

# Proximity Algorithms for Image Models: Denoising <sup>\*</sup>

Charles A. Micchelli<sup>†</sup>

Lixin Shen<sup>‡</sup>

Yuesheng Xu<sup>§</sup>

## Abstract

This article introduces a novel framework for the study of the total variation model for image denoising. In this model, the denoised image is the proximity operator of the total variation evaluated at the given noisy image. Moreover, the total variation can be viewed as the composition of a convex function (the  $\ell^1$  norm for the anisotropic total variation or the  $\ell^2$  norm for the isotropic total variation) with a linear transformation (the first order difference operator). These two facts lead us to investigate the proximity operator of the composition of a convex function with a linear transformation. Under the assumption that the proximity operator of a given convex function (e.g., the  $\ell^1$  norm or the  $\ell^2$  norm) can be readily obtained, we propose a fixed point algorithm for computing the proximity operator of the composition of this convex function with a linear transformation. We then specialize this fixed point methodology to the total variation denoising models. The resulting algorithms are compared with the Goldstein-Osher split Bregman denoising algorithm. An important advantage of the fixed point framework leads to a convenient analysis for convergence of the proposed algorithms as well as a platform for us to develop efficient numerical algorithms via various fixed point iterations. Our numerical experience indicates that the method proposed here performs favorably.

## 1 Introduction

Noise reduction is a long standing problem in image processing. Many applications such as image segmentation, surveillance, or medical image analysis require effective noise suppression to produce reliable results. To suppress noise in noisy images, many efficient mathematical methods such as variational PDE methods and multiscale methods have been successfully developed and made a significant impact on image denoising and other fields of imaging science. The strength of multiscale analysis in image denoising is that it can provide a sparse representation of the underlying image. This key strength is fully exploited in developing wavelet-based image denoising algorithms [5, 12, 20, 26]. The strength of variational PDE methods is that visually important geometric features of images, such as gradients, curvatures, and level sets, can be directly integrated into a functional to be minimized [7, 27, 28]. Among these image denoising models, the well-known Rudin-Osher-Fatemi (ROF) total variation model is one of the most popular models.

---

<sup>\*</sup>The research was supported by the US National Science Foundation under grant DMS-0712827 and by Guangdong Provincial Government of China through the “Computational Science Innovative Research Team” program.

<sup>†</sup>Department of Mathematics and Statistics, SUNY at Albany, Albany, NY, and Department of Mathematics, City University of Hong Kong, Kowloon, Hong Kong. The research was partially supported by US Air Force Office of Scientific Research under grant FA9550-09-1-0511.

<sup>‡</sup>Department of Mathematics, Syracuse University, Syracuse, NY 13244. Email: lshen03@syr.edu.

<sup>§</sup>All correspondence should be sent to this author. Department of Mathematics, Syracuse University, Syracuse, NY 13244, USA, and School of Mathematics and Computational Sciences, Sun Yat-sen University, Guangzhou 510275, P. R. China. E-mail address: yxu06@syr.edu. The research was partially supported in part by US Air Force Office of Scientific Research under grant FA9550-09-1-0511 and by the Natural Science Foundation of China under grants 10371122 and 10631080.

The ROF model produces the denoised image given by

$$\operatorname{argmin} \left\{ \frac{1}{2} \|u - x\|_2^2 + \mu \|u\|_{\text{TV}} : u \in \mathbb{R}^d \right\}, \quad (1)$$

where  $x \in \mathbb{R}^d$  denotes the noisy image to be denoised and  $\|u\|_{\text{TV}}$  is the total variation of  $u$ . The distinctive feature of the total variation regularization and its various variants is that edges of images are preserved in the denoised image (1). Difficulty in minimizing the functional appearing in (1) lies in the nondifferentiability of the total variation norm and the large dimension of the underlying images. A number of ideas have been proposed to address these issues. For instance, PDE based models with explicit, semi-implicit, and fixed point schemes were proposed in [1, 27, 28, 29, 30] for a modified total variation model. Based on the discretized cost functional appearing in (1), various methods which include subgradient descents, subgradient projections, the Newton-like method, second-order cone programming, interior point methods, graph-based approaches, Nesterov's first-order explicit schemes were proposed, see [8, 10, 14, 19, 23, 31, 34] and the references cited therein. Dual and primal-dual approaches were also studied in [4, 6, 16]. Recently, the Bregman iteration was successfully used in image denoising due to its speed, simplicity, efficiency and stability, see for example, [15, 25, 32, 33].

The ROF model can be viewed as the proximity operator of  $\mu \|\cdot\|_{\text{TV}}$  evaluated at the noisy image  $x$ . Although this is a convenient interpretation of the method it is not immediately useful since the proximity operator of  $\mu \|\cdot\|_{\text{TV}}$  cannot be easily computed. To tackle this problem, we treat  $\|\cdot\|_{\text{TV}}$  as the composition of a convex function (the norm  $\|\cdot\|_1$  for the anisotropic total variation or a certain combination of the norm  $\|\cdot\|_2$  for the isotropic total variation) with a linear transformation (the first order difference operator). The motivation of viewing the TV-norm as a composition map is that the norm appearing in this composition has an explicit proximity map. This triggers us to develop new algorithms for the ROF model by fully exploiting the explicit form of the proximity map of the norm  $\|\cdot\|_1$  or the norm  $\|\cdot\|_2$ .

Our discussion is conducted in a general setting having the ROF model as its special case. We shall describe how to compute the proximity operator of a composition of a convex function with a linear transformation under an assumption that the proximity operator of this convex function can be computed efficiently. With the additional observation that the subdifferential of this convex function can be completely characterized by its proximity operator, we are able to formulate the problem of computing the proximity operator of the composition function as a fixed point problem. The fixed point problem is analyzed and numerical algorithms for solving it are developed.

This paper is organized in the following manner. In Section 2 we review the concepts of proximity operator and subdifferential and provide a characterization of the proximity operator and the subdifferential of a convex function. This characterization plays a central role in this paper. In Section 3 we represent the proximity operator of the composition of a convex function and a linear transformation as a fixed point problem. We conduct a theoretical study of the fixed point problem and provide a numerical treatment for its solution. In Section 4 we specify the ROF total variation denoising problem as a special case of our general setting on proximity operators of composition functions. We develop new algorithms for the ROF model and compare them with several existing algorithms including the Chambolle algorithm [4], the Goldstein-Osher split Bregman method [15], and the Jai-Zhao denoising algorithm [17]. In Section 5 we give a Gauss-Seidel variation of our proposed algorithms to speed up convergence. In Section 6 we focus on the implementation of the proposed algorithms in the context of image denoising. Moreover, we compare the numerical performance of our proposed algorithms and the split Bregman algorithms. Our conclusions about the algorithms proposed here are summarized in Section 7.

## 2 Proximity Operator and Subdifferential

The proximity operator and subdifferential of a convex function are two important concepts in convex analysis. In this section, we review some aspects of these concepts needed in this paper. In addition, we provide a characterization of the subdifferential of a convex function in terms of its associated proximity operator. This characterization will play a crucial role in the development of fixed point algorithms based upon the proximity operator which will be described in the subsequent sections.

We begin by introducing our notation. We denote by  $\mathbb{R}^d$  the usual  $d$ -dimensional Euclidean space. For  $x, y \in \mathbb{R}^d$ , we define  $\langle x, y \rangle := \sum_{i=1}^d x_i y_i$ , the standard inner product of  $\mathbb{R}^d$ . We denote by  $\|\cdot\|$  a norm of  $\mathbb{R}^d$ . In particular, for  $p \geq 1$ , we define the  $p$ -norm of a vector  $x \in \mathbb{R}^d$  as  $\|x\|_p := (\sum_{i=1}^d |x_i|^p)^{\frac{1}{p}}$ . For a norm  $\|\cdot\|$  on  $\mathbb{R}^d$ , its dual norm  $\|\cdot\|_*$  is defined by

$$\|y\|_* := \max\{\langle x, y \rangle : \|x\| = 1\}.$$

The proximity operator was introduced by Moreau in [21, 22]. We recall its definition as follows:

**Definition 2.1** *Let  $\psi$  be a real-valued convex function on  $\mathbb{R}^d$ . The proximity operator of  $\psi$  is defined for  $x \in \mathbb{R}^d$  by*

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

**Definition 2.2** *Let  $\psi$  be a real-valued convex function on  $\mathbb{R}^d$ . The subdifferential of  $\psi$  at  $x \in \mathbb{R}^d$  is defined by*

$$\partial\psi(x) := \{y : y \in \mathbb{R}^d \text{ and } \psi(z) \geq \psi(x) + \langle y, z - x \rangle \text{ for all } z \in \mathbb{R}^d\}. \quad (3)$$

*Elements in  $\partial\psi(x)$  are called subgradients.*

Definition 2.2 says that the subdifferential of  $\psi$  is a set-valued mapping from  $\mathbb{R}^d$  into a nonempty convex compact set in  $\mathbb{R}^d$  (see, e.g., [2, Page 732]).

We now present three examples for which we can explicitly calculate their subdifferentials and proximity operators. These examples are also useful for developing algorithms for total variation denoising models.

The first example concerns the subdifferential and the proximity operator of the absolute value function  $\psi = \frac{1}{\lambda} |\cdot|$ , where  $\lambda > 0$ .

**Example 2.3** *If  $\lambda > 0$  and  $x \in \mathbb{R}$  then*

$$\partial \left( \frac{1}{\lambda} |\cdot| \right) (x) = \begin{cases} \frac{1}{\lambda} \{\text{sign}(x)\}, & \text{if } x \neq 0; \\ [-\frac{1}{\lambda}, \frac{1}{\lambda}], & \text{otherwise} \end{cases}$$

*and*

$$\text{prox}_{\frac{1}{\lambda} |\cdot|} x = \max \left( |x| - \frac{1}{\lambda}, 0 \right) \text{sign}(x).$$

*Clearly,  $\text{prox}_{\frac{1}{\lambda} |\cdot|}$  is the well-known soft thresholding operator with  $\frac{1}{\lambda}$  as the threshold, [13].*

The second example is a direct extension of the first example to  $\mathbb{R}^m$ .

**Example 2.4** If  $\lambda > 0$  and  $x \in \mathbb{R}^m$  then

$$\partial \left( \frac{1}{\lambda} \|\cdot\|_1 \right) (x) = \left\{ y : y \in \mathbb{R}^m, y_k \in \partial \left( \frac{1}{\lambda} |\cdot| \right) (x_k), k = 1, \dots, m \right\}, \quad (4)$$

and

$$\text{prox}_{\frac{1}{\lambda} \|\cdot\|_1} x = \left[ \text{prox}_{\frac{1}{\lambda} |\cdot|} x_1, \text{prox}_{\frac{1}{\lambda} |\cdot|} x_2, \dots, \text{prox}_{\frac{1}{\lambda} |\cdot|} x_m \right]^t. \quad (5)$$

The third example is about the subdifferential and the proximity operator of the  $\ell^2$ -norm on  $\mathbb{R}^m$ , that is  $\psi = \frac{1}{\lambda} \|\cdot\|_2$ , where  $\lambda > 0$ .

**Example 2.5** If  $\lambda > 0$  and  $x \in \mathbb{R}^m$ , then

$$\partial \left( \frac{1}{\lambda} \|\cdot\|_2 \right) (x) = \begin{cases} \frac{1}{\lambda} \left\{ \frac{x}{\|x\|_2} \right\}, & \text{if } x \neq 0; \\ \left\{ y : y \in \mathbb{R}^m : \|y\|_2 \leq \frac{1}{\lambda} \right\}, & \text{otherwise.} \end{cases}$$

and

$$\text{prox}_{\frac{1}{\lambda} \|\cdot\|_2} x = \max \left( \|x\|_2 - \frac{1}{\lambda}, 0 \right) \frac{x}{\|x\|_2} = \text{prox}_{\frac{1}{\lambda} |\cdot|} (\|x\|_2) \frac{x}{\|x\|_2}. \quad (6)$$

The following result characterizes the relationship between the proximity operator and the subdifferential of a convex function. This proposition serves as a basic tool for the algorithmic development of many tasks in image analysis. We shall explain this in detail in the following sections.

**Proposition 2.6** If  $\psi$  is a convex function on  $\mathbb{R}^d$  and  $x \in \mathbb{R}^d$ , then

$$y \in \partial_\psi(x) \quad \text{if and only if} \quad x = \text{prox}_\psi(x + y). \quad (7)$$

Proof: From Definition 2.2 of subdifferential, the inclusion  $y \in \partial_\psi(x)$  is equivalent to the requirement that

$$\psi(z) - \psi(x) - \langle y, z - x \rangle \geq 0, \quad \text{for all } z \in \mathbb{R}^d. \quad (8)$$

By Definition 2.1 of the proximity operator, the equality  $x = \text{prox}_\psi(x + y)$  is equivalent to the inequality

$$\frac{1}{2} \|x - (x + y)\|_2^2 + \psi(x) \leq \frac{1}{2} \|z - (x + y)\|_2^2 + \psi(z), \quad \text{for all } z \in \mathbb{R}^d. \quad (9)$$

Simplifying the above inequality (9) yields the equivalent inequality

$$-\frac{1}{2} \|z - x\|_2^2 \leq \psi(z) - \psi(x) - \langle y, z - x \rangle, \quad \text{for all } z \in \mathbb{R}^d. \quad (10)$$

Now, if the inequality (8) holds, so does (10). In other words, we clearly have that the inclusion  $y \in \partial_\psi(x)$  implies  $x = \text{prox}_\psi(x + y)$ .

We next prove that (10) implies (8). If (10) is true, choosing  $v \in \mathbb{R}^d$ ,  $\rho > 0$  and setting  $z = x + \rho v$  in (10), we obtain that

$$-\frac{1}{2} \rho \|v\|_2^2 \leq \frac{\psi(x + \rho v) - \psi(x)}{\rho} - \langle y, v \rangle.$$

Note that  $(\psi(x + \rho v) - \psi(x))/\rho$  is an increasing function of the variable  $\rho$ . Consequently, from the above inequality it follows, for  $0 < \rho \leq 1$ , that

$$-\frac{1}{2} \rho \|v\|_2^2 \leq \frac{\psi(x + \rho v) - \psi(x)}{\rho} - \langle y, v \rangle \leq \psi(x + v) - \psi(x) - \langle y, v \rangle.$$

Letting  $\rho \rightarrow 0+$ , we get that  $0 \leq \psi(x+v) - \psi(x) - \langle y, v \rangle$  which implies equation (8).  $\square$

As a result of Proposition 2.6, we have for  $x \in \mathbb{R}^d$  that

$$\partial_\psi(x) = \{y : y \in \mathbb{R}^d, x = \text{prox}_\psi(x+y)\}.$$

### 3 Fixed Point Algorithms Based on Proximity Operator

In this section, we develop fixed point algorithms based on the proximity operator for the composition of a convex function and a linear transformation. To this end, for a convex function  $\varphi$  on  $\mathbb{R}^m$  and an  $m \times d$  matrix  $B$ , we define a function  $\psi$  for every  $x \in \mathbb{R}^d$  by

$$\psi(x) := (\varphi \circ B)(x).$$

We remark that  $\psi$  is also convex. Under the assumption that we can explicitly compute the proximity operator  $\text{prox}_\varphi$ , our aim is to develop an algorithm for evaluating  $\text{prox}_{\varphi \circ B}$ . That is, we consider the following minimization problem

$$\min \left\{ \frac{1}{2} \|u - x\|_2^2 + (\varphi \circ B)(u) : u \in \mathbb{R}^d \right\}, \quad (11)$$

where  $x$  is a given vector in  $\mathbb{R}^d$ . The reason for considering the minimization problem (11) is that in special cases it corresponds to the ROF total variation denoising model (1). For example, for the ROF model with the anisotropic total variation, one can write  $\|\cdot\|_{\text{TV}} = \|\cdot\|_1 \circ B$ , where  $B$  is a first order difference operator and the proximity operator of  $\|\cdot\|_1$  as shown in Example 2.4 has an explicit expression. More details on the anisotropic and isotropic total variation will be presented later in Section 4.

#### 3.1 Formulation of Fixed Point Problems

The vector which achieves the minimum in (11), denoted by  $\text{prox}_{\varphi \circ B}x$ , exists and is unique. Our aim is to provide a constructive method to find it. Clearly,  $\text{prox}_{\varphi \circ B}x$  can be characterized by the inclusion

$$\text{prox}_{\varphi \circ B}x \in x - \partial(\varphi \circ B)(\text{prox}_{\varphi \circ B}x). \quad (12)$$

To identify the subdifferential  $\partial(\varphi \circ B)$  appearing in this inclusion, we recall the chain rule (see, e.g., [35, Theorem 2.8.3]) which gives us the formula

$$\partial(\varphi \circ B) = B^t \circ (\partial\varphi) \circ B. \quad (13)$$

Combining equations (12) and (13) yields the fact that

$$\text{prox}_{\varphi \circ B}x \in x - B^t \partial\varphi(B \text{prox}_{\varphi \circ B}x). \quad (14)$$

Equation (14) indicates that  $\text{prox}_{\varphi \circ B}$  is related to the subdifferential of  $\varphi$ . Furthermore, the relationship between the subdifferential of  $\varphi$  and the proximity operator of  $\varphi$  has been established in Proposition 2.6. These facts enable us to establish a relationship between the proximity operator of  $\varphi \circ B$  and the proximity operator of  $\varphi$ . To formulate our observation we introduce the affine transformation  $A : \mathbb{R}^m \rightarrow \mathbb{R}^m$  defined, for a fixed  $x \in \mathbb{R}^d$ , at  $y \in \mathbb{R}^m$  by

$$Ay := Bx + (I - \lambda BB^t)y, \quad \text{for all} \quad (15)$$

and the operator  $H : \mathbb{R}^m \rightarrow \mathbb{R}^m$

$$H := (I - \text{prox}_{\frac{1}{\lambda}\varphi}) \circ A. \quad (16)$$

**Theorem 3.1** *If  $\varphi$  is a convex function on  $\mathbb{R}^m$ ,  $B$  is an  $m \times d$  matrix,  $x \in \mathbb{R}^d$  and  $\lambda$  is a positive number then*

$$\text{prox}_{\varphi \circ B} x = x - \lambda B^t v \quad (17)$$

*if and only if  $v \in \mathbb{R}^m$  is a fixed point of  $H$ .*

Proof: From (14), we conclude that  $\text{prox}_{\varphi \circ B}$  is characterized by the fact that

$$\text{prox}_{\varphi \circ B} x = x - \lambda B^t v \quad (18)$$

where  $v$  is a vector in the set  $\partial \left( \frac{1}{\lambda} \varphi \right) (B \text{prox}_{\varphi \circ B} x)$ . Thus, it follows that  $v \in \partial \left( \frac{1}{\lambda} \varphi \right) (B(x - \lambda B^t v))$ . Using Proposition 2.6 we conclude that

$$Bx - \lambda B B^t v = \text{prox}_{\frac{1}{\lambda} \varphi} (Bx + (I - \lambda B B^t) v), \quad (19)$$

that is,  $v$  is a fixed point of  $H$ .

Conversely, if  $v$  is a fixed point of  $H$  then equation (19) holds. Using again Proposition 2.6 and the chain rule (13), we conclude that

$$\lambda B^t v \in \partial(\varphi \circ B)(x - \lambda B^t v).$$

Since  $\varphi \circ B$  is convex, Proposition 2.6 together with the above inclusion implies (18).  $\square$

Theorem 3.1 demonstrates the solution of the minimization problem (11) corresponds to a fixed point of the operator  $H$ . We summarize this observation for finding the minimum of problem (11) as follows:

**Fixed Point Problem Based on Proximity Operator (FP<sup>2</sup>O)**

$$v = H v$$

$$\text{prox}_{\varphi \circ B} x = x - \lambda B^t v.$$

Based on the above observation, we will focus our attention on finding the fixed points of the operator  $H$  by exploiting existing results from fixed point theory.

## 3.2 Fixed Points of the Operator $H$

In this subsection, we will present conditions concerning the existence of fixed points for the operator  $H$  defined by (16) which are a direct consequence of Theorem 3.1.

**Proposition 3.2** *If  $\varphi$  is a convex function on  $\mathbb{R}^m$ ,  $B$  is an  $m \times d$  matrix,  $x \in \mathbb{R}^d$  and  $\lambda$  is a positive number, then  $H$  has a fixed point.*

Proof: Since the vector  $\text{prox}_{\varphi \circ B} x$  which achieves the minimum in (11) exists and is unique, the first part of the proof for Theorem 3.1 tells us that  $H$  has a fixed point.  $\square$

The next result is needed in order to develop a convergent algorithm for finding the fixed point of  $H$ . To this end, we begin with the definition of nonexpansivity. A nonlinear operator  $\mathcal{T} : \mathbb{R}^d \rightarrow \mathbb{R}^d$  is nonexpansive if for  $x, y \in \mathbb{R}^d$ ,

$$\|\mathcal{T}(x) - \mathcal{T}(y)\|_2 \leq \|x - y\|_2.$$

It was proved in [9] that  $\text{prox}_\psi$  satisfies for all  $x, y \in \mathbb{R}^d$  the inequality

$$\|\text{prox}_\psi x - \text{prox}_\psi y\|_2^2 \leq \langle x - y, \text{prox}_\psi x - \text{prox}_\psi y \rangle, \quad (20)$$

as a result of (3.3) and the Cauchy-Schwarz inequality we conclude that  $\text{prox}_\psi$  is nonexpansive.

Likewise, from a direct computation using (3.3) we conclude for all  $x, y \in \mathbb{R}^d$  that

$$\|(I - \text{prox}_\psi)(x) - (I - \text{prox}_\psi)(y)\|_2^2 \leq \langle x - y, (I - \text{prox}_\psi)x - (I - \text{prox}_\psi)y \rangle. \quad (21)$$

Therefore, another application of the Cauchy-Schwarz inequality applied to the inequality (21) proves that  $I - \text{prox}_\psi$  is also nonexpansive.

**Lemma 3.3** *If  $\varphi$  is a convex function on  $\mathbb{R}^m$ ,  $B$  is an  $m \times d$  matrix and  $\lambda$  is a positive number such that  $\|I - \lambda BB^t\|_2 \leq 1$  then  $H$  is nonexpansive.*

*Proof:* We need to show that  $H$  satisfies the condition  $\|Hv - Hw\|_2 \leq \|v - w\|_2$  for all vectors  $v, w \in \mathbb{R}^m$ . Since the operator  $I - \text{prox}_{\frac{1}{\lambda}\varphi}$  is nonexpansive (see (21)), we have that

$$\|Hv - Hw\|_2 = \|(I - \text{prox}_{\frac{1}{\lambda}\varphi})(Av) - (I - \text{prox}_{\frac{1}{\lambda}\varphi})(Aw)\|_2 \leq \|Av - Aw\|_2$$

The definition of  $A$  in (15) and the choice of  $\lambda$  yield that

$$\|Av - Aw\|_2 = \|(I - \lambda BB^t)(v - w)\|_2 \leq \|I - \lambda BB^t\|_2 \|v - w\|_2 \leq \|v - w\|_2.$$

This together with the previous estimate completes the proof.  $\square$

In the following series of lemmas, we obtain a preliminary observation on the location of the fixed points of  $H$ . To this end, we recall that a real-valued function  $\varphi$  is Lipschitz continuous with Lipschitz constant  $K$  provided that  $|\varphi(u) - \varphi(v)| \leq K\|u - v\|$ , for all  $u, v \in \mathbb{R}^m$ . The smallest constant  $K$  in this inequality is denoted by  $\text{Lip}_\varphi$ .

**Lemma 3.4** *If  $\varphi$  is a Lipschitz continuous convex function and  $y \in \partial_\varphi(x)$  for some  $x \in \mathbb{R}^m$  then*

$$\|y\|_* \leq \text{Lip}_\varphi. \quad (22)$$

*Proof:* Since  $y \in \partial_\varphi(x)$  for some  $x \in \mathbb{R}^m$ , we have from the definition of subdifferential of  $\varphi$  that

$$\varphi(z) \geq \varphi(\text{prox}_\varphi x) + \langle y, z - \text{prox}_\varphi x \rangle, \quad \text{for all } z \in \mathbb{R}^m.$$

We let  $z := \text{prox}_\varphi x + \rho v$ , for  $\rho > 0$  and  $v \in \mathbb{R}^m$ , and substitute it into the above inequality to conclude that

$$\langle y, v \rangle \leq |(\varphi(\text{prox}_\varphi x + \rho v) - \varphi(\text{prox}_\varphi x))|/\rho \leq \text{Lip}_\varphi \|v\|$$

for arbitrary  $v$ . Hence, by the definition of the dual norm, we conclude the validity of the inequality (22).  $\square$

As a consequence of this lemma and the triangle inequality, we have for any norm  $\|\cdot\|$  on  $\mathbb{R}^m$  and  $x \in \mathbb{R}^m$  that

$$\partial\|\cdot\|(x) \subseteq \{y : \|y\|_* \leq 1\}.$$

For the next lemma, we introduce the set  $C_\varphi := \{z : z \in \mathbb{R}^m, \|z\|_* \leq \text{Lip}_\varphi/\lambda\}$ .

**Lemma 3.5** *If  $\varphi$  is a Lipschitz continuous convex function and  $B$  is an  $m \times d$  matrix then  $H$  maps  $\mathbb{R}^m$  into  $C_\varphi$ .*

Proof: Since the operator  $H$  defined by (16) is the composition of the operator  $I - \text{prox}_{\frac{1}{\lambda}\varphi}$  with the affine transformation  $A$ , it suffices to show that for any  $v \in \mathbb{R}^m$ ,  $v - \text{prox}_{\frac{1}{\lambda}\varphi} v \in C_\varphi$ . However, by (14), we have  $v - \text{prox}_{\frac{1}{\lambda}\varphi} v \in \partial\left(\frac{1}{\lambda}\varphi\right)\left(\text{prox}_{\frac{1}{\lambda}\varphi} v\right)$  and so, by Lemma 3.4, the result follows.  $\square$

From Lemma 3.5, we have the result on the location of the fixed points of  $H$ .

**Proposition 3.6** *If  $\varphi$  is a Lipschitz continuous convex function and  $B$  is an  $m \times d$  matrix such that  $\|I - \lambda BB^t\|_2 \leq 1$  then all the fixed points of  $H$  are in  $C_\varphi$  and  $C_\varphi$  is nonempty.*

Proof: By Proposition 3.6, the hypotheses of this proposition ensure that  $H$  has a fixed point. By Lemma 3.5, the fixed points of  $H$  are in  $C_\varphi$ , and thus,  $C_\varphi$  is nonempty.  $\square$

### 3.3 Iterative Algorithms for Fixed Points

The main focus of this subsection is to give methods which permit the computation of the fixed points of  $H$  defined by (16). Since the operator  $H$  is nonexpansive under certain conditions on  $\varphi$  and  $B$  (see Lemma 3.3), some methods in the existing literature can be used to find the fixed points of  $H$ . For example, it was pointed out in [11] that a sequence defined by using the iterations of scaled  $H$  converges to a fixed point of  $H$ . Here, we are concern with the convergence of the Picard iterates of  $H$ , to be defined below. We recall some basic definitions used for finding fixed points by Picard iterates.

For a given  $v^0 \in \mathbb{R}^m$  and an operator  $\mathcal{P} : \mathbb{R}^m \rightarrow \mathbb{R}^m$  we define  $v^{n+1} := \mathcal{P}(v^n)$  for  $n \in \mathbb{N}$ . The sequence  $\{v^n : n \in \mathbb{N}\}$ , is called the Picard sequence of the operator  $\mathcal{P}$ . For any  $\kappa \in (0, 1)$ , the  $\kappa$ -averaged operator  $\mathcal{P}_\kappa$  of  $\mathcal{P}$  is defined by

$$\mathcal{P}_\kappa := \kappa I + (1 - \kappa)\mathcal{P}. \quad (23)$$

The following lemma considers the convergence of the Picard sequence associated to the  $\kappa$ -averaged operator of  $\mathcal{P}$ .

**Lemma 3.7 (Opial  $\kappa$ -averaged Theorem [24])** *If  $C$  is a closed and convex set in  $\mathbb{R}^m$  and  $\mathcal{P} : C \rightarrow C$  is a nonexpansive mapping with at least one fixed point then for  $\kappa \in (0, 1)$   $\mathcal{P}_\kappa$  is nonexpansive, maps  $C$  to itself, and has the same set of the fixed points as  $\mathcal{P}$ . Furthermore, for any  $u \in C$  and any  $\kappa \in (0, 1)$ , the Picard sequence of  $\mathcal{P}_\kappa$  converges to a fixed point of  $\mathcal{P}$ .*

Basically, the Opial  $\kappa$ -averaged Theorem says that a fixed point problem of a nonexpansive mapping can be studied through the corresponding  $\kappa$ -averaged operator. As a direct consequence of the Opial  $\kappa$ -averaged Theorem to the operator  $H$ , we have the following result.

**Corollary 3.8** *Under the conditions given in Proposition 3.6, for any  $u \in C_\varphi$  and any  $\kappa \in (0, 1)$ , the Picard sequence  $H_\kappa$  converges to a fixed point of  $H$ .*

Proof: From Lemmas 3.3, 3.5, and 3.6, we know that the operator  $H$  is nonexpansive, maps the set  $C_\varphi$  to itself, and has a fixed point. By the Opial  $\kappa$ -averaged Theorem, we conclude that for any  $u \in C_\varphi$  and any  $\kappa \in (0, 1)$ , the Picard sequence of  $H_\kappa$  converges to a fixed point of  $H$ .  $\square$

Note that the parameter  $\kappa$  is required to be strictly between 0 and 1 in Corollary 3.8. We are unable to extend it to the case of  $\kappa = 0$ . The next corollary gives some information about this issue. To this end, we recall that a mapping  $\mathcal{P} : \mathbb{R}^m \rightarrow \mathbb{R}^m$  is called firmly nonexpansive provided for any  $x, y \in \mathbb{R}^m$ ,

$$\|\mathcal{P}x - \mathcal{P}y\|_2^2 \leq \langle x - y, \mathcal{P}x - \mathcal{P}y \rangle.$$

Observe that the form of the mapping  $H$  given by (16) suggests that we consider it for arbitrary affine map  $A$  from  $\mathbb{R}^m$  to  $\mathbb{R}^m$ . Clearly, as in the proof of Lemma 3.3 we conclude that in this generality  $H$  still remains nonexpansive provided that  $A$  is nonexpansive. The simplest choice of nonexpansive affine map corresponds to a shift operator. That is, there exists a  $a \in \mathbb{R}^m$  such that for any  $x \in \mathbb{R}^m$ ,  $Ax = a + x$ . This choice also has the crucial property that the corresponding  $H$  is firmly nonexpansive. Therefore, the next corollary establishes that the Picard iterates of  $H$  corresponding to a shift operator always converges.

**Proposition 3.9** *If  $\mathcal{P} : \mathbb{R}^m \rightarrow \mathbb{R}^m$  is firmly nonexpansive and has at least one fixed point then its Picard sequence converges.*

Proof: We show that  $2\mathcal{P} - I$  is nonexpansive on  $\mathbb{R}^m$ . To this end, for any vectors  $u, v \in \mathbb{R}^m$ , we have that

$$\|(2\mathcal{P} - I)(u) - (2\mathcal{P} - I)(v)\|_2^2 = 4\|\mathcal{P}u - \mathcal{P}v\|_2^2 - 4\langle \mathcal{P}u - \mathcal{P}v, u - v \rangle + \|u - v\|_2^2.$$

Since the operator  $\mathcal{P}$  is firmly nonexpansive, we conclude that

$$\|(2\mathcal{P} - I)(u) - (2\mathcal{P} - I)(v)\|_2^2 \leq \|u - v\|_2^2.$$

That is,  $2\mathcal{P} - I$  is nonexpansive. It is clear that the set of fixed points of  $2\mathcal{P} - I$  is identical to that of  $\mathcal{P}$ . We further note that

$$\mathcal{P} = \frac{1}{2}(2\mathcal{P} - I) + \frac{1}{2}I.$$

By Lemma 3.7 with  $\kappa = \frac{1}{2}$ , we know that for any given initial vector the Picard sequence of the operator  $\mathcal{P}$  converges to a fixed point of  $\mathcal{P}$ .  $\square$

We elaborate on two special cases of this proposition. The first concerns the  $\ell^1$  norm and the other treats the case of the  $\ell^2$  norm. In the first case, we provide the additional information that the Picard iterates converge in a *finite* number of steps.

**Proposition 3.10** *If  $\lambda > 0$  and  $a, x^0 \in \mathbb{R}^m$  then the iterative scheme*

$$x^{k+1} = (I - \text{prox}_{\frac{1}{\lambda}|\cdot|})(x^k + a), \quad k = 0, 1, \dots, \quad (24)$$

*converges to its limit in a finite number of steps and for  $i = 1, 2, \dots, m$ ,*

$$\lim_{k \rightarrow \infty} (x^k)_i = \begin{cases} \frac{\text{sign}(a_i)}{\lambda}, & a_i \neq 0, \\ (I - \text{prox}_{\frac{1}{\lambda}|\cdot|})(x^0)_i, & a_i = 0. \end{cases}$$

Proof: It suffices to prove this result in the case of  $m = 1$  since all mappings act coordinate-wise. We divide the proof into the three cases  $a > 0$ ,  $a < 0$ , and  $a = 0$ . We start with the case that  $a > 0$ . In this case, by Example 2.3, we have that  $\frac{1}{\lambda} = (I - \text{prox}_{\frac{1}{\lambda}|\cdot|})(\frac{1}{\lambda} + a)$ . That is, the number  $\frac{1}{\lambda}$  is a fixed point of the iterative scheme. If  $x^0 + a > \frac{1}{\lambda}$ , then  $x^1 = \frac{1}{\lambda}$ . Hence,  $x^k = \frac{1}{\lambda}$  for all  $k \geq 1$ . In other words, one iteration is enough to reach the limit of the iterative scheme. If  $|x^0 + a| \leq \frac{1}{\lambda}$ , we have  $x^k = x^0 + ka$  if  $0 \leq k < \lceil \frac{1}{\lambda a} - \frac{x^0}{a} \rceil$  and  $x^k = \frac{1}{\lambda}$  if  $k \geq \lceil \frac{1}{\lambda a} - \frac{x^0}{a} \rceil$ . Here,  $\lceil c \rceil$  represents the smallest integer which exceeds  $c$ . The iterative scheme reaches its limit in  $\lceil \frac{1}{\lambda a} - \frac{x^0}{a} \rceil$  steps. If  $x^0 + a < -\frac{1}{\lambda}$ , then  $x^1 = -\frac{1}{\lambda}$ . Using the preceding result, the scheme reaches its limit in  $\lceil \frac{2}{\lambda a} \rceil + 1$  steps.

The proof for the case  $a < 0$  is similar to that for the case  $a > 0$ . Specifically, the iterative scheme converges to limit  $-\frac{1}{\lambda}$  in a finite number of steps.

Finally, we consider the case  $a = 0$ . In this case, the iterative scheme becomes  $x^{k+1} = (I - \text{prox}_{\frac{1}{\lambda}|\cdot|})x^k$ . It is easy to see that for all  $k \geq 1$ ,  $x^k = \frac{1}{\lambda}$  if  $x^0 > \frac{1}{\lambda}$ ;  $x^k = -\frac{1}{\lambda}$  if  $x^0 < -\frac{1}{\lambda}$ ;  $x^k = x^0$  if  $|x^0| \leq \frac{1}{\lambda}$ . Consequently, the iteration reaches its limit in just one step.  $\square$

We remark that the limit of the Picard iterates in Proposition 3.10 can also be expressed for any  $1 \leq p < \infty$  in the form

$$\lim_{k \rightarrow \infty} x^k = \arg \min \left\{ \|x - x^0\|_p : x \in \frac{1}{\lambda} \partial \|\cdot\|_1(a) \right\}.$$

**Proposition 3.11** *If  $\lambda > 0$  and  $a, x^0 \in \mathbb{R}^m$  then the iterative scheme*

$$x^{k+1} = (I - \text{prox}_{\frac{1}{\lambda}\|\cdot\|_2})(x^k + a), \quad k = 0, 1, \dots, \quad (25)$$

*converges to its limits in a finite number of steps and*

$$\lim_{k \rightarrow \infty} x^k = \begin{cases} x^0 - \max(\|x^0\|_2 - \frac{1}{\lambda}, 0) \frac{x^0}{\|x^0\|_2}, & a = 0, \\ \frac{a}{\lambda\|a\|_2}, & a \neq 0. \end{cases}$$

Moreover, for any  $1 \leq p < \infty$ ,

$$\lim_{k \rightarrow \infty} x^k = \arg \min \left\{ \|x - x^0\|_p : x \in \frac{1}{\lambda} \partial \|\cdot\|_2(a) \right\}.$$

Proof: First, we consider the case when  $a = 0$ . In this case, equation (25) reduces to  $x^{k+1} = (I - \text{prox}_{\frac{1}{\lambda}\|\cdot\|_2})(x^k)$ . If  $\|x^0\|_2 \geq \frac{1}{\lambda}$ , we have that  $x^k = \frac{x^0}{\lambda\|x^0\|_2}$  for all  $k \geq 1$  and if  $\|x^0\|_2 < \frac{1}{\lambda}$ , we have that  $x^k = x^0$  for all  $k \geq 0$ . Hence,  $\lim_{k \rightarrow \infty} x^k = x^0 - \max(\|x^0\|_2 - \frac{1}{\lambda}, 0) \frac{x^0}{\|x^0\|_2}$ .

For the case  $a \neq 0$ , Proposition 3.9 yields that  $x^\infty = (I - \text{prox}_{\frac{1}{\lambda}\|\cdot\|_2})(x^\infty + a)$ , i.e.,  $a = \text{prox}_{\frac{1}{\lambda}\|\cdot\|_2}(x^\infty + a)$ , where  $x^\infty$  is the limit of the Picard iterates. Clearly, by Example 2.5 and equation (7), we have that  $x^\infty = \frac{a}{\lambda\|a\|_2}$ .  $\square$

## 4 Fixed Point Algorithms Based on the Proximity Operator for TV Denoising

Recall that in the previous section we presented a fixed point algorithm based on the proximity operator for the model (11). In this section, we will identify the well-known ROF total variation image denoising model (1) as a special case of (11). To this end, we recall two definitions of total variation which appeared in the literature. Let  $D$  denote the  $N \times N$  matrix defined by the equation

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

and choose  $B$  to be an  $2N^2 \times N^2$  matrix given by

$$B := \begin{bmatrix} I_N \otimes D \\ D \otimes I_N \end{bmatrix} \quad (26)$$

where  $I_N$  is the  $N \times N$  identity matrix and the notation  $P \otimes Q$  denotes the Kronecker product of matrices  $P$  and  $Q$ .

Let  $u$  be an image in  $\mathbb{R}^{N^2}$ . There are two possible definitions of the total variation  $\|u\|_{\text{TV}}$  in the literature. The first one is called the *anisotropic total variation* defined by the formula

$$\|u\|_{\text{TV}} := \|Bu\|_1 \quad (27)$$

while the second definition of total variation is called the *isotropic total variation* and is defined by the equation

$$\|u\|_{\text{TV}} := \sum_{i=1}^{N^2} \left\| \begin{bmatrix} (Bu)_i \\ (Bu)_{N^2+i} \end{bmatrix} \right\|_2. \quad (28)$$

To identify the ROF (1) as a special case of model (11) for the anisotropic total variation (27), we choose  $\varphi : \mathbb{R}^{2N^2} \rightarrow \mathbb{R}$  as

$$\varphi(z) := \mu \|z\|_1, \quad z \in \mathbb{R}^{2N^2} \quad (29)$$

while for the isotropic total variation (28), we choose  $\varphi : \mathbb{R}^{2N^2} \rightarrow \mathbb{R}$  as

$$\varphi(z) := \mu \sum_{i=1}^{N^2} \left\| \begin{bmatrix} z_i \\ z_{N^2+i} \end{bmatrix} \right\|_2, \quad z \in \mathbb{R}^{2N^2}. \quad (30)$$

It is clear that in either of the above cases the function  $\varphi$  is convex, and for these choices, equation (11) reduces to the ROF model (1). We shall develop algorithms for both choices of total variations described above.

#### 4.1 Anisotropic Total Variation Denoising

In this subsection, we consider the anisotropic total variation (27).

**Lemma 4.1** *For the anisotropic total variation (27), the corresponding operator  $H$  maps  $\mathbb{R}^{2N^2}$  into  $[-\frac{\mu}{\lambda}, \frac{\mu}{\lambda}]^{2N^2}$ .*

Proof: The proof follows by specializing Lemma 3.5 to the current situation.  $\square$

**Lemma 4.2** *The eigenvalues,  $\lambda_{ij}$ ,  $0 \leq i, j < N$ , of the  $N^2 \times N^2$  matrix  $B^t B$  are given by the formula*

$$\lambda_{ij} = 2 - \frac{\cos \frac{3i\pi}{2N}}{\cos \frac{i\pi}{2N}} - \frac{\cos \frac{3j\pi}{2N}}{\cos \frac{j\pi}{2N}} \quad (31)$$

and all lie in the interval  $[0, 8)$ .

Proof: A direct computation shows that

$$D^t D = \begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{bmatrix}$$

and the eigenvalues of  $D^t D$  are

$$1 - \frac{\cos \frac{3i\pi}{2N}}{\cos \frac{i\pi}{2N}}, \quad \text{for } i = 0, 1, \dots, N-1.$$

For  $0 \leq z < \pi/2$ , the equalities

$$\left(1 - \frac{\cos 3z}{\cos z}\right)' = \frac{2(\sin z + \sin 3z)}{\cos z} \geq 0 \quad \text{and} \quad \lim_{z \rightarrow \frac{\pi}{2}^-} \left(1 - \frac{\cos 3z}{\cos z}\right) = 4,$$

yield that all eigenvalues of  $D^t D$  are in the interval  $[0, 4)$ . We remark that this result can be alternatively obtained by the Gershgorin disc theorem. Here, exact values of the eigenvalues are provided.

Next, we study the eigenvalues of the matrix  $B^t B$ . Note that

$$B^t B = (I_N \otimes D)^t (I_N \otimes D) + (D \otimes I_N)^t (D \otimes I_N) = I_N \otimes (D^t D) + (D^t D) \otimes I_N.$$

Hence, the eigenvalues of  $B^t B$  are given as

$$2 - \frac{\cos \frac{3i\pi}{2N}}{\cos \frac{i\pi}{2N}} - \frac{\cos \frac{3j\pi}{2N}}{\cos \frac{j\pi}{2N}},$$

for  $0 \leq i, j \leq N-1$ . These eigenvalues are clearly in the interval  $[0, 8)$ .  $\square$

**Lemma 4.3** *For the anisotropic total variation (27), the corresponding operator  $H$  is nonexpansive whenever*

$$\lambda \leq \frac{1}{4 \sin^2 \frac{(N-1)\pi}{2N}}. \quad (32)$$

Proof: By Lemma 4.2, the largest eigenvalue of  $B^t B$  is  $4 - 4 \cos \frac{(N-1)\pi}{N} = 8 \sin^2 \frac{(N-1)\pi}{2N}$ . It follows that  $\|I - \lambda B^t B\|_2 \leq 1$ , if  $\lambda$  satisfies (32). Therefore,  $H$  is nonexpansive by Lemma 3.3.  $\square$

The next proposition provides a sufficient condition for the convergence of the Picard iterations for the anisotropic total variation model.

**Proposition 4.4** *If  $\varphi$  and  $B$  are defined by (29) and (26), respectively,  $H$  defined by (16),  $\mu, \lambda > 0$  where  $\lambda$  satisfies (32), and  $\kappa \in (0, 1)$  then the Picard iteration of  $H_\kappa$  converges to a fixed point of  $H$ .*

Proof: The proof follows from Lemma 4.3 and Corollary 3.8.  $\square$

With Proposition 4.4, we have the following fixed point algorithm based on proximity operator with anisotropic TV (FP<sup>2</sup>O-ATV) for the ROF denoising model (1).

---

**Fixed Point Algorithm Based on the Proximity Operator with ATV (FP<sup>2</sup>O-ATV)**

**Given:** noisy image  $x$ ;  $\lambda = \frac{1}{4} \sin^{-2} \frac{(N-1)\pi}{2N}$ ,  $\mu > 0$ ,  $\kappa \in (0, 1)$

**Initialization:**  $v^0 = 0$

**For**  $n = 0, 1, 2, \dots$

$$v^{n+1} \leftarrow \kappa v^n + (1 - \kappa)(I - \text{prox}_{\frac{\mu}{\lambda} \|\cdot\|_1})(Bx + (I - \lambda B B^t)v^n)$$

**End**

**Write the output of  $v^n$  from the above loop as  $v^\infty$  and compute**

$$\text{prox}_{\varphi \circ B} x = x - \lambda B^t v^\infty$$

As given in Example 2.4, the operator  $\text{prox}_{\frac{\mu}{\lambda}\|\cdot\|_1}$  acting on a vector can be easily computed in a coordinate-wise fashion. Therefore, the algorithm FP<sup>2</sup>O-ATV can be efficiently implemented.

Next, we compare the algorithm FP<sup>2</sup>O-ATV with the Goldstein-Osher split Bregman anisotropic TV denoising algorithm [15] which is considered to be one of the best algorithms for solving the ROF model. To this end, we recall that the Goldstein-Osher split Bregman denoising algorithm has the following form:

---

Split Bregman Anisotropic TV Denoising (see [15])

Given: noisy image  $x$ ;  $\lambda > 0$ ,  $\mu > 0$   
Initialization:  $b^0 = 0$ ,  $d^0 = 0$ , and  $x^0 = x$   
For  $n = 0, 1, 2, \dots$

$$\begin{aligned} (I + \lambda B^t B)x^{n+1} &= x - \lambda B^t(b^n - d^n) \\ d^{n+1} &\leftarrow \text{prox}_{\frac{\mu}{\lambda}\|\cdot\|_1}(Bx^{n+1} + b^n) \\ b^{n+1} &\leftarrow Bx^{n+1} + b^n - d^{n+1} \end{aligned}$$

End

---

The convergence of the above split Bregman denoising algorithm was proved in [3]. It can be seen that in each iteration one needs to solve a linear system  $(I + \lambda B^t B)x^{n+1} = x - \lambda B^t(b^n - d^n)$ . It was suggested in [15] that a one step Gauss-Seidel iteration yields a satisfactory approximation to  $x^{n+1}$ .

From the relation  $b^{n+1} = Bx^{n+1} + b^n - d^{n+1}$  in the Goldstein-Osher split Bregman denoising algorithm, we know that  $Bx^{n+1} \approx d^{n+1}$ , therefore  $B^t Bx^{n+1} \approx B^t d^{n+1} \approx B^t d^n$ , when  $n$  is large enough. With this in mind, if we simply drop the term  $\lambda B^t Bx^{n+1}$  and the term  $\lambda B^t d^n$  from the left and right hand sides of the linear system  $(I + \lambda B^t B)x^{n+1} = x - \lambda B^t(b^n - d^n)$ , this system becomes  $x^{n+1} = x - \lambda B^t b^n$ . With this modification, the split Bregman denoising algorithm reduces to the Jia-Zhao denoising algorithm [17] which can be considered as Algorithm FP<sup>2</sup>O-ATV for the case  $\kappa = 0$ . Moreover, we would like to point out that the formulation and convergence proofs of both the Goldstein-Osher split Bregman algorithm and the Jia-Zhao denoising algorithm use extensively properties of the Bregman distance. In particular, the parameter  $\lambda$  which guarantees convergence of the Jia-Zhao denoising algorithm is required to be less than  $\frac{1}{8}$  while this parameter for Algorithm FP<sup>2</sup>O-ATV to converge is relaxed to a number less than  $\frac{1}{4} \sin^{-2} \frac{(N-1)\pi}{2N}$  which is slightly bigger than  $\frac{1}{4}$ . Furthermore, as shown in Sections 2 and 3, our motivation for developing Algorithm FP<sup>2</sup>O-ATV and its convergence proof are totally different from those of the above two algorithms. Our method follows from general considerations.

## 4.2 Isotropic Total Variation Denoising

In this subsection, we consider the isotropic total variation (28). The corresponding function  $\varphi$  and the operator  $B$  are defined in (30) and (26), respectively. We will study the nonexpansivity of the resulting operator  $H$  defined by (16).

**Lemma 4.5** *For the isotropic total variation (28), the corresponding operator  $H$  maps  $\mathbb{R}^{2N^2}$  into the set*

$$\left\{ z : z \in \mathbb{R}^{2N^2}, \left\| \begin{bmatrix} z_i \\ z_{i+N^2} \end{bmatrix} \right\|_2 \leq \frac{\mu}{\lambda}, \quad i = 1, 2, \dots, N^2 \right\}.$$

Proof: The proof follows by specializing Lemma 3.5 to the current situation.  $\square$

Following the same arguments used in Lemma 4.3, we have the following result on the nonexpansivity of operator  $H$  for the isotropic total variation model.

**Lemma 4.6** *For the isotropic total variation (28), the corresponding operator  $H$  is nonexpansive whenever  $\lambda$  satisfies (32).*

Again, following the similar arguments used in Proposition 4.4, we have the following result on the convergence of the Picard iterations of  $H_\kappa$  for the isotropic total variation model.

**Proposition 4.7** *If  $\varphi$  and  $\mathcal{B}$  are defined by (30),  $H$  defined by (16),  $\mu, \lambda > 0$  where  $\lambda$  satisfies (32), and  $\kappa \in (0, 1)$  then the Picard iterations of  $H_\kappa$  converges to a fixed point of  $H$ .*

Proof: The proof follows from Lemma 4.6 and Corollary 3.8.  $\square$

As a result of Proposition 4.7, we propose the following fixed point algorithm based on the proximity operator with isotropic TV (FP<sup>2</sup>O-ITV) for the ROF model.

---

Fixed Point Algorithm Based on the Proximity Operator with ITV (FP<sup>2</sup>O-ITV)

**Given:** noisy image  $x$ ;  $\lambda = \frac{1}{4} \sin^{-2} \frac{(N-1)\pi}{2N}$ ,  $\mu > 0$ ,  $\kappa \in (0, 1)$

**Initialization:**  $v^0 = 0$

**For**  $n = 0, 1, \dots$

$$v^{n+1} \leftarrow \kappa v^n + (1 - \kappa)(I - \text{prox}_{\frac{1}{\lambda}\varphi})(Bx + (I - \lambda BB^t)v^n)$$

**End**

**Write the output of  $v^n$  from the above loop as  $v^\infty$  and compute**

$$\text{prox}_{\varphi \circ B} x = x - \lambda B^t v^\infty$$


---

As given in Example 2.5, the output of the operator  $\text{prox}_{\frac{1}{\lambda}\varphi}$  acting on a vector is computable explicitly. Accordingly, FP<sup>2</sup>O-ITV can be implemented easily.

Next, we will comment on the relationship among Algorithm FP<sup>2</sup>O-ITV, the Chambolle algorithm [4], and the Goldstein-Osher split Bregman denoising algorithm [15] for the ROF model with isotropic total variation. To this end, we first recall the Chambolle algorithm.

---

Chambolle's Denoising Algorithm (see [4])

**Given:** noisy image  $x$ ;  $\mu > 0$

**Define:**  $\Gamma$  is a diagonal matrix whose  $(i, i)$ -th entry is the absolute value of the  $i$ -th entry of the vector  $\mu BB^t p - Bx$

**Solve:**  $\mu BB^t p - Bx + \Gamma p = 0$

**Compute:**  $\text{prox}_{\varphi \circ B} x = x - \mu B^t p$

---

It is evident that  $\lambda B^t v^\infty$  in Algorithm FP<sup>2</sup>O-ITV is equal to  $\mu B^t p$  in the Chambolle algorithm. Therefore, the difference  $\lambda v^\infty - \mu p$  lies in the null space of  $B^t$ . It was proved in [4] that the vector  $\mu B^t p$  in the Chambolle algorithm is the projection of  $x$  into the set

$$C = \mu \left\{ B^t z : z \in \mathbb{R}^{2N^2}, \left\| \begin{bmatrix} z_i \\ z_{i+N^2} \end{bmatrix} \right\|_2 \leq 1, \quad i = 1, 2, \dots, N^2 \right\},$$

and therefore, so is  $\lambda B^t v^\infty$ . Hence, both Algorithm FP<sup>2</sup>O-ITV and the Chambolle algorithm provide possible ways for computing the projection of  $x$  onto the set  $C$ , but they act differently. We further remark that Chambolle proposed a semi-implicit gradient descent algorithm for solving the nonlinear system  $\mu B B^t p - Bx + \Gamma p = 0$ . Specifically, he chooses a  $\tau > 0$  (a time step), let  $p^0 = 0$  and for any  $n \geq 0$ ,

$$p_{ij}^{n+1} = \frac{p_{ij}^n - \tau \mu^{-1} (\mu B B^t p^n - Bx)_{ij}}{1 + \tau \mu^{-1} |(\mu B B^t p^n - Bx)_{ij}|}.$$

We next recall the Goldstein-Osher split Bregman denoising algorithm for the isotropic total variation model.

---

**Split Bregman Isotropic TV Denoising (see [15])**

**Initialization:**  $\lambda > 0$ ,  $\mu > 0$ ,  $b^0 = 0$ ,  $d^0 = 0$ , and  $x^0 = x$

**For**  $n = 0, 1, \dots$

$$\begin{aligned} (I + \lambda B^t B)x^{n+1} &= x - \lambda B^t (b^n - d^n) \\ d^{n+1} &\leftarrow \text{prox}_{\frac{1}{\lambda} \varphi}(Bx^{n+1} + b^n) \\ b^{n+1} &\leftarrow Bx^{n+1} + b^n - d^{n+1} \end{aligned}$$

**End**

---

The convergence of the above split Bregman isotropic TV denoising algorithm was proved in [3, 18]. As in the split Bregman anisotropic TV denoising algorithm, in each iteration of the split Bregman isotropic TV denoising algorithm one needs to solve the linear system  $(I + \lambda B^t B)x^{n+1} = x - \lambda B^t (b^n - d^n)$ . Again, one step of the Gauss-Seidel iteration was used to obtain a satisfactory approximation to  $x^{n+1}$ , [15]. If the linear system is solved by using one step of the Gauss-Seidel iteration, the analysis of the convergence of the resulting iterative scheme seems not to be available in the literature.

Based on Proposition 2.6, we can give an alternative derivation of the fact that the sequence  $x^n$  generated by the Goldstein-Osher split Bregman denoising algorithm converges to the solution of the ROF with isotropic total variation model. This fact is stated as follows:

**Proposition 4.8** *If the sequences  $x^n$ ,  $b^n$ , and  $d^n$  in the Goldstein-Osher split Bregman isotropic total variation denoising algorithm are convergent to  $x^\infty$ ,  $b^\infty$ , and  $d^\infty$ , respectively, then the vector  $x^\infty$  is the solution of the ROF model with isotropic total variation model, i.e.,  $x^\infty = \text{prox}_{\varphi \circ B} x$ .*

*Proof:* If the sequences  $x^n$ ,  $b^n$ , and  $d^n$  converge to  $x^\infty$ ,  $b^\infty$ , and  $d^\infty$ , respectively, then it follows that  $d^\infty = Bx^\infty$ ,  $x^\infty = x - \lambda B^t b^\infty$ , and  $Bx^\infty = \text{prox}_{\frac{1}{\lambda} \varphi}(Bx^\infty + b^\infty)$ . Hence, by Proposition 2.6, we obtain that  $b^\infty \in \frac{1}{\lambda} \partial \varphi(Bx^\infty)$  which together with equation  $x^\infty = x - \lambda B^t b^\infty$  and the chain rule (13) implies

$$x^\infty \in x - \partial(\varphi \circ B)(x^\infty).$$

Thus,  $x^\infty$  must be the solution of the ROF isotropic total variation model. The uniqueness solution of the ROF model leads to  $x^\infty = \text{prox}_{\varphi \circ B} x$ .  $\square$

Summarizing the results we obtained in this subsection, we know that the vectors  $\lambda B^t b^\infty$  from the split Bregman iteration,  $\mu B^t p$  from the Chambolle algorithm, and  $\lambda B^t v^\infty$  from FP<sup>2</sup>O-ITV are the same and are the projection of  $x$  on the set  $C$ , but they are computed in the different ways.

Adopting arguments similar to those used for the split Bregman anisotropic TV algorithm, we simply discard the terms  $\lambda B^t Bx^{n+1}$  and  $\lambda B^t d^n$  from the left and right hand sides of the linear

system  $(I + \lambda B^t B)x^{n+1} = x - \lambda B^t(b^n - d^n)$ , thereby obtaining this system becomes  $x^{n+1} = x - \lambda B^t b^n$ . In this fashion, the split Bregman isotropic TV denoising algorithm becomes Algorithm FP<sup>2</sup>O-ITV in the case of  $\kappa = 0$ .

## 5 A Gauss-Seidel Variation of the Proposed Algorithms

In this section, we present a Gauss-Seidel variation of Algorithms FP<sup>2</sup>O-ATV and FP<sup>2</sup>O-ITV, which were developed in the previous section. Recall that in an iteration of Algorithm FP<sup>2</sup>O-ATV or FP<sup>2</sup>O-ITV we do not make use of an updated vector until its all components are updated. To speed up convergence of the iteration, we propose to make use of an updated component immediately once it becomes available, not to wait until the updating of the entire vector is complete.

To present a Gauss-Seidel variation of Algorithms FP<sup>2</sup>O-ATV and FP<sup>2</sup>O-ITV, we specialize the linear transformation  $A$  in (15) to our current setting. That is,  $A : \mathbb{R}^{2N^2} \rightarrow \mathbb{R}^{2N^2}$  for a given  $x \in \mathbb{R}^{N^2}$  is defined by

$$Av = Bx + (I - \lambda BB^t)v, \quad \text{for all } v \in \mathbb{R}^{2N^2}.$$

By (26), we can rewrite the above equation as

$$Av = \begin{bmatrix} (I \otimes D)x + v_U - \lambda((I \otimes DD^t)v_U + (D^t \otimes D)v_L) \\ (D \otimes I)x + v_L - \lambda((D \otimes D^t)v_U + (DD^t \otimes I)v_L) \end{bmatrix}. \quad (33)$$

Due to the fact that images are usual represented as matrices, it will be useful to rewrite vectors in Algorithms FP<sup>2</sup>O-ATV and FP<sup>2</sup>O-ITV in matrix forms. To this end, any  $N^2$ -dimensional vector  $y$  appeared in Algorithms FP<sup>2</sup>O-ATV and FP<sup>2</sup>O-ITV will be viewed as a  $N \times N$  matrix  $Y$ . Specifically, the elements of  $Y$  are generated by the formula  $Y_{ij} = y_{i+(j-1)N}$ ,  $i, j = 1, 2, \dots, N$ . For any  $2N^2$ -dimensional vector  $z$ , the vector  $z_U$  formed from the first  $N^2$  elements of  $z$  will be viewed as a  $N \times N$  matrix  $Z_U$  and the vector  $z_L$  formed from the rest  $N^2$  elements of  $z$  will be viewed as another  $N \times N$  matrix  $Z_L$  in the above way. With this connection, the  $N^2$ -dimensional vectors  $(I \otimes D)x$ ,  $(D \otimes I)x$ ,  $v_U$ ,  $v_L$ ,  $(I \otimes DD^t)v_U$ ,  $(D \otimes D^t)v_U$ ,  $(D^t \otimes D)v_L$ , and  $(DD^t \otimes I)v_L$  in (33) can be viewed as the  $N \times N$  matrices  $DX$ ,  $XD^t$ ,  $V_U$ ,  $V_L$ ,  $DD^tV_U$ ,  $D^tV_UD^t$ ,  $DV_LD$ ,  $V_LDD^t$ , respectively. Therefore, the  $N \times N$  matrix corresponding to the upper half of the vector  $Av$  in (33) is

$$DX + V_U - \lambda(DD^tV_U + DV_LD) \quad (34)$$

and the  $N \times N$  matrix corresponding to the lower half of the vector  $Av$  in (33) is

$$XD^t + V_L - \lambda(D^tV_UD^t + V_LDD^t). \quad (35)$$

With these preparation, we are ready to present the Gauss-Seidel variation of Algorithms FP<sup>2</sup>O-ATV and FP<sup>2</sup>O-ITV. We first present the Gauss-Seidel variation of Algorithm FP<sup>2</sup>O-ATV. Taking the block structure of  $Av$  as demonstrated in (33), (34), (35) and the idea of Gauss-Seidel explained earlier into consideration in Algorithm FP<sup>2</sup>O-ATV, we have the following Gauss-Seidel variation (FP<sup>2</sup>O-ATV-GS).

---

FP<sup>2</sup>O-ATV – GS

**Given:** noisy image  $x$ ;  $\lambda = \frac{1}{4} \sin^{-2} \frac{(N-1)\pi}{2N}$ ,  $\mu > 0$ ,  $\kappa \in (0, 1)$   
**Initialization:**  $V_U^0 = V_L^0 = 0$

Auxiliary Parameters:  $p$  and  $q$   
For  $n = 0, 1, 2, \dots$   
  For  $j = 1$  to  $N$   
    For  $i = 1$  to  $N$   
       $p = (V_U^n)_{ij} - \lambda(DD^tV_U^n)_{ij} - \lambda(DV_L^nD)_{ij} + (DX)_{ij}$   
       $q = (V_L^n)_{ij} - \lambda(D^tV_U^nD^t)_{ij} - \lambda(V_L^nDD^t)_{ij} + (XD^t)_{ij}$   
       $(V_U^n)_{ij} := \kappa(V_U^n)_{ij} + (1 - \kappa)(I - \text{prox}_{\frac{\mu}{\lambda}|\cdot|})(p)$   
       $(V_L^n)_{ij} := \kappa(V_L^n)_{ij} + (1 - \kappa)(I - \text{prox}_{\frac{\mu}{\lambda}|\cdot|})(q)$   
    End  
  End  
 $V_U^{n+1} := V_U^n$   
 $V_L^{n+1} := V_L^n$   
End  
Write the outputs of  $V_U^n$  and  $V_L^n$  from the most outer loop as  $V_U^\infty$   
and  $V_L^\infty$ , respectively, compute  $\text{prox}_{\varphi \circ A}X = X - \lambda(D^tV_U^\infty + V_L^\infty D)$

---

Likewise, we present the Gauss-Seidel variation of FP<sup>2</sup>O-ITV.

---

FP<sup>2</sup>O-ITV – GS

Given: noisy image  $x$ ;  $\lambda = \frac{1}{4} \sin^{-2} \frac{(N-1)\pi}{2N}$ ,  $\mu > 0$ ,  $\kappa \in (0, 1)$   
Initialization:  $V_U^0 = V_L^0 = 0$   
Auxiliary Parameters:  $p$  and  $q$   
For  $n = 0, 1, 2, \dots$   
  For  $j = 1$  to  $N$   
    For  $i = 1$  to  $N$   
       $p = (V_U^n)_{ij} - \lambda(DD^tV_U^n)_{ij} - \lambda(DV_L^nD)_{ij} + (DX)_{ij}$   
       $q = (V_L^n)_{ij} - \lambda(D^tV_U^nD^t)_{ij} - \lambda(V_L^nDD^t)_{ij} + (XD^t)_{ij}$   
       $\begin{bmatrix} (V_U^n)_{ij} \\ (V_L^n)_{ij} \end{bmatrix} = \kappa \begin{bmatrix} (V_U^n)_{ij} \\ (V_L^n)_{ij} \end{bmatrix} + (1 - \kappa)(I - \text{prox}_{\frac{\mu}{\lambda}\|\cdot\|_2}) \begin{bmatrix} p \\ q \end{bmatrix}$   
    End  
  End  
 $V_U^{n+1} := V_U^n$   
 $V_L^{n+1} := V_L^n$   
End  
Write the outputs of  $V_U^n$  and  $V_L^n$  from the most outer loop as  $V_U^\infty$   
and  $V_L^\infty$ , respectively, compute  $\text{prox}_{\varphi \circ A}X = X - \lambda(D^tV_U^\infty + V_L^\infty D)$

---

## 6 Numerical Experiments

In this section, we present numerical results for our proposed algorithms. Specifically, we compare the computational performance of Algorithms FP<sup>2</sup>O-ATV-GS and FP<sup>2</sup>O-ITV-GS with that of the split Bregman algorithm. For simplicity of presentation, we will simply drop the letters ‘‘GS’’ from Algorithms FP<sup>2</sup>O-ATV-GS and FP<sup>2</sup>O-ITV-GS.

In our experiments, we choose images of ‘‘Cameraman’’ and ‘‘Lena’’ with sizes  $256 \times 256$  and  $512 \times 512$ , respectively, as two original images  $f$ . These images are shown in Figure 1. The noisy

images are modeled as

$$x = f + \eta, \quad \text{and} \quad \eta \sim N(0, \sigma^2)$$

with  $N(0, \sigma^2)$  being Gaussian noise. The quality of denoised images  $\tilde{x}$  obtained from various denoising algorithms is evaluated by the peak-signal-to-noise ratio

$$\text{PSNR} := 20 \log_{10} \left( \frac{255}{\|f - \tilde{x}\|_2} \right).$$

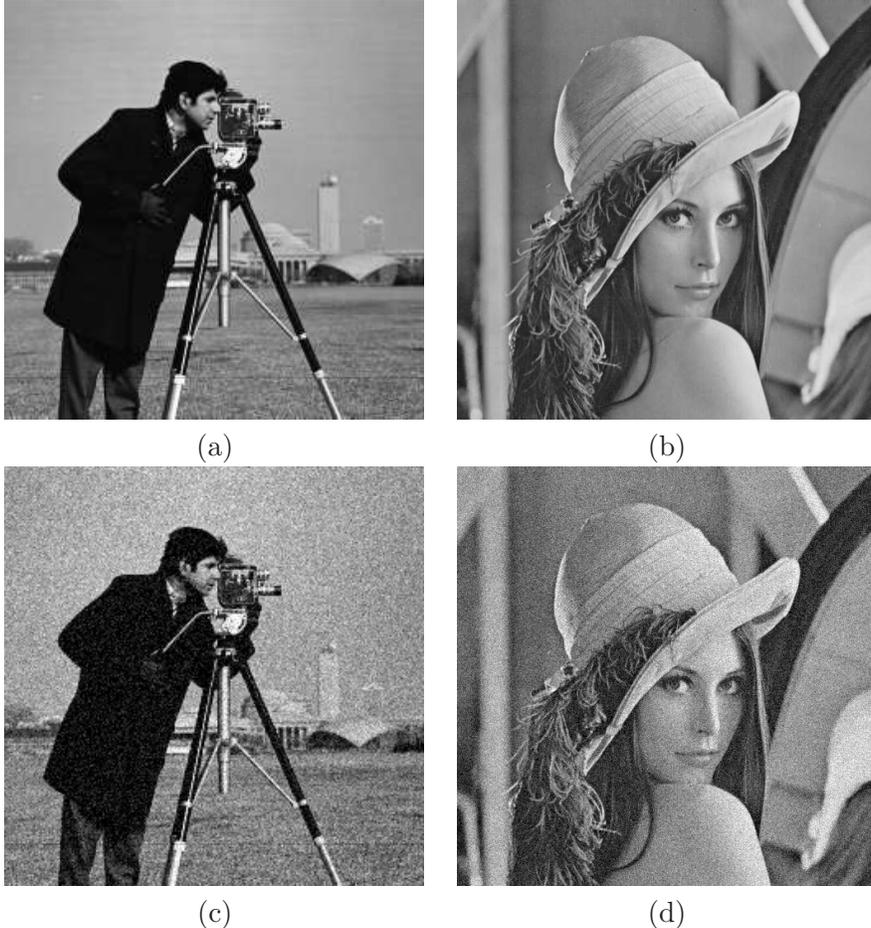


Figure 1: (a) The original image of “Cameraman”; (b) The original image of “Lena”; (c) The noisy image of “Cameraman” with Gaussian noise at level  $\sigma = 20$ ; and (d) The noisy image of “Lena” with Gaussian noise at level  $\sigma = 20$ .

For parameters in the proposed algorithms FP<sup>2</sup>O-ATV and FP<sup>2</sup>O-ITV, we choose  $\kappa = 0.0001$  and  $\lambda = \frac{1}{4} \sin^{-2} \frac{(N-1)\pi}{2N}$  for an  $N \times N$  test image. In the split Bregman anisotropic and isotropic TV denoising algorithms, we choose  $\lambda = 2$  as suggested in [15]. Iterations in these algorithms are terminated when the following condition is satisfied

$$\frac{\|x^n - x^{n+1}\|_2}{\|x^{n+1}\|_2} \leq \text{TOL},$$

where TOL denotes a prescribed tolerance value. In our experiments, we set  $\text{TOL} = 0.9 \times 10^{-3}$ .

In our first experiment, we apply Algorithm FP<sup>2</sup>O-ITV to noisy image of “Cameraman” with noise level  $\sigma = 20$ . The PSNR values for the denoised images and the number of iterations used for various values of  $\mu$  for Algorithm FP<sup>2</sup>O-ITV are listed in Table 1. The resulting denoised images obtained from Algorithm FP<sup>2</sup>O-ITV are shown in Figure 2. From Figure 2(a)-(b), we observe that the denoised images with  $\mu^{-1} = .02, .04$  are too smooth, which are cartoon-like images whose different objects are separated by sharp boundaries. The denoised image with  $\mu^{-1} = .06$  shown in Figure 2(c) has good visual quality. The denoised image with  $\mu^{-1} = .08$  shown in Figure 2(d) is a bit noisy, not as good as that with  $\mu^{-1} = .06$ . We obtain similar numerical and visual results for Algorithm FP<sup>2</sup>O-ATV.

Table 1: Numerical results for Algorithm FP<sup>2</sup>O-ITV.

$\mu^{-1}$	.02	.04	.06	.08
PSNR (dB)	24.73	27.42	28.67	28.82
Number of Iterations (Ite)	23	16	13	11

In our second experiment, we compare performance of the proposed algorithms with that of the split Bregman anisotropic TV algorithm (SB-ATV) and the split Bregman isotropic TV algorithm (SB-ITV). In this experiment, we choose the noisy images of “Cameraman” and “Lena” with noise levels  $\sigma = 15$  and  $\sigma = 25$ . Numerical results of the four algorithms for the images of “Cameraman” with  $\sigma = 15$  and  $\sigma = 25$  are reported in Table 2. In Table 2, the pair  $(\cdot, \cdot)$  is used to report both the PSNR value (the first number) and the number of iterations (the second number). From Table 2, we observe that under the same stopping criterion, for various values of  $\mu$ , FP<sup>2</sup>O-ATV (resp. FP<sup>2</sup>O-ITV) always uses less number of iterations than SB-ATV (resp. SB-ITV) does. At noise level  $\sigma = 15$ , the performance of FP<sup>2</sup>O-ATV and FP<sup>2</sup>O-ITV is superior to that of SB-ATV and SB-ITV, respectively, in terms of PSNR values and number of iterations for each fixed values of  $\mu$ . At noise level  $\sigma = 25$ , we observe a similar scenario for small values of  $\mu^{-1}$ . The resulting denoised images obtained from these four algorithms are shown in Figure 3.

Numerical results of the four algorithms for the images of “Lena” with  $\sigma = 15$  and  $\sigma = 25$  are reported in Table 3, from which we have derived similar observations. We show the denoised images in Figure 4.

Finally, we report in Tables 4 and 5 for the images of “Cameraman” and “Lena”, respectively, the numbers of iterations used in the four algorithms. These numbers are obtained with the same stopping criterion applied to the four algorithms. The numerical results indicate that the number of iterations used in Algorithms FP<sup>2</sup>O-ATV and FP<sup>2</sup>O-ITV reduces as the value of  $\mu$  decreases. However, the number of iterations used in Algorithms SB-ATV and SB-ITV does not seem to have the pattern.

## 7 Concluding Remarks

This paper studies the proximity operator of the total variation for the ROF denoising model. By making use of the fact that the total variation is the composition of an  $\ell^1$  or  $\ell^2$  norm (depending on ATV or ITV) with the first order difference operator, we express the proximity operator of the total variation in terms of the proximity operator of the norm via a fixed point equation. This naturally leads to fixed point algorithms for finding an approximate solution of the ROF denoising model. We identify connections of the proposed proximity operator based fixed point algorithms with several



Figure 2: Denoising the noisy image of “Cameraman” with Gaussian noise at level  $\sigma = 20$  by FP<sup>2</sup>O-ITV. (a)  $\mu^{-1} = .02$ , PSNR = 24.73dB, Ite = 23; (b)  $\mu^{-1} = .04$ , PSNR = 27.42dB, Ite = 16; (c)  $\mu^{-1} = .06$ , PSNR = 28.67dB, Ite = 13; and (d)  $\mu^{-1} = .08$ , PSNR = 28.82dB, Ite = 11;

recently developed algorithms known in the literature, including the split Bregman algorithm. This fixed point framework for the ROF denoising model has advantages in both theoretical analysis and algorithmic development. On one hand, it provides a platform to make use of rich ideas in the well-established fixed point theory for algorithmic development and on the other hand it offers convenient convergence analysis tools. Numerical experiments presented in this paper confirm that the proposed algorithms perform favorably and are promising.

**Acknowledgment:** We would like to thank Drs. Raymond Chan, Jerome Darbon, Wotao Yin, Junfeng Yang, and Xiaoqun Zhang for their valuable comments when we presented this work at the International Conference on Mathematical Methods for Imaging held at Sun Yat-sen University, in Guangzhou, China, August 4th-6th, 2010. We also thank Professor Rong-Qing Jia for sharing his paper [17] with us, which influenced our viewpoint on split Bregman iterations.

## References

- [1] G. Aubert and L. Vese. A variational method in image recovery. *SIAM Journal on Numerical Analysis*, 34(5):1948–1979, 1997.

Table 2: Numerical results of the FP<sup>2</sup>O-ATV, the FP<sup>2</sup>O-ITV, SB-ATV, and SB-ITV for the images of “Cameranman”.

$\mu^{-1}$	FP <sup>2</sup> O-ATV	FP <sup>2</sup> O-ITV	SB-ATV	SB-ITV
Gaussian noise at level $\sigma = 15$				
.02	(24.17, 26)	(24.75, 23)	(24.15, 28)	(24.71, 28)
.03	(25.72, 21)	(26.37, 18)	(25.68, 27)	(26.32, 26)
.04	(26.85, 18)	(27.53, 16)	(26.79, 25)	(27.46, 23)
.05	(27.73, 16)	(28.41, 14)	(27.65, 23)	(28.30, 20)
.06	(28.43, 15)	(29.11, 13)	(28.31, 20)	(28.94, 18)
.07	(29.01, 14)	(29.65, 12)	(28.86, 19)	(29.47, 18)
.08	(29.48, 13)	(30.03, 11)	(29.32, 19)	(29.85, 18)
.09	(29.85, 12)	(30.28, 11)	(29.69, 19)	(30.10, 18)
.1	(30.12, 12)	(30.38, 10)	(29.97, 19)	(30.25, 19)
Gaussian noise at level $\sigma = 25$				
.02	(24.12, 27)	(24.70, 23)	(24.11, 31)	(24.69, 29)
.03	(25.61, 22)	(26.24, 19)	(25.60, 26)	(26.23, 24)
.04	(26.63, 19)	(27.21, 17)	(26.61, 22)	(27.17, 18)
.05	(27.33, 17)	(27.70, 15)	(27.29, 19)	(27.63, 17)
.06	(27.71, 16)	(27.70, 13)	(27.67, 19)	(27.68, 18)
.07	(27.77, 15)	(27.40, 12)	(27.73, 19)	(27.41, 19)
.08	(27.60, 14)	(26.93, 10)	(27.59, 20)	(26.99, 20)
.09	(27.29, 13)	(26.45, 9)	(27.31, 20)	(26.52, 21)
.1	(26.91, 12)	(25.98, 9)	(26.96, 21)	(26.08, 21)

- [2] D. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, Massachusetts, 2003.
- [3] J.-F. Cai, S. Osher, and Z. Shen. Split Bregman methods and frame based image restoration. *Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal*, 2:337–369, 2009.
- [4] A. Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 20:89–97, 2004.
- [5] A. Chambolle, R. DeVore, N.-Y. Lee, and B. Lucier. Nonlinear wavelet image processing: Variational problems, compression, and noise removal through wavelet shrinkage. *IEEE Transactions on Image Processing*, 7:319–335, 1998.
- [6] T. Chan, G. H. Golub, and P. Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM Journal on Scientific Computing*, 20(6):1964–1977, 1999.
- [7] T. Chan and J. Shen. *Image Processing and Analysis: Variational, PDE, Wavelet, and Stochastic Methods*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2005.
- [8] P. Combettes and J. Luo. An adaptive level set method for nondifferentiable constrained image recovery. *IEEE Transactions on Image Processing*, 11:1295–1304, 2002.
- [9] P. Combettes and V. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal*, 4:1168–1200, 2005.
- [10] J. Darbon and M. Sigelle. Image restoration with discrete constrained total variation part i: Fast and exact minimization. *Journal of Mathematical Imaging and Vision*, 26:261–276, 2006.

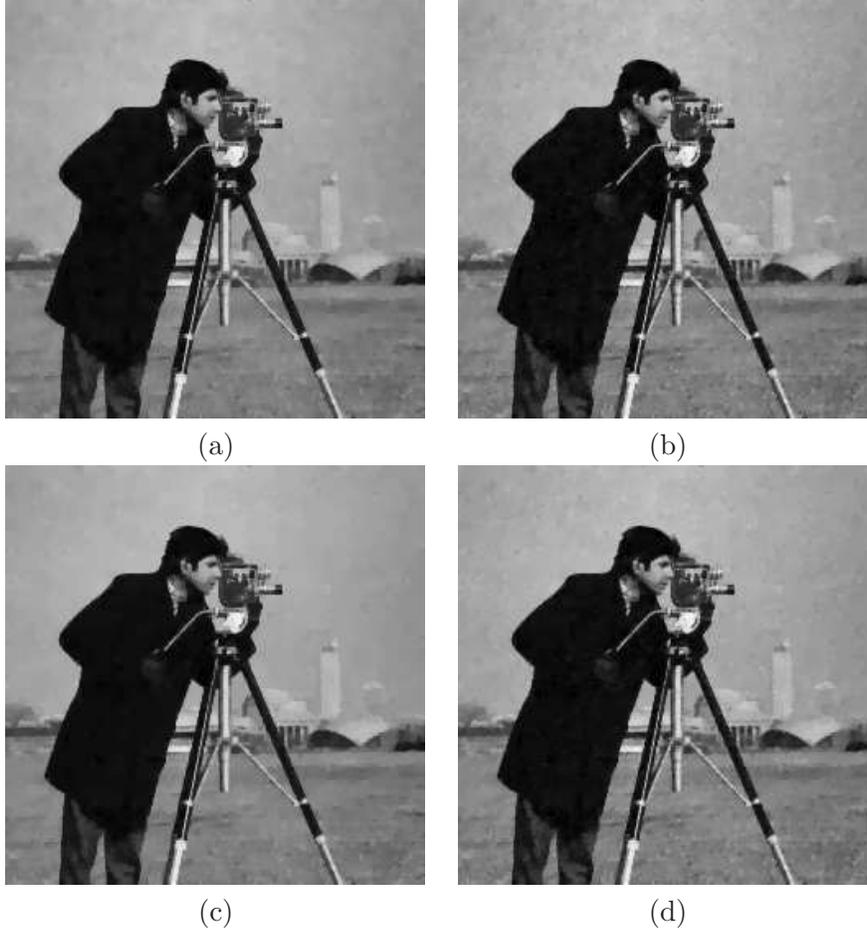


Figure 3: Denoising the noisy image of “Cameraman” with Gaussian noise at level  $\sigma = 20$  by using (a) FP<sup>2</sup>O-ATV (PSNR = 28.61dB, Ite = 14); (b) FP<sup>2</sup>O-ITV (PSNR = 28.86dB, Ite = 12); (c) SB-ATV (PSNR = 28.53dB, Ite = 19); and (d) SB-ITV (PSNR = 28.78dB, Ite = 18). The regularization  $\mu$  is  $1/0.07$ .

- [11] D. de Figueiredo. Topics in nonlinear functional analysis. Number 48, University of Maryland, College Park, MD, 1967.
- [12] D. Donoho. Compressed sensing. *IEEE Transaction on Information Theory*, 52(4):1289–1306, 2006.
- [13] D. Donoho and I. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81:425–455, 1994.
- [14] D. Goldfarb and W. Yin. Second-order cone programming methods for total variation based image restoration. *SIAM Journal of Scientific Computing*, 27(2):622–645, 2005.
- [15] T. Goldstein and S. Osher. The split Bregman method for  $\ell^1$  regularization problems. *SIAM Journal on Imaging Sciences*, 2:323–343, 2009.
- [16] M. Hintermuller and G. Stadler. An feasible primal-dual algorithm for TV-based inf-convolution-type image restoration. *SIAM Journal on Scientific Computing*, 28:1–23, 2006.

Table 3: Numerical results of the FP<sup>2</sup>O-ATV, the FP<sup>2</sup>O-ITV, SB-ATV, and SB-ITV for the images of “Lena”.

$\mu^{-1}$	FP <sup>2</sup> O-ATV	FP <sup>2</sup> O-ITV	SB-ATV	SB-ITV
Gaussian noise at level $\sigma = 15$				
.02	(27.35, 33)	(28.01, 29)	(27.12, 29)	(27.79, 27)
.03	(28.73, 26)	(29.45, 22)	(28.53, 26)	(29.27, 24)
.04	(29.72, 21)	(30.46, 18)	(29.55, 23)	(30.33, 21)
.05	(30.45, 19)	(31.20, 16)	(30.34, 21)	(31.11, 18)
.06	(31.02, 17)	(31.73, 14)	(30.95, 18)	(31.68, 15)
.07	(31.47, 15)	(32.08, 13)	(31.42, 16)	(32.06, 14)
.08	(31.80, 14)	(32.26, 12)	(31.76, 15)	(32.26, 13)
.09	(32.01, 13)	(32.29, 11)	(32.00, 15)	(32.33, 13)
.1	(32.13, 13)	(32.17, 10)	(32.14, 15)	(32.28, 13)
Gaussian noise at level $\sigma = 25$				
.02	(27.33, 32)	(27.97, 28)	(27.10, 30)	(27.77, 28)
.03	(28.61, 25)	(29.26, 22)	(28.45, 25)	(29.16, 22)
.04	(29.43, 21)	(29.96, 18)	(29.35, 20)	(29.95, 17)
.05	(29.87, 19)	(30.07, 16)	(29.86, 17)	(30.13, 15)
.06	(29.94, 17)	(29.64, 14)	(29.97, 16)	(29.79, 14)
.07	(29.67, 15)	(28.90, 12)	(29.76, 16)	(29.16, 15)
.08	(29.16, 14)	(28.09, 11)	(29.32, 17)	(28.44, 14)
.09	(28.56, 13)	(27.32, 10)	(28.80, 17)	(27.74, 17)
.1	(27.94, 12)	(26.65, 9)	(28.23, 18)	(27.04, 19)

- [17] R.-Q. Jia and H. Zhao. A fast algorithm for the total variation model of image denoising. *Advances in Computational Mathematics*, 33:231–241, 2010.
- [18] R.-Q. Jia, H. Zhao, and W. Zhao. Convergence analysis of the Bregman method for the variational model of image denoising. *Applied and Computational Harmonic Analysis*, 27:367–379, 2009.
- [19] Y. Li and F. Santosa. A computational algorithm for minimizing total variation in image restoration. *IEEE Transactions on Image Processing*, 5:987–995, 1996.
- [20] S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, New York, 2nd edition, 1999.
- [21] 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.
- [22] J.-J. Moreau. Proximité et dualité dans un espace hilbertien. *Bull. Soc. Math. France*, 93:273–299, 1965.
- [23] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103:127–152, 2005.
- [24] Z. Opial. Weak convergence of the sequence of successive approximations for nonexpansive mappings. *Bulletin American Mathematical Society*, 73:591–597, 1967.



Figure 4: Denoising the noisy image of “Lena” with Gaussian noise at level  $\sigma = 20$  by using (a) FP<sup>2</sup>O-ATV (PSNR = 30.66dB, Ite = 17); (b) FP<sup>2</sup>O-ATV (PSNR = 31.04dB, Ite = 14); (c) SB-ATV (PSNR = 30.64dB, Ite = 17); and (d) SB-ITV (PSNR = 31.07dB, Ite = 14). The regularization  $\mu$  is 1/0.06.

- [25] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal*, 4:460–489, 2005.
- [26] J. Portilla, V. Strela, M. Wainwright, and E. P. Simoncelli. Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Transactions on Image Processing*, 12(11):1338–1351, November 2003.
- [27] L. Rudin and S. Osher. Total variation based image restoration with free local constraints. In *IEEE International Conference on Image Processing*, pages 31–35, 1994.
- [28] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.
- [29] C. Vogel and M. Oman. Iterative methods for total variation denoising. *SIAM Journal on Scientific Computing*, 17(1):227–238, 1996.

Table 4: Number of iterations used in the FP<sup>2</sup>O-ATV, the FP<sup>2</sup>O-ITV, SB-ATV, and SB-ITV for the image of “Cameranman” corrupted with Gaussian noise at level  $\sigma = 20$  for various  $\mu$ .

Algorithm	$\mu^{-1}$																	
	.02	.03	.04	.05	.06	.07	.08	.09	.1	.2	.3	.4	.5	.6	.7	.8	.9	1
FP <sup>2</sup> O-ATV	26	21	18	16	15	14	13	13	12	8	6	5	4	4	4	3	3	3
FP <sup>2</sup> O-ITV	23	19	16	15	13	12	11	10	9	5	4	3	3	3	3	3	2	2
SB-ATV	30	27	24	20	19	19	19	19	20	23	25	26	27	27	28	28	28	28
SB-ITV	29	25	21	18	18	18	18	19	20	24	26	27	27	28	28	28	28	29

Table 5: Number of iterations used in the FP<sup>2</sup>O-ATV, the FP<sup>2</sup>O-ITV, SB-ATV, and SB-ITV for the image of “Lena” corrupted with Gaussian noise at level  $\sigma = 20$  for various  $\mu$ .

Algorithm	$\mu^{-1}$																	
	.02	.03	.04	.05	.06	.07	.08	.09	.1	.2	.3	.4	.5	.6	.7	.8	.9	1
FP <sup>2</sup> O-ATV	32	25	21	19	17	15	14	13	12	8	6	5	4	4	4	3	3	3
FP <sup>2</sup> O-ITV	29	22	18	16	14	13	12	10	10	5	4	3	3	3	3	3	3	2
SB-ATV	30	26	22	19	17	16	15	15	16	20	23	24	25	26	26	27	27	27
SB-ITV	28	24	19	16	14	13	14	14	15	22	24	25	26	26	27	27	27	28

- [30] C. Vogel and M. Oman. Fast, robust total variation-based reconstruction of noisy, blurring images. *IEEE Transactions on Image Processing*, 7:813–824, 1998.
- [31] P. Weiss, L. Blanc-Feraud, and G. Aubert. Efficient schemes for total variation minimization under constraints in image processing. *SIAM Journal on Scientific Computing*, 31:2047–2080, 2009.
- [32] J. Yang, W. Yin, Y. Zhang, and Y. Wang. A fast algorithm for edge-preserving variational multichannel image restoration. *SIAM Journal on Imaging Sciences*, 2:569–592, 2009.
- [33] W. Yin, S. Osher, D. Goldfarb, and J. Darbon. Bregman iterative algorithms for  $\ell^1$  minimization with applications to compressed sensing. *SIAM Journal on Imaging Sciences*, 1:143–168, 2008.
- [34] M. Zhu and T. Chan. An efficient primal-dual hybrid gradient algorithm for total variation image restoration. Technical report, 2008.
- [35] C. Zălinescu. *Convex Analysis in General Vector Spaces*. World Scientific, River Edge, NJ, 2002.