

RESEARCH

Open Access



Parallel linearized ADMM with application to multichannel image restoration and reconstruction

Chuan He^{1*} , Wenshen Peng¹, Junwei Wang¹, Xiaowei Feng¹ and Licheng Jiao²

*Correspondence:
hechuan8512@163.com

¹ PLA Rocket Force University
of Engineering, Xi'an 710025,
China

² Xidian University, School
of Artificial Intelligence,
Xi'an 710126, China

Abstract

Many large-scale regularized inverse problems in imaging such as image restoration and reconstruction can be modeled as a generic objective function involves sum of nonsmooth but proximable terms, which are usually linear-operator-coupled. For the solution of these problems, a parallel linearized alternating direction method of multipliers (PLADMM) is proposed in this paper. At each step of the proposed algorithm, the proximity operators of the nondifferential terms are called individually. This leads to a highly parallel algorithm structure, where most sub-steps can be simultaneously solved. Profiting from the linearization step, the linear inverse operation is excluded. The convergence property of the proposed method is analyzed. The image deblurring, inpainting, and pMRI reconstruction experiments show that the proposed method has vast applicable vistas. Compared with the state-of-the-art methods, such as PADMM [21], FTVd-v4 [22], PPDS [33], FUSL [8], LADM [36], and ALADMM [27], it gains competitive results both in terms of quantitative indicators, such as PSNR or SSIM, and in terms of visual impression.

Keywords: Nonsmooth convex optimization, Parallel linearized alternating direction method of multipliers (PLADMM), Image restoration and reconstruction, Total generalized variation (TGV)

1 Introduction

Numerous inverse problems in image processing and computer vision, image restoration [1–3], image denoising [4], inpainting [5, 6], super-resolution [7], and compressive sensing [8–10], e.g., can be formulated as convex optimization problems involving sum of several convex functions, which may be nondifferentiable and coupled with linear operators. The function can be a data fidelity term of the observation as well as a regularization term reflecting the prior knowledge imposed on the solution. In most cases, the regularizer is desired nonsmooth for preferable results, e.g., the l1-based total variation and the wavelet frame.

In this paper, we aim at solving the following generic convex optimization problem:

$$\min_{\mathbf{u} \in U} g(\mathbf{u}) + \sum_{i=1}^I f_i(\mathbf{L}_i \mathbf{u}). \quad (1)$$

Let $\Gamma_0\{U\}$ denote the set of all proper, convex, and lower semicontinuous functions [13] from U to $\mathbb{R} \cup \{+\infty\}$. We define $g \in \Gamma_0\{U\}$ and $f_i \in \Gamma_0\{V_i\}$, and they have closed-form or simply solvable proximity operators. The bounded linear operator $\mathbf{L}_i(U \rightarrow V_i)$ has the induced norm $\|\mathbf{L}_i\| = \sup\{\|\mathbf{L}_i \mathbf{u}\|_2 : \|\mathbf{u}\|_2 = 1\} < +\infty$ and adjoint operator noted by $\mathbf{L}_i^*$. The set of minimizers of (1) is supposed to be nonempty and this assumption is generally rational for practical applications. Formulation (1) is generic for covering many imaging inverse problems, such as deblurring, inpainting, compressive sensing, etc. Note that some terms in (1) may be zero for a specific problem. With the indicator function $\iota_\Omega(\mathbf{u})$ of a convex set Ω in U (if $\mathbf{u} \in \Omega$, then $\iota_\Omega = 0$, or else, $\iota_\Omega = +\infty$), we can transform the constrained optimizing problem $\left\{ \min_{\mathbf{u} \in U} g(\mathbf{u}) \text{ s.t. } \mathbf{u} \in \Omega \right\}$ into an unconstrained form $\min_{\mathbf{u} \in U} \{g(\mathbf{u}) + \iota_\Omega(\mathbf{u})\}$. Thus, formulation (1) is generic for many inverse problems, either they have an unconstrained form or have a constrained form.

Difficulties from two aspects usually make the solution of (1) more difficult. On the one hand, U and V_i are high-dimensional in many imaging applications. On the other hand, the function g and f_i are usually nonsmooth and the classical gradient descent method becomes invalid. Fortunately, the operator splitting method provide a viable approach for the efficient solution of (1) [11].

Up to now, for the solution of ill-conditioned inverse problems, several operator splitting methods have been proposed, such as Douglas–Rachford splitting [12], forward–backward splitting [13], the Bregman splitting [13], alternating direction method of multipliers [14], and linearized ADMM [27], etc. However, these methods are designed to solve functions with two terms in general. This limits the application prospect of these methods in complex image restoration and reconstruction problems, such as multichannel deblurring and parallel magnetic resonance imaging reconstruction. Their extension to parallel splitting to solve more difficult problems like (1) is of great importance for many large-scale imaging applications.

For the efficient solution of Eq. (1) and for effective solution of complex image restoration and reconstruction problems, in conference paper [1], a parallel linearized alternating direction method of multipliers (PLADMM) based on the basic LADMM [12, 15] is derived, a.k.a. linear Bregman splitting method [16, 17]. The following advantages make the proposed method more appealing compared with the existent splitting methods. First, it possesses a "full splitting" structure. At each step, only the proximity operators of g and f_i (or g^* and f_i^*), and the linear operator $\mathbf{L}_i$ (or its conjugate $\mathbf{L}_i^*$) are referred to. This makes the proposed method highly parallel in structure, so that the parallel calculation can be adopted for acceleration. Second, each $\mathbf{L}_i$ is assigned a different weight, which makes the method more flexible in practice. Third, with linearization, the linear inverse operator commonly existent in the previous methods solving inverse problems [14, 18–22] is avoided, and thus, the one-step efficiency is promoted. However, the convergence behavior and how to use the characteristics of the background problems to tap the parallelism potential of ADMM [23, 24] deserves in-depth study.

In conference paper [1], we proposed the parallel linearized ADMM (PLADMM) with a highly parallel structure which excludes linear inverse operation and applied it to color image deblurring. In contrast, the contribution of this paper is as follows. First, a rigorous convergence behavior analysis, including convergence proof and $O(1/k)$ convergence rate calculation, is presented based on the variational inequality, which is a tool usually used in the convergence analysis of operator splitting-based methods. Second, we further extend PLADMM to other image inverse problems, such as inpainting and pMRI. Experiment results show that the proposed algorithm is vast applicable and is competitive compared with the state-of-the-art methods.

In the following, the derivation of the proposed method and the detailed convergence behavior analysis are given in Sect. 2. In Sect. 3, we apply our method to the multichannel image inverse problem with both box constraints and discrepancy-based feasible constraint. In Sect. 4, the proposed method is compared with several famous algorithms. Experiment results of image deblurring, inpainting, and pMRI reconstruction demonstrate the superiority and potential of the proposed algorithm. Finally, we conclude this paper and point out the future work in Sect. 5.

2 PLADMM algorithm

The definition of the set-valued subdifferential of $g\{\mathbf{u}\} \in \Gamma_0\{U\}$ at $\mathbf{u}$ is as follows:

$$\partial g(\mathbf{u}) = \{\mathbf{a} \in U | (\forall \mathbf{u}' \in U) \langle \mathbf{u}' - \mathbf{u}, \mathbf{a} \rangle + g(\mathbf{u}) \leq g(\mathbf{u}')\}. \quad (2)$$

We denote the Fenchel conjugate of a convex function as

$$g^*(\mathbf{u}) = \sup_{\mathbf{u}' \in U} \{\langle \mathbf{u}, \mathbf{u}' \rangle - g(\mathbf{u}')\}. \quad (3)$$

We denote the proximity operator of $g \in \Gamma_0\{U\}$ as

$$\text{prox}_{\lambda g}(\mathbf{u}) : \mathbf{u} \rightarrow \arg \min_{\mathbf{u}' \in U} \left\{ g(\mathbf{u}') + \frac{1}{2\lambda} \|\mathbf{u} - \mathbf{u}'\|_2^2 \right\}. \quad (4)$$

The proximity operator of the indicator function ι_Ω is simply the projector P_Ω onto the set Ω . Thus, the proximity operator is usually seen as the generalization of the projection operator.

With the above definitions, we get the Moreau decomposition [13] as follows:

$$\mathbf{u} = \text{prox}_{\beta g^*} \mathbf{u} + \beta \text{prox}_{g/\beta}(\mathbf{u}/\beta). \quad (5)$$

2.1 Saddle point condition

We first introduce an auxiliary variable $\mathbf{b}_i \in V_i$ for every $\mathbf{L}_i \mathbf{u}$, i.e., $\mathbf{b}_i = \mathbf{L}_i \mathbf{u}$. Then, the augmented Lagrangian (AL) function of (1) is as follows:

$$\mathcal{L}_{\mathcal{A}}(\mathbf{u}, \mathbf{b}_1, \dots, \mathbf{b}_I; \mathbf{v}_1, \dots, \mathbf{v}_I) = g(\mathbf{u}) + \sum_{i=1}^I \left(f_i(\mathbf{b}_i) + \langle \mathbf{v}_i, \mathbf{L}_i \mathbf{u} - \mathbf{b}_i \rangle + \frac{\beta_i}{2} \|\mathbf{L}_i \mathbf{u} - \mathbf{b}_i\|_2^2 \right), \quad (6)$$

where $\beta_i > 0$ is the penalty parameter and $\mathbf{v}_i \in V_i$ is the dual variable. Unlike the classical ADMM, different β_i is assigned to each $\mathbf{L}_i \mathbf{u}$. The following application will indicate

that this makes the proposed method more flexible and the convergence behavior will not be ruined.

A point $(\mathbf{u}^*, \mathbf{b}_1^*, \dots, \mathbf{b}_I^*; \mathbf{v}_1^*, \dots, \mathbf{v}_I^*)$ is called a saddle point of (6) if

$$\begin{aligned} & \mathcal{L}_{\mathcal{A}}(\mathbf{u}^*, \mathbf{b}_1^*, \dots, \mathbf{b}_I^*; \mathbf{v}_1, \dots, \mathbf{v}_I) \mathcal{L}_{\mathcal{A}}(\mathbf{u}^*, \mathbf{b}_1^*, \dots, \mathbf{b}_I^*; \mathbf{v}_1^*, \dots, \mathbf{v}_I^*) \\ & \mathcal{L}_{\mathcal{A}}(\mathbf{u}, \mathbf{b}_1, \dots, \mathbf{b}_I; \mathbf{v}_1^*, \dots, \mathbf{v}_I^*) \\ & \forall (\mathbf{u}, \mathbf{b}_1, \dots, \mathbf{b}_I; \mathbf{v}_1, \dots, \mathbf{v}_I) \in U \times V_1 \times \dots \times V_I \times V_1 \times \dots \times V_I. \end{aligned} \quad (7)$$

Lemma 1 builds the bridge from the saddle point of (7) to the solution of (1). The proof of this lemma can be found in [21].

Lemma 1 *The sufficient and necessary condition of that $\mathbf{u}^* \in U$ is a solution of (1), is that there exist $\mathbf{b}_i^*, \mathbf{v}_i^* \in V_i, i = 1, \dots, I$, such that $(\mathbf{u}^*, \mathbf{b}_1^*, \dots, \mathbf{b}_I^*; \mathbf{v}_1^*, \dots, \mathbf{v}_I^*)$ is a saddle point of the AL (7).*

2.2 Derivation of PLADMM

For the convenience of derivation, we transform (6) into

$$\begin{aligned} & \mathcal{L}'_{\mathcal{A}}(\mathbf{u}, \sqrt{\beta_1} \mathbf{b}_1, \dots, \sqrt{\beta_I} \mathbf{b}_I; \frac{\mathbf{v}_1}{\sqrt{\beta_1}}, \dots, \frac{\mathbf{v}_I}{\sqrt{\beta_I}}) = \\ & g(\mathbf{u}) + \sum_{i=1}^I \left(\bar{f}_i(\sqrt{\beta_i} \mathbf{b}_i) + \left\langle \frac{\mathbf{v}_i}{\sqrt{\beta_i}}, \sqrt{\beta_i} \mathbf{L}_i \mathbf{u} - \sqrt{\beta_i} \mathbf{b}_i \right\rangle \right) \\ & + \sum_{i=1}^I \left(\frac{1}{2} \|\sqrt{\beta_i} \mathbf{L}_i \mathbf{u} - \sqrt{\beta_i} \mathbf{b}_i\|_2^2 \right). \end{aligned} \quad (8)$$

We denote $\mathbf{b} = (\sqrt{\beta_1} \mathbf{b}_1, \dots, \sqrt{\beta_I} \mathbf{b}_I) \in V \triangleq V_1 \times \dots \times V_I$, $\mathbf{v} = (\mathbf{v}_1/\sqrt{\beta_1}, \dots, \mathbf{v}_I/\sqrt{\beta_I}) \in V$, $\mathbf{L}\mathbf{u} = (\sqrt{\beta_1} \mathbf{L}_1 \mathbf{u}, \dots, \sqrt{\beta_I} \mathbf{L}_I \mathbf{u}) \in V$, and $f(\mathbf{b}) = \sum_{i=1}^I f_i(\mathbf{b}_i) = \sum_{i=1}^I \bar{f}_i(\sqrt{\beta_i} \mathbf{b}_i)$. As a result, we obtain $\mathbf{L}^* \mathbf{v} = \mathbf{L}_1^* \mathbf{v}_1 + \dots + \mathbf{L}_I^* \mathbf{v}_I = \sum_{i=1}^I \mathbf{L}_i^* \mathbf{v}_i$. With these notations, we rewrite (6) as

$$L_A(\mathbf{u}, \mathbf{b}; \mathbf{v}) = g(\mathbf{u}) + f(\mathbf{b}) + \langle \mathbf{v}, \mathbf{L}\mathbf{u} - \mathbf{b} \rangle + \frac{1}{2} \|\mathbf{L}\mathbf{u} - \mathbf{b}\|_2^2. \quad (9)$$

Therefore, the saddle point condition (7) is equivalent to

$$\begin{aligned} & L_A(\mathbf{u}^*, \mathbf{b}^*; \mathbf{v}) \leq L_A(\mathbf{u}^*, \mathbf{b}^*; \mathbf{v}^*) \leq L_A(\mathbf{u}, \mathbf{b}; \mathbf{v}^*) \\ & \forall (\mathbf{u}, \mathbf{b}; \mathbf{v}) \in U \times V \times V. \end{aligned} \quad (10)$$

To find the saddle point of (9), we refer to the following ADMM framework:

$$\begin{cases} \mathbf{u}^{k+1} = \arg \min_{\mathbf{u}} g(\mathbf{u}) + \frac{1}{2} \|\mathbf{L}\mathbf{u} - \mathbf{b}^k + \mathbf{v}^k\|_2^2; \\ \mathbf{b}^{k+1} = \arg \min_{\mathbf{b}} f(\mathbf{b}) + \frac{1}{2} \|\mathbf{L}\mathbf{u}^{k+1} - \mathbf{b} + \mathbf{v}^k\|_2^2; \\ \mathbf{v}^{k+1} = \mathbf{v}^k + \mathbf{L}\mathbf{u}^{k+1} - \mathbf{b}^{k+1}. \end{cases} \quad (11)$$

Elaborating Eq. (11), we obtain the following Parallel ADMM (PADMM) iterative scheme [21]:

$$\begin{cases} \mathbf{u}^{k+1} = \arg \min_{\mathbf{u}} g(\mathbf{u}) + \sum_{i=1}^I \frac{\beta_i}{2} \left\| \mathbf{L}_i \mathbf{u} - \mathbf{b}_i^k + \frac{\mathbf{v}_i^k}{\beta_i} \right\|_2^2; \\ \mathbf{b}_i^{k+1} = \text{prox}_{f_i/\beta_i} \left(\mathbf{L}_i \mathbf{u}^{k+1} + \frac{\mathbf{v}_i^k}{\beta_i} \right), i = 1, \dots, I; \\ \mathbf{v}_i^{k+1} = \mathbf{v}_i^k + \beta_i (\mathbf{L}_i \mathbf{u}^{k+1} - \mathbf{b}_i^{k+1}), i = 1, \dots, I. \end{cases} \quad (12)$$

The PADMM algorithm is of high efficiency for many inverse problems [21]. However, the linear inverse operator like $(\sum \mathbf{L}_i^* \mathbf{L}_i)^{-1}$ in it usually suffers from vast computation cost. For instance, in multichannel image deblurring, the solution of matrix inverse operation could not be solved efficiently by FFT due to a cross-channel blur. In contrast with PADMM, we linearize the quadratic term around $\mathbf{u}^k$ [12] in the $\mathbf{u}$ subproblem of (12) by

$$\begin{aligned} \mathbf{u}^{k+1} &= \arg \min_{\mathbf{u}} g(\mathbf{u}) + \frac{1}{2t} \left\| \mathbf{u} - \mathbf{u}^k \right\|_2^2 + \left\langle \mathbf{u} - \mathbf{u}^k, \sum_{i=1}^I \beta_i \mathbf{L}_i^* \left(\mathbf{L}_i \mathbf{u}^k - \mathbf{b}_i^k + \frac{\mathbf{v}_i^k}{\beta_i} \right) \right\rangle \\ &= \arg \min_{\mathbf{u}} g(\mathbf{u}) + \frac{1}{2t} \left\| \mathbf{u} - \mathbf{u}^k + t \sum_{i=1}^I \beta_i \mathbf{L}_i^* \left(\mathbf{L}_i \mathbf{u}^k - \mathbf{b}_i^k + \frac{\mathbf{v}_i^k}{\beta_i} \right) \right\|_2^2. \end{aligned} \quad (13)$$

Then, the update of $\mathbf{u}$ can be easily solved by the proximity operation of g . Our alternating direction scheme to find the saddle point of (6) can be described as

$$\begin{cases} \mathbf{u}^{k+1} = \text{prox}_{tg} \left(\mathbf{u}^k - t \sum_{i=1}^I \beta_i \mathbf{L}_i^* \left(\mathbf{L}_i \mathbf{u}^k - \mathbf{b}_i^k + \frac{\mathbf{v}_i^k}{\beta_i} \right) \right); \\ \mathbf{b}_i^{k+1} = \text{prox}_{f_i/\beta_i} \left(\mathbf{L}_i \mathbf{u}^{k+1} + \frac{\mathbf{v}_i^k}{\beta_i} \right), \quad i = 1, \dots, I; \\ \mathbf{v}_i^{k+1} = \mathbf{v}_i^k + \beta_i (\mathbf{L}_i \mathbf{u}^{k+1} - \mathbf{b}_i^{k+1}), \quad i = 1, \dots, I. \end{cases} \quad (14)$$

We emphasize that the key to making the $\mathbf{u}$ step proximal and avoiding linear inverse operator is imposing variable splitting to all the linear terms but not part of them. The prox_{tg} will be removed in the case that $g = 0$. Algorithm 1 summarizes the PLADMM algorithm based on (14). It is obvious that the updates of the auxiliary variables are independent, as well as the updates of the dual variables. Therefore, Algorithm 1 has a “parallel” structure. The parameter selection rule ensures that the convergence of the algorithm and its derivation will be shown in the convergence analysis.

Algorithm 1 Parallel Linearized ADMM (PLADMM1)

```

1: Set  $\mathbf{u}^0 = \mathbf{0}$ ,  $\mathbf{v}_i^0 = \mathbf{0}$ ,  $k = 0$ ,  $\beta_i > 0$ ,  $0 < t < \frac{1}{\sum_{i=1}^I \beta_i \|\mathbf{L}_i \mathbf{L}_i\|}$ ;
2: while the stopping criterion is not satisfied, do
3:   execute the iterative scheme (14);
4:    $k = k + 1$ ;
5: end while, set  $\mathbf{u}^* = \mathbf{u}^{k+1}$ , and return  $\mathbf{u}^*$ .

```

If we update $\mathbf{u}$ behind $\mathbf{v}_i$ and utilize Moreau decomposition (5), the auxiliary variables will be removed and (14) can be transformed into

$$\begin{cases} \mathbf{v}_i^{k+1} = \text{prox}_{\beta_i f_i^*}(\beta_i \mathbf{L}_i \mathbf{u}^k + \mathbf{v}_i^k), i = 1, \dots, I; \\ \mathbf{u}^{k+1} = \text{prox}_{tg}(\mathbf{u}^k - t \sum_{h=1}^H \mathbf{L}_i^* (2\mathbf{v}_i^{k+1} - \mathbf{v}_i^k)). \end{cases} \quad (15)$$

With (15), we achieve a compacter algorithm presented by Algorithm 2.

Algorithm 2 Primal-dual form of PLADMM (PLADMM2)

```

1: Set  $\mathbf{u}^0 = \mathbf{0}$ ,  $\mathbf{v}_i^0 = \mathbf{0}$ ,  $k = 0$ ,  $\beta_i > 0$ ,  $0 < t < \frac{1}{\sum_{i=1}^I \beta_i \|\mathbf{L}_i \mathbf{L}_i\|}$ ;
2: while the stopping criterion is not satisfied, do
3:   execute the iterative scheme (15);
4:    $k = k + 1$ ;
5: end while, set  $\mathbf{u}^* = \mathbf{u}^{k+1}$ , and return  $\mathbf{u}^*$ .

```

Compared with Algorithm 1, only the primal and dual variables should be updated in Algorithm 2. This means that a lower one-step complexity is achieved and less high-dimensional variables should be loaded in distributed computation [25]. Therefore, Algorithm 2 is preferable to Algorithm 1, though they are equivalent to each other. Figure 1 demonstrates the flow chat of the proposed Algorithms 1 and 2.

2.3 Convergence behavior analysis

This subsection shows that the variable sequence produced by the PLADMM algorithm departing from any initial location converges to a saddle point of (1), and a worst-case $O(1/k)$ convergence rate is obtained. The convergence proof is inspired by [12], and different forms of convergence analysis can be found in [26, 27]. On account of the equivalence of the two proposed algorithms, we only present the convergence analysis of Algorithm 1. We first present a lemma which is important for the convergence proof.

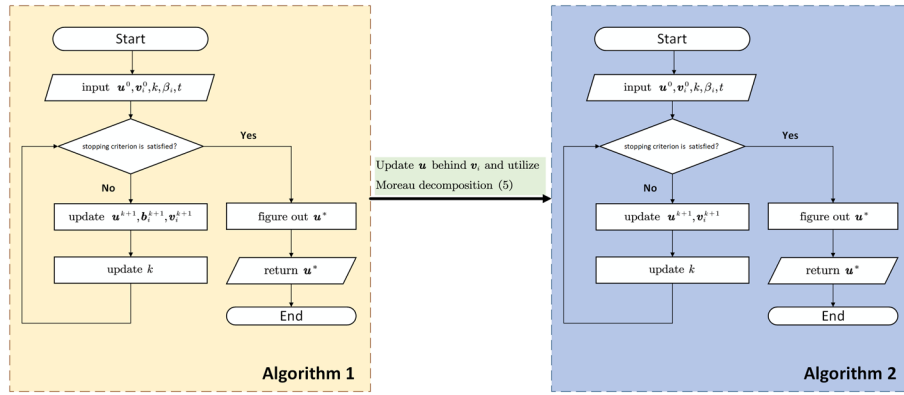


Fig. 1 Flowchart of the proposed method

Lemma 2 [28] Denote $f = f_1 + f_2$, where $f_1, f_2 \in \Gamma_0\{U\}$. If f_1 is differentiable with differential f_1' , then $\langle f_1'(u^*), u - u^* \rangle + f_2(u) - f_2(u^*) \geq 0 (\forall u \in U)$ is the necessary and sufficient condition of that $u^* \in U$ is a solution of $\text{Inf}_u(f(u))$.

With the notations of $\mathbf{b}, \mathbf{v}, \mathbf{L}$, and f , we can rewrite (14) as

$$\begin{cases} \mathbf{u}^{k+1} = \text{prox}_{tg}(\mathbf{u}^k - t\mathbf{L}^*(\mathbf{L}\mathbf{u}^k - \mathbf{b}^k + \mathbf{v}^k)); \\ \mathbf{b}^{k+1} = \text{prox}_f(\mathbf{L}\mathbf{u}^{k+1} + \mathbf{v}^k); \\ \mathbf{v}^{k+1} = \mathbf{v}^k + \mathbf{L}\mathbf{u}^{k+1} - \mathbf{b}^{k+1}. \end{cases} \quad (16)$$

We define $\mathbf{P} \triangleq \frac{1}{t}\mathbf{I} - \mathbf{L}^*\mathbf{L}$ and set

$$\mathbf{G} = \begin{pmatrix} \mathbf{P} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_b & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_b \end{pmatrix}, \mathbf{T} = \begin{pmatrix} \mathbf{I}_u & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_b & \mathbf{0} \\ \mathbf{0} & -\mathbf{I}_b & \mathbf{I}_b \end{pmatrix}, \quad (17)$$

$$\mathbf{Q} = \mathbf{GT} = \begin{pmatrix} \mathbf{P} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_b & \mathbf{0} \\ \mathbf{0} & -\mathbf{I}_b & \mathbf{I}_b \end{pmatrix}.$$

With the assumption $0 < t < \frac{1}{\sum_{i=1}^I \beta_i \|\mathbf{L}_i^* \mathbf{L}_i\|} \leq \frac{1}{\|\mathbf{L}^* \mathbf{L}\|}$, we straightforwardly find that $\mathbf{P}, \mathbf{G}$, and $\mathbf{Q}$ are positive definite. Besides, $(\mathbf{Q}^* + \mathbf{Q}) - \mathbf{T}^* \mathbf{GT}$ is positive semidefinite. Denote $\langle \mathbf{z}, \mathbf{Qz} \rangle = \|\mathbf{z}\|_{\mathbf{Q}}^2$ if $\mathbf{Q}$ is semipositive. Then, the subproblem of $\mathbf{u}$ in (16) can be transformed into

$$\mathbf{u}^{k+1} = \arg \min_{\mathbf{u}} g(\mathbf{u}) + \frac{1}{2} \|\mathbf{L}\mathbf{u} - \mathbf{b}^k + \mathbf{v}^k\|_2^2 + \frac{1}{2} \|\mathbf{u} - \mathbf{u}^k\|_{\mathbf{P}}^2 \quad (18)$$

According to Lemma 2, AL (9) is corresponding to the following variational inequality problem: identifying $(\mathbf{u}^*, \mathbf{b}^*, \mathbf{v}^*) \in U \times V \times V$, such that

$$\begin{cases} g(\mathbf{u}) - g(\mathbf{u}^*) + \langle \mathbf{u} - \mathbf{u}^*, \mathbf{L}^* \mathbf{v}^* \rangle \geq 0; \\ f(\mathbf{b}) - f(\mathbf{b}^*) + \langle \mathbf{b} - \mathbf{b}^*, -\mathbf{v}^* \rangle \geq 0; \\ \langle \mathbf{v} - \mathbf{v}^*, -\mathbf{L}\mathbf{u}^* + \mathbf{b}^* \rangle \geq 0. \end{cases} \quad (19)$$

With the notation $\mathbf{y} = (\mathbf{u}, \mathbf{b}, \mathbf{v}) \in Y \triangleq U \times V \times V$ and the linear operator $F(\mathbf{y}) = (L^* \mathbf{v}, -\mathbf{v}, -L\mathbf{u} + \mathbf{b}) \in Y$, (19) is equivalent to

$$\text{VI}(Y, F, f) : e(\mathbf{u}) + f(\mathbf{b}) - e(\mathbf{u}^*) - f(\mathbf{b}^*) + \langle \mathbf{y} - \mathbf{y}^*, F(\mathbf{y}^*) \rangle \geq 0 \quad \forall \mathbf{y} \in Y. \quad (20)$$

Let Y^* be the solution set of $\text{VI}(Y, F, f)$, i.e., $Y^* = (\mathbf{u}^*, \mathbf{b}^*, \mathbf{v}^*)$. It is easy to find that $\langle \mathbf{y} - \mathbf{y}', F(\mathbf{y}) - F(\mathbf{y}') \rangle \geq 0, \forall \mathbf{y}, \mathbf{y}' \in Y$, i.e., $F(\mathbf{y})$ is monotone.

The contractive property of the sequence generated by Algorithm 1 is summarized in Lemma 3. The corresponding convergence analysis is presented in the Appendix A.

Lemma 3 Let $\{\mathbf{u}^k, \mathbf{b}_1^k, \dots, \mathbf{b}_I^k; \mathbf{v}_1^k, \dots, \mathbf{v}_I^k\}$ be the sequence produced by the Algorithm 1. Then, $\{\mathbf{y}^k\} = \{\mathbf{u}^k, \mathbf{b}^k, \mathbf{v}^k\} = \{\mathbf{u}^k, \sqrt{\beta_1} \mathbf{b}_1^k, \dots, \sqrt{\beta_I} \mathbf{b}_I^k, \mathbf{v}_1^k / \sqrt{\beta_1}, \dots, \mathbf{v}_I^k / \sqrt{\beta_I}\}$ satisfies.

$$\|\mathbf{y}^{k+1} - \mathbf{y}^*\|_G^2 \leq \|\mathbf{y}^k - \mathbf{y}^*\|_G^2 - \|\mathbf{y}^{k+1} - \mathbf{y}^k\|_G^2 \quad \forall \mathbf{y}^* \in Y^*. \quad (21)$$

From Lemma 3, we find that the bound and nonnegative sequence $\{\|\mathbf{y}^k - \mathbf{y}^*\|_G^2\}$ is nonincreasing. Then, it must get a limit. If $k \rightarrow +\infty$, then $\|\mathbf{y}^{k+1} - \mathbf{y}^k\|_G^2 \rightarrow 0$. Referring to (37), while $k \rightarrow +\infty$, $\{\mathbf{y}^{k+1}\}$ converges to a solution of $\text{VI}(Y, F, f)$ and $\{\mathbf{u}^{k+1}, \mathbf{b}^{k+1}, \mathbf{v}^{k+1}\}$ converges to a saddle point of (9). Thus, we gain the following theorem of Algorithm 1 according to the equivalence of (6) and (9).

Theorem 1 Supposing that $\{\mathbf{u}^k, \mathbf{b}_1^k, \dots, \mathbf{b}_I^k; \mathbf{v}_1^k, \dots, \mathbf{v}_I^k\}$ is produced by Algorithm 1, then it converges to a saddle point of (6) and $\{\mathbf{u}^k\}$ converges to a solution of (1).

Lemma 4 shows the monotonicity of sequence $\|\mathbf{y}^{k+1} - \mathbf{y}^k\|_G^2$, and Theorem 2 gives the convergence rate of the proposed algorithm. Their convergence analysis is presented in the Appendix B and Appendix C, respectively

Lemma 4 Let $\{\mathbf{u}^k, \mathbf{b}_1^k, \dots, \mathbf{b}_I^k; \mathbf{v}_1^k, \dots, \mathbf{v}_I^k\}$ be the sequence produced by the Algorithm 1. Then, $\{\mathbf{y}^k\}$ satisfies

$$\|\mathbf{y}^{k+1} - \mathbf{y}^k\|_G^2 \leq \|\mathbf{y}^k - \mathbf{y}^{k-1}\|_G^2 \quad \forall k \geq 1. \quad (22)$$

Theorem 2 Let $\{\mathbf{u}^k, \mathbf{b}_1^k, \dots, \mathbf{b}_I^k; \mathbf{v}_1^k, \dots, \mathbf{v}_I^k\}$ be the sequence produced by the Algorithm 1. Then, $\{\mathbf{y}^k\} = \{\mathbf{u}^k, \mathbf{b}^k, \mathbf{v}^k\} = \{\mathbf{u}^k, \sqrt{\beta_1} \mathbf{b}_1^k, \dots, \sqrt{\beta_I} \mathbf{b}_I^k, \mathbf{v}_1^k / \sqrt{\beta_1}, \dots, \mathbf{v}_I^k / \sqrt{\beta_I}\}$ satisfies.

$$\|\mathbf{y}^{k+1} - \mathbf{y}^k\|_G^2 \leq \frac{\|\mathbf{y}^0 - \mathbf{y}^*\|_G^2}{k+1} \quad \forall \mathbf{y}^* \in Y^*, \quad (23)$$

i.e., a worst-case $O(1/k)$ convergence rate is gained.

3 Application to multichannel imaging inverse problem

In this section, we apply the proposed PLADMM to the TGV-regularized multichannel imaging inverse problem. We adopt the total generalized variation (TGV) as the regularizer of the ill-conditioned inverse problem [29, 30]. As a generalization of the total variation (TV) model, TGV employs image's high order derivatives to suppress the stair-casing effect introduced by TV while preserving edges well. We utilize the TGV model of second order for simplicity in this paper, and then get the objective function to restore an image as follows:

$$\begin{aligned} (\mathbf{x}^*, \mathbf{p}^*) &= \arg \min_{\mathbf{x}, \mathbf{p}} \alpha_1 \|\nabla \mathbf{x} - \mathbf{p}\|_1 + \alpha_2 \|\mathcal{E} \mathbf{p}\|_1 \\ &\quad \text{s.t. } \mathbf{x} \in \Omega \cap \Psi \\ \Omega &\triangleq \{\mathbf{x} : \mathbf{0} \leq \mathbf{x} \leq 255\}, \Psi \triangleq \{\mathbf{x} : \|\mathbf{K}\mathbf{x} - \mathbf{f}\|_2^2 \leq c\}. \end{aligned} \quad (24)$$

The objective function (24) is general enough to cover many imaging inverse problems, e.g., deblurring, inpainting, and compressive sensing reconstruction. In (24), the coupled l_1 terms compose the TGV regularizer TGV_α^2 . In the case that $\alpha_2 = 0$ and $\mathbf{p} = \mathbf{0}$, the TGV_α^2 degrades to the classical TV. α_1 and α_2 are positive weights balancing the weight of the first and second derivatives. The observed image and estimated image are denoted as $\mathbf{f}$ and $\mathbf{x}$, both possessing an $m \times n \times o$ (o -channel) dimension. The ill-conditioned matrix $\mathbf{K}$ models the degradation process of image. Ω is the box constraints restricting the image pixel value to $[0, 255]$, whereas Ψ is the data-fidelity constraint restricting the loss between original and degraded images. In the TGV regularizer, $\|\nabla \mathbf{x} - \mathbf{p}\|_1$ penalizes discontinuity of the image and $\|\mathcal{E} \mathbf{p}\|_1$ penalizes moderate slopes of the image. Let $\mathbf{p} \in \mathbb{R}^{mno} \times \mathbb{R}^{mno}$ ($\mathbf{p}_{i,j,l} = (p_{i,j,l,1}, p_{i,j,l,2})$) and $\mathcal{E}$ be the symmetrized difference operator. Then, $(\mathcal{E} \mathbf{p})_{i,j,l}$ is defined as

$$(\mathcal{E} \mathbf{p})_{i,j,l} = \begin{bmatrix} (\mathcal{E} \mathbf{p})_{i,j,l,1} & (\mathcal{E} \mathbf{p})_{i,j,l,3} \\ (\mathcal{E} \mathbf{p})_{i,j,l,3} & (\mathcal{E} \mathbf{p})_{i,j,l,2} \end{bmatrix} = \begin{bmatrix} \nabla_1 p_{i,j,l,1} & \frac{(\nabla_2 p_{i,j,l,1} + \nabla_1 p_{i,j,l,2})}{2} \\ \frac{(\nabla_2 p_{i,j,l,1} + \nabla_1 p_{i,j,l,2})}{2} & \nabla_2 p_{i,j,l,2} \end{bmatrix}, \quad (25)$$

where ∇_1 is the horizontal-direction difference operator and ∇_2 is the vertical-direction difference operator. The $\|\cdot\|_1$ s of $\mathbf{p}$ and $\mathcal{E} \mathbf{p}$ are defined as $\|\mathbf{p}\|_1 = \sum_{i,j=1}^{m,n} \sqrt{\sum_{l=1}^o (p_{i,j,l,1}^2 + p_{i,j,l,2}^2)}$ and $\|\mathcal{E} \mathbf{p}\|_1 = \sum_{i,j=1}^{m,n} \sqrt{\sum_{l=1}^o ((\mathcal{E} \mathbf{p})_{i,j,l,1}^2 + (\mathcal{E} \mathbf{p})_{i,j,l,2}^2 + 2(\mathcal{E} \mathbf{p})_{i,j,l,3}^2)}$. The multichannel total

variation is defined as $\|\nabla \mathbf{x}\|_1 = \sum_{i,j=1}^{m,n} \sqrt{\sum_{l=1}^o |\nabla x_{i,j,l}|^2} = \sum_{i,j=1}^{m,n} \sqrt{\sum_{l=1}^o (\nabla x_{i,j,l,1}^2 + \nabla x_{i,j,l,2}^2)}$, where subscripts 1 and 2 represent the horizontal and vertical directions, respectively.

We assign $\mathbf{u} = (\mathbf{x}, \mathbf{p})$, $g(\mathbf{u}) = \iota_\Omega(\mathbf{x})$, $f_1(\mathbf{L}_1 \mathbf{u}) = \alpha_1 \|\nabla \mathbf{x} - \mathbf{p}\|_1$, $f_2(\mathbf{L}_2 \mathbf{u}) = \alpha_2 \|\mathcal{E} \mathbf{p}\|_1$, and $f_3(\mathbf{L}_3 \mathbf{u}) = \iota_\Psi(\mathbf{x})$. Define $\tilde{\mathbf{v}}_i^{k+1} \triangleq 2\mathbf{v}_i^{k+1} - \mathbf{v}_i^k$. Referring to Algorithm 2, we gain the iterative scheme to solve (24) as follows:

$$\begin{cases} \mathbf{v}_{1,i,j,l}^{k+1} = P_{B_{\alpha_1}} \left(\beta_1 \left((\nabla \mathbf{x}^k)_{i,j,l} - \mathbf{p}_{i,j,l}^k \right) + \mathbf{v}_{1,i,j,l}^k \right); \\ \mathbf{v}_{2,i,j,l}^{k+1} = P_{B_{\alpha_2}} \left(\beta_2 \left(\mathcal{E} \mathbf{p}^k \right)_{i,j,l} + \mathbf{v}_{2,i,j,l}^k \right); \\ \mathbf{v}_3^{k+1} = \beta_3 S_{\sqrt{c}} \left(\frac{\mathbf{v}_3^k}{\beta_3} + \mathbf{K} \mathbf{x}^k - \mathbf{f} \right); \\ \mathbf{x}^{k+1} = P_\Omega \left(\mathbf{x}^k - t \left(\nabla^T \tilde{\mathbf{v}}_1^{k+1} + \mathbf{K}^T \tilde{\mathbf{v}}_3^{k+1} \right) \right); \\ \mathbf{p}_1^{k+1} = \mathbf{p}_1^k - t \left(\nabla_1^T \tilde{\mathbf{v}}_{2,1}^{k+1} + \nabla_2^T \tilde{\mathbf{v}}_{2,3}^{k+1} - \tilde{\mathbf{v}}_{1,1}^{k+1} \right); \\ \mathbf{p}_2^{k+1} = \mathbf{p}_2^k - t \left(\nabla_1^T \tilde{\mathbf{v}}_{2,3}^{k+1} + \nabla_2^T \tilde{\mathbf{v}}_{2,2}^{k+1} - \tilde{\mathbf{v}}_{1,2}^{k+1} \right). \end{cases} \quad (26)$$

In (26), $P_{B_{\alpha_1}}$ and $P_{B_{\alpha_2}}$ are two dimensional and four dimensional projection operators, respectively, where the projection operator P_{B_r} is defined as $P_{B_r}(\mathbf{q}_{i,j,l}) = \min \left(\|\mathbf{q}_{i,j,l}\|_2, r \right) \frac{\mathbf{q}_{i,j,l}}{\|\mathbf{q}_{i,j,l}\|_2}$. $S_{\sqrt{c}}$ is an mno dimensional shrinkage operator, where the shrinkage operator S_r is defined as $S_r(\mathbf{z}) = \max(\|\mathbf{z}\|_2 - r, 0) \frac{\mathbf{z}}{\|\mathbf{z}\|_2}$. Variables $\mathbf{p}_1$ and $\mathbf{p}_2$ are the combinations of $p_{i,j,l,1}$ and $p_{i,j,l,2}$, $1 \leq i \leq m$, $1 \leq j \leq n$, $1 \leq l \leq o$, respectively. Similarly, $\mathbf{v}_{1,1}$ and $\mathbf{v}_{1,2}$ are defined. $\mathbf{v}_2$ shares the same structure with $\mathcal{E} \mathbf{p}$, and $\mathbf{v}_{2,1}$, $\mathbf{v}_{2,2}$ and $\mathbf{v}_{2,3}$ are the combinations of all $v_{2,i,j,l,1}$, $v_{2,i,j,l,2}$ and $v_{2,i,j,l,3}$, respectively.

It is notable that iterative scheme (26) is more attractive in multichannel image processing, such as image deconvolution, since no linear inverse operator is involved. Generally speaking, there exists a closed form solution for edge-preserving variational multichannel image restoration problems by referring to Cramer's rule in theory. However, a complex matrix inverse operation with low calculation efficiency is needed due to the blur matrix with cross blur, which is not easy to be diagonalized [22, 31]. In contrast, the updates of $\mathbf{v}_1$, $\mathbf{v}_2$ and $\mathbf{v}_3$ in (26) are independent as well as the updates of $\mathbf{u}$, $\mathbf{p}_1$ and $\mathbf{p}_2$. Besides, the calculation of $\mathbf{v}_1$ and $\mathbf{v}_2$ are pixelwise. The classical ADMM scheme can also be used to solve problem (24). However, in this case, the solution of $\mathbf{u}$, $\mathbf{p}_1$ and $\mathbf{p}_2$ will be coupled together due to the special form of TGV and we should refer to the Cramer's rule to solve the more complex coupled linear function. Besides, this will destroy the parallel solution of $\mathbf{u}$, $\mathbf{p}_1$ and $\mathbf{p}_2$. In contrast, the proposed PLADMM guarantees the parallel solution form of $\mathbf{u}$, $\mathbf{p}_1$ and $\mathbf{p}_2$ by avoiding the complex linear inverse problem.

Algorithm 3 PLADMM with total generalized variation regularization (PLADMM-TGV)

```

1: Set  $\mathbf{x}^0 = \mathbf{0}$ ,  $\mathbf{v}_i^0 = \mathbf{0}$ ,  $k = 0$ ,  $\beta_i > 0$ ,  $0 < t \leq (1/\sum_{h=1}^3 \beta_h \|\mathbf{L}_h^* \mathbf{L}_h\|)$ ;
2: while the stopping criterion is not satisfied, do
3:   for  $i = 1, \dots, m$ ;  $j = 1, \dots, n$ 
4:      $\mathbf{v}_{1,i,j,l}^{k+1} = P_{\beta_{\alpha_1}} \left( \beta_1 \left( (\nabla \mathbf{x}^k)_{i,j,l} - \mathbf{p}_{i,j,l}^k \right) + \mathbf{v}_{1,i,j,l}^k \right)$ 
5:      $\mathbf{v}_{2,i,j,l}^{k+1} = P_{\beta_{\alpha_2}} \left( \beta_2 (\mathcal{E} \mathbf{p}^k)_{i,j,l} + \mathbf{v}_{2,i,j,l}^k \right)$ 
6:   end for
7:    $\mathbf{v}_3^{k+1} = \beta_3 S_{\sqrt{c}} \left( \frac{\mathbf{v}_3^k}{\beta_3} + \mathbf{K} \mathbf{x}^k - \mathbf{f} \right)$ ;
8:    $\mathbf{x}^{k+1} = P_{\Omega} \left( \mathbf{x}^k - t \left( \nabla^T \tilde{\mathbf{v}}_1^{k+1} + \mathbf{K}^T \tilde{\mathbf{v}}_3^{k+1} \right) \right)$ 
9:    $\mathbf{p}_1^{k+1} = \mathbf{p}_1^k - t \left( \nabla_1^T \tilde{\mathbf{v}}_{2,1}^{k+1} + \nabla_2^T \tilde{\mathbf{v}}_{2,3}^{k+1} - \tilde{\mathbf{v}}_{1,1}^{k+1} \right)$ 
10:   $\mathbf{p}_2^{k+1} = \mathbf{p}_2^k - t \left( \nabla_1^T \tilde{\mathbf{v}}_{2,3}^{k+1} + \nabla_2^T \tilde{\mathbf{v}}_{2,2}^{k+1} - \tilde{\mathbf{v}}_{1,2}^{k+1} \right)$ 
11:   $k = k + 1$ ;
12: end while, set  $\mathbf{x}^* = \mathbf{x}^{k+1}$ , and return  $\mathbf{x}^*$ .

```

Following the operator assignment mentioned above, it is simple to verify that $\sum_{i=1}^3 \beta_i \|\mathbf{L}_i^* \mathbf{L}_i\| = \beta_1 \|\nabla^T \nabla\| + \beta_2 \|\mathcal{E}^T \mathcal{E}\| + \beta_3 \|\mathbf{K}^T \mathbf{K}\| \leq 8\beta_1 + 8\beta_2 + \beta_3$. We can prove that $\|\mathbf{K}^T \mathbf{K}\| = 1$ with the basic condition $\mathbf{K} \mathbf{1} = \mathbf{1}$ [32]. We refer to [29] for the norm analysis of ∇ and $\mathcal{E}$. With the above analysis, $t = \frac{1}{8\beta_1 + 8\beta_2 + \beta_3}$ is chosen for the TGV-based imaging inverse problem. Besides, $(\alpha_1, \alpha_2) = (1, 6)$ are set for the TGV model.

In (24), we use $c = \tau m n \sigma^2$ to estimate the noise-dependent c , where σ^2 denotes the noise variance and $\tau = -0.0009 \text{BSNR} + 0.99$. τ is calculated with linear regression computing method [14]. We define the blurred signal-to-noise ratio (BSNR) as equation $\text{BSNR} = 10 \log_{10} (\text{var}(\mathbf{f})/\sigma^2)$ (dB), and $\text{var}(\mathbf{f})$ is the variance of $\mathbf{f}$. We empirically set $\beta_1 = \beta_2 = 0.04$ and $\beta_3 = \text{BSNR} \times \beta_1$ to accelerate the convergence. For different problems, such as deblurring or denoising, we should find proper parameters by trial-and-error method. Although we should adjust the parameters for different types of problems, abundant experiments on natural images indicate that, under a given type of problem, the parameter setting is robust to a certain extent towards the variation of the image type and the image size [14].

4 Experiment results

The following three experiments were performed with MATLAB and the hardware platform is a PC with Intel Core i5 CPU and 8 GB RAM. The quality of the degraded and restored/reconstructed images are assessed with two indexes: PSNR in dB (peak signal-to-noise ratio) and SSIM (the structural similarity index). PSNR and SSIM are both important image quality evaluation indicators. PSNR places more emphasis on pixel accuracy but does not pay attention to the structural information of images. SSIM

considers the structural information of images, which is beneficial for more accurate evaluation of image quality. Major initialization for parameter settings of the proposed method obeys the criteria described in the former section.

4.1 Multichannel image deblurring

In this subsection, the proposed TGV-regularized PLADMM (PLADMM-TGV) is compared with some famous methods in terms of RGB image deconvolution, including ADMM-based and primal-dual [33–35] methods. The competitors are the parallel ADMM with TGV regularizer (PADMM-TGV) [21], the fast total variation deconvolution algorithm (FTVD-v4) [22], and the parallel primal-dual splitting method with TV regularizer (PPDS-TV) [33]. PADMM-TGV solves problem (24) without the box constraints and FTVD-v4 solves the unconstrained TV-based deblurring problem. In contrast with PLADMM and PPDS, both the other two competitors need to solve the inverse operation. Unlike PLADMM, PADMM and PPDS, FTVD-v4 needs to manually select the regularization parameter which balances the regularizer and data-fidelity by trial-and-error. For RGB images, multichannel means that the channel dimension of the image is 3.

It is noticed that all the four methods possess a worst-case $O(1/k)$ convergence rate. To ensure these methods sufficiently converge, the stopping criterion of the compared methods in this experiment is the iteration reaching 300. In the following, $Av(z)$ denotes average blur kernel with a blurring size z ; $Ga(z, \delta)$ denotes Gaussian blur kernel with a blurring size z and standard deviation δ ; $Mo(z, \theta)$ denotes motion blur kernel with a length z and an angle θ . For assessing the performance of the proposed method and competitors under different image sizes, blur types, and noise levels, four sub-experiments detailed in Table 1 are designed. The four test images are shown in Fig. 2: they are image Lena of size 256×256 , image Peppers of size 512×512 , image Monarch of size 768×512 , and image Race of size 768×512 . The four sets of image blur for each image are designed as follows: 1. Generate 9 kernels first: $\{Av(13), Av(15), Av(17), Ga(11; 9), Ga(21; 11), Ga(31; 13), Mo(21; 45), Mo(41; 90), Mo(61; 135)\}$. 2. Assign these nine kernels to $\{Kc_{11}, Kc_{12}, Kc_{13}; Kc_{21}, Kc_{22}, Kc_{23}; Kc_{31}, Kc_{32}, Kc_{33}\}$, where Kc_{jj} , $j = 1, 2, 3$, are within-channel kernels whereas the others are cross-channel kernels. 3. The final four sets of blurs for comparison are generated by multiplying different relative weights $\{1, 0, 0; 0, 1, 0; 0, 0, 1\}$ (Set 1), $\{0.6, 0.2, 0.2; 0.15, 0.7, 0.15; 0.1, 0.1, 0.8\}$ (Set 2), $\{0.7, 0.15, 0.15; 0.1, 0.8, 0.1; 0.2, 0.2, 0.6\}$ (Set 3), and $\{0.5, 0.25, 0.25; 0.25, 0.5, 0.25; 0.25, 0.25, 0.5\}$ (Set 4) to the corresponding blur kernels. Then, the four test images are blurred by these four

Table 1 Detailed settings of the image deblurring experiment

No.	Test Image	Blur kernel	σ	PSNR	SSIM
1	Lena	Set 1	3	20.05	0.5239
2	Peppers	Set 2	8	17.55	0.5140
3	Monarch	Set 3	10	17.95	0.4608
4	Race	Set 4	14	16.99	0.2206

(a) Lena of size 256×256 (b) Peppers of size 512×512 (c) Monarch of size 768×512 (d) Race of size 768×512 **Fig. 2** Test images for multichannel image deblurring**Table 2** Comparison results for image deblurring [1]

Problem	Method	PSNR	SSIM	Time (s)
1	PLADMM-TGV	26.21	0.7680	28.70
	PADMM-TGV	26.21	0.7649	35.62
	PLADMM-TV	26.10	0.7608	16.67
	PPDS-TV	26.09	0.7611	17.94
	FTVD-v4	26.04	0.7583	10.37
2	PLADMM-TGV	25.57	0.7632	155.46
	PADMM-TGV	25.56	0.7623	199.93
	PLADMM-TV	25.43	0.7545	85.84
	PPDS-TV	25.44	0.7544	89.34
	FTVD-v4	25.25	0.7507	54.69
3	PLADMM-TGV	23.85	0.8083	234.45
	PADMM-TGV	23.83	0.8063	302.68
	PLADMM-TV	23.76	0.8026	131.81
	PPDS-TV	23.77	0.8028	138.44
	FTVD-v4	23.61	0.7965	84.43
4	PLADMM-TGV	19.96	0.4335	230.57
	PADMM-TGV	19.88	0.4282	298.24
	PLADMM-TV	19.87	0.4278	132.05
	PPDS-TV	19.87	0.4277	138.50
	FTVD-v4	19.85	0.4277	84.23

The bold values indicate that these values are optimal under their corresponding indicators

blurs, respectively. Different levels of Gaussian noise are imposed on the four blurred images to get the final degraded images.

The objective comparison results are shown in Table 2 [1]. It can be found that PLADMM-TGV and PADMM-TGV restores better images than PLADMM-TV, PPDS-TV and FTVD-v4, since they gain higher PSNRs and SSIMs. This is because that PLADMM-TGV and PADMM-TGV employ more complicated TGV regularizer. Although FTVD-v4 seems to consume less time than PLADMM-TGV, PADMM-TGV, PLADMM-TV and PPDS-TV for a single run, it actually costs much more time due to the trial-and-error process for finding the optimal regularization parameter. In contrast, the other methods can adaptively find the optimal regularization parameter without trial-and-error. By avoiding the matrix inverting operation with Kramer's rule for cross-channel blurring, PLADMM and PPDS converges much faster than PADMM due to higher one-step efficiency. It is interesting that the speed superiority of PLADMM-TGV over PADMM-TGV for Problem 1 is not as large as that for Problems 2, 3, and 4. The reason is that Problem 1 only involves within-channel blurs, whereas Problems 2, 3, and 4 involve cross-channel blurs. The cross-channel blurs prevent the blur matrix from being wholly diagonalized with FFT. Besides, PLADMM is of high potential in efficiency in the sense that it can be deeply accelerated by parallel calculation equipment like GPU, due to a highly parallel structure.

Figures 3, 4, 5 and 6 demonstrate the degraded and deblurred images by the three methods for Problems 1, 2, 3, and 4, respectively. It can be observed from Figs. 3, 4, 5 and 6 that PLADMM-TGV and PADMM-TGV restore better images in visual quality compared with the TV-based PLADMM-TV, PPDS-TV and FTVD-v4. The TGV-based algorithms successfully suppress the staircasing effects that are existent in the results of PLADMM-TV, PPDS-TV and FTVD-v4 (for better observation, some staircasing effects are labeled with rectangles). The robustness of PLADMM-TGV is also



Fig. 3 Degradation and restorations of the Lena image by PLADMM, PADMM, PPDS, and FTVD-v4



Fig. 4 Degradation and restorations of the Peppers image by PLADMM, PADMM, and FTVD-v4 [1]. PSNR=43.46 dB, SSIM=0.9841 PSNR=43.30 dB, SSIM=0.9829 PSNR=42.62 dB

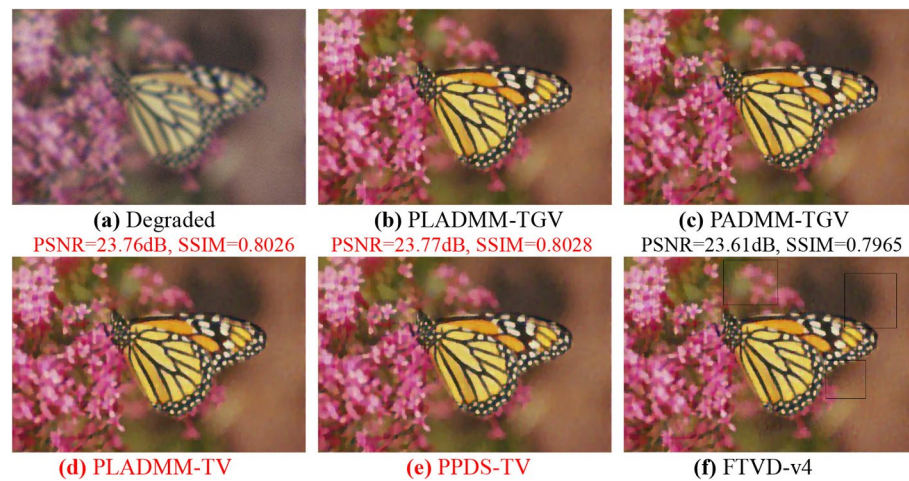


Fig. 5 Degradation and restorations of the Monarch image by PLADMM, PADMM, and FTVD-v4 [1]

demonstrated in the background of images with different sizes, noises with different standard deviations, and blurs with different types.

PSNR = 20.05 dB, SSIM = 0.5239 PSNR = 26.21 dB, SSIM = 0.7680 PSNR = 26.21 dB, SSIM = 0.7649.

Although PLADMM-TGV and PADMM-TGV achieve similar deblurring quality due to the same TGV regularizer, Fig. 7a, b, c, and d shows that PLADMM converges faster than PADMM due to the higher one-step efficiency. However, PLADMM may fall behind at the beginning since the inexact calculation of the $\mathbf{x}$ step.

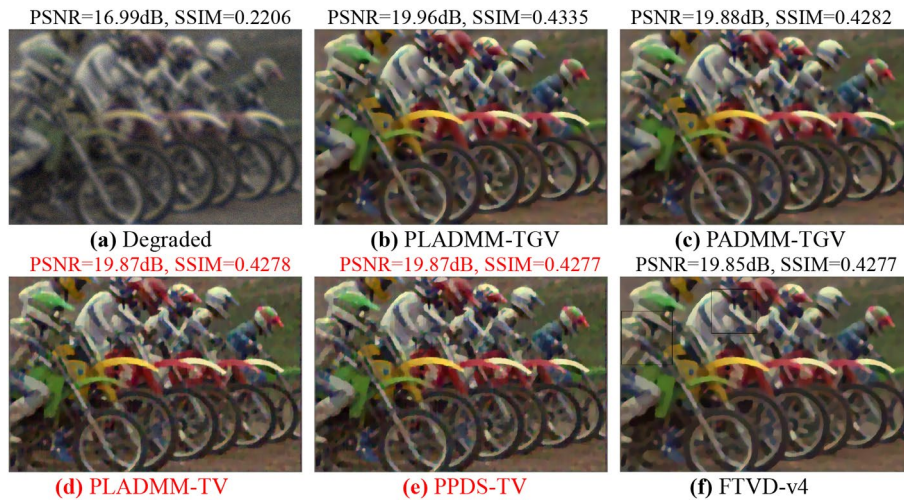
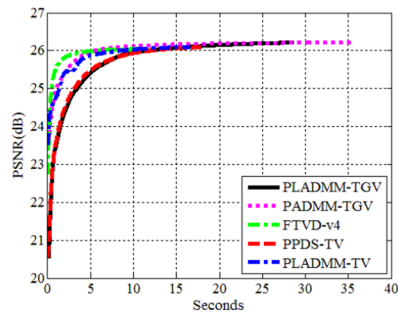
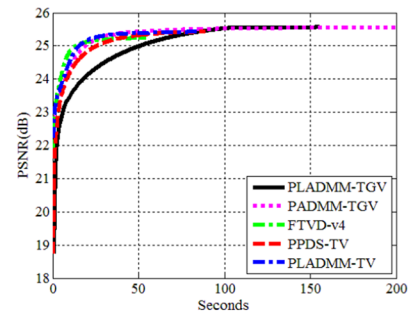


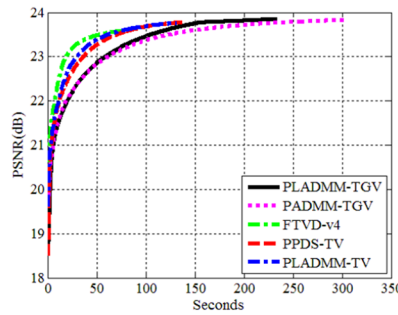
Fig. 6 Degradation and restorations of the race image by PLADMM, PADMM, PPDS, and FTVD-v4



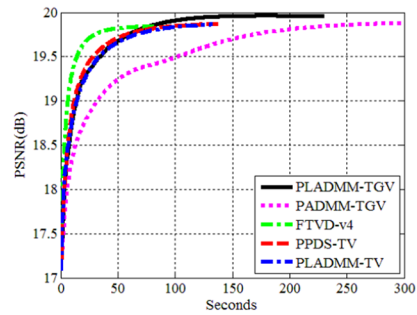
(a) PSNR versus CPU time for Lena



(b) PSNR versus CPU time for Peppers



(c) PSNR versus CPU time for Monarch



(d) PSNR versus CPU time for Race

Fig. 7 PSNR versus CPU time for the deblurring experiment of images **a** Lena, **b** Peppers, **c** Monarch, and **d** Race, respectively

4.2 RGB image inpainting

In this subsection, an RGB image inpainting experiment is designed to show the potential of the proposed PLADMM. The PLADMM-TGV, PLADMM-TV [with $\alpha_2=0$ and $\mathbf{p}=\mathbf{0}$ in (24)], and FTVD-v4 are compared. Image Girl of size 768×512 , which has abundant detailed features, is used in this experiment. We first randomly let the test image lose 0%, 5%, 15%, ..., 65% of pixels. Gaussian noise with $\sigma = 20$ (standard deviation) then corrupts the image to get the final observation. PLADMM-TGV

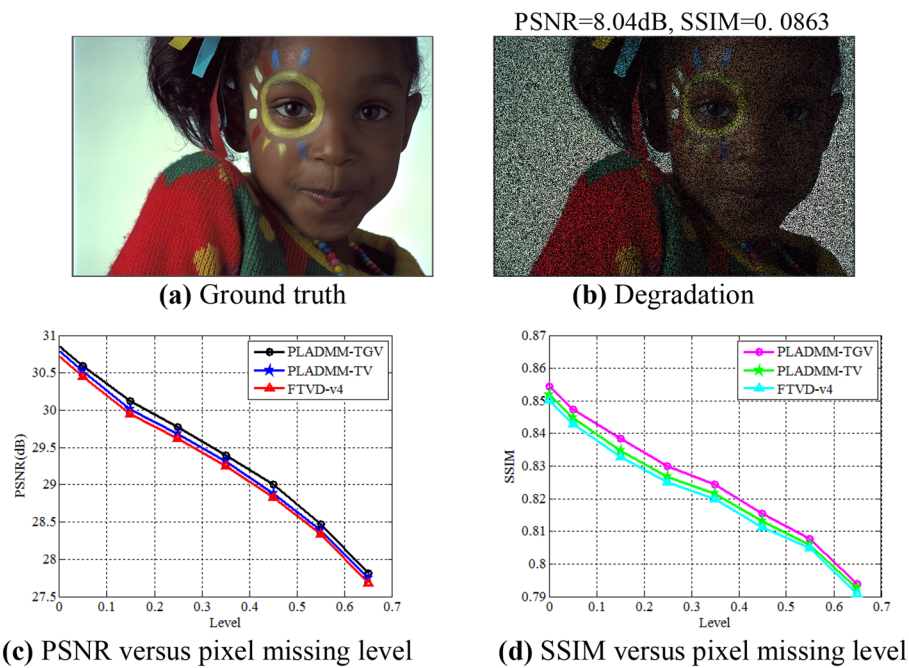


Fig. 8 Ground truth (a), degradation (b) with a 55% pixel loss and Gaussian noise with $\sigma = 20$, and PSNR (c) and SSIM (d) versus pixel loss level for the Girl image, respectively

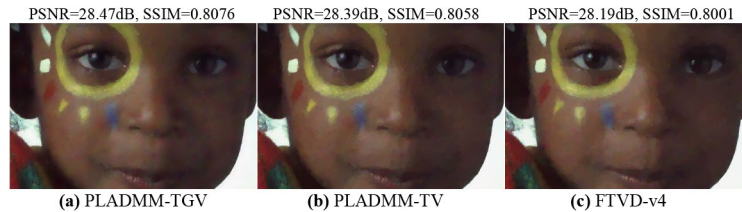


Fig. 9 Restored outputs (close-ups) of the Girl image for PLADMM-TGV in (a), PLADMM-TV in (b), and FTVD-v4 in (c)

and PLADMM-TV are used to restore the observations. If the pixel loss level is 0, then the inpainting degrades to simple denoising. The stopping criterion of the compared methods for inpainting is the iteration reaching 150.

Figure 8a, b shows the ground truth and the degraded Girl image with a 55% pixel loss which means that it loses most of its pixel values. The PSNR and SSIM versus the pixel loss level for the inpainting experiment are presented in Fig. 8c, d, respectively. The PLADMM-TGV shows superiority over the PLADMM-TV and FTVD-v4 both on PSNR and on SSIM. Figure 9 displays inpainting outputs of the three algorithms, as well as the corresponding local close-ups, where the Girl image is set to lose its 55% of pixels. The following phenomena can be found from the close-up outputs of PLADMM-TGV (Fig. 9a), PLADMM-TV (Fig. 9b), and FTVD-v4 (Fig. 9c). First, although the ground truth loses most of its pixel values, the three algorithms effectively inpaint the degraded Girl image, which indicates the efficiency of the PLADMM in image inpainting. Second, there exist staircasing effects in the inpainted image of

PLADMM-TV and FTVD-v4. However, PLADMM-TGV successfully suppresses these staircasing effects with the more complex TGV regularizer.

4.3 Parallel magnetic resonance imaging reconstruction

Parallel magnetic resonance imaging (pMRI) is a technique to accelerate sampling speed in MRI, by only acquiring part of the k-space data. To reduce scan time, pMRI extracts extra spatial information from an array of surface coils, surrounding the scanned objects by multiple receivers and collecting in the parallel part of the Fourier components at each receiver. However, this reduction in the number of recorded Fourier components leads to aliasing artifacts in images. In this experiment, we show that the proposed PLADMM-TGV can effectively suppress these artifacts and reconstruct the original image while maintaining higher speed.

Sensitivity encoding is one of the most commonly used parallel imaging method [36]. The imaging process can be modeled as

$$\mathbf{b}_j = \mathbf{P}_a \mathbf{F} \mathbf{S}_j \mathbf{x} + \mathbf{e} \quad (27)$$

where $\mathbf{x}$ is the original image, $\mathbf{b}_j$ is the partial k-space data measured by receiver j , $\mathbf{e}$ is Gaussian noise, $\mathbf{P}_a$ is a binary matrix representing the undersampling pattern (mask), and $\mathbf{S}_j$ is the sensitivity map for the j th channel. We can reconstruct the original image with the proposed method described in Sect. III with the following form:

$$(\mathbf{x}^*, \mathbf{p}^*) = \arg \min_{\mathbf{x}, \mathbf{p}} \alpha_1 \|\nabla \mathbf{x} - \mathbf{p}\|_1 + \alpha_2 \|\mathcal{E} \mathbf{p}\|_1 \text{ s.t. } \sum_{j=1}^M \|\mathbf{P}_a \mathbf{F} \mathbf{S}_j \mathbf{x} - \mathbf{b}_j\|_2^2 \leq c \quad (28)$$

In the pMRI experiment, we use an MRI image of size 512×512 and a radial mask with acquisition rate of 0.147. In the radial mask, the more spokes, means the higher the sampling rate. The ground truth image and the radial mask are shown in Fig. 10a, b. The ground truth image contains a wealth of details about the human brain, including brain circuit structures and vascular structures. The image was acquired from a 3 T Philips scanner (Phillips, Best, Netherlands) using T2-weighted turbo spin echo sequence [37]. The number of coils is 8, i.e., $M = 8$. The Radial mask utilized for different coils is the same and the sensitivity map for every coil is given in the MRI image data set. To demonstrate the feasibility of the proposed method for solving pMRI problem, we infer to the proposed PLADMM-TGV and its degenerating form PLADMM-TV to solve the problem (28). We compare them with the other four famous methods: FUSL [8], LADM [37], and ALADMM [27]. The three competitors are all TV-based and solve the unconstrained form of problem (28) [36]. In the pMRI experiment, the stopping rule is the iteration counters reaching 100.

Table 3 presents the experiment results in terms of PSNR, SSIM, Relative Error, and CPU time. The relative error is defined as $\|\mathbf{x}^* - \mathbf{x}_0\|_2 / \|\mathbf{x}_0\|_2$, where $\mathbf{x}_0$ and $\mathbf{x}^*$ are the ground truth and reconstructed images, respectively.

Table 3 shows that all the involved methods can successfully reconstruct the pMRI image; PLADMM-TGV outperforms the other methods in PSNR, SSIM, and relative error, since it adopts the more efficient TGV regularization; whereas, PLADMM-TV is superior in terms of CPU time. In contrast with the TV-based FUSL, LADM, and

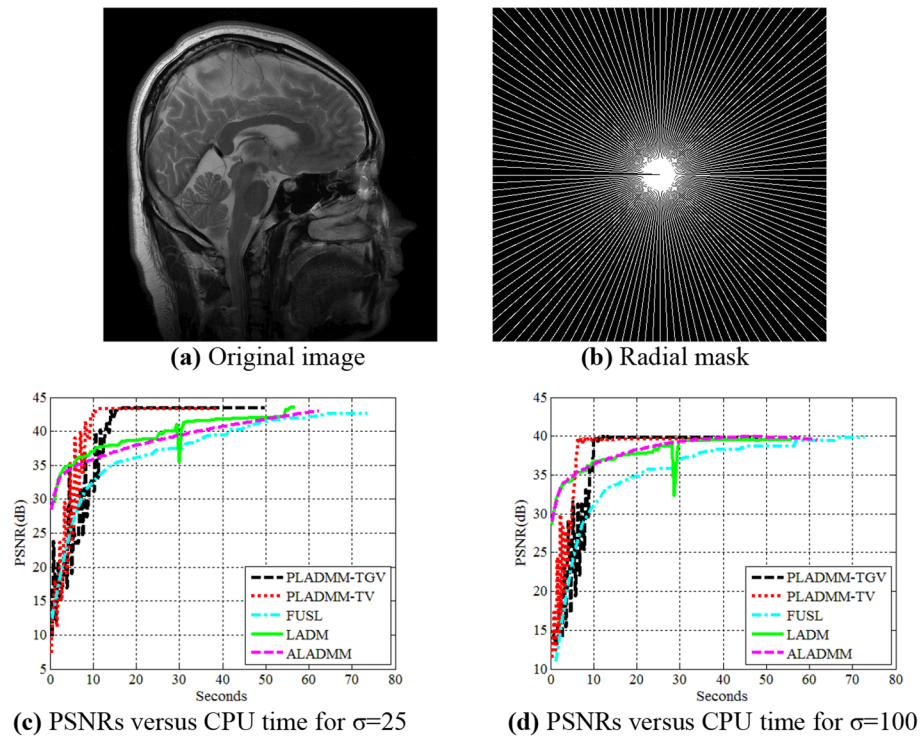


Fig. 10 Original image, radial mask, and PSNRs versus CPU time of the different methods for pMRI experiment

Table 3 Experimental results of the pMRI reconstruction experiment

Noise level	Method	PSNR	SSIM	Relative Error	Time (s)
$\sigma=25$	PLADMM-TGV	43.46	0.9841	0.0281	49.88
	PLADMM-TV	43.30	0.9829	0.0286	39.04
	FUSL	42.62	0.9749	0.0310	73.63
	LADM	43.27	0.9825	0.0287	56.81
	ALADMM	42.99	0.9831	0.0297	62.42
$\sigma=100$	PLADMM-TGV	39.87	0.9705	0.0425	48.68
	PLADMM-TV	39.59	0.9696	0.0439	37.62
	FUSL	39.80	0.9695	0.0420	72.94
	LADM	39.52	0.9690	0.0497	55.95
	ALADMM	39.56	0.9645	0.0467	69.72

ALADMM, as well as the Plug-and-Play-based PAP, although PLADMM-TGV adopts the more complex TGV regularizer, it consumes less CPU time than the other competitors. Figure 10c, d presents the PSNR versus CPU time of these methods when the noise standard deviation σ is equal to 25 and 100, respectively. PLADMM-TGV and PLADMM-TV step into convergency faster than FUSL, LADM, and ALADMM, which shows the speed superiority of the proposed PLADMM. Figures 11 and 12 further present the reconstructed images and residue images ($|\mathbf{x}^* - \mathbf{x}_0|$) of the five methods, from which we can observe that PLADMM-TGV reconstructs the most proximate image of the ground truth.

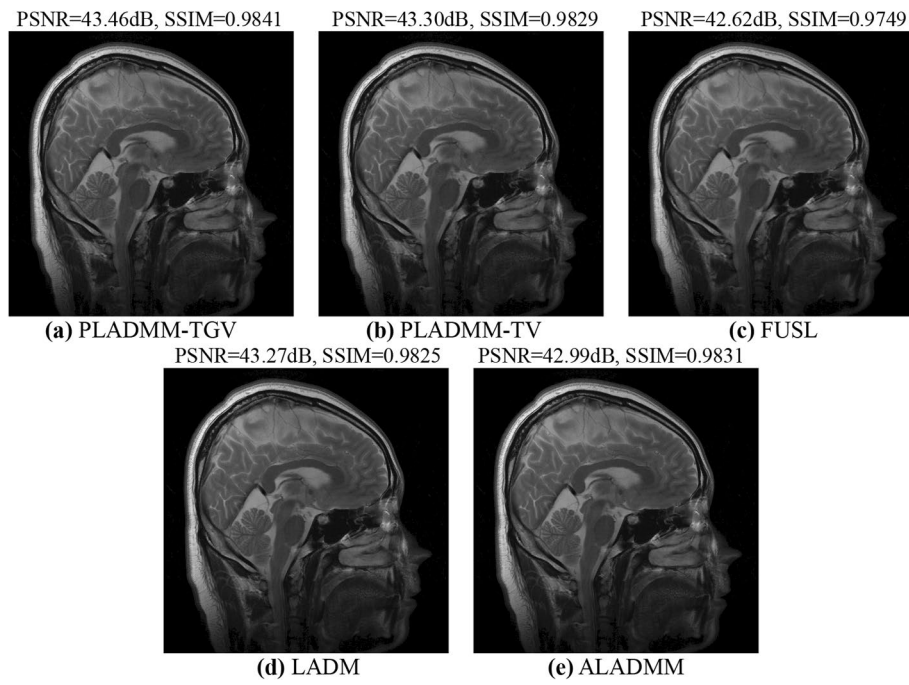


Fig. 11 Reconstructed images by PLADMM-TGV, PLADMM-TV, FUSL, LADM, and ALADMM under Gaussian noise with a standard deviation $\sigma=25$

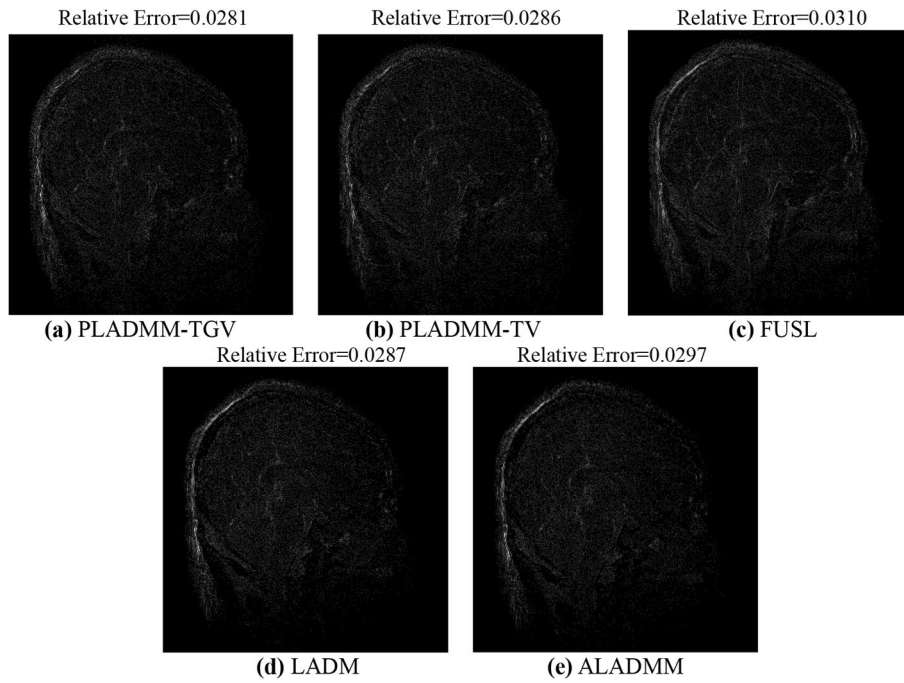


Fig. 12 Residue images between the original image and reconstructed results by PLADMM-TGV, PLADMM-TV, FUSL, LADM, and ALADMM under Gaussian noise with $\sigma=25$

5 Conclusions

A generic method named PLADMM is proposed in this paper to solve large-scale regularized imaging inverse problems, such as multichannel deblurring, inpainting, and pMRI reconstruction. By using the proximal splitting and further exploiting the function structure, the PLADMM method achieves a concise iterative frame. The proposed method simplifies the sub-problems via closed-form solution by linearization of the quadratic terms. The linear inverse operation and nested inner loop that are commonly seen in inverse problems are excluded. The parallel structure makes the PLADMM method being of high potential for a wide range of practical applications. Theoretical analysis shows that the proposed method has strict convergence property and fast single-step execution efficiency compared with PADMM, FTVD-v4, and other methods. Experimental results validate the efficiency of our method for image deblurring, inpainting, and pMRI reconstruction. The classical TGV model adopted in our work is a manual feature model which regularizes the smoothness of the restored image. However, for a specific image processing task, using learning feature is a feasible way to raise the quality of the restored image. In the future, we will put effort on the combination of PLADMM and learning features to construct more powerful algorithms.

Appendix A

Proof of Lemma 3

Proof As $\mathbf{b}^{k+1}$ is the solution of the problem with respect to $\mathbf{b}$, according to Lemma 2, we have.

$$f(\mathbf{b}) - f(\mathbf{b}^{k+1}) + \langle \mathbf{b} - \mathbf{b}^{k+1}, \mathbf{b}^{k+1} - \mathbf{L}\mathbf{u}^{k+1} - \mathbf{v}^k \rangle \geq 0 \quad \forall \mathbf{b} \in V. \quad (29)$$

Plugging into (29) the third equation of (16), we get

$$f(\mathbf{b}) - f(\mathbf{b}^{k+1}) + \langle \mathbf{b} - \mathbf{b}^{k+1}, -\mathbf{v}^{k+1} \rangle \geq 0 \quad \forall \mathbf{b} \in V. \quad (30)$$

For the same reason, we have

$$f(\mathbf{b}) - f(\mathbf{b}^k) + \langle \mathbf{b} - \mathbf{b}^k, -\mathbf{v}^k \rangle \geq 0 \quad \forall \mathbf{b} \in V. \quad (31)$$

Considering $\mathbf{b} = \mathbf{b}^k$ and $\mathbf{b} = \mathbf{b}^{k+1}$ in (30) and (31), respectively, by addition, we obtain

$$\langle \mathbf{b}^k - \mathbf{b}^{k+1}, \mathbf{v}^k - \mathbf{v}^{k+1} \rangle \geq 0. \quad (32)$$

According to the optimality condition with respect to $\mathbf{u}$ in (16), we get

$$\begin{aligned} g(\mathbf{u}) - g(\mathbf{u}^{k+1}) + \langle \mathbf{u} - \mathbf{u}^{k+1}, \mathbf{L}^*(\mathbf{L}\mathbf{u}^{k+1} - \mathbf{b}^k + \mathbf{v}^k) \rangle \\ + \langle \mathbf{u} - \mathbf{u}^{k+1}, \mathbf{P}(\mathbf{u}^{k+1} - \mathbf{u}^k) \rangle \geq 0 \quad \forall \mathbf{u} \in U. \end{aligned} \quad (33)$$

Plugging into (33) the third equation of (16), we have

$$g(\mathbf{u}) - g(\mathbf{u}^{k+1}) + \langle \mathbf{u} - \mathbf{u}^{k+1}, \mathbf{L}^*(\mathbf{v}^{k+1} + \mathbf{b}^{k+1} - \mathbf{b}^k) \rangle + \langle \mathbf{u} - \mathbf{u}^{k+1}, \mathbf{P}(\mathbf{u}^{k+1} - \mathbf{u}^k) \rangle \geq 0 \quad \forall \mathbf{u} \in U. \quad (34)$$

The third equation of (16) indicates that

$$\langle \mathbf{v} - \mathbf{v}^{k+1}, -\mathbf{L}\mathbf{u}^{k+1} + \mathbf{b}^{k+1} + \mathbf{v}^{k+1} - \mathbf{v}^k \rangle = 0 \quad \forall \mathbf{v} \in V. \quad (35)$$

Adding (30), (34), and (35), we achieve

$$g(\mathbf{u}) + f(\mathbf{b}) - g(\mathbf{u}^{k+1}) - f(\mathbf{b}^{k+1}) + \langle \mathbf{y} - \mathbf{y}^{k+1}, F(\mathbf{y}^{k+1}) \rangle + \langle \mathbf{u} - \mathbf{u}^{k+1}, \mathbf{L}^*(\mathbf{b}^{k+1} - \mathbf{b}^k) + \mathbf{P}(\mathbf{u}^{k+1} - \mathbf{u}^k) \rangle + \langle \mathbf{v} - \mathbf{v}^{k+1}, \mathbf{v}^{k+1} - \mathbf{v}^k \rangle \geq 0 \quad \forall \mathbf{y} \in Y. \quad (36)$$

Inequality (36) implies that if $\|\mathbf{y}^{k+1} - \mathbf{y}^k\|_2^2 = \|\mathbf{u}^{k+1} - \mathbf{u}^k\|_2^2 + \|\mathbf{b}^{k+1} - \mathbf{b}^k\|_2^2 + \|\mathbf{v}^{k+1} - \mathbf{v}^k\|_2^2 = 0$, then

$$g(\mathbf{u}) + f(\mathbf{b}) - g(\mathbf{u}^{k+1}) - f(\mathbf{b}^{k+1}) + \langle \mathbf{y} - \mathbf{y}^{k+1}, F(\mathbf{y}^{k+1}) \rangle \geq 0 \quad \forall \mathbf{y} \in Y. \quad (37)$$

This is, $\mathbf{y}^{k+1}$ is VI(Y, F, f)'s solution and $(\mathbf{u}^{k+1}, \mathbf{b}^{k+1}; \mathbf{v}^{k+1})$ is (9)'s saddle point. With $\mathbf{y} = \mathbf{y}^*$ in (36), we obtain

$$\begin{aligned} & \langle \mathbf{u}^* - \mathbf{u}^{k+1}, \mathbf{P}(\mathbf{u}^{k+1} - \mathbf{u}^k) \rangle + \langle \mathbf{b}^* - \mathbf{b}^{k+1}, \mathbf{b}^{k+1} - \mathbf{b}^k \rangle + \langle \mathbf{v}^* - \mathbf{v}^{k+1}, \mathbf{v}^{k+1} - \mathbf{v}^k \rangle \\ & \geq g(\mathbf{u}^{k+1}) + f(\mathbf{b}^{k+1}) - g(\mathbf{u}^*) - f(\mathbf{b}^*) + \langle \mathbf{y}^{k+1} - \mathbf{y}^*, F(\mathbf{y}^{k+1}) \rangle + \langle \mathbf{b}^* - \mathbf{b}^{k+1}, \mathbf{b}^{k+1} - \mathbf{b}^k \rangle \\ & \quad - \langle \mathbf{L}(\mathbf{u}^* - \mathbf{u}^{k+1}), \mathbf{b}^{k+1} - \mathbf{b}^k \rangle. \end{aligned} \quad (38)$$

As $\mathbf{y}^*$ is the optimal solution of VI(Y, F, f)

$$g(\mathbf{u}^{k+1}) + f(\mathbf{b}^{k+1}) - g(\mathbf{u}^*) - f(\mathbf{b}^*) + \langle \mathbf{y}^{k+1} - \mathbf{y}^*, F(\mathbf{y}^*) \rangle \geq 0. \quad (39)$$

Since F is monotone, we have

$$\langle \mathbf{y}^{k+1} - \mathbf{y}^*, F(\mathbf{y}^{k+1}) \rangle \geq \langle \mathbf{y}^{k+1} - \mathbf{y}^*, F(\mathbf{y}^*) \rangle. \quad (40)$$

According to (32)

$$\begin{aligned} & \langle \mathbf{b}^* - \mathbf{b}^{k+1}, \mathbf{b}^{k+1} - \mathbf{b}^k \rangle - \langle \mathbf{L}(\mathbf{u}^* - \mathbf{u}^{k+1}), \mathbf{b}^{k+1} - \mathbf{b}^k \rangle \\ & = \langle \mathbf{L}\mathbf{u}^{k+1} - \mathbf{b}^{k+1}, \mathbf{b}^{k+1} - \mathbf{b}^k \rangle = \langle \mathbf{v}^{k+1} - \mathbf{v}^k, \mathbf{b}^{k+1} - \mathbf{b}^k \rangle \geq 0. \end{aligned} \quad (41)$$

Combining (38) and (41), we obtain

$$\begin{aligned} & \langle \mathbf{u}^* - \mathbf{u}^{k+1}, \mathbf{P}(\mathbf{u}^{k+1} - \mathbf{u}^k) \rangle + \langle \mathbf{b}^* - \mathbf{b}^{k+1}, \mathbf{b}^{k+1} - \mathbf{b}^k \rangle \\ & + \langle \mathbf{v}^* - \mathbf{v}^{k+1}, \mathbf{v}^{k+1} - \mathbf{v}^k \rangle = \langle \mathbf{y}^* - \mathbf{y}^{k+1}, \mathbf{G}(\mathbf{y}^{k+1} - \mathbf{y}^k) \rangle \geq 0. \end{aligned} \quad (42)$$

Therefore

$$\begin{aligned}
\|y^k - y^*\|_G^2 &= \|y^* - y^{k+1}\|_G^2 + \|y^{k+1} - y^k\|_G^2 + 2\langle y^* - y^{k+1}, G(y^{k+1} - y^k) \rangle \\
&\geq \|y^* - y^{k+1}\|_G^2 + \|y^{k+1} - y^k\|_G^2,
\end{aligned} \tag{43}$$

and the proof of Lemma 3 is completed.

Appendix B

Proof of Lemma 4

Proof Denote $\bar{u}^k = u^{k+1}$, $\bar{b}^k = b^{k+1}$, $\bar{v}^k = Lu^{k+1} - b^k + v^k = v^{k+1} + b^{k+1} - b^k$, $\bar{z}^k = (\bar{b}^k, \bar{v}^k)$ and $\bar{y}^k = (\bar{u}^k, \bar{b}^k, \bar{v}^k)$. Denote $T : (b, v) \rightarrow (b, -b + v)$. Then, we have.

$$y^{k+1} = y^k - T(y^k - \bar{y}^k). \tag{44}$$

With the above notations, (36) can be rewritten as

$$\begin{aligned}
g(u) + f(b) - g(\bar{u}^k) - f(\bar{b}^k) + \langle y - \bar{y}^k, F(\bar{y}^k) \rangle \\
+ \langle y - \bar{y}^k, Q(\bar{y}^k - y^k) \rangle \geq 0 \quad \forall y \in Y.
\end{aligned} \tag{45}$$

For the same reason, we have

$$\begin{aligned}
g(u) + f(b) - g(\bar{u}^{k+1}) - f(\bar{b}^{k+1}) + \langle y - \bar{y}^{k+1}, F(\bar{y}^{k+1}) \rangle \\
+ \langle y - \bar{y}^{k+1}, Q(\bar{y}^{k+1} - y^{k+1}) \rangle \geq 0 \quad \forall y \in Y.
\end{aligned} \tag{46}$$

Set $y = \bar{y}^{k+1}$ and $y = \bar{y}^k$ in (45) and (46), respectively, and then do addition. We have

$$-\langle \bar{y}^{k+1} - \bar{y}^k, F(\bar{y}^{k+1}) - F(\bar{y}^k) \rangle + \langle \bar{y}^{k+1} - \bar{y}^k, Q((y^{k+1} - y^k) - (\bar{y}^{k+1} - \bar{y}^k)) \rangle \geq 0. \tag{47}$$

Since F is monotone, we have

$$\langle \bar{y}^{k+1} - \bar{y}^k, Q((y^{k+1} - y^k) - (\bar{y}^{k+1} - \bar{y}^k)) \rangle \geq 0. \tag{48}$$

Adding

$$\langle y^{k+1} - y^k - \bar{y}^{k+1} + \bar{y}^k, Q(y^{k+1} - y^k - \bar{y}^{k+1} + \bar{y}^k) \rangle \geq 0, \tag{49}$$

to both sides of (48) and using $\langle y, Qy \rangle = \frac{1}{2} \langle y, (Q + Q^*)y \rangle$, we obtain

$$\langle y^{k+1} - y^k, Q((y^{k+1} - y^k) - (\bar{y}^{k+1} - \bar{y}^k)) \rangle \geq \frac{1}{2} \|(y^{k+1} - y^k) - (\bar{y}^{k+1} - \bar{y}^k)\|_{Q+Q^*}^2. \tag{50}$$

According to (44), we have

$$\langle Q(y^k - \bar{y}^k), Q((y^k - \bar{y}^k) - (y^{k+1} - \bar{y}^{k+1})) \rangle \geq \frac{1}{2} \|(y^k - \bar{y}^k) - (y^{k+1} - \bar{y}^{k+1})\|_{Q+Q^*}^2. \tag{51}$$

Therefore, we obtain

$$\begin{aligned}
 & \left\| \mathbf{y}^{k+1} - \mathbf{y}^k \right\|_G^2 - \left\| \mathbf{y}^{k+2} - \mathbf{y}^{k+1} \right\|_G^2 = \left\| T(\mathbf{y}^k - \bar{\mathbf{y}}^k) \right\|_G^2 - \left\| T(\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}) \right\|_G^2 \\
 & = 2 \left\langle T(\mathbf{y}^k - \bar{\mathbf{y}}^k), GT \left((\mathbf{y}^k - \bar{\mathbf{y}}^k) - (\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}) \right) \right\rangle \\
 & \quad - \left\| T \left((\mathbf{y}^k - \bar{\mathbf{y}}^k) - (\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}) \right) \right\|_G^2 \\
 & \geq \left\| (\mathbf{y}^k - \bar{\mathbf{y}}^k) - (\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}) \right\|_{Q+Q^*}^2 - \left\| T \left((\mathbf{y}^k - \bar{\mathbf{y}}^k) - (\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}) \right) \right\|_G^2 \\
 & = \left\| (\mathbf{y}^k - \bar{\mathbf{y}}^k) - (\mathbf{y}^{k+1} - \bar{\mathbf{y}}^{k+1}) \right\|_{(Q+Q^*-T^*GT)}^2 \geq 0.
 \end{aligned} \tag{52}$$

Referring to (44), we obtain $\left\| \mathbf{y}^{k+1} - \mathbf{y}^k \right\|_G^2 \leq \left\| \mathbf{y}^k - \mathbf{y}^{k-1} \right\|_G^2$ and the proof of Lemma 4 is completed.

Appendix C

Proof of Theorem 2

Proof From (21), we have.

$$\sum_{i=0}^{\infty} \left\| \mathbf{y}^{i+1} - \mathbf{y}^i \right\|_G^2 \leq \left\| \mathbf{y}^0 - \mathbf{y}^* \right\|_G^2 \quad \forall \mathbf{y}^* \in Y^*. \tag{53}$$

According to Lemma 4, we obtain

$$(k+1) \left\| \mathbf{y}^{k+1} - \mathbf{y}^k \right\|_G^2 \leq \sum_{i=0}^k \left\| \mathbf{y}^{i+1} - \mathbf{y}^i \right\|_G^2 \leq \left\| \mathbf{y}^0 - \mathbf{y}^* \right\|_G^2 \quad \forall \mathbf{y}^* \in Y^*, \tag{54}$$

and the proof of Theorem 2 is completed.

Acknowledgements

The authors would like to thank Dr. B. Zhang of Guilin University of Electronic Technology, Dr. G. Lan of University of Florida, Dr. Y. Ouyang of Clemson University, and Dr. X. Ye of Georgia State University for sharing their MATLAB code and data, and free download from their homepages.

Author contributions

Conceptualization, methodology and writing—original draft preparation, CH; supervision and writing—review and editing, WP; writing—review and editing, JW; writing—review and editing, XF; writing—review and editing, LJ. All authors have read and agreed to the published version of the manuscript.

Funding

This research was funded by National Natural Science Foundation of China (Grant Nos. 62473376), Natural Science Foundation of Shaanxi Province (2021KJXX-22), and Special Support Plan for High-level Talents in Shaanxi Province (TZ0328).

Availability of data and materials

The datasets used during the current study are available from the corresponding author on reasonable request.

Declarations

Competing interests

All authors confirm that there are no competing interests.

Received: 29 March 2024 Accepted: 15 October 2024

Published online: 28 October 2024

References

1. C. He, C. Hu, X. Li, "A Parallel Linearized ADMM with Application to Multichannel TGV-based Image Restoration", in *Proc. 2017 IEEE International Conference on Image Processing (ICIP)*, (Beijing, China, 2017), pp. 1187–1191.
2. S. Hurault, U. Kamilov, A. Leclaire et al., Convergent bregman plug-and-play image restoration for poisson inverse problems. *Adv. Neural. Inf. Process. Syst.* **36**, 1–30 (2024)
3. M. Pandey, G. Rawat, P. Kanti, "Image Restoration Application and Methods for Different Images: A Review", in *Proc. 2022 International Conference on Advances in Computing, Communication and Materials (ICACCM)*, (Dehradun, India, 2022), pp. 1–4.
4. W. Wu, Z. Wu, H. Zhang, Weighted bilinear factorization of low-rank matrix with structural smoothness for image denoising. *Multimedia Syst.* **30**(1), 1–21 (2024)
5. N. Kämper, V. Chizhov, J. Weickert, Efficient parallel algorithms for inpainting-based representations of 4K images--part II: spatial and tonal data optimization. *arXiv preprint*. 1–29 (2024). <https://doi.org/10.48550/arXiv.2401.06747>.
6. X. Li, Z. Wang, C. Chen et al., SemID: blind image inpainting with semantic inconsistency detection. *Tsinghua Sci. Technol.* **29**(4), 1053–1068 (2024)
7. J. Zhao, Z. Sun, Z. Zhou et al., An interpretable lightweight deep network with ℓ_p ($0 < p < 1$) model-driven for single image super-resolution. *Neurocomputing* **580**, 127521 (2024)
8. Y. Chen, G. Lan, Y. Ouyang et al., Fast bundle-level methods for unconstrained and ball-constrained convex optimization. *Comput. Optim. Appl.* **73**(1), 159–199 (2019)
9. M.A. Abdelhay, N.O. Korany, S.E. El-Khamy, Synthesis of uniformly weighted sparse concentric ring arrays based on off-grid compressive sensing framework. *IEEE Antennas Wirel. Propag. Lett.* **20**(4), 448–452 (2021)
10. K. A. Bapat, M. Chakraborty, "Convex Combination of Compressed Sensing Algorithms", in *Proc. 2023 31st European Signal Processing Conference (EUSIPCO)*, (Helsinki, Finland, 2023), pp. 1923–1927.
11. J. Yin, J. Jian, X. Jiang et al., A proximal fully parallel splitting method with a relaxation factor for separable convex programming. *Appl. Numer. Math.* **195**, 17–38 (2024)
12. Y. Qu, H. He, D. Han, A partially inertial customized Douglas-Rachford splitting method for a class of structured optimization problems. *J. Sci. Comput.* **98**(1), 1–24 (2024)
13. H. Bauschke, P. Combettes, *Convex analysis and monotone operator theory in hilbert spaces* (Springer, New York, 2017), pp.139–156
14. C. He, C. Hu, W. Zhang et al., A fast adaptive parameter estimation for total variation image restoration. *IEEE Trans. Image Process.* **23**(12), 4954–4967 (2014)
15. R. Liang, X. Wu, Z. Zhang, Linearized alternating direction method of multipliers for elastic-net support vector machines. *Pattern Recogn.* **148**, 110134 (2024)
16. X. Jiang, L. Vandenbergh, Bregman three-operator splitting methods. *J. Optim. Theory Appl.* **196**(3), 936–972 (2023)
17. X. Liu, Y. Wang, D. Li et al., Sparse reconstruction of EMT based on compressed sensing and L_p regularization with the split Bregman method. *Flow Meas. Instrum.* **94**, 102473 (2023)
18. H. Zhang, L. Zhang, H. Yang, Revisiting linearized Bregman iterations under Lipschitz-like convexity condition. *Math. Comput.* **92**(340), 779–803 (2023)
19. T. Randrianarisoa, B. Szabo, Variational Gaussian processes for linear inverse problems. *Adv. Neural. Inf. Process. Syst.* **36**, 28960–28972 (2023)
20. S.K. Sahoo, M. Salo, The linearized Calderón problem for polyharmonic operators. *J. Diff. Equations* **360**, 407–451 (2023)
21. C. He, C. Hu, X. Li et al., A parallel alternating direction method with application to compound l_1 -regularized imaging inverse problems. *Inf. Sci.* **348**, 179–197 (2016)
22. C. He, C. Hu, *Parallel Operator Splitting Algorithms with Application to Imaging Inverse Problems*. (Springer Nature, Singapore, 2023), pp. 107–139.
23. A.M. Teodoro, J.M. Bioucas-Dias, M.A. Figueiredo, A convergent image fusion algorithm using scene-adapted Gaussian-mixture-based denoising. *IEEE Trans. Image Process.* **28**(1), 451–463 (2018)
24. V.S. Unni, K. N. Chaudhury, "Non-Local Patch-Based Regularization for Image Restoration", in *Proc. The 25th IEEE International Conference on Image Processing (ICIP)*, (Athens, Greece, 2018), pp. 1108–1112.
25. Z. Zhang, Y. Lei, D. Wang, et al., "Distributed ADMM Based on Sparse Computation and Allreduce Communication", in *Proc. 2022 IEEE Intl Conf on Parallel and Distributed Processing with Applications, Big Data and Cloud Computing, Sustainable Computing and Communications, Social Computing and Networking (ISPA/BDCloud/SocialCom/SustainCom)*, (Melbourne, Australia, 2022), pp. 722–729.
26. F. Xue, The operator splitting schemes revisited: primal-dual gap and degeneracy reduction by a unified analysis. *Optimization Letters* **18**(1), 155–194 (2024)
27. R. Zhao, C. Xu, W. Mo et al., A second-order generalized total variation with improved alternating direction method of multipliers algorithm for electrical impedance tomography reconstruction. *Appl. Sci.* **14**(4), 1485 (2024)
28. I. Ekeland, R. Témam, *Convex analysis and variational problems* (SIAM, Philadelphia, 1999), pp.385–402
29. K. Bredies, K. Kunisch, T. Pock, Total generalized variation. *SIAM J. Imag. Sci.* **3**(3), 492–526 (2010)
30. J. Xu, S. You, Y. Guo et al., Combining deep image prior and second-order generalized total variance for image denoising. *IEEE Access*. **11**, 67912–67921 (2023)
31. A.E. Mahdaoui, A. Ouahabi, M.S. Moulay, Image denoising using a compressive sensing approach based on regularization constraints. *Sensors* **22**(6), 2199 (2022)
32. T.F. Chan, J. Shen, *Image processing and analysis: variational, pde, wavelet, and stochastic methods* (SIAM, Philadelphia, 2005), pp.245–308
33. C. He, C. Hu, X. Li, W. Zhang, A parallel primal-dual splitting method for image restoration. *Inf. Sci.* **358–359**, 73–91 (2016)
34. Y.C. Tang, C.X. Zhu, M. Wen et al., A splitting primal-dual proximity algorithm for solving composite optimization problems. *Acta Math. Sinica* **33**, 868–886 (2017)
35. P. Chen, J. Huang, X. Zhang, A primal-dual fixed point algorithm for multi-block convex minimization. *J. Comput. Math.* **34**, 723–738 (2016)

36. B. Zhang, Z. Zhu, Linearized proximal alternating direction method of multipliers for parallel magnetic resonance imaging. *IEEE/CAA J. Automatica Sinica*. **4**(4), 763–769 (2016)
37. Y. Chen, W. Hager, M. Yashtini, X. Ye, H. Zhang, Bregman operator splitting with variable stepsize for total variation image reconstruction. *Comput. Optim. Appl.* **54**, 317–342 (2013)

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.