Hindawi Publishing Corporation
Computational and Mathematical Methods in Medicine
Volume 2016, Article ID 1724630, 14 pages
http://dx.doi.org/10.1155/2016/1724630

Research Article

Sparse Parallel MRI Based on Accelerated Operator Splitting Schemes

Nian Cai,[1] Weisi Xie,[2,3] Zhenghang Su,[1,2,3] Shanshan Wang,[2,3] and Dong Liang[2,3]

[1] *School of Information Engineering, Guangdong University of Technology, Guangzhou 510006, China*
[2] *Paul C. Lauterbur Research Centre for Biomedical Imaging, Shenzhen Institutes of Advanced Technology, Shenzhen, China*
[3] *Shenzhen Key Laboratory for MRI, Shenzhen, Guangdong, China*

Correspondence should be addressed to Dong Liang; dong.liang@siat.ac.cn

Received 26 April 2016; Accepted 29 August 2016

Academic Editor: Po-Hsiang Tsui

Recently, the sparsity which is implicit in MR images has been successfully exploited for fast MR imaging with incomplete acquisitions. In this paper, two novel algorithms are proposed to solve the sparse parallel MR imaging problem, which consists of l_1 regularization and fidelity terms. The two algorithms combine forward-backward operator splitting and Barzilai-Borwein schemes. Theoretically, the presented algorithms overcome the nondifferentiable property in l_1 regularization term. Meanwhile, they are able to treat a general matrix operator that may not be diagonalized by fast Fourier transform and to ensure that a well-conditioned optimization system of equations is simply solved. In addition, we build connections between the proposed algorithms and the state-of-the-art existing methods and prove their convergence with a constant stepsize in Appendix. Numerical results and comparisons with the advanced methods demonstrate the efficiency of proposed algorithms.

1. Introduction

Reducing encoding is one of the most important ways for accelerating magnetic resonance imaging (MRI). Partially parallel imaging (PPI) is a widely used reduced-encoding technique in clinic due to many desirable properties such as linear reconstruction, easy use, and g-factor for clearly characterizing the noise property [1–6]. Specifically, PPI exploits the sensitivity prior in multichannel acquisitions to take less encodings than the conventional methods [7]. Its acceleration factor is restricted to the number of channels. More and more large coil arrays, such as 32-channel [8–11], 64-channel [12], and even 128-channel [13], have been used for faster imaging. However, the acceleration ability of PPI under the condition of ensuring certain signal noise ratio (SNR) is still limited because the imaging system is highly ill-posed and can enlarge the sampling noise with higher acceleration factor. One solution is to introduce some other prior information as the regularization term into the imaging equation. Sparsity prior, becoming more and more popular due to the emergence of compressed sensing (CS) theory [14–16], has been extensively exploited to reconstruct target image from a small amount of acquisition data (i.e., below the Nyquist sampling rate) in many MRI applications [17–20]. Because PPI and compressed sensing MRI (CSMRI) are based on different ancillary information (sensitivity for the former and sparseness for the latter), it is desirable to combine them for further accelerating the imaging speed.

Recently, SparseSENSE and its equivalence [1, 3, 21–25] have been proposed as a straightforward method to combine PPI and CS. The formulation of this method is similar to that in SparseMRI, except that the Fourier encoding is replaced by the sensitivity encoding (comprising Fourier encoding and sensitivity weighting). Generally, SparseSENSE aims to solve the following optimization problem:

$$\min_{x \in \mathbb{C}^n} \|Dx\|_1 + \frac{\lambda}{2} \|Ax - y\|_2^2, \qquad (1)$$

where the first term is the regularization term and the second one is the data consistency term. $\|\cdot\|_1$ and $\|\cdot\|_2$ represent separately 1-norm and 2-norm and $x \in \mathbb{C}^n$ is the to-be-reconstructed image. $D \in \mathbb{C}^{m\times n}$ denotes a special transform (e.g., spatial finite difference and wavelet) and the term $\|Dx\|_1$ controls the solution sparsity. A and y are the encoding matrix and the measured data, respectively:

$$A = \begin{pmatrix} F_p S_1 \\ \vdots \\ F_p S_k \end{pmatrix} \in \mathbb{C}^{kl\times n} \quad (l < n), \qquad y = \begin{pmatrix} y_1 \\ \vdots \\ y_k \end{pmatrix} \in \mathbb{C}^{kl}, \tag{2}$$

where F_p is the partial Fourier transform and $S_k \in \mathbb{C}^{n\times n}$ is the diagonal sensitivity map for receiver k. $y_k \in \mathbb{C}^{l\times 1}$ is the measured k-space data at receiver k. In this paper, we mainly solve the popular total variation (or its improved version: total generalized variation) based SparseSENSE model, that is, $\|Dx\|_1 = \|x\|_{\mathrm{TV}}$ (or $\|D\|_1 = \|x\|_{\mathrm{TGV}}$).

For the minimization (1), there exists computational challenge not only from the nondifferentiability of l_1 norm term but also from the ill-condition of the large size inversion matrix A. Further, the computational complexity becomes more and more huge if we try to improve the performance of SparseSENSE by using large coil arrays, high undersampling factor, or some more powerful transformations (which are usually nonorthogonal) to squeeze sparsity. Therefore, rapid and efficient numerical algorithms are highly desirable, especially for large coil arrays, high undersampling factor, and general sparsifying transform.

Several rapid numerical algorithms can solve the numerical difficulties, which are, for example, alternating direction method of multipliers (ADMM) [26], augmented Lagrangian method (ALM) [27], splitting Bregman algorithm (SBA) [28], splitting Barzilai-Borwein (SBB) [24], and Bregman operator splitting (BOS) [29]. The efficiency of these methods largely depends on the special structure of the matrix operator $D^T D$ (such as Toeplitz matrix and orthogonal matrix) and the encoding kernel (without the sensitivity maps). However, they are not suitable for simultaneously dealing with general regularization operator D and the parallel encoding matrix A. That is, these algorithms are not able to solve the problem (1) efficiently because the complex inversion of the large size matrix has to be computed, if $D^T D$ and/or $A^T A$ cannot be diagonalized directly by fast Fourier transform (FFT). Alternating minimization (AM) algorithm can address the issue of general D and A, which is a powerful optimization scheme that breaks a complex problem into simple subproblems [3]. But the addition of new variable may slow the speed of convergence. Our numerical results in Section 4 also demonstrate that the alternating minimization algorithm for large coil data is not very effective in the aspect of convergence speed. Table 1 illustrates the ability of working on general D and A (without any preconditioning) for these algorithms. We can see that only AM is able to deal with general operators simultaneously.

Table 1: The classification of algorithms for solving the SparseSENSE model.

	ADMM	ALM	SBA	BOS	SBB	AM
Works for general D?	No	No	No	No	No	Yes
Works for general A?	No	No	No	Yes	Yes	Yes

To solve the problems existing in the algorithms mentioned above, this paper develops two fast numerical algorithms based on the operator splitting and Barzilai-Borwein techniques. The proposed algorithms can be classified into the forward backward splitting (FBS) method [30] or its variations. Different from some existing fast algorithms, the proposed algorithms can treat general matrix operators D and A and avoid solving a partial differential equation so as to save huge computational cost. The superiority of our algorithms lies in that operator splitting is applied to both regularization term and data consistency term. Meanwhile, they ensure that a well-posed optimization system of equation is simply solved. Barzilai-Borwein (BB) stepsize selection scheme [31] is adopted for much faster computation speed.

This paper is organized as follows. In Section 2, a review on some related numerical methods for solving the SparseSENSE model is given. In Section 3, two algorithms are proposed as variations of forward-backward splitting scheme. We compare the proposed algorithms with popular algorithms based on the SparseSENSE model in Section 4. In Section 5, we discuss the parameters selection of the proposed algorithms and the connections between the proposed algorithms and the existing algorithms. Section 6 concludes this paper. Appendix proves the convergence of the proposed algorithms with constant stepsizes.

2. Related Work

In the early works, gradient descent methods with explicit or semi-implicit schemes [19, 32] were usually used to solve problem (1), in which the nondifferentiable norm was approximated by a smooth term:

$$\|x\|_{\mathrm{TV},\varepsilon} = \sum_{i=1}^{n} \sqrt{\|D_i x\|_2^2 + \varepsilon}, \tag{3}$$

where $D_i x \in \mathbb{R}^2$ contains the forward finite differences of x. The selection of the regulating positive parameter ε is crucial for the reconstruction results and convergence speed. A large parameter encourages a fast convergence rate but fails to preserve high quality details. A small one preserves fine structures in the reconstruction at the expense of slow convergence.

The methods in [33, 34] and the split Bregman method [28] equivalent to the alternating direction method of multipliers [26] were presented for solving minimization (1). The efficiency of the algorithms benefits from the soft shrinkage operator and the special structure of the encoding matrix and sparse transform. This requires that both $A^T A$ and $D^T D$ in the optimal equation on x can be directly diagonalized by FFT. But these methods may not be suitable for the parallel encoding matrix A and more general transform D. They are even ill-posed if $\mathrm{Null}(A) \cap \mathrm{Null}(D) \neq \{0\}$, where $\mathrm{Null}(\cdot)$ represents the null space of the operator. In addition, the augmented Lagrangian method in [27] preconditioned the encoding matrix A and inevitably computed the inversion of the matrix including general $D^T D$. So, it is also invalid in the computational efficiency.

The Bregman operator splitting (BOS) method replaces $\|Ax - y\|_2^2$ by a proximal-like term [29]. BOS is able to deal with A of uncertainty structure by the following iterations:

$$\begin{aligned} z^{k+1} &= \delta x^k - A^T\left(Ax^k - y\right) \\ s^{k+1} &= \arg\min_s \|s\|_1 \\ &\quad + \frac{\rho}{2}\left\|s - \left(w^k + Dx^k\right)\right\|_2^2 \\ \left(\rho D^T D + \lambda\delta I\right) x^{k+1} &= \rho D^T\left(s^{k+1} - w^k\right) + \lambda z^{k+1} \\ w^{k+1} &= w^k + Dx^{k+1} - s^{k+1}. \end{aligned} \tag{4}$$

However, a partial differential equation including $D^T D$ should be solved as indicated in (4). This equation may bring heavy computation for the general regularization operator D. To solve the problem of heavy computation, Ye et al. presented a SBB scheme by utilizing the BB stepsize [24]. However, these algorithms may be not efficient for the general D if the matrix operator $D^T D$ cannot be diagonalized by fast transform.

Consequently, minimization (1) can be written as a saddle-point problem:

$$\min_{x\in\mathbb{C}^n}\max_{w\in X}\left\langle x, D^T w\right\rangle + \frac{\lambda}{2}\|Ax - y\|_2^2, \tag{5}$$

where $X = \{z : z \in \mathbb{C}^m, |w_j| \leq 1 \text{ for } j = 1, 2, \ldots, m\}$. Although the primal-dual hybrid gradient (PDHG) method alternately updates the primal and dual variables x and z [35], its efficiency relies on the special structure of A. That is, $A^T A$ can be diagonalized directly by fast transform. The alternating minimization (AM) algorithm [3] reduces the PPI reconstruction problem with regularization into the TV-based image denoising and least square (LS) subproblems as

$$\begin{aligned} v^{k+1} &= \arg\min_v \|v\|_{\mathrm{TV}} + \alpha\|v - x\|_2^2 \\ x^{k+1} &= \arg\min_x \frac{\lambda}{2}\|Ax - y\|_2^2 + \alpha\|v - x\|_2^2. \end{aligned} \tag{6}$$

The AM algorithm is updated as follows:

$$\begin{aligned} z^{k+1} &= \delta_k x^k - A^T\left(Ax^k - y\right) \\ w^{k+1} &= \arg\min_{w\in X}\frac{1}{2}\left\|w - \left(w^k + \tau_k D^k v^k\right)\right\|_2^2 \\ v^{k+1} &= \left(1 + 2\alpha\theta_k\right)^{-1}\left(v^k + 2\alpha\theta_k x^k - \theta_k D^T w^{k+1}\right) \\ x^{k+1} &= \left(\lambda\delta_k + 2\alpha\right)^{-1}\left(2\alpha v^{k+1} + \lambda z^{k+1}\right) \\ \delta_{k+1} &= \frac{\left\|A\left(x^{k+1} - x^k\right)\right\|_2^2}{\left\|x^{k+1} - x^k\right\|_2^2}, \end{aligned} \tag{7}$$

where the stepsize is updated by the rules $\tau_k = 0.2 + 0.08k$, $\theta_k = (0.5 - 5/(15 + l))/\tau_k$ [35].

3. Algorithm Framework

In this section, we propose two algorithms for solving the SparseSENSE model (1), which are based on the operator splitting scheme and connected by Yosida approximation. We deduce the algorithms with the fixed-point technology as follows.

For the convenience of derivation, we denote $\|D\cdot\|_1$ by $(\varphi \circ D)(\cdot)$ and rewrite the SparseSENSE model as

$$\min_{x\in\mathbb{C}^n}(\varphi\circ D)(x) + \frac{\lambda}{2}\|Ax - y\|_2^2. \tag{8}$$

By the classic arguments of convex analysis, solution x^* to (8) satisfies the first-order optimality condition:

$$0 \in \partial(\varphi\circ D)(x^*) + \lambda A^T(Ax^* - y), \tag{9}$$

where $\partial(\varphi\circ D)(z)$ is the subdifferential of $(\varphi\circ D)$ at point z. According to the chain rule, subdifferential $\partial(\varphi\circ D)$ is identified by

$$\partial(\varphi\circ D) = D^T \circ (\partial\varphi) \circ D. \tag{10}$$

By substituting it into (9) and splitting, we get the equivalent formulation:

$$\begin{aligned} z^* &= x^* - \delta^{-1}A^T(Ax^* - y), \\ w^* &\in \partial\varphi(Dx^*), \\ 0 &= D^T w^* + \lambda\delta(x^* - z^*). \end{aligned} \tag{11}$$

In addition, for any positive number γ, we have

$$\begin{aligned} w^* \in \partial\varphi(Dx^*) &\Longleftrightarrow \\ \gamma w^* + Dx^* \in \gamma\partial\varphi(Dx^*) + Dx^* &\Longleftrightarrow \\ Dx^* = \mathrm{Prox}_{\gamma\varphi}(\gamma w^* + Dx^*) &\Longleftrightarrow \\ \gamma w^* = \gamma w^* + Dx^* - \mathrm{Prox}_{\gamma\varphi}(\gamma w^* + Dx^*), \end{aligned} \tag{12}$$

where the proximal operator $\text{Prox}_{\gamma\varphi}(v)$ is defined as

$$\text{Prox}_{\gamma\varphi}(v) = \arg\min_{u} \varphi(u) + \frac{1}{2\gamma}\|u - v\|_2^2. \tag{13}$$

Therefore, when the Barzilai-Borwein technique is involved, the solution to the minimization problem (1) can be obtained quickly based on the following updating:

$$\begin{aligned} z^{k+1} &= x^k - \delta_k^{-1} A^T \left(Ax^k - y\right) \\ s^{k+1} &= \arg\min_{s} \varphi(s) + \frac{1}{2\gamma}\left\|s - \left(\gamma w^k + Dx^k\right)\right\|_2^2 \\ \gamma w^{k+1} &= \gamma w^k + Dx^k - s^{k+1} \\ x^{k+1} &= z^{k+1} - \left(\lambda\delta_k\right)^{-1} D^T w^{k+1} \\ \delta_{k+1} &= \frac{\left\|A\left(x^{k+1} - x^k\right)\right\|_2^2}{\left\|x^{k+1} - x^k\right\|_2^2}. \end{aligned} \tag{14}$$

The above iterations can be identified as a forward-backward operator splitting method [36]. Comparing (14) to (4), the partial differential equation in (4) is not involved. Since $\varphi(\cdot) = \|\cdot\|_1$ in the SparseSENSE model (1), the proximal operator is a shrinkage operator that is able to solve s^{k+1} quickly from the following formulation:

$$\text{Prox}_{\gamma\|\cdot\|_1}(v) = \frac{v}{\|v\|_2}\max\left(\|v\|_2 - \gamma, 0\right). \tag{15}$$

The proposed algorithm (14) is referred to as the forward-backward operator splitting shrinkage (FBOSS) algorithm.

Considering Moreau's decomposition [30], for the proximal operator there exists a connection as follows:

$$v = \text{Prox}_{\gamma\varphi}(v) + \gamma\text{Prox}_{\varphi^*/\gamma}\left(\frac{v}{\gamma}\right), \tag{16}$$

where φ^* represents the conjugate function of φ. Applying (16) to the updates in (14), for all k we get a modified iterating sequences as

$$\begin{aligned} z^{k+1} &= x^k - \delta_k^{-1} A^T \left(Ax^k - y\right) \\ w^{k+1} &= \arg\min_{t\in X} \varphi^*(t) + \frac{\gamma}{2}\left\|t - \left(w^k + \gamma^{-1}Dx^k\right)\right\|_2^2 \\ x^{k+1} &= z^{k+1} - \left(\lambda\delta_k\right)^{-1} D^T w^{k+1} \\ \delta_{k+1} &= \frac{\left\|A\left(x^{k+1} - x^k\right)\right\|_2^2}{\left\|x^{k+1} - x^k\right\|_2^2}. \end{aligned} \tag{17}$$

Obviously, the proximal operator $\text{Prox}_{\gamma\varphi^*}$ for minimization (1) is a projection operator and $X = \{x \in \mathbb{C}^n : |x_j| \le 1 \text{ for } j = 1, 2, \ldots, n\}$. According to (15) and (16), we obtain the following computation through a simple derivation:

$$\text{Prox}_{\|\cdot\|_1^*/\gamma}\left(\frac{v}{\gamma}\right) = \frac{v}{\|v\|_2}\min\left(\frac{\|v\|_2}{\gamma}, 1\right). \tag{18}$$

This shows that the modified iteration algorithm (17) is also a fast numerical algorithm for solving minimization (1), which is called the forward-backward operator splitting projection (FBOSP) method. This is because the projection operator keeps the calculation speed same as the shrinkage.

4. Numerical Experiments

Three sets of MR data are utilized in the experiments. Data 1 was acquired on a GE 3T scanner (GE Healthcare, Waukesha, WI) with a 32-channel head coil and a 3D T1-weighted spoiled gradient echo sequence (TE = minimum full, TR = 7.5 ms, FOV = 24 × 24 cm, matrix = 256 × 256, and slice thickness = 1.7 mm). Data 2 was of one frame of the dynamic MR data, which was acquired on a 3T Siemens Verio scanner (Siemens Medical Solutions, Erlangen, Germany) (flip angle = 50 degree, TE/TR = 56.6/1.89 ms, Size = 256×225×17×15, FOV = 340 mm × 287 mm, and slice thickness = 6 mm). Data 3 (size = 256 × 256 × 8) was downloaded from the following website: http://www.ece.tamu.edu/~jimji/pulsarweb/ [37].

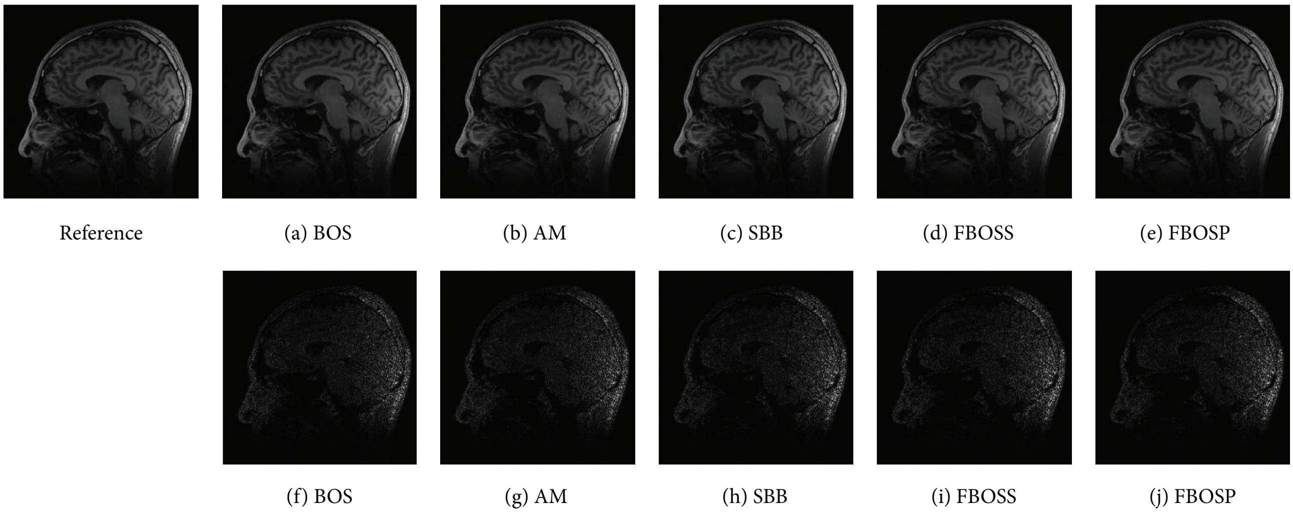
We can directly reconstruct the sparse data acquired from MRI by means of our proposed schemes. However, we employ the fully acquired data as the reference to make quantitative comparisons between our proposed schemes and the other methods. So, all data sets were fully acquired and then artificially undersampled using nonuniform random sampling masks corresponding to different undersampling factors. Here, a ground truth or reference image x^* is set to the square-root of sum of squares of coil images obtained by fully sampled k-space data from all channels. Peak signal noise ratio (PSNR) and relative error are employed as the performance metrics, which are defined as $20\log_{10}(255/\|x - x^*\|_2^2)$ and $\|x - x^*\|_2/\|x^*\|_2$, respectively. The sensitivity maps S_j are simulated with the full k-space data to avoid the effect of inaccurate sensitivity estimation.

All the comparison algorithms were implemented in MATLAB (version R2013a) and performed on a computer with an Intel(R) Xeon(R) CPU X5690 3.47 GHz, 64 GB of memory, and a Windows operating system. Empirically, we set $\lambda = 1.0 \times 10^3$ for all algorithms, $\rho = 0.5$ for BOS, and $\alpha = 1.0 \times 10^2$ for AM. For BOS, $\delta = 1$. γ is set in the range $[10^{-1}, 10^1]$ for FBOSS and FBOSP. Each algorithm is terminated when the relative change $\|x^k - x^{k-1}\|_2/\|x^k\|_2$ reaches the predefined stopping criterion $\varepsilon = 5 \times 10^{-5}$. This criterion could guarantee that the iterative solution for all algorithms approximates to the optimal solution sufficiently.

4.1. Comparisons on the TV-Based SparseSENSE Model. In this subsection, the comparisons on the TV-based SparseSENSE model were carried out by BOS [29], AM [3], SBB [24], FBOSS, and FBOSP. We tested on each data set with different undersampling factors. Table 2 illustrates the numerical comparison results. Here, we did not use 10-fold undersampling factor for data 2 because the reconstruction error for all the algorithms is very large due to high undersampling rate and noise in data acquisition. As shown in Table 2, the proposed algorithms FBOSS and FBOSP achieve

Table 2: Numerical results for the TV-based SparseSENSE model.

	Coil number	Undersampling factor	Performance metric	BOS	AM	SBB	FBOSS	FBOSP
Data 1	32	10	PSNR (dB)	46.87	47.36	49.53	49.61	49.57
			Relative error ($e-2$)	3.04	2.87	2.24	2.22	2.23
			CPU time (sec.)	325.4	132.3	95.3	55.3	53.0
Data 1	32	6	PSNR (dB)	55.93	56.99	57.62	58.11	57.76
			Relative error ($e-3$)	10.68	9.49	8.82	8.34	8.68
			CPU time (sec.)	162.7	78.2	45.4	32.2	31.2
Data 1	32	4	PSNR (dB)	61.00	62.41	62.80	63.56	63.56
			Relative error ($e-3$)	5.98	5.08	4.86	4.45	4.45
			CPU time (sec.)	99.9	48.0	29.3	27.2	26.5
Data 2	15	6	PSNR (dB)	54.22	54.35	57.17	57.25	57.95
			Relative error ($e-3$)	14.82	14.60	10.55	10.46	9.65
			CPU time (sec.)	199.4	75.1	35.1	29.3	24.7
Data 2	15	4	PSNR (dB)	63.21	65.34	65.93	66.22	66.70
			Relative error ($e-3$)	5.26	4.12	3.85	3.72	3.50
			CPU time (sec.)	77.7	30.8	16.0	12.7	10.8
Data 3	8	10	PSNR (dB)	40.20	40.21	40.61	40.64	40.68
			Relative error ($e-2$)	6.06	6.05	5.79	5.77	5.73
			CPU time (sec.)	60.5	24.5	18.0	13.2	10.8
Data 3	8	6	PSNR (dB)	48.86	49.29	50.20	50.13	50.21
			Relative error ($e-2$)	2.24	2.13	1.92	1.93	1.92
			CPU time (sec.)	51.1	21.4	16.6	11.2	9.8
Data 3	8	4	PSNR (dB)	51.72	51.80	51.78	51.86	51.79
			Relative error ($e-2$)	1.61	1.60	1.60	1.58	1.60
			CPU time (sec.)	34.0	13.2	10.3	8.5	7.6


Reference

(a) BOS

(b) AM

(c) SBB

(d) FBOSS

(e) FBOSP

(f) BOS

(g) AM

(h) SBB

(i) FBOSS

(j) FBOSP

Figure 1: (a)–(e) Reconstructed images with relative error, 3.04%, 2.87%, 2.24%, 2.22%, and 2.23%, separately on data 1 when the undersampling factor equals 10. (f)–(j) Absolute error images corresponding to (a)–(e).

better reconstruction performances with less computational time compared to the other three methods.

As shown in Figures 1, 2, and 3, images reconstructed by the five algorithms have similar reconstruction qualities. In Figures 4 and 5, we give the reconstructed results and error images based on FBOSS and FBOSP at different iterations to illustrate that the proposed algorithms can improve image quality iteratively.

Besides, we plot the relative error and PSNR as functions of the CPU time to examine the efficiency of FBOSS, FBOSP

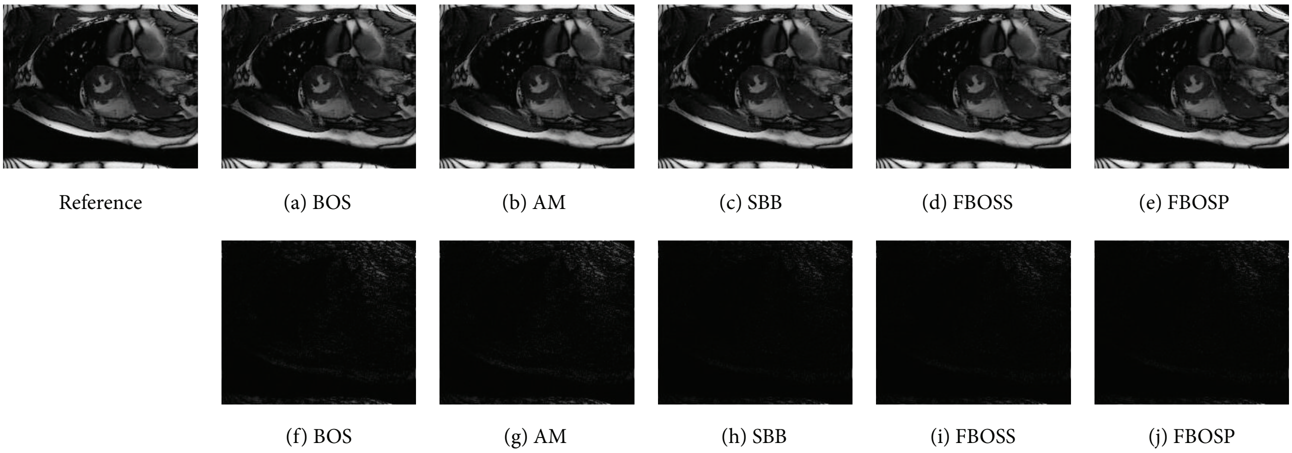


Figure 2: (a)–(e) Reconstructed images with relative error, 1.48%, 1.46%, 1.06%, 1.05%, and 0.97%, separately on data 2 when the undersampling factor equals 6. (f)–(j) Absolute error images corresponding to (a)–(e).

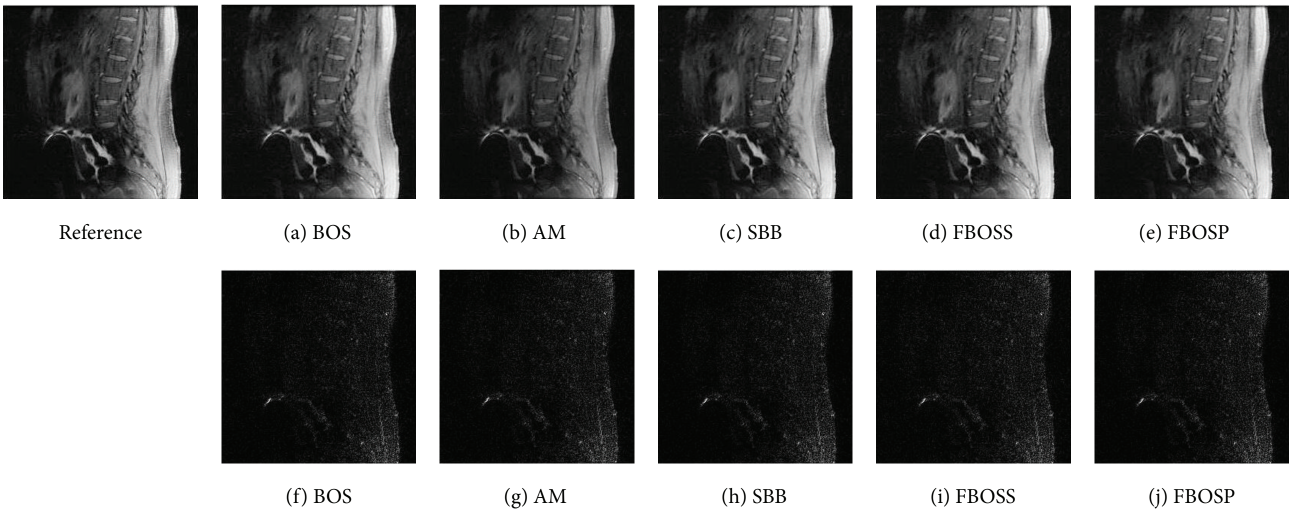
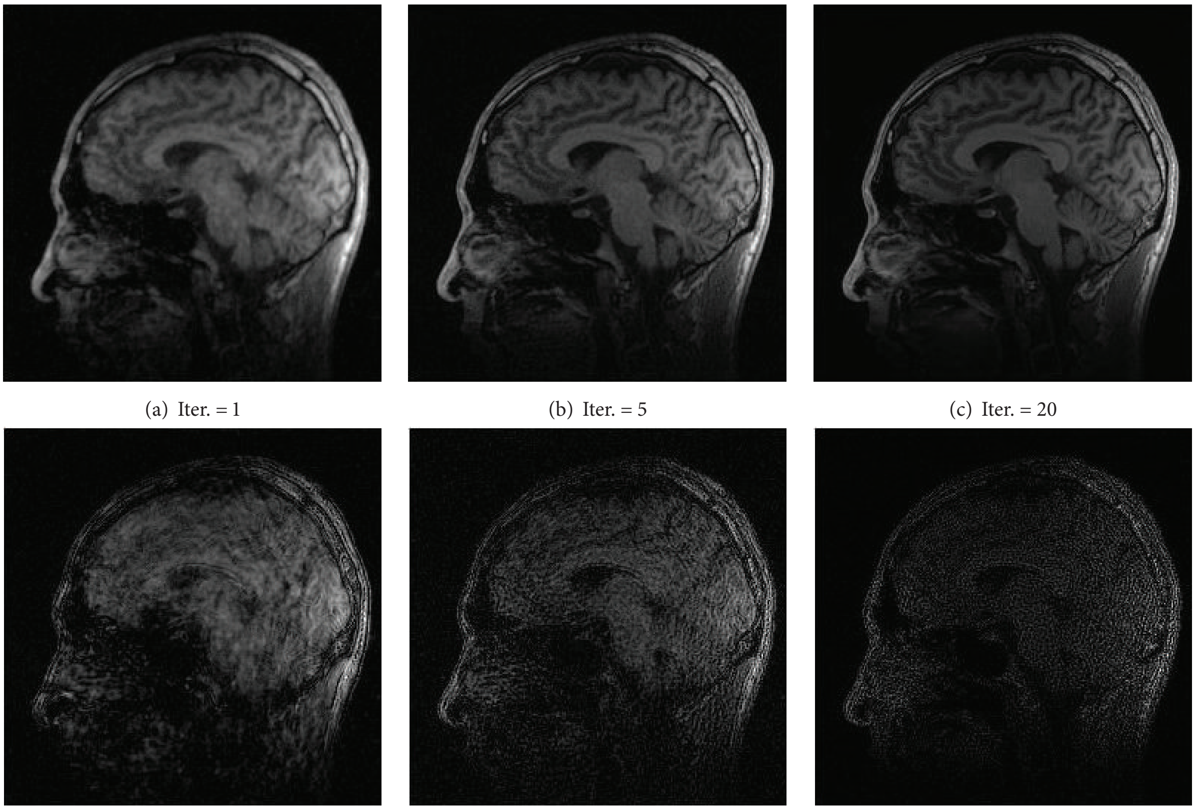


Figure 3: (a)–(e) Reconstructed images with relative error, 6.06%, 6.05%, 5.79%, 5.77%, and 5.73%, separately on data 3 when the undersampling factor equals 10. (f)–(j) Absolute error images corresponding to (a)–(e).

compared to BOS, AM, and SBB. As indicated in Figures 6(a), 6(b), and 6(c), the relative error curves of FBOSS and FBOSP descend faster than those of BOS, AM, and SBB. The curve of BOS is far above the others, which implies its lower efficiency. FBOSS and FBOSP have almost identical performance. Figures 6(d), 6(e), and 6(f) demonstrate that PSNR curves of FBOSS and FBOSP have the fastest ascents among these five algorithms.

The above experiments illustrate that the proposed algorithms FBOSS and FBOSP have better performance than BOS, SBB, and AM in terms of computational efficiency, although D^TD generated from the TV-based SparseSENSE model can be diagonalized by FFT. Besides, as indicated in Table 2, the larger the undersampling factor of the test data, the bigger the CPU time gap between the proposed methods and the others. This gap for the small downsampling factor is not obvious. This is because the initial solution x^0 for the data with small undersampling factor is close to the optimal solution x^*; the iterations for the algorithms only take a small amount of time to meet the stopping criterion. Nevertheless, the case for the data with large undersampling factor is opposite. In addition, we find out that the larger the coil numbers of the test data are, the more significant the performances of the proposed methods become. To demonstrate this point, we plotted the CPU time under similar relative error as the function of the coil number of data in Figures 7(a) and 7(b). From the figures we can see that as the coil number increases the proposed algorithms ascend more slowly than the others. The robustness for the coil numbers of data set becomes another advantage of the presented algorithms. Thus, these observations indicate that the proposed algorithms are superior especially in the



(a) Iter. = 1

(b) Iter. = 5

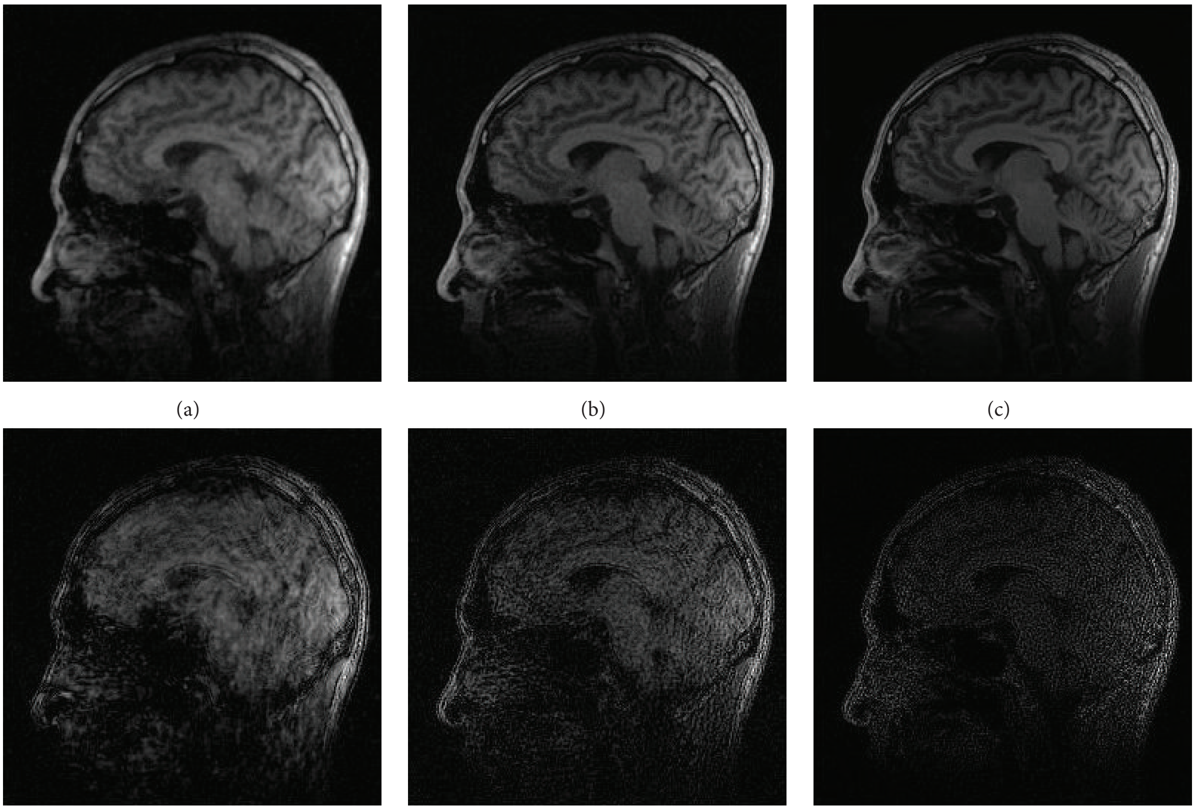
(c) Iter. = 20

(d) Iter. = 1

(e) Iter. = 5

(f) Iter. = 20

Figure 4: (a)–(c) Reconstruction of data 1 by FBOSS at different iterations when the undersampling factor equals 10. (d)–(f) Absolute error images corresponding to (a)–(c).



(a)

(b)

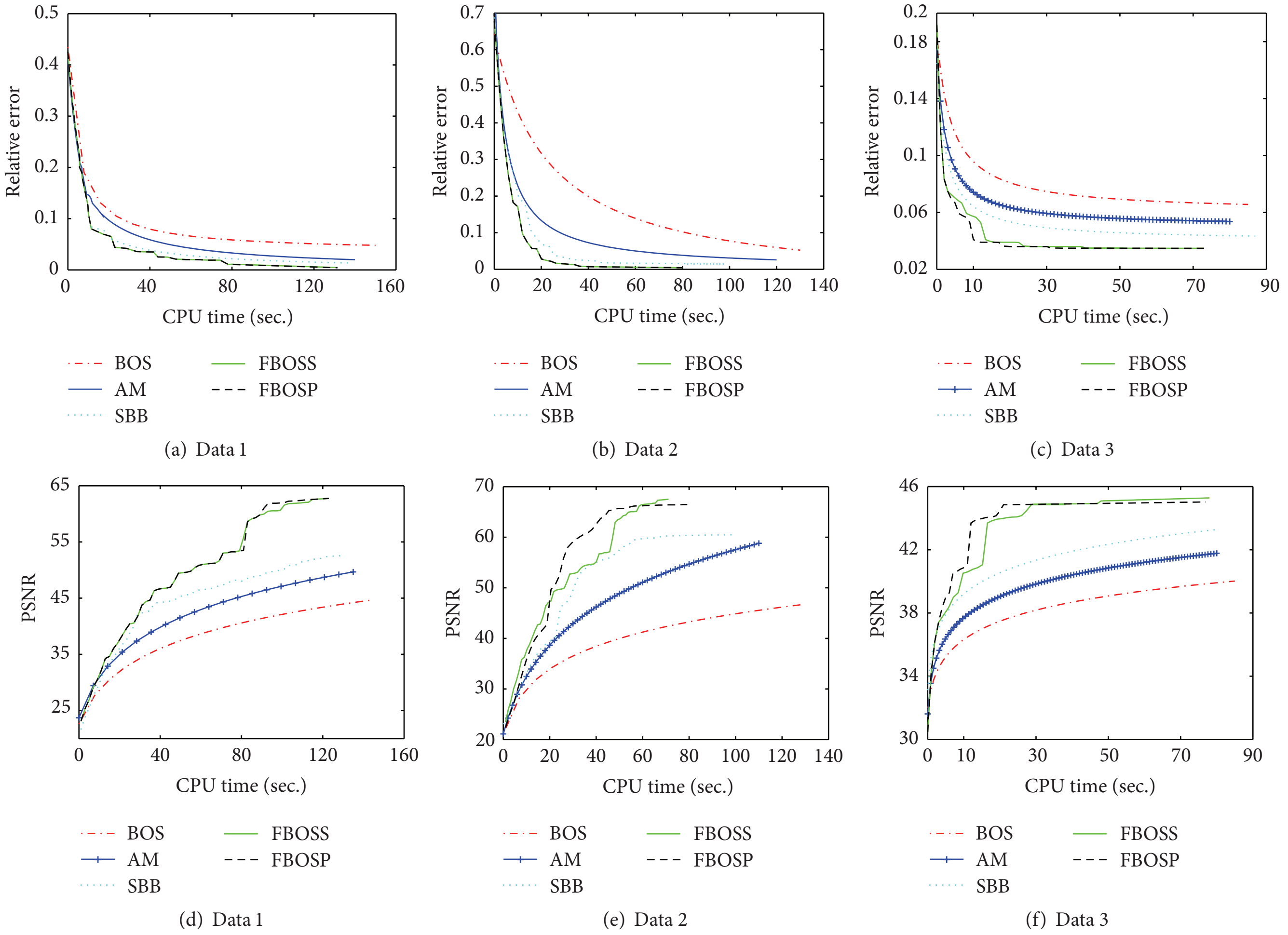
(c)

(d)

(e)

(f)

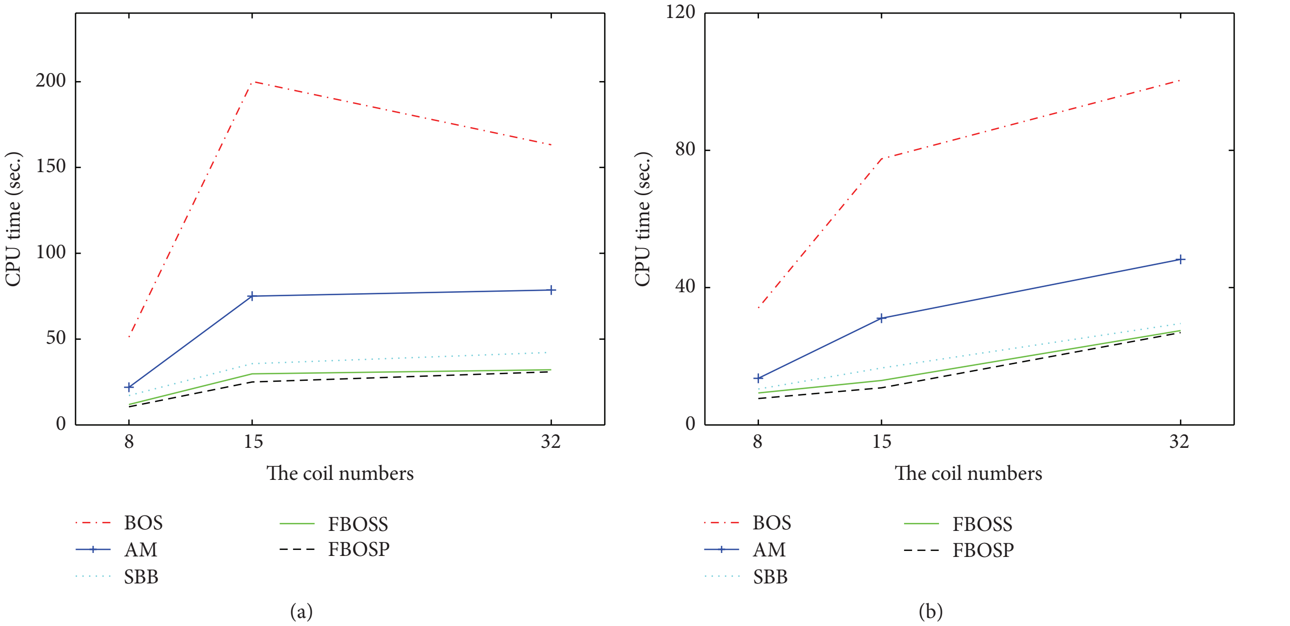
Figure 5: (a)–(c) Reconstruction of data 1 by FBOSP at different iterations when the undersampling factor equals 10. (d)–(f) Absolute error images corresponding to (a)–(c).



(a) Data 1 (b) Data 2 (c) Data 3

(d) Data 1 (e) Data 2 (f) Data 3

Figure 6: The performance comparisons of the five algorithms for the TV-based SparseSENSE model when the undersampling factors for data 1, data 2, and data 3 are 10, 6, and 10, respectively. (a, b, c) Relative error versus CPU time of data 1, data 2, and data 3. (d, e, f) PSNR versus CPU time of data 1, data 2, and data 3.



(a) (b)

Figure 7: The CPU time of the five algorithms for the TV-based SparseSENSE model as functions of the coil numbers on different undersampling factors. (a) CPU time versus the coil numbers on the 6-fold undersampling factor; (b) CPU time versus the coil numbers on the 4-fold undersampling factor.

Table 3: Numerical results for the TGV-based SparseSENSE model.

	Undersampling factor	Performance metric	AM	FBOSP
Data 1	10	PSNR (dB)	47.37	49.62
		Relative error ($e-2$)	2.87	2.21
		CPU time (sec.)	142.2	60.0
Data 2	6	PSNR (dB)	54.23	55.95
		Relative error ($e-2$)	1.48	1.22
		CPU time (sec.)	78.4	28.1
Data 3	4	PSNR (dB)	52.01	53.34
		Relative error ($e-2$)	1.56	1.34
		CPU time (sec.)	19.8	12.0

harsh situation where the data is of large-scale and highly undersampled.

4.2. Comparisons on the Second-Order TGV-Based SparseSENSE Model. In this subsection, we only compared FBOSP with AM based on a special second-order TGV reconstruction model to demonstrate the performance of the proposed algorithms for general D. This is because BOS and SBB cannot work well for the general sparse transform D and FBOSP has an inexact connection to AM.

The second-order total generalized variation [38] is defined as

$$\mathrm{TVG}_{\alpha}^{2}(x)=\sup\left\{\int_{\Omega}x\,\mathrm{div}^{2}w\,dx,w\in X^{*}\right\},\qquad(19)$$

where $X^{*}=\{w\mid w\in C_{c}^{k}(\Omega,\mathrm{Sym}^{2}(\mathbb{R}^{n})),\|\mathrm{div}^{l}v\|_{\infty}\leq a_{l},l=0,1\}$, $C_{c}^{k}(\Omega,\mathrm{Sym}^{2}(\mathbb{R}^{n}))$ is the space of compactly supported symmetric tensor field, and $\mathrm{Sym}^{2}(\mathbb{R}^{n})$ is the space of symmetric tensors on $\mathbb{R}^{n}$. By taking $\alpha_{1}=+\infty$, $\alpha_{0}=1$, a special form of second-order TGV can be written as

$$\|Dx\|_{1}=\left\|\begin{matrix}D_{11}x & \dfrac{(D_{12}x+D_{21}x)}{2}\\ \dfrac{(D_{12}x+D_{21}x)}{2} & D_{22}x\end{matrix}\right\|_{1},\qquad(20)$$

where D_{ij} $(i,j=1,2)$ is the second-order finite difference and 1 and 2 represent the horizontal and vertical directions, respectively. Obviously, operator D in (20) cannot be diagonalized by fast transform. Therefore, BOS and SBB are absent in the comparisons as they are not efficient for the special TGV SparseSENSE model.

For the comparisons between FBOSP and AM, we test data 1, data 2, and data 3 corresponding to undersampling factors 10, 6, and 4, respectively. Table 3 demonstrates the superiority of FBOSP. The reconstructed and error images for AM and FBOSP on data 3 are shown in Figure 8. In addition, we give the reconstructed and error images at different iterations to demonstrate that FBOSP is able to improve image quality iteratively (shown in Figure 9). As shown in Figures 10(a), 10(b), and 10(c), the proposed algorithm FBOSP keeps its superiority in convergence speed.

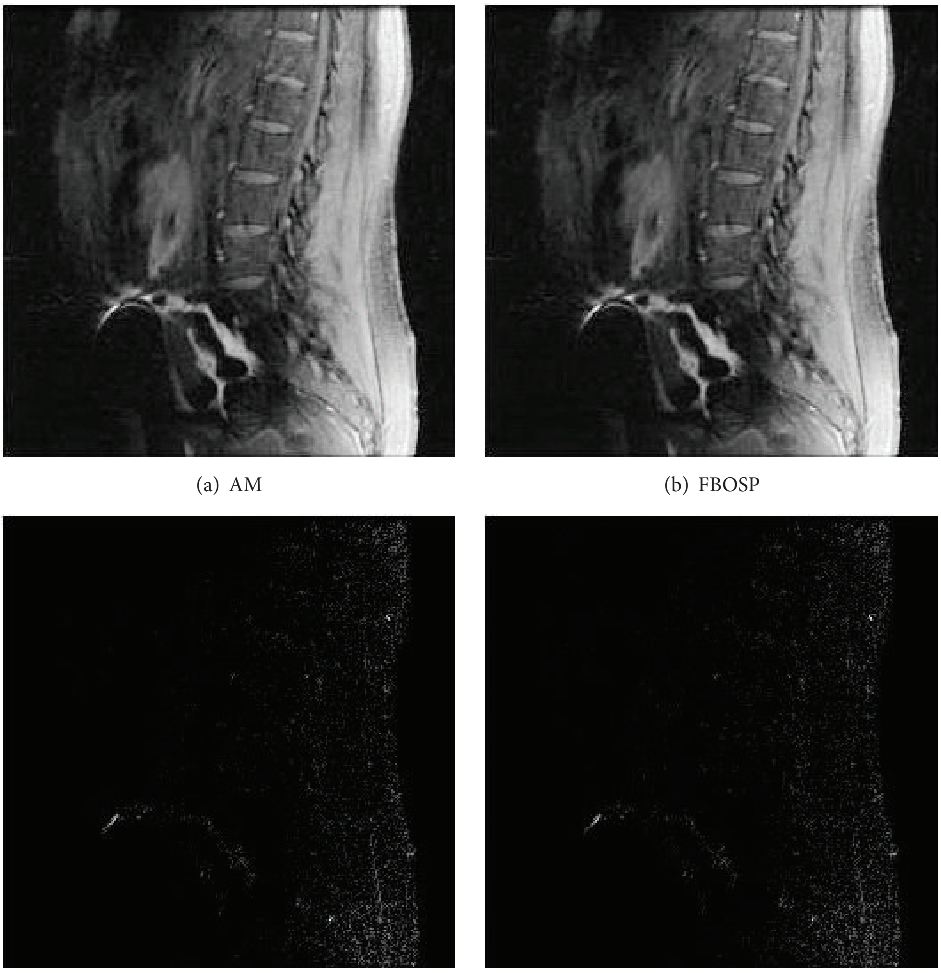
5. Discussions

5.1. Parameters Selection. In this subsection, we discuss the effect of the involved parameters γ and λ on the performance of the proposed algorithms for the TV-based SparseSENSE model. Here, data 1 is employed for the test. We plot the relative error as the function of CPU time under the different parameters for FBOSS and FBOSP. As shown in Figure 11(a), FBOSS with smaller γ or larger λ seems to have better performance than other parameter combinations. The reason is that for a small γ the information of gradient image is not easy to lose in the shrinkage threshold procedure. Meanwhile, x^{k} can better approximate the optimal solution when λ is sufficiently large. In contrast, FBOSP has nearly stable performance under different parameters. Therefore, we selected the parameters of FBOSS for moderated performance and fix the parameters of FBOSP in the experiments.

5.2. Connection to the Existing Methods. It is known that the main computational steps of BOS (or SBB) and AM are the splitting Bregman iteration, which are equivalent to ADMM and the PDHG iterations, respectively. In this subsection, we analyze two relations between the proposed algorithms and the existing popular methods: (1) FBOSS and ADMM and (2) FBOSP and PDHG. For simplicity, we simplify (1) by replacing A with an identity operator Id.

(1) Relation between FBOSS and ADMM. FBOSS can be interpreted as an ADMM method with a semi-implicit scheme applied to the augmented Lagrangian of the original problem as follows:

$$\begin{aligned}L(x,s;w)=&\left(\|s\|_{1}-\langle w,s-Dx\rangle+\frac{\beta}{2}\|s-Dx\|_{2}^{2}\right)\\&+\frac{\lambda}{2}\|x-y\|_{2}^{2},\end{aligned}\qquad(21)$$



(a) AM

(b) FBOSP

(c) AM

(d) FBOSP

Figure 8: (a)-(b) Reconstructed images with relative error, 1.56% and 1.34%, separately on data 3. (c)-(d) Absolute error images corresponding to (a)-(b).

where w is Lagrangian multiplier. By using ADMM for $L(x;s;w)$, the solution to problem (1) ($A = I$) can be gained from the following sequence:

$$\begin{aligned} s^{k+1} &= \arg\min_{s} \|s\|_1 + \frac{\beta}{2}\left\|s - Dx^k - \beta^{-1}w^k\right\|_2^2, \\ w^{k+1} &= w^k + \beta\left(Dx^k - s^{k+1}\right), \\ x^{k+1} &= \arg\min_{x} \frac{\beta}{2}\left\|s^{k+1} - Dx - \beta^{-1}w^{k+1}\right\|_2^2 \\ &\quad + \frac{\lambda}{2}\|x - y\|_2^2. \end{aligned} \tag{22}$$

If $\beta = \gamma^{-1}$, FBOSS is almost equivalent to ADMM, except that the former employs a semi-implicit scheme for the x-subproblem, while the latter employs an implicit scheme. That is why the proposed algorithm performs better than the SBB method although SBB also adopts the BB stepsize scheme.

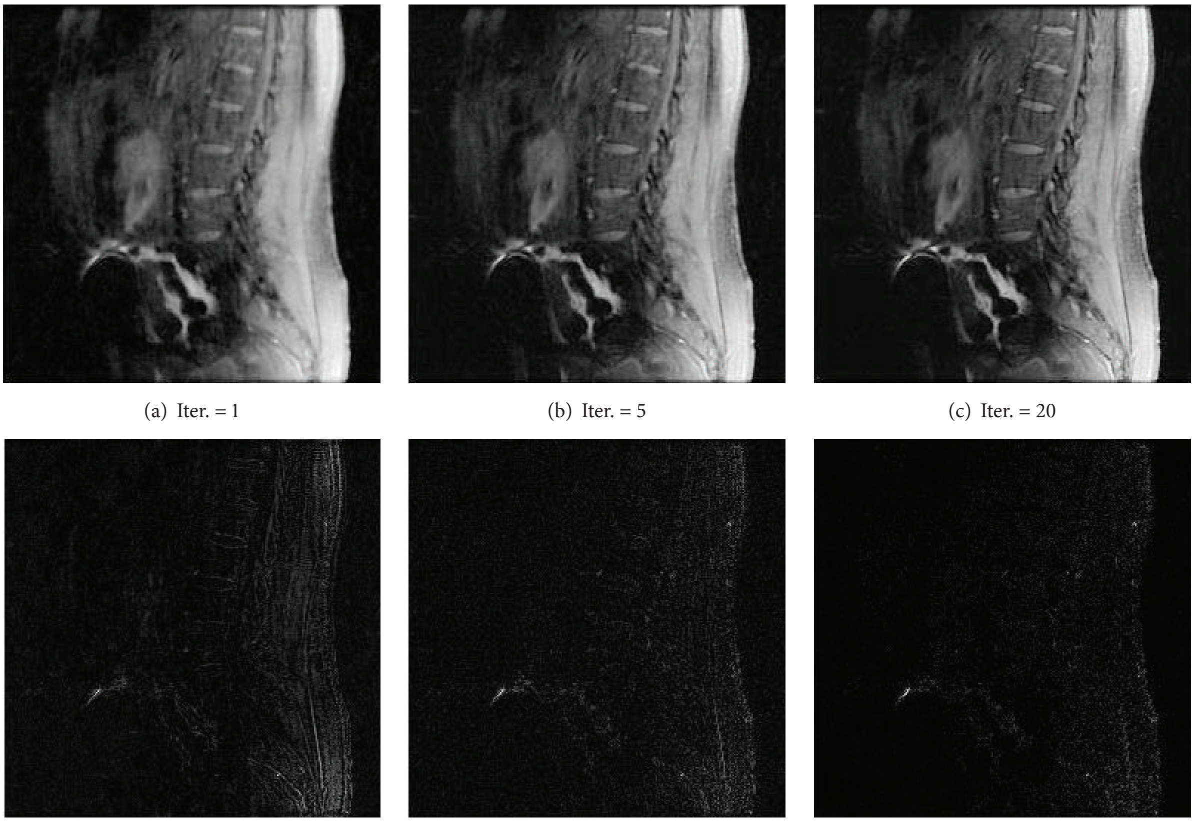
(2) Relation between FBOSP and PDHG. FBOSP based on the projection gradient method can be regarded as a PDHG technique without the proximal term in the gradient descent step, applied to the primal-dual formulation:

$$\min_{x} \max_{w \in X'} \Phi(x, w) := \left\langle x, D^T w \right\rangle + \frac{\lambda}{2}\|x - y\|_2^2, \tag{23}$$

where $w \in X'$ is dual variable. According to the PDHG iteration, the max-min problem (23) is solved by the update of the sequence:

$$\begin{aligned} w^{k+1} &= \arg\max_{w \in X} \Phi\left(x^k, w\right) - \frac{\gamma}{2}\left\|w - w^k\right\|_2^2, \\ x^{k+1} &= \arg\max_{x} \Phi\left(x, w^{k+1}\right) + \frac{\theta}{2}\left\|x - x^k\right\|_2^2. \end{aligned} \tag{24}$$

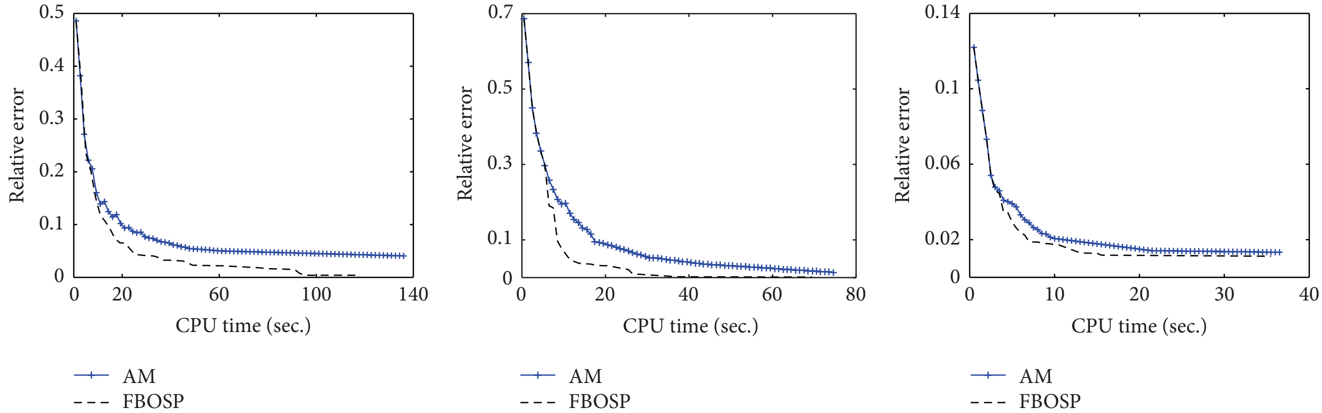
By replacing w with t and taking $\theta = 0$ in the above iteration, it is the main procedure of the FBOSP method for minimization (1).



(a) Iter. = 1 (b) Iter. = 5 (c) Iter. = 20

(d) Iter. = 1 (e) Iter. = 5 (f) Iter. = 20

FIGURE 9: (a, b, c) Reconstruction of data 1 by FBOSP at different iteration. (d, e, f) Absolute error images corresponding to (a, b, c).

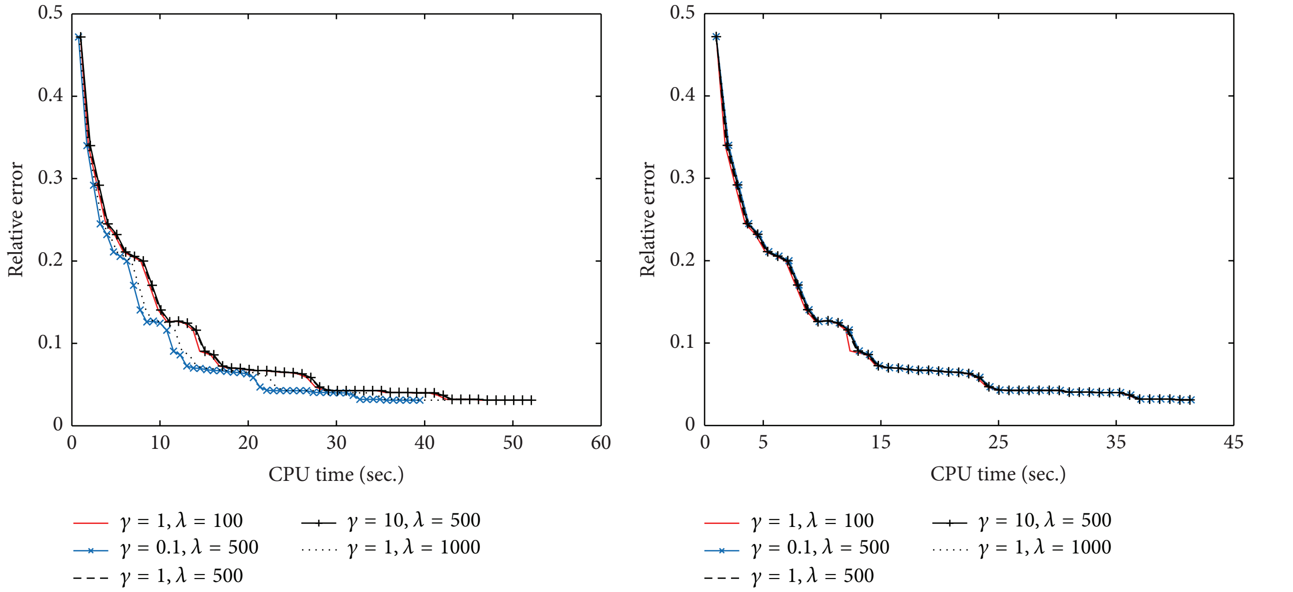


(a) Data 1 (b) Data 2 (c) Data 3

FIGURE 10: The performance comparisons of AM and FBOSP for the second-order TGV-based SparseSENSE model when the undersampling factors for data 1, data 2, and data 3 are 10, 6, and 4, respectively. (a)–(c) Relative error versus CPU time of data 1, data 2, and data 3.

6. Conclusions

In this paper, two fast numerical algorithms based on the forward-backward splitting scheme are proposed for solving the SparseSENSE model. Both of them effectively overcome the difficulty brought by the nondifferentiable l_1 norm according to the fine property of proximity operator. They also avoid solving a partial differential equation with huge computation, which has to be solved in the conventional gradient descent methods. The proposed algorithms can treat the general matrix operators $D^T D$ and $A^T A$ that may be not diagonalized by the fast Fourier transform. We compare them with Bregman algorithms [24, 29] and the alternating minimization method [3]. The results show that the proposed algorithms are efficient for solving the SparseSENSE model quickly.



(a)

(b)

Figure 11: The performance comparisons of different parameters. (a) Relative error versus CPU time on FBOSS; (b) relative error versus CPU time on FBOSP.

Appendix

To illustrate the convergence of the presented algorithms, we show that the proposed algorithm FBOSS with the constant stepsize δ converges by utilizing the nonexpansivity of proximity operator in this section. Based on the connection between FBOSS and FBOSP, it is easy to deduce the convergence of FBOSP with the constant stepsize δ because the projection operator keeps the contractive property. First, the related theories about the proximity operator are given as follows.

Lemma A.1. *For the given convex function φ and its conjugate function φ^*, it holds for any $a, b \in \mathbb{R}^d$ that*

$$\begin{aligned}\left\|\operatorname{Prox}_{\gamma\varphi}(a) - \operatorname{Prox}_{\gamma\varphi}(b)\right\|_2^2 &\leq \|a-b\|_2^2 \\ &\quad - \left\|\left(I - \operatorname{Prox}_{\gamma\varphi}\right)(a) - \left(I - \operatorname{Prox}_{\gamma\varphi}\right)(b)\right\|_2^2.\end{aligned} \tag{A.1}$$

Next, we give the main conclusion for the convergence of FBOSS with constant δ under the nonexpansivity of proximity operator.

Theorem A.2. *Assume $\delta \geq \|A^TA\| \geq (1/2\lambda\gamma)\|D^TD\|$, the sequence $\{(s^{k+1}, w^{k+1}, x^{k+1})\}$ generated by Algorithm 1 with constant δ converges to the point (s^*, w^*, x^*), where x^* is the solution to the SparseSENSE model.*

```
Input $\lambda$, $\gamma$ and $\delta$. Set $x^0 = A^T y$, $w^0 = 0$ and $\delta_0 = 1$.
Repeat
    Given $x^k$ and $\delta_k$, compute $z^{k+1}$;
    Given $w^k$ and $x^k$, compute $s^{k+1}$;
    Given $w^k$, $x^k$ and $s^{k+1}$, compute $w^{k+1}$;
    Given $z^{k+1}$, $w^{k+1}$ and $\delta_k$, compute $x^{k+1}$;
    Given $x^k$ and $x^{k+1}$, compute $\delta_{k+1}$;
    $k \leftarrow k + 1$
until $\|x^k - x^{k-1}\|/\|x^k\| < \varepsilon$
return
```

Algorithm 1: FBOSS for the SparseSENSE model.

Proof. Suppose that the point (s^*, w^*, x^*) satisfies (14), according to (16) and Lemma A.1, we have

$$\begin{aligned}&\left\|\gamma\left(w^{k+1} - w^*\right)\right\|_2^2 \\ &= \left\|\operatorname{Prox}_{\gamma\varphi^*}\left(\gamma w^k + Dx^k\right) - \operatorname{Prox}_{\gamma\varphi^*}\left(\gamma w^* + Dx^*\right)\right\|_2^2 \\ &\leq \left\|\gamma\left(w^k - w^*\right) + D\left(x^k - x^*\right)\right\|_2^2 \\ &= \left\|\gamma\left(w^k - w^*\right)\right\|_2^2 + 2\left\langle\gamma\left(w^k - w^*\right), D\left(x^k - x^*\right)\right\rangle \\ &\quad + \left\|D\left(x^k - x^*\right)\right\|_2^2.\end{aligned} \tag{A.2}$$

In addition, by combining the first and the fourth equations in (14), we get

$$\begin{aligned}\frac{1}{\lambda\gamma}\left\|\gamma\left(w^{k+1}-w^{*}\right)\right\|_{2}^{2}+\left\|x^{k}-x^{*}\right\|_{H}^{2}&\\ \leq\frac{1}{\lambda\gamma}\left\|\gamma\left(w^{k}-w^{*}\right)\right\|_{2}^{2}+\left\|x^{k-1}-x^{*}\right\|_{H}^{2}&\\ -\left\|x^{k-1}-x^{k}\right\|_{H}^{2}&\\ -\left\langle\left(2A^{T}A-\frac{1}{\lambda\gamma}D^{T}D\right)\left(x^{k}-x^{*}\right),x^{k}-x^{*}\right\rangle&\\ \leq\frac{1}{\lambda\gamma}\left\|\gamma\left(w^{k}-w^{*}\right)\right\|_{2}^{2}+\left\|x^{k-1}-x^{*}\right\|_{H}^{2}&\\ -\left\|x^{k-1}-x^{k}\right\|_{H}^{2}.&\end{aligned} \tag{A.3}$$

Let $(1/\lambda\gamma)\|\gamma(w^{k+1}-w^{*})\|_{2}^{2}+\|x^{k}-x^{*}\|_{H}^{2}=M_{k+1}$, then $M_{1}\geq M_{2}\geq\cdots$ and $M_{k+1}\geq0$ for all k. Hence, $\lim_{k\to\infty}M_{k}$ exists, which implies that

$$\lim_{k\to\infty}\left\|x^{k-1}-x^{k}\right\|_{H}^{2}=0. \tag{A.4}$$

It leads to $\lim_{k\to\infty}x^{k}=x^{*}$, which deduces that

$$\lim_{k\to\infty}\frac{\gamma}{\lambda}\left\|w^{k+1}-w^{*}\right\|_{2}^{2}=\lim_{k\to\infty}M_{k+1}. \tag{A.5}$$

Moreover, it follows from (14), (16), and Lemma A.1 that

$$\begin{aligned}\left\|\mathrm{Prox}_{\gamma\varphi}\left(\gamma w^{k}+Dx^{k}\right)-\mathrm{Prox}_{\gamma\varphi}\left(\gamma w^{*}+Dx^{*}\right)\right\|_{2}^{2}&\\ \leq\left\|\gamma\left(w^{k}-w^{*}\right)+D\left(x^{k}-x^{*}\right)\right\|_{2}^{2}&\\ -\left\|\left(I-\mathrm{Prox}_{\gamma\varphi}\right)\left(\gamma w^{k}+Dx^{k}\right)\right.&\\ \left.-\left(I-\mathrm{Prox}_{\gamma\varphi}\right)\left(\gamma w^{*}+Dx^{*}\right)\right\|_{2}^{2}=\left\|\gamma\left(w^{k}-w^{*}\right)\right.&\\ \left.+D\left(x^{k}-x^{*}\right)\right\|_{2}^{2}-\left\|\gamma\left(w^{k+1}-w^{*}\right)\right\|_{2}^{2}.&\end{aligned} \tag{A.6}$$

Therefore, by taking limits for both sides of the above inequality, we obtain

$$\begin{aligned}\lim_{k\to\infty}\left\|s^{k+1}-s^{*}\right\|_{2}=\lim_{k\to\infty}\left\|\mathrm{Prox}_{\gamma\varphi}\left(\gamma w^{k}+Dx^{k}\right)\right.&\\ \left.-\mathrm{Prox}_{\gamma\varphi}\left(\gamma w^{*}+Dx^{*}\right)\right\|_{2}=\lim_{k\to\infty}\left\|\gamma\left(w^{k}-w^{*}\right)\right.&\\ \left.+D\left(x^{k}-x^{*}\right)\right\|_{2}=\lim_{k\to\infty}\left\|\gamma w^{k}-w^{k+1}\right\|_{2}=0,&\end{aligned} \tag{A.7}$$

which immediately gets

$$\begin{aligned}\lim_{k\to\infty}s^{k}&=s^{*},\\ \lim_{k\to\infty}w^{k}&=w^{*}.\end{aligned} \tag{A.8}$$

On the other hand, by combining (9), (10), and (12), it is easy to conclude that x^{*} is the solution to the SparseSENSE model (1) if point (s^{*},w^{*},x^{*}) satisfies (14). □

```
Input $\lambda$, $\gamma$ and $\delta$. Set $x^0 = A^T y$, $w^0 = 0$ and $\delta_0 = 1$.
Repeat
    Given $x^k$ and $\delta_k$, compute $z^{k+1}$;
    Given $w^k$ and $x^k$, compute $w^{k+1}$;
    Given $z^{k+1}$, $w^{k+1}$ and $\delta_k$ compute $x^{k+1}$;
    Given $x^k$ and $x^{k+1}$, compute $\delta_{k+1}$;
    $k \leftarrow k + 1$
until $\|x^k - x^{k-1}\|/\|x^k\| < \varepsilon$
return
```

Algorithm 2: FBOSP for the SparseSENSE model.

Through the proof for Theorem A.2, we can immediately obtain the following convergence conclusion for Algorithm 2 with constant δ.

Theorem A.3. *Assume $\delta\geq\|A^{T}A\|\geq(1/\lambda\gamma)\|D^{T}D\|$, the sequence $\{(w^{k+1},x^{k+1})\}$ generated by Algorithm 2 with constant δ converges to the point (w^{*},x^{*}), where x^{*} is the solution to the SparseSENSE model.*

Competing Interests

The authors declare that there is no conflict of interests regarding the publication of this article.

Acknowledgments

This research was partly supported by the National Natural Science Foundation of China (nos. 61471350, 11301508, and 61401449), the Natural Science Foundation of Guangdong (nos. 2015A020214019, 2015A030310314, and 2015A030313740), the Guangdong Science and Technology Plan (Grant nos. 2015B010124001 and 2015B090903017), the Basic Research Program of Shenzhen (nos. JCYJ20140610152828678, JCYJ20150630114942318, and JCYJ20140610151856736), and the SIAT Innovation Program for Excellent Young Researchers (no. 201403).

References

[1] K. T. Block, M. Uecker, and J. Frahm, "Undersampled radial MRI with multiple coils. Iterative image reconstruction using a total variation constraint," *Magnetic Resonance in Medicine*, vol. 57, no. 6, pp. 1086–1098, 2007.

[2] M. Bydder and M. D. Robson, "Partial fourier partially parallel imaging," *Magnetic Resonance in Medicine*, vol. 53, no. 6, pp. 1393–1401, 2005.

[3] Y. Chen, W. Hager, F. Huang, D. Phan, X. Ye, and W. Yin, "Fast algorithms for image reconstruction with application to partially parallel MR imaging," *SIAM Journal on Imaging Sciences*, vol. 5, no. 1, pp. 90–118, 2012.

[4] M. A. Griswold, P. M. Jakob, R. M. Heidemann et al., "Generalized Autocalibrating Partially Parallel Acquisitions (GRAPPA)," *Magnetic Resonance in Medicine*, vol. 47, no. 6, pp. 1202–1210, 2002.

[5] C. Liu, J. Zhang, and M. E. Moseley, "Auto-calibrated parallel imaging reconstruction for arbitrary trajectories using k-Space sparse matrices (kSPA)," *IEEE Transactions on Medical Imaging*, vol. 29, no. 3, pp. 950–959, 2010.

[6] D. S. Weller, S. Ramani, and J. A. Fessler, "Augmented lagrangian with variable splitting for faster non-cartesian L1-SPIRiT MR image reconstruction," *IEEE Transactions on Medical Imaging*, vol. 33, no. 2, pp. 351–361, 2014.

[7] J. M. Santos, C. H. Cunningham, M. Lustig et al., "Single breath-hold whole-heart MRA using variable-density spirals at 3t," *Magnetic Resonance in Medicine*, vol. 55, no. 2, pp. 371–379, 2006.

[8] T. Henzler, O. Dietrich, R. Krissak et al., "Half-Fourier-acquisition single-shot turbo spin-echo (HASTE) MRI of the lung at 3 Tesla using parallel imaging with 32-receiver channel technology," *Journal of Magnetic Resonance Imaging*, vol. 30, no. 3, pp. 541–546, 2009.

[9] K. Nael, M. Fenchel, M. Krishnam, J. P. Finn, G. Laub, and S. G. Ruehm, "3.0 Tesla high spatial resolution contrast-enhanced magnetic resonance angiography (CE-MRA) of the pulmonary circulation: initial experience with a 32-channel phased array coil using a high relaxivity contrast agent," *Investigative Radiology*, vol. 42, no. 6, pp. 392–398, 2007.

[10] A. F. Stalder, Z. Dong, Q. Yang et al., "Four-dimensional flow-sensitive MRI of the thoracic aorta: 12-versus 32-channel coil arrays," *Journal of Magnetic Resonance Imaging*, vol. 35, no. 1, pp. 190–195, 2012.

[11] S.-Y. Tsai, R. Otazo, S. Posse et al., "Accelerated proton echo planar spectroscopic imaging (PEPSI) using GRAPPA with a 32-channel phased-array coil," *Magnetic Resonance in Medicine*, vol. 59, no. 5, pp. 989–998, 2008.

[12] M. P. McDougall and S. M. Wright, "64-Channel array coil for single echo acquisition magnetic resonance imaging," *Magnetic Resonance in Medicine*, vol. 54, no. 2, pp. 386–392, 2005.

[13] M. Schmitt, A. Potthast, D. E. Sosnovik et al., "A 128-channel receive-only cardiac coil for highly accelerated cardiac MRI at 3 tesla," *Magnetic Resonance in Medicine*, vol. 59, no. 6, pp. 1431–1439, 2008.

[14] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, no. 3, pp. 969–985, 2007.

[15] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[16] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[17] T. Chang, L. He, and T. Fang, "MR image reconstruction from sparse radial samples using Bregman iteration," in *Proceedings of the 13th Annual Meeting of ISMRM*, p. 696, Seattle, Wash, USA, 2006.

[18] D. Liang, E. V. R. DiBella, R.-R. Chen, and L. Ying, "K-t ISD: dynamic cardiac MR imaging using compressed sensing with iterative support detection," *Magnetic Resonance in Medicine*, vol. 68, no. 1, pp. 41–53, 2012.

[19] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: the application of compressed sensing for rapid MR imaging," *Magnetic Resonance in Medicine*, vol. 58, no. 6, pp. 1182–1195, 2007.

[20] J. C. Ye, S. Tak, Y. Han, and H. W. Park, "Projection reconstruction MR imaging using FOCUSS," *Magnetic Resonance in Medicine*, vol. 57, no. 4, pp. 764–775, 2007.

[21] L. Chaâri, J.-C. Pesquet, A. Benazza-Benyahia, and P. Ciuciu, "A wavelet-based regularized reconstruction algorithm for SENSE parallel MRI with applications to neuroimaging," *Medical Image Analysis*, vol. 15, no. 2, pp. 185–201, 2011.

[22] B. Liu, F. Sebert, Y. Zou et al., "SparseSENSE: randomly-sampled parallel imaging using compressed sensing," in *Proceedings of the 16th Annual Meeting of ISMRM*, p. 3154, Toronto, Canada, 2008.

[23] M. Murphy, M. Alley, J. Demmel, K. Keutzer, S. Vasanawala, and M. Lustig, "Fast ℓ1-SPIRiT compressed sensing parallel imaging MRI: scalable parallel implementation and clinically feasible runtime," *IEEE Transactions on Medical Imaging*, vol. 31, no. 6, pp. 1250–1262, 2012.

[24] X. Ye, Y. Chen, and F. Huang, "Computational acceleration for MR image reconstruction in partially parallel imaging," *IEEE Transactions on Medical Imaging*, vol. 30, no. 5, pp. 1055–1063, 2011.

[25] L. Ying, B. Liu, M. C. Steckner, G. Wu, M. Wu, and S.-J. Li, "A statistical approach to SENSE regularization with arbitrary k-space trajectories," *Magnetic Resonance in Medicine*, vol. 60, no. 2, pp. 414–421, 2008.

[26] E. Esser, "Applications of Lagrangian-based alternating direction methods and connections to split Bregman," *CAM Report*, vol. 9, p. 31, 2009.

[27] S. Ramani and J. A. Fessler, "Parallel MR image reconstruction using augmented lagrangian methods," *IEEE Transactions on Medical Imaging*, vol. 30, no. 3, pp. 694–706, 2011.

[28] T. Goldstein and S. Osher, "The split Bregman method for L1 -regularized problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 323–343, 2009.

[29] X. Zhang, M. Burger, X. Bresson, and S. Osher, "Bregmanized nonlocal regularization for deconvolution and sparse reconstruction," *SIAM Journal on Imaging Sciences*, vol. 3, no. 3, pp. 253–276, 2010.

[30] P. L. Combettes and V. R. Wajs, "Signal recovery by proximal forward-backward splitting," *Multiscale Modeling & Simulation*, vol. 4, no. 4, pp. 1168–1200, 2005.

[31] J. Barzilai and J. M. Borwein, "Two-point step size gradient methods," *IMA Journal of Numerical Analysis*, vol. 8, no. 1, pp. 141–148, 1988.

[32] C. R. Vogel and M. E. Oman, "Iterative methods for total variation denoising," *SIAM Journal on Scientific Computing*, vol. 17, no. 1, pp. 227–238, 1996.

[33] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM Journal on Imaging Sciences*, vol. 1, no. 3, pp. 248–272, 2008.

[34] J. Yang, W. Yin, Y. Zhang, and Y. Wang, "A fast algorithm for edge-preserving variational multichannel image restoration," *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 569–592, 2009.

[35] M. Zhu and T. Chan, "An efficient primal-dual hybrid gradient algorithm for total variation image restoration," *UCLA CAM Report*, pp. 8–34, 2008.

[36] P.-L. Lions and B. Mercier, "Splitting algorithms for the sum of two nonlinear operators," *SIAM Journal on Numerical Analysis*, vol. 16, no. 6, pp. 964–979, 1979.

[37] J. X. Ji, B. S. Jong, and S. D. Rane, "PULSAR: a MATLAB toolbox for parallel magnetic resonance imaging using array coils and multiple channel receivers," *Concepts in Magnetic Resonance Part B: Magnetic Resonance Engineering*, vol. 31, no. 1, pp. 24–36, 2007.

[38] K. Bredies, K. Kunisch, and T. Pock, "Total generalized variation," *SIAM Journal on Imaging Sciences*, vol. 3, no. 3, pp. 492–526, 2010.