# Proximity Algorithms for Image Models II: L1/TV Denoising

Charles A. Micchelli* Lixin Shen[†] Yuesheng Xu [†§‡] Xueying Zeng[§]

**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 $\ell^1$-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 the convex function with the $\ell^1$-norm or the $\ell^2$-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 a fixed point of a mapping expressed in terms of the proximity operator of the $\ell^1$-norm or the $\ell^2$-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 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. We establish convergence results of the proposed fixed-point algorithms by showing that the mappings which define the fixed-point iterations are nonexpansive. 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.

## 1 Introduction

Total-variation based variational models are widely used in image denoising. The well-known Rudin-Osher-Fatemi (ROF) image denoising model [38] seeks a minimizer of the sum of a data fidelity term measured in the square of $\ell^2$-norm and the total-variation regularization term. This minimization problem is often referred to as the L2/TV model. The $\ell^2$-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 $\ell^2$-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 $\ell^1$-norm and the total-variation regularization term. The $\ell^1$-norm can effectively suppress the effect of outliers

*Department of Mathematics and Statistics, SUNY at Albany, Albany, NY, and Department of Mathematics, City University of Hong Kong, Kowloon, Hong Kong.

[†]Department of Mathematics, Syracuse University, Syracuse, NY 13244.

[‡]All correspondence should be sent to this author. E-mail address: yxu06@syr.edu.

[§]School of Mathematics and Computational Sciences, Sun Yat-sen University, Guangzhou 510275, P. R. China.

that may contaminate a given image. It was demonstrated in [2, 37] that the $\ell^1$-norm fidelity term is particularly suitable for handling non-Gaussian additive noise such as impulsive noise and Laplace noise [6]. 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 [17, 37, 44]. Applications of the L1/TV model include computer vision [19], biomedical imaging [43], optical flow and object tracking [46], and shape denoising [46]. 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 $\ell^1$-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 [17] 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 $\ell^1$ 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 [25], 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 [27], a method based on second order cone programming was proposed. In recent work on the numerical treatment of the the L1/TV model (see for example, [3, 20, 29, 42]), a commonly used approach is 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 [23] to avoid treating directly the non-differentiable functional in the L1/TV model, the associated dual problem was treated by adding two regularization terms to make the dual problem itself semi-smooth, and the resulting primal-dual system was then solved by a semi-smooth Newton solver. Therefore, it appears that all existing work so far does not solve the L1/TV model directly, but rather solves various modified versions of it.

The main purpose of this paper is to introduce a proximity operator framework for the *direct* treatment of the L1/TV model. In [35], by identifying the total-variation as a composition of the convex function which defines the $\ell^1$-norm or the $\ell^2$-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 $\ell^1$-norm or the $\ell^2$-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 $\ell^1$ 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 $\ell^1$ fidelity term plus the 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 the proximity operator of the convex function and two equations, one reflecting the composition with the matrix and the other addressing the non-differentiability of the $\ell^1$ 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 an $\ell^1$ 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, and the convergence of the resulting fixed-point algorithm is proved by employing the well-known fixed-point theory which states that the map that defines the fixed-point iteration is nonexpansive. 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 the state-of-the-art algorithms developed in [23, 42]. 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 denote by the L1/$\varphi \circ$ B model. This model minimizes the sum of the $\ell^1$ 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. This section naturally divides into two parts. The first part pertains to a fixed-point formulation of the solutions of the L1/$\varphi \circ$ B model in terms of the proximity operator. In the second part, we show that the mapping that defines the fixed-point equation in the characterization of solutions of the L1/$\varphi \circ$ B model is averaged nonexpansive under certain differentiability conditions on the convex function.

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 $\ell^1$ and $\ell^2$ vector norm on $\mathbb{R}^d$.

We now describe the general model. Suppose that $\varphi$ 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

$$\iota_{\mathcal{C}}(u) := \begin{cases} 0, & u \in \mathcal{C}; \\ +\infty, & u \notin \mathcal{C}. \end{cases} \tag{2.1}$$

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

$$\min\{J_{\mathrm{L1}/\varphi\circ\mathrm{B}}(u) : u \in \mathbb{R}^d\} \tag{2.2}$$

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 (2.2) 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 (2.2) exists. On the other hand, since the functional $J_{\mathrm{L1}/\varphi\circ\mathrm{B}}$ is not in general strictly convex, solutions of model (2.2) may not be unique. This is an intrinsic feature of model (2.2).

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 $\varphi$ is the norm $\|\cdot\|_1$ while for the isotropic total-variation $\varphi$ 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., [14, 18, 34, 39, 40, 45]), $B$ is 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., [9, 12, 13, 16, 22, 24, 41]). Images are well-represented by sparse expansions with respect to wavelets or framelets. In a wavelet framework, $\varphi$ is the norm $\|\cdot\|_1$ and $B$ usually is a wavelet transform matrix.

- Optimal MRI reconstruction. The regularization functional in the optimal MRI reconstruction [33] is defined in terms of a combination of the total variation and a Besov norm. In this case, $\varphi$ 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

$$\mathcal{C} := \{y : P_\Lambda y = P_\Lambda x\} \tag{2.3}$$

where $P_\Lambda$ is a $d \times d$ diagonal matrix with diagonal entries 1 for indices in $\Lambda$ 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 (2.2) 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 $\psi$ at $x \in \mathbb{R}^d$ with $\psi(x) < +\infty$ is the set defined by*

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

It is well-known (cf. [5, Page 732]) that the subdifferential of a convex function $\psi$ 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 $\psi$ is defined for $x \in \mathbb{R}^d$ by*

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

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

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

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

A simple proof of this fact was given in [35]. As a direct consequence of Lemma 1, we have that

$$y \in \partial\psi(x) \quad \text{if and only if} \quad y = (I - \operatorname{prox}_{\psi})(x + y). \tag{2.4}$$

With the help of Lemma 1, we next characterize a solution of model (2.2). To this end, for a given $x \in \mathbb{R}^d$, a positive number $\lambda$, and a convex set $\mathcal{C}$ in $\mathbb{R}^d$, we define a function $\varrho$ on $\mathbb{R}^d$ at $v \in \mathbb{R}^d$ as

$$\varrho(v) := \lambda\|v - x\|_1 + \iota_{\mathcal{C}}(v). \tag{2.5}$$

**Proposition 1.** *If $\varphi$ is a real-valued convex function on $\mathbb{R}^m$, $B$ an $m \times d$ matrix, and $u \in \mathbb{R}^d$ is a solution of model (2.2), then for any $\alpha, \beta > 0$ there exists a vector $b \in \mathbb{R}^m$ such that*

$$u = \operatorname{prox}_{\frac{1}{\lambda\alpha}\varrho}\left(u - \frac{\beta}{\lambda\alpha}B^\top b\right), \tag{2.6}$$

$$b = \left(I - \operatorname{prox}_{\frac{1}{\beta}\varphi}\right)(Bu + b). \tag{2.7}$$

*Conversely, if there exist $\alpha, \beta > 0$, $b \in \mathbb{R}^m$, and $u \in \mathbb{R}^d$ satisfying equations (2.6) and (2.7), then $u$ is a solution of model (2.2).*

*Proof.* We assume that $u \in \mathbb{R}^d$ is a solution of model (2.2). 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

$$0 \in \partial\varrho(u) + B^\top\partial\varphi(Bu). \tag{2.8}$$

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

$$0 = \lambda\alpha a + \beta B^\top b. \tag{2.9}$$

By Lemma 1, from the inclusion $b \in \frac{1}{\beta}\partial\varphi(Bu)$ we get (2.7) and from the inclusion $a \in \frac{1}{\lambda\alpha}\partial\varrho(u)$ we have that

$$u = \operatorname{prox}_{\frac{1}{\lambda\alpha}\varrho}(u + a). \tag{2.10}$$

Using equation (2.9) we conclude that $a = -\frac{\beta}{\lambda\alpha}B^\top b$ and substitute it into (2.10) we obtain (2.6).

Conversely, suppose that there exist $\alpha, \beta > 0$, $b \in \mathbb{R}^m$, and $u \in \mathbb{R}^d$ satisfying equations (2.6) and (2.7). Again, by Lemma 1, equations (2.7) and (2.6) 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 (2.2). $\square$

The next result gives the proximity operator of the function $\varrho$ defined by equation (2.5).

**Lemma 2.** *If $\lambda, \alpha > 0$, $x \in \mathbb{R}^d$, $\Lambda \subset \mathbb{N}_d$, $\mathcal{C}$ is the set defined by* (2.3) *and $\varrho$ is the function defined by* (2.5)*, then for $v \in \mathbb{R}^d$,*

$$\mathrm{prox}_{\frac{1}{\lambda\alpha}\varrho}(v) = x + (I - P_\Lambda)\mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(v - x). \tag{2.11}$$

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

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

Since the convex set $\mathcal{C}$ has the structure described in (2.3), we get that

$$\mathrm{prox}_{\frac{1}{\lambda\alpha}\varrho}(v) = P_\Lambda x + (I - P_\Lambda) \cdot \arg\min\left\{\frac{1}{2}\|y - v\|_2^2 + \frac{1}{\alpha}\|y - x\|_1 : y \in \mathbb{R}^d\right\}. \tag{2.12}$$

Again, using the definition of the proximity operator, we derive from (2.12) that

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

which establishes equation (2.11). □

Specializing Lemma 2 to the convex set $\mathcal{C}$ defined by (2.3), equation (2.6) becomes

$$u \quad = \quad x + (I - P_\Lambda)\mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(u - x - \frac{\beta}{\lambda\alpha}B^\top b\right). \tag{2.13}$$

As a result, a solution of model (2.2) for the convex set $\mathcal{C}$ defined by (2.3) is characterized in terms of the fixed-point equations (2.13) and (2.7).

Next, we show that if the convex function $\varphi$ is differential, a solution of the corresponding model is characterized in terms of only one fixed-point equation.

**Proposition 2.** *Let $x \in \mathbb{R}^d$ be given, $\lambda$ be a positive number, and $B$ be an $m \times d$ matrix. Suppose that $\mathcal{C}$ is a convex set in $\mathbb{R}^d$, $\varrho$ is the function defined by* (2.5)*, and $\varphi$ is a differential convex function on $\mathbb{R}^m$. If $u \in \mathbb{R}^d$ is a solution of model* (2.2)*, then for any $\alpha > 0$*

$$u \quad = \quad \mathrm{prox}_{\frac{1}{\lambda\alpha}\varrho}\left(u - \frac{1}{\lambda\alpha}B^\top \nabla\varphi(Bu)\right). \tag{2.14}$$

*Conversely, if $u \in \mathbb{R}^d$ satisfies equation* (2.14) *for some $\alpha > 0$, then $u$ is a solution of model* (2.2)*.*

*Proof.* According to Proposition 1, a solution $u$ of model (2.2) satisfies the fixed-point equations (2.6) and (2.7). Notice that the subdifferential of a differentiable function $\varphi$ 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 $\varphi$ at $v$. Hence, by (2.4), equation (2.7) implies that

$$b = \frac{1}{\beta}\nabla\varphi(Bu). \tag{2.15}$$

Substituting equation (2.15) into (2.6) yields the fixed-point equation (2.14). □

An approximate solution of the fixed-point equation (2.14) gives an approximate solution of model (2.2) with a differentiable convex function $\varphi$. In particular, when the convex set $\mathcal{C}$ is defined by (2.3), according to Lemma 2, equation (2.14) becomes

$$u = x + (I - P_\Lambda)\mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(u - x - \frac{1}{\lambda\alpha}B^\top \nabla\varphi(Bu)\right). \tag{2.16}$$

Proposition 2 expresses a solution of model (2.2) as a fixed-point of a nonlinear mapping. We now formulate the nonlinear mapping. For a given $x \in \mathbb{R}^d$ and positive numbers $\lambda$ and $\alpha$, we define an operator $H : \mathbb{R}^d \to \mathbb{R}^d$ at $u \in \mathbb{R}^d$ by the equation

$$Hu := u - \frac{1}{\lambda\alpha}B^\top \nabla\varphi(Bu) \tag{2.17}$$

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

$$Pu := (\mathrm{prox}_{\frac{1}{\lambda\alpha}\varrho} \circ H)u. \tag{2.18}$$

With this notation, (2.14) is rewritten simply as

$$u = Pu. \tag{2.19}$$

A straightforward numerical method to iterate (2.19) is the Picard iteration. Specifically, for a given initial point $x^1 \in \mathbb{R}^d$, the Picard iterates of operator $P$ generate the sequence $\{x^k : k \in \mathbb{N}\}$ where $x^{k+1} := Px^k$, $k \in \mathbb{N}$. That is, $x^{k+1} := P^k x^1$ where $P^k$ denotes the $k$-th power of $P$. We next study the convergence property of the Picard sequence. To this end, we briefly review the notion of nonexpansive mappings which is crucial for convergence analysis of the Picard sequence.

Let $\Omega$ be a closed convex nonempty subset of $\mathbb{R}^d$. According to [4, 8], 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

$$(1 - \kappa)I + \kappa N, \tag{2.20}$$

for some positive number $\kappa$ in $(0, 1)$ and some nonexpansive operator $N$. Moreover, a mapping of the form given in (2.20) is called $\kappa$-averaged and an operator is called firmly nonexpansive if it is $\frac{1}{2}$-averaged. The next two lemmas were proved in [8].

**Lemma 3.** *If $\{Q_i : i \in \mathbb{N}_s\}$ is a family of averaged nonexpansive mappings, then both the composition mapping $Q_s \circ \cdots \circ Q_1$ and $\sum_{i\in\mathbb{N}_s} w_i Q_i$, where $\sum_{i\in\mathbb{N}_s} w_i = 1$ and $w_i \ge 0$ for $i \in \mathbb{N}_s$, are averaged nonexpansive.*

For any operator $Q$ on $\mathbb{R}^d$, the fixed-point set of $Q$ is $\mathrm{Fix}(Q) := \{x : Qx = x\}$. The Krasnoselskii-Mann approach is a powerful way to find fixed-points of nonexpansive operators.

**Lemma 4.** *If $Q$ is an averaged nonexpansive mapping on $\mathbb{R}^d$ and $\mathrm{Fix}(Q)$ is nonempty, then for any initial vector in $\mathbb{R}^d$ the corresponding Picard sequence converges to a member of $\mathrm{Fix}(Q)$.*

The next lemma is due to [26].

**Lemma 5.** *If $\Omega$ is a closed subset of $\mathbb{R}^d$ and $Q : \Omega \to \mathbb{R}^d$ is a mapping, then the following statements are equivalent:*

*(i) The mapping $Q$ is firmly nonexpansive.*

*(ii) The inequality $\|Qx - Qy\|_2^2 \le \langle Qx - Qy, x - y\rangle$ holds for all $x, y \in \Omega$.*

*(iii) The mapping $2Q - I$ is nonexpansive.*

Since $I - Q = \frac{1}{2}I + \frac{1}{2}(I - 2Q)$, by Lemma 5, we know that $I - Q$ is firmly nonexpansive, if $Q$ is firmly nonexpansive. With this fact, we can show that $I - \mathrm{prox}_\psi$ is firmly nonexpansive since $\mathrm{prox}_\psi$ is firmly nonexpansive [21]. This result is stated in the following.

**Lemma 6.** *If $\psi$ is a convex function on $\mathbb{R}^d$, then both $\mathrm{prox}_\psi$ and $I - \mathrm{prox}_\psi$ are firmly nonexpansive.*

We now return to the study of the operator $P$ given by (2.18). We will prove that $P$ is averaged nonexpansive for proper choices of parameters $\lambda$ and $\alpha$.

**Proposition 3.** *Let $x \in \mathbb{R}^d$ be given, $B$ be an $m \times d$ matrix, $\varphi$ be a differentiable convex function on $\mathbb{R}^m$, and $\nabla\varphi$ be Lipchitz continuous, that is, for some $L > 0$,*

$$\|\nabla\varphi(p) - \nabla\varphi(q)\|_2 \le L\|p - q\|_2 \quad \textit{for all} \quad p, q \in \mathbb{R}^m. \tag{2.21}$$

*If $\alpha$ is chosen to satisfy*

$$\frac{L}{\lambda\alpha} \le \frac{1}{\|B\|_2^2}, \tag{2.22}$$

*where $\|B\|_2$ denotes the largest singular value of $B$, then operator $H$ defined by* (2.17) *is firmly nonexpansive.*

*Proof.* We define the operator $F : \mathbb{R}^d \to \mathbb{R}^d$ at $u \in R^d$ by $Fu := \frac{1}{\lambda\alpha}B^\top\nabla\varphi(Bu)$ and observe that for any $u \in \mathbb{R}^d$ that $Hu = (I - F)u$. By referring back to the argument made prior to Lemma 6 to establish the firm nonexpansivity of $H$, we need to prove that $F$ is firmly nonexpansive. For this purpose, we see from [30, 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

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

By Lemma 5 and hypothesis (2.22), the above inequality ensures that $F$ is firmly nonexpansive. □

We are now ready to establish the averaged nonexpansivity of operator $P$ defined by (2.18).

**Proposition 4.** *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 $\varrho$ be a function defined by* (2.5)*. Suppose that $\varphi$ is a differentiable convex function on $\mathbb{R}^m$, and $\nabla\varphi$ is Lipschitz continuous with the Lipschitz constant $L$. If the parameter $\alpha$ is chosen to satisfy condition* (2.22)*, then operator $P$ defined by* (2.18) *is averaged nonexpansive.*

*Proof.* Since the parameter $\alpha$ is chosen to satisfy hypothesis (2.22), by Proposition 3, $H$ is firmly nonexpansive. Moreover, by Lemma 6, $\mathrm{prox}_{\frac{1}{\lambda\alpha}\varrho}$ is firmly nonexpansive. Therefore, they both are averaged nonexpansive. Lemma 3 ensures that $\mathrm{prox}_{\frac{1}{\lambda\alpha}\varrho} \circ H$ is averaged nonexpansive. It follows that $P$ defined by (2.18) is averaged nonexpansive. □

The next result shows that the Picard sequence of the operator (2.18) converges to a solution of model (2.2).

**Proposition 5.** *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 $\varrho$ be a function defined by* (2.5)*. Suppose that $\varphi$ is a differentiable convex function on $\mathbb{R}^m$, and $\nabla\varphi$ is Lipschitz continuous with the Lipschitz constant $L$. If the parameter $\alpha$ is chosen to satisfy condition* (2.22)*, 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* (2.2)*.*

*Proof.* Proposition 2 ensures that a fixed-point of operator $P$ defined by (2.18) gives a solution of model (2.2). It remains to prove the Picard iteration sequence $\{P^k u^0 : k \in \mathbb{N}\}$ converges to a fixed-point of $P$. Since the parameter $\alpha$ satisfies condition (2.22), by Proposition 4 the operator $P$ is averaged nonexpansive. Thus, Lemma 4 ensures that the sequence $\{P^k u^0 : k \in \mathbb{N}\}$ converges to a solution of the fixed-point equation (2.14) for any vector $u^0 \in \mathbb{R}^d$. □

# 3 The L1/env $\circ$ B Model

In the preceding section, we studied the L1/$\varphi \circ$ B model under various hypotheses. We noticed, by imposing an assumption on the differentiability of the convex function $\varphi$, that the coupled system of fixed-point equations (2.6) and (2.7) for a solution of the model reduces to a single fixed-point equation (2.14). 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 the need of preserving edges with computational efficiency, we propose to replace the convex function $\varphi$ by its "smoothed" version which preserves main geometric features of $\varphi$ but, at the same time, is differentiable. For this purpose, we study in this section the L1/env $\circ$ B model which is a variant of the L1/$\varphi \circ$ B model with $\varphi$ being replaced by its Moreau envelope.

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

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

Clearly, the minimum of (3.1) is attained at $\mathrm{prox}_{\gamma\psi} z$. It can be shown that the envelope of a convex function is convex. For example, the envelope of $\psi := \gamma\|\cdot\|_2$ is given for $z \in \mathbb{R}^m$ by

$$\mathrm{env}_{\gamma\|\cdot\|_2}(z) := \begin{cases} \frac{1}{2\gamma}\|z\|_2^2, & \text{if } \|z\|_2 \leq \gamma; \\ \|z\|_2 - \frac{\gamma}{2}, & \text{otherwise.} \end{cases} \tag{3.2}$$

It is worth noting that the well-known Huber function [31] is actually the envelope $\mathrm{env}_{\gamma\|\cdot\|_2}$. Recall for $\gamma > 0$ and $z \in \mathbb{R}^m$ that the proximity operator $\mathrm{prox}_{\gamma\|\cdot\|_2}(z)$ has the expression

$$\mathrm{prox}_{\gamma\|\cdot\|_2}(z) = \max\{\|z\|_2 - \gamma, 0\}\frac{z}{\|z\|_2}. \tag{3.3}$$

It can be readily verified that the envelope and the proximity operator for this specific convex function are connected in the following way

$$\gamma\nabla\mathrm{env}_{\gamma\|\cdot\|_2}(z) = (I - \mathrm{prox}_{\gamma\|\cdot\|_2})(z).$$

In fact, this relation holds for any convex function $\psi$. Indeed, the envelope of a convex function is always differentiable and its gradient is given by

$$\nabla(\mathrm{env}_{\gamma\psi}) = \frac{1}{\gamma}(I - \mathrm{prox}_{\gamma\psi}). \tag{3.4}$$

Moreover, for any positive number $\gamma$ the envelope $\mathrm{env}_{\gamma\psi}$ is bounded above by $\psi$ and it converges to $\psi$ as $\gamma \to 0^+$ (see, e.g., [36]). That is, for $z \in \mathbb{R}^m$ we have that

$$\mathrm{env}_{\gamma\psi}(z) \leq \psi(z) \quad \text{and} \quad \mathrm{env}_{\gamma\psi}(z) \to \psi(z), \quad \text{as } \gamma \to 0^+.$$

These remarkable properties of the envelope motivate us to replace the convex function $\varphi$ in the L1/$\varphi \circ$ B model by its envelope $\mathrm{env}_{\frac{1}{\beta}\varphi}$ for some positive number $\beta$. The resulting model is then given by

$$\min \left\{ \lambda \|u - x\|_1 + \iota_{\mathcal{C}}(u) + \mathrm{env}_{\frac{1}{\beta}\varphi}(Bu) : u \in \mathbb{R}^d \right\}. \tag{3.5}$$

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

We present below the main results of this section.

**Proposition 6.** *Let $x \in \mathbb{R}^d$ be given, $\lambda$ be a positive number, and $B$ be an $m \times d$ matrix. Suppose that $\mathcal{C}$ is a convex set in $\mathbb{R}^d$, $\varrho$ is a function defined by* (2.5)*, and $\varphi$ is a convex function on $\mathbb{R}^m$. If $u$ is a solution of model* (3.5)*, then for any $\alpha > 0$, $u \in \mathbb{R}^d$ is a fixed-point of the mapping $Q$ defined by*

$$Qu := \mathrm{prox}_{\frac{1}{\lambda\alpha}\varrho} \left( u - \frac{\beta}{\lambda\alpha} B^\top (I - \mathrm{prox}_{\frac{1}{\beta}\varphi})(Bu) \right). \tag{3.6}$$

*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* (3.5)*.*

*Proof.* By equation (3.4), we know that $\mathrm{env}_{\frac{1}{\beta}\varphi}$ is differentiable. Using Proposition 2 with $\varphi$ replaced by its envelope and the gradient of $\mathrm{env}_{\frac{1}{\beta}\varphi}$

$$\nabla \mathrm{env}_{\frac{1}{\beta}\varphi}(u) = \beta(I - \mathrm{prox}_{\frac{1}{\beta}\varphi})(u),$$

we conclude that the result of this proposition holds. □

When the convex set $\mathcal{C}$ is given by (2.3), we appeal to Lemma 2 and conclude that the fixed-point equation of mapping $Q$ becomes

$$u = x + (I - P_\Lambda)\mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1} \left( u - x - \frac{\beta}{\lambda\alpha} B^\top (I - \mathrm{prox}_{\frac{1}{\beta}\varphi})(Bu) \right). \tag{3.7}$$

**Proposition 7.** *Let $x \in \mathbb{R}^d$ be given, $\lambda$, $\beta$ be positive numbers, $\varrho$ be a function defined by* (2.5)*, and $\varphi$ be a convex function on $\mathbb{R}^m$. If the parameter $\alpha$ is chosen to satisfy the condition*

$$\frac{\beta}{\lambda\alpha} \leq \frac{1}{\|B\|_2^2}, \tag{3.8}$$

*then $Q$ defined by* (3.6) *is averaged nonexpansive.*

*Proof.* By equation (3.4) and Lemma 6, we have for any $p, q \in \mathbb{R}^d$ that

$$\|\nabla \mathrm{env}_{\frac{1}{\beta}\varphi}(p) - \nabla \mathrm{env}_{\frac{1}{\beta}\varphi}(q)\|_2 = \|\beta(I - \mathrm{prox}_{\frac{1}{\beta}\varphi})(p) - \beta(I - \mathrm{prox}_{\frac{1}{\beta}\varphi})(q)\|_2 \leq \beta\|p - q\|_2.$$

This shows that $\nabla \mathrm{env}_{\frac{1}{\beta}\varphi}$ is Lipschtiz continuous with the Lipschitz constant $\beta$. Hence, by Proposition 4, the mapping $Q$ defined by (3.6) is averaged nonexpansive. □

The following result shows the Picard sequence converges to a solution of model (3.5).

**Proposition 8.** *Let $x \in \mathbb{R}^d$ be given, $\lambda$, $\beta$ be positive numbers, $\varrho$ be a function defined by* (2.5)*, and $\varphi$ be a convex function on $\mathbb{R}^m$. If the parameter $\alpha$ is chosen to satisfy condition* (3.8)*, then for any initial vector $u^0 \in \mathbb{R}^d$ the sequence $\{Q^k u^0 : k \in \mathbb{N}\}$ converges to a solution of model* (3.5)*.*

*Proof.* This is a direct consequence of Lemma 4 and Proposition 7. □

# 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 $(in+j)$-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 $\varphi$ 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 $\varphi$ 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 $\varphi$ according to whether the total-variation is anisotropic or isotropic. For the anisotropic or the isotropic total-variation, $\varphi : \mathbb{R}^{2n^2} \to \mathbb{R}$ is defined at $z \in \mathbb{R}^{2n^2}$, respectively, as

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

Hence, the L1/TV image denoising model is model (2.2) with the convex function $\varphi$ 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 fix-point equations (2.6) and (2.7), with the above special choice of the convex function $\varphi$, the matrix $B$ and $\mathcal{C}$. In fact, in this special case, equation (2.6) reduces to

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

The following algorithm describes specific procedures for finding a solution of the above L1/TV image denoising model according to the fix-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* $\mu$*, 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) $u^{k+1} \longleftarrow x + \mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(u^k - x - \frac{\beta}{\lambda\alpha} B^\top b^k)$

(c) $b^{k+1} \longleftarrow (I - \mathrm{prox}_{\frac{1}{\beta}\varphi})(Bu^{k+1} + b^k)$

(d) *Stop, if* $u^{k+1}$ *converges or satisfies a stopping criteria; otherwise, set* $k \leftarrow k+1$*.*

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

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

For a given vector $z \in \mathbb{R}^{2n^2}$ and a positive number $\beta$, we write $y := \mathrm{prox}_{\frac{1}{\beta}\varphi}(z)$. If $\varphi$ is the $\ell^1$-norm on $\mathbb{R}^{2n^2}$, then $y_i$ the $i$th component of $y$ is computed according to (4.3) with $u_i$ being replaced by $z_i$ and $\alpha$ by $\beta$. If $\varphi$ 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 $\ell^2$-norm on the $\mathbb{R}^2$ evaluated at $[z_i, z_{n^2+i}]^\top$. That is, we have that

$$[y_i, y_{n^2+i}]^\top = \mathrm{prox}_{\frac{1}{\beta}\|\cdot\|_2}\left([z_i, z_{n^2+i}]^\top\right), \tag{4.4}$$

which can be evaluated explicitly by using (3.3). Hence, Step 3(b) and 3(c) can be efficiently computed.

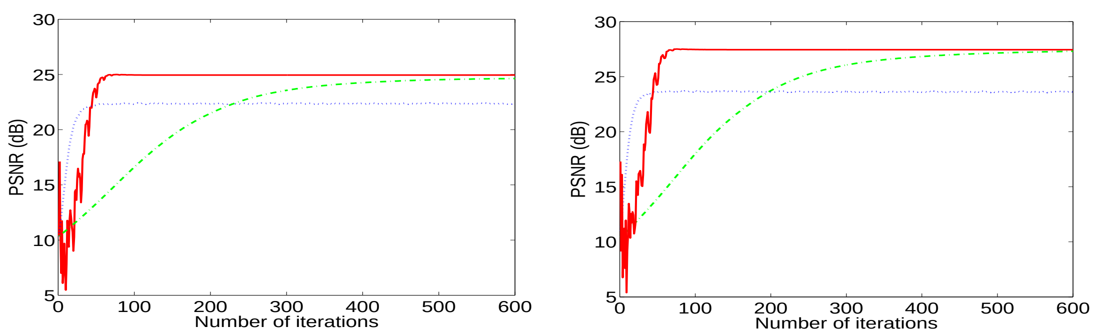



Figure 1: Convergence rate of Algorithm 1 with different strategies of selecting parameters $\alpha$ and $\beta$ in Step 3(a) for (a) the image of "Cameraman" and (b) the image of "Lena". The dotted lines are with $\alpha_{\max} = \beta_{\max} = 0.1$ and $\alpha = \beta = 0.1$; the dash dotted lines 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 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 $\alpha$ and $\beta$ 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 $(\alpha_{\max}, \beta_{\max}) = (0.1, 0.1)$, i.e., the pair $(\alpha, \beta)$ will be kept unchanged during iterations. The PSNR-value of the restored image at each iteration is plotted (dotted line) in Figure 1(a). 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 $(\alpha, \beta)$ will be kept unchanged during iterations. The dash dotted line in Figure 1(a) 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 $\alpha$ is small, i.e., $1/\alpha$ is large, 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 $\alpha$ 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 Figure 1(a) 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 $(\alpha, \beta)$ with large and small values. Under the same setting of parameters, a quite similar phenomenon (see Figure 1(b)) 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 $\alpha$ and $\beta$ every fixed number of iterations until they exceed the pre-given numbers $\alpha_{\max}$ and $\beta_{\max}$. Also, the parameter $\mu$ in the algorithm is to indicate that the underlying algorithm with a pair $(\alpha, \beta)$ will iterate $\mu$ times before the algorithm with the pair $(2\alpha, 2\beta)$ runs another $\mu$ times. According to our numerical experiments, $\mu$ ranging from 10 to 25 usually produces reasonable results.

In the above L1/TV image denoising model, if we replace the convex function $\varphi$ 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. We apply the result on the L1/env $\circ$ B model (3.5) with $\mathcal{C} = \mathbb{R}^{n^2}$ to the modified L1/TV image denoising model to obtain the following result.

**Proposition 9.** *Let $x \in \mathbb{R}^{n^2}$ be given, $\lambda$ and $\beta$ be positive numbers, $B$ be the matrix given by (4.1), and $\varphi$ 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*

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

*then $u$ is a solution of the modified L1/TV image denoising model. Furthermore, if $\alpha$ satisfies*

$$\alpha \geq \frac{8\beta}{\lambda}\sin^2\frac{(n-1)\pi}{2n}, \tag{4.6}$$

*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 6 with (3.7) 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 [35]). This fact, together with hypothesis (4.6), ensures that the assumption of Proposition 8 is satisfied. Hence, by Proposition 8 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 9 leads to the following algorithm, with guaranteed convergence, if $\alpha$ satisfies the inequality (4.6).

**Algorithm 2.**

1. *Given: noisy image* $x \in \mathbb{R}^{n^2}$*;* $\lambda > 0$*,* $\beta > 0$*,* $\beta_{\max}$*,* $\mu \in \mathbb{N}$*, and* $\alpha = \frac{8\beta}{\lambda}$
2. *Initialization:* $u^1 = x$*, and* $b^1 = 0$
3. *For* $k \in \mathbb{N}$*,*
   (a) *if* $\beta < \beta_{\max}$ *and* $k$ *is a multiple of* $\mu$*, update* $\alpha \longleftarrow 2\alpha$ *and* $\beta \longleftarrow 2\beta$*; if* $\beta \geq \beta_{\max}$*, then set* $\alpha = \frac{8\beta_{\max}}{\lambda}$ *and* $\beta = \beta_{\max}$*;*
   (b) $u^{k+1} \longleftarrow x + \mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(u^k - x - \frac{\beta}{\lambda\alpha} B^\top (\mathrm{I} - \mathrm{prox}_{\frac{1}{\beta}\varphi})(Bu^k))$
   (c) *Stop, if* $u^{k+1}$ *converges or satisfies a stopping criteria; otherwise, set* $k \leftarrow k+1$*.*

The parameter $\beta$ appeared in Algorithm 2 is an essential parameter in the L1/env $\circ$ B model (3.5). The output from Algorithm 2 can be understood as a solution of the L1/env $\circ$ B model (3.5) with $\beta$ being $\beta_{\max}$, that is,

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

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

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* $\mu$*, 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) $u^{k+1} \longleftarrow x + (I - P_\Lambda)\mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(u^k - x - \frac{\beta}{\lambda\alpha} B^\top b^k)$
   (c) $b^{k+1} \longleftarrow (I - \mathrm{prox}_{\frac{1}{\beta}\varphi})(Bu^{k+1} + b^k)$
   (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 $\varphi$ by its envelope, we call the model the modified L1/TV image denoising model with prior information and similar to Proposition 10 have the following result.

**Proposition 10.** *Let* $x \in \mathbb{R}^{n^2}$ *be given,* $\lambda$ *and* $\beta$ *be positive numbers,* $B$ *be the matrix given by* (4.1)*, and* $\varphi$ *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 equation* (3.7)*, then* $u$ *is a solution of the the modified L1/TV image denoising model with prior known information. Furthermore, if* $\alpha$ *satisfies* (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.*

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

**Algorithm 4.**

1. *Given: noisy image* $x \in \mathbb{R}^{n^2}$; $\lambda > 0$, $\beta > 0$, $\beta_{\max}$, $\mu \in \mathbb{N}$, *and* $\alpha = \frac{8\beta}{\lambda}$

2. *Initialization:* $u^1 = x$, *and* $b^1 = 0$

3. *For* $k \in \mathbb{N}$,

   (a) *if* $\beta < \beta_{\max}$ *and* $k$ *is a multiple of* $\mu$, *update* $\alpha \longleftarrow 2\alpha$ *and* $\beta \longleftarrow 2\beta$; *if* $\beta \geq \beta_{\max}$, *then set* $\alpha = \frac{8\beta_{\max}}{\lambda}$ *and* $\beta = \beta_{\max}$;

   (b) $u^{k+1} \longleftarrow x + (I - P_\Lambda)\mathrm{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(u^k - x - \frac{\beta}{\lambda\alpha}B^\top(\mathrm{I} - \mathrm{prox}_{\frac{1}{\beta}\varphi})(Bu^k))$

   (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

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

# 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. Salt-pepper noise is often caused by malfunctioning pixels in camera sensors, faulty memory locations in hardware, or transmission in a noisy channel [7]. A considerable amount of work has been done for removing salt-pepper noise from a noisy image. Roughly speaking, there are two types of methods for salt-pepper noise removal. One is the median filter and its variants (see, e.g., [1, 32] and references therein); the other one is the variational approach (see, e.g., [15, 37]). As an alternative to these methods, recent work in [10, 11] employed multiscale analysis based methods for salt-pepper noise removal.

Let $u$ be an ideal $n \times n$ image whose pixel values are in the dynamic range $[S_{min}, S_{max}]$. Suppose that $x$ is a noisy image of $u$ corrupted by salt-pepper noise. The noisy image $x$ is generated in the following manner:

$$x_i = \begin{cases} S_{min} & \textit{with probability } p \\ S_{max} & \textit{with probability } q \\ u_i & \textit{with probability } 1 - p - q, \end{cases} \qquad i \in \mathbb{N}_{n^2},$$

where $p$ and $q$ are two positive numbers with $p + q < 1$. The number $p + q$ is called the salt-pepper noise level of $x$. In our experiments, $p$ and $q$ are chosen to be the same.

In our numerical experiments, the testing images shown in Figure 2 are the "Cameraman" and "Lena" with size of $256 \times 256$ and the "Baboon" with size of $512 \times 512$. The quality of the restored images is evaluated in terms of the peak-signal-to-noise ratio (PSNR) defined by

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

(a) (b) (c)

Figure 2: Original images. (a) “Cameraman”; (b) “Lena”; and (c) “Baboon”.

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 [23] and Algorithm FTVd in [42]. 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 [23]. The parameters $\lambda$ and $\gamma$ used in Algorithm PDTV are chosen to be $\lambda = 0.001$ and $\gamma = 0.01$. The stopping criterion is set according to the detailed description in [23]. The implementation and the website of the corresponding source code package “FTVdv4.1” of Algorithm FTVd can be found in [42]. The parameters $\beta_1$ and $\beta_2$ 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 $\alpha_{\max} = 4$ and the initial value $\alpha = \frac{1}{64}$ and in the meanwhile keep $\beta = \frac{1}{8}\alpha\lambda$ in every iteration. For both of these algorithms, we double the values of $\alpha$ and $\beta$ every ten iterations. The stopping criterion of Algorithm FTVd and Algorithms 1 and 2 is that the relative error between the successive iterates of the restored images should satisfy the following inequality

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

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

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

| | | Cameraman | | Lena | | Baboon | |
|---|---|---|---|---|---|---|---|
| Level | Method | PSNR (dB) | Time (seconds) | PSNR (dB) | Time (seconds) | PSNR (dB) | Time (seconds) |
| 10% | PDTV | 28.80 | 17.03 | 31.42 | 18.08 | 26.49 | 128.98 |
| | FTVd | 28.68 | 3.49 | 31.03 | 3.31 | 26.29 | 18.39 |
| | Algorithm 1 | 28.81 | 2.17 | 31.46 | 2.36 | 26.43 | 10.68 |
| | Algorithm 2 | 28.91 | 1.15 | 31.75 | 1.23 | 26.66 | 5.21 |
| 20% | PDTV | 26.28 | 16.85 | 29.11 | 17.12 | 23.84 | 119.47 |
| | FTVd | 26.15 | 3.46 | 28.76 | 3.33 | 23.73 | 20.21 |
| | Algorithm 1 | 26.35 | 2.43 | 29.11 | 2.43 | 23.92 | 11.17 |
| | Algorithm 2 | 26.59 | 1.18 | 29.58 | 1.47 | 24.24 | 6.44 |
| 30% | PDTV | 24.83 | 15.66 | 27.55 | 16.46 | 22.61 | 122.74 |
| | FTVd | 24.71 | 3.80 | 27.25 | 3.66 | 22.38 | 21.72 |
| | Algorithm 1 | 24.92 | 2.74 | 27.57 | 2.75 | 22.61 | 12.53 |
| | Algorithm 2 | 25.15 | 1.50 | 27.88 | 1.67 | 22.84 | 7.77 |
| 40% | PDTV | 23.64 | 17.35 | 26.06 | 15.55 | 21.56 | 129.51 |
| | FTVd | 23.59 | 4.25 | 25.79 | 3.99 | 21.41 | 23.59 |
| | Algorithm 1 | 23.77 | 2.75 | 26.00 | 2.81 | 21.56 | 13.09 |
| | Algorithm 2 | 23.95 | 1.63 | 26.46 | 1.82 | 21.68 | 9.59 |
| 50% | PDTV | 22.57 | 16.45 | 24.77 | 17.40 | 20.77 | 135.11 |
| | FTVd | 22.39 | 4.66 | 24.52 | 4.52 | 20.69 | 25.31 |
| | Algorithm 1 | 22.49 | 2.85 | 24.30 | 2.84 | 20.74 | 14.19 |
| | Algorithm 2 | 22.74 | 1.85 | 24.82 | 1.93 | 20.81 | 9.93 |

that no prior knowledge of potential uncorrupted pixels are utilized in these four algorithms. To better visualize the performance of the four algorithms, the results in Table 1 are plotted in Figure 3.

The restored images obtained from the four algorithms for the noise level at 30% are displayed in Figure 4. 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) [28] is used for detecting and labeling the salt-pepper noise, therefore, 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 $\alpha$ and $\beta$ for Algorithm 3 are set to 1, which are larger than that in Algorithms 1 and 2. We choose $\alpha_{\max} = \beta_{\max} = 128$ in Algorithm 3. For Algorithm 4, we set $\alpha_{\max} = 128$ and $\alpha = 1$ and keep $\beta = \frac{1}{8}\alpha\lambda$. The parameter $\mu$ 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 Figure 5, 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 (2.2) and (3.5). On the other hand, the L1/$\varphi \circ$ B and L1/env $\circ$ B models (2.2) and (3.5) 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

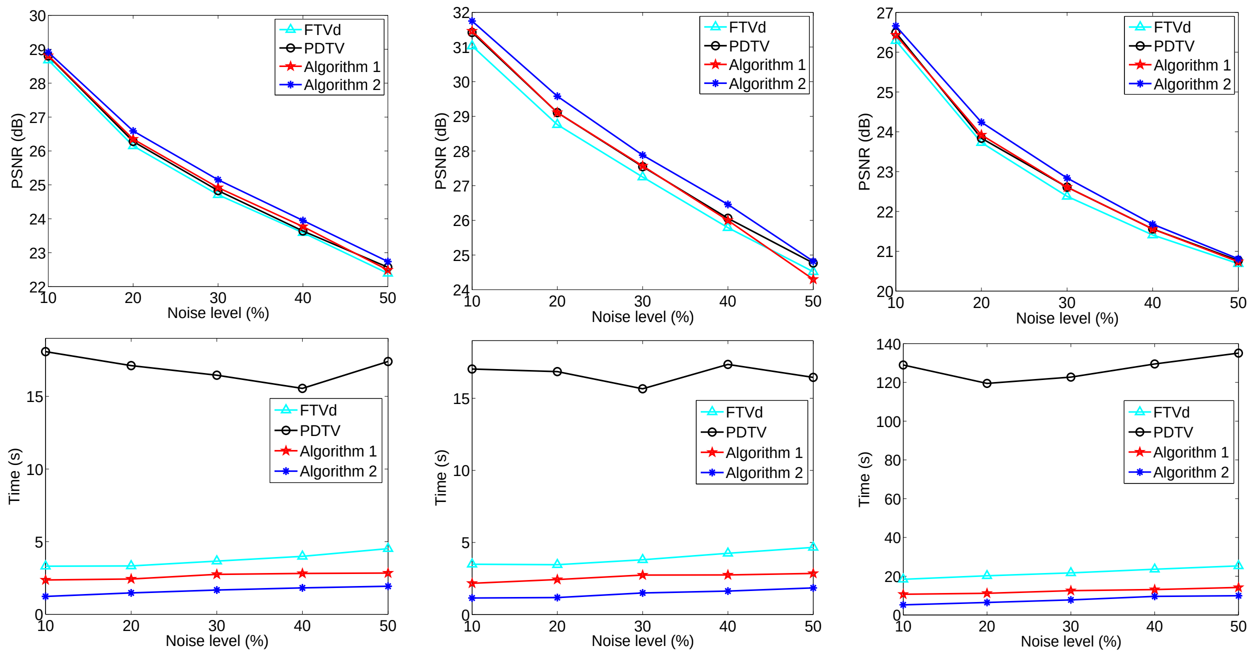



Figure 3: Performance of Algorithm PDTV, Algorithm FTVd and Algorithms 1, 2 on the images of "Cameraman", "Lena", and "Baboon" contaminated by different levels of salt-pepper noise. Row 1: The PSNR-values against noise levels; Row 2: CPU-times against noise levels.

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

| | | Cameraman | | Lena | | Baboon | |
|---|---|---|---|---|---|---|---|
| Level | Method | PSNR (dB) | Time (seconds) | PSNR (dB) | Time (seconds) | PSNR (dB) | Time (seconds) |
| 10% | Algorithm 3 | 36.93 | 2.17 | 40.03 | 2.14 | 32.66 | 10.63 |
| | Algorithm 4 | 36.67 | 0.49 | 39.76 | 0.57 | 33.03 | 3.00 |
| 20% | Algorithm 3 | 33.29 | 2.37 | 36.58 | 2.27 | 29.51 | 11.45 |
| | Algorithm 4 | 33.20 | 0.71 | 36.34 | 0.72 | 29.90 | 3.81 |
| 30% | Algorithm 3 | 30.88 | 2.37 | 34.05 | 2.45 | 27.46 | 11.50 |
| | Algorithm 4 | 30.86 | 0.91 | 33.87 | 0.96 | 27.89 | 4.59 |
| 40% | Algorithm 3 | 29.21 | 2.42 | 32.40 | 2.44 | 25.83 | 11.84 |
| | Algorithm 4 | 29.18 | 1.24 | 32.25 | 1.21 | 26.26 | 6.08 |
| 50% | Algorithm 3 | 27.69 | 2.38 | 30.57 | 2.52 | 24.56 | 11.68 |
| | Algorithm 4 | 27.66 | 1.49 | 30.42 | 1.53 | 24.85 | 8.00 |

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 Figure 5.

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", "Lena", and "Baboon" images corrupted by salt-pepper noise at noise levels of 70% and 90% are depicted in Figure 6 and Figure 7, respectively. These clearly show that Algorithms 3 and 4 are indeed very efficient.

Figure 4: Results of the four methods when restoring noisy "Cameraman", "Lena", and "Baboon" 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.

## 6 Concluding Remarks

We establish proximity operator frameworks for the numerical treatment of the $\mathrm{L1}/\varphi \circ \mathrm{B}$ and $\mathrm{L1}/\mathrm{env} \circ \mathrm{B}$ models and apply them to the image denoising models for their numerical solutions.

The resulting fixed-point equations are formed via the proximity operators of the $\ell^1$-norm and $\ell^2$-norm that have explicit expressions. This naturally leads to simple, efficient fixed-point algorithms for finding approximate solutions of the $\mathrm{L1}/\varphi \circ \mathrm{B}$ and $\mathrm{L1}/\mathrm{env} \circ \mathrm{B}$ image denoising models. We prove convergence of the algorithms for the $\mathrm{L1}/\mathrm{env} \circ \mathrm{B}$ 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.

**Acknowledgment**: This research is supported in part by the US National Science Foundation under grant DMS-0712827, 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] E. Abreu, M. Lightstone, S. Mitra, and K. Arakawa, *A new efficient approach for the removal of impulse noise from highly corrupted images*, IEEE Transactions on Image Processing, 5 (1996), pp. 1012–1025.

[2] S. Alliney, *A property of the minimum vectors of a regularizing functional defined by means of the absolute norm*, IEEE Transactions on Signal Processing, 45 (1997), pp. 913–917.

[3] J. E. Aujol, G. Gilboa, T. Chan, and S. Osher, *Structure-texture image decomposition-modeling, algorithms, and parameter selection*, International Journal of Computer Vision, 67 (2006), pp. 111–136.

[4] H. Bauschke and J. Borwein, *On projection algorithms for solving convex feasibility problems*, SIAM Review, 38 (1996), pp. 367–426.

[5] D. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, Massachussets, 2003.

[6] P. Bloomfield and W. Steiger, *Least Absolute Deviations*, Birkhauser, Boston-Basel, 1983.

[7] A. Bovik, *Handbook of Image and Video Processing*, Academic Press, San Diego, 2000.

[8] C. Byrne, *A unified treatment of some iterative algorithms in signal processing and image reconstruction*, Inverse Problems, 20 (2004), pp. 103–120.

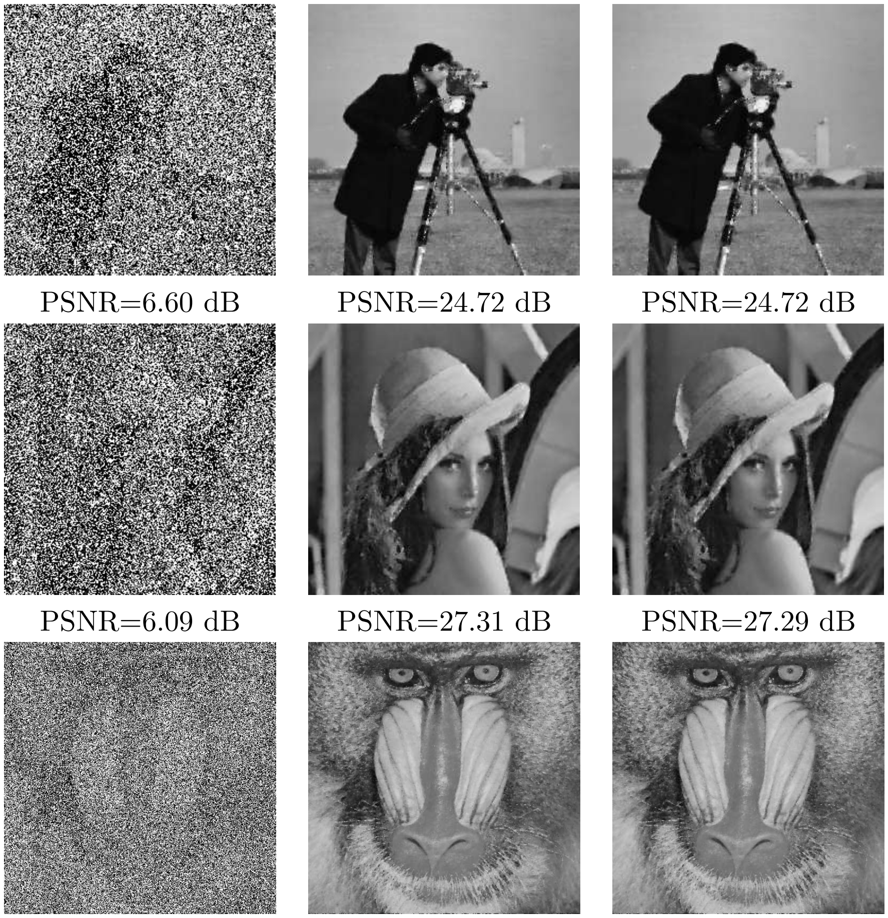
[9] J.-F. Cai, R. Chan, L. Shen, and Z. Shen, *Simultaneously inpainting in image and transformed domains*, Numerische Mathematik, 112 (2009), pp. 509–533.

[10] J.-F. Cai, R. Chan, and Z. Shen, *A framelet-based image inpainting algorithm*, Applied and Computational Harmonic Analysis, 24 (2007), pp. 131–149.

[11] J.-F. Cai, R. H. Chan, L. Shen, and Z. Shen, *Convergence analysis of tight framelet approach for missing data recovery*, Advances in Computational Mathematics, 31 (2009), pp. 87–113.

Figure 5: Results of Algorithm 1 (Row 2), Algorithm 2 (Row 3), Algorithm 3 (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”, “Lena”, and “Baboon”.

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



PSNR=6.60 dB PSNR=24.72 dB PSNR=24.72 dB

PSNR=6.09 dB PSNR=27.31 dB PSNR=27.29 dB

PSNR=7.12 dB PSNR=22.33 dB PSNR=22.39 dB

Figure 6: Restoration results of corrupted "Cameraman", "Lena", and "Baboon" images (Left column) with noise level $r = 70\%$ by Algorithm 3 (Middle column) and Algorithm 4 (Right column).


[13] E. Candes and T. Tao, *Near optimal signal recovery from random projections: Universal encoding strategies?*, IEEE Transactions on Information Theory, 52 (2006), pp. 5406–5425.

[14] A. Chambolle and P. L. Lions, *Image recovery via total variational minimization and related problems*, Numerische Mathematik, 76 (1997), pp. 167–188.

[15] R. Chan, C.-W. Ho, and M. Nikolova, *Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization*, IEEE Transactions on Image Procssing, 14 (2005), pp. 1479–1485.

[16] R. Chan, S. D. Riemenschneider, L. Shen, and Z. Shen, *Tight frame: The efficient way for high-resolution image reconstruction*, Applied and Computational Harmonic Analysis, 17 (2004), pp. 91–115.

[17] T. Chan and S. Esedoglu, *Aspects of total variation regularized $l^1$ function approximation*, SIAM Journal on Applied Mathematics, 65 (2005), pp. 1817–1837.

[18] T. Chan, A. Marquina, and P. Mulet, *High-order total variation-based image restoration*, SIAM Journal on Scientific Computing, 22 (2000), pp. 503–516.

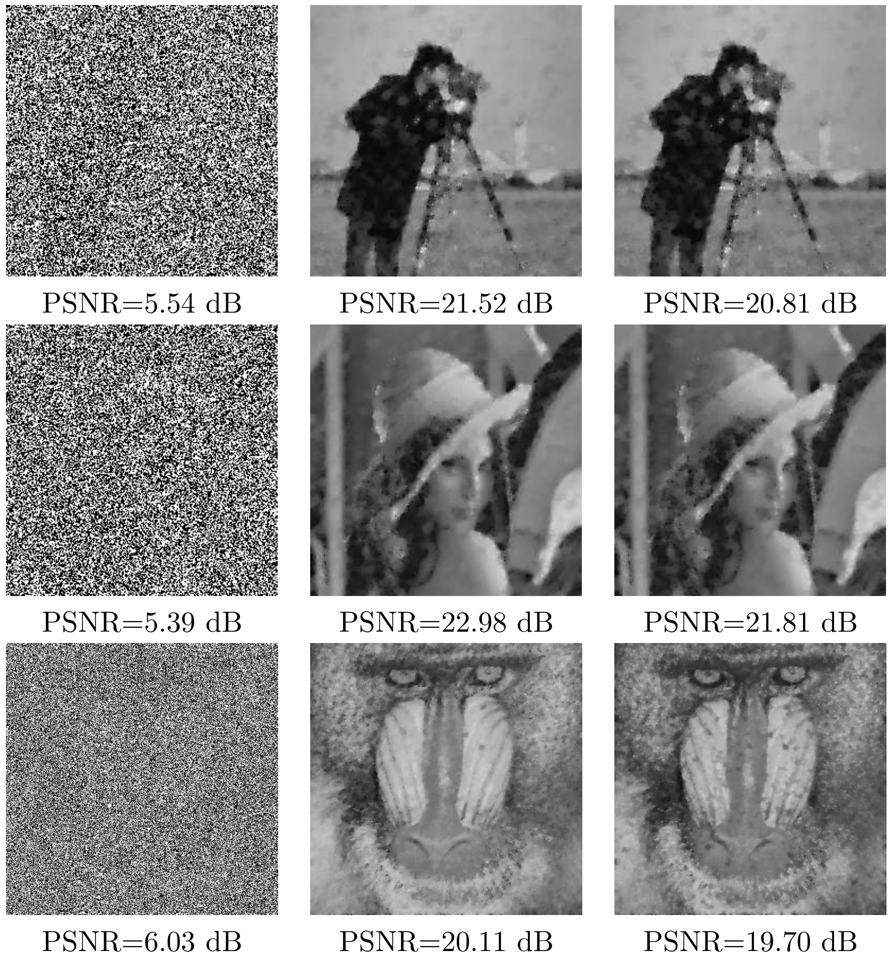



Figure 7: Restoration results of corrupted "Cameraman", "Lena", and "Baboon" images (Left column) with noise level $r = 90\%$ by Algorithm 3 (Middle column) and Algorithm 4 (Right column).

[19] T. Chen, W. Yin, X. S. Zhou, D. Comaniciu, and T. Huang, *Total variation models for variable lighting face recognition*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 28 (2006), pp. 1519–1524.

[20] C. Clason, B. Jin, and K. Kunisch, *A duality-based splitting method for L1-TV image restoration with automatic regularization parameter choice*, SIAM Journal on Scientific Computing, 32 (2010), pp. 1484–1505.

[21] P. Combettes and V. Wajs, *Signal recovery by proximal forward-backward splitting*, Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal, 4 (2005), pp. 1168–1200.

[22] I. Daubechies, M. Defrise, and C. D. Mol, *An iterative thresholding algorithm for linear inverse problems with a sparsity constraint*, Communications on Pure and Applied Mathematics, 57 (2004), pp. 1413–1541.

[23] Y. Dong, M. Hintermuller, and M. Neri, *A primal-dual method for $l^1$-TV image denoising*, SIAM Journal on Imaging Sciences, 2 (2009), pp. 1168–1189.

[24] M. Fornasier and R. Rauhut, *Recovery algorithms for vector-valued data with joint sparsity constraints*, SIAM Journal on Numerical Analysis, 46 (2008), pp. 577–613.

[25] H. Fu, M. K. Ng, M. Nikolova, and J. L. Barlow, *Efficient minimization of mixed $\ell^2 - \ell^1$ and $\ell^1 - \ell^1$ nomrs for image restoration*, SIAM Journal on Scientific Computing, 27 (2006), pp. 1881–1902.

[26] K. Goebel and S. Reich, *Uniform Convexity, Hyperbolic Geometry, and Nonexpansive Mappings*, Marcel Dekker, New York, NY, 1984.

[27] D. Goldfard and W. Yin, *Second-order cone programming methods for total variation based image restoration*, SIAM Journal on Scientific Computing, 27 (2005), pp. 622–645.

[28] R. Gonzalez and R. Woods, *Digital Image Processing*, Addison-Wesley, Boston, MA, 1993.

[29] X. Guo, F. Li, and M. Ng, *A fast $\ell^1$-TV algorithm for image restoration*, SIAM Journal of Scientific Computing, 31 (2009), pp. 2322–2341.

[30] J.-B. Hiriart-Urruty and C. Lemarechal, *Convex analysis and minimization algorithms II*, vol. 306 of Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences], Springer-Verlag, Berlin, 1993.

[31] P. Huber, *Robust Statistics*, second ed., John Wiley & Sons Inc., Hoboken, New Jersey, 2009.

[32] H. Hwang and R. Haddad, *Adaptive median filters: new algorithms and results*, IEEE Transanctions on Image Processing, 4 (1995), pp. 499–502.

[33] M. Lustig, D. Donoho, and J. M. Pauly, *Sparse MRI: The application of compressed sensing for rapid MR imaging*, Magnetic Resonance in Medicine, 58 (2007), pp. 1182–1195.

[34] M. Lysaker and X. Tai, *Iterative image restoration combining total variation minimization and a second-order functional*, International Journal of Computer Vision, 66 (2006), pp. 5–18.

[35] C. A. Micchelli, L. Shen, and Y. Xu, *Proximity algorithms for image models: Denosing*, Inverse Problems, 27(2011), 045009 (30pp).

[36] J.-J. Moreau, *Proximité et dualité dans un espace hilbertien*, Bull. Soc. Math. France, 93 (1965), pp. 273–299.

[37] M. Nikolova, *A variational approach to remove outliers and impulse noise*, Journal of Mathematical Imaging and Vision, 20 (2004), pp. 99–120.

[38] L. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Physica D, 60 (1992), pp. 259–268.

[39] O. Scherzer, *Denoising with higher-order derivatives of bounded variation and an application to parameter estimation*, Computing, 60 (1998), pp. 5–27.

[40] W. Stefan, R. A. Renaut, and A. Gelb, *Improved total variation-type regularization using higher order edge detectors*, SIAM Journal on Imaging Sciences, 3 (2010), pp. 232–251.

[41] J. Tropp, *Greed is good: Algorithmic results for sparse approximation*, IEEE Transactions on Information Theory, 50 (2004), pp. 2231–2242.

[42] J. Yang, Y. Zhang, and W. Yin, *An efficient TVL1 algorithm for deblurring multichannel images corrupted by impulsive noise*, SIAM Journal on Scientific Computing, 31 (2009), pp. 2842–2865.

[43] W. Yin, T. Chen, X. S. Zhou, and A. Chakraborty, *Background correction for cDNA microarray image using the TV+L1 model*, Bioinformatics, 10 (2005), pp. 2410–2416.

[44] W. Yin, D. Goldfard, and S. Osher, *The total variation regularized $l^1$ model for multiscale decomposition*, Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal, 6 (2007), pp. 190–211.

[45] Y. You and M. Kaveh, *Fourth-order partial differential equations for noise removal*, IEEE Transactions on Image Processing, 9 (2000), pp. 1723–1730.

[46] C. Zach, T. Pock, and H. Bischof, *A duality based approach for realtime TV-$l^1$ optical flow*, in Lecture Notes in Computer Sciences, vol. 4713, Berlin Heidelberg, 2007, Springer-Verlag, pp. 214–223.