

# The Moreau Envelope Approach for the L1/TV Image Denoising Model <sup>\*</sup>

Feishe Chen<sup>†</sup>    Lixin Shen<sup>†‡</sup>    Yuesheng Xu<sup>†‡</sup>    Xueying Zeng<sup>§</sup>

October 3, 2012

## Abstract

This paper presents the Moreau envelope viewpoint for the L1/TV image denoising model. The main algorithmic difficulty for the numerical treatment of the L1/TV model lies in the non-differentiability of both the fidelity and regularization terms of the model. To overcome this difficulty, we propose five modified L1/TV models by replacing one or two the non-differentiable functions in the L1/TV model with their corresponding Moreau envelopes. We prove that several existing approaches for the L1/TV model essentially solve some of the modified models, but not the original L1/TV model. Algorithms for the L1/TV model and its five variants are proposed under a unified framework based on the proximity operator. Depending upon whether we smoothen the regularization term or not, two different types of proximity algorithms are presented. The convergence rates of both types of the algorithms are improved significantly by exploring either the strategy of Gauss-Seidel iteration, or the FISTA, or the both. We compare the performance of various modified L1/TV models for the problem of impulsive noise removal, and make recommendations based on our numerical experiments for using these models in applications.

## 1 Introduction

Given a noisy image  $x$  which was contaminated by impulsive noise, we consider a denoised image of  $x$  as a minimizer of the following L1/TV model

$$\min_u \{\lambda \|u - x\|_1 + \|u\|_{\text{TV}}\}, \quad (1.1)$$

where  $\|\cdot\|_1$  represents the  $\ell^1$ -norm,  $\|\cdot\|_{\text{TV}}$  denotes the total-variation, and  $\lambda$  is the regularization parameter balancing the fidelity term  $\|\cdot - x\|_1$  and the regularization term  $\|\cdot\|_{\text{TV}}$ . It is well accepted that the  $\ell^1$ -norm fidelity term can effectively suppress the effect of outliers that may contaminate a given image, and is therefore particularly suitable for handling non-Gaussian additive noise such as impulsive noise [4]. The L1/TV model (1.1) has many distinctive and desirable features. For example, the 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 [8, 21, 27]. Applications of the L1/TV model

---

<sup>\*</sup>This research is supported in part by the US National Science Foundation under grant DMS-1115523, by Guangdong Provincial Government of China through the “Computational Science Innovative Research Team” program.

<sup>†</sup>Department of Mathematics, Syracuse University, Syracuse, NY 13244, USA.

<sup>‡</sup>Guangdong Province Key Lab of Computational Science, School of Mathematics and Computational Sciences, Sun Yat-sen University, Guangzhou 510275, P. R. China.

<sup>§</sup>School of Mathematical Sciences, Ocean University of China, Qingdao 266100, P. R. China.

include computer vision [10], biomedical imaging [26], optical flow and object tracking [28], and shape denoising [28].

In light of the useful 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. An algorithmic difficulty for the numerical treatment of the L1/TV model is the non-differentiability of both the fidelity and regularization terms in the objective function of (1.1). To overcome the difficulty, several numerical approaches were proposed recently for solving the L1/TV model. Depending on how the non-differentiability of the both terms is treated, these approaches can be roughly divided into two categories. In the first category, prior to developing a numerical scheme for the model, additional quadratic terms with some auxiliary variables are added to the objective function of the L1/TV model to tackle either the non-differentiability of its fidelity or regularization term, or the both. For instance, a quadratic term involving an auxiliary variable was introduced to overcome the non-differentiability of the fidelity term in [1]. Depending on whether we view the total variation itself as a whole the non-differentiable function or as a composition of a non-differentiable function with a first order difference matrix, quadratic terms were added to the model in two different ways for the purpose of remedying the regularization term in [11, 19]. Non-differentiability of both the fidelity and regularization terms was handled simultaneously by introducing two extra quadratic terms into the primal formulation of the L1/TV model in [15, 25] and the dual formulation of the model in [13]. Such techniques will certainly benefit the development of the numerical treatment of the model, but change the original L1/TV model to its various modified versions. These modified models may lose the desirable features of the L1/TV model. Unlike the ones in the first category, algorithms in the second category are to faithfully solve the L1/TV model directly without introducing additional terms to the model. The algorithms in this category are developed by using the proximity operator [17, 19], the primal-dual formulation [7, 16], and the augmented Lagrangian method [24].

The main purpose of this paper has twofold. On one hand, the models corresponding to the all approaches classified into the first category can be reformulated in terms of the notion of the Moreau envelope. This, in turn, allows us to develop algorithms for these models in a unified way based upon the proximity operator. On the other hand, we compare the performance of various modified L1/TV models for the problem of impulsive noise removal and make recommendations for the use of these models in applications.

By replacing the non-differentiable functions in the L1/TV model with their corresponding differentiable Moreau envelopes, we obtain a total of six modified L1/TV models including the original one. Based on our previous work in [17, 19], all of the modified L1/TV models are solved under a unified framework based on the proximity operator. In this framework, a solution of a model is characterized in terms of the proximity operators of the convex functions associated with the fidelity and regularization terms. Depending on whether the regularization term of a model is differentiable or not, characterizations to the solutions of the models are presented separately. The characterization for models having a non-differentiable regularization term motivates us to develop the corresponding algorithms for the models that could be implemented by exploring a strategy of the Gauss-Seidel (GS) iteration to speed up its rate of convergence. For the models having a differentiable regularization term, the well-known algorithm FISTA in [3] could be applied.

The performance of the proposed algorithms for the original L1/TV model and its variants is evaluated for the problem of impulsive noise removal. We confirm that for removing impulsive noise the fidelity term of a model should be the  $\ell^1$ -norm of the difference of the image to be sought and the given noisy image. A choice of the regularization term of a model relies upon the nature of the underlying image. It was pointed out in [17] that the original L1/TV model is particularly suitable for cartoon-like images which are approximately piecewise constants. For nature images which

can be viewed as piecewise smooth functions, we comment in this paper via intensive numerical experiments that a model with the  $\ell^1$ -norm fidelity term and a smoothed total-variation as its regularization term is more suitable for the problem of impulsive noise removal. As mentioned in the above, the total-variation can be viewed as a composition of a non-differentiable function with a first order difference matrix. Here, the smoothed total-variation is derived from the total-variation by simply replacing the non-differentiable function by its Moreau envelope.

The outline of this paper is as follows: In Section 2 we propose several possible modified L1/TV models by approximating the non-smooth  $\ell^1$  fidelity term or the total variation term by their corresponding Moreau envelopes. These modified L1/TV models can have several splitting based models in the literature as their special cases. In Section 3, the solutions to the L1/TV models are characterized via fixed-point equations. The corresponding algorithms based on these characterizations and their convergence analysis and accelerating schemes are given in Section 4 for the models having a non-differentiable regularization term and in Section 5 for the models having a differentiable regularization term. In Section 6 we report the numerical experiments, discuss the difference of the modified L1/TV models and compare the performance of the proposed algorithms in restoring images from those corrupted by impulsive noise. We draw our conclusions in the last section.

## 2 The L1/TV Model and Its Moreau Envelope Variants

In this section, we first explain the suitability of model (1.1) for impulsive noise removal from the viewpoint of statistics. We then give five modified versions of model (1.1) in terms of the Moreau envelope. Finally, we point out the connection of the modified L1/TV models with several existing models for solving the original model (1.1).

We begin with a statistical explanation on the L1/TV denoising model for the problem of impulsive noise removal. Impulsive noise is caused by malfunctioning pixels in camera sensors, faulty memory locations in hardware, or transmission in a noisy channel [5]. We denote by  $u \in \mathbb{R}^d$  the original image. Its noisy version  $x$  corrupted by salt-pepper noise (a special type of impulsive noise) with a noise level  $0 < r < 1$  is modeled as

$$x_i = \begin{cases} 0, & \text{with probability } \frac{r}{2}, \\ 255, & \text{with probability } \frac{r}{2}, \\ u_i, & \text{with probability } 1 - r, \end{cases} \quad (2.1)$$

where  $x_i, u_i$  are the  $i$ -th pixel values of  $x$  and  $u$ , respectively. A maximum a posteriori expectation-maximization could be used to find an estimate of  $u$  by maximizing the conditional a posteriori probability  $p(u|x)$ , the probability that  $u$  occurs when  $x$  is observed. Using Bayes' theorem, we have that

$$p(u|x) = \frac{p(u) \cdot p(x|u)}{p(x)}.$$

By taking the negative logarithm of the above equation, the estimate is a solution of the following minimization problem

$$\min_u \{-\log p(x|u) - \log p(u)\}. \quad (2.2)$$

The expression  $-\log p(x|u)$  in (2.2) can be viewed as a fidelity term measuring the discrepancy between the estimate  $u$  and the noisy image  $x$ . The term  $-\log p(u)$  is used to regularize a solution that has a low probability.



Table 1: The L1/TV denoising model and its five variants.

Case	Fidelity term	Regularization term	Model
1	$\ u - x\ _1$	$\varphi(Bu)$	$\min_u \{\lambda \ u - x\ _1 + \varphi(Bu)\}$
2	$\text{env}_{\alpha\ \cdot\ _1}(u - x)$	$\varphi(Bu)$	$\min_u \{\lambda \text{env}_{\alpha\ \cdot\ _1}(u - x) + \varphi(Bu)\}$
3	$\ u - x\ _1$	$\text{env}_{\beta\varphi}(Bu)$	$\min_u \{\lambda \ u - x\ _1 + \text{env}_{\beta\varphi}(Bu)\}$
4	$\text{env}_{\alpha\ \cdot\ _1}(u - x)$	$\text{env}_{\beta\varphi}(Bu)$	$\min_u \{\lambda \text{env}_{\alpha\ \cdot\ _1}(u - x) + \text{env}_{\beta\varphi}(Bu)\}$
5	$\ u - x\ _1$	$\text{env}_{\beta\varphi \circ B}(u)$	$\min_u \{\lambda \ u - x\ _1 + \text{env}_{\beta\varphi \circ B}(u)\}$
6	$\text{env}_{\alpha\ \cdot\ _1}(u - x)$	$\text{env}_{\beta\varphi \circ B}(u)$	$\min_u \{\lambda \text{env}_{\alpha\ \cdot\ _1}(u - x) + \text{env}_{\beta\varphi \circ B}(u)\}$

With the above preparation,  $\|u\|_{\text{TV}}$  the total-variation of  $u$  can be written as  $\varphi(Bu)$ . Hence, the L1/TV image denoising model in (1.1) has the form of

$$\min_u \{\lambda \|u - x\|_1 + \varphi(Bu)\}. \quad (2.8)$$

Notice that in the formulation (2.8) of the L1/TV denoising model the functions  $\|\cdot - x\|_1$  and  $\varphi$  are convex, but, not differentiable. To deal with this difficulty we propose a smooth version of the L1/TV denoising model by smoothing either  $\|\cdot - x\|_1$  or  $\varphi$ , or both.

A typical way of smoothing a non-smooth convex function is to use the Moreau envelope. We denote by  $\Gamma_0(\mathbb{R}^d)$  the class of all lower semicontinuous convex functions  $\psi : \mathbb{R}^d \rightarrow (-\infty, +\infty]$  such that the domain  $\text{dom}(\psi) := \{x \in \mathbb{R}^d : \psi(x) < +\infty\}$  of  $\psi$ , is nonempty. For a positive number  $\beta$  and a function  $\psi \in \Gamma_0(\mathbb{R}^d)$ , the Moreau envelope of  $\psi$  with index  $\beta$  at  $z \in \mathbb{R}^d$  is defined as

$$\text{env}_{\beta\psi}(z) := \min_{y \in \mathbb{R}^d} \left\{ \frac{1}{2\beta} \|y - z\|_2^2 + \psi(y) \right\}, \quad (2.9)$$

which is still in  $\Gamma_0(\mathbb{R}^d)$  (see, e.g., [20, 22]). For any positive number  $\beta$  the Moreau envelope  $\text{env}_{\beta\psi}$  is bounded above by  $\psi$  and it converges to  $\psi$  as  $\beta \rightarrow 0^+$  (see, e.g. [23]). More importantly, the Moreau envelope of a convex function is always differentiable and its gradient is given by

$$\nabla(\text{env}_{\beta\psi}) = \frac{1}{\beta}(I - \text{prox}_{\beta\psi}), \quad (2.10)$$

where

$$\text{prox}_{\beta\psi} := \arg \min_{y \in \mathbb{R}^m} \left\{ \frac{1}{2} \|y - \cdot\|_2^2 + \beta\psi(y) \right\} \quad (2.11)$$

is called the proximity operator of  $\psi$  with index  $\beta$ .

Considering the non-differentiability of the  $\ell^1$ -norm and the function  $\varphi$  in (2.8) and the above remarkable properties of the Moreau envelope, we may propose several smoothed variants of the original model (2.8). For its fidelity term  $\|\cdot - x\|_1$ , we can keep it unchanged or replace it by its envelope  $\text{env}_{\alpha\|\cdot\|_1}(\cdot - x)$ . For the regularization term  $\varphi \circ B$ , besides keeping it unchanged, depending on whether we view the non-differentiable total variation as a whole  $\varphi \circ B$  or as a composition of the non-differentiable function  $\varphi$  with the difference matrix  $B$ , we have two other alternatives:  $\text{env}_{\beta\varphi \circ B}(u)$  and  $\text{env}_{\beta\varphi}(Bu)$ . Considering all of the possible choices, we have five different modified models listed in Table 1, in addition to the original model.

In the rest of this section, we shall point out that our proposed modified L1/TV models in Table 1 are consistent with several existing models which use the splitting technique dealing with

the original L1/TV model. In [1], by introducing an auxiliary variable  $v$  to replace the difference  $u - x$  in the L1/TV model (2.8) and then imposing a condition to penalize the closeness between  $v$  and  $u - x$ , a model of the form

$$\min_{u,v} J_{\text{AGCO}}(u, v) \quad (2.12)$$

is considered, where

$$J_{\text{AGCO}}(u, v) := \lambda \|v\|_1 + \frac{1}{2\gamma} \|v - (u - x)\|_2^2 + \varphi(Bu), \quad (2.13)$$

with  $\gamma$  being a positive number.

**Proposition 1.** *Let  $\lambda$  and  $\gamma$  be positive, let  $x$  be a noisy image in  $\mathbb{R}^d$ , let  $B$  be the  $2d \times d$  matrix defined by (2.6), and let  $\varphi$  be the function defined by (2.7). Then the solution set of the optimization problem (2.12) is nonempty. If the pair  $(u_\star, v_\star)$  is a solution of the optimization problem (2.12), then  $v_\star = \text{prox}_{\alpha\|\cdot\|_1}(u_\star - x)$  where  $\alpha = \lambda\gamma$ . Furthermore, the pair  $(u_\star, v_\star)$  is a solution of the optimization problem (2.12) if and only if  $u_\star$  is that of the optimization problem*

$$\min_u J_2(u), \quad (2.14)$$

where

$$J_2(u) := \lambda \text{env}_{\alpha\|\cdot\|_1}(u - x) + \varphi(Bu), \quad (2.15)$$

*Proof.* We first show the non-emptiness of the solution set of the optimization problem (2.12). Since the negative functions  $\|\cdot\|_1$ ,  $\|\cdot\|_2$ , and  $\varphi$  are convex, so is  $J_{\text{AGCO}}$ . Further, since

$$\lim_{\|(u,v)\|_2 \rightarrow \infty} \left( \lambda \|v\|_1 + \frac{1}{2\gamma} \|v - (u - x)\|_2^2 \right) = +\infty,$$

then  $\lim_{\|(u,v)\|_2 \rightarrow \infty} J_{\text{AGCO}}(u, v) = +\infty$ , that is,  $J_{\text{AGCO}}$  is coercive. Hence, the existence of minimizers of the objective function  $J_{\text{AGCO}}$  follows.

Now, assume that the pair  $(u_\star, v_\star)$  is a solution of the optimization problem (2.12). It follows immediately that  $J_{\text{AGCO}}(u_\star, v_\star) \leq J_{\text{AGCO}}(u_\star, \text{prox}_{\alpha\|\cdot\|_1}(u_\star - x))$ . For ease of exposition, we write  $P(v) := \lambda \|v\|_1 + \frac{1}{2\gamma} \|v - (u_\star - x)\|_2^2$ . Then the above inequality yields  $P(v_\star) \leq P(\text{prox}_{\alpha\|\cdot\|_1}(u_\star - x))$ . By the definition of the Moreau envelope, we know that  $\text{prox}_{\alpha\|\cdot\|_1}(u_\star - x)$  is the unique point such that  $P(v) \geq P(\text{prox}_{\alpha\|\cdot\|_1}(u_\star - x))$  for all  $v \in \mathbb{R}^{2d}$ . Hence,  $v_\star = \text{prox}_{\alpha\|\cdot\|_1}(u_\star - x)$ .

Using the definition of the Moreau envelope again, we know that  $P(\text{prox}_{\alpha\|\cdot\|_1}(u - x)) = \lambda \text{env}_{\alpha\|\cdot\|_1}(u - x)$  for all vectors  $u \in \mathbb{R}^d$ . Therefore, it holds that

$$J_{\text{AGCO}}(u, \text{prox}_{\alpha\|\cdot\|_1}(u - x)) = J_2(u). \quad (2.16)$$

If the pair  $(u_\star, v_\star)$  is a solution of the optimization problem (2.12), we can show that  $u_\star$  is a solution of the optimization problem (2.14). If not, there exists a vector  $\tilde{u}$  such that  $J_2(\tilde{u}) < J_2(u_\star)$ . By (2.16), we have  $J_{\text{AGCO}}(\tilde{u}, \text{prox}_{\alpha\|\cdot\|_1}(\tilde{u} - x)) < J_{\text{AGCO}}(u_\star, \text{prox}_{\alpha\|\cdot\|_1}(u_\star - x))$ , this contradicts the assumption of  $(u_\star, v_\star)$  being the solution of the optimization problem (2.12).

Now, if  $u_\star$  is a solution of the optimization problem (2.14), we show that the pair  $(u_\star, \text{prox}_{\alpha\|\cdot\|_1}(u_\star - x))$  is a solution of the optimization problem (2.12). Indeed, if this statement is not true, then there exists a vector  $\tilde{u}$  such that  $J_{\text{AGCO}}(\tilde{u}, \text{prox}_{\alpha\|\cdot\|_1}(\tilde{u} - x)) < J_{\text{AGCO}}(u_\star, \text{prox}_{\alpha\|\cdot\|_1}(u_\star - x))$ . This, by (2.16), implies that  $J_2(\tilde{u}) < J_2(u_\star)$ , which violates the assumption of  $u_\star$  being a solution of the optimization problem (2.14).  $\square$

By Proposition 1, we conclude that the model (2.12) is “Case 2” in Table 1.

In model (2.12), when an additional auxiliary variable  $w$  is introduced to replace  $Bu$ , we have the following model

$$\min_{u,v,w} J_{YZY}(u, v, w), \quad (2.17)$$

where

$$J_{YZY}(u, v, w) := \lambda \|v\|_1 + \frac{1}{2\gamma} \|v - (u - x)\|_2^2 + \varphi(w) + \frac{1}{2\beta} \|w - Bu\|_2^2, \quad (2.18)$$

with  $\gamma$  and  $\beta$  being positive. Model (2.17) was originally introduced in [25].

**Proposition 2.** *Let  $\lambda$ ,  $\gamma$  and  $\beta$  be positive, let  $x$  be a noisy image in  $\mathbb{R}^d$ , let  $B$  be the  $2d \times d$  matrix defined by (2.6), and let  $\varphi$  be the function defined by (2.7). Then the solution set of the optimization problem (2.17) is nonempty. If the triple  $(u_*, v_*, w_*)$  is a solution of the optimization problem (2.17), then  $v_* = \text{prox}_{\alpha\|\cdot\|_1}(u_* - x)$  and  $w_* = \text{prox}_{\beta\varphi}(Bu_*)$ , where  $\alpha = \lambda\gamma$ . Furthermore, the triple  $(u_*, v_*, w_*)$  is a solution of the optimization problem (2.17) if and only if  $u_*$  is that of the optimization problem*

$$\min_u J_4(u),$$

where

$$J_4(u) := \lambda \text{env}_{\alpha\|\cdot\|_1}(u - x) + \text{env}_{\beta\varphi}(Bu).$$

*Proof.* In a similar way that leads to (2.16), we have that  $J_{YZY}(u, \text{prox}_{\alpha\|\cdot\|_1}(u - x), \text{prox}_{\beta\varphi}(Bu)) = J_4(u)$  for all vectors  $u \in \mathbb{R}^d$ . With this equation, the rest proof of this result is similar to that of Proposition 1, and it is omitted.  $\square$

By Proposition 2, model (2.17) is “Case 4” in Table 1.

In [15], the model

$$\min_{u,v,w} J_{GLN}(u, v, w) \quad (2.19)$$

was proposed, where

$$J_{GLN}(u, v, w) := \|u - w\|_2^2 + \alpha_1 \|u - v\|_2^2 + \alpha_2 \|w - x\|_1 + \alpha_3 \varphi(Bv) \quad (2.20)$$

with  $\alpha_1$ ,  $\alpha_2$ , and  $\alpha_3$  being positive numbers.

**Proposition 3.** *Let  $\alpha_1$ ,  $\alpha_2$ , and  $\alpha_3$  be positive, let  $x$  be a noisy image in  $\mathbb{R}^d$ , let  $B$  be the  $2d \times d$  matrix defined by (2.6), and let  $\varphi$  be the function defined by (2.7). Then the solution set of the optimization problem (2.19) is nonempty. If the triple  $(u_*, v_*, w_*)$  is a solution of the optimization problem (2.19), then  $v_* = \text{prox}_{\beta\varphi \circ B}(u_*)$  and  $w_* = x + \text{prox}_{\alpha\|\cdot\|_1}(u_* - x)$ , where  $\alpha = \frac{\alpha_2}{2}$  and  $\beta = \frac{\alpha_3}{2\alpha_1}$ . Furthermore, the triple  $(u_*, v_*, w_*)$  is a solution of the optimization problem (2.19) if and only if  $u_*$  is that of the optimization problem*

$$\min_u J_6(u),$$

where

$$J_6(u) := \lambda \text{env}_{\alpha\|\cdot\|_1}(u - x) + \text{env}_{\beta\varphi \circ B}(u)$$

with  $\lambda = \frac{\alpha_2}{\alpha_3}$ .

*Proof.* By using the definitions of the proximity operator and the Moreau envelope, we have that for all vectors  $u \in \mathbb{R}^d$   $J_{GLN}(u, \text{prox}_{\beta\varphi \circ B}(u), x + \text{prox}_{\alpha\|\cdot\|_1}(u - x)) = \alpha_3 J_6(u)$ . Using this equation, the proof of this proposition is similar to that of Proposition 1. Therefore, we omit its proof here.  $\square$

We conclude from Proposition 3 that the alternative model (2.19) is “Case 6” in Table 1. In [11], the model

$$\min_{u,v} J_{\text{CJK}}(u,v), \quad (2.21)$$

was discussed, where

$$J_{\text{CJK}}(u,v) := \lambda \|u - x\|_1 + \varphi(Bv) + \frac{1}{2\beta} \|u - v\|_2^2 \quad (2.22)$$

with  $\lambda$  and  $\beta$  being positive numbers.

**Proposition 4.** *Let  $\lambda$  and  $\beta$  be positive, let  $x$  be a noisy image in  $\mathbb{R}^d$ , let  $B$  be the  $2d \times d$  matrix defined by (2.6), and let  $\varphi$  be the function defined by (2.7). Then the solution set of the optimization problem (2.21) is nonempty. If the pair  $(u_*, v_*)$  is a solution of the optimization problem (2.21), then  $v_* = \text{prox}_{\beta\varphi \circ B}(u_*)$ . Furthermore, the pair  $(u_*, v_*)$  is a solution of the optimization problem (2.21) if and only if  $u_*$  is that of the optimization problem*

$$\min_u J_5(u),$$

where

$$J_5(u) := \lambda \|u - x\|_1 + \text{env}_{\beta\varphi \circ B}(u).$$

*Proof.* By using the definitions of the proximity operator and the Moreau envelope, we have that  $J_{\text{CJK}}(u, \text{prox}_{\beta\varphi \circ B}(u)) = J_5(u)$  for all vectors  $u \in \mathbb{R}^d$ . Using this equation, the proof of this proposition is similar to that of Proposition 1, and we skip the details.  $\square$

Proposition 4 says that model (2.21) is “Case 5”.

In summary, the key difference of the work in [1, 11, 15, 25] is that where the splitting happens. We remark that the splitting technique in these work was initially motivated for convenience in applying alternating minimization approaches to the generated alternative L1/TV models or the corresponding dual models. However, as we pointed out above, the splitting technique is essentially to replace the non-differentiable functions in the original L1/TV model by their corresponding differentiable Moreau envelopes. This understanding provides us an opportunity to numerically consider all the six models in a unified framework based on the proximity operator.

### 3 Characterization of Solutions to the L1/TV Models

In this section, we characterize the solutions of the modified L1/TV models listed in Table 1. All these models can be cast into the following general form

$$\min_u \{\lambda F(u - x) + R(Wu)\} \quad (3.1)$$

by specifying the functions  $F$  and  $R$  and the matrix  $W$  properly. The correspondence between a model listed in Table 1 and a choice of  $F$ ,  $R$  and  $W$  is illustrated in Table 2.

We first provide a characterization of a solution of model (3.1). This characterization will be further refined if the function  $R$  is differentiable. To characterize a solution of model (3.1), we need to briefly review the concept of subdifferential in convex analysis.

The subdifferential of a proper convex function  $\psi \in \Gamma_0(\mathbb{R}^d)$  at a given vector  $u \in \mathbb{R}^d$  is the set defined by

$$\partial\psi(u) := \{v : v \in \mathbb{R}^d \text{ and } \psi(w) \geq \psi(u) + \langle v, w - u \rangle \text{ for all } w \in \mathbb{R}^d\}. \quad (3.2)$$

Table 2: Relations of model (3.1) to modified L1/TV models

Case	F	R	W
1	$\ \cdot\ _1$	$\varphi$	$B$
2	$\text{env}_\alpha\ \cdot\ _1$	$\varphi$	$B$
3	$\ \cdot\ _1$	$\text{env}_{\beta\varphi}$	$B$
4	$\text{env}_\alpha\ \cdot\ _1$	$\text{env}_{\beta\varphi}$	$B$
5	$\ \cdot\ _1$	$\text{env}_{\beta\varphi\circ B}$	$I$
6	$\text{env}_\alpha\ \cdot\ _1$	$\text{env}_{\beta\varphi\circ B}$	$I$

The subdifferential and the proximity operator of the function  $\psi$  are related in the following way (see, e.g. [18]): For  $u$  in the domain of  $\psi$  and  $v \in \mathbb{R}^d$

$$v \in \partial\psi(u) \text{ if and only if } u = \text{prox}_\psi(u + v). \quad (3.3)$$

With the help of the relationship between the subdifferential and the proximity operator, we can characterize a solution of model (3.1) via fixed-point equations.

**Proposition 5.** *Let  $x$  be in  $\mathbb{R}^d$  and let  $\lambda$  be a positive number. If  $u \in \mathbb{R}^d$  is a solution of model (3.1), where  $F \in \Gamma_0(\mathbb{R}^d)$ ,  $R \in \Gamma_0(\mathbb{R}^m)$ , and  $W$  is a  $m \times d$  matrix, then for any  $\gamma, \sigma > 0$ , there exist two vectors  $v, b$  in  $\mathbb{R}^m$  such that*

$$u = x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( \left( I - \frac{\sigma}{\gamma}W^\top W \right) u - \frac{\sigma}{\gamma}W^\top(b - v) - x \right) \quad (3.4)$$

$$v = \text{prox}_{\frac{1}{\sigma}R}(b + Wu) \quad (3.5)$$

$$b = b + Wu - v \quad (3.6)$$

Conversely, if there exist  $\gamma > 0$ ,  $\sigma > 0$ ,  $v \in \mathbb{R}^m$ ,  $b \in \mathbb{R}^m$ , and  $u \in \mathbb{R}^d$  satisfying equations (3.4), (3.5), and (3.6), then  $u$  is a solution of model (3.1).

*Proof.* We first assume that  $u \in \mathbb{R}^d$  is a solution of model (3.1). By the Fermat rule in convex analysis, we have that

$$0 \in \partial(\lambda F)(u - x) + W^\top \partial R(Wu).$$

For any positive numbers  $\gamma, \sigma$ , we can select a vector  $a \in \frac{1}{\gamma}\partial(\lambda F)(u - x)$  and a vector  $b \in \frac{1}{\sigma}\partial R(Wu)$  such that  $0 = \gamma a + \sigma W^\top b$ , that is,  $a = -\frac{\sigma}{\gamma}W^\top b$ . Hence,  $-\frac{\sigma}{\gamma}W^\top b \in \frac{1}{\gamma}\partial(\lambda F)(u - x)$ . By equation (3.3), we have that

$$u = x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( u - \frac{\sigma}{\gamma}W^\top b - x \right). \quad (3.7)$$

By equation (3.3) again, from the inclusion  $b \in \frac{1}{\sigma}\partial R(Wu)$  we have that  $Wu = \text{prox}_{\frac{1}{\sigma}R}(Wu + b)$ . Set

$$v := \text{prox}_{\frac{1}{\sigma}R}(Wu + b), \quad (3.8)$$

which is equation (3.5). Clearly,  $Wu = v$ , therefore, equation (3.6) holds. Substituting  $b$  in equation (3.7) by  $b + Wu - v$  leads to equation (3.4).

Conversely, suppose that there exist  $\gamma > 0, \sigma > 0, v, b \in \mathbb{R}^m$  and  $u \in \mathbb{R}^d$  satisfying equations (3.4)-(3.6). By equation (3.6), it holds that  $Wu = v$ . With this relation, by equation (3.3),

equations (3.4) and (3.5) ensure that  $-\frac{\sigma}{\gamma}W^T b \in \frac{1}{\gamma}\partial(\lambda F)(u-x)$  and  $b \in \frac{1}{\sigma}\partial R(Wu)$  respectively. Combining these two inclusion relations yields that

$$0 = \gamma\left(-\frac{\sigma}{\gamma}W^T b\right) + \sigma W^T b \in \partial(\lambda F)(u-x) + W^T \partial R(Wu)$$

This in turn implies that  $u \in \mathbb{R}^d$  is a solution of model (3.1).  $\square$

We remark that the characterization given in Proposition 5 reduces to the one given in [17] when the function  $F$  in model (3.1) is the  $\ell^1$ -norm. The integer  $m$ , the number of the rows of the matrix  $W$ , is  $2d$  for “Case 1”-“Case 4” and  $d$  for “Case 5”-“Case 6”.

Next, we show that if the convex function  $R$  is differentiable, a solution of model (3.1) can be characterized with only one fixed-point equation.

**Proposition 6.** *Let  $x$  be in  $\mathbb{R}^d$ , let  $\lambda$  be a positive number, let  $F \in \Gamma_0(\mathbb{R}^d)$ , let  $W$  be a  $m \times d$  matrix, and let  $R$  be in  $\Gamma_0(\mathbb{R}^m)$  and differentiable. If  $u \in \mathbb{R}^d$  is a solution of the model (3.1), then for any  $\gamma > 0$*

$$u = x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( u - \frac{1}{\gamma}W^T \nabla R(Wu) - x \right). \quad (3.9)$$

*Conversely, if  $u \in \mathbb{R}^d$  satisfies equations (3.9) for some  $\gamma > 0$ , then  $u$  is a solution of model (3.1).*

*Proof.* According to Proposition 5, we need only proof the fixed-point equations (3.4)-(3.6) can be refined to equation (3.9) when  $R$  is differentiable. Notice that, by omitting  $v$ , equations (3.5) and (3.6) imply that  $Wu = \text{prox}_{\frac{1}{\sigma}R}(b + Wu)$ . Hence,  $b \in \partial R(Wu)$  by equation (3.3). Recall that the subdifferential of a differentiable function  $R$  at a given point  $z \in \mathbb{R}^m$  is a singleton set, that is  $\partial R(z) = \{\nabla R(z)\}$ , where  $\nabla R(z)$  denotes the gradient of  $R$  at  $z$ . Hence, we have that  $b = \nabla R(Wu)$ . Substituting this result into (3.4) and eliminating variables  $b$  and  $v$  yield the fixed-point equation (3.9). This completes the proof.  $\square$

## 4 Proximity Algorithms

In this section, we present proximity algorithms for solving the model (3.1) based on the characterizations of solutions given in the previous section, and establish their convergence results.

According to Proposition 5, a solution of the model (3.1) is characterized by the fixed-point equations (3.4)-(3.6). Based on these fixed-point equations, we can naturally propose an iterative algorithm that generates three sequences  $\{u^k : k \in \mathbb{N}\}$ ,  $\{v^k : k \in \mathbb{N}\}$ , and  $\{b^k : k \in \mathbb{N}\}$ , from arbitrary initial vectors  $u^0 \in \mathbb{R}^d$ ,  $v^0 \in \mathbb{R}^m$ , and  $b^0 \in \mathbb{R}^m$ . Here  $\mathbb{N}$  is the set of all natural numbers. This iterative procedure is stated as follows:

$$\begin{cases} u^{k+1} = x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( \left( I - \frac{\sigma}{\gamma}W^T W \right) u^k - \frac{\sigma}{\gamma}W^T (b^k - v^k) - x \right), \\ v^{k+1} = \text{prox}_{\frac{1}{\sigma}R}(b^k + Wu^{k+1}), \\ b^{k+1} = b^k + Wu^{k+1} - v^{k+1}. \end{cases} \quad (4.1)$$

The following theorem establishes the convergence result of the proposed iterative scheme (4.1). Its proof is omitted here since it follows directly from that of Theorem 3.5 in [17].

**Theorem 1.** *Let  $x$  be in  $\mathbb{R}^d$ , let  $\lambda$  be a positive number, let  $F \in \Gamma_0(\mathbb{R}^d)$ , let  $W$  be a  $m \times d$  matrix, and let  $R \in \Gamma_0(\mathbb{R}^m)$ . If the positive numbers  $\gamma$  and  $\sigma$  are chosen to satisfy*

$$\frac{\sigma}{\gamma} < \frac{1}{\|W\|_2^2} \quad (4.2)$$

then the sequence  $\{u^k : k \in \mathbb{N}\}$  generated by the iterative scheme (4.1) converges to a solution of model (3.1).

Implementing the iterative procedure (4.1) requires the availability of two proximity operators  $\text{prox}_{\frac{\lambda}{\gamma}F}$  and  $\text{prox}_{\frac{1}{\beta}R}$ . We shall show that for  $F$  being either  $\|\cdot\|_1$  or  $\text{env}_{\alpha\|\cdot\|_1}$  and  $R$  being either  $\varphi$  or  $\text{env}_{\beta\varphi}$ , where  $\varphi$  is given in (2.7), both  $\text{prox}_{\frac{\lambda}{\gamma}F}$  and  $\text{prox}_{\frac{1}{\beta}R}$  have explicit forms. Therefore, the scheme (4.1) is particularly suitable for solving models for “Case 1”-“Case 4” listed in Table 1. However, it is not suitable for solving the modified L1/TV models for “Case 5” and “Case 6” due to the fact that the proximity operator of  $\text{env}_{\beta\varphi \circ B}$  is lack of an explicit expression.

We first give the closed-form solutions of the proximity operators of  $\|\cdot\|_1$  and its Moreau envelope  $\text{env}_{\alpha\|\cdot\|_1}$ . For  $t > 0$  and  $u \in \mathbb{R}^d$ , set  $y := \text{prox}_{t\|\cdot\|_1}(u)$  and  $z := \text{prox}_{t\text{env}_{\alpha\|\cdot\|_1}}(u)$ . Then  $y_i$  and  $z_i$ , the  $i$ -th component of  $y$  and  $z$ , are given by

$$y_i = \max\{|u_i| - t, 0\} \cdot \text{sign}(u_i) \quad \text{and} \quad z_i = \begin{cases} u_i - t, & \text{if } u_i > t + \alpha, \\ \frac{\alpha}{\alpha+t}u_i, & \text{if } |u_i| \leq t + \alpha, \\ u_i + t, & \text{if } u_i < -t - \alpha, \end{cases} \quad (4.3)$$

respectively, for  $i = 1, 2, \dots, d$ .

The proximity operators of the function  $\varphi$  defined by (2.7) and its Moreau envelope  $\text{env}_{\beta\varphi}$  have closed-form solutions as well. For  $t > 0$  and  $b \in \mathbb{R}^{2d}$ , we denote  $y := \text{prox}_{t\varphi}(b)$  and  $z := \text{prox}_{t\text{env}_{\beta\|\cdot\|_2}}(b)$ . Write  $\tilde{b}_i = [b_i, b_{d+i}]^\top$ . Then for  $i = 1, 2, \dots, d$  we have that

$$\begin{bmatrix} y_i \\ y_{d+i} \end{bmatrix} = \max\left\{\|\tilde{b}_i\|_2 - t, 0\right\} \frac{\tilde{b}_i}{\|\tilde{b}_i\|_2} \quad \text{and} \quad \begin{bmatrix} z_i \\ z_{d+i} \end{bmatrix} = \begin{cases} \frac{\beta}{\beta+t}\tilde{b}_i, & \text{if } \|\tilde{b}_i\|_2 \leq \beta + t, \\ \left(1 - \frac{t}{\|\tilde{b}_i\|_2}\right)\tilde{b}_i, & \text{otherwise.} \end{cases} \quad (4.4)$$

From equations in (4.3), we notice that both the proximity operators of  $\ell^1$ -norm and  $\text{env}_{\alpha\|\cdot\|_1}$  can be performed component-wise. This allows us to exploit a Gauss-Seidel strategy to implement the first equation in the iterative scheme (4.1) for models “Case 1”-“Case 4”. For these four cases, the matrix  $W$  is always equal to  $B$ . Hence, in view of the first equation of (4.1), the  $i$ -th entry  $u_i^k$  and those from its neighbor have an impact to the  $i$ -th entry  $u_i^{k+1}$  of  $u^{k+1}$ . By  $b_U^k$  we denote the upper half of the vector  $b^k$  and  $b_L^k$  its lower half of  $b^k$ . Likewise, we define  $v_U^k$  and  $v_L^k$ . For convenience of exposition, we shall view all vectors  $u^k, u^{k+1}, b_L^k, b_U^k, v_L^k, v_U^k$  in the iterative scheme (4.1) as  $q \times q$  matrices for the time being. The component-wise expression of the first equation in the iterative scheme (4.1) for the interior pixel at the  $(i, j)$ -th location can be rewritten as

$$\begin{aligned} u_{ij}^{k+1} &= x_{ij} + \text{prox}_{\frac{\lambda}{\gamma}f}\left(\left(1 - \frac{4\sigma}{\gamma}\right)u_{ij}^k + \frac{\sigma}{\gamma}(u_{i-1,j}^k + u_{i+1,j}^k + u_{i,j-1}^k + u_{i,j+1}^k) \right. \\ &\quad \left. - \frac{\sigma}{\gamma}((b_U^k)_{ij} - (b_U^k)_{i,j+1} + (b_L^k)_{ij} - (b_L^k)_{i+1,j}) \right. \\ &\quad \left. + \frac{\sigma}{\gamma}((v_U^k)_{ij} - (v_U^k)_{i,j+1} + (v_L^k)_{ij} - (v_L^k)_{i+1,j}) - x_{ij}\right), \end{aligned} \quad (4.5)$$

where  $f$  is either the  $|\cdot|$  or the  $\text{env}_{\alpha|\cdot|}$  depending on whether  $F$  is the  $\ell^1$ -norm or its Moreau envelope  $\text{env}_{\alpha\|\cdot\|_1}$ . Similar expressions can be derived for boundary pixels. The proximity operator of  $f$  has been explicitly given in equations (4.3). It can be seen from equation (4.5) that the pixel  $u_{i,j}^k$  and its four neighboring pixels  $u_{i-1,j}^k, u_{i,j-1}^k, u_{i+1,j}^k, u_{i,j+1}^k$  all contribute to the updated pixel  $u_{i,j}^{k+1}$ . However, in the iterative scheme (4.1), one does not use the most recently available information when computing  $u_i^{k+1}$ . For example, if the order for updating pixels at the  $k$ -th iteration to obtain

$u^{k+1}$  at the  $(k+1)$ -th iteration accords to an order of top-to-bottom and column by column,  $u_{i-1,j}^k$  and  $u_{i,j-1}^k$  are used in the calculation of  $u_{i,j}^{k+1}$  even though  $u_{i-1,j}^{k+1}$  and  $u_{i,j-1}^{k+1}$  have already been known. Therefore, following the idea of the Gauss-Seidel iteration, we revise the calculation of the pixel  $u_{i,j}^{k+1}$  so that we always use the most current estimations of the pixels  $u_{i-1,j}$  and  $u_{i,j-1}$ . That is,  $u_{i,j}^{k+1}$  can be computed by

$$\begin{aligned} u_{ij}^{k+1} = & x_{ij} + \text{prox}_{\frac{\lambda}{\gamma}f} \left( \left(1 - \frac{4\sigma}{\gamma}\right)u_{ij}^k + \frac{\sigma}{\gamma}(u_{i-1,j}^{k+1} + u_{i+1,j}^k + u_{i,j-1}^{k+1} + u_{i,j+1}^k) \right. \\ & - \frac{\sigma}{\gamma}((b_U^k)_{ij} - (b_U^k)_{i,j+1} + (b_L^k)_{ij} - (b_L^k)_{i+1,j}) \\ & \left. + \frac{\sigma}{\gamma}((v_U^k)_{ij} - (v_U^k)_{i,j+1} + (v_L^k)_{ij} - (v_L^k)_{i+1,j}) - x_{ij} \right). \end{aligned} \quad (4.6)$$

This modification will not increase any computational complexity compared to equation (4.5) and is expected to speed up the convergence. With such a modification in the first equation of iterative scheme (4.1), we have Algorithm 1 which can be used for solving the models of Cases 1-4. Note that we could choose  $\frac{\sigma}{\gamma} < \frac{1}{8}$  because the norm  $\|B\|_2^2$  is always less than 8 (see, e.g., [18]).

---

**Algorithm 1** Proximity algorithm for Case 1 - Case 4 accelerated by GS iteration

---

Given: A noisy image  $x$  in  $\mathbb{R}^d$ ;  $\lambda > 0$ ,  $\beta > 0$ ,  $\gamma > 0$ ,  $\sigma > 0$  such that  $\frac{\sigma}{\gamma} < \frac{1}{8}$

Initialization:  $u^0 = x$ ,  $v^0 = 0$ ,  $b^0 = 0$

**repeat**

(a) Update the components of  $u^{k+1}$  according to equation (4.6)

(b)  $v^{k+1} \leftarrow \text{prox}_{\frac{1}{\sigma}R}(b^k + Bu^{k+1})$

(c)  $b^{k+1} \leftarrow b^k + Bu^{k+1} - v^{k+1}$

**until**  $u^{k+1}$  converges or satisfies a stopping criteria.

---

To close this section, we point out the relation between our proposed iterative algorithm (4.1) and the CPPD method proposed in [7]. The CPPD method is a primal-dual based algorithm developed for solving the associated saddle-point formulation of a minimization problem. The saddle-point formulation of the model (3.1) has the form of

$$\min_u \max_s \{ \lambda F(u - x) + \langle s, Wu \rangle - R^*(s) \} \quad (4.7)$$

The iterative scheme of the CPPD algorithm for solving model (4.7) is

$$\begin{cases} s^{k+1} = \arg \min_s \{ R^*(s) + \frac{1}{2\delta} \|s - s^k - \delta W \bar{u}^k\|_2^2 \}, \\ u^{k+1} = \arg \min_u \{ \lambda F(u - x) + \frac{1}{2\tau} \|u - u^k + \tau W^\top s^{k+1}\|_2^2 \}, \\ \bar{u}^{k+1} = 2u^{k+1} - u^k, \end{cases} \quad (4.8)$$

where  $\delta, \tau$  are two positive numbers,  $R^*$  is the convex conjugate function of  $R$ .

We rewrite the iterative scheme in (4.1) to establish the relationship between the CPPD and our algorithm. From the third step of the iterative scheme (4.1), we can get  $v^k = Wu^k + b^{k-1} - b^k$ . Substituting this equation into the first equation in (4.1), and combining the second and third equations in (4.1) together, the iterative scheme (4.1) can be rewritten as

$$\begin{cases} u^{k+1} = x + \text{prox}_{\frac{\lambda}{\gamma}F}(u^k - \frac{\sigma}{\gamma}W^\top(2b^k - b^{k-1}) - x), \\ b^{k+1} = (I - \text{prox}_{\frac{1}{\sigma}R})(b^k + Wu^{k+1}). \end{cases} \quad (4.9)$$

Recall that the Moreau decomposition  $I = \text{prox}_{tR} + t \text{prox}_{\frac{1}{t}R^*} \circ \frac{1}{t}I$  for any  $t > 0$ . Let  $s^k := \sigma b^k$ , then the iterative scheme in (4.9) has the equivalent form of

$$\begin{cases} u^{k+1} = \arg \min_u \{\lambda F(u - x) + \frac{\gamma}{2} \|u - u^k + \frac{1}{\gamma}W^\top \bar{s}^k\|_2^2\}, \\ s^{k+1} = \arg \min_s \{R^*(s) + \frac{1}{2\sigma} \|s - s^k - \sigma Wu^{k+1}\|_2^2\}, \\ \bar{s}^{k+1} = 2s^{k+1} - s^k, \end{cases} \quad (4.10)$$

which is exactly the iterative scheme of CPPD applied to the following saddle-point formulation

$$\min_s \max_u \{R^*(s) + \langle u, -W^\top s \rangle - \lambda F(u - x)\} \quad (4.11)$$

of the dual problem of model (3.1). Therefore, our proposed iterative algorithm (4.1) for solving the primal problem (3.1) is algebraically equivalent to the CPPD algorithm applied to the saddle-point formulation of its dual problem. But they are not algorithmically equivalent. As pointed out in [17], the component-wise Gauss-Seidel iteration applied to CPPD will not result in acceleration of convergence.

## 5 Special Proximity Algorithms for Model (3.1) with a Differentiable $R$

This section is devoted to present proximity algorithms for model (3.1) when the convex function  $R$  is differentiable. The last four modified L1/TV models for “Case 3”-“Case 6” summarized in Table 1 possess this property.

Recall that Proposition 6 presents a characterization to solutions of model (3.1) when the function  $R$  in the regularization term is differentiable. From the fixed-point equation (3.9), for any initial value  $u^0 \in \mathbb{R}^d$ , we propose the following iterative scheme for finding a solution of the model (3.1)

$$u^{k+1} = x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( u^k - \frac{1}{\gamma}W^\top \nabla R(Wu^k) - x \right). \quad (5.1)$$

We remark that the fixed-point equation (3.9) can be viewed as a special case of the proximal forward-backward splitting described in [12]. Furthermore, if the gradient  $\nabla R$  is Lipschitz continuous with Lipschitz constant  $L$ , that is

$$\|\nabla R(y) - \nabla R(z)\|_2 \leq L\|y - z\|_2,$$

for all  $y, z \in \mathbb{R}^m$  and if  $\gamma$  is chosen to satisfy

$$\frac{1}{\gamma} < \frac{2}{L\|W\|_2^2},$$

then it was proved in [12] that for any initial guess  $u^0 \in \mathbb{R}^d$ , the sequence  $\{u^k : k \in \mathbb{N}\}$  generated from the iterative scheme (5.1) converges to a fixed-point of equation (3.9), which, by Proposition 6, is a solution of model (3.1).

The iterative scheme (5.1) can be applied to the models “Case 3”-“Case 6”. Implementing the iterative procedure (5.1) requires the availability of the proximity operator  $\text{prox}_{\frac{\lambda}{\gamma}F}$  and the

gradient  $\nabla R$ . Since the function  $F$  is either the  $\ell^1$ -norm or the Moreau envelope  $\text{env}_{\alpha\|\cdot\|_1}$ , its proximity operator  $\text{prox}_{\frac{\lambda}{\gamma}F}$  can be explicitly computed according to equations (4.3). We know from Table 2 that  $R$  is the Moreau envelope of either  $\varphi$  or  $\varphi \circ B$ . If  $R$  is  $\text{env}_{\beta\varphi}$  (i.e. Case 3 and Case 4), by equation (2.10) we have that

$$\nabla \text{env}_{\beta\varphi} = \frac{1}{\beta}(I - \text{prox}_{\beta\varphi}). \quad (5.2)$$

Hence, the iterative algorithm (5.1) applied to Case 3 and Case 4 has the form of

$$u^{k+1} = x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( u^k - \frac{1}{\gamma\beta} B^\top (I - \text{prox}_{\beta\varphi})(Bu^k) - x \right). \quad (5.3)$$

Since the proximity operator  $\text{prox}_{\beta\varphi}$  can also be explicitly computed according to equation (4.4), the calculation of  $u^{k+1}$  in the iterative scheme (5.3) has a closed-form. Furthermore, due to the non-expansiveness of the proximity operator [2], the gradient of  $\text{env}_{\beta\varphi}$  in equation (5.2) is Lipschitz continuous with Lipschitz constant  $\frac{1}{\beta}$ . That is, if the parameter  $\gamma$  is chosen to satisfy

$$\frac{1}{\gamma\beta} < \frac{1}{4},$$

then the sequence  $\{u^k : k \in \mathbb{N}\}$  generated from the iterative scheme (5.3) for any initial value  $u^0 \in \mathbb{R}^d$  converges to a solution of the model corresponding to Case 3 or Case 4. The flow of finding this solution is described in Algorithm 2.

---

**Algorithm 2** Proximity algorithm for Case 3 and Case 4

---

Given: A noisy image  $x$  in  $\mathbb{R}^d$ ;  $\lambda > 0$ ,  $\beta > 0$ ,  $\gamma > 0$  such that  $\frac{1}{\gamma\beta} < \frac{1}{4}$

Initialization:  $u^0 = x$

**repeat**

(a)  $u^{k+1} = x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( u^k - \frac{1}{\gamma\beta} B^\top (I - \text{prox}_{\beta\varphi})(Bu^k) - x \right)$

**until**  $u^{k+1}$  converges or satisfies a stopping criteria.

---

If  $R$  is  $\text{env}_{\beta\varphi \circ B}$  (Case 5 or Case 6), by equation (2.10) we have that  $\nabla \text{env}_{\beta\varphi \circ B} = \frac{1}{\beta}(I - \text{prox}_{\beta\varphi \circ B})$ , whose gradient is Lipschitz continuous with Lipschitz constant  $\frac{1}{\beta}$ . Hence the iterative algorithm (5.1) applied to Case 5 or Case 6 has the form of

$$u^{k+1} = x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( \frac{\beta\gamma - 1}{\beta\gamma} u^k + \frac{1}{\beta\gamma} \text{prox}_{\beta\varphi \circ B}(u^k) \right). \quad (5.4)$$

The sequence  $\{u^k : k \in \mathbb{N}\}$  generated from iterative scheme (5.4) converges to a solution of Case 5 or Case 6 if the number  $\gamma$  is chosen to satisfy  $\frac{1}{\gamma\beta} < 2$ .

In the iterative scheme (5.4), the proximity operator of  $\varphi \circ B$  at  $u^k$  needs to be evaluated at each iteration. Although this proximity operator has no closed-form, algorithms proposed in [6, 14, 18] can be directly applied to evaluate the proximity operator  $\text{prox}_{\beta\varphi \circ B}$ . In this paper, we apply the approach proposed in [18] to compute  $\text{prox}_{\beta\varphi \circ B}(u^k)$ . In [18], it was shown that

$$\text{prox}_{\beta\varphi \circ B}(u^k) = u^k - \delta\beta B^\top b^k, \quad (5.5)$$

where  $\delta$  is positive and  $b^k$  is a solution of the fixed-point equation

$$b = \left( I - \text{prox}_{\frac{1}{8}\varphi} \right) (Bu^k + (I - \delta\beta BB^\top)b). \quad (5.6)$$

It was proved in [18] that if we choose the number  $\delta > 0$  such that  $\delta\beta < \frac{1}{4}$ , then for any initial guess  $b^{k,1} \in \mathbb{R}^{2d}$  the sequence  $\{b^{k,j} : j \in \mathbb{N}\}$  generated from

$$b^{k,j+1} = \left( I - \text{prox}_{\frac{1}{8}\varphi} \right) (Bu^k + (I - \delta\beta BB^\top)b^{k,j}) \quad (5.7)$$

converges to a fixed-point  $b^k$  of equation (5.6). As pointed out in [18], a Gauss-Seidel strategy can be used to accelerate the rate of convergence of the iterative scheme (5.7). With these preparation, the detail of the implementation for Case 5 or Case 6 is described in Algorithm 3.

---

**Algorithm 3** Proximity algorithm for Case 5 and Case 6

---

Given: A noisy image  $x$  in  $\mathbb{R}^d$ ;  $\lambda > 0$ ,  $\beta > 0$ ,  $\gamma > 0$ ,  $\delta > 0$  such that  $\frac{1}{\gamma\beta} < 2$ ,  $\delta\beta < \frac{1}{4}$

Initialization:  $u^0 = x$

**repeat**

(a) Initialization  $b^{k,1} = 0$

for  $j$  from 1 to  $n - 1$

Computes  $b^{k,j+1}$  from  $b^{k,j}$  by the Gauss-Seidel variation of iterative scheme (5.7)

end

(b)  $u^{k+\frac{1}{2}} = u^k - \delta\beta B^\top b^{k,n}$

(c)  $u^{k+1} = x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( \frac{\beta\gamma-1}{\beta\gamma}u^k + \frac{1}{\beta\gamma}u^{k+\frac{1}{2}} - x \right)$

**until**  $u^{k+1}$  converges or satisfies a stopping criteria.

---

We would like to elaborate a connection between Algorithm 2 and Algorithm 3. We shall show that if we set  $n = 2$  in Algorithm 3, that is, we use

$$u^{k+\frac{1}{2}} := u^k - \delta\beta B^\top b^{k,2}$$

as an approximate solution of  $\text{prox}_{\beta\varphi \circ B}(u^k)$ , then it reduces to Algorithm 2 for a specific model. Now, we give a detailed explanation for this statement. Since the initial value  $b^{k,1} = 0$ , we have that  $b^{k,2} = \left( I - \text{prox}_{\frac{1}{8}\varphi} \right) (Bu^k)$  which implies that

$$u^{k+\frac{1}{2}} = u^k - \delta\beta B^\top \left( I - \text{prox}_{\frac{1}{8}\varphi} \right) (Bu^k). \quad (5.8)$$

Substituting equation (5.8) into step (c) of Algorithm 3, we get

$$\begin{aligned} u^{k+1} &= x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( \frac{\beta\gamma-1}{\beta\gamma}u^k + \frac{1}{\beta\gamma}(u^k - \delta\beta B^\top (I - \text{prox}_{\frac{1}{8}\varphi})(Bu^k)) - x \right) \\ &= x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( u^k - \frac{\delta}{\gamma}B^\top (I - \text{prox}_{\frac{1}{8}\varphi})(Bu^k) - x \right). \end{aligned} \quad (5.9)$$

By comparing with (5.4), we know that if  $\frac{\delta}{\gamma} < \frac{1}{4}$ , the sequence  $\{u^k : k \in \mathbb{N}\}$  generated from the iterative scheme (5.9) converges to a solution of the problem

$$\arg \min_u \left\{ \lambda F(u - x) + \text{env}_{\frac{1}{\delta}\varphi}(Bu) \right\},$$

which is exactly the situation of Case 3 or Case 4 depending on whether  $F$  is the  $\ell^1$ -norm or its envelope  $\text{env}_{\alpha\|\cdot\|_1}$ .

Finally, we would like to emphasize that both Algorithms 2 and 3 can be accelerated by employing FISTA technique [3]. Recall that the function  $R$  in models for “Case 3”-“Case 6” is the Moreau envelope of either  $\varphi$  or  $\varphi \circ B$ . Hence, the function  $R$  has a Lipschitz continuous gradient. This property allows us to accelerate the convergence of the iterative scheme (5.1) by using the FISTA in [3]. To this end, we write the procedures of Algorithms 2 and 3 into a unified form

$$u^{k+1} = P(u^k), \quad (5.10)$$

where  $P$  denotes step (a) in Algorithm 2 or steps (a)-(c) in Algorithm 3. However, the FISTA applies  $P$  on a point  $y^k$  which uses a very specific linear combination of the previous two points  $\{u^k, u^{k-1}\}$ . The detailed iterative scheme of the FISTA is described as

$$\begin{aligned} u^k &= P(y^k), \\ t^{k+1} &= \frac{\sqrt{1+4(t^k)^2+1}}{2}, \\ y^{k+1} &= u^k + \frac{t^k-1}{t^{k+1}}(u^k - u^{k-1}), \end{aligned} \quad (5.11)$$

where  $t^1 = 1$ . Clearly the main computational effort in both equations (5.10) and (5.11) remains the same. The required additional computation for the last two steps of equation (5.11) is clearly marginal. The procedures of Algorithm 2 and Algorithm 3 accelerated by FISTA are described in Algorithm 4 and Algorithm 5, respectively.

---

**Algorithm 4** Proximity algorithm for Case 3 or Case 4 accelerated by FISTA

---

Given: A noisy image  $x$  in  $\mathbb{R}^d$ ;  $\lambda > 0$ ,  $\beta > 0$ ,  $\gamma > 0$  such that  $\frac{1}{\gamma\beta} < \frac{1}{4}$

Initialization:  $y^1 = u^0 = x$ ,  $u^{-1} = 0$ ,  $t^1 = 1$

**repeat**

$$(a) \quad u^k = x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( y^k - \frac{1}{\gamma\beta} B^\top (I - \text{prox}_{\beta\varphi})(By^k) - x \right)$$

$$(b) \quad t^{k+1} = \frac{\sqrt{1+4(t^k)^2+1}}{2}$$

$$(c) \quad y^{k+1} = u^k + \frac{t^k-1}{t^{k+1}}(u^k - u^{k-1})$$

**until**  $u^k$  converges or satisfies a stopping criteria.

---

We note that step (a) in Algorithm 4 can be written as

$$\begin{aligned} y^{k+1} &= x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( \left( I - \frac{1}{\gamma\beta} B^\top B \right) y^k + \frac{1}{\gamma\beta} B^\top \text{prox}_{\beta\varphi}(By^k) - x \right), \\ u^k &= y^{k+1}. \end{aligned}$$

By doing so, in a way similar to what we have done for the first equation in (4.1), a Gauss-Seidel iteration can be applied. To do this, we let

$$w^k := \text{prox}_{\beta\varphi}(By^k)$$

---

**Algorithm 5** Proximity algorithm for Case 5 or Case 6 accelerated by FISTA

---

Given: A noisy image  $x$  in  $\mathbb{R}^d$ ;  $\lambda > 0$ ,  $\beta > 0$ ,  $\gamma > 0$ ,  $\delta > 0$  such that  $\frac{1}{\gamma\beta} < 2$ ,  $\delta\beta < \frac{1}{4}$

Initialization:  $y^1 = u^0 = x$ ,  $u^{-1} = 0$ ,  $t^1 = 1$

**repeat**

(a) Initialization  $b^{k,1} = 0$

for  $j$  from 1 to  $n - 1$

Computes  $b^{k,j+1}$  from  $b^{k,j}$  by the Gauss-Seidel variation of iterative scheme (5.7)

end

(b)  $y^{k+\frac{1}{2}} = y^k - \delta\beta B^\top b^{k,j+1}$

(c)  $u^k = x + \text{prox}_{\frac{\lambda}{\gamma}F} \left( \frac{\beta\gamma-1}{\beta\gamma} y^k + \frac{1}{\beta\gamma} y^{k+\frac{1}{2}} \right)$

(d)  $t^{k+1} = \frac{\sqrt{1+4(t^k)^2+1}}{2}$

(e)  $y^{k+1} = u^k + \frac{t^k-1}{t^{k+1}}(u^k - u^{k-1})$

**until**  $u^{k+1}$  converges or satisfies a stopping criteria.

---

and denote by  $w_U^k$  and  $w_L^k$  the upper and the lower half of the vector  $w^k$ , respectively. We view all vectors  $x$ ,  $y^k$ ,  $y^{k+1}$ ,  $w_U^k$  and  $w_L^k$  in equation (5.12) as  $q \times q$  matrices momentarily. Then, the interior pixel  $y_{ij}^{k+1}$  of  $y^{k+1}$  is updated in the following way

$$\begin{aligned} y_{ij}^{k+1} = & x_{ij} + \text{prox}_{\frac{\lambda}{\gamma}|\cdot|_1} \left( \left(1 - \frac{4}{\gamma\beta}\right) y_{ij}^k + \frac{1}{\gamma\beta} (y_{i-1,j}^{k+1} + y_{i+1,j}^k + y_{i,j-1}^{k+1} + y_{i,j+1}^k) \right. \\ & \left. + \frac{1}{\gamma\beta} ((w_U^k)_{ij} - (w_U^k)_{i,j+1} + (w_L^k)_{ij} - (w_L^k)_{i+1,j}) - x_{ij} \right). \end{aligned} \quad (5.12)$$

Similar calculation for the boundary pixels of  $y^{k+1}$  can be proceeded. The proximity algorithm for solving Case 3 or Case 4 accelerated by both the FISTA and the Gauss-Seidel iteration is described in Algorithm 6.

---

**Algorithm 6** Proximity algorithm for Case 3 or Case 4 accelerated by FISTA and GS iteration

---

Given: A noisy image  $x$  in  $\mathbb{R}^d$ ;  $\lambda > 0$ ,  $\beta > 0$ ,  $\gamma > 0$  such that  $\frac{1}{\gamma\beta} < \frac{1}{4}$

Initialization:  $y^1 = u^0 = x$ ,  $u^{-1} = 0$ ,  $t^1 = 1$

**repeat**

(a) Update the components of  $y^{k+1}$  according to equation (5.12).

(b)  $u^k = y^{k+1}$

(b)  $t^{k+1} = \frac{\sqrt{1+4(t^k)^2+1}}{2}$

(c)  $y^{k+1} = u^k + \frac{t^k-1}{t^{k+1}}(u^k - u^{k-1})$

**until**  $u^k$  converges or satisfies a stopping criteria.

---

We close this section with Table 3. This table summarizes the proposed algorithms and acceleration techniques that could be applied for solving the L1/TV model and its variants.

Table 3: The algorithms towards solving the L1/TV models

Case	Algorithms for choice	Acceleration
1	Algorithm 1	GS
2	Algorithm 1	GS
3	Algorithms 1, 4, 6	FISTA & GS
4	Algorithms 1, 4, 6	FISTA & GS
5	Algorithm 5	FISTA
6	Algorithm 5	FISTA

## 6 Numerical Experiments

In this section we present numerical results to demonstrate the difference of various modified L1/TV models and compare the performance of the proposed algorithms on restoring images from those corrupted by salt-pepper noise. We choose the ‘‘Cameraman’’ and ‘‘Lena’’ as our test images. Both of the images are of size  $256 \times 256$ . The quality of the restored image is evaluated in terms of the peak-signal-to-noise ratio (PSNR) defined by

$$\text{PSNR} := 10 \log_{10} \frac{255^2 d}{\|u - \tilde{u}\|_2^2} (\text{dB}),$$

where  $\tilde{u}$  denotes the restored image with respect to the original image  $u \in \mathbb{R}^d$ . Each PSNR-value reported in all tables in this section is the average over five runs. For all numerical experiments, the regularization parameters  $\lambda$  are determined experimentally for the restored images to achieve the best possible PSNR-values. 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 first experiment, by fixing the regularization term, we test whether replacing the fidelity term  $\|\cdot\|_1$  with  $\text{env}_{\alpha\|\cdot\|_1}$  can yield better performance in terms of values of PSNR. To this end, we compare the performance of model ‘‘Case 1’’ with that of ‘‘Case 2’’. Both the models in these two cases use the total variation as the regularization term. To see the effect of the smooth parameter  $\alpha$  in ‘‘Case 2’’, we set five different values for  $\alpha$  (1, 2, 3, 5, and 10). The numerical results are listed in Table 4. All of the PSNR-values are obtained by implementing Algorithm 1 with large enough iteration numbers which can guarantee the convergence. We observe that the PSNR-values of the restored images by ‘‘Case 1’’ are higher than those by ‘‘Case 2’’. Furthermore, the PSNR-values of the restored images by ‘‘Case 2’’ decrease as the smooth parameter  $\alpha$  increases. This is essentially because

$$\lim_{\alpha \rightarrow 0} \text{env}_{\alpha\|\cdot\|_1}(u - x) = \|u - x\|_1, \quad \text{for all } u \in \mathbb{R}^d$$

and the use of  $\|\cdot - x\|_1$  is based on the statistical analysis of the salt-pepper noise in Section 2. This experiment confirms that the  $\ell^1$ -norm fidelity term should not be changed for the purpose of removing impulsive noise.

In our second experiment the fidelity term is fixed to be  $\|\cdot\|_1$ . We shall compare the total variation regularization term with its two smoothed versions  $\text{env}_{\beta\varphi}(Bu)$  and  $\text{env}_{\beta\circ B}(u)$ . To this end, we compare the performance of model ‘‘Case 1’’ with those of ‘‘Case 3’’ and ‘‘Case 5’’. Similarly, we set different values for the smooth parameter  $\beta$  in model ‘‘Case 3’’ and model ‘‘Case 5’’ to see its effect on restored images. We solve model ‘‘Case 3’’ by Algorithm 1 with large enough iteration numbers to guarantee its convergence. Model ‘‘Case 5’’ is solved by Algorithm 5 in which the

Table 4: The summary of the restoration results of the models “Case 1” and “Case 2”.

Case	Cameraman			Lena		
	10%	30%	50%	10%	30%	50%
1	<b>28.83</b>	<b>24.74</b>	<b>22.55</b>	<b>31.01</b>	<b>27.42</b>	<b>24.60</b>
2 with $\alpha = 1$	28.79	24.72	22.55	30.94	27.38	24.58
2 with $\alpha = 2$	28.74	24.70	22.54	30.86	27.33	24.55
2 with $\alpha = 3$	28.68	24.68	22.53	30.78	27.27	24.52
2 with $\alpha = 5$	28.60	24.61	22.45	30.70	27.19	24.43
2 with $\alpha = 10$	28.41	24.44	22.23	30.51	27.05	24.30

parameter  $n$  is chosen as 50. It was shown numerically in [18] that this value for  $n$  is accurate enough to evaluate  $\text{prox}_{\varphi \circ B}(u)$ . We summarize the numerical results in Table 5. Unlike  $\varphi$  and  $\text{env}_{\beta\varphi}$ , the proximity operators of  $\varphi \circ B$  and  $\text{env}_{\beta\varphi \circ B}$  have no closed-forms which introduce more difficulty in numerical treatment of model “Case 5”. Moreover, model “Case 5” could not improve the quality of restored images in comparison with model “Case 1”. We therefore conclude that replacing the total variation  $\varphi \circ B$  by  $\text{env}_{\varphi \circ B}(u)$  is not a favorable choice. However, model “Case 3” with the smoothed regularization term  $\text{env}_{\beta\varphi}(Bu)$  can outperform model “Case 1” when the parameter  $\beta$  is properly selected (usually in the range of 10 to 20). This is mainly because this smoothed regularization term uses  $\text{env}_{\beta\varphi}$  rather than  $\varphi$  to penalize the gradients of the image. For  $z \in \mathbb{R}^{2d}$ , we denote  $y := \text{env}_{\beta\varphi}(z)$  and let  $\tilde{z}_i := [z_i, z_{d+i}]^\top$ . Then for  $i = 1, 2, \dots, d$ , by the definition of  $\varphi$  in equation (2.7) and the closed-form solution of its proximity operator in equation (4.4), we have that

$$y_i = \begin{cases} \frac{1}{2\beta} \|\tilde{z}_i\|_2^2, & \|\tilde{z}_i\|_2 \leq \beta, \\ \|\tilde{z}_i\|_2 - \frac{\beta}{2}, & \|\tilde{z}_i\|_2 > \beta. \end{cases} \quad (6.1)$$

As can be seen from equation (6.1), the smoothed function  $\text{env}_{\beta\varphi}$  differs from  $\varphi$  by discriminating gradients with small magnitudes and penalizing them quadratically. As a result, it has an effect on smoothing small gradients of the restored image while keeping the sharp edges well. Consequently, the staircasing effects usually caused by the total variation term can be removed to some extent. The parameter  $\beta$  is used to specify the gradient value at which the regularization term  $\text{env}_{\beta\varphi}(Bu)$  switches from penalizing quadratically to behaving like the total variation. When  $\beta$  is too small, the regularization term  $\text{env}_{\beta\varphi}(Bu)$  performs like the total variation, while it gradually imitates the Tikhonov regularization with  $\beta$  increases. Hence, we recommend  $\text{env}_{\beta\varphi}(Bu)$  as the preferred regularization term in applications. The experiment results in Table 5 show that the best value for  $\beta$  is usually in the range of 10 to 20.

Based on the above numerical study for models with various fidelity and regularization terms, model “Case 3” is a favorable choice for impulsive noise removal. Since the function  $\text{env}_{\beta\varphi}$  in model “Case 3” is differentiable and its proximity operator has a closed-form, the model can be solved by the algorithms accelerated by either the Gauss-Seidel iteration or FISTA, or the both, presented in Section 4 and Section 5. A detailed comparison of the algorithms for impulsive noise removal is presented in the rest of this section.

We first validate the acceleration of the Gauss-Seidel iteration and FISTA. To see whether the Gauss-Seidel iteration can accelerate the convergence, we compare the performance of the iterative scheme given by (4.1), the CPPD algorithm in equation (4.8) and Algorithm 1 in restoring the “Cameraman” image corrupted by salt-pepper noise with the level of 30%. We select three different values 1, 10, and 20 for  $\beta$  to investigate its effect on the acceleration. The PSNR-values of the

Table 5: The summary of the restoration results of the models “Case 1”, “Case 3” and “Case 5”.

Case	Cameraman			Lena		
	10%	30%	50%	10%	30%	50%
1	28.83	24.74	22.55	31.01	27.42	24.60
3 with $\beta = 5$	28.93	24.86	22.66	31.09	27.62	24.79
3 with $\beta = 10$	<b>28.97</b>	24.95	<b>22.71</b>	<b>31.10</b>	<b>27.72</b>	<b>24.88</b>
3 with $\beta = 15$	28.94	<b>24.98</b>	<b>22.71</b>	30.98	27.71	24.87
3 with $\beta = 20$	28.84	24.96	22.65	30.74	27.59	24.77
3 with $\beta = 25$	28.70	24.88	22.53	30.41	27.37	24.57
3 with $\beta = 30$	28.55	24.74	22.36	28.99	27.06	24.30
3 with $\beta = 35$	28.34	24.57	22.15	29.52	27.69	23.97
5 with $\beta = 1$	28.83	24.72	22.55	30.92	27.36	24.42
5 with $\beta = 2$	28.77	24.69	22.54	30.71	27.20	24.31
5 with $\beta = 3$	28.65	24.63	22.48	30.43	27.03	24.16
5 with $\beta = 4$	28.50	24.53	22.42	30.09	26.83	23.99
5 with $\beta = 5$	28.31	24.41	22.33	29.72	26.59	23.81

restored images obtained by three approaches at each iteration are plotted in Figure 1. We observe that the proposed algorithm in equation (4.1) has performance similar to the CPPD algorithm. This does not surprise to us as these algorithms are closely related as discussed at the end of Section 4. However, applying the Gauss-Seidel iteration in Algorithm 1 can obviously improve the convergence rate and the restored image can be approximatively solved within 20 iterations by Algorithm 1. It can also be seen from Figure 1 that the parameter  $\beta$  does not affect much the rate of convergence of these three algorithms.

We then discuss the acceleration effects of the FISTA and Gauss-Seidel iteration on the Algorithm 2. To this end, we apply Algorithms 2, 4 and 6 to restore the same noisy image (“Cameraman” corrupted by salt-pepper noise with the noise level 30%) that we used above. Three different values 1, 10, and 20 are also selected for  $\beta$  to investigate its effect on the rate of convergence of the algorithms. Figure 2 shows the PSNR-values of the restored images obtained by three algorithms at each iteration. We can make two conclusions from the results in Figure 2. One is the Algorithm 4 using the FISTA can yield much faster convergence rate than Algorithm 2, and the use of Gauss-Seidel iteration in Algorithm 6 could further speed up the convergence of Algorithm 4. Secondly, the convergence rates of three algorithms heavily depend on the value of  $\beta$ , the larger we select for  $\beta$ , the faster convergence rate can be achieved.

As we already have seen that model “Case 3” can be solved by employing the proposed algorithms in equation (4.1) and Algorithm 2 which have the corresponding accelerated schemes described in Algorithm 1 and Algorithm 6. A natural question follows immediately: which accelerated scheme should we choose for solving model “Case 3”? To answer this question, we compare below the performance of Algorithm 1 and Algorithm 6 in details. The stopping criterion of both algorithms in this experiment is that the relative error between the successive iterates of the restored images should satisfy the following inequality  $\|u^{i+1} - u^i\|_2^2 / \|u^i\|_2^2 < 0.001$ , where  $u^i$  is the denoised image at the  $i$ -th iteration. We set six different values 1, 5, 10, 15, 20 and 25 for the parameter  $\beta$ . Considering restoring the “Cameraman” image corrupted by the salt-pepper noise with three different levels at 10%, 30% and 50%, we show in Figure 3 the running time consumed, the PSNR-values of the restored images and the needed iteration numbers of both algorithms. The first observation is that the convergence rates of both algorithms can be improved when increasing

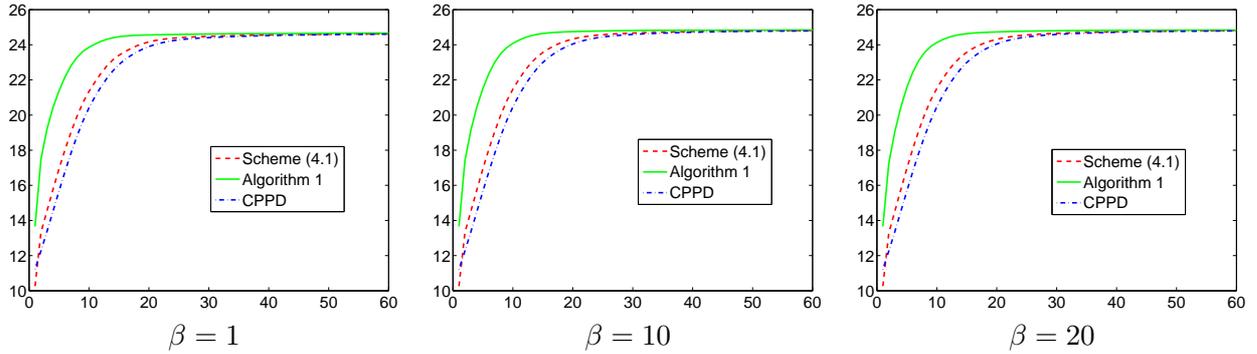


Figure 1: Acceleration results of applying Gauss-Seidel iteration to the algorithm in (4.1) when restoring a noisy “Cameraman” image corrupted by salt-pepper noise with the noise level 30%.

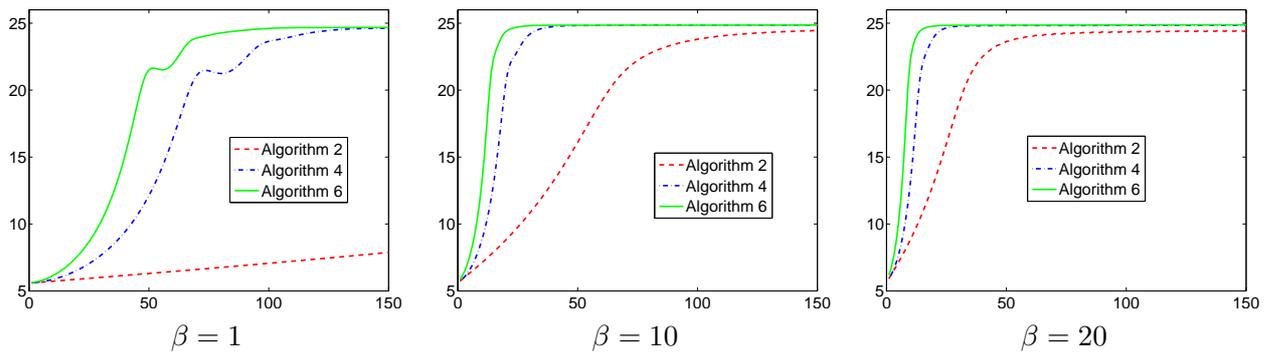


Figure 2: Acceleration results of applying FISTA and Gauss-Seidel iteration to Algorithm 2 when restoring noisy “Cameraman” image corrupted by salt-pepper noise with the noise level 30%.

the parameter  $\beta$ . However, the performance of Algorithm 6 is much more dependent on this parameter than that of Algorithm 1. Specifically, we prefer using Algorithm 1 when  $\beta$  is less than 10, while tend to like the Algorithm 6 when  $\beta$  becomes larger. We also observe that the best choice for the parameter  $\beta$  that can generate the restored image with the highest PSNR-value usually locates in the range 10 to 20. The higher the level of noise in an image is, the larger  $\beta$  will be preferred.

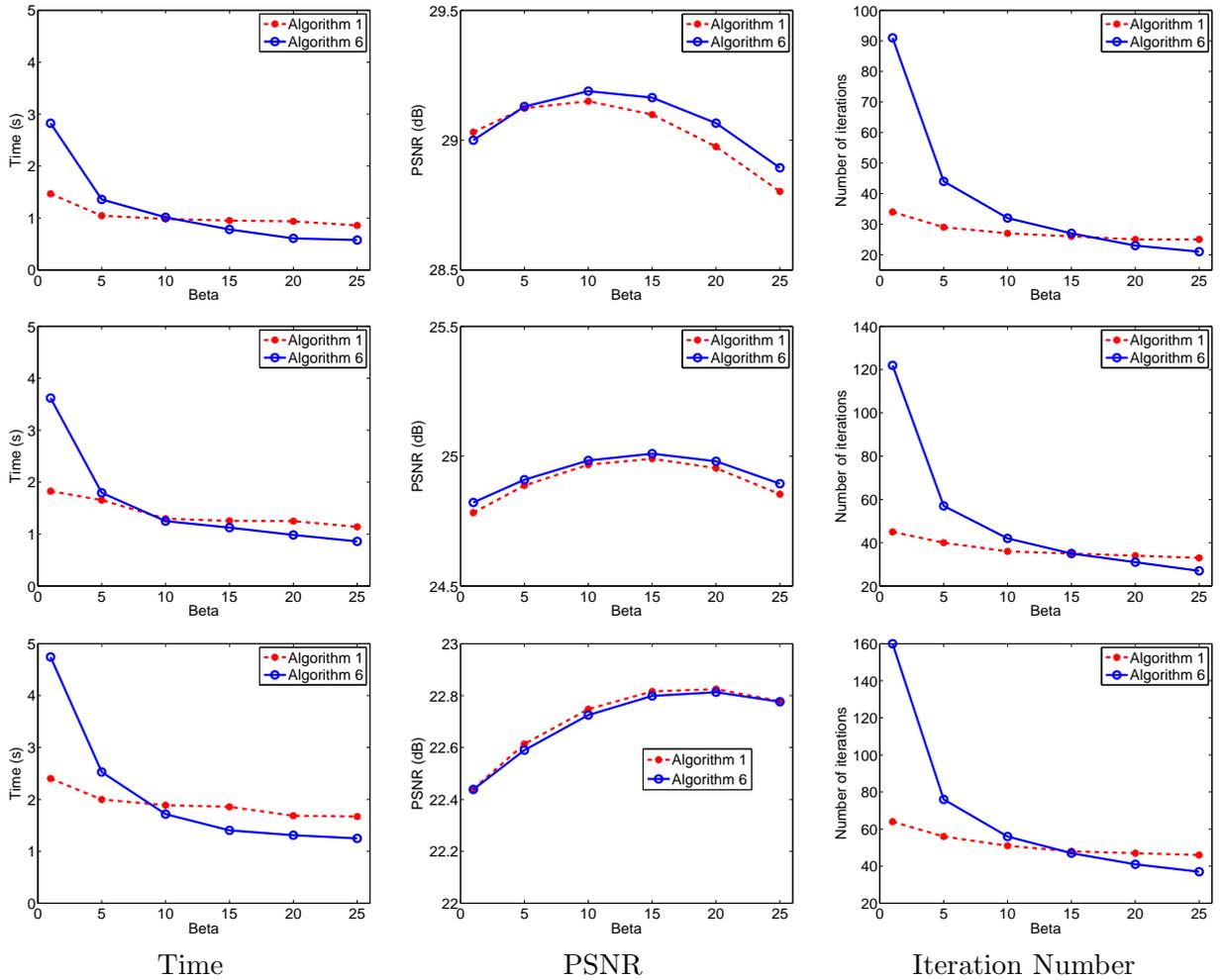


Figure 3: The numerical results of Algorithms 1 and 6 when restoring the “Cameraman” images corrupted by the salt-pepper noise with 10% (Row 1), 30% (Row 2), and 50% (Row 3) different noise levels.

## 7 Concluding remarks

We propose several modified L1/TV models by replacing the non-differentiable functions in the objective function of the original L1/TV model with their corresponding differentiable Moreau envelopes. Some existing approaches for the L1/TV model in the literature are intrinsically to solve one of our modified models. A unified proximity algorithm framework is developed for the numerical treatment of all models. The algorithms are motivated by the fixed-point equations characterizing the solutions of the models. The corresponding accelerated schemes that can speed up the convergence rates of the proposed algorithms are given. We compare the performance of

the modified models in the applications of impulsive noise removal, and single out a model as the best choice for this purpose. Two types of the proximity algorithms that can solve this model are compared in details, some recommendations on which algorithm should be used are made based on the numerical experiments.

## References

- [1] J. E. Aujol, G. Gilboa, T. Chan, and S. Osher. Structure-texture image decomposition-modeling, algorithms, and parameter selection. *International Journal of Computer Vision*, 67:111–136, 2006.
- [2] Heinz H. Bauschke and Patrick L. Combettes. *Convex analysis and Monotone Operator Theory in Hilbert Space*. Springer Press, New York, 2010.
- [3] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Science*, 2(1):183–202, 2009.
- [4] P. Bloomfield and W.L. Steiger. *Least Absolute Deviations: Theory, Applications, and Algorithms*. Birkhäuser Boston, Boston, 1983.
- [5] A. Bovik. *Handbook of Image and Video Processing*. Academic Press, San Diego, 2000.
- [6] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical imaging and vision*, 20(1):89–97, 2004.
- [7] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of Mathematical Imaging and Vision*, 40(1):120–145, 2011.
- [8] T.F. Chan and S. Esedoglu. Aspects of total variation regularized l1 function approximation. *SIAM Journal on Applied Mathematics*, 65:1817–1837, 2005.
- [9] R. Chartrand and V. Staneva. Total variation regularization of images corrupted by non-gaussian noise using a quasi-newton method. *IET Image Processing*, 2(6):295–303, 2008.
- [10] T. Chen, W. Yin, X.S. Zhou, D. Comaniciu, and T.S. Huang. Total variation models for variable lighting face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(9):1519–1524, 2006.
- [11] 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*, 32:1484–1505, 2010.
- [12] P. Combettes and V. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Model and Simulation*, 4:1168–1200, 2005.
- [13] Y. Dong, M. Hintermuller, and M. Neri. A primal-dual method for  $l^1$ -TV image denoising. *SIAM Journal on Imaging Sciences*, 2:1168–1189, 2009.
- [14] T. Goldstein and S. Osher. The split bregman method for l1-regularized problems. *SIAM Journal on Imaging Science*, 2(2):323–343, 2009.
- [15] X. Guo, F. Li, and M. Ng. A fast  $\ell^1$ -TV algorithm for image restoration. *SIAM Journal of Scientific Computing*, 31:2322–2341, 2009.

- [16] B. He and X. Yuan. Convergence analysis of primal-dual algorithms for a saddle-point problem: from contraction perspective. *SIAM Journal on Imaging Sciences*, 5:119–149, 2012.
- [17] Q. Li, C.A. Micchelli, L. Shen, and Y. Xu. A proximity algorithm accelerated by Gauss-Seidel iterations for L1/TV denoising models. *Inverse Problems*, 28:095033 20pp, 2012.
- [18] Charles A. Micchelli, Lixin Shen, and Yuesheng Xu. Proximity algorithms for image models: Denosing. *Inverse Problems*, 27:045009 30pp, 2011.
- [19] Charles A. Micchelli, Lixin Shen, Yuesheng Xu, and Xueying Zeng. Proximity algorithms for the L1/TV image denosing models. *Advances in Computational Mathematics*, to appear, 2011.
- [20] J.-J. Moreau. Fonctions convexes duales et points proximaux dans un espace hilbertien. *C.R. Acad. Sci. Paris Sér. A Math.*, 255:1897–2899, 1962.
- [21] M. Nikolova. A variational approach to remove outliers and impulsive noise. *Journal of Mathematical Imaging and Vision*, 20:99–120, 2004.
- [22] R. Rockafellar. Augmented Lagrangians and applications of the proximal point algorithm in convex programming. *Mathematics of Operations Research*, 8:421–439, 1976.
- [23] R. T. Rockafellar and R. Wets. *Variational Analysis*. Vol. 317 Grundlehren der mathematischen Wissenschaften, Springer, 1998.
- [24] C. Wu, J. Zhang, and X. Tai. Augmented Lagrangian method for total variation restoration with non-quadratic fidelity. *Inverse Problems and Imaging*, 5(1):237–261, 2011.
- [25] J. Yang, Y. Zhang, and W. Yin. An efficient TVL1 algorithm for deblurring multichannel images corrupted by impulsive noise. *SIAM Journal on Scientific Computing*, 31:2842–2865, 2009.
- [26] W. Yin, T. Chen, S.X. Zhou, and A. Chakraborty. Background correction for cdna microarray images using the TV+L1 model. *Bioinformatics*, 21(10):2410–2416, 2005.
- [27] W. Yin, D. Goldfard, and S. Osher. The total variation l1 model for multiscale decomposition. *Multiscale Modelling and Simulation: A SIAM Interdisciplinary Journal*, 6:190–211, 2007.
- [28] C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime TV-L1 optical flow. *Pattern Recognition*, 4713:214–223, 2007.