

## A CONDITIONAL GRADIENT METHOD FOR PRIMAL-DUAL TOTAL VARIATION-BASED IMAGE DENOISING\*

ABDESLEM HAFID BENTBIB<sup>†</sup>, ABDERRAHMAN BOUHAMIDI<sup>‡</sup>, AND KARIM KREIT<sup>†</sup>

**Abstract.** In this paper, we consider the problem of image denoising by total variation regularization. We combine the conditional gradient method with the total variation regularization in the dual formulation to derive a new method for denoising images. The convergence of this method is proved. Some numerical examples are given to illustrate the effectiveness of the proposed method.

**Key words.** ill-posed problem, regularization, total variation, conditional gradient, image restoration, image denoising, convex programming

**AMS subject classifications.** 65F10, 65R30

**1. Introduction.** Image denoising is a fundamental research field in image processing. The purpose of image denoising is to recover an unknown true or original image  $\hat{u}$  (noise-free image) from an observed image  $f$  that is contaminated by unknown noise  $\eta$ . In general,  $\eta$  is Gaussian white additive noise. The model equation of image denoising may be written as

$$f(x) = \hat{u}(x) + \eta(x), \quad x = (x, y) \in \Omega,$$

where  $\Omega$  is the domain of the image. It is assumed to be a convex, connected, and bounded Lipschitz open subset of  $\mathbb{R}^2$ . One of the most popular regularization methods is Tikhonov regularization based on least-squares techniques with a parameter and a regularization operator. Tikhonov regularization has been widely studied in the literature; see, e.g., [5, 6, 7] and the references therein. The drawback of Tikhonov regularization is the fact that it does not preserve sharp discontinuities in the image; see [11]. To overcome this difficulty one may use a technique based on the minimization of the total variation norm subject to a noise constraint. The total variation model is well known and was first introduced in image processing by Rudin, Osher, and Fatemi in [19]. The model is referred to as the ROF model. We consider the Banach space

$$BV(\Omega) = \left\{ u \in L^1(\Omega) : TV(u) < \infty \right\}$$

of functions of bounded variation endowed with the norm

$$\|u\|_{BV(\Omega)} = \|u\|_{L^1} + TV(u),$$

where

$$TV(u) := \sup \left\{ \int_{\Omega} \nabla u \cdot \varphi \, dx; \quad \varphi \in \mathcal{C}_c^1(\Omega, \mathbb{R}^2), \|\varphi\|_{\infty} \leq 1 \right\}$$

represents the total variation of  $u$ ; see [1]. The notation  $\nabla u$  stands for the gradient of  $u$  defined in a distributional sense. The dot stands for the Euclidean scalar product in  $\mathbb{R}^2$ . If the

---

\*Received November 14, 2017. Accepted May 16, 2018. Published online on August 28, 2018. Recommended by Fiorella Sgallari.

<sup>†</sup>Université Cadi Ayyad, Marrakesh, Morocco ({a.bentbib, karim.krei}@uca.ac.ma).

<sup>‡</sup>Université du Littoral, L.M.P.A, 50 rue F. Buisson BP699, F-62228 Calais-Cedex, France (abderrahman.bouhamidi@lmpa.univ-littoral.fr).

gradient  $\nabla u$  of  $u$  belongs to the space  $L^1(\Omega, \mathbb{R}^2)$ , then the total variation of  $u$  is given by the expression

$$TV(u) = \int_{\Omega} |\nabla u(x)| dx,$$

where  $|\cdot|$  stands for the Euclidean norm in  $\mathbb{R}^2$  and  $dx$  stands for the measure  $dxdy$ . The ROF model eliminates noise from images and preserve edges by minimizing the functional

$$(1.1) \quad \min_{u \in BV(\Omega)} TV(u)$$

subject to the constraint

$$(1.2) \quad \|f - u\|_2^2 = \sigma^2 |\Omega|^2,$$

where  $\|\cdot\|_2$  denotes the 2-norm in  $L^2(\Omega)$ ,  $\sigma$  is the standard derivation of the noise  $\eta$ , and  $|\Omega|$  is the measure of the domain  $\Omega$ . Applying a Lagrange multiplier to (1.1) and (1.2) leads to the minimization problem

$$(1.3) \quad \min_{u \in BV(\Omega)} \left( TV(u) + \frac{\lambda}{2} \|u - f\|_2^2 \right),$$

where  $\lambda > 0$  is a Lagrange multiplier. As in Tikhonov regularization, the parameter  $\lambda$  may be viewed as a regularization parameter. The functional in (1.3) has a unique minimizer in the Banach space  $BV(\Omega)$ .

A direct method for finding the minimizer of the functional (1.3) is to solve its associated Euler-Lagrange equation. Due to the presence of a highly nonlinear and non-differentiable term, an application of the direct method is not an easy task; see [11]. To avoid the difficulties of solving the Euler-Lagrange equation, Chan, Golub, and Mulet (CGM) were the first to propose in [11] a primal-dual method based on the duality principle and applied it to the TV-regularization for image restoration.

In this paper, we propose to use the conditional gradient method based on the dual formulation to derive a new method for solving the problem (1.3). The conditional gradient algorithm, called the Frank-Wolfe method, is well known and was introduced in 1956; see [14, 20]. Such a technique was successfully applied in [4] in the case of Tikhonov regularization. Many variants of the gradient algorithm were proposed for the total variation in image reconstruction; see, for instance, [21, 22, 23]. To the best of our knowledge, the Frank-Wolfe algorithm has not been used previously in the literature in the case of primal-dual total variation-based image denoising. We will show how it is also possible to adopt the conditional gradient method to a TV-denoising problem. We will prove the convergence of this method, and we will give some numerical examples to illustrate the effectiveness of our proposed method.

The paper is organized as follow: In Section 2, we recall the dual total variation formulation and its discrete form. In Section 3, we introduce our proposed method applied to the dual discrete formulation. The convergence of this method is established in Section 4. In Section 5, we show how to select the Lagrange multiplier parameter automatically. Section 6 is devoted to numerical tests illustrating the effectiveness of the proposed method. We also compare our method to state-of-the-art algorithms, e.g., Chambolle's projection algorithm, the split Bregman method, and primal dual hybrid gradient algorithms.

**2. Dual total variation formulation.** The dual formulation of the ROF model was first introduced in image restoration by Chan, Golub, and Mulet [11]. In this section, we recall briefly some well-known results in the dual formulation of the total variation problem (1.3); for more details we refer the readers to [8, 11, 22] and the references therein. It is standard in convex analysis (see for instance [1, 12]) that the  $TV(u)$  semi-norm in the minimization problem (1.3) is given in the following form:

$$\int_{\Omega} |\nabla u(x)| dx = \max_{\substack{\omega \in C_c^1(\Omega, \mathbb{R}^2) \\ \|\omega\|_{\infty} \leq 1}} \int_{\Omega} \nabla u(x) \cdot \omega(x) dx = \max_{\substack{\omega \in C_c^1(\Omega, \mathbb{R}^2) \\ \|\omega\|_{\infty} \leq 1}} \int_{\Omega} -u(x) \nabla \cdot \omega(x) dx.$$

The function  $\omega : \Omega \rightarrow \mathbb{R}^2$  is the dual variable and  $u$  the primal one. The notation  $u \cdot v$  stands for the standard scalar product of two vectors  $u$  and  $v$  in  $\mathbb{R}^2$ . The space  $C_c^1(\Omega, \mathbb{R}^2)$  denotes the space of functions  $v : \Omega \rightarrow \mathbb{R}^2$  of class  $C^1$  that are compactly supported on  $\Omega$ . The gradient of  $u$  is denoted by  $\nabla u$  and  $\nabla \cdot \omega$  is the divergence of  $\omega$ . It is well known that the solution  $u^*$  of the minimization problem (1.3) is given by the expression

$$(2.1) \quad u^* = f + \frac{1}{\lambda} \nabla \cdot \omega^*,$$

where the dual variable  $\omega^*$  may be regarded as the solution of the minimization problem

$$(2.2) \quad \omega^* = \arg \min_{\omega \in K} \left\| f + \frac{1}{\lambda} \nabla \cdot \omega \right\|_2^2$$

with

$$K = \{ \omega \in C_c^1(\Omega, \mathbb{R}^2) : \|\omega\|_{\infty} \leq 1 \}.$$

We recall that the norm  $\|\cdot\|_{\infty}$  is given for  $\omega = (\omega_1, \omega_2) \in C_c^1(\Omega, \mathbb{R}^2)$  by

$$\|\omega\|_{\infty} = \sup_{x \in \Omega} |\omega(x)|,$$

where  $|\omega(x)|$  stands for the Euclidean norm of  $\omega(x)$  in  $\mathbb{R}^2$ . Solving the TV-regularization problem (1.3) is equivalent to solving the dual problem (2.1)–(2.2). The main advantage of the dual formulation (2.1)–(2.2) is that the objective function in (2.2) is quadratic.

From now on, we consider the discretized formulation. The domain of the images is assumed to be a rectangle of size  $n \times m$ . The space of the images is  $\mathbb{X} = \mathbb{R}^{n \times m}$  of matrices of size  $n \times m$ . The space  $\mathbb{X}$  is equipped with the classical inner product

$$\langle u | v \rangle_{\mathbb{X}} = \text{tr}(u^T v),$$

where  $u$  and  $v$  are two matrices in  $\mathbb{X}$ ,  $\text{tr}(Z)$  denotes the trace of the square matrix  $Z$ , and  $u^T$  is the transpose of the matrix  $u$ . The associated norm is the well-known Frobenius norm denoted here by

$$\|u\|_{\mathbb{X}} = \sqrt{\langle u | u \rangle_{\mathbb{X}}}.$$

To define the discrete total variation, we recall some definitions and basic result from [8]. The image  $u$  is represented as a matrix in  $\mathbb{X}$ , where  $u_{i,j}$ , the component of  $u$ , represents the value of  $u$  at the point indexed by  $(i, j)$ . The discrete gradient operator, whose two

components at each point  $(i, j)$  are two matrices  $\nabla_1 u$  and  $\nabla_2 u$  in  $\mathbb{X}$ , is defined by their components  $(\nabla_1 u)_{i,j}$  and  $(\nabla_2 u)_{i,j}$  as follows:

$$\begin{aligned}
 (\nabla_1 u)_{i,j} &= \begin{cases} u_{i+1,j} - u_{i,j} & \text{if } i < n, \quad j = 1, \dots, m, \\ 0 & \text{if } i = n, \end{cases} \\
 (\nabla_2 u)_{i,j} &= \begin{cases} u_{i,j+1} - u_{i,j} & \text{if } j < m, \quad i = 1, \dots, n, \\ 0 & \text{if } j = m. \end{cases}
 \end{aligned}$$

We have  $\nabla u = (\nabla_1 u, \nabla_2 u)$ , and the discrete gradient, also denoted by  $\nabla$ , may be seen as an operator  $\nabla : \mathbb{X} \rightarrow \mathbb{Y}$ , where  $\mathbb{Y} = \mathbb{R}^{n \times m \times 2} = \mathbb{X} \times \mathbb{X}$ . The space  $\mathbb{Y}$  is endowed with the inner product and its associated norm given by

$$\langle p|q \rangle_{\mathbb{Y}} = \langle p^1|q^1 \rangle_{\mathbb{X}} + \langle p^2|q^2 \rangle_{\mathbb{X}}, \quad \|p\|_{\mathbb{Y}} = \sqrt{\langle p|p \rangle_{\mathbb{Y}}},$$

for all  $p = (p^1, p^2)$  and  $q = (q^1, q^2)$  in  $\mathbb{Y}$ . The discrete total variation of  $u \in \mathbb{X}$  is defined by

$$(2.3) \quad TV(u) = \sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}} \sqrt{|(\nabla_1 u)_{i,j}|^2 + |(\nabla_2 u)_{i,j}|^2}.$$

Therefore, the discrete total variation problem (1.3) is defined as

$$(2.4) \quad \min_{u \in \mathbb{X}} \left[ TV(u) + \frac{\lambda}{2} \|u - f\|_{\mathbb{X}}^2 \right],$$

where  $u$  and  $f$  are the discretizations of the related continuous variables. The discrete divergence operator  $\text{div} : \mathbb{Y} \rightarrow \mathbb{X}$ , also denoted by  $\nabla \cdot$ , is defined such that  $\text{div} = -\nabla^*$ , that is, for any  $\omega \in \mathbb{Y}$  and  $u \in \mathbb{X}$ ,

$$(2.5) \quad \langle -\nabla \cdot \omega | u \rangle_{\mathbb{X}} = \langle \omega | \nabla u \rangle_{\mathbb{Y}}.$$

The discrete divergence is then (see [8])

$$(2.6) \quad (\nabla \cdot \omega)_{i,j} = \begin{cases} \omega_{i,j}^1 - \omega_{i-1,j}^1 & \text{if } 1 < i < n, \\ \omega_{i,j}^1 & \text{if } i = 1, \\ -\omega_{i-1,j}^1 & \text{if } i = n, \end{cases} + \begin{cases} \omega_{i,j}^2 - \omega_{i,j-1}^2 & \text{if } 1 < j < m, \\ \omega_{i,j}^2 & \text{if } j = 1, \\ -\omega_{i,j-1}^2 & \text{if } j = m. \end{cases}$$

Hence, the discrete form of the problem (2.2) may be written as

$$(2.7) \quad \min_{\omega \in K} \left\| f + \frac{1}{\lambda} \nabla \cdot \omega \right\|_{\mathbb{X}}^2,$$

where  $K$  is the discrete version of the closed convex subset  $K$ , denoted by the same symbol. We have

$$(2.8) \quad K = \{ \omega = (\omega^1, \omega^2) \in \mathbb{Y} : \sqrt{|\omega_{i,j}^1|^2 + |\omega_{i,j}^2|^2} \leq 1, \forall i = 1, \dots, n, \forall j = 1, \dots, m \}.$$

Thus, as in continuous total variation, the solution  $u_*$  of the discrete total variation problem (2.4) is given by the following expression

$$u_* = f + \frac{1}{\lambda} \nabla \cdot \omega_*,$$

where  $\omega_*$  is the solution of the least-squares problem (2.7).

**3. The conditional gradient total variation method.** In this section, we describe the conditional gradient method for solving the convex constrained optimization problem (2.7), and we give some results for this method related to our problem. We consider the function

$$F_\lambda : K \rightarrow \mathbb{R}$$

$$\omega \mapsto F_\lambda(\omega) = \left\| f + \frac{1}{\lambda} \nabla \cdot \omega \right\|_{\mathbb{X}}^2.$$

Hence, the convex constrained minimization problem (2.7) may be rewritten as

$$(3.1) \quad \min_{\omega \in K} F_\lambda(\omega).$$

Since  $K$  given by (2.8) is a compact set in  $\mathbb{X}$  and  $F_\lambda$  is continuous on  $K$ , the problem (3.1) has a solution  $\omega_*$  in  $K$ , i.e.,

$$\min_{\omega \in K} F_\lambda(\omega) = F_\lambda(\omega_*).$$

We have the following result:

**PROPOSITION 3.1.** *The function  $F_\lambda : K \rightarrow \mathbb{R}$  is Gâteaux differentiable. Its Gâteaux derivative is*

$$(3.2) \quad F'_\lambda(\omega) = -\frac{2}{\lambda} \nabla \left( f + \frac{1}{\lambda} \nabla \cdot \omega \right),$$

and the Taylor expansion of  $F_\lambda$  at  $\omega \in K$  is given for all  $H \in \mathbb{Y}$  such that  $\omega + H \in K$  by

$$(3.3) \quad F_\lambda(\omega + H) = F_\lambda(\omega) + \langle F'_\lambda(\omega) | H \rangle_{\mathbb{Y}} + o(\|H\|_{\mathbb{Y}}),$$

where

$$o(\|H\|_{\mathbb{Y}}) \leq \frac{8}{\lambda^2} \|H\|_{\mathbb{Y}}^2.$$

*Proof.* It is straightforward that

$$F_\lambda(\omega + H) = F_\lambda(\omega) + \frac{2}{\lambda} \left\langle f + \frac{1}{\lambda} \nabla \cdot \omega \middle| \nabla \cdot H \right\rangle_{\mathbb{X}} + \frac{1}{\lambda^2} \|\nabla \cdot H\|_{\mathbb{X}}^2.$$

From the duality condition (2.5), we get

$$\frac{2}{\lambda} \left\langle f + \frac{1}{\lambda} \nabla \cdot \omega \middle| \nabla \cdot H \right\rangle_{\mathbb{X}} = \left\langle -\frac{2}{\lambda} \nabla \left( f + \frac{1}{\lambda} \nabla \cdot \omega \right) \middle| H \right\rangle_{\mathbb{Y}}.$$

Now, we can see that for all  $\omega \in K$ ,  $H \in \mathbb{Y}$ , and  $t \neq 0$ , we have

$$\lim_{t \rightarrow 0} \frac{F_\lambda(\omega + tH) - F_\lambda(\omega)}{t} = \left\langle -\frac{2}{\lambda} \nabla \left( f + \frac{1}{\lambda} \nabla \cdot \omega \right) \middle| H \right\rangle_{\mathbb{Y}},$$

which shows that the Gâteaux derivative of  $F_\lambda$  is given by (3.2). According to (2.6) and the fact that  $2ab \leq a^2 + b^2$ , we obtain

$$\|\nabla \cdot H\|_{\mathbb{X}}^2 \leq \sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}} \left( H_{i,j}^1 - H_{i-1,j}^1 + H_{i,j}^2 - H_{i,j-1}^1 \right)^2 \leq 8 \|H\|_{\mathbb{Y}}^2. \quad \square$$

The main idea of the conditional gradient method may be summarized as follows: Suppose that the iterate matrix  $\omega_k \in K$  at step  $k$  is given. Then the search direction  $d_k$  is obtained by solving an appropriate linear programming problem. Let

$$F_\lambda(\omega) = F_\lambda(\omega_k) + \langle F'_\lambda(\omega_k) | \omega - \omega_k \rangle_{\mathbb{Y}} + o(\|\omega - \omega_k\|_{\mathbb{Y}})$$

be the first-order Taylor expansion of  $F_\lambda$  at  $\omega_k$ . Then the problem (3.1) is approximated at each step  $k$  by the following linear optimization problem:

$$(3.4) \quad \min_{\omega \in K} g_\lambda(\omega),$$

where  $g_\lambda(\omega) = F_\lambda(\omega_k) + \langle F'_\lambda(\omega_k) | \omega - \omega_k \rangle_{\mathbb{Y}}$ . Since  $\omega_k$  is fixed at this stage of the algorithm, the minimization problem (3.4) is equivalent to

$$(3.5) \quad \min_{\omega \in K} \langle F'_\lambda(\omega_k) | \omega \rangle_{\mathbb{Y}}.$$

Let  $\hat{\omega}_k$  be a solution of the problem (3.5). Since  $K$  is a convex set, the line segment  $[\hat{\omega}_k, \omega_k]$  connecting  $\hat{\omega}_k$  and  $\omega_k$  lies entirely inside  $K$ , and  $(\hat{\omega}_k - \omega_k)$  is a feasible direction. Thus, we can perform a line-search over this line segment. That is, we solve the one-dimensional problem

$$(3.6) \quad \min_{0 \leq \alpha \leq 1} F_\lambda(\omega_k + \alpha(\hat{\omega}_k - \omega_k)).$$

Let  $\alpha_k^*$  be a solution to the line-search problem (3.6). We update

$$\omega_{k+1} = \omega_k + \alpha_k^*(\hat{\omega}_k - \omega_k), \quad k = k + 1,$$

and repeat this process. The conditional gradient method is summarized by Algorithm 1.

---

**Algorithm 1:**


---

- 1 Choose a tolerance  $tol$ , an initial matrix  $\omega_0 \in K$ , and set  $k = 0$ ;
- 2 Solve the minimization problem of a linear function over the feasible set  $K$ :

$$\min_{\omega \in K} \langle F'_\lambda(\omega_k) | \omega \rangle_{\mathbb{Y}}, \quad (*)$$

- and let  $\hat{\omega}_k$  be a solution to problem (\*);
- 3 Compute the value:  $\eta_k = \langle F'_\lambda(\omega_k) | \hat{\omega}_k - \omega_k \rangle_{\mathbb{Y}}$ ;
- 4 **If**  $|\eta_k| < tol$  **stop** **else** **continue**;
- 5 Solve the one-dimensional minimization problem

$$\min_{0 \leq \alpha \leq 1} F_\lambda(\omega_k + \alpha(\hat{\omega}_k - \omega_k)), \quad (**)$$

- and let  $\alpha_k^*$  be a solution to problem (\*\*);
  - 6 Update  $\omega_{k+1} = \omega_k + \alpha_k^*(\hat{\omega}_k - \omega_k)$ , set  $k = k + 1$  and go to Step 2.
- 

Let  $P_K$  denote the projection on the closed convex subset  $K$  of the Hilbert space  $\mathbb{Y}$ . We recall that the projection  $P_K$  is not linear. The following lemma may be seen in a general context.

**LEMMA 3.2.** *Let  $K$  be a closed convex and bounded subset of an Hilbert space  $\mathbb{Y}$  such that  $0_{\mathbb{Y}} \in K$ . Then, for any  $y \in \mathbb{Y}$  with  $y \neq 0_{\mathbb{Y}}$  and for all  $\omega \in K$ , depending on the sign of*

$\langle y|\omega \rangle_{\mathbb{Y}}$ , we have

$$\langle y| - P_K[\hat{y}] \rangle_{\mathbb{Y}} \leq 0 \leq \langle y|\omega \rangle_{\mathbb{Y}},$$

or

$$\langle y| P_K[-\hat{y}] \rangle_{\mathbb{Y}} \leq \langle y|\omega \rangle_{\mathbb{Y}} \leq 0,$$

where  $\hat{y} = \alpha \frac{y}{\|y\|_{\mathbb{Y}}}$  and  $\alpha$  is an upper bound for  $K$ , i.e.,  $\|\omega\|_{\mathbb{Y}} \leq \alpha$  for all  $\omega \in K$ .

*Proof.* For all  $y \in \mathbb{Y}$ , the projection  $P_K(y)$  on the closed convex subset  $K$  is characterized by

$$\langle y - P_K(y)|\omega - P_K(y) \rangle_{\mathbb{Y}} \leq 0, \quad \forall \omega \in K.$$

As  $0_{\mathbb{Y}} \in K$ , it follows that  $\langle y - P_K(y)| - P_K(y) \rangle_{\mathbb{Y}} \leq 0$ . Then  $\langle y| P_K(y) \rangle_{\mathbb{Y}} \geq \|P_K(y)\|_{\mathbb{Y}}^2$ . In the last inequality, replacing  $y$  by  $\hat{y}$  implies that for all  $y \in \mathbb{Y}$  we have

$$\langle y| P_K(\hat{y}) \rangle_{\mathbb{Y}} \geq \alpha \|y\|_{\mathbb{Y}} \left( \frac{\|P_K(\hat{y})\|_{\mathbb{Y}}}{\alpha} \right)^2.$$

Multiplying the last inequality by  $-1$ , we obtain that for all  $\omega \in K$ ,

$$\langle y - P_K(\hat{y}) \rangle_{\mathbb{Y}} \leq -\alpha \|y\|_{\mathbb{Y}} \left( \frac{\|P_K(\hat{y})\|_{\mathbb{Y}}}{\alpha} \right)^2 \leq -\|\omega\|_{\mathbb{Y}} \|y\|_{\mathbb{Y}} \left( \frac{\|P_K(\hat{y})\|_{\mathbb{Y}}}{\alpha} \right)^2.$$

Now, if  $\langle y|\omega \rangle_{\mathbb{Y}} \geq 0$ , then using the Schwarz inequality, we obtain

$$\langle y - P_K(\hat{y}) \rangle_{\mathbb{Y}} \leq -\|\omega\|_{\mathbb{Y}} \|y\|_{\mathbb{Y}} \left( \frac{\|P_K(\hat{y})\|_{\mathbb{Y}}}{\alpha} \right)^2 \leq \langle y|\omega \rangle_{\mathbb{Y}} \left( \frac{\|P_K(\hat{y})\|_{\mathbb{Y}}}{\alpha} \right)^2 \leq \langle y|\omega \rangle_{\mathbb{Y}}.$$

Contrary, if  $\langle y|\omega \rangle_{\mathbb{Y}} \leq 0$ , then

$$\langle y - P_K(\hat{y}) \rangle_{\mathbb{Y}} \leq -\|\omega\|_{\mathbb{Y}} \|y\|_{\mathbb{Y}} \left( \frac{\|P_K(\hat{y})\|_{\mathbb{Y}}}{\alpha} \right)^2 \leq (-\langle y|\omega \rangle_{\mathbb{Y}}) \left( \frac{\|P_K(\hat{y})\|_{\mathbb{Y}}}{\alpha} \right)^2 \leq \langle -y|\omega \rangle_{\mathbb{Y}}.$$

Replacing  $y$  by  $-y$ , we obtain

$$\langle y| P_K(-\hat{y}) \rangle_{\mathbb{Y}} \leq \langle y|\omega \rangle_{\mathbb{Y}}. \quad \square$$

Let us notice that in our context, the projection  $P_K$  on the convex set  $K$  in (2.8) is given for all  $y = (y^1, y^2) \in \mathbb{Y}$  by

$$P_K(y) = \omega = (\omega^1, \omega^2), \quad \text{with}$$

$$\omega_{i,j}^k = \frac{y_{i,j}^k}{\max(1, |y_{i,j}^k|)}, \quad k = 1, 2, \quad i = 1, \dots, n, \quad j = 1, \dots, m,$$

where  $|y_{i,j}| = \sqrt{|y_{i,j}^1|^2 + |y_{i,j}^2|^2}$ . Then, in this case  $P_K(-y) = -P_K(y)$ . The following proposition provides solutions of the problems (3.5) and (3.6).

**PROPOSITION 3.3.**

1. At iteration  $k$ , a solution of the problem (3.5) is given by

$$\hat{\omega}_k = P_K \left[ -\frac{\sqrt{2nm} F'_\lambda(\omega_k)}{\|F'_\lambda(\omega_k)\|_{\mathbb{Y}}} \right].$$

2. At iteration  $k$ , a solution of the problem (3.6) is given by

$$(3.7) \quad \alpha_k^* = \begin{cases} \alpha_k & \text{if } 0 \leq \alpha_k \leq 1, \\ 1 & \text{if } \alpha_k > 1, \end{cases}$$

where

$$\alpha_k = -\frac{\lambda^2 \langle F'_\lambda(\omega_k) | H_k \rangle_{\mathbb{Y}}}{2 \|\nabla \cdot H_k\|_{\mathbb{X}}^2}, \quad \text{and} \quad H_k = \widehat{\omega}_k - \omega_k.$$

*Proof.* 1. The proof of item 1 is an immediate consequence of Lemma 3.2. Indeed, the closed convex set  $K$  in (2.8) is bounded, and we have  $\|\omega\|_{\mathbb{Y}} \leq \sqrt{2nm}$  for all  $\omega \in K$ . The convex set  $K$  contains  $0_{\mathbb{Y}}$ . Let  $y_k = F'_\lambda(\omega_k)$  and  $\widehat{y}_k = \sqrt{2nm} \frac{y_k}{\|y_k\|_{\mathbb{Y}}}$ . For all  $\omega \in K$ , by Lemma 3.2, we have

$$\langle y_k | \omega \rangle_{\mathbb{Y}} \geq \langle y_k | P_K[-\widehat{y}_k] \rangle_{\mathbb{Y}}.$$

2. According to (3.3) we have

$$\begin{aligned} F_\lambda(\omega_k + \alpha H_k) &= F_\lambda(\omega_k) + \langle F'_\lambda(\omega_k) | \alpha H_k \rangle_{\mathbb{Y}} + o(\|\alpha H_k\|_{\mathbb{Y}}) \\ &= \alpha^2 \frac{1}{\lambda^2} \|\nabla \cdot H_k\|_{\mathbb{X}}^2 + \alpha \langle F'_\lambda(\omega_k) | H_k \rangle_{\mathbb{Y}} + \left\| f + \frac{1}{\lambda} \nabla \cdot \omega_k \right\|_{\mathbb{X}}^2 \\ &= a_k \alpha^2 + b_k \alpha + c_k, \end{aligned}$$

where  $H_k = \widehat{\omega}_k - \omega_k$ ,  $a_k = \frac{1}{\lambda^2} \|\nabla \cdot H_k\|_{\mathbb{X}}^2$ ,  $b_k = \langle F'_\lambda(\omega_k) | H_k \rangle_{\mathbb{Y}}$ , and  $c_k = \left\| f + \frac{1}{\lambda} \nabla \cdot \omega_k \right\|_{\mathbb{X}}^2$ . Then, it follows that the minimum of the quadratic one-dimensional problem,  $\min_{\alpha} F_\lambda(\omega_k + \alpha H_k)$ , is given by  $\alpha_k = -\frac{b_k}{2a_k} = -\frac{\lambda^2 \langle F'_\lambda(\omega_k) | H_k \rangle_{\mathbb{Y}}}{2 \|\nabla \cdot H_k\|_{\mathbb{X}}^2}$ . Since  $\widehat{\omega}_k$  is a solution at step  $k$  of the problem (3.5),  $\eta_k = \langle F'_\lambda(\omega_k) | H_k \rangle_{\mathbb{Y}} \leq 0$  and  $\alpha_k \geq 0$ . It follows that the solution of the problem (3.6) is given by (3.7).  $\square$

We describe the Conditional Gradient Total Variation (CGTV) method in Algorithm 2.

---

**Algorithm 2:** The Conditional Gradient Total Variation (CGTV) Algorithm.

---

- 1 **Input:** a tolerance  $tol$ , an integer  $k_{\max}$ , a value  $\lambda > 0$  and  $e_k > tol$ ;
- 2 **initialization:** choose an initial  $\omega_0 \in K$ , and set  $u_0 = f$ ,  $k = 0$ ;
- 3 **while**  $k < k_{\max}$  **and**  $e_k > tol$  **do**
- 4     Compute  $g_k = -\frac{2}{\lambda} \nabla u_k$  and  $\widehat{g}_k = \sqrt{2nm} \frac{g_k}{\|g_k\|_{\mathbb{Y}}}$ ;
- 5     Compute  $\widehat{\omega}_k = P_K[-\widehat{g}_k]$  and set  $H_k = \widehat{\omega}_k - \omega_k$ ;
- 6     Compute  $h_k = \nabla \cdot H_k$  and  $\alpha_k = -\frac{\lambda^2 \langle g_k | H_k \rangle_{\mathbb{Y}}}{2 \|h_k\|_{\mathbb{X}}^2}$ ;
- 7     **if**  $\alpha_k > 1$  **then**
- 8         |  $\alpha_k^* = 1$ ;
- 9     **else**
- 10        |  $\alpha_k^* = \alpha_k$ ;
- 11     **end**
- 12     Update  $\omega_{k+1} = \omega_k + \alpha_k^* H_k$      $d_{k+1} = \nabla \cdot \omega_{k+1}$ ,     $u_{k+1} = f + \frac{1}{\lambda} d_{k+1}$ ,    **and**  
       set  $k = k + 1$ ;
- 13     Compute  $e_k = \frac{\|u_{k+1} - u_k\|_{\mathbb{Y}}}{\|u_k\|_{\mathbb{Y}}}$ ;
- 14 **end**

**Result:** Return  $u_\lambda = u_k$ .

---



**4. Convergence.** We state some results on the convergence of the sequence generated by Algorithm 2.

**THEOREM 4.1.** *Let  $\{\eta_k\}$  be the sequence given by  $\eta_k = \langle F'_\lambda(\omega_k)|\hat{\omega}_k - \omega_k \rangle_{\mathbb{Y}}$ , where  $\hat{\omega}_k$  is a solution at step  $k$  of the problem (3.5). Then*

$$\lim_{k \rightarrow \infty} \eta_k = 0.$$

*Proof.* For any  $\alpha \in [0, 1]$ , we set  $\gamma_k(\alpha) = \omega_k + \alpha(\hat{\omega}_k - \omega_k)$ . According to (3.3), we get

$$F_\lambda(\gamma_k(\alpha)) - F_\lambda(\omega_k) = \alpha \langle F'_\lambda(\omega_k)|\hat{\omega}_k - \omega_k \rangle_{\mathbb{Y}} + o(\alpha \|\hat{\omega}_k - \omega_k\|_{\mathbb{Y}}),$$

where

$$|o(\alpha \|\hat{\omega}_k - \omega_k\|_{\mathbb{Y}})| \leq \frac{8\alpha^2}{\lambda^2} \|\hat{\omega}_k - \omega_k\|_{\mathbb{Y}}^2 \leq \frac{8\alpha^2 C^2}{\lambda^2},$$

and  $C$  is the diameter of the compact set  $K$ , i.e.,  $C = \max_{z, w \in K} \|z - w\|_{\mathbb{Y}} \leq 2mn$ . Then we get, for all  $\alpha \in [0, 1]$  and for all positive integers  $k$ , that

$$(4.1) \quad F_\lambda(\gamma_k(\alpha)) - F_\lambda(\omega_k) \leq \alpha\eta_k + \frac{8\alpha^2 C^2}{\lambda^2}.$$

As  $\alpha^*$  is a solution of the problem (3.6) and  $\omega_{k+1} = \gamma_k(\alpha^*)$ , it is clear that, for all  $\alpha \in [0, 1]$  and for all positive integers  $k$ ,

$$(4.2) \quad F_\lambda(\omega_{k+1}) \leq F_\lambda(\gamma_k(\alpha)).$$

In particular, for  $\alpha = 0$ , we obtain for all positive integers  $k$  that

$$F_\lambda(\omega_{k+1}) \leq F_\lambda(\gamma_k(0)) = F_\lambda(\omega_k).$$

We also have for all positive integers  $k$  that

$$F_\lambda(\omega_k) \geq \inf_{\omega \in K} F_\lambda(\omega).$$

It follows that the sequence  $\{F_\lambda(\omega_k)\}$  is monotonically decreasing and bounded from below, hence the sequence  $\{F_\lambda(\omega_k)\}$  is convergent. Consequently,

$$(4.3) \quad \lim_{k \rightarrow \infty} (F_\lambda(\omega_k) - F_\lambda(\omega_{k+1})) = 0.$$

From (4.1) and (4.2), we have, for all  $\alpha \in [0, 1]$  and for all positive integers  $k$ ,

$$F_\lambda(\omega_{k+1}) - F_\lambda(\omega_k) \leq \alpha\eta_k + \frac{8\alpha^2 C^2}{\lambda^2}.$$

Since  $\eta_k = \langle F'_\lambda(\omega_k)|\hat{\omega}_k - \omega_k \rangle_{\mathbb{Y}} \leq 0$ , it holds that

$$F_\lambda(\omega_k) - F_\lambda(\omega_{k+1}) \geq -\alpha\eta_k - \frac{8\alpha^2 C^2}{\lambda^2} = \alpha|\eta_k| - \frac{8\alpha^2 C^2}{\lambda^2}.$$

For all  $\alpha \in (0, 1]$  and for all positive integers  $k$  this gives

$$(4.4) \quad 0 < |\eta_k| \leq \frac{8\alpha C^2}{\lambda^2} + \frac{F_\lambda(\omega_k) - F_\lambda(\omega_{k+1})}{\alpha}.$$

According to (4.3), if we take the limit in (4.4) for  $k \rightarrow \infty$ , we obtain

$$0 \leq \liminf_{k \rightarrow \infty} |\eta_k| \leq \limsup_{k \rightarrow \infty} |\eta_k| \leq \frac{8\alpha C^2}{\lambda^2}, \quad \forall \alpha \in (0, 1].$$

Now, we take the limit in the last inequality for  $\alpha \rightarrow 0$  obtaining the desired result.  $\square$

**THEOREM 4.2.** *The sequence  $\{\omega_k\}$  generated by Algorithm 2 is a minimizing sequence, i.e.,*

$$\lim_{k \rightarrow \infty} F_\lambda(\omega_k) = \min_{\omega \in K} F_\lambda(\omega).$$

*Proof.* For any  $\alpha \in [0, 1]$ , we set  $\gamma_k(\alpha) = \omega_k + \alpha(\widehat{\omega}_k - \omega_k)$ . From Algorithm 1, we have

$$\omega_{k+1} = \omega_k + \alpha_k^*(\widehat{\omega}_k - \omega_k) = \gamma_k(\alpha_k^*),$$

where  $\alpha_k^*$  is a solution of the problem (3.6). From the convexity of the function  $F_\lambda$ , we have

$$F_\lambda(\omega_*) - F_\lambda(\omega_k) \geq \langle F'_\lambda(\omega_k) | \omega_k - \omega_* \rangle_{\mathbb{Y}},$$

where  $\omega_* \in K$  is a solution of the problem (3.1). It follows that

$$\begin{aligned} 0 \leq F_\lambda(\omega_k) - F_\lambda(\omega_*) &\leq -\langle F'_\lambda(\omega_k) | \omega_k - \omega_* \rangle_{\mathbb{Y}} \\ &\leq \min_{\omega \in K} \langle F'_\lambda(\omega_k) | \omega - \omega_k \rangle_{\mathbb{Y}} \leq -\eta_k = |\eta_k|. \end{aligned}$$

Then, we have

$$0 \leq F_\lambda(\omega_k) - F_\lambda(\omega_*) \leq |\eta_k|.$$

By Theorem 4.1, we deduce that  $\lim_{k \rightarrow \infty} F_\lambda(\omega_k) = F_\lambda(\omega_*)$ . This proves the theorem.  $\square$

**5. Selection of the parameter  $\lambda$ .** In the case of Tikhonov regularization, there are two well-known techniques for an optimal selection of the parameter  $\lambda$ , generalized cross-validation [16] and the L-curve method [17]. Unfortunately, adopting these techniques to total variation seems to be a difficult task. Nevertheless, we can adopt a method proposed in [8] by constructing a sequence  $\{\lambda_k\}$  that in the limit satisfies the standard deviation noise constraints (5.2). We assume that some statistical parameters of the additive Gaussian noise (mean, variance, ...) are known or at least are estimated. As usual, we assume here that the variance  $\sigma_\gamma^2$  is estimated by the value  $\sigma^2$  given by

$$\sigma^2 = \frac{1}{mn} \|f - \bar{f}\|_{\mathbb{X}}^2 = \frac{1}{nm} \sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}} (f_{i,j} - \bar{f})^2,$$

where  $\bar{f} = \frac{1}{nm} \sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}} f_{i,j}$  is the mean value of the observed noisy image  $f$ . The discrete problem corresponding to (1.1)–(1.2) may be written as

$$(5.1) \quad \min_{u \in \mathbb{X}} TV(u),$$

subject to

$$(5.2) \quad \|f - u\|_{\mathbb{X}}^2 = mn\sigma^2,$$

where  $TV(u)$  is now given in the discrete form (2.3). As in the continuous case, by introducing the Lagrange multiplier, the problem (5.1)–(5.2) is equivalent to the problem (2.4). The task is to find  $\lambda > 0$  such that  $\|f - u_\lambda\|_{\mathbb{X}}^2 = mn\sigma^2$ , where  $u_\lambda$  is the solution of (2.4) corresponding to  $\lambda$ . According to [8], for  $\lambda > 0$ , we consider the function  $\varphi(\lambda) = \|P_{\lambda K}f\|_{\mathbb{X}}$ . The problem is to find  $\lambda^*$  such that  $\varphi(\lambda^*) = \sigma\sqrt{mn}$ . We recall that  $P_{\lambda K}f$  is the projection of  $f$  to the convex set  $\lambda K$ . We have  $u_{\lambda^*} = f + \frac{1}{\lambda^*} \nabla \cdot \omega_\lambda = f - P_{\lambda K}f$ . It follows that

$$\varphi(\lambda^*) = \|P_{\lambda^* K}f\|_{\mathbb{X}} = \|f - u_{\lambda^*}\|_{\mathbb{X}} = \sigma\sqrt{mn} = \|f - \bar{f}\|_{\mathbb{X}}.$$

Let us now generate a sequence  $\{\lambda_\ell\}$  such that

$$(5.3) \quad \lambda_{\ell+1} = \frac{\|f - u_{\lambda_\ell}\|_{\mathbb{X}}}{\|f - \bar{f}\|_{\mathbb{X}}} \lambda_\ell, \quad \ell \geq 0,$$

where the initial guess is chosen such that  $\lambda_0 > 0$ . The property that  $\lim_{\ell \rightarrow +\infty} \lambda_\ell = \lambda^* > 0$  was proved in [8], and consequently

$$\lim_{\ell \rightarrow +\infty} \|f - u_{\lambda_\ell}\|_{\mathbb{X}} = \|f - \bar{f}\|_{\mathbb{X}} = \sigma\sqrt{mn}.$$

Based on the sequence (5.3), we may derive the following version of our proposed algorithm.

---

**Algorithm 3:** The Conditional Gradient Total Variation (CGTV) Algorithm.

---

- 1 **Input:** a tolerance  $tol_1$ , an integer  $\ell_{\max}$ ;
- 2 **Initialization:** choose an initial value  $\lambda_0 > 0$  and  $r_\ell > tol_1$  and set  $\lambda_\ell = \lambda_0$ , and  $\ell = 0$ ;
- 3 **while**  $\ell < \ell_{\max}$  **and**  $r_\ell > tol_1$  **do**
- 4     Compute:  $u_\lambda$  by using Algorithm 2;
- 5     Update  $\lambda_{\ell+1} = \frac{\|f - u_{\lambda_\ell}\|_{\mathbb{X}}}{\|f - \bar{f}\|_{\mathbb{X}}} \lambda_\ell$ ;
- 6     Compute  $r_\ell = |\lambda_{\ell+1} - \lambda_\ell|$  and set  $\ell = \ell + 1$ ;
- 7 **end**

**Result:** Return  $u_\lambda$ .

---

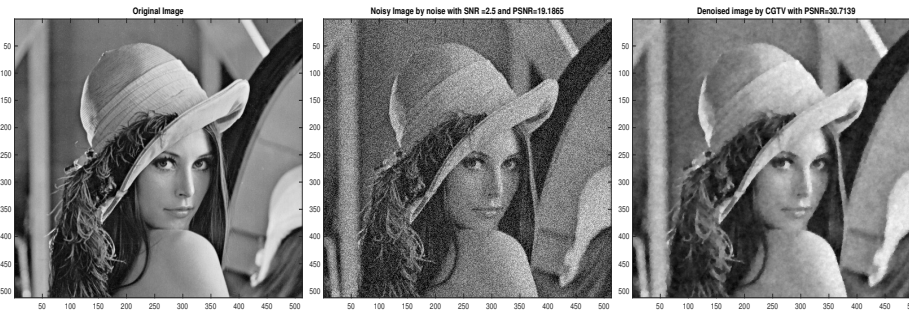


FIG. 5.1. True image (left), the noisy image with  $SNR = 2.5$  and  $PSNR = 19.186$  (center) and the denoised image (right) with  $PSNR = 30.714$  with  $\lambda = 0.034$ ,  $R_e(u) = 7.158 \times 10^{-2}$ .

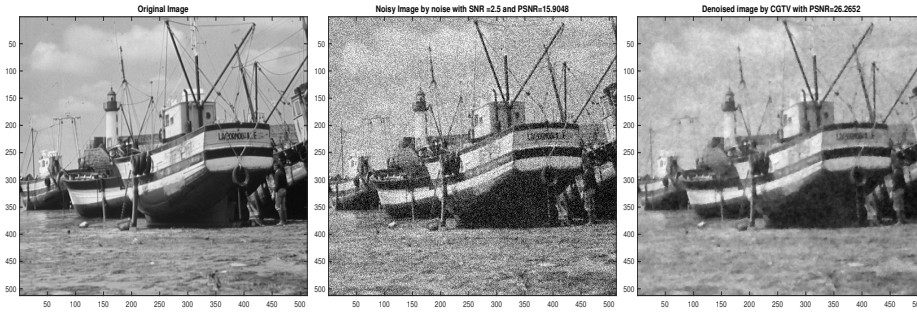


FIG. 5.2. True image (left), the noisy image with  $SNR = 2.5$  and  $PSNR = 15.905$  (center) and the denoised image (right) with  $PSNR = 26.265$  with  $\lambda = 0.026$ ,  $R_e(u) = 8.193 \times 10^{-2}$ .

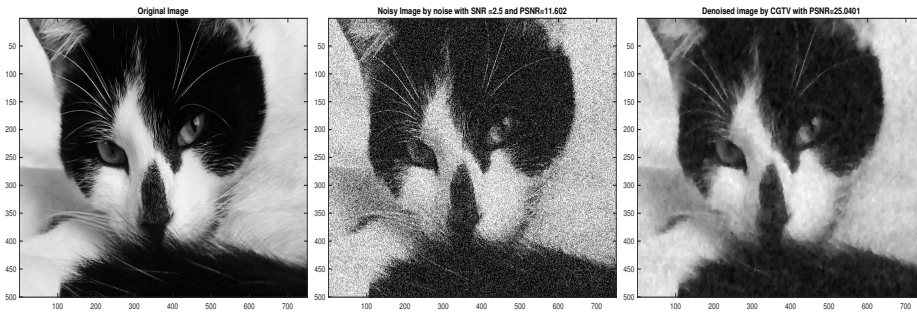


FIG. 5.3. True image (left), the noisy image with  $SNR = 2.5$  and  $PSNR = 11.602$  (center) and the denoised image (right) with  $PSNR = 25.040$  with  $\lambda = 0.014$ ,  $R_e(u) = 1.013 \times 10^{-1}$ .

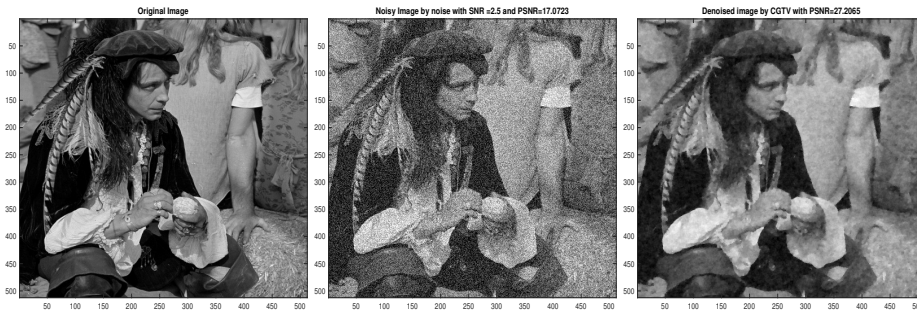


FIG. 5.4. True image (left), the noisy image with  $SNR = 2.5$  and  $PSNR = 17.072$  (center) and the denoised image (right) with  $PSNR = 27.206$  with  $\lambda = 0.031$ ,  $R_e(u) = 9.164 \times 10^{-2}$ .

**6. Numerical results.** In this section, we present three numerical tests that illustrate the effectiveness of our proposed method. We test our algorithm by taking known images as matrices of size  $n \times m$  with grayscale pixel values in the range  $[0, d]$ , where  $d = 255$  is the maximum possible pixel value of the image. The true image is denoted by  $\hat{u}$ . We generate a noisy image  $f = \hat{u} + \eta$ , where  $\eta$  is a noise matrix whose entries are normally distributed and Gaussian random with zero mean and a value of its variance such that the signal to noise ratio

( $SNR$ ) has an appropriate (dB) value. We recall that the  $SNR$  is defined by

$$SNR = 10 \log_{10} \left( \frac{\sigma_{\hat{u}}^2}{\sigma_{\eta}^2} \right),$$

where  $\sigma_{\eta}^2$  and  $\sigma_{\hat{u}}^2$  are the variances of the noise and the original image, respectively. Therefore in our tests, we generate an observed noisy image  $f$  by

$$(6.1) \quad f = \hat{u} + \eta_0 \times \sigma_{\hat{u}} \times 10^{-\frac{SNR}{20}},$$

where  $\eta_0$  is a random noise matrix with zero mean and variance equal to one. The performance of the proposed method is evaluated by computing the  $PSNR$  (peak signal to noise ratio) of the denoised (restored) image  $u = u_{\lambda}$  (for an appropriate  $\lambda$ ), which is defined by

$$PSNR(u) = 10 \log_{10} \left( \frac{d^2 n \times m}{\|\hat{u} - u\|_F^2} \right).$$

To evaluate the precision of the estimates, the following relative error is also computed

$$R_e(u) = \frac{\|\hat{u} - u\|_F}{\|\hat{u}\|_F},$$

where  $\|\cdot\|_F$  is the Frobenius norm. All our computations were carried out using MATLAB 12 on an 1.7 GHZ Intel(R) core i7 with 8 GB of RAM.

**6.1. Examples of image denoising.** In this section, we illustrate the effectiveness of our proposed algorithm by choosing four benchmark images: the ‘‘Lena’’, ‘‘Boat’’, ‘‘Man’’ images of size  $512 \times 512$  and the ‘‘Cat’’ image of size  $500 \times 750$ . Random Gaussian noise has been added to each image to obtain a noisy image with a specific  $SNR$ . The additive noise was constructed as in (6.1). The values of the SNR were  $SNR = 2.5$ ,  $SNR = 5$ , and  $SNR = 7.5$ . Table 6.1 gives the SNR of the noise for each image as well as the PSNR of the corresponding noisy image. We have applied our algorithm to each image to obtain a denoised image  $u_{\lambda}$  with an appropriate value of  $\lambda$  for each image. The criterion for stopping Algorithm 2 was that the relative error between two successive iterates of approximated primal variable is less than the tolerance  $tol = 10^{-5}$  with the maximum number of iterations  $k_{max} = 500$ . In Table 6.1 we have reported the  $PSNR(u_{\lambda})$  of the denoised image  $u_{\lambda}$  as well as the relative error  $R_e(u_{\lambda})$  and the CPU-time. Figures 5.1–5.4 display the true, noisy, and denoised images for each case, respectively. In Figure 6.1, we have plotted the curve  $\lambda \rightarrow PSNR(u_{\lambda})$ , where  $\lambda$  is considered as a variable in the range  $[0, 0.5]$ . We observed that the curves have a maximum  $PSNR$  value at an optimal value of  $\lambda = \lambda_{opt}$  located in the range  $[0, 0.5]$ . The optimal value  $\lambda_{opt}$  is the value of  $\lambda$  for which the best result, both in terms of the maximum  $PSNR$  as well as in terms of the relative error, was obtained. In this example, our proposed algorithm was run for each image with the optimal value  $\lambda_{opt}$ . The optimal value was obtained by hand. Indeed, the parameter  $\lambda$  needs to be tuned for each noised image to get an optimal value.

Based on the tests reported in Table 6.1 and many more unreported tests, we remark that our proposed algorithm works very effectively for image denoising problems both in terms of the PSNR as well as in terms of the relative error. It is very fast in terms of CPU-time. However, the determination of an optimal value of the parameter  $\lambda$  is still a serious problem. This is a general drawback for all total variation methods. In this example, we did not use any criterion for choosing a good parameter  $\lambda$ . We hand-tuned the optimal value of the parameter  $\lambda$  by choosing the value of  $\lambda$  corresponding to the maximum  $PSNR$ . We notice that such a technique is not exploitable in practice since the  $PSNR$  is computed from the true image, which is unknown in general. In the following section, we use Algorithm 3 as a remedy. Using the convergent sequence given by (5.3), we may obtain a good value of the parameter  $\lambda$ .

TABLE 6.1  
*The computational results for optimal parameter  $\lambda$  with  $SNR = 2.5$ ,  $SNR = 5$ , and  $SNR = 7.5$ .*

Image	$SNR$	$\lambda$	$PSNR(f)$	$PSNR(u_\lambda)$	$R_e(u_\lambda)$	CPU-time
Lena	2.5	0.034	19.1865	30.714	$7.158 \times 10^{-2}$	1.74s
	5	0.046	21.686	31.838	$6.290 \times 10^{-2}$	1.25s
	7.5	0.069	24.186	33.056	$5.467 \times 10^{-2}$	0.86s
Boat	2.5	0.026	15.905	26.265	$8.193 \times 10^{-2}$	1.47s
	5	0.039	18.405	28.290	$7.162 \times 10^{-2}$	1.04s
	7.5	0.054	20.905	28.651	$6.225 \times 10^{-2}$	0.63s
Cat	2.5	0.014	11.602	25.040	$1.013 \times 10^{-1}$	4.17s
	5	0.019	14.102	25.957	$9.115 \times 10^{-2}$	2.83s
	7.5	0.029	16.602	26.924	$8.156 \times 10^{-2}$	2.47s
Man	2.5	0.031	17.072	27.206	$9.164 \times 10^{-2}$	1.13s
	5	0.044	19.572	28.290	$8.089 \times 10^{-2}$	1.14s
	7.5	0.064	22.072	29.492	$7.044 \times 10^{-2}$	0.73s

**6.2. Image denoising with an automatic selection of the parameter  $\lambda$ .** In this section, we show the efficiency of Algorithm 3 for selecting a good value of the parameter  $\lambda$ . The criterion for stopping Algorithm 2 consists of the tolerance  $tol = 10^{-4}$  and the maximum number of iterations  $k_{\max} = 200$  while the tolerance in Algorithm 3 is set to  $tol_1 = 5 \times 10^{-2}$  with the maximum number of iterations  $\ell_{\max} = 100$  and the initial value  $\lambda_0 = 100$ . We use the same benchmark images as in the previous section. The results obtained with Algorithm 3 are summarized in Table 6.2. We observe that there are slight differences between the results of the PSNR given in Table 6.1 and Table 6.2. The value of  $\lambda$  is here obtained as a limit of the sequence given by Algorithm 3 while in the previous section the value of the parameter  $\lambda$  was hand-tuned to obtain a maximum PSNR. The determination of an algorithm for finding an optimal value of the parameter  $\lambda$  is still a serious problem. Nevertheless, this proposed selection method gives a procedure to find a parameter that provides a good PSNR value as well as a small relative error.

**6.3. Comparisons.** In this section, we compare numerically, under the same conditions, our algorithm referred to as the CGTV algorithm with three state-of-art algorithms developed for TV regularization: Chambolle’s projection algorithm, referred to as the CPA algorithm, the split Bregman algorithm, referred to as the SpB algorithm, and a primal-dual hybrid gradient algorithm, referred to as the PDHG algorithm. The CPA algorithm was introduced in [8]

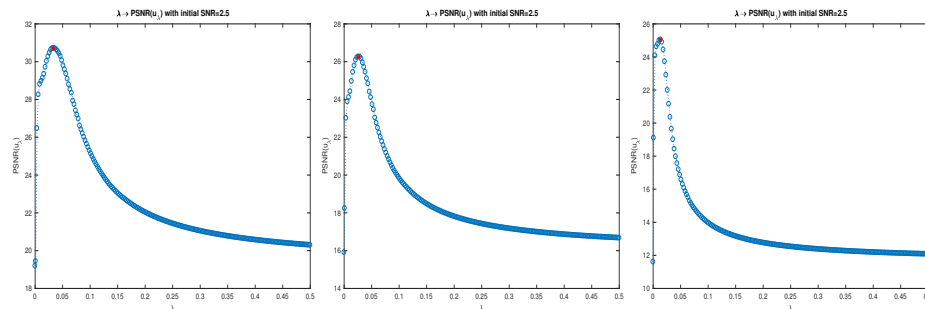


FIG. 6.1. *The PSNR curves  $\lambda \rightarrow PSNR(u_\lambda)$  for the images from the left to the right: Lena, Boat, and Cat with the  $SNR = 2.5$ .*

TABLE 6.2  
*The computational results for the automatic section of the parameter  $\lambda$  by Algorithm 3.*

Image	$SNR$	$\lambda$	$PSNR(f)$	$PSNR(u_\lambda)$	$R_e(u_\lambda)$	CPU-time
Lena	2.5	0.022	19.186	30.663	$7.200 \times 10^{-2}$	1.97s
	5	0.020	21.689	31.771	$6.338 \times 10^{-2}$	4.31s
	7.5	0.017	24.186	32.425	$5.878 \times 10^{-2}$	3.46s
Boat	2.5	0.015	15.905	26.313	$8.148 \times 10^{-2}$	3.920s
	5	0.014	18.405	27.169	$7.384 \times 10^{-2}$	2.95s
	7.5	0.012	20.904	27.620	$7.009 \times 10^{-2}$	3.77s
Cat	2.5	0.010	11.602	24.897	$1.029 \times 10^{-1}$	3.13s
	5	0.009	14.102	25.887	$9.190 \times 10^{-2}$	2.67s
	7.5	0.008	16.602	26.757	$8.786 \times 10^{-2}$	1.97s
Man	2.5	0.017	17.072	27.227	$9.142 \times 10^{-2}$	3.56s
	5	0.016	19.572	28.044	$8.321 \times 10^{-2}$	2.49s
	7	0.014	22.072	28.442	$7.949 \times 10^{-2}$	4.45s

by Chambolle as a projection algorithm for solving the problem (1.3); see also [9, 10]. The main idea behind CPA is based on a dual formulation and is related to the work in [11]. To ensure convergence of CPA, it is shown in [8] that a parameter  $\tau$  must satisfy the condition  $0 < \tau \leq \frac{1}{8}$ . In [2], Aujol revisited CPA and showed that CPA is in fact a particular case of an algorithm previously proposed by Bermúdez and Moreno in [3], which is an adaptation of Uzawa’s algorithm to the problem (1.3). So, the range of convergence may be extended to  $0 < \tau < \frac{1}{4}$ , and the best convergence is obtained for  $0.24 \leq \tau \leq 0.249$ . In our tests, the value of the parameter was fixed at  $\tau = 0.245$ . The split Bregman (SpB) algorithm was introduced by Goldstein and Osher in [15]; see also [18]. The main idea behind this method is the use of the alternating direction method of multipliers adapted to L1-problems. The split Bregman method is used to solve a variety of L1-regularized optimization problems and is particularly effective for problems involving TV regularization. The primal-dual hybrid gradient algorithm (PDHG) was proposed by Zhu and Chan in [23] for a broader variety of convex optimization problems; see also [13]. PDHG is an example of a first-order method, since it requires only functional and gradient evaluations. The PDHG method has been widely used in image processing. For comparing the accuracy obtained by our proposed algorithm with those of the three algorithms, we have used the three benchmark images “Lena”, “Boat”, and “Man” of size  $512 \times 512$  from the previous sections. We generated a noisy image from a true image  $\hat{u}$  by using an additive Gaussian noise with specific values of the SNR as in (6.1). We used three different values of  $SNR = 1$ ,  $SNR = 3$ , and  $SNR = 7$ . In order to provide a fair and unified framework for comparison, we have carefully tuned the value of the parameter  $\lambda$  for optimal performance of the PSNR of the denoised image corresponding to a maximum PSNR obtained by one of these methods.

All these algorithms are endowed with the same convergence criterion, i.e., the iterations for all algorithms were terminated when the relative error between two successive iterates of approximated primal variable is less than the tolerance  $tol = 10^{-5}$  or when a maximum of 500 iterations has been performed. In Table 6.3, and for each algorithm, we report the number of iterations (iter) reached, the relative error, the PSNR, as well as the CPU-time in seconds. In Figure 6.2, we plot the curves of the relative error for each algorithm corresponding to different images for the values of the  $SNR = 1$ ,  $SNR = 3$ , and  $SNR = 7$ . We observe, from Figure 6.2, that CGTV is better in terms of rapid convergence than the CPA, SpB, and PDHG algorithms. This may be also observed from Table 6.3 by comparing the number of

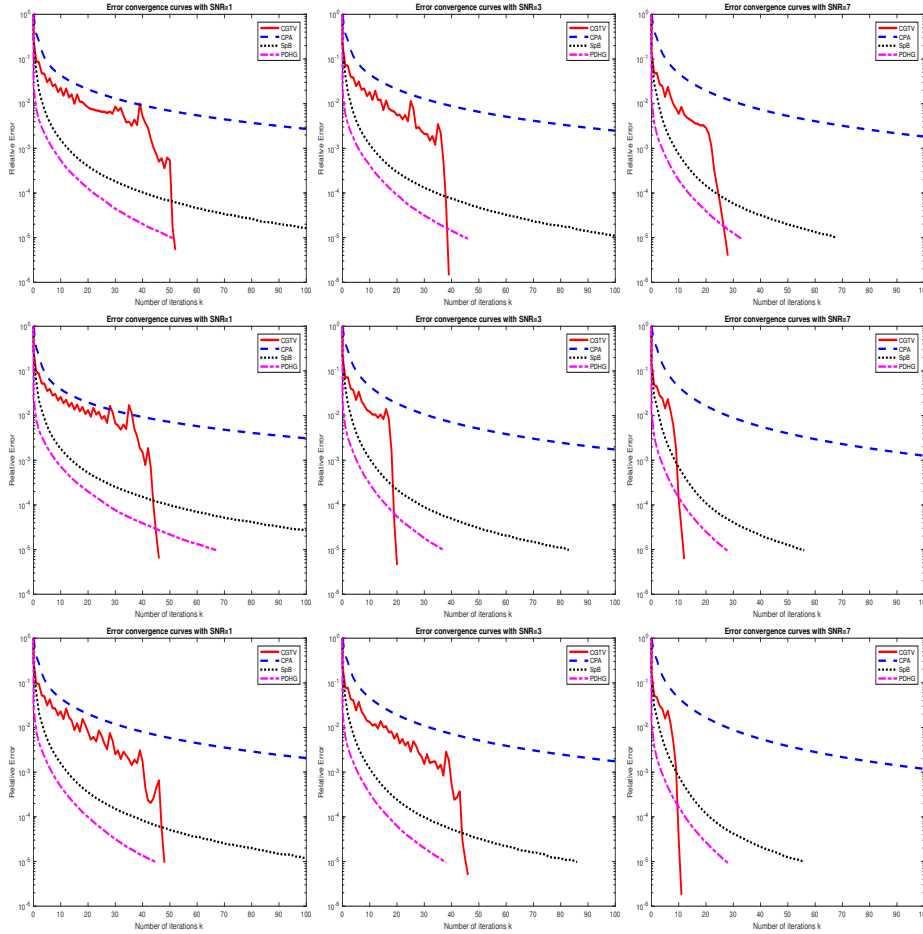


FIG. 6.2. Relative error curves of CGTV, CPA, SpB and PDHG algorithms with SNR = 1 (left), SNR = 3 (center) and SNR = 7 (right) for “Lena” image (top), “Boat” image (center) and “Man” image (bottom).

iterations reached for each algorithm and the CPU-time. From Table 6.3, it is apparent that in most cases, our proposed algorithm restored the image with fewer iterations and in less CPU-time than the other three above algorithms with a comparable visual quality confirmed by the values of the obtained PSNR which is in most cases close to the PSNR obtained by the three other algorithms. We notice that the PSNR reached by the CPA, SpB, and PDHG algorithms is still slightly better than that obtained by CGTV. The slowest algorithm in our reported tests was the CPA, which in most cases reached the maximum number of iterations. The observations were confirmed in many others tests, not reported in this table, for different values of the SNR and of the parameter  $\lambda$ . There are some cases where our algorithm obtains the best PSNR with rapid convergence. For instance, in the case of the Boat image with SNR = 1, we observe that the relative error and the PSNR obtained by the CGTV were  $R_e(u_\lambda^{CGTV}) = 8.920 \times 10^{-2}$  and  $PSNR(u_\lambda^{CGTV}) = 25.527$  with CPU-time 4.70s, while



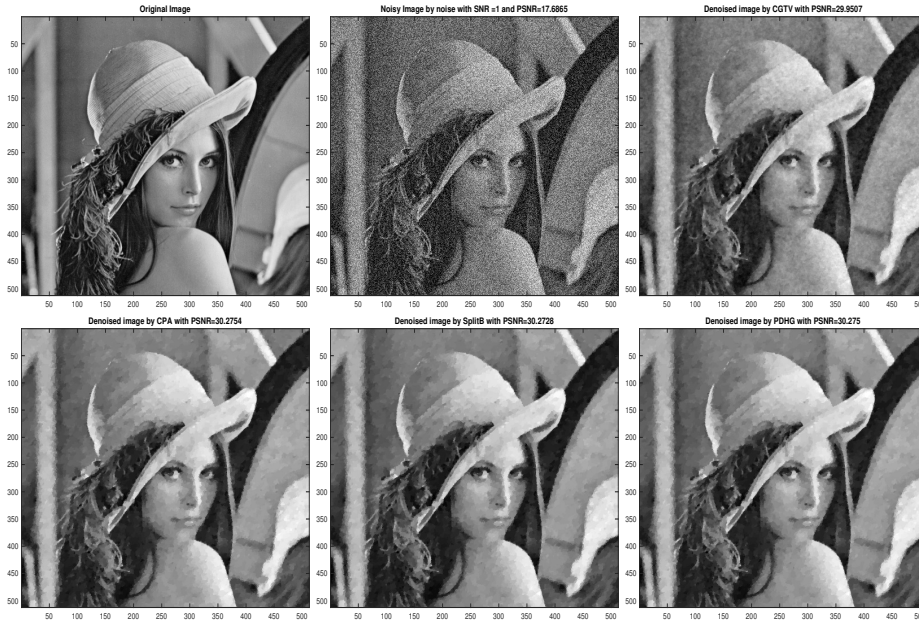


FIG. 6.3. From left to right and from top to bottom: Original image, denoised image, restored image by CGTV, CPA, SpB, and PDHG algorithms for Lena image with  $SNR = 1$ .

the results obtained by CPA, SpB, and PDHG were

$$\begin{aligned}
 R_e(u_\lambda^{CPA}) &= 9.071 \times 10^{-2}, & PSNR(u_\lambda^{CPA}) &= 25.382, \\
 R_e(u_\lambda^{SpB}) &= 9.068 \times 10^{-2}, & PSNR(u_\lambda^{SpB}) &= 25.384, \\
 R_e(u_\lambda^{PDHG}) &= 9.070 \times 10^{-2}, & PSNR(u_\lambda^{PDHG}) &= 25.381,
 \end{aligned}$$

where  $u_\lambda^{CGTV}$ ,  $u_\lambda^{CPA}$ ,  $u_\lambda^{SpB}$ , and  $u_\lambda^{PDHG}$  denote the denoised image obtained by the CGTV, CPA, SpB, and PDHG algorithms, respectively. All the four algorithms are very sensitive to the value of the parameter  $\lambda$ . In Figure 6.3, we display the corresponding noisy and denoised images by the four algorithms for the “Lena” image with  $SNR = 1$ .

**7. Conclusion.** In this paper we have introduced the conditional gradient method to the primal-dual formulation of the total variation problem to derive a new algorithm that is competitive both in precision and convergence. We have proved the convergence of our algorithm, and we have tested it numerically showing its effectiveness as a method for denoising images. The value of the Lagrange multiplier may be obtained as the limit of a convergent sequence. We have compared our proposed method to three state-of-art algorithms developed for TV regularization, namely Chambolle’s projection algorithm, the split Bregman algorithm, and a primal-dual hybrid gradient algorithm. In terms of speed and convergence, our algorithm is competitive yielding excellent quality of the restored image.

**Acknowledgements.** The authors would like to thank the anonymous referees for their valuable suggestions and comments, which improved the quality of this paper.

REFERENCES

[1] H. ATTOUCH, G. BUTTAZZO, AND G. MICHAILLE, *Variational Analysis in Sobolev and BV Spaces: Applications to PDEs and Optimization*, SIAM, Philadelphia, 2014.

TABLE 6.3  
*The computational results for the CGTV, CPA, SpB and PDHG algorithms.*

Image	SNR	$\lambda$	$PSNR(f)$	Method	iter	$R_e(u)$	$PSNR(u)$	CPU-time
Lena	1	0.035	17.686	CGTV	53	$7.816 \times 10^{-2}$	29.951	6.18s
				CPA	500	$7.529 \times 10^{-2}$	30.275	35.31s
				SpB	128	$7.531 \times 10^{-2}$	30.273	37.10s
				PDHG	52	$7.529 \times 10^{-2}$	30.275	12.85s
	3	0.045	19.686	CGTV	40	$7.001 \times 10^{-2}$	30.907	5.22s
				CPA	500	$6.766 \times 10^{-2}$	31.204	40.96s
				SpB	107	$6.766 \times 10^{-2}$	31.203	34.91s
				PDHG	47	$6.766 \times 10^{-2}$	31.204	11.45s
	7	0.079	23.686	CGTV	29	$5.728 \times 10^{-2}$	32.651	3.06s
				CPA	500	$5.498 \times 10^{-2}$	33.007	27.68s
				SpB	69	$5.499 \times 10^{-2}$	33.004	18.96s
				PDHG	34	$5.497 \times 10^{-2}$	33.007	7.33s
Boat	1	0.020	14.405	CGTV	47	$8.920 \times 10^{-2}$	25.527	4.70s
				CPA	500	$9.071 \times 10^{-2}$	25.382	26.85s
				SpB	169	$9.068 \times 10^{-2}$	25.384	48.84s
				PDHG	68	$9.070 \times 10^{-2}$	25.381	15.09s
	3	0.035	16.405	CGTV	21	$8.107 \times 10^{-2}$	26.357	2.73s
				CPA	500	$7.858 \times 10^{-2}$	26.627	35.52s
				SpB	84	$7.855 \times 10^{-2}$	26.631	24.34s
				PDHG	38	$7.9023 \times 10^{-2}$	26.627	9.17s
	7	0.060	20.405	CGTV	13	$6.496 \times 10^{-2}$	28.281	1.38s
				CPA	500	$6.268 \times 10^{-2}$	28.592	31.68s
				SpB	57	$6.272 \times 10^{-2}$	28.586	16.49s
				PDHG	29	$6.268 \times 10^{-2}$	28.591	6.59s
Man	1	0.030	15.572	CGTV	49	$1.003 \times 10^{-1}$	26.425	5.02s
				CPA	500	$9.789 \times 10^{-2}$	26.633	34.29s
				SpB	111	$9.781 \times 10^{-2}$	26.640	30.88s
				PDHG	46	$9.789 \times 10^{-2}$	26.633	10.71s
	3	0.040	17.572	CGTV	47	$9.077 \times 10^{-2}$	27.289	5.68s
				CPA	500	$8.841 \times 10^{-2}$	27.518	37.60s
				SpB	87	$8.840 \times 10^{-2}$	27.518	25.17s
				PDHG	39	$8.841 \times 10^{-2}$	27.517	9.40s
	7	0.065	21.572	CGTV	12	$7.328 \times 10^{-2}$	29.149	2.08s
				CPA	500	$7.119 \times 10^{-2}$	29.399	28.94s
				SpB	57	$7.112 \times 10^{-2}$	29.408	16.36s
				PDHG	29	$7.119 \times 10^{-2}$	29.399	6.19s

[2] J. AUJOL, *Some first-order algorithms for total variation based image restoration*, J. Math. Imaging Vision, 34 (2009), pp. 307–327.

[3] A. BERMÚDEZ AND M. MORENO, *Duality methods for solving variational inequalities*, Comput. Math. Appl., 7 (1981), pp. 43–58.

[4] A. BOUHAMIDI, R. ENKHBAT, AND K. JBILOU, *Conditional gradient Tikhonov method for a convex optimization problem in image restoration*. J. Comput. Appl. Math., 255 (2014), pp. 580–592.

[5] A. BOUHAMIDI AND K. JBILOU, *Sylvester Tikhonov-regularization methods in image restoration*, J. Comput. Appl. Math., 206 (2007), pp. 86–98.

[6] D. CALVETTI, B. LEWIS, L. REICHEL, AND F. SGALLARI, *Tikhonov regularization with nonnegative con-*

- straints*, Electron. Trans. Numer. Anal., 18 (2004), pp. 153–173.  
<http://etna.ricam.oeaw.ac.at/vol.18.2004/pp153-173.dir/pp153-173.pdf>
- [7] D. CALVETTI, S. MORIGI, L. REICHEL, AND F. SGALLARI, *Tikhonov regularization and the L-curve for large discrete ill-posed problems*, J. Comput. Appl. Math., 123 (2000), pp. 423–446.
  - [8] A. CHAMBOLLE, *An algorithm for total variation minimization and applications*, J. Math. Imaging Vision, 20 (2004), pp. 89–97.
  - [9] ———, *Total variation minimization and a class of binary MRF models*, in Energy Minimization Methods in Computer Vision and Pattern Recognition EMMCVPR 05, A. Rangarajan, B. Vemuri, and A. L. Yuille, eds., Lecture Notes in Computer Sciences vol. 3757, Springer, Berlin, 2005, pp. 136–152.
  - [10] A. CHAMBOLLE AND T. POCK, *A first-order primal-dual algorithm for convex problems with applications to imaging*, J. Math. Imaging Vision, 40 (2011), pp. 120–145.
  - [11] T. F. CHAN, G. H. GOLUB, AND P. MULET, *A nonlinear primal-dual method for total variation-based image restoration*, SIAM J. Sci. Comput., 20 (1999), pp. 1964–1977.
  - [12] I. EKELAND AND R. TÉMAM, *Convex Analysis and Variational Problems*, SIAM, Philadelphia, 1999.
  - [13] E. ESSER, X. ZHANG, AND T. F. CHAN, *A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science*, SIAM J. Imaging Sci., 3 (2010), pp. 1015–1046.
  - [14] M. FRANK AND P. WOLFE, *An algorithm for quadratic programming*, Naval Res. Logist. Quart., 3 (1956), pp. 95–110.
  - [15] T. GOLDSTEIN AND S. OSHER, *The split Bregman algorithm for L1 regularized problems*, SIAM J. Imaging Sci., 2 (2009), pp. 323–343.
  - [16] G. H. GOLUB, M. HEATH, AND G. WAHBA, *Generalized cross-validation as a method for choosing a good ridge parameter*, Technometrics, 21 (1979), pp. 215–223.
  - [17] P. C. HANSEN, *Analysis of discrete ill-posed problems by means of the L-curve*, SIAM Rev., 34 (1992), pp. 561–580.
  - [18] S. OSHER, M. BURGER, D. GOLDFARB, J. XU, AND W. YIN, *An iterative regularization method for total variation-based image restoration*, Multiscale Model. Simul., 4 (2005), pp. 460–489.
  - [19] L. I. RUDIN, S. OSHER, AND E. FATEMI, *Nonlinear total variation based noise removal algorithms*, Physica D, 60 (1992), pp. 259–268.
  - [20] O. V. VASILIEV, *Optimization Methods*, World Federation Publishers, Atlanta, 1996.
  - [21] B. ZHANG, Z. ZHU, AND S. LI, *A modified spectral conjugate gradient projection algorithm for total variation image restoration*, Appl. Math. Lett., 27 (2014), pp. 26–35.
  - [22] M. ZHU, S. J. WRIGHT, AND T. F. CHAN, *Duality-based algorithms for total-variation-regularized image restoration*, Comput. Optim. Appl., 47 (2010), pp. 377–400.
  - [23] M. ZHU AND T. F. CHAN, *An efficient primal-dual hybrid gradient algorithm for total variation image restoration* Tech. Report, CAM Report 08-34, UCLA Center for Applied Math., UCLA, Los Angeles, 2008.