Adv Comput Math (2013) 38:401–426
DOI 10.1007/s10444-011-9243-y

Proximity algorithms for the L1/TV image denoising model

Charles A. Micchelli · Lixin Shen · Yuesheng Xu · Xueying Zeng

Received: 10 September 2011 / Accepted: 7 October 2011 /
Published online: 21 October 2011

Abstract This paper introduces a proximity operator framework for studying the L1/TV image denoising model which minimizes the sum of a data fidelity term measured in the ℓ^1-norm and the total-variation regularization term. Both terms in the model are non-differentiable. This causes algorithmic difficulties for its numerical treatment. To overcome the difficulties, we formulate the total-variation as a composition of a convex function (the ℓ^1-norm or the ℓ^2-norm) and the first order difference operator, and then express the solution of the model in terms of the proximity operator of the composition. By developing a "chain rule" for the proximity operator of the composition, we identify the solution as fixed point of a nonlinear mapping expressed in terms of the proximity operator of the ℓ^1-norm or the ℓ^2-norm, each of which is explicitly given. This formulation naturally leads to fixed-point algorithms for the numerical treatment of the model. We propose an alternative model by

Communicated by Zhongying Chen.

C. A. Micchelli
Department of Mathematics and Statistics, SUNY,
Albany, NY, USA

C. A. Micchelli
Department of Mathematics, City University of Hong Kong,
Kowloon, Hong Kong

L. Shen · Y. Xu (✉)
Department of Mathematics, Syracuse University,
Syracuse, NY 13244, USA
e-mail: yxu06@syr.edu

Y. Xu · X. Zeng
Guangdong Province Key Lab of Computational Science, Sun Yat-sen University,
Guangzhou 510275, People's Republic of China

replacing the non-differentiable convex function in the formulation of the total variation with its differentiable Moreau envelope and develop corresponding fixed-point algorithms for solving the new model. When partial information of the underlying image is available, we modify the model by adding an indicator function to the minimization functional and derive its corresponding fixed-point algorithms. Numerical experiments are conducted to test the approximation accuracy and computational efficiency of the proposed algorithms. Also, we provide a comparison of our approach to two state-of-the-art algorithms available in the literature. Numerical results confirm that our algorithms perform favorably, in terms of PSNR-values and CPU-time, in comparison to the two algorithms.

Keywords L1/TV model · Proximity operator · Moreau envelope · Inpainting

Mathematics Subject Classifications (2010) 65T60 · 68U10 · 94A08

1 Introduction

Total-variation based variational models are widely used in image denoising. The well-known Rudin–Osher–Fatemi (ROF) image denoising model [31] seeks a minimizer of the sum of a data fidelity term measured in the square of ℓ^2-norm and the total-variation regularization term. This minimization problem is often referred to as the L2/TV model. The ℓ^2-norm fidelity term is particularly effective for treating the Gaussian noise and the total-variation regularization allows the reconstructed image to have sharp edges. However, both theoretical study and numerical experiments show that the ℓ^2-norm fidelity term is less effective for the non-Gaussian additive noise because it tends to amplify the effect of outliers in the given image. To overcome the drawback of the L2/TV model, an alternative formulation has been used to minimize the sum of a data fidelity term measured in the ℓ^1-norm and the total-variation regularization term. The ℓ^1-norm can effectively suppress the effect of outliers that may contaminate a given image. It was demonstrated in [1, 30] that the ℓ^1-norm fidelity term is particularly suitable for handling non-Gaussian additive noise such as impulsive noise and Laplace noise [5]. We shall refer to such a formulation as the L1/TV (image denoising) model. In contrast to the L2/TV model, the L1/TV model has many distinctive and desirable features. For example, the L1/TV model does not erode geometric structures of the images under processing and possesses properties such as contrast invariant, data driven parameter selection, multiscale image decomposition, and morphological invariance [13, 30, 38]. Applications of the L1/TV model include computer vision [15], biomedical imaging [37], optical flow and object tracking [40], and shape denoising [40]. In light of the interesting features of the

L1/TV model and its successful applications, it is highly desirable to develop efficient and fast algorithms for numerical treatment of the model.

The use of the ℓ^1-norm fidelity term and the total-variation regularization term in the L1/TV model introduces algorithmic difficulties for the numerical treatment of the model due to the non-differentiability of both terms in its minimization functional. To overcome the difficulties, several numerical approaches were proposed recently for solving the L1/TV model. The paper [13] introduced a time-marching PDE approach in which the solution of the L1/TV model is viewed as a steady solution of the Euler–Lagrange equation of the model and it is obtained by solving the related PDE. This approach also has its drawback. Due to the nonsmoothness of the ℓ^1 norm, the artificial time step needs to be very small when the system associated with the Euler–Lagrange equation is close to its steady state. This causes numerical difficulties. In [21], the L1/TV model was formulated as a linear or quadratic programming problem which was solved by the interior point method. At each iteration of the interior point method, a structured linear system must be solved. The preconditioned conjugate gradient method with factorized sparse inverse preconditioners was employed to solve such structured inner systems. In [22], a method based on second order cone programming was proposed. In [35], the augmented Lagrangian method was applied to the L1/TV model. In recent work [2, 16, 24, 36] on the numerical treatment of the L1/TV model, a commonly used approach was used to introduce various auxiliary variables in the L1/TV model aiming at overcoming the non-differentiability of its fidelity and regularization terms. A Fenchel-duality principle was used in [19] to avoid treating directly the non-differentiable functional in the L1/TV model, the associated dual problem was treated by adding two regularization terms, and the resulting primal-dual system was then solved by a semi-smooth Newton solver. Therefore, it appears that all aforementioned work avoids solving the L1/TV model directly, but rather solves its various modified versions.

The main purpose of this paper is to introduce a proximity operator framework for the *direct* treatment of the L1/TV model. In [28], by identifying the total-variation as a composition of the convex function which defines the ℓ^1-norm or the ℓ^2-norm and the first order difference operator, we characterized the solution of the L2/TV model in terms of the proximity operator of the convex function with a fixed-point equation related to the convex function and the first order difference operator, and developed fixed-point algorithms for the numerical treatment of the model based on this characterization. Adopting this viewpoint, we shall again formulate the total-variation as a composition of the convex function which defines the ℓ^1-norm or the ℓ^2-norm and the first order difference operator, and characterize the solution of the L1/TV model in terms of the proximity operator of the composition. Developing proximity algorithms for the L1/TV model is more involved than that for the L2/TV model since the use of the ℓ^1 norm in the fidelity term causes further difficulty in the numerical treatment of the model due to its non-differentiability. This

difficulty will be overcome by introducing a new fixed-point equation. For potential wider applicability of our methods, we consider a general model which minimizes the ℓ^1 fidelity term plus a regularization term in the form of the composition of a convex function and a matrix over a closed convex set in the Euclidean space. This model contains the L1/TV model and the L1/TV inpainting model as special cases. By developing a chain rule for the proximity operator of the composition, we identify a solution of the model in terms of fixed point equations via the proximity operator of the convex function. The fixed point equations contain two equations, one reflecting the composition of the convex function with the matrix and the other addressing the non-differentiability of the ℓ^1 fidelity term.

This paper is organized into six sections. In Section 2 we investigate a general model constructed as a minimization of the sum of the ℓ^1 fidelity term, an indicator term and a regularization term in the form of the composition of a convex function and a matrix. This model includes the L1/TV model and the L1/TV inpainting model as special cases. The solution of this model is identified as a fixed-point of a system of two equations. When the convex function is differentiable, the system reduces to a single fixed-point equation. Section 3 is devoted to a study of the model when the convex function involved in the regularization term is replaced by its Moreau envelope. The envelope preserves the main geometric feature of the convex function and at the same time it is differentiable. We specify in Section 4 the results obtained in Sections 2 and 3 to the L1/TV image denoising model and derive specific algorithms accordingly. Numerical results are presented in Section 5 for impulsive noise removal to demonstrate the approximation accuracy and computational efficiency of the proposed algorithms. They confirm that the PSNR-values and CPU-time of the proposed algorithms compare favorably to those obtained from the state-of-the-art algorithms developed in [19, 36]. The fast convergence of the proposed algorithms comes from the fact that each iterative step only uses the operations of shrinkage and finite differences. We summarize our conclusions in the last section.

2 The L1/$\varphi \circ$ B model

In this section, we study a general minimization model which we refer to as the L1/$\varphi \circ$ B model. This model minimizes the sum of the ℓ^1 fidelity term and the composition of a convex function with a matrix and includes the L1/TV denoising model and the L1/TV inpainting model as special cases.

We begin with our preferred notation. By $\mathbb{R}^d$, we denote the usual d-dimensional Euclidean space. For a natural number k, we let $\mathbb{N}_k := \{1, 2, \ldots, k\}$. For $x \in \mathbb{R}^d$, its i-th entry is denoted by x_i. For $x, y \in \mathbb{R}^d$, we define $\langle x, y\rangle := \sum_{i\in\mathbb{N}_d} x_i y_i$, the standard inner product of $\mathbb{R}^d$. We use $\|\cdot\|_1$ and $\|\cdot\|_2$, respectively, to denote the ℓ^1 and ℓ^2 norm on $\mathbb{R}^d$.

We now describe the general model. Suppose that φ is a given convex function on $\mathbb{R}^m$ and B is a given $m \times d$ matrix. We denote by $\mathcal{C}$ a nonempty

closed convex set of $\mathbb{R}^d$. For a given $x \in \mathbb{R}^d$ and a nonempty closed convex subset $\mathcal{C}$ of $\mathbb{R}^d$, we consider the following constrained minimization problem

$$\min \{\lambda \|u - x\|_1 + (\varphi \circ B)(u) : u \in \mathcal{C}\}.$$

The convex set $\mathcal{C}$ in the above model may be used to describe available constraints. This constrained model may be rewritten as a non-constrained model by introducing the indicator function of the set $\mathcal{C}$, defined at $u \in \mathbb{R}^d$ as

$$\iota_{\mathcal{C}}(u) := \begin{cases} 0, & u \in \mathcal{C}; \\ +\infty, & u \notin \mathcal{C}. \end{cases}$$

Using the indicator function, the above constrained minimization problem is reformulated as an equivalent unconstrained minimization problem

$$\min \left\{ J_{\mathrm{L1}/\varphi\circ\mathrm{B}}(u) : u \in \mathbb{R}^d \right\} \tag{2.1}$$

where for $u \in \mathbb{R}^d$ we define

$$J_{\mathrm{L1}/\varphi\circ\mathrm{B}}(u) := \lambda \|u - x\|_1 + \iota_{\mathcal{C}}(u) + (\varphi \circ B)(u), \quad u \in \mathbb{R}^d.$$

We refer the minimization problem (2.1) to as the L1/$\varphi \circ$ B model. The functional $J_{\mathrm{L1}/\varphi\circ\mathrm{B}}(u)$ is convex and approaches to infinity whenever $\|u\|_2$ goes to infinity. Hence, a solution to model (2.1) exists. On the other hand, since the functional $J_{\mathrm{L1}/\varphi\circ\mathrm{B}}$ is not in general strictly convex, solutions of model (2.1) may not be unique. This is an intrinsic feature of model (2.1).

The composition $\varphi \circ B$ appearing in the L1/$\varphi \circ$ B model can cover several important scenarios in image processing, including the following:

- Total-variation regularization. *Anisotropic total-variation* and *isotropic total-variation* are the standard choices used for discrete grayscale images. For the anisotropic total-variation φ is the norm $\|\cdot\|_1$ while for the isotropic total-variation φ is a linear combination of the norm $\|\cdot\|_2$ in $\mathbb{R}^2$. The matrix B for the both cases is the first order difference matrix. For higher-order total-variations (see, e.g., [11, 14, 27, 32, 33, 39]), B may be chosen to be a higher-order difference matrix.
- Sparse regularization of wavelet coefficients. The use of sparsity for image restoration has gained much attention recently (see, e.g., [8–10, 12, 18, 20, 34]). Images are well-represented by sparse expansions with respect to wavelets or framelets. In the wavelet framework, φ is the norm $\|\cdot\|_1$ and B is usually chosen to be a wavelet transform matrix.
- Optimal MRI reconstruction. The regularization functional in the optimal MRI reconstruction [26] is defined in terms of a combination of the total variation and a Besov norm. In this case, φ is the norm $\|\cdot\|_1$ and B is the first order difference matrix concatenated by the Haar wavelet transform matrix.

When $\mathcal{C}$ is chosen to be the entire space $\mathbb{R}^d$ and $\varphi \circ B$ the total-variation, the resulting L1/$\varphi \circ$ B model becomes the L1/TV model. If the convex set $\mathcal{C}$, associated with a given vector $x \in \mathbb{R}^d$ and a proper subset $\Lambda \subset \mathbb{N}_d$, is chosen as

$$\mathcal{C} := \{y : P_\Lambda y = P_\Lambda x\} \tag{2.2}$$

where P_Λ is a $d \times d$ diagonal matrix with diagonal entries 1 for indices in Λ and 0 otherwise, the resulting L1/$\varphi \circ$ B model is used for an inpainting problem.

We now characterize a solution of the L1/$\varphi \circ$ B model (2.1) in terms of proximity operators. We recall the notions of subdifferential and the proximity operator of a convex function.

Definition 1 Let $\psi : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ be a convex function, not identically equal to $+\infty$. The subdifferential of ψ at $x \in \mathbb{R}^d$ with $\psi(x) < +\infty$ is the set defined by

$$\partial\psi(x) := \left\{y : \ y \in \mathbb{R}^d \textit{ and } \psi(z) \geq \psi(x) + \langle y, z - x\rangle \ \textit{ for all } z \in \mathbb{R}^d\right\}.$$

It is well-known (cf. [4, page 732]) that the subdifferential of a convex function ψ is a set-valued mapping from $\mathbb{R}^d$ into nonempty convex compact sets in $\mathbb{R}^d$.

Definition 2 Let $\psi : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ be a convex function, not identically equal to $+\infty$. The proximity operator of ψ is defined for $x \in \mathbb{R}^d$ by

$$\text{prox}_\psi(x) := \arg\min\left\{\frac{1}{2}\|u - x\|_2^2 + \psi(u) : u \in \mathbb{R}^d\right\}.$$

The subdifferential of a convex function may be characterized in terms of its proximity operator.

Lemma 1 *If ψ is a convex function on $\mathbb{R}^d$ and $x \in \mathbb{R}^d$, then*

$$y \in \partial\psi(x) \quad \textit{if and only if} \quad x = \text{prox}_\psi(x + y).$$

A simple proof of this fact was given in [28]. As a direct consequence of Lemma 1, we have that

$$y \in \partial\psi(x) \quad \text{if and only if} \quad y = \left(I - \text{prox}_\psi\right)(x + y). \tag{2.3}$$

With the help of Lemma 1, we next characterize a solution of model (2.1). To this end, for a given $x \in \mathbb{R}^d$, a positive number λ, and a convex set $\mathcal{C}$ in $\mathbb{R}^d$, we define a function ϱ on $\mathbb{R}^d$ at $v \in \mathbb{R}^d$ as

$$\varrho(v) := \lambda\|v - x\|_1 + \iota_\mathcal{C}(v). \tag{2.4}$$

Proposition 1 *If φ is a real-valued convex function on $\mathbb{R}^m$, B an $m \times d$ matrix, and $u \in \mathbb{R}^d$ a solution of model (2.1), then for any $\alpha, \beta > 0$ there exists a vector $b \in \mathbb{R}^m$ such that*

$$u = \operatorname{prox}_{\frac{1}{\lambda\alpha}\varrho}\left(u - \frac{\beta}{\lambda\alpha} B^\top b\right), \tag{2.5}$$

$$b = \left(I - \operatorname{prox}_{\frac{1}{\beta}\varphi}\right)(Bu + b). \tag{2.6}$$

Conversely, if there exist $\alpha, \beta > 0$, $b \in \mathbb{R}^m$, and $u \in \mathbb{R}^d$ satisfying (2.5) and (2.6), then u is a solution of model (2.1).

Proof We assume that $u \in \mathbb{R}^d$ is a solution of model (2.1). By the Fermat rule in convex analysis

$$0 \in \partial J_{\mathrm{L1}/\varphi\circ\mathrm{B}}(u)$$

which, by the chain rule

$$\partial\,(\varphi \circ B)\,(u) = B^\top \partial\varphi(Bu)$$

is equivalent to the condition that

$$0 \in \partial\varrho(u) + B^\top \partial\varphi(Bu). \tag{2.7}$$

For any numbers $\alpha, \beta > 0$, we can choose a vector $a \in \frac{1}{\lambda\alpha}\partial\varrho(u)$ and a vector $b \in \frac{1}{\beta}\partial\varphi(Bu)$ such that

$$0 = \lambda\alpha a + \beta B^\top b\,. \tag{2.8}$$

By Lemma 1, from the inclusion $b \in \frac{1}{\beta}\partial\varphi(Bu)$ we get (2.6) and from the inclusion $a \in \frac{1}{\lambda\alpha}\partial\varrho(u)$ we have that

$$u = \operatorname{prox}_{\frac{1}{\lambda\alpha}\varrho}(u + a). \tag{2.9}$$

Using (2.8) we conclude that $a = -\frac{\beta}{\lambda\alpha} B^\top b$ and substitute it into (2.9) we obtain (2.5).

Conversely, suppose that there exist $\alpha, \beta > 0$, $b \in \mathbb{R}^m$, and $u \in \mathbb{R}^d$ satisfying (2.5) and (2.6). Again, by Lemma 1, (2.6) and (2.5) ensure that $b \in \frac{1}{\beta}\partial\varphi(Bu)$ and $-\frac{\beta}{\lambda\alpha} B^\top b \in \frac{1}{\lambda\alpha}\partial\varrho(u)$, respectively. Combining these two inclusion relations yields that

$$0 = \lambda\alpha \cdot \left(-\frac{\beta}{\lambda\alpha} B^\top b\right) + \beta B^\top b \in \partial\varrho(u) + B^\top \partial\varphi(Bu).$$

This in turn implies that $u \in \mathbb{R}^d$ is a solution of model (2.1). □

The following result is a direct consequence of Proposition 1 for the convex function φ being differentiable. A solution of the corresponding model is characterized in terms of only one fixed-point equation.

Corollary 1 *Let $x \in \mathbb{R}^d$ be given, λ be a positive number, and B be an $m \times d$ matrix. Suppose that $\mathcal{C}$ is a convex set in $\mathbb{R}^d$, ϱ is the function defined by (2.4), and φ is a differentiable convex function on $\mathbb{R}^m$. If $u \in \mathbb{R}^d$ is a solution of model (2.1), then for any $\alpha > 0$*

$$u = \operatorname{prox}_{\frac{1}{\lambda\alpha}\varrho}\left(u - \frac{1}{\lambda\alpha} B^\top \nabla\varphi(Bu)\right). \tag{2.10}$$

Conversely, if $u \in \mathbb{R}^d$ satisfies (2.10) for some $\alpha > 0$, then u is a solution of model (2.1).

Proof According to Proposition 1, a solution u of model (2.1) satisfies the fixed-point (2.5) and (2.6). Notice that the subdifferential of a differentiable function φ at a point $v \in \mathbb{R}^m$ is a singleton set, that is, $\partial\varphi(v) = \{\nabla\varphi(v)\}$, where $\nabla\varphi(v)$ denotes the gradient of φ at v. Hence, by (2.3), (2.6) implies that $b = \frac{1}{\beta}\nabla\varphi(Bu)$. Therefore, (2.5) yields the fixed-point (2.10). □

We remark that the fixed-point (2.10) can be viewed as a special case of the splitting forward-backward formulation described in [17]. Furthermore, if the gradient of the convex function φ in Corollary 1 is Lipschitz continuous with Lipschitz constant L, that is,

$$\|\nabla\varphi(p) - \nabla\varphi(q)\|_2 \le L\|p - q\|_2, \quad \text{for all } p, q \in \mathbb{R}^m$$

and if α is chosen to satisfy

$$\frac{1}{\lambda\alpha} < \frac{2}{L\|B\|_2^2} \tag{2.11}$$

where $\|B\|_2$ denotes the largest singular value of B, then it was proved in [17] that for any initial guess $u^0 \in \mathbb{R}^d$, the Picard iteration

$$u^{k+1} = \operatorname{prox}_{\frac{1}{\lambda\alpha}\varrho}\left(u^k - \frac{1}{\lambda\alpha} B^\top \nabla\varphi\left(Bu^k\right)\right) \tag{2.12}$$

converges to a fixed-point of (2.10), which is a minimizer of model (2.1). In order to make this paper self-contained we provide a brief proof for convergence of iteration scheme (2.12). To this end we briefly review the notion of nonexpansive mappings which are crucial for convergence analysis of the Picard sequence.

Let Ω be a closed convex nonempty subset of $\mathbb{R}^d$. According to [3, 7], a mapping $Q : \Omega \to \mathbb{R}^d$ is called nonexpansive if for all $x, y \in \Omega$ we have that

$$\|Qx - Qy\|_2 \le \|x - y\|_2.$$

An averaged nonexpansive mapping is a nonexpansive mapping that can be written as

$$(1 - \kappa)I + \kappa N, \tag{2.13}$$

for some positive number κ in $(0, 1)$ and some nonexpansive operator N. Moreover, a mapping of the form given in (2.13) is called κ-averaged and an

operator is called firmly nonexpansive if it is $\frac{1}{2}$-averaged. As an example, if ψ is a convex function on $\mathbb{R}^d$ not identically equal to $+\infty$, then both prox_ψ and $I - \text{prox}_\psi$ are firmly nonexpansive [17]. The Krasnosel'skii–Mann algorithm is a powerful way to find fixed-points of averaged nonexpansive operators.

Lemma 2 (Krasnosel'skii–Mann Algorithm) *If Q is an averaged nonexpansive mapping on $\mathbb{R}^d$ and the fixed-point set $\text{Fix}(Q) := \{x : Qx = x\}$ of Q is nonempty, then for any initial vector in $\mathbb{R}^d$ the corresponding Picard sequence converges to a member of $\text{Fix}(Q)$.*

We now return to the study of convergence of the sequence generated by (2.12). To this end, for a given $x \in \mathbb{R}^d$ and positive numbers λ and α, we define operator $H : \mathbb{R}^d \to \mathbb{R}^d$ at $u \in \mathbb{R}^d$ by the equation

$$Hu := u - \frac{1}{\lambda\alpha} B^\top \nabla\varphi(Bu) \tag{2.14}$$

and mapping $P : \mathbb{R}^d \to \mathbb{R}^d$ at $u \in \mathbb{R}^d$ by

$$Pu := \left(\text{prox}_{\frac{1}{\lambda\alpha}\varrho} \circ H\right) u. \tag{2.15}$$

With this notation, (2.12) is rewritten as

$$u^{k+1} = Pu^k. \tag{2.16}$$

The averaged nonexpansivity of the operator P defined by (2.15) is established below.

Proposition 2 *Let $x \in \mathbb{R}^d$ be given, B be an $m \times d$ matrix, φ be a differentiable convex function on $\mathbb{R}^m$, and $\nabla\varphi$ be Lipchitz continuous, that is, for some $L > 0$,*

$$\|\nabla\varphi(p) - \nabla\varphi(q)\|_2 \le L\|p - q\|_2 \quad \text{for all} \quad p, q \in \mathbb{R}^m. \tag{2.17}$$

If α is chosen to satisfy (2.11) then the operator P defined by (2.15) is averaged nonexpansive.

Proof Since the proximity operator $\text{prox}_{\frac{1}{\lambda\alpha}\varrho}$ is $\frac{1}{2}$-averaged and the composition of two averaged nonexpansive operators is still averaged nonexpansive, to prove the averaged nonexpansivity of operator P, it suffices to show that the operator H defined by (2.14) is averaged nonexpansive.

Let η be a positive number in (0, 1). We define the operator $F : \mathbb{R}^d \to \mathbb{R}^d$ at $u \in R^d$ by

$$Fu := \frac{1}{2\lambda\alpha\eta} B^\top \nabla\varphi(Bu)$$

and observe that the operator H can be decomposed as

$$H = (1 - \eta)I + \eta(I - 2F).$$

Clearly, H is η-averaged if $I - 2F$ is nonexpansive. Indeed, under the condition (2.11), $I - 2F$ is nonexpansive. Since the nonexpansivity of $I - 2F$ is

equivalent to the firm nonexpansivity of F, we shall prove the later. For this purpose, we see from [25, Theorem 4.2.2] that our hypothesis on $\nabla\varphi$ implies that for any $u, v \in \mathbb{R}^d$

$$\|\nabla\varphi(Bu) - \nabla\varphi(Bv)\|_2^2 \le L \langle \nabla\varphi(Bu) - \nabla\varphi(Bv), Bu - Bv\rangle .$$

This inequality, combined with the definition of F, yields for $u, v \in \mathbb{R}^d$ the inequality

$$\begin{aligned}\|Fu - Fv\|_2^2 &\le \frac{L}{(2\lambda\alpha\eta)^2} \left\|B^\top\right\|_2^2 \langle \nabla\varphi(Bu) - \nabla\varphi(Bv), Bu - Bv\rangle \\ &= \frac{L}{2\lambda\alpha\eta} \left\|B^\top\right\|_2^2 \langle Fu - Fv, u - v\rangle .\end{aligned}$$

In view of $\eta \in (0, 1)$ and the inequality (2.11), the above inequality ensures that F is firmly nonexpansive. This completes the proof of the proposition. □

Proposition 3 *Let $x \in \mathbb{R}^d$ be given, B be an $m \times d$ matrix, $\mathcal{C}$ be a convex set in $\mathbb{R}^d$ and ϱ be a function defined by (2.4). Suppose that φ is a differentiable convex function on $\mathbb{R}^m$, and $\nabla\varphi$ is Lipschitz continuous with Lipschitz constant L. If the parameter α is chosen to satisfy condition (2.11), then for any initial vector $u^0 \in \mathbb{R}^d$ the sequence $\{P^k u^0 : k \in \mathbb{N}\}$ converges to a solution of model (2.1).*

Proof Proposition 1 ensures that a fixed-point of operator P defined by (2.15) gives a solution of model (2.1). It remains to prove that the Picard iteration sequence $\{P^k u^0 : k \in \mathbb{N}\}$ converges to a fixed-point of P. Since the parameter α satisfies condition (2.11), by Proposition 2 the operator P is averaged nonexpansive. Thus, Lemma 2 ensures that the sequence $\{P^k u^0 : k \in \mathbb{N}\}$ converges to a solution of the fixed-point (2.10) for any vector $u^0 \in \mathbb{R}^d$. □

The following lemma is necessary to give the proximity operator of the function ϱ defined by (2.4).

Lemma 3 *If $\lambda, \alpha > 0$, $x \in \mathbb{R}^d$, $\Lambda \subset \mathbb{N}_d$, $\mathcal{C}$ is the set defined by (2.2) and ϱ is the function defined by (2.4), then for $v \in \mathbb{R}^d$,*

$$\operatorname{prox}_{\frac{1}{\lambda\alpha}\varrho}(v) = x + (I - P_\Lambda) \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(v - x). \tag{2.18}$$

Proof By the definition of the proximity operator, we have for $v \in \mathbb{R}^d$ that

$$\operatorname{prox}_{\frac{1}{\lambda\alpha}\varrho}(v) = \arg\min\left\{\frac{1}{2}\|y - v\|_2^2 + \frac{1}{\alpha}\|y - x\|_1 + \iota_{\mathcal{C}}(y) : y \in \mathbb{R}^d\right\}.$$

Since the convex set $\mathcal{C}$ has the structure described in (2.2), we get that

$$\operatorname{prox}_{\frac{1}{\lambda\alpha}\varrho}(v) = P_\Lambda x + (I - P_\Lambda) \cdot \arg\min \left\{ \frac{1}{2}\|y - v\|_2^2 + \frac{1}{\alpha}\|y - x\|_1 : y \in \mathbb{R}^d \right\}. \tag{2.19}$$

Again, using the definition of the proximity operator, we derive from (2.19) that

$$\operatorname{prox}_{\frac{1}{\lambda\alpha}\varrho}(v) = P_\Lambda x + (I - P_\Lambda)\left(x + \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(v - x)\right),$$

which establishes (2.18). □

Specializing Lemma 3 to the convex set $\mathcal{C}$ defined by (2.2), equation (2.5) becomes

$$u = x + (I - P_\Lambda) \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(u - x - \frac{\beta}{\lambda\alpha} B^\top b\right). \tag{2.20}$$

As a result, a solution of model (2.1) for the convex set $\mathcal{C}$ defined by (2.2) is characterized in terms of the fixed-point (2.20) and (2.6).

An approximate solution of the fixed-point (2.10) gives an approximate solution of model (2.1) with a differentiable convex function φ. In particular, in this case, when the convex set $\mathcal{C}$ is defined by (2.2), according to Lemma 3, (2.10) becomes

$$u = x + (I - P_\Lambda) \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(u - x - \frac{1}{\lambda\alpha} B^\top \nabla\varphi(Bu)\right). \tag{2.21}$$

3 The L1/env ∘ B model

In the preceding section, we studied the L1/φ ∘ B model under various hypotheses. We noticed, by imposing the differentiability assumption on the convex function φ, that the coupled system of fixed-point (2.5) and (2.6) for a solution of the model reduces to a single fixed-point (2.10). This has obvious computational advantages. However, in the image denoising context, the convex function used in the definition of the total-variation is often not differentiable so that the denoised images preserve sharp edges. To balance between the need of preserving edges and computational efficiency, we propose to replace the convex function φ by its "smoothed" version which preserves main geometric features of φ but, at the same time, is differentiable. For this purpose, we study in this section the L1/env ∘ B model which is a variant of the L1/φ ∘ B model with φ being replaced by its Moreau envelope.

For a positive number γ and a convex function ψ on $\mathbb{R}^m$, the Moreau envelope of ψ with parameter γ at $z \in \mathbb{R}^m$ is defined as

$$\operatorname{env}_{\frac{1}{\gamma}\psi}(z) := \min \left\{ \frac{\gamma}{2}\|y - z\|_2^2 + \psi(y) : y \in \mathbb{R}^m \right\}. \tag{3.1}$$

The envelope $\text{env}_{\frac{1}{\gamma}\psi}$ is always differentiable and its gradient given by

$$\nabla\left(\text{env}_{\frac{1}{\gamma}\psi}\right) = \gamma\left(I - \text{prox}_{\frac{1}{\gamma}\psi}\right) \tag{3.2}$$

is Lipschitz continuous with Lipschitz constant γ due to the fact that $I - \text{prox}_{\frac{1}{\gamma}\psi}$ is Lipschtiz continuous with Lipschitz constant 1 (see, e.g., [17]). Moreover, for any positive number γ the envelope $\text{env}_{\frac{1}{\gamma}\psi}$ is bounded above by ψ and it converges to ψ as $\gamma \to +\infty$. That is, for $z \in \mathbb{R}^m$ we have that $\text{env}_{\frac{1}{\gamma}\psi}(z) \le \psi(z)$ and $\text{env}_{\frac{1}{\gamma}\psi}(z) \to \psi(z)$, as $\gamma \to +\infty$ (see, e.g., [29]). These remarkable properties of the envelope motivate us to replace the convex function φ in the L1/$\varphi \circ$ B model by its envelope $\text{env}_{\frac{1}{\gamma}\varphi}$ for some positive number γ. The resulting model is then given by

$$\min\left\{\lambda\|u - x\|_1 + \iota_{\mathcal{C}}(u) + \text{env}_{\frac{1}{\gamma}\varphi}(Bu) : u \in \mathbb{R}^d\right\}. \tag{3.3}$$

We refer to this model as the L1/env $\circ$ B model.

We present below the main result of this section.

Proposition 4 *Let $x \in \mathbb{R}^d$ be given, λ be a positive number, and B be an $m \times d$ matrix. Suppose that $\mathcal{C}$ is a convex set in $\mathbb{R}^d$, ϱ is a function defined by (2.4), and φ is a convex function on $\mathbb{R}^m$. If u is a solution of model (3.3), then for any $\alpha > 0$, $u \in \mathbb{R}^d$ is a fixed-point of the mapping Q defined by*

$$Qu := \text{prox}_{\frac{1}{\lambda\alpha}\varrho}\left(u - \frac{\gamma}{\lambda\alpha}B^\top\left(I - \text{prox}_{\frac{1}{\gamma}\varphi}\right)(Bu)\right). \tag{3.4}$$

Conversely, if there exist $\alpha > 0$ and $u \in \mathbb{R}^d$ such that u is a fixed-point of mapping Q, then u is a solution of model (3.3).

Proof By (3.2), we know that $\text{env}_{\frac{1}{\gamma}\varphi}$ is differentiable. Using Corollary 1 with φ replaced by its envelope and the gradient of $\text{env}_{\frac{1}{\gamma}\varphi}$ by $\nabla\text{env}_{\frac{1}{\gamma}\varphi} = \gamma(I - \text{prox}_{\frac{1}{\gamma}\varphi})$, we conclude that the result of this proposition holds. □

By Proposition 3, if the parameter α is chosen such that $\frac{\gamma}{\lambda\alpha} < \frac{2}{\|B\|_2^2}$, for any initial vector $u^0 \in \mathbb{R}^d$, the iteration $u^{k+1} = Qu^k$ converges to a solution of model (3.3).

Finally, when the convex set $\mathcal{C}$ is given by (2.2), we appeal to Lemma 3 and conclude that the fixed-point equation of mapping Q becomes

$$u = x + (I - P_\Lambda)\,\text{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(u - x - \frac{\gamma}{\lambda\alpha}B^\top\left(I - \text{prox}_{\frac{1}{\gamma}\varphi}\right)(Bu)\right). \tag{3.5}$$

4 The L1/TV image denoising model

In this section, we apply the general methods and theory developed in the previous two sections to the L1/TV image denoising model.

We consider a digital image as an $n \times n$ square matrix for some positive integer n. As a matter of fact, for convenience of exposition, we treat an image matrix as a vector in the vector space $\mathbb{R}^{n^2}$ in such a way that the ij-th entry of the image matrix corresponds to the $((i-1)n+j)$-th entry of the associated vector in $\mathbb{R}^{n^2}$. The L1/TV image denoising model (resp. the modified L1/TV image denoising model) is a special case of the L1/$\varphi \circ$ B model (resp. the L1/env $\circ$ B model), with appropriate identification of the convex function φ and the matrix B in the general models.

To reexpress the total-variation of an image in the form of the composition of a convex function φ and a matrix B, we need an $n \times n$ "local difference" matrix D.

$$D := \begin{bmatrix} 0 & & & \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix}.$$

The matrix D will be used to "differentiate" a row or a column of an image matrix. For matrices P and Q, we use $P \otimes Q$ to denote their Kronecker product. Corresponding to the $n \times n$ identity matrix I_n and the local difference matrix D, we define the $2n^2 \times n^2$ "global difference" matrix B by

$$B := \begin{bmatrix} I_n \otimes D \\ D \otimes I_n \end{bmatrix}. \tag{4.1}$$

The matrix B will be used to "differentiate" the entire image matrix.

A generic form of the total-variation of an image $u \in \mathbb{R}^{n^2}$ is $\varphi(Bu)$ with B given by (4.1) above and the convex function φ according to whether the total-variation is anisotropic or isotropic. For the anisotropic and the isotropic total-variation, $\varphi : \mathbb{R}^{2n^2} \to \mathbb{R}$ is defined at $z \in \mathbb{R}^{2n^2}$, respectively, as

$$\varphi(z) := \|z\|_1 \quad \text{and} \quad \varphi(z) := \sum_{i \in \mathbb{N}_{n^2}} \left\| \left[z_i, z_{n^2+i} \right]^\top \right\|_2. \tag{4.2}$$

Hence, the L1/TV image denoising model is model (2.1) with the convex function φ and the matrix B chosen above, and $\mathcal{C} = \mathbb{R}^{n^2}$. According to Proposition 1, a solution of the L1/TV image denoising model is characterized by the fixed-point (2.5) and (2.6), with the above special choice of the convex function φ, the matrix B and $\mathcal{C}$. In fact, in this special case, (2.5) reduces to

$$u = x + \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1} \left(u - x - \frac{\beta}{\lambda\alpha} B^\top b \right).$$

The following algorithm describes specific procedures for finding a solution of the above L1/TV image denoising model according to the fixed-point equations.

Algorithm 1

1. Given: noisy image $x \in \mathbb{R}^{n^2}$; $\lambda > 0, \alpha > 0, \beta > 0, \alpha_{\max}, \beta_{\max}$ and $\mu \in \mathbb{N}$
2. Initialization: $u^1 = x$, and $b^1 = 0$
3. For $k \in \mathbb{N}$,
   (a) if $\alpha < \alpha_{\max}$ and $\beta < \beta_{\max}$, and k is a multiple of μ, update $\alpha \longleftarrow 2\alpha$ and $\beta \longleftarrow 2\beta$; if $\alpha \geq \alpha_{\max}$ or $\beta \geq \beta_{\max}$, then set $\alpha = \alpha_{\max}$ and $\beta = \beta_{\max}$;
   (b) $u^{k+1} \longleftarrow x + \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(u^k - x - \frac{\beta}{\lambda\alpha} B^\top b^k\right)$
   (c) $b^{k+1} \longleftarrow \left(I - \operatorname{prox}_{\frac{1}{\beta}\varphi}\right)\left(Bu^{k+1} + b^k\right)$
   (d) Stop, if u^{k+1} converges or satisfies a stopping criteria; otherwise, set $k \leftarrow k + 1$.

In Algorithm 1, implementing Steps 3(b) and 3(c) requires the availability of specific formulas for computing the nonlinear operators $\operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}$ and $\operatorname{prox}_{\frac{1}{\beta}\varphi}$. In fact, for $\alpha > 0$ and $u \in \mathbb{R}^{n^2}$, if $v := \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(u)$, then the components of this vector are given for $i \in N_{n^2}$ as

$$v_i = \max\{|u_i| - 1/\alpha, 0\} \cdot \operatorname{sign}(u_i). \tag{4.3}$$

For a given vector $z \in \mathbb{R}^{2n^2}$ and a positive number β, we write $y := \operatorname{prox}_{\frac{1}{\beta}\varphi}(z)$. If φ is the ℓ^1-norm on $\mathbb{R}^{2n^2}$, then y_i the ith component of y is computed according to (4.3) with u_i being replaced by z_i and α by β. If φ is defined by (4.2), then each vector $[y_i, y_{n^2+i}]^\top, i \in \mathbb{N}_{n^2}$, formed from the components of y, is the proximity operator of the ℓ^2-norm on the $\mathbb{R}^2$ evaluated at $[z_i, z_{n^2+i}]^\top$. That is, we have that $[y_i, y_{n^2+i}]^\top = \operatorname{prox}_{\frac{1}{\beta}\|\cdot\|_2}\left([z_i, z_{n^2+i}]^\top\right)$ with

$$\operatorname{prox}_{\frac{1}{\beta}\|\cdot\|_2}\left(\left[z_i, z_{n^2+i}\right]^\top\right) = \max\left\{\left\|\left[z_i, z_{n^2+i}\right]^\top\right\|_2 - \frac{1}{\beta}, 0\right\} \frac{\left[z_i, z_{n^2+i}\right]^\top}{\left\|\left[z_i, z_{n^2+i}\right]^\top\right\|_2}. \tag{4.4}$$

Hence, Steps 3(b) and 3(c) can be efficiently computed.

In Algorithm 1, the iterate u^{k+1} can exactly reconstruct the given image x at the pixels where the absolute values of $u^k - x - \frac{\beta}{\lambda\alpha} B^\top b^k$ are less or equal to $1/\alpha$. This property is particularly desirable for denoising x having impulsive noise.

The selection of parameters α and β are crucial in Algorithm 1 and other algorithms to be described later in this section. We now explain the strategy of determining these parameters in Step 3(a) of Algorithm 1 via a numerical example of impulsive noise removal. We consider the image of "Cameraman" having 30% of its pixels corrupted by salt-pepper noise. In the first experiment, we choose $(\alpha, \beta) = (0.1, 0.1)$ and $(\alpha_{\max}, \beta_{\max}) = (0.1, 0.1)$, that is, the pair (α, β) will be kept unchanged during iterations. The PSNR-value of the restored

image at each iteration is plotted (dotted line) in Fig. 1a. In the second experiment, we choose $(\alpha, \beta) = (1, 1)$ and $(\alpha_{\max}, \beta_{\max}) = (1, 1)$ which is larger than the one used in the first experiment. Again, the pair (α, β) will be kept unchanged during iterations. The dash dotted line in Fig. 1a depicts the corresponding PSNR-value of the restored image at each iteration. It can be seen that the PSNR-values of the restored image with the pair $(\alpha, \beta) = (0.1, 0.1)$ approach to a stable value much quicker than that with the pair $(\alpha, \beta) = (1, 1)$. However, the stable value in the first case is smaller than that in the second case. These observation can be intuitively explained from Step 3(b) of Algorithm 1. If α is small, Step 3(b) of Algorithm 1 tends to stop updating the noisy image with new information. As a result, useful information contained in the neighbors of noisy pixels cannot permeate into the corresponding noisy pixels to recover them. On the other hand, if α becomes large, it tends to gradually restore noisy pixels, but at a slow rate of convergence. In the third experiment, we choose $(\alpha_{\max}, \beta_{\max}) = (4, 4)$, $(\alpha, \beta) = (1/128, 1/128)$, and $\mu = 10$. The solid line in Fig. 1a displays the corresponding PSNR-value of the restored image at each iteration. The selection of the parameters in the third experiment makes Algorithm 1 working well in assimilating the strengths of the pairs (α, β) with large and small values. Under the same setting of parameters, a quite similar phenomenon (see Fig. 1b) has also been observed for the "Lena" image having 30% of its pixels corrupted by salt-pepper noise. In summary, it is a reasonable strategy to double parameters α and β every fixed number of iterations until they exceed the pre-given numbers $\alpha_{\max}$ and $\beta_{\max}$. Also, the parameter μ in the algorithm is to indicate that the underlying algorithm with a pair (α, β) will iterate μ times before the algorithm with the pair $(2\alpha, 2\beta)$ runs another μ times. According to our numerical experiments, μ ranging from 10 to 25 usually produces reasonable results.

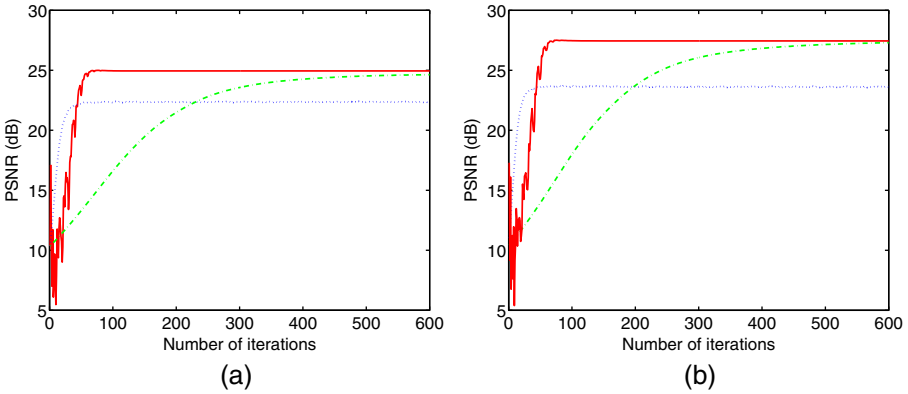

Fig. 1 Convergence rate of Algorithm 1 with different strategies of selecting parameters α and β in Step 3(a) for **a** the image of "Cameraman" and **b** the image of "Lena". The *dotted lines* are with $\alpha_{\max} = \beta_{\max} = 0.1$ and $\alpha = \beta = 0.1$; the *dash dotted lines* are with $\alpha_{\max} = \beta_{\max} = 1$ and $\alpha = \beta = 1$; the *solid lines* are $\alpha_{\max} = \beta_{\max} = 4$, $\alpha = \beta = 1/128$, and $\mu = 10$

In the above L1/TV image denoising model, if we replace the convex function φ by its envelope, then it becomes a special case of the L1/env $\circ$ B model. We call such a model the modified L1/TV image denoising model and apply the result in Section 3 on the L1/env $\circ$ B model (3.3) with $\mathcal{C} = \mathbb{R}^{n^2}$ to the modified L1/TV image denoising model to obtain the following result.

Proposition 5 *Let $x \in \mathbb{R}^{n^2}$ be given, λ and γ be positive numbers, B be the matrix given by (4.1), and φ be either $\|\cdot\|_1$ or the convex function defined by (4.2). If there exists $\alpha > 0$ such that $u \in \mathbb{R}^{n^2}$ is a solution of the following fixed-point equation $u = Qu$ where*

$$Qu := x + \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(u - \frac{\gamma}{\lambda\alpha} B^\top \left(I - \operatorname{prox}_{\frac{1}{\gamma}\varphi}\right)(Bu)\right), \tag{4.5}$$

then u is a solution of the modified L1/TV image denoising model. Furthermore, if α satisfies

$$\alpha \geq \frac{4\gamma}{\lambda} \sin^2 \frac{(n-1)\pi}{2n}, \tag{4.6}$$

then for any initial vector $u^0 \in \mathbb{R}^{n^2}$ the sequence $\{Q^k u^0 : k \in \mathbb{N}\}$ converges to a solution of the model.

Proof The first part of this proposition follows directly from Proposition 4 with (3.5) and $\mathcal{C} = \mathbb{R}^{n^2}$. For the proof of the second part, we recall that

$$\|B\|_2^2 = 8 \sin^2 \frac{(n-1)\pi}{2n}$$

(see [28]). This fact, together with hypothesis (4.6), ensures that the assumption of Proposition 3 is satisfied. Hence, by Proposition 3 the sequence $\{Q^k u^0 : k \in \mathbb{N}\}$ converges to a fixed point of (4.5) for any initial vector $u^0 \in \mathbb{R}^{n^2}$. □

Proposition 5 leads to the following algorithm, with guaranteed convergence, if α satisfies the inequality (4.6).

Algorithm 2

1. Given: noisy image $x \in \mathbb{R}^{n^2}$; $\lambda > 0$, $\gamma > 0$, $\gamma_{\max}$, $\mu \in \mathbb{N}$, and $\alpha = \frac{4\gamma}{\lambda}$
2. Initialization: $u^1 = x$, and $b^1 = 0$
3. For $k \in \mathbb{N}$,
   (a) if $\gamma < \gamma_{\max}$ and k is a multiple of μ, update $\alpha \longleftarrow 2\alpha$ and $\gamma \longleftarrow 2\gamma$; if $\gamma \geq \gamma_{\max}$, then set $\alpha = \frac{4\gamma_{\max}}{\lambda}$ and $\gamma = \gamma_{\max}$;
   (b) $u^{k+1} \longleftarrow x + \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(u^k - x - \frac{\gamma}{\lambda\alpha} B^\top \left(\mathrm{I} - \operatorname{prox}_{\frac{1}{\gamma}\varphi}\right)(Bu^k)\right)$
   (c) Stop, if u^{k+1} converges or satisfies a stopping criteria; otherwise, set $k \leftarrow k+1$.

The parameter γ appeared in Algorithm 2 is an essential parameter in the L1/env $\circ$ B model (3.3). The output from Algorithm 2 can be understood as a solution of the L1/env $\circ$ B model (3.3) with γ being $\gamma_{\max}$, that is,

$$\min\left\{\lambda\|u-x\|_1+\mathrm{env}_{\frac{\varphi}{\gamma_{\max}}}(Bu):u\in\mathbb{R}^{n^2}\right\}.$$

Basically, what Algorithm 2 does in the situation of $\gamma<\gamma_{\max}$ is to find a proper initial vector for the iterative procedure Step 3(b) in the situation of $\gamma=\gamma_{\max}$.

In the rest of this section, we consider the situation $\mathcal{C}\subsetneq\mathbb{R}^{n^2}$ which corresponds to the L1/$\varphi\circ$B image denoising model with prior known information or inpainting. For this case, we have the following algorithm.

Algorithm 3

1. Given: noisy image $x\in\mathbb{R}^{n^2}$; $\lambda>0$, $\alpha>0$, $\beta>0$, $\alpha_{\max}$, $\beta_{\max}$ and $\mu\in\mathbb{N}$
2. Initialization: $u^1=x$, and $b^1=0$
3. For $k\in\mathbb{N}$,
   (a) if $\alpha<\alpha_{\max}$ and $\beta<\beta_{\max}$, and k is a multiple of μ, update $\alpha\longleftarrow 2\alpha$ and $\beta\longleftarrow 2\beta$; if $\alpha\geq\alpha_{\max}$ or $\beta\geq\beta_{\max}$, then set $\alpha=\alpha_{\max}$ and $\beta=\beta_{\max}$;
   (b) $u^{k+1}\longleftarrow x+(I-P_\Lambda)\,\mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(u^k-x-\frac{\beta}{\lambda\alpha}B^\top b^k\right)$
   (c) $b^{k+1}\longleftarrow\left(I-\mathrm{prox}_{\frac{1}{\beta}\varphi}\right)\left(Bu^{k+1}+b^k\right)$
   (d) Stop, if u^{k+1} converges or satisfies a stopping criteria; otherwise, set $k\leftarrow k+1$.

In the L1/TV image denoising model with prior known information or inpainting, if we replace the convex function φ by its envelope, we call the model the modified L1/TV image denoising model with prior information and similar to Proposition 5 have the following result.

Proposition 6 *Let $x\in\mathbb{R}^{n^2}$ be given, λ and γ be positive numbers, B be the matrix given by (4.1), and φ be either $\|\cdot\|_1$ or the convex function defined by (4.2). If there exists $\alpha>0$ such that $u\in\mathbb{R}^{n^2}$ is a solution of the fixed-point (3.5), then u is a solution of the the modified L1/TV image denoising model with prior known information. Furthermore, if α satisfies (4.6) then for any initial vector $u^0\in\mathbb{R}^{n^2}$ the sequence $\{Q^ku^0:k\in\mathbb{N}\}$ converges to a solution of the model.*

This proposition leads to our Algorithm 4, with guaranteed convergence, if α satisfies the inequality (4.6).

Algorithm 4

1. Given: noisy image $x \in \mathbb{R}^{n^2}$; $\lambda > 0$, $\gamma > 0$, $\gamma_{\max}$, $\mu \in \mathbb{N}$, and $\alpha = \frac{4\gamma}{\lambda}$
2. Initialization: $u^1 = x$, and $b^1 = 0$
3. For $k \in \mathbb{N}$,
   (a) if $\gamma < \gamma_{\max}$ and k is a multiple of μ, update $\alpha \longleftarrow 2\alpha$ and $\gamma \longleftarrow 2\gamma$; if $\gamma \geq \gamma_{\max}$, then set $\alpha = \frac{4\gamma_{\max}}{\lambda}$ and $\gamma = \gamma_{\max}$;
   (b) $u^{k+1} \longleftarrow x + (I - P_\Lambda)\operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(u^k - x - \frac{\gamma}{\lambda\alpha}B^\top\left(\mathrm{I} - \operatorname{prox}_{\frac{1}{\gamma}\varphi}\right)(Bu^k)\right)$
   (c) Stop, if u^{k+1} converges or satisfies a stopping criteria; otherwise, set $k \leftarrow k + 1$.

We remark that the output of Algorithm 4 should be understood as a solution of the following L1/TV image denoising model

$$\min\left\{\lambda\|u - x\|_1 + \iota_{\mathcal{C}}(u) + \operatorname{env}_{\frac{\varphi}{\gamma_{\max}}}(Bu) : u \in \mathbb{R}^{n^2}\right\}.$$

5 Numerical results

We present in this section numerical results to demonstrate approximation accuracy and computational efficiency of the proposed algorithms. We shall compare results of the proposed algorithms to those of two the-state-of-the-art methods in the literature. All the experiments are performed under Windows 7 and MATLAB 7.6 (R2008a) running on a PC equipped with an Intel Core 2 Quad CPU at 2.66 GHz and 4G RAM memory.

In our numerical experiments, we consider the problem of restoring images from those corrupted by salt-pepper noise [6]. The testing images shown in Fig. 2 are the "Cameraman" and "Lena" with size of 256×256. The quality of the restored images is evaluated in terms of the peak-signal-to-noise ratio (PSNR) defined by

$$\mathrm{PSNR} := 10\log_{10}\frac{n^2 255^2}{\|u - \widetilde{u}\|_2^2}(\mathrm{dB}),$$

where $\widetilde{u}$ denotes the restored image with respect to the original image u. Each PSNR-value reported in all tables in this section is the average over five runs.

In the first experiment, we compare the performance of Algorithms 1 and 2 developed in the previous section with that of Algorithm PDTV proposed in [19] and Algorithm FTVd in [36]. Both Algorithm PDTV and Algorithm FTVd were proposed recently for the simultaneous deblurring and denoising of images subject to impulse noise. We choose them for comparison because they were both for solving the L1/TV model, and can be used for impulse noise removal. The source code of Algorithm PDTV was obtained from the authors of [19]. The parameters λ and γ used in Algorithm PDTV are

Fig. 2 Original images. **a** "Cameraman"; and **b** "Lena"

chosen to be $\lambda = 0.001$ and $\gamma = 0.01$. The stopping criterion is set according to the detailed description in [19]. The implementation and the website of the corresponding source code package "FTVdv4.1" of Algorithm FTVd can be found in [36]. The parameters β_1 and β_2 in Algorithm FTVd are set to be 5 and 20, respectively. We further set the blurring kernel in Algorithm FTVd to be the identity matrix for the denoising purpose. For Algorithm 1, we set $\alpha_{\max} = \beta_{\max} = 4$ and the initial parameters $\alpha = \beta = \frac{1}{128}$. For Algorithm 2, we set set $\alpha_{\max} = 16$ and the initial value $\alpha = \frac{1}{64}$ and in the meanwhile keep $\gamma = \frac{1}{4}\alpha\lambda$ in every iteration. For both of these algorithms, we double the values of α and γ every ten iterations. The stopping criterion of Algorithm FTVd and

Table 1 The summary of the restoration results of Algorithm PDTV, Algorithm FTVd, and Algorithms 1 and 2

Level	Method	Cameraman		Lena	
		PSNR (dB)	Time (s)	PSNR (dB)	Time (s)
10%	PDTV	28.80	17.03	31.42	18.08
	FTVd	28.68	3.49	31.03	3.31
	Algorithm 1	28.81	2.17	31.46	2.36
	Algorithm 2	28.86	1.54	31.47	1.77
20%	PDTV	26.28	16.85	29.11	17.12
	FTVd	26.15	3.46	28.76	3.33
	Algorithm 1	26.35	2.43	29.11	2.43
	Algorithm 2	26.36	2.27	29.22	2.14
30%	PDTV	24.83	15.66	27.55	16.46
	FTVd	24.71	3.80	27.25	3.66
	Algorithm 1	24.92	2.74	27.57	2.75
	Algorithm 2	24.99	2.41	27.67	2.38
40%	PDTV	23.64	17.35	26.06	15.55
	FTVd	23.59	4.25	25.79	3.99
	Algorithm 1	23.77	2.75	26.00	2.81
	Algorithm 2	23.85	2.44	26.22	2.40
50%	PDTV	22.57	16.45	24.77	17.40
	FTVd	22.39	4.66	24.52	4.52
	Algorithm 1	22.49	2.85	24.30	2.84
	Algorithm 2	22.66	2.62	24.80	2.32

Fig. 3 Results of the four methods when restoring noisy "Cameraman" and "Lena" images corrupted by salt-pepper noise with the noise level $r = 30\%$. (Row 1) The noisy images. (Row 2) The result of FTVd-algorithm. (Row 3) The result of PDTV-algorithm. (Row 4) The result of Algorithm 1. (Row 5) The result of Algorithm 2

Algorithms 1 and 2 is that the relative error between the successive iterates of the restored images should satisfy the following inequality

$$\frac{\left\|u^{i+1}-u^{i}\right\|_2^2}{\left\|u^{i}\right\|_2^2} < 0.001 \tag{5.1}$$

where u^i is the denoised image at the i-th iteration. For each of the four algorithms, the regularization parameter is determined experimentally for the restored image to achieve the best possible PSNR-value.

The numerical results are listed in Table 1. We observe that the PSNR-values of the restored images by Algorithm 1 are higher than those by Algorithm FTVd, and equally match those by Algorithm PDTV. Specifically, when the noise levels are lower than 30%, Algorithm 1 constantly produces the best results in terms of PSNR values. The PSNR-values of the restored images by Algorithm 2 are always higher than those by the other three algorithms. Table 1 also contains the CPU-time consumed by these four algorithms. Clearly, the proposed Algorithms 1 and 2 use much less CPU-time than Algorithm PDTV and Algorithm FTVd. The fast convergence of our algorithms benefits mainly from the low cost of each iteration which consists only of operations of shrinkage and finite differences. In contrast, Algorithm FTVd requires computing FFTs and Gaussian elimination besides operations of shrinkage and finite differences. Moreover, Algorithm 2 is more efficient than Algorithm 1 in terms of the PSNR-values and the used CPU-time. We remark that no prior knowledge of potential uncorrupted pixels are utilized in this experiment.

The restored images obtained from the four algorithms for the noise level at 30% are displayed in Fig. 3. As it can be seen, the denoised images by these four methods have similar visual quality as they are all based on the L1/TV model which ultimately determines the quality of the restoration.

In the second experiment, we test the performance of Algorithms 3 and 4 for salt-pepper noise removal. For these algorithms, the adaptive median filter (AMF) [23] is used for detecting and labeling the salt-pepper noise, therefore,

Table 2 The summary of the restoration results of Algorithms 3 and 4

Level	Method	Cameraman		Lena	
		PSNR (dB)	Time (s)	PSNR (dB)	Time (s)
10%	Algorithm 3	36.93	2.17	40.03	2.14
	Algorithm 4	36.94	1.42	40.04	1.35
20%	Algorithm 3	33.29	2.37	36.58	2.27
	Algorithm 4	33.30	1.64	36.60	1.49
30%	Algorithm 3	30.88	2.37	34.05	2.45
	Algorithm 4	30.90	1.59	34.06	1.65
40%	Algorithm 3	29.21	2.42	32.40	2.44
	Algorithm 4	29.22	1.66	32.40	1.60
50%	Algorithm 3	27.69	2.38	30.57	2.52
	Algorithm 4	27.70	1.79	30.55	1.62

Fig. 4 Results of Algorithm 1 (Row 2), Algorithm 2 (Row 3), Algorithm 4 (Row 4) and Algorithm 4 (Row 5) when restoring noisy images (Row 1) are corrupted by salt-pepper noise with the noise level $r = 50\%$ for the images of "Cameraman" and "Lena"

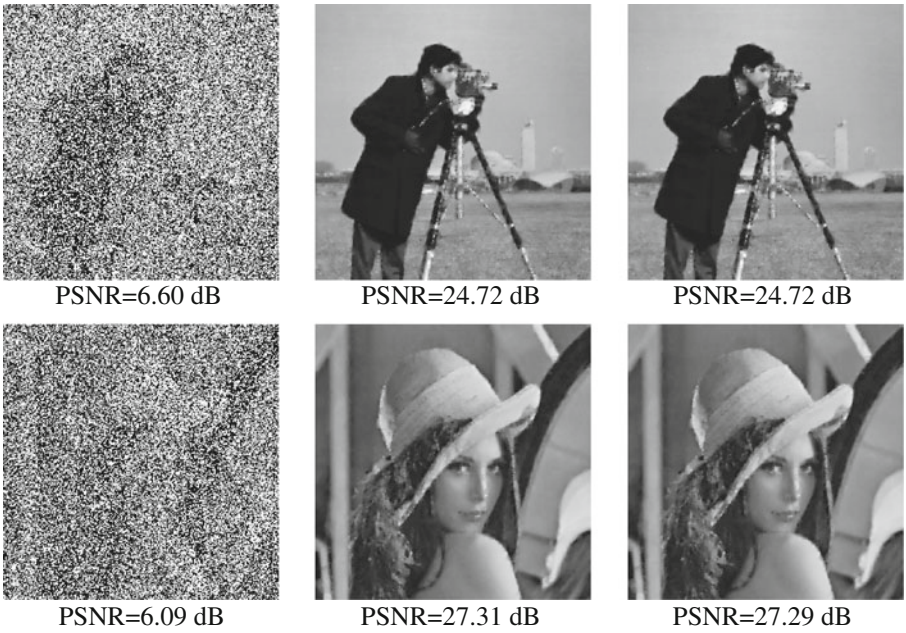

Fig. 5 Restoration results of corrupted "Cameraman" and "Lena" images (*left column*) with noise level $r = 70\%$ by Algorithm 3 (*middle column*) and Algorithm 4 (*right column*)

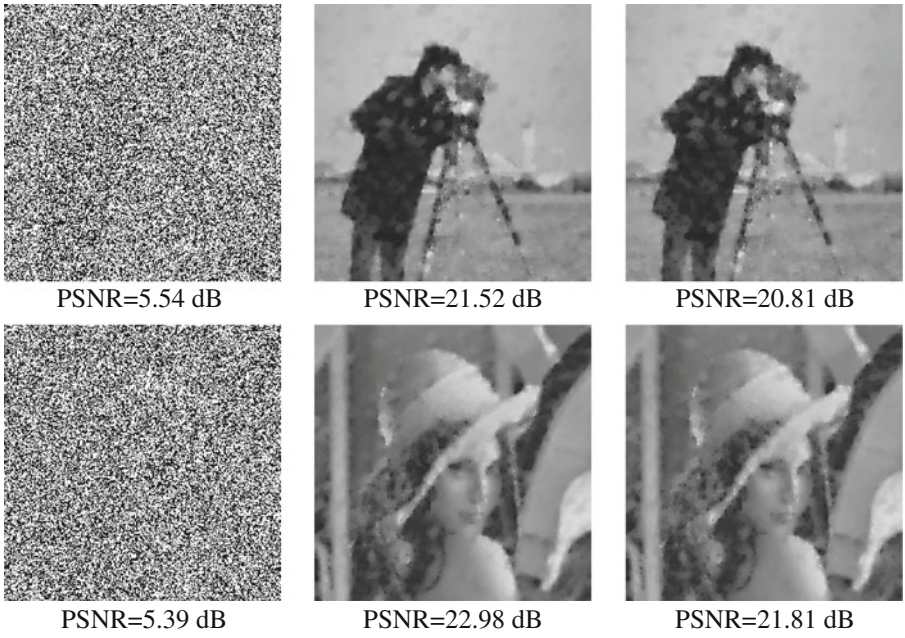

Fig. 6 Restoration results of corrupted "Cameraman" and "Lena" images (*left column*) with noise level $r = 90\%$ by Algorithm 3 (*middle column*) and Algorithm 4 (*right column*)

generating the convex set $\mathcal{C}$. Since the pixels which are detected as uncorrupted by impulse noise will be placed back at each iteration, the starting values of α and β for Algorithm 3 are set to 1, which are larger than those in Algorithms 1 and 2. We choose $\alpha_{\max} = \beta_{\max} = 128$ in Algorithm 3. For Algorithm 4, we set $\alpha_{\max} = 128$ and $\alpha = 8$ and keep $\gamma = \frac{1}{4}\alpha\lambda$. The parameter μ is set to 10 in both Algorithms 3 and 4. Table 2 summarizes the restoration results of Algorithms 3 and 4. Algorithms 3 and 4 produce significantly higher PSNR-values than Algorithms 1 and 2 even when noise levels are high. As it can be seen from Fig. 4, the visual quality of the restored images can also be greatly improved when using Algorithms 3 and 4. This is mainly based on the accurate noise detection by the AMF and the good noise recovery capacity of the L1/TV models (2.1) and (3.3). On the other hand, the L1/$\varphi \circ$ B and L1/env $\circ$ B models (2.1) and (3.3) with $\mathcal{C} = \mathbb{R}^{n^2}$ include all corrupted pixels, which can greatly affect the restoration process especially when the noise level is high. As a result, both of the noisy pixels and uncorrupted pixels are tended to be modified in the restored images and hence the fine image details will be destroyed. This can be seen from the the difference among the restored images shown in Fig. 4.

Finally, we demonstrate that Algorithms 3 and 4 can efficiently remove salt-pepper noise with high noise levels. The visual quality and the PSNR-values of the restored images for the "Cameraman" and "Lena" images corrupted by salt-pepper noise at noise levels of 70% and 90% are depicted in Figs. 5 and 6, respectively. These clearly show that Algorithms 3 and 4 are indeed very efficient.

6 Concluding remarks

We establish proximity operator frameworks for the numerical treatment of the L1/$\varphi \circ$ B and L1/env $\circ$ B models and apply them to the image denoising models for their numerical solutions. Solutions of these models are characterized via fixed-point equations in terms of the proximity operators of the ℓ^1-norm or ℓ^2-norm that have explicit expressions. This naturally leads to simple, efficient fixed-point algorithms for finding approximate solutions of the L1/$\varphi \circ$ B and L1/env $\circ$ B image denoising models. We prove convergence of the algorithms for the L1/env $\circ$ B model. Numerical results presented in this paper confirm that the proposed algorithms perform favorably and use much less CPU-time than the two state-of-the-arts algorithms in the literature.

Acknowledgements This research is supported in part by the US National Science Foundation under grants DMS-0712827 and DMS-1115523, by Guangdong Provincial Government of China through the "Computational Science Innovative Research Team" program, and by Guangdong Province Key Lab of Computational Science. The first and the third authors are partially supported by US Air Force Office of Scientific Research under grant FA9550-09-1-0511. The third author is also supported in part by the Natural Science Foundation of China under grant 11071286. The authors are indebted to Drs. Wotao Yin and Junfeng Yang for making their FTVd code available online, and to Dr. Yiqiu Dong for sharing her PDTV code with them.

References

1. Alliney, S.: A property of the minimum vectors of a regularizing functional defined by means of the absolute norm. IEEE Trans. Signal Process. **45**, 913–917 (1997)
2. Aujol, J.E., Gilboa, G., Chan, T., Osher, S.: Structure-texture image decomposition-modeling, algorithms, and parameter selection. Int. J. Comput. Vis. **67**, 111–136 (2006)
3. Bauschke, H., Borwein, J.: On projection algorithms for solving convex feasibility problems. SIAM Rev. **38**, 367–426 (1996)
4. Bertsekas, D.: Nonlinear Programming. Athena Scientific, Belmont, MA (2003)
5. Bloomfield, P., Steiger, W.: Least Absolute Deviations. Birkhauser, Boston-Basel (1983)
6. Bovik, A.: Handbook of Image and Video Processing. Academic, San Diego (2000)
7. Byrne, C.: A unified treatment of some iterative algorithms in signal processing and image reconstruction. Inverse Probl. **20**, 103–120 (2004)
8. Cai, J.F., Chan, R., Shen, L., Shen, Z.: Simultaneously inpainting in image and transformed domains. Numer. Math. **112**, 509–533 (2009)
9. Candes, E., Romberg, J., Tao, T.: Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Trans. Inf. Theory **52**, 489–509 (2006)
10. Candes, E., Tao, T.: Near optimal signal recovery from random projections: universal encoding strategies? IEEE Trans. Inf. Theory **52**, 5406–5425 (2006)
11. Chambolle, A., Lions, P.L.: Image recovery via total variational minimization and related problems. Numer. Math. **76**, 167–188 (1997)
12. Chan, R., Riemenschneider, S.D., Shen, L., Shen, Z.: Tight frame: the efficient way for high-resolution image reconstruction. Appl. Comput. Harmon. Anal. **17**, 91–115 (2004)
13. Chan, T., Esedoglu, S.: Aspects of total variation regularized l^1 function approximation. SIAM J. Appl. Math. **65**, 1817–1837 (2005)
14. Chan, T., Marquina, A., Mulet, P.: High-order total variation-based image restoration. SIAM J. Sci. Comput. **22**, 503–516 (2000)
15. Chen, T., Yin, W., Zhou, X.S., Comaniciu, D., Huang, T.: Total variation models for variable lighting face recognition. IEEE Trans. Pattern Anal. Mach. Intell. **28**, 1519–1524 (2006)
16. Clason, C., Jin, B., Kunisch, K.: A duality-based splitting method for L1-TV image restoration with automatic regularization parameter choice. SIAM J. Sci. Comput. **32**, 1484–1505 (2010)
17. Combettes, P., Wajs, V.: Signal recovery by proximal forward-backward splitting. Multiscale Model. Simul.: SIAM Interdiscip. J. **4**, 1168–1200 (2005)
18. Daubechies, I., Defrise, M., Mol, C.D.: An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Commun. Pure Appl. Math. **57**, 1413–1541 (2004)
19. Dong, Y., Hintermuller, M., Neri, M.: A primal-dual method for l^1-TV image denoising. SIAM J. Math. Anal. **2**, 1168–1189 (2009)
20. Fornasier, M., Rauhut, R.: Recovery algorithms for vector-valued data with joint sparsity constraints. SIAM J. Numer. Anal. **46**, 577–613 (2008)
21. Fu, H., Ng, M.K., Nikolova, M., Barlow, J.L.: Efficient minimization of mixed $\ell^2 - \ell^1$ and $\ell^1 - \ell^1$ nomrs for image restoration. SIAM J. Sci. Comput. **27**, 1881–1902 (2006)
22. Goldfard, D., Yin, W.: Second-order cone programming methods for total variation based image restoration. SIAM J. Sci. Comput. **27**, 622–645 (2005)
23. Gonzalez, R., Woods, R.: Digital Image Processing. Addison-Wesley, Boston, MA (1993)
24. Guo, X., Li, F., Ng, M.: A fast ℓ^1-TV algorithm for image restoration. SIAM J. Sci. Comput. **31**, 2322–2341 (2009)
25. Hiriart-Urruty, J.B., Lemarechal, C.: Convex analysis and minimization algorithms II. In: Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences], vol. 306. Springer, Berlin (1993)
26. Lustig, M., Donoho, D., Pauly, J.M.: Sparse MRI: the application of compressed sensing for rapid MR imaging. Magn. Reson. Med. **58**, 1182–1195 (2007)
27. Lysaker, M., Tai, X.: Iterative image restoration combining total variation minimization and a second-order functional. Int. J. Comput. Vis. **66**, 5–18 (2006)
28. Micchelli, C.A., Shen, L., Xu, Y.: Proximity algorithms for image models: denosing. Inverse Probl. **27**, 045009, 30 pp. (2011)
29. Moreau, J.-J.: Proximité et dualité dans un espace hilbertien. Bull. Soc. Math. Fr. **93**, 273–299 (1965)

30. Nikolova, M.: A variational approach to remove outliers and impulse noise. J. Math. Imaging Vis. **20**, 99–120 (2004)
31. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D **60**, 259–268 (1992)
32. Scherzer, O.: Denoising with higher-order derivatives of bounded variation and an application to parameter estimation. Computing **60**, 5–27 (1998)
33. Stefan, W., Renaut, R.A., Gelb, A.: Improved total variation-type regularization using higher order edge detectors. SIAM J. Imaging Sci. **3**, 232–251 (2010)
34. Tropp, J.: Greed is good: algorithmic results for sparse approximation. IEEE Trans. Inf. Theory **50**, 2231–2242 (2004)
35. Wu, C., Zhang, J., Tai, X.-C.: Augmented lagrangian method for total variation restoration with non-quadratic fidelity. Inverse Probl. Imaging **5**, 237–261 (2011)
36. Yang, J., Zhang, Y., Yin, W.: An efficient TVL1 algorithm for deblurring multichannel images corrupted by impulsive noise. SIAM J. Sci. Comput. **31**, 2842–2865 (2009)
37. Yin, W., Chen, T., Zhou, X.S., Chakraborty, A.: Background correction for cDNA microarray image using the TV+L1 model. Bioinformatics **10**, 2410–2416 (2005)
38. Yin, W., Goldfard, D., Osher, S.: The total variation regularized l^1 model for multiscale decomposition. Multiscale Model. Simul.: SIAM Interdiscip. J. **6**, 190–211 (2007)
39. You, Y., Kaveh, M.: Fourth-order partial differential equations for noise removal. IEEE Trans. Image Process. **9**, 1723–1730 (2000)
40. Zach, C., Pock, T., Bischof, H.: A duality based approach for realtime TV-l^1 optical flow. In: Lecture Notes in Computer Sciences, vol. 4713, pp. 214–223. Springer, Berlin Heidelberg (2007)