to serve as the argument of $\|\boldsymbol{\Theta}\|_1$ under the constraint $\boldsymbol{\Theta} = \boldsymbol{\Xi}$. Then, the augmented ℓ_1-regularized problem (6) is clearly equivalent to the constrained optimization problem as follows.

$$\min_{\boldsymbol{\Theta},\boldsymbol{\Xi}} \frac{1}{2} \|\mathbf{A}\Theta - \mathbf{y}\|_2^2 + \gamma \|\boldsymbol{\Theta}\|_2^2 + \tau \|\boldsymbol{\Xi}\|_1 \text{ s.t. } \boldsymbol{\Theta} = \boldsymbol{\Xi} \tag{12}$$

In problem (12), $\frac{1}{2} \|\mathbf{A}\Theta - \mathbf{y}\|_2^2 + \gamma \|\boldsymbol{\Theta}\|_2^2$ is strongly convex and has Lipschitz continuous gradient; $\tau \|\boldsymbol{\Xi}\|_1$ is convex and nonsmooth. Note that if $\gamma = 0$, then $\frac{1}{2} \|\mathbf{A}\Theta - \mathbf{y}\|_2^2 + \gamma \|\boldsymbol{\Theta}\|_2^2$ may not be strongly convex if the matrix $\mathbf{A}$ does not have full column rank. In many applications, this is indeed the case since the number of measurement $\mathbf{y}$ is usually smaller than their dimensions (i.e., $M < N$). However, the parameter $\gamma > 0$ guarantees the strong convexity of $\frac{1}{2} \|\mathbf{A}\Theta - \mathbf{y}\|_2^2 +$

$\gamma \|\mathbf{\Theta}\|_2^2 + \tau \|\mathbf{\Xi}\|_1$ and hence the global linear convergence of FLADMM algorithm when applied to problem (12).

The augmented Lagrangian function of the problem (12) is given as follows.

$$\Gamma(\mathbf{\Theta}, \mathbf{\Xi}, \boldsymbol{\lambda}) = \frac{1}{2}\|\mathbf{A}\Theta - \mathbf{y}\|_2^2 + \gamma \|\mathbf{\Theta}\|_2^2 + \tau \|\mathbf{\Xi}\|_1 - \boldsymbol{\lambda}^T(\mathbf{\Theta} - \mathbf{\Xi}) + \frac{\mu}{2}\|\mathbf{\Theta} - \mathbf{\Xi}\|_2^2 \quad (13)$$

Now, we apply the ADMM algorithm to optimization problem (12). First, for given $\mathbf{\Theta}_k$ and $\boldsymbol{\lambda}_k$, we can get

$$\begin{aligned}\bar{\mathbf{\Xi}}_{k+1} &= \arg\min_{\mathbf{\Xi}} \Gamma(\mathbf{\Theta}_k, \mathbf{\Xi}, \boldsymbol{\lambda}_k) = \arg\min_{\mathbf{\Xi}} \tau \|\mathbf{\Xi}\|_1 \\ &\quad -\boldsymbol{\lambda}_k^{\mathrm{T}}(\mathbf{\Theta}_k - \mathbf{\Xi}) + \frac{\mu}{2}\|\mathbf{\Theta}_k - \mathbf{\Xi}\|_2^2 \\ &= \arg\min_{\mathbf{\Xi}} \tau \|\mathbf{\Xi}\|_1 + \frac{\mu}{2}\left\|\mathbf{\Xi} - \left(\mathbf{\Theta}_k - \frac{\boldsymbol{\lambda}_k}{\mu}\right)\right\|_2^2 \\ &= \arg\min_{\mathbf{\Xi}} \frac{\tau}{\mu}\|\mathbf{\Xi}\|_1 \\ &\quad +\frac{1}{2}\left\|\mathbf{\Xi} - \left(\mathbf{\Theta}_k - \frac{\boldsymbol{\lambda}_k}{\mu}\right)\right\|_2^2 \\ &= \mathrm{soft}\left(\mathbf{\Theta}_k - \frac{\boldsymbol{\lambda}_k}{\mu}, \frac{\tau}{\mu}\right)\end{aligned} \quad (14)$$

Let $\mathbf{d}_k = \frac{\boldsymbol{\lambda}_k}{\mu}$, and we can get

$$\bar{\mathbf{\Xi}}_{k+1} = \mathrm{soft}\left(\mathbf{\Theta}_k - \mathbf{d}_k, \frac{\tau}{\mu}\right) \quad (15)$$

where $\mathrm{soft}(\bullet, Th) = \mathrm{sgn}(\bullet)\max\{|\bullet| - Th, 0\}$ is the soft thresholding function with threshold Th, and $\mathrm{sgn}(\bullet)$ is the sign function. A key ingredient of (15) is the so-called soft thresholding function, which is the Moreau proximal mapping [15,30] of the regularization $\|\mathbf{\Theta}\|_1$.

In order to accelerate the convergence of the above iteration, the fast method of Beck et al. [26] is applied to solve this problem. In this method, $\mathbf{\Xi}$ is updated again as follows,

$$\mathbf{\Xi}_{k+1} = \bar{\mathbf{\Xi}}_{k+1} + \left(\frac{t_k - 1}{t_{k+1}}\right)(\bar{\mathbf{\Xi}}_{k+1} - \bar{\mathbf{\Xi}}_k) \quad (16)$$

where $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$ with $t_0 = 1$, and $\frac{t_k - 1}{t_{k+1}} \in [0, 1]$ is the step length.

Second, for given $\mathbf{\Xi}_{k+1}$ and $\boldsymbol{\lambda}_k$, the minimized $\bar{\mathbf{\Theta}}_{k+1}$of problem (12) with respect to $\mathbf{\Theta}$ is given by

$$\begin{aligned}\bar{\mathbf{\Theta}}_{k+1} &= \arg\min_{\mathbf{\Theta}} \Gamma(\mathbf{\Theta}, \mathbf{\Xi}_{k+1}, \boldsymbol{\lambda}_k) \\ &= \arg\min_{\mathbf{\Theta}} \frac{1}{2}\|\mathbf{A}\Theta - \mathbf{y}\|_2^2 + \gamma\|\mathbf{\Theta}\|_2^2 - \boldsymbol{\lambda}_k^{\mathrm{T}}(\mathbf{\Theta} - \mathbf{\Xi}_{k+1}) \\ &\quad +\frac{\mu}{2}\|\mathbf{\Theta} - \mathbf{\Xi}_{k+1}\|_2^2 \\ &= \arg\min_{\mathbf{\Theta}} \frac{1}{2}\|\mathbf{A}\Theta - \mathbf{y}\|_2^2 + \gamma\|\mathbf{\Theta}\|_2^2 \\ &\quad +\frac{\mu}{2}\left\|\mathbf{\Theta} - \left(\mathbf{\Xi}_{k+1} + \frac{\boldsymbol{\lambda}_k}{\mu}\right)\right\|_2^2 \\ &= \arg\min_{\mathbf{\Theta}} \frac{1}{2}\|\mathbf{A}\Theta - \mathbf{y}\|_2^2 + \gamma\|\mathbf{\Theta}\|_2^2 \\ &\quad +\frac{\mu}{2}\|\mathbf{\Theta} - (\mathbf{\Xi}_{k+1} + \mathbf{d}_k)\|_2^2\end{aligned} \quad (17)$$

So we can get

$$\left((2\gamma + \mu)\mathbf{I} + \mathbf{A}^{\mathrm{T}}\mathbf{A}\right)\bar{\mathbf{\Theta}}_{k+1} = \mathbf{A}^{\mathrm{T}}\mathbf{y} + \mu(\mathbf{\Xi}_{k+1} + \mathbf{d}_k) \quad (18)$$

Since $\mathbf{A}$ is a random Gaussian matrix, $\mathbf{A}^{\mathrm{T}}\mathbf{A} \neq \mathbf{I}$ (where $\mathbf{I}$ is identity matrix), it is costly to solve (18). Inspired by the work of Yang et al. in [27], we use a linearized strategy for the quadratic term $\frac{1}{2}\|\mathbf{A}\mathbf{\Theta} - \mathbf{y}\|_2^2$ to accelerate the convergence of the FLADMM. With this linearization, the quadratic term can be approximated by

$$\frac{1}{2}\|\mathbf{A}\mathbf{\Theta} - \mathbf{y}\|_2^2 \approx \frac{1}{2}\|\mathbf{A}\mathbf{\Theta}_k - \mathbf{y}\|_2^2 + \langle \mathbf{g}_k\mathbf{\Theta} - \mathbf{\Theta}_k\rangle + \frac{1}{2\alpha}\|\mathbf{\Theta} - \mathbf{\Theta}_k\|_2^2 \quad (19)$$

where $\alpha > 0$ is a proximal parameter, $\langle \mathbf{g}_k, \mathbf{\Theta} - \mathbf{\Theta}_k\rangle$ denotes the inner product of the vectors $\mathbf{g}_k$ and the vector $\mathbf{\Theta} - \mathbf{\Theta}_k$, and

$$\mathbf{g}_k = \mathbf{A}^{\mathrm{T}}(\mathbf{A}\Theta_k - \mathbf{y}) \quad (20)$$

is the gradient of $\frac{1}{2}\|\mathbf{A}\Theta - \mathbf{y}\|_2^2$ at $\mathbf{\Theta}_k$. Using (19) into (17) results in

$$\begin{aligned}\bar{\mathbf{\Theta}}_{k+1} &= \arg\min_{\mathbf{\Theta}} \langle \mathbf{g}_k, \mathbf{\Theta} - \mathbf{\Theta}_k\rangle + \frac{1}{2\alpha}\|\mathbf{\Theta} - \mathbf{\Theta}_k\|_2^2 \\ &\quad +\gamma\|\mathbf{\Theta}\|_2^2 \\ &\quad +\frac{\mu}{2}\left\|\mathbf{\Theta} - \left(\mathbf{\Xi}_{k+1} + \frac{\boldsymbol{\lambda}_k}{\mu}\right)\right\|_2^2 \\ &= \arg\min_{\mathbf{\Theta}} \frac{1}{2\alpha}\|\mathbf{\Theta} - (\mathbf{\Theta}_k - \alpha\mathbf{g}_k)\|_2^2 + \gamma\|\mathbf{\Theta}\|_2^2 \\ &\quad +\frac{\mu}{2}\|\mathbf{\Theta} - (\mathbf{\Xi}_{k+1} + \mathbf{d}_k)\|_2^2\end{aligned} \quad (21)$$

By solving the above optimization problem, we can obtain the following closed form solution for $\bar{\mathbf{\Theta}}_{k+1}$.

$$\begin{aligned}\bar{\boldsymbol{\Theta}}_{k+1} &= \frac{1}{1+2\gamma\alpha+\mu\alpha}\left(\boldsymbol{\Theta}_k - \alpha\mathbf{g}_k + \mu\alpha\boldsymbol{\Xi}_{k+1} + \alpha\boldsymbol{\lambda}_k\right)\\ &= \frac{1}{1+2\gamma\alpha+\mu\alpha}\Big(\boldsymbol{\Theta}_k - \alpha\mathbf{A}^{\mathrm{T}}\left(\mathbf{A}\Theta_k - \mathbf{y}\right)\\ &\quad +\mu\alpha\left(\boldsymbol{\Xi}_{k+1} + \mathbf{d}_k\right)\Big) \qquad (22)\\ &= \frac{1}{1+\alpha\left(2\gamma+\mu\right)}\Big(\left(\mathbf{I} - \alpha\mathbf{A}^{\mathrm{T}}\mathbf{A}\right)\boldsymbol{\Theta}_k\\ &\quad +\alpha\left(\mathbf{A}^{\mathrm{T}}\mathbf{y} + \mu\left(\boldsymbol{\Xi}_{k+1} + \mathbf{d}_k\right)\right)\Big)\end{aligned}$$

Let $\omega = 1+\alpha\left(2\gamma+\mu\right)$, $\mathbf{P} = \mathbf{I} - \alpha\mathbf{A}^{\mathrm{T}}\mathbf{A}$, and $\mathbf{q} = \mathbf{A}^{\mathrm{T}}\mathbf{y}$, then we can get

$$\bar{\boldsymbol{\Theta}}_{k+1} = \frac{1}{\omega}\left(\mathbf{P}\Theta_k + \alpha\left(\mathbf{q} + \mu\left(\boldsymbol{\Xi}_{k+1} + \mathbf{d}_k\right)\right)\right) \qquad (23)$$

Inspired by the idea of the fast method in [26], $\boldsymbol{\Theta}$ is updated again as follows,

$$\boldsymbol{\Theta}_{k+1} = \bar{\boldsymbol{\Theta}}_{k+1} + \left(\frac{t_k - 1}{t_{k+1}}\right)\left(\bar{\boldsymbol{\Theta}}_{k+1} - \bar{\boldsymbol{\Theta}}_k\right) \qquad (24)$$

Finally, we update the Lagrange multiplier $\mathbf{d}$ as follows,

$$\mathbf{d}_{k+1} = \mathbf{d}_k - \left(\boldsymbol{\Theta}_{k+1} - \boldsymbol{\Xi}_{k+1}\right) \qquad (25)$$

The FLADMM proposed above to solve optimization problem (12) can be summarized as follows.

Algorithm 2: FLADMM algorithm

(1) Initialization: Given $\mu > 0, \gamma > 0, \alpha > 0, \tau > 0$, starting points $\mathbf{d}_0, \boldsymbol{\Theta}_0, \bar{\boldsymbol{\Theta}}_0, \bar{\boldsymbol{\Xi}}_0, t_0 = 1$, and iteration index $k = 0$;
(2) Compute ω; $\omega = 1+\alpha\left(2\gamma+\mu\right)$;
(3) Compute matrix $\mathbf{P}$; $\mathbf{P} = \mathbf{I} - \alpha\mathbf{A}^{\mathrm{T}}\mathbf{A}$;
(4) Compute vector $\mathbf{q}$; $\mathbf{q} = \mathbf{A}^{\mathrm{T}}\mathbf{y}$;
(5) Update t; $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$;
(6) Update $\boldsymbol{\Xi}$: $\bar{\boldsymbol{\Xi}}_{k+1}$=soft$(\boldsymbol{\Theta}_k - \mathbf{d}_k, \frac{\tau}{\mu})$;
(7) Update $\boldsymbol{\Xi}$ again: $\boldsymbol{\Xi}_{k+1} = \bar{\boldsymbol{\Xi}}_{k+1} + \left(\frac{t_k-1}{t_{k+1}}\right)\left(\bar{\boldsymbol{\Xi}}_{k+1} - \bar{\boldsymbol{\Xi}}_k\right)$;
(8) Update $\boldsymbol{\Theta}$: $\bar{\boldsymbol{\Theta}}_{k+1} = \frac{1}{\omega}\left(\mathbf{P}\Theta_k + \alpha\left(\mathbf{q} + \mu\left(\boldsymbol{\Xi}_{k+1} + \mathbf{d}_k\right)\right)\right)$;
(9) Update $\boldsymbol{\Theta}$ again: $\boldsymbol{\Theta}_{k+1} = \bar{\boldsymbol{\Theta}}_{k+1} + \left(\frac{t_k-1}{t_{k+1}}\right)\left(\bar{\boldsymbol{\Theta}}_{k+1} - \bar{\boldsymbol{\Theta}}_k\right)$
(10) Update $\mathbf{d}$: $\mathbf{d}_{k+1} = \mathbf{d}_k - \left(\boldsymbol{\Theta}_{k+1} - \boldsymbol{\Xi}_{k+1}\right)$;
(11) The iteration is terminated if the termination condition is satisfied; otherwise, set $k = k+1$ and return to step 5).

The termination condition in the above algorithm will be discussed in Sect. 3.

The computational complexity of the FLADMM algorithm is dominated by step 8 at each iteration, which is of $O\left(N^2\right)$, whereas the computational complexities of step 7 and step 9 are both $O\left(N\right)$, and the computational complexities of steps 5, 6 and 10 are all $O\left(1\right)$ only. While the computational complexities of the LBM, FLBI and FADMM algorithms are all $O\left(MN^2\right)$. The advantage of the FLADMM algorithm is that the accelerated strategy of the fast method and the linearization method is used for it. For these reasons, the FLADMM has better performance, more robust as well as a faster convergence rate as compared to LBM, FLBI and FADMM algorithms.

When FLADMM is applied to solve the augmented ℓ_1-regularized problem (6), both of the resulting subproblems admit explicit solutions. FLADMM algorithm requires one soft thresholding projection and matrix-vector multiplications at each iteration, which can be carried out effectively. Moreover, this algorithm belongs to the classical ADMM framework. Hence, the convergence of the FLADMM method is ensured due to the convergence of the ADMM, which can be found in [29]. Since the FLADMM only requires one soft thresholding projection and matrix-vector multiplications at each iteration, besides, this algorithm is substantially accelerated according to the fast idea of Beck et al. in [26] and the linearized method of Yang et al. in [27], and it convergences faster than some other existing signal reconstruction algorithms such as LBM [22], FLBI [24] and FADMM [20] algorithms.

3 Experimental results and analysis

In this section, some experimental results are presented to evaluate the performance of the proposed FLADMM.

The image can be represented with a sparse vector by an orthogonal basis. Many researches have been shown that images in the real world are known to have a sparse representation in the discrete wavelet transform (DWT) domain [31–34], so we choose the DWT basis as the sparse orthogonal basis $\boldsymbol{\Psi}$ in this paper. Each of the same experiments is repeated for 10 times, and thus, the PSNR and run times are the averaged results of 10 experiments.

We terminate the FLADMM algorithm when the relative change in the sparse coefficient vector between two consecutive iterations becomes small enough, i.e.,

$$\frac{\|\boldsymbol{\Theta}_{k+1} - \boldsymbol{\Theta}_k\|_2}{\|\boldsymbol{\Theta}_k\|_2} < 10^{-3} \qquad (26)$$

A straightforward implementation of CS on 2D images recasts the 2D array-based problem as a 1D vector-based problem. Firstly, the test image of size $n \times n$ is arranged into a column vector of length $N = n^2$. Then, this vector is divided into n frames of size n, and we solve one frame at a time. The sampling rate of the test image is defined as $r = \frac{m}{n}$, where n is the frame size, and m is the dimension of the measurement of each frame [26–28].

Here, we take three different groups of images for testing: Lena (256×256), Peppers (256×256) and Man (512×512), and compare the FLADMM reconstruction algorithm with

LBM [22], FLBI [24] and FADMM [20] algorithms. The proposed FLADMM reconstruction algorithm is applied to these test images at the sampling rate $r = 0.5$. Experimental results on these test images are shown in Figs. 1, 2 and 3, respectively.

Figure 1a is the original Lena image, and Fig. 1b–e is the reconstructed Lena images obtained by the FLADMM, LBM, FLBI and FADMM algorithms, respectively. The images in Figs. 2 and 3 have similar situations. It is clear from Figs. 1, 2 and 3 that the reconstructed images of our proposed FLADMM have more detailed information and are much closer to the original image as compared with the LBM, FLBI and FADMM algorithms. Besides, in all these figures, image (b) looks much smoother and clearer than the other reconstructed images. In short, the proposed reconstruction algorithm performs better in human perception of

Fig. 1 Experimental results from different reconstruction algorithms for Lena image. **a** Original image **b** FLADMM reconstruction **c** LBM reconstruction **d** FLBI reconstruction **e** FADMM reconstruction

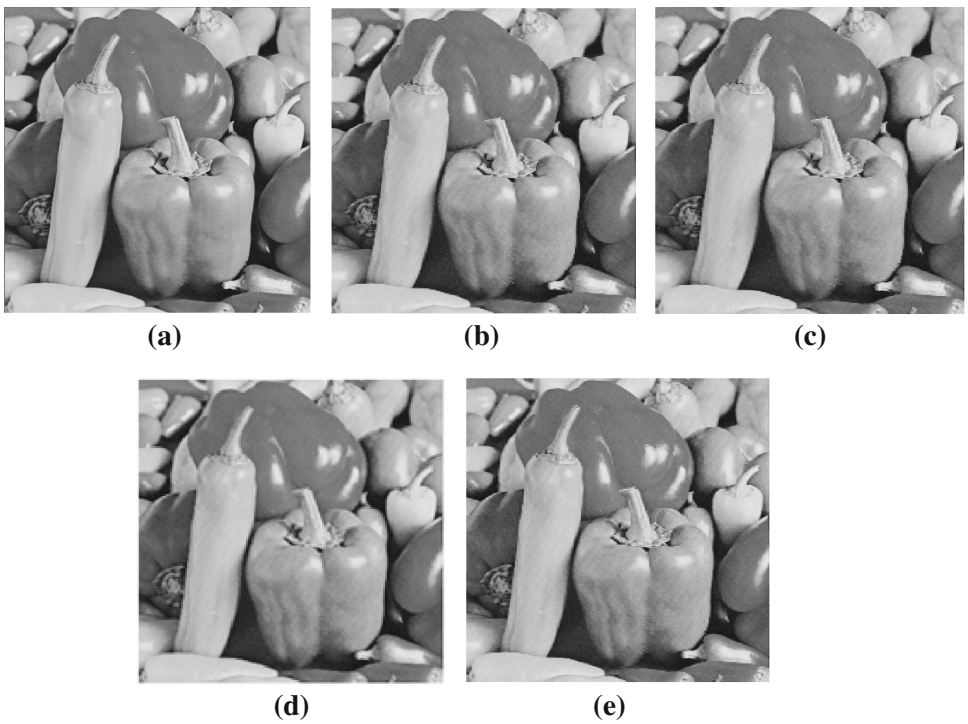

Fig. 2 Experimental results from different reconstruction algorithms for Peppers image. **a** Original image **b** FLADMM reconstruction **c** LBM reconstruction **d** FLBI reconstruction **e** FADMM reconstruction

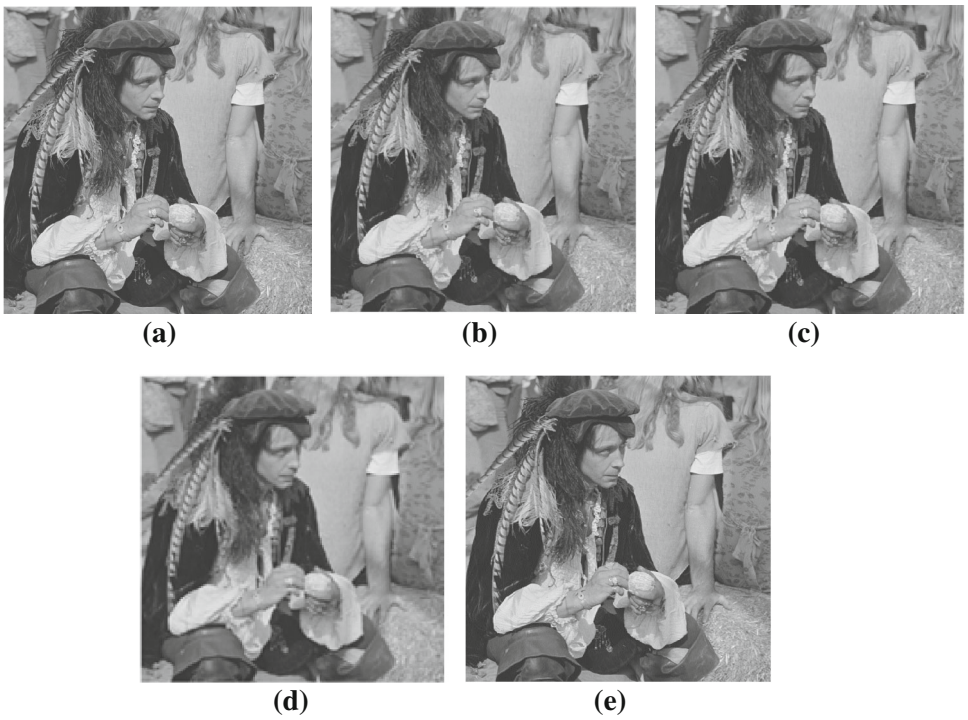

Fig. 3 Experimental results from different reconstruction algorithms for Man image. **a** Original image **b** FLADMM reconstruction **c** LBM reconstruction **d** FLBI reconstruction **e** FADMM reconstruction

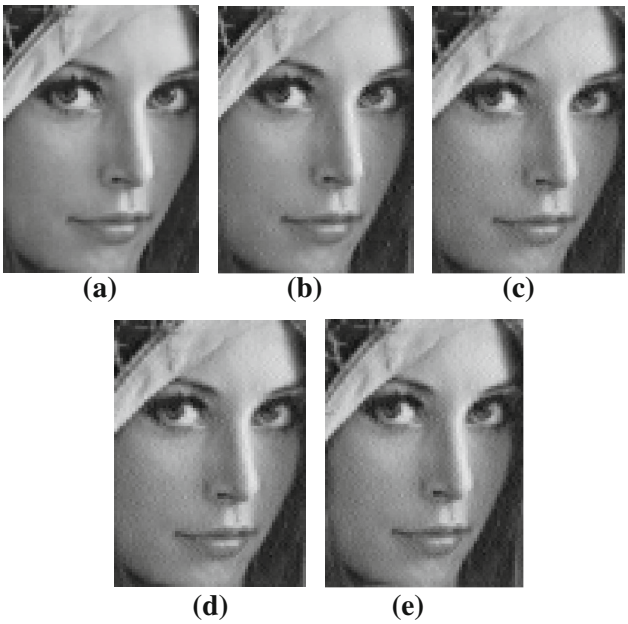

Fig. 4 Experimental results from different reconstruction algorithms for Lena's face image. **a** Original image **b** FLADMM reconstruction **c** LBM reconstruction **d** FLBI reconstruction **e** FADMM reconstruction

global information, which can enhance the definition of the reconstructed images greatly.

For easy observation, we take the Lena image as an example and magnify the face regions of the reconstruction results with different algorithms. The magnified face regions of the reconstruction results with different algorithms are shown in Fig. 4.

Table 1 The PSNR (dB) for Lena image

Algorithms	Sampling rate r				
	0.1	0.2	0.3	0.4	0.5
FLADMM	31.642	32.287	33.321	34.516	35.848
LBM [22]	31.563	32.168	32.789	33.452	34.397
FLBI [24]	31.565	32.174	32.791	33.456	34.405
FADMM [20]	31.561	32.157	32.723	33.403	34.289

It is clear from Fig. 4 that the reconstruction result obtained by FLADMM has better appearances than LBM, FLBI and FADMM algorithms. We also can see that the reconstruction result of the proposed FLADMM algorithm can maintain salient features of the face in the original Lena image and has less noises and artificial effects on the face.

For further comparison, two objective criteria, PSNR in dB and the run time in seconds, are used to measure the performance of the proposed algorithm for the Lena image (256×256).

The PSNR in dB of these methods to reconstruct Lena image at different sampling rates is listed in Table 1.

Table 1 gives the quantitative results of the FLADMM, LBM, FLBI and FADMM algorithms. It is clear that all these methods provide very good performance. The results also clearly state that, with increasing the sampling rate, the PSNR becomes higher for all these methods, that is, better quality reconstruction image can be obtained by taking more

Table 2 The run time (s) for Lena image

Algorithms	Sampling rate r				
	0.1	0.2	0.3	0.4	0.5
FLADMM	0.753	1.372	2.751	4.393	5.706
LBM [22]	1.376	2.564	4.963	8.876	17.385
FLBI [24]	0.815	1.574	2.945	4.872	6.987
FADMM [20]	1.068	1.851	3.418	5.342	6.824

measurements. Moreover, with increasing the sampling rate, the superiority of the FLADMM becomes more obvious. For example, when the sampling rate is $r = 0.5$, the PSNR of Lena image using our proposed FLADMM is 35.848 dB, while that produced by LBM, FLBI and FADMM algorithms is only 34.397, 34.405 and 34.289 dB, respectively. FLADMM has better performance at the same sampling rate than that of the LBM, FLBI and FADMM methods is because that the FLADMM is used for solving the augmented ℓ_1-regularized model (6), while the LBM and FLBI algorithms are both used for solving the augmented BP model (5), and the FADMM is used for solving the ℓ_1-regularized problem (4). It has been shown that the augmented ℓ_1-regularized model (6) outperforms the ℓ_1-regularized problem (4) and the augmented BP model (5) on reported real-world regression problems [22,25].

The run time in seconds required by the FLADMM, LBM, FLBI and FADMM methods to reconstruct Lena image at different sampling rates is listed in Table 2. In general, with increasing sampling rate, the run time increases for all these methods. However, the FLADMM is superior to LBM, FLBI and FADMM methods at the same sampling rate. For example, for the Lena image, at the sampling rate $r = 0.5$, the LBM, FLBI and FADMM methods require, respectively, 17.385, 6.987 and 6.824 s, for the test, while our FLADMM takes only about 5.706 s. Therefore, the convergence rate of the FLADMM algorithm is faster than the LBM, FLBI and FADMM methods in comparison. FLADMM has a faster convergence rate at the same sampling rate than the LBM, FLBI and FADMM algorithms is because that the accelerated strategy which inspired by the idea of the fast method [26] and the linearized method [27] is used to substantially accelerate the convergence rate.

To confirm the universality of the proposed FLADMM algorithm, we apply it now to reconstruct the four different groups of the test images: Peppers (256×256), Man (512×512), Barbara (256×256) and Mandrill (512×512). The PSNR in dB of the reconstructed images and the required run time in seconds at different sampling rates resulting from the FLADMM and LBM algorithms are listed in Table 3.

Clearly, our proposed method has the best performance and the fastest convergence rate among all these test images with different sampling rates.

Since the PSNR performance and the convergence rate depend on the sampling rate, it is necessary to choose an appropriate sampling rate for image reconstruction, and the sampling rate in the following test is set to 0.5. To illustrate the FLADMM robust to noise, a zero-mean Gaussian noise with variance σ^2 is added to the three different groups of the

Table 3 The PSNR (dB) and run time (s) of FLADMM and LBM for different test images

Test images	Evaluation measures	Algorithms	Sampling rate r				
			0.1	0.2	0.3	0.4	0.5
Peppers	PSNR	FLADMM	32.812	33.519	34.705	35.846	37.319
		LBM [22]	32.789	33.267	33.991	34.766	35.562
	Run time	FLADMM	0.798	1.586	2.835	4.462	5.857
		LBM [22]	1.595	2.811	5.312	9.338	17.416
Man	PSNR	FLADMM	33.985	34.696	35.568	36.579	37.948
		LBM [22]	33.782	34.245	35.137	36.012	36.987
	Run time	FLADMM	1.873	4.375	8.931	19.037	32.356
		LBM [22]	3.854	9.669	20.901	41.454	74.923
Barbara	PSNR	FLADMM	27.203	27.724	28.435	29.319	30.265
		LBM [22]	26.442	26.983	27.592	28.311	29.659
	Run time	FLADMM	1.877	3.253	9.765	14.941	17.985
		LBM [22]	5.406	10.782	11.608	24.415	35.763
Mandrill	PSNR	FLADMM	28.773	29.264	29.958	30.754	31.663
		LBM [22]	27.864	28.547	29.019	29.865	30.692
	Run time	FLADMM	4.788	12.067	20.507	29.275	44.255
		LBM [22]	7.075	16.399	38.007	75.248	130.94

test images: Lena (256×256), Peppers (256×256) and Man (512×512) images.

The proposed FLADMM reconstruction algorithm is applied to these test images with Gaussian noise of zero mean and variance $\sigma^2 = 0.001$. Experimental results on these noisy images are shown in Figs. 5, 6 and 7, respectively.

Figure 5a is the noisy Lena image with Gaussian noise of zero mean and variance $\sigma^2 = 0.001$, Fig. 5b–e is the reconstructed Lena images obtained by the FLADMM, LBM, FLBI and FADMM algorithms, respectively. The images in Figs. 6 and 7 have similar situations. It is clear from Figs. 5, 6 and 7 that the reconstructed images of our proposed FLADMM algorithm have more detailed information and are much closer to the original image as compared with

Fig. 5 Experimental results from different reconstruction algorithms for noisy Lena image. **a** Noisy image **b** FLADMM reconstruction **c** LBM reconstruction **d** FLBI reconstruction **e** FADMM reconstruction

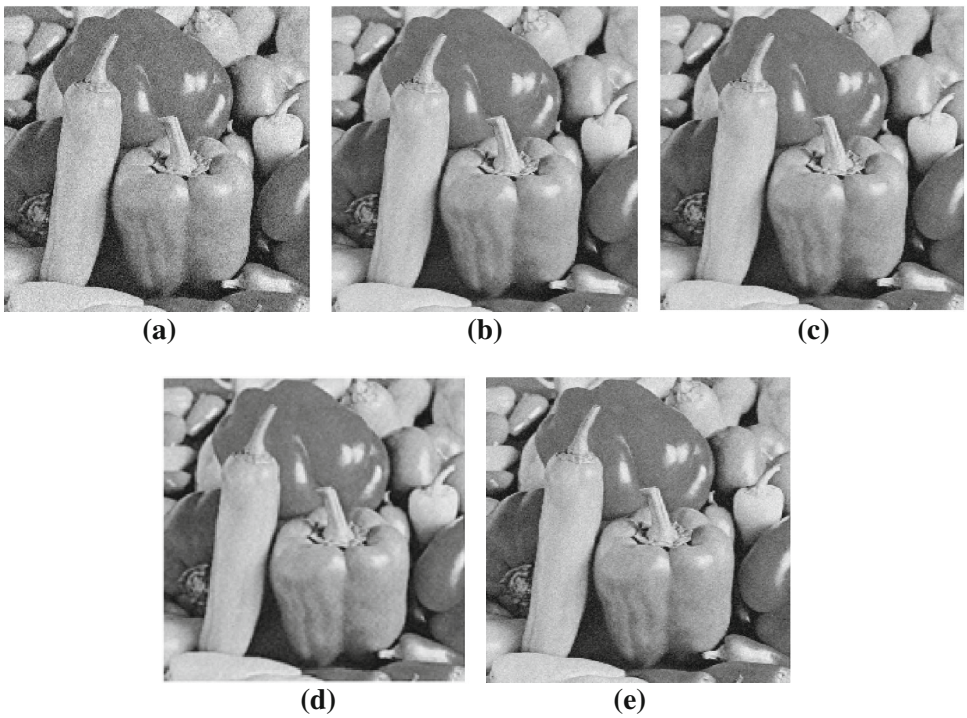

Fig. 6 Experimental results from different reconstruction algorithms for noisy Peppers image. **a** Noisy image **b** FLADMM reconstruction **c** LBM reconstruction **d** FLBI reconstruction **e** FADMM reconstruction

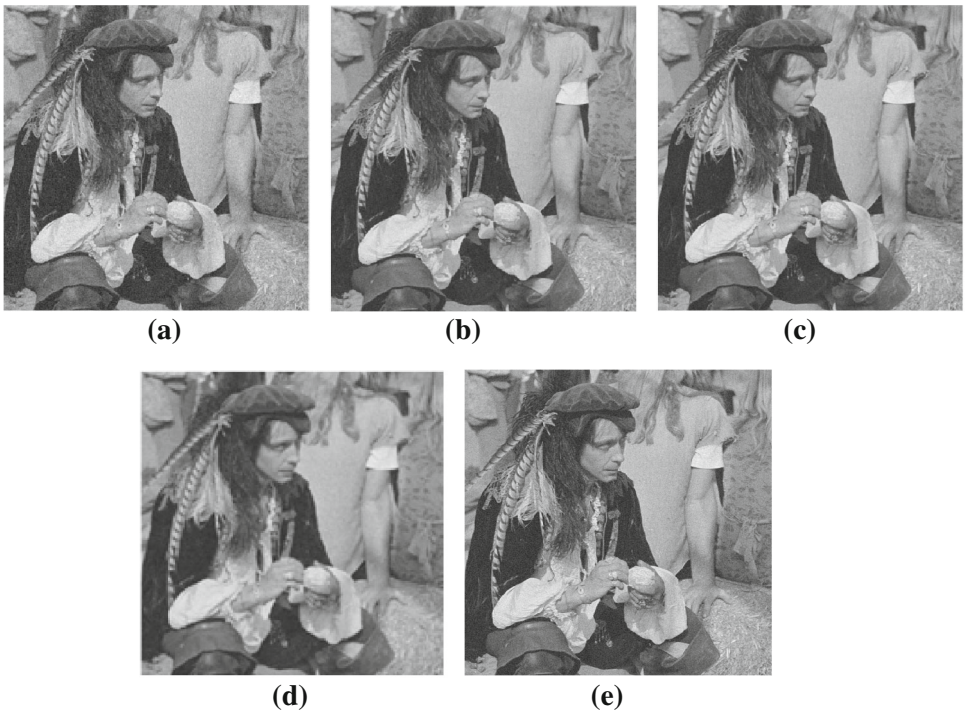

Fig. 7 Experimental results from different reconstruction algorithms for noisy Man image. **a** Noisy image **b** FLADMM reconstruction **c** LBM reconstruction **d** FLBI reconstruction **e** FADMM reconstruction

the LBM, FLBI and FADMM algorithms. In short, the proposed reconstruction algorithm performs slightly better in human perception of global information, which can not only prevent from the emergence of stripes and noises effectively, but also enhance the definition of the reconstructed image greatly.

For further comparison, the PSNR in dB is used to measure the robustness of the proposed algorithm for the five different groups of the noisy test images: Lena (256 × 256), Peppers (256 × 256), Man (512 × 512), Barbara (256 × 256), and Mandrill (512 × 512).

Table 4 gives the PSNR of the reconstructed three different groups of the noisy test images at different noise levels resulting from the FLADMM, LBM, FLBI and FADMM algorithms. It is read from the results that, with increasing the variance, the PSNR becomes lower for all these methods. Moreover, with increasing the variance, the superiority of the FLADMM becomes more obvious. In all, our proposed FLADMM algorithm outperforms LBM, FLBI and FADMM in terms of the PSNR at the same noise levels.

Another implementation of CS on 2D images breaks the image into smaller blocks and process the blocks independently. In this way, the pixels in each processing unit become more correlated and the transform domain representation becomes more sparse. Such an approach was proposed in [35] for block-based compressed sensing (BCS) for 2D images. In BCS, an image is divided into $g \times g$ nonoverlapping blocks (where $g = \sqrt{n}$) and acquired using an appropriately sized measurement matrix.

Here, we also take the Lena image (256 × 256) for testing and compare the FLADMM reconstruction algorithm with

Table 4 The PSNR (dB) for different noisy test images

Test images	Algorithms	Variance σ^2				
		0.001	0.005	0.01	0.05	0.1
Lena	FLADMM	33.703	29.838	27.547	21.219	19.706
	LBM [22]	32.025	27.912	25.536	19.098	17.162
	FLBI [24]	32.028	27.914	25.537	19.096	17.169
	FADMM [20]	32.065	28.041	25.636	19.824	17.632
Peppers	FLADMM	33.785	29.966	27.746	21.782	19.864
	LBM [22]	32.387	28.206	25.667	19.703	17.316
	FLBI [24]	32.401	28.209	25.670	19.708	17.319
	FADMM [20]	32.653	28.265	25.704	19.732	17.532
Man	FLADMM	34.105	30.257	28.082	22.263	20.306
	LBM [22]	33.121	29.138	26.535	20.494	18.213
	FLBI [24]	33.126	29.141	26.539	20.498	18.217
	FADMM [20]	33.334	29.355	26.727	20.641	18.442
Barbara	FLADMM	28.949	26.295	24.514	19.387	17.365
	LBM [22]	27.786	24.706	22.453	17.348	14.984
	FLBI [24]	27.798	24.738	22.499	17.353	15.016
	FADMM [20]	27.939	24.964	22.525	17.754	15.628
Mandrill	FLADMM	30.099	27.051	24.897	19.624	18.185
	LBM [22]	28.035	24.998	22.564	17.498	15.073
	FLBI [24]	28.087	25.031	22.595	17.524	15.099
	FADMM [20]	28.123	25.224	25.786	17.773	15.308

LBM [22], FLBI [24], FADMM [20] and SPL (Smoothed Projected Landweber) [35] algorithms.

For further comparison, PSNR in dB is used to measure the performance of the proposed algorithm for the Lena image

Table 5 The PSNR (dB) for Lena image

Algorithms	Sampling rate r				
	0.1	0.2	0.3	0.4	0.5
FLADMM	32.352	33.081	34.132	35.295	36.326
LBM [22]	31.733	32.441	33.522	34.201	35.108
FLBI [24]	31.738	32.447	33.526	34.205	35.113
FADMM [20]	31.712	32.315	33.363	34.106	35.074
SPL [35]	31.805	32.779	33.994	34.627	35.541

Table 6 The PSNR (dB) for noisy Lena image

Algorithms	Variance σ^2				
	0.001	0.005	0.01	0.05	0.1
FLADMM	34.359	30.542	28.565	22.673	20.971
LBM [22]	33.038	28.974	26.995	20.502	18.713
FLBI [24]	33.041	28.976	26.998	20.505	18.718
FADMM [20]	33.125	29.014	27.137	20.796	19.189
SPL [35]	33.207	29.138	27.291	20.956	19.469

(256×256) at different sampling rates and the noisy Lena image (256×256) at different noise levels.

The PSNR in dB of these methods to reconstruct Lena image at different sampling rates is listed in Table 5.

Table 5 gives the quantitative results of the FLADMM, LBM, FLBI, FADMM and SPL algorithms at different sampling rates. It is clear that all these methods provide very good performance. The results also clearly state that, with increasing the sampling rate, the PSNR becomes higher for all these methods. Moreover, the superiority of the FLADMM becomes more obvious. For example, when the sampling rate is $r = 0.5$, the PSNR of Lena image using our proposed FLADMM is 36.326 dB, while that produced by LBM, FLBI, FADMM and SPL algorithms is 35.108, 35.113, 35.074 and 35.541 dB only, respectively. These experimental results of Table 5 are consistent with that of Table 1. In contrast to Tables 1 and 5, we can also see that the quantitative results of all these methods for BCS are better than the quantitative results for CS at the same sampling rate.

The PSNR in dB of these methods to reconstruct noisy Lena image at different noise levels is listed in Table 6.

Table 6 gives the PSNR of the reconstructed noisy Lena images at different noise levels resulting from the FLADMM, LBM, FLBI, FADMM and SPL algorithms. It is read from the results that, with increasing the variance, the PSNR becomes lower for all these methods. Moreover, with increasing the variance, the superiority of the FLADMM becomes more obvious. In all, our proposed FLADMM algorithm outperforms LBM, FLBI, FADMM and SPL algorithms in terms of the PSNR at the same noise levels. These experimental results of Table 6 are consistent with that of Table 4 for the noisy Lena image. In contrast to Tables 4 and 6, we can also see that the quantitative results of all these methods for BCS are better than the quantitative results for CS at the same noise levels.

4 Conclusion

In this paper, an effective, robust and fast algorithm, referred to as FLADMM, has been proposed to reconstruct the sparse coefficients from the novel augmented ℓ_1-regularized problem, thereby to reconstruct the signal. The proposed algorithm is a variant of the classical ADMM, which takes full advantage of the separable structures of the problem. The computational cost of the proposed algorithm mainly depends on one soft thresholding projection and matrix-vector multiplications at each iteration. The experimental results have demonstrated that the novel FLADMM yields a higher PSNR reconstructed image as well as a faster convergence rate at the same sampling rate than the LBM, FLBI and FADMM algorithms do. Besides, this method is more robust than the LBM, FLBI and FADMM algorithms at the same noise level. FLADMM has a higher PSNR reconstructed image and more robust than the LBM, FLBI and FADMM methods is because that the FLADMM is used for solving the augmented ℓ_1-regularized model (6), while the LBM and FLBI algorithms are both used for solving the augmented BP model (5), and the FADMM is used for solving the ℓ_1-regularized problem (4). It has been shown that the augmented ℓ_1-regularized model (6) outperforms the ℓ_1-regularized problem (4) and the augmented BP model (5) on reported real-world regression problems [22,25]. And the FLADMM has a faster convergence rate at the same sampling rate than the LBM, FLBI and FADMM algorithms is because that the accelerated strategy which inspired by the idea of the fast method [26] and the linearized method [27] is used to substantially accelerate the convergence rate.

Acknowledgments This work is supported by the National Basic Research Program of China (973 Program) (No. 2011CB302903), the National Natural Science Foundation of China (No. 60971129, 61070234, 61271335, 61271240), the Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions–Information and Communication Engineering, the Research and Innovation Project for College Graduates of Jiangsu Province (No. CXZZ12_0469) and the Natural Science Foundation of the Jiangsu Higher Education Institutions of China(No. 13KJB510020).

References

1. Donoho, D.: Compressed sensing. IEEE Trans. Inf. Theory **52**(4), 1289–1306 (2006)
2. Donoho, D., Tsaig, Y.: Extensions of compressed sensing. Signal Process. **86**(3), 533–548 (2006)

3. Candes, E., Wakin, M.: An introduction to compressive sampling. IEEE Signal Process. Mag. **25**(2), 21–30 (2008)
4. Davenport, M., Duarte, M., Eldar, Y., Kutyniok, G.: Compressed Sensing: Theory and Applications. Cambridge University Press, Cambridge (2012)
5. Romberg, J.: Imaging via compressive sampling. IEEE Signal Process. Mag. **25**(2), 14–20 (2008)
6. Wipf, D., Rao, B.: Bayesian learning for sparse signal reconstruction. Proc. IEEE Int. Conf. Acoust. Speech Signal Process. **2003**(6), 601–604 (2003)
7. Lv, X., Bi, G., Wan, C.: The group Lasso for stable recovery of block-sparse signal representations. IEEE Trans. Signal Process. **59**(4), 1371–1382 (2011)
8. Masood, M., Al-Naffouri, T.: Sparse reconstruction using distribution agnostic Bayesian matching pursuit. IEEE Trans. Signal Process. **61**(21), 5298–5309 (2013)
9. Peleq, T., Eldar, Y., Elad, M.: Exploiting statistical dependencies in sparse representations for signal recovery. IEEE Trans. Signal Process. **60**(5), 2286–2303 (2012)
10. Hurtado, M., Muravchik, C., Nehorai, A.: Enhanced sparse Bayesian learning via statistical thresholding for signals in structured noise. IEEE Trans. Signal Process. **61**(21), 5430–5443 (2013)
11. Ramirez, A., Arce, G., Otero, D., Paredes, J., Sadler, B.: Reconstruction of sparse signals from ℓ_1 dimensionality-reduced Cauchy random projections. IEEE Trans. Signal Process. **60**(11), 5725–5737 (2012)
12. Chen, S., Donoho, D., Saunders, M.: Atomic decomposition by basis pursuit. SIAM Rev. **43**(1), 129–159 (2001)
13. Kose, K., Gunay, O., Cetin, A.: Compressive sensing using the modified entropy functional. Digit. Signal Process.: A Rev. J. **24**, 63–70 (2014)
14. Figueiredo, M., Nowak, R., Wright, S.: Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems. IEEE J. Sel. Top. Signal Process. **1**(4), 586–598 (2007)
15. Afonso, M., Bioucas, D., Figueiredo, M.: An augmented Lagrangian approach to the constrained optimization formulation of image inverse problems. IEEE Trans. Image Process. **20**(3), 681–695 (2011)
16. Kose, K., Cevher, V., Cetin, A.: Filtered variation method for denoising and sparse signal processing. Proc. IEEE Int Conf. Acoust. Speech Signal Process. **2012**, 3329–3332 (2012)
17. Pustelnik, N., Chaux, C., Pesquet, J.: Parallel proximal algorithm for image restoration using hybrid regularization. IEEE Trans. Image Process. **20**(9), 2450–2462 (2011)
18. Yang, J., Zhang, Y.: Alternating direction algorithm for ℓ_1-problems in compressed sensing. SIAM J. Sci. Comput. **33**(1), 250–278 (2011)
19. Patrick, L., Pesquet, J.: Proximal splitting methods in signal processing. In: Fixed-Point Algorithms for Inverse Problems in Science and Engineering Springer Optimization and its Applications. pp. 185–212 (2011)
20. Goldstein, T., O'Donoghue, B., Setzer, S.: Fast alternating direction optimization methods. SIAM J. Imaging. Submitted, 2012. Available from ftp://ftp.math.ucla.edu/pub/camreport/cam12-35.pdf (2012)
21. Lai, M., Yin, W.: Augmented ℓ_1 and nuclear-norm models with a globally linearly convergent algorithm. Submitted, 2013. Available from http://arxiv.org/abs/1201.4615 (2013)
22. Cai, J., Osher, S., Shen, Z.: Linearized Bregman iterations for compressed sensing. Math. Comput. **78**(267), 1515–1536 (2009)
23. Huang, B., Ma, S., Goldfarb, D.: Accelerated linearized Bregman method. J. Sci. Comput. **54**(2–3), 428–453 (2013)
24. Osher, S., Mao, Y., Dong, B., Yin, W.: Fast linearized Bregman iterations for compressed sensing and sparse denoising. Commun. Math. Sci. **8**, 93–111 (2010)
25. Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. J. R. Stat. Soc.: Ser. B (Statistical Methodology) **67**(2), 301–320 (2005)
26. Beck, A., Teboulle, M.: A fast iterative shrinkage/thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. **2**(1), 183–202 (2009)
27. Yang, J., Yuan, X.: Linearized augmented Lagrangian and alternating direction methods for nuclear norm minimization. Math. Comput. **82**(281), 301–329 (2013)
28. Nocedal, J., Wright, S.: Numerical Optimization. Springer, Berlin (2000)
29. Eckstein, J., Bertsekas, D.: On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators. Math. Program. **55**, 293–318 (1992)
30. Yamada, I., Yukawa, M., Yamagishi, M.: Minimizing the Moreau envelope of nonsmooth convex functions over the fixed point set of certain quasi-nonexpansive mappings. In: Fixed-Point Algorithms for Inverse Problems in Science and Engineering Springer Optimization and its Applications, pp. 345–390 (2011)
31. Preda, R., Vizireanu, D.: Quantisation-based video watermarking in the wavelet domain with spatial and temporal redundancy. Int. J. Electron. **98**(3), 393–405 (2011)
32. Yang, Z., Yang, Z.: Novel multifocus image fusion and reconstruction framework based on compressed sensing. IET Image Process. **7**(9), 837–847 (2013)
33. Yang, Z., Yang, Z.: ℓ_0-regularization signal reconstruction based on fast alternating direction method of multipliers for compressed sensing. J. Electron. Inf. Technol. **35**(4), 826–831 (2013)
34. Yang, Z., Yang, Z.: Fast linearized Bregman method for compressed sensing. KSII Trans. Internet Inf. Syst. **7**(9), 2284–2298 (2013)
35. Fowler, J., Mun, S., Tramel, E.: Block-based compressed sensing of images and video. Found. Trends Signal Process. **4**(4), 297–416 (2012)