electronics

MDPI

Article

Image Restoration Using Fixed-Point-Like Methods for New TVL1 Variational Problems

Jae Heon Yun * and Hyo Jin Lim

Department of Mathematics, College of Natural Sciences, Chungbuk National University, Cheongju 28644, Korea; jellya@naver.com

* Correspondence: gmjae@chungbuk.ac.kr; Tel.: +82-10-8843-2250

Received: 29 February 2020; Accepted: 27 April 2020; Published: 29 April 2020

Abstract: In this paper, we first propose two TVL1 variational problems for restoring images degraded by blurring and impulse noise, and then we propose two fixed-point-like methods, using proximal operators, for solving the new proposed TVL1 problems. Numerical experiments for several test images blurred by Gaussian kernel and corrupted by salt-and-pepper impulse noise are provided to demonstrate the efficiency and reliability of the proposed fixed-point-like methods. Numerical results show that two fixed-point-like methods for solving the new TVL1 variational problems perform very well in both PSNR (Peak signal-to-noise ratio) values and CPU time as compared with the fixed-point-like methods for solving two existing TVL1 variational problems.

Keywords: TVL1 problems; fixed-point-like method; image restoration; impulse noise; proximal operator

MSC: 94A08; 54E05; 49Q20; 35M85

1. Introduction

In this paper, we consider the problem of restoring images degraded by blurring and impulse noise. Impulse noise is often generated by malfunctioning pixels in camera senses, faulty memory locations in hardware, or erroneous transmission. Two common types of impulse noise are salt-and-pepper noise and random-valued noise. Assume that an intensity range of an image is $[d_{min}, d_{max}]$. Salt-and-pepper noise corrupts a portion of pixels with only two values of d_{min} or d_{max} while keeping other pixels unaffected. For random-valued noise, a portion of pixels is corrupted in the same manner as salt-and-pepper noise except that the corrupted pixels can take any random value between d_{min} and d_{max}.

Let us assume that the true image U has an $N \times N$ square array. For convenience of exposition, the image U is represented by a long vector u of size $m = N^2$ which is defined by stacking the columns of U, i.e.,

$$u = (u_{*1}^T, u_{*2}^T, \ldots, u_{*N}^T)^T$$

where $u_{*\ell} \in \mathbb{R}^N$ denotes the ℓth column of U. In this paper, we only consider the reflexive boundary condition, which means that the scene outside the image boundaries is a mirror image of the scene inside the image boundaries, and we assume that an observed (or degraded) image $f \in \mathbb{R}^m$ is represented by

$$f = Au + \eta \tag{1}$$

where $A \in \mathbb{R}^{m\times m}$ is a blurring operator, $u \in \mathbb{R}^m$ is the original image, and $\eta \in \mathbb{R}^m$ denotes the impulse noise. Then, for salt-and-pepper noise, the noisy image $f = (f_i) \in \mathbb{R}^m$ is defined as

$$f_i = \begin{cases} d_{min} & \text{with probability } \frac{s}{2} \\ d_{max} & \text{with probability } \frac{s}{2} \\ \tilde{u}_i & \text{with probability } 1-s \end{cases},$$

where $\tilde{u} = Au$ and s is the noise level of the salt-and-pepper noise. For random-valued noise, f is defined as

$$f_i = \begin{cases} d_i & \text{with probability } r \\ \tilde{u}_i & \text{with probability } 1-r \end{cases},$$

where d_i is the uniformly distributed random variable in $[d_{min}, d_{max}]$ and r is the noise level of the random-valued noise. Our objective of this paper is to restore u from the blurred and noisy image f as well as possible.

The classic TVL1 model for recovering a true image u from an observed image f with impulse noise is given by the following variational problem with the l_1-norm data fidelity term and total variational regularization term

$$\min_u \{\|Au - f\|_1 + \rho\|u\|_{\mathrm{TV}} : u \in \mathbb{R}^m\}, \tag{2}$$

where $\rho > 0$ is a regularization parameter and $\|u\|_{\mathrm{TV}}$ denotes the total variation (TV) of u. There are two possible definitions for $\|u\|_{\mathrm{TV}}$; one is the anisotropic TV, and the other is the isotropic TV. In this paper, we only consider the isotropic TV of $u \in \mathbb{R}^m$, which is defined by

$$\|u\|_{\mathrm{TV}} = \sum_{i=1}^{m} \|(\nabla u)_i\|_2 = \sum_{i=1}^{m} \left\| \begin{pmatrix} (\nabla_x u)_i \\ (\nabla_y u)_{m+i} \end{pmatrix} \right\|_2, \tag{3}$$

where the discrete gradient operator $\nabla : \mathbb{R}^m \to \mathbb{R}^{2m}$ is defined as follows:

$$(\nabla u)_i = \left((\nabla_x u)_i, (\nabla_y u)_{m+i}\right)^T \text{ for each } i = 1, 2, \ldots, m$$

with

$$(\nabla_x u)_i = \begin{cases} 0 & , \text{ if } i \bmod N = 1, \\ u_i - u_{i-1}, & \text{if } i \bmod N \neq 1, \end{cases} \text{ and } (\nabla_y u)_{m+i} = \begin{cases} 0 & , \text{ if } i \leq N, \\ u_i - u_{i-N}, & \text{if } i > N. \end{cases}$$

Note that, if ℓ_1-norm is used instead of ℓ_2-norm in Equation (3), then the resulting TV norm is anisotropic.

Notice that there have been many existing mathematical models other than the TVL1 model (2) for recovering a true image from an observed image corrupted by blur and impulse noise (e.g., see [1,2] and the references therein). In this paper, we only consider fixed-point-like methods for solving several types of TVL1 variational models. In the last few decades, the problem of solving the classical TVL1 model Equation (2) has been studied by many researchers (see [3–12] and the references therein). It was shown in [9–11] that the TVL1 model works successfully in recovering blurred images corrupted by impulse noise. Notice that the TVL1 model has many difficulties in finding its solutions both mathematically and numerically since both the l_1-norm data fidelity and regularization terms are not differentiable.

Recently, Lu et al. [13] proposed a fixed-point method for solving the following TVL1 variational problem:

$$\min_u \left\{ \|Au - f\|_1 + \frac{\lambda}{2}\|u\|_2^2 + \rho\|u\|_{\mathrm{TV}} : u \in \mathbb{R}^m \right\}, \tag{4}$$

where λ and ρ are positive numbers. They showed that the fixed-point method performs remarkably better in the image quality measured by PSNR and preserves more features than FTVd (Fast total variation deconvolution) proposed in [11] at the expense of much increase in computational time. Furthermore, Han and Yun [14] proposed a fixed-point-like method for solving the following TVL1 variational problem:

$$\min_u \{\|Au - f\|_1 + \lambda\|u\|_2 + \rho\|u\|_{\mathrm{TV}} : u \in \mathbb{R}^m\}, \tag{5}$$

where λ and ρ are positive numbers. They showed that the fixed-point-like method restores the true image much better than the fixed-point method proposed by Lu et al. [13]. These two approaches motivate us to propose the following two TVL1 variational problems

$$\min_u \left\{\|Au - f\|_1 + \frac{\lambda}{2}\|Du\|_2^2 + \rho\|u\|_{\mathrm{TV}} : u \in \mathbb{R}^m\right\}, \tag{6}$$

$$\min_u \{\|Au - f\|_1 + \lambda\|Du\|_2 + \rho\|u\|_{\mathrm{TV}} : u \in \mathbb{R}^m\}, \tag{7}$$

where $D = -\Delta$ and Δ denote a discrete Laplacian operator. Under the reflexive boundary condition, the discrete Laplacian operator is represented by a singular matrix in $\mathbb{R}^{m\times m}$ (see Section 6). The new TVL1 model Equation (6) uses a smooth term $\|Du\|_2^2$ instead of using a smooth term $\|u\|_2^2$. The reason for using the term $\|Du\|_2^2$ is that it may better recover a true image u from an observed image f by taking four neighborhood pixels into account when updating an approximate image during the iteration step of iteration methods to be proposed in this paper. In addition, the new TVL1 model Equation (7) uses a non-smooth term $\|Du\|_2$ instead of using a smooth term $\|Du\|_2^2$ for the purpose of better preserving the edges and corners in the restored images.

Notice that the TVL1 problem Equation (4) has a unique solution since its objective function is strictly convex, while the TVL1 problems Equations (5)–(7) may not have a unique solution since its objective functions are just convex, not strictly convex.

This paper is organized as follows. In Section 2, we provide some definitions and useful properties which are fundamental tools for developing numerical algorithms for solving the TVL1 variational problems Equations (6) and (7). In Section 3, we just provide the fixed-point algorithm proposed in [13] for solving the TVL1 problem Equation (4) and the fixed-point-like algorithm proposed in [14] for solving the TVL1 problem Equation (5) for the purpose of comparison. In Section 4, we propose a fixed-point-like method, using proximal operators, for solving the new proposed TVL1 problem Equation (6). In Section 5, we propose a fixed-point-like method, using proximal operators, for solving the new proposed TVL1 problem Equation (7). In Section 6, we provide numerical experiments for several test images blurred by Gaussian kernel and corrupted by salt-and-pepper impulse noise in order to demonstrate the efficiency and reliability of two fixed-point-like methods for solving the TVL1 problems Equations (6) and (7). Performance evaluation for these two methods is done by comparing their numerical results with those of two fixed-point-like methods for the existing TVL1 problems Equations (4) and (5). Lastly, we provide some concluding remarks.

2. Preliminaries

Some definitions and useful results which we refer to later in this paper are provided below. We first provide the proximal operator introduced by Moreau [15].

Definition 1. *Let $\psi : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ be a proper, convex, and lower semi-continuous (l.s.c) function. The proximal operator of ψ at $v \in \mathbb{R}^m$ is defined by*

$$prox_{\psi}(v) = \operatorname*{argmin}_u \left\{\frac{1}{2} \parallel u - v \parallel_2^2 + \psi(u) : u \in \mathbb{R}^m\right\}. \tag{8}$$

Definition 2. *Let $\psi : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ be a proper, convex, and l.s.c function. The subdifferential of ψ at $v \in \mathbb{R}^m$ is defined by*

$$\partial\psi(v) = \{y \in \mathbb{R}^m : \psi(z) \geq \psi(v) + < y, z - v >, \ \forall z \in \mathbb{R}^m\}. \tag{9}$$

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

It is well known that a subdifferential of a convex function ψ is a set-valued mapping from $\mathbb{R}^m$ into a nonempty convex compact set in $\mathbb{R}^m$ [16]. We now provide explicit formulas for three proximal operators used in this paper, which are the l_1-norm, l_2-norm, and isotropic TV norm [14,16]. The first example gives two proximal operators for the l_1-norm and l_2-norm defined on $\mathbb{R}^m$.

Example 1. *Let $||\cdot||_1$ and $||\cdot||_2$ be the l_1-norm and l_2-norm defined on $\mathbb{R}^m$, respectively. For any $\lambda > 0$ and $v \in \mathbb{R}^m$*

$$\text{prox}_{\frac{1}{\lambda}\|\cdot\|_1}(v) = \max\left\{|v| - \frac{1}{\lambda}, 0\right\} \text{sign}(v),$$

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

where $|v|$ denotes elementwise absolute value of the vector v and $.*$ denotes the elementwise multiplication.

Notice that the isotropic TV of $u \in \mathbb{R}^m$ defined by Equation (3) can be expressed as

$$\|u\|_{\text{TV}} = (\varphi \circ B)(u), \tag{10}$$

where $\varphi : \mathbb{R}^{2m} \to \mathbb{R}$ is a convex function defined by

$$\varphi(v) = \sum_{i=1}^{m} \left\| \begin{pmatrix} v_i \\ v_{m+i} \end{pmatrix} \right\|_2 \text{ for each } v = (v_i) \in \mathbb{R}^{2m}$$

and B is a $d \times m$ matrix which represents a discrete gradient operator ∇ with $m = N^2$ and $d = 2m$ (see Section 6). The next example gives the proximal operator of the convex function $\psi = \frac{1}{\lambda}\varphi$ on $\mathbb{R}^{2m}$ which is called the generalized shrinkage formula, where $\lambda > 0$.

Example 2. *If $\psi = \frac{1}{\lambda}\varphi$ and $v = (v_i) \in \mathbb{R}^{2m}$, then*

$$\text{prox}_{\frac{1}{\lambda}\varphi}(v) = \prod_{i=1}^{m} \left(\text{prox}_{\frac{1}{\lambda}\|\cdot\|_2} \begin{pmatrix} v_i \\ v_{m+i} \end{pmatrix} \right),$$

where $\prod$ denotes Cartesian product of vector spaces.

The following theorem outlines a relationship between the proximal operator and the subdifferential of a convex function.

Theorem 1 ([13,15])**.** *If ψ is a proper, convex and l.s.c. function on $\mathbb{R}^m$ and $v \in \mathbb{R}^m$, then*

$$y \in \partial\psi(v) \ \Leftrightarrow \ v = \text{prox}_{\psi}(v + y) \ \Leftrightarrow \ y = (I - \text{prox}_{\psi})(v + y), \tag{11}$$

where I denotes an identity operator on $\mathbb{R}^m$.

3. Fixed-Point-Like Methods for the TVL1 Problems Equations (4) and (5)

In this section, we just provide the fixed-point algorithm proposed in [13] for solving the TVL1 problem Equation (4) and the fixed-point-like algorithm proposed in [14] for solving the TVL1 problem

Equation (5) for the purpose of comparison with two fixed-point-like algorithms to be proposed in this paper. The fixed-point method, called Algorithm 1, for the TVL1 problem Equation (4) and the fixed-point-like method, called Algorithm 2, for the TVL1 problem Equation (5) are described below (see [13,14] for detailed description of algorithms).

Algorithm 1 Fixed-point method for the TVL1 problem Equation (4)

Given degraded image f, choose positive parameters $\alpha, \beta, \lambda, \rho$
Initialization : $u^0 = 0$, $a^0 = 0$ and $b^0 = 0$
for $k = 0$ to *maxit* **do**
  $a^{k+1} = \left(I - \text{prox}_{\frac{1}{\alpha\lambda}\|\cdot\|_1}\right)\left(Au^k - f + a^k\right)$
  $b^{k+1} = \left(I - \text{prox}_{\frac{1}{\beta\lambda}\varphi}\right)\left(Bu^k + b^k\right)$
  $u^{k+1} = -\left(\alpha A^T a^{k+1} + \rho\beta B^T b^{k+1}\right)$
  if $\frac{\|u^{k+1}-u^k\|_2}{\|u^{k+1}\|_2} < tol$ **then**
    Stop
  end if
end for

Algorithm 2 Fixed-point-like method for the TVL1 problem Equation (5)

Given degraded image f, choose positive parameters $\alpha, \beta, \gamma, \lambda, \rho$
Initialization : $u^0 = 0$, $a^0 = 0$ and $b^0 = 0$
for $k = 0$ to *maxit* **do**
  $a^{k+1} = \left(I - \text{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\right)\left(Au^k - f + a^k\right)$
  $b^{k+1} = \left(I - \text{prox}_{\frac{\rho}{\beta}\varphi}\right)\left(Bu^k + b^k\right)$
  $c^{k+1} = -\frac{1}{\gamma\lambda}\left(\alpha A^T a^{k+1} + \beta B^T b^{k+1}\right)$
  $u^{k+1} = \text{prox}_{\frac{1}{\gamma}\|\cdot\|_2}\left(u^k + c^{k+1}\right)$
  if $\frac{\|u^{k+1}-u^k\|_2}{\|u^{k+1}\|_2} < tol$ **then**
    Stop
  end if
end for

For all algorithms considered in this paper, *maxit* denotes the maximum number of iterations and *tol* denotes the tolerance value of the stopping criterion.

4. Fixed-Point-Like Method for the TVL1 Problem Equation (6)

In this section, we propose a fixed-point-like method, using the proximal operators, for solving the new proposed TVL1 variational problem Equation (6). Equation (6) can be expressed as

$$\min_u \left\{ \|Au - f\|_1 + \frac{\lambda}{2}\|Du\|_2^2 + \rho(\varphi \circ B)(u) : u \in \mathbb{R}^m \right\}, \tag{12}$$

where λ and ρ are positive numbers, φ and B are defined the same as in Equation (10). Using Theorem 1, we can obtain the following property for a solution to the TVL1 problem Equation (12).

Theorem 2. *If φ is a real-valued convex function on $\mathbb{R}^d$, B is an $d \times m$ matrix, A is an $m \times m$ matrix, and u is a solution to the TVL1 problem Equation (12), then, for any $\alpha, \beta > 0$, there exist vectors $a \in \mathbb{R}^m$ and $b \in \mathbb{R}^d$ such that*

$$a = \left(I - \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\right)(Au - f + a), \tag{13}$$

$$b = \left(I - \operatorname{prox}_{\frac{\rho}{\beta}\varphi}\right)(Bu + b), \tag{14}$$

$$0 = \alpha A^T a + \lambda D^T D u + \beta B^T b. \tag{15}$$

Conversely, if there exist positive numbers α, β, $a \in \mathbb{R}^m$, $b \in \mathbb{R}^d$ *and* $u \in \mathbb{R}^m$ *satisfying Equations (13)–(15), then* u *is a solution to the TVL1 problem Equation (12).*

Proof. We assume that $u \in \mathbb{R}^m$ is a solution to the TVL1 problem Equation (12). By the Fermat rule in convex analysis for problem Equation (12) and using the relations $\partial(\varphi \circ B) = B^T \circ (\partial\varphi) \circ B$, we have

$$0 \in A^T(\partial\|\cdot\|_1)(Au - f) + \lambda D^T D u + \rho B^T(\partial\varphi)(Bu). \tag{16}$$

From relation (16), for any $\alpha, \beta > 0$, we can choose a vector $a \in \partial(\frac{1}{\alpha}\|\cdot\|_1)(Au - f)$ and $b \in \partial(\frac{\rho}{\beta}\varphi)(Bu)$ satisfying

$$\alpha A^T a + \lambda D^T D u + \beta B^T b = 0. \tag{17}$$

From Equation (17), we obtain Equation (15). By Theorem 1, the inclusions $a \in \partial(\frac{1}{\alpha}\|\cdot\|_1)(Au - f)$ and $b \in \partial(\frac{\rho}{\beta}\varphi)(Bu)$ lead to Equations (13) and (14), respectively.

Conversely, suppose that there exist α, β, $a \in \mathbb{R}^m$, $b \in \mathbb{R}^d$ and $u \in \mathbb{R}^m$ satisfying Equations (13)–(15). Again, by Theorem 1, Equations (13) and (14) ensure that $a \in \partial(\frac{1}{\alpha}\|\cdot\|_1)(Au - f)$ and $b \in \partial(\frac{\rho}{\beta}\varphi)(Bu)$, respectively. From Equation (15), relation (16) holds. Hence, $u \in \mathbb{R}^m$ is a solution to the TVL1 problem Equation (12). □

By splitting Equations (13) and (14) of Theorem 2 and rearranging the resulting equations, we can obtain the following corollary.

Corollary 1. *If* φ *is a real-valued convex function on* $\mathbb{R}^d$, B *is an* $d \times m$ *matrix,* A *is an* $m \times m$ *matrix, and* u *is a solution to the TVL1 problem Equation (12), then, for any* $\alpha, \beta > 0$, *there exist vectors* $a \in \mathbb{R}^m$ *and* $b \in \mathbb{R}^d$ *such that*

$$a^* = a - f - \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(Au - f + a), \tag{18}$$

$$b^* = b - \operatorname{prox}_{\frac{\rho}{\beta}\varphi}(Bu + b), \tag{19}$$

$$(\alpha A^T A + \lambda D^T D + \beta B^T B)u = -\alpha A^T a^* - \beta B^T b^*, \tag{20}$$

$$a = Au + a^*, \tag{21}$$

$$b = Bu + b^*. \tag{22}$$

Conversely, if there exist positive numbers α, β, $a \in \mathbb{R}^m$, $b \in \mathbb{R}^d$ *and* $u \in \mathbb{R}^m$ *satisfying Equations (18)–(22), then* u *is a solution to the TVL1 problem Equation (31).*

Proof. It is sufficient to show that Equations (13)–(15) in Theorem 2 are equivalent to Equations (18)–(22) in Corollary 1. Splitting Equations (13) and (14) and using notations (18) and (19) for a^* and b^*, one obtains

$$a = Au - f + a - \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(Au - f + a) = Au + a^*, \tag{23}$$

$$b = Bu + b - \operatorname{prox}_{\frac{\rho}{\beta}\varphi}(Bu + b) = Bu + b^*. \tag{24}$$

Hence, we obtain Equations (21) and (22). Substituting Equations (21) and (22) into Equation (15), Equation (20) is also obtained. □

From Equations (18)–(22) of Corollary 1, we can obtain a fixed-point-like method, called Algorithm 2, using the proximal operators for the TVL1 problem Equation (6).

Algorithm 3 Fixed-point-like method for the TVL1 problem Equation (6)

1: Given degraded image f, choose positive parameters $\alpha, \beta, \lambda, \rho$
2: Initialization : $u^0 = f$, $a^0 = 0$ and $b^0 = 0$
3: **for** $k = 0$ to *maxit* **do**
4: $\quad a^{k+\frac{1}{2}} = a^k - f - \text{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(Au^k - f + a^k\right)$
5: $\quad b^{k+\frac{1}{2}} = b^k - \text{prox}_{\frac{\rho}{\beta}\varphi}\left(Bu^k + b^k\right)$
6: $\quad$ Solve $(\alpha A^T A + \beta B^T B + \lambda D^T D)u^{k+1} = -\alpha A^T a^{k+\frac{1}{2}} - \beta B^T b^{k+\frac{1}{2}}$ for u^{k+1}
7: $\quad a^{k+1} = Au^{k+1} + a^{k+\frac{1}{2}}$
8: $\quad b^{k+1} = Bu^{k+1} + b^{k+\frac{1}{2}}$
9: $\quad$ **if** $\frac{\|u^{k+1} - u^k\|_2}{\|u^{k+1}\|_2} < tol$ **then**
10: $\quad\quad$ Stop
11: $\quad$ **end if**
12: **end for**

Notice that the linear system in line 6 of Algorithm 2 is equivalent to solving the following least squares problem

$$\min_u \left\| \begin{pmatrix} -\sqrt{\alpha}\, a^{k+\frac{1}{2}} \\ -\sqrt{\beta}\, b^{k+\frac{1}{2}} \\ 0 \end{pmatrix} - \begin{pmatrix} \sqrt{\alpha}\, A \\ \sqrt{\beta}\, B \\ \sqrt{\lambda}\, D \end{pmatrix} u \right\|_2^2. \tag{25}$$

Hence, the linear system in line 6 of Algorithm 2 is solved by applying the CGLS (Conjugate gradient least squares method [17]) to the problem Equation (25). The following theorem provides a convergence analysis for Algorithm 2.

Theorem 3. *Let $\{a^n\}$, $\{b^n\}$, and $\{u^n\}$ be sequences generated by Algorithm 2. If we can find two consecutive vectors u^k and u^{k+1} such that $u^{k+1} = u^k$ for some positive values of α, β, λ and ρ, then u^{k+1} is a solution to the TVL1 problem Equation (6).*

Proof. Substituting Equations (13) and (14) into Equation (15), one obtains

$$\begin{aligned} 0 = {} & \alpha A^T \left(I - \text{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\right)(Au - f + a) \\ & + \beta B^T \left(I - \text{prox}_{\frac{\rho}{\beta}\varphi}\right)(Bu + b) + \lambda D^T Du. \end{aligned} \tag{26}$$

From Theorem 2, it can be easily seen that, if u, a, and b satisfy Equation (26) for some positive values of α, β, λ and ρ, then u is a solution to the TVL1 problem Equation (6). In Algorithm 2, if $u^{k+1} = u^k$ for some positive values of α, β, λ and ρ, then it can be easily shown that

$$a^{k+1} = (I - \text{prox}_{\frac{1}{\alpha}\|\cdot\|_1})\left(Au^{k+1} - f + a^k\right), \tag{27}$$

$$b^{k+1} = (I - \text{prox}_{\frac{\rho}{\beta}\varphi})\left(Bu^{k+1} + b^k\right). \tag{28}$$

Simplifying the linear system in line 6 of Algorithm 2 using the relations in lines 7 and 8 of Algorithm 2, one can obtain

$$\alpha A^T a^{k+1} + \beta B^T b^{k+1} + \lambda D^T Du^{k+1} = 0. \tag{29}$$

Substituting Equations (27) and (28) into Equation (29), one obtains

$$\begin{aligned} 0 = \alpha A^T \left(I - \text{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\right)\left(Au^{k+1} - f + a^k\right) \\ + \beta B^T \left(I - \text{prox}_{\frac{\rho}{\beta}\varphi}\right)\left(Bu^{k+1} + b^k\right) + \lambda D^T D u^{k+1}. \end{aligned} \tag{30}$$

From Equation (30), it can be seen that u^{k+1}, a^k, and b^k satisfy Equation (26) for some positive values of α, β, λ and ρ. Hence, u^{k+1} is a solution to the TVL1 problem Equation (6). □

Theorem 3 gives an idea of how to stop Algorithm 2. In practical applications, we do not have to find u^{k+1} which is equal to u^k. Instead, we need to find u^{k+1} which is reasonably close to u^k. Hence, we have used the following stopping criterion in Algorithm 2

$$\frac{\|u^{k+1} - u^k\|_2}{\|u^{k+1}\|_2} < tol,$$

where *tol* is a suitably chosen small tolerance value.

5. Fixed-Point-Like Method for the TVL1 Problem Equation (7)

In this section, we propose a fixed-point-like method, using the proximal operators, for solving the new proposed TVL1 variational problem Equation (7). Equation (7) can be expressed as

$$\min_u \left\{\|Au - f\|_1 + \lambda\|Du\|_2 + \rho(\varphi \circ B)(u) : u \in \mathbb{R}^m\right\}, \tag{31}$$

where λ and ρ are positive numbers, φ and B are defined the same as in Equation (10). Using Theorem 1, we can obtain the following property for a solution to the TVL1 problem Equation (31).

Theorem 4. *If φ is a real-valued convex function on $\mathbb{R}^d$, B is an $d \times m$ matrix, A is an $m \times m$ matrix, and u is a solution to the TVL1 problem Equation (31), then, for any $\alpha, \beta, \gamma > 0$, there exist vectors $a, c \in \mathbb{R}^m$ and $b \in \mathbb{R}^d$ such that*

$$a = \left(I - \text{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\right)(Au - f + a), \tag{32}$$

$$b = \left(I - \text{prox}_{\frac{\rho}{\beta}\varphi}\right)(Bu + b), \tag{33}$$

$$c = \left(I - \text{prox}_{\frac{1}{\gamma}\|\cdot\|_2}\right)(Du + c), \tag{34}$$

$$0 = \alpha A^T a + \beta B^T b + \gamma\lambda D^T c. \tag{35}$$

Conversely, if there exist positive numbers α, β, γ and vectors $a, c \in \mathbb{R}^m, b \in \mathbb{R}^d$, and $u \in \mathbb{R}^m$ satisfying Equations (32)–(35), then u is a solution to the TVL1 problem Equation (31).

Proof. We assume that $u \in \mathbb{R}^m$ is a solution to the TVL1 problem (31). Applying the Fermat rule in convex analysis to problem (31), one can obtain

$$0 \in A^T(\partial\|\cdot\|_1)(Au - f) + B^T(\partial\rho\varphi)(Bu) + \lambda D^T(\partial\|\cdot\|_2)(Du). \tag{36}$$

From relation (36), for any $\alpha, \beta, \gamma > 0$, we can choose a vector $a \in \partial(\frac{1}{\alpha}\|\cdot\|_1)(Au - f)$, $b \in \partial(\frac{\rho}{\beta}\varphi)(Bu)$ and $c \in \partial(\frac{1}{\gamma}\|\cdot\|_2)(Du)$ satisfying

$$\alpha A^T a + \beta B^T b + \gamma\lambda D^T c = 0.$$

Hence, Equation (35) holds. By Theorem 1, the inclusions $a \in \partial(\frac{1}{\alpha}\|\cdot\|_1)(Au-f)$, $b \in \partial(\frac{\rho}{\beta}\varphi)(Bu)$ and $c \in \partial(\frac{1}{\gamma}\|\cdot\|_2)(Du)$ lead to Equations (32)–(34), respectively.

Conversely, suppose that there exist $\alpha, \beta, \gamma > 0$, $a, c, u \in \mathbb{R}^m$, and $b \in \mathbb{R}^d$ satisfying Equations (32)–(35). By Theorem 1, Equations (32)–(34) ensure that $a \in \partial(\frac{1}{\alpha}\|\cdot\|_1)(Au-f)$, $b \in \partial(\frac{\rho}{\beta}\varphi)(Bu)$ and $c \in \partial(\frac{1}{\gamma}\|\cdot\|_2)(Du)$, respectively. From Equation (35), relation (36) holds. Hence, $u \in \mathbb{R}^m$ is a solution to the TVL1 problem Equation (31). □

By splitting Equations (32)–(34) of Theorem 4 and rearranging the resulting equations, we can obtain the following corollary.

Corollary 2. *If φ is a real-valued convex function on $\mathbb{R}^d$, B is an $d \times m$ matrix, A is an $m \times m$ matrix, and u is a solution to the TVL1 problem Equation (31), then, for any $\alpha, \beta, \gamma > 0$ there exist vectors $a, c \in \mathbb{R}^m$ and $b \in \mathbb{R}^d$ such that*

$$a^* = a - f - \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(Au - f + a), \tag{37}$$

$$b^* = b - \operatorname{prox}_{\frac{\rho}{\beta}\varphi}(Bu + b), \tag{38}$$

$$c^* = c - \operatorname{prox}_{\frac{1}{\gamma}\|\cdot\|_2}(Du + c), \tag{39}$$

$$(\alpha A^TA + \beta B^TB + \gamma\lambda D^TD)u = -\alpha A^Ta^* - \beta B^Tb^* - \gamma\lambda D^Tc^*, \tag{40}$$

$$a = Au + a^*, \tag{41}$$

$$b = Bu + b^*, \tag{42}$$

$$c = Du + c^*. \tag{43}$$

Conversely, if there exist positive numbers α, β, γ and vectors $a, c \in \mathbb{R}^m$, $b \in \mathbb{R}^d$, and $u \in \mathbb{R}^m$ satisfying Equations (37)–(43), then u is a solution to the TVL1 problem Equation (31).

Proof. It is sufficient to show that Equations (32)–(35) in Theorem 4 are equivalent to Equations (37)–(43) in Corollary 2. Splitting Equations (32)–(34) and using notations (37)–(39) for a^*, b^* and c^*, one obtains

$$a = Au - f + a - \operatorname{prox}_{\frac{1}{\alpha}\|\cdot\|_1}(Au - f + a) = Au + a^*, \tag{44}$$

$$b = Bu + b - \operatorname{prox}_{\frac{\rho}{\beta}\varphi}(Bu + b) = Bu + b^*, \tag{45}$$

$$c = Du + c - \operatorname{prox}_{\frac{1}{\gamma}\|\cdot\|_2}(Du + c) = Du + c^*. \tag{46}$$

From Equations (44)–(46), Equations (41)–(43) are obtained. Substituting Equations (41)–(43) into Equation (35), Equation (40) is also obtained. □

From Equations (37)–(43) of Corollary 2, we can obtain a fixed-point-like method, called Algorithm 3, using the proximal operators for the TVL1 problem Equation (7).

Notice that the linear system in line 7 of Algorithm 3 is equivalent to solving the following least squares problem:

$$\min_u \left\| \begin{pmatrix} -\sqrt{\alpha}\, a^{k+\frac{1}{2}} \\ -\sqrt{\beta}\, b^{k+\frac{1}{2}} \\ -\sqrt{\gamma\lambda}\, c^{k+\frac{1}{2}} \end{pmatrix} - \begin{pmatrix} \sqrt{\alpha}\, A \\ \sqrt{\beta}\, B \\ \sqrt{\gamma\lambda}\, D \end{pmatrix} u \right\|_2^2. \tag{47}$$

Hence, the linear system in line 7 of Algorithm 3 is also solved by applying the CGLS to the problem Equation (47). The following theorem provides a convergence analysis for Algorithm 3.

Algorithm 4 Fixed-point-like method for the TVL1 problem Equation (7)

1: Given degraded image f, choose positive parameters $\alpha, \beta, \gamma, \lambda, \rho$
2: Initialization : $u^0 = 0$, $a^0 = 0$, $b^0 = 0$ and $c^0 = 0$
3: **for** $k = 0$ to *maxit* **do**
4: $a^{k+\frac{1}{2}} = a^k - f - \text{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\left(Au^k - f + a^k\right)$
5: $b^{k+\frac{1}{2}} = b^k - \text{prox}_{\frac{\rho}{\beta}\varphi}\left(Bu^k + b^k\right)$
6: $c^{k+\frac{1}{2}} = c^k - \text{prox}_{\frac{1}{\gamma}\|\cdot\|_2}\left(Du^k + c^k\right)$
7: Solve $(\alpha A^TA + \beta B^TB + \gamma\lambda D^TD)u^{k+1} = -\alpha A^Ta^{k+\frac{1}{2}} - \beta B^Tb^{k+\frac{1}{2}} - \gamma\lambda D^Tc^{k+\frac{1}{2}}$ for u^{k+1}
8: $a^{k+1} = Au^{k+1} + a^{k+\frac{1}{2}}$
9: $b^{k+1} = Bu^{k+1} + b^{k+\frac{1}{2}}$
10: $c^{k+1} = Du^{k+1} + c^{k+\frac{1}{2}}$
11: **if** $\frac{\|u^{k+1}-u^k\|_2}{\|u^{k+1}\|_2} < tol$ **then**
12: Stop
13: **end if**
14: **end for**

Theorem 5. *Let* $\{a^n\}$, $\{b^n\}$, $\{c^n\}$, *and* $\{u^n\}$ *be sequences generated by Algorithm 3. If we can find two consecutive vectors* u^k *and* u^{k+1} *such that* $u^{k+1} = u^k$ *for some positive values of* $\alpha, \beta, \gamma, \lambda$ *and* ρ, *then* u^{k+1} *is a solution to the TVL1 problem Equation (7).*

Proof. Substituting Equations (32)–(34) into Equation (35), one obtains

$$\begin{aligned} 0 = {} & \alpha A^T\left(I - \text{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\right)(Au - f + a) \\ & + \beta B^T\left(I - \text{prox}_{\frac{\rho}{\beta}\varphi}\right)(Bu + b) \\ & + \gamma\lambda D^T\left(I - \text{prox}_{\frac{1}{\gamma}\|\cdot\|_2}\right)(Du + c)\,. \end{aligned} \tag{48}$$

Theorem 4 implies that if u, a, b and c satisfy Equation (48) for some positive values of $\alpha, \beta, \gamma, \lambda$ and ρ, then u is a solution to the TVL1 problem Equation (7). In Algorithm 3, if $u^{k+1} = u^k$ for some positive values of $\alpha, \beta, \gamma, \lambda$ and ρ, then it can be shown that

$$a^{k+1} = (I - \text{prox}_{\frac{1}{\alpha}\|\cdot\|_1})(Au^{k+1} - f + a^k), \tag{49}$$

$$b^{k+1} = (I - \text{prox}_{\frac{\rho}{\beta}\varphi})(Bu^{k+1} + b^k), \tag{50}$$

$$c^{k+1} = (I - \text{prox}_{\frac{1}{\gamma}\|\cdot\|_2})(Du^{k+1} + c^k). \tag{51}$$

Simplifying the linear system in line 7 of Algorithm 3 using the relations in lines 8 to 10 of Algorithm 3, one can obtain

$$\alpha A^Ta^{k+1} + \beta B^Tb^{k+1} + \gamma\lambda D^Tc^{k+1} = 0. \tag{52}$$

Substituting Equations (49)–(51) into Equation (52), one obtains

$$\begin{aligned} 0 = {} & \alpha A^T\left(I - \text{prox}_{\frac{1}{\alpha}\|\cdot\|_1}\right)\left(Au^{k+1} - f + a^k\right) \\ & + \beta B^T\left(I - \text{prox}_{\frac{\rho}{\beta}\varphi}\right)\left(Bu^{k+1} + b^k\right) \\ & + \gamma\lambda D^T\left(I - \text{prox}_{\frac{1}{\gamma}\|\cdot\|_2}\right)\left(Du^{k+1} + c^k\right). \end{aligned} \tag{53}$$

Equation (53) implies that u^{k+1}, a^k, b^k and c^k satisfy Equation (48) for some positive values of $\alpha, \beta, \gamma, \lambda$ and ρ. Hence, u^{k+1} is a solution to the TVL1 problem Equation (7). □

Theorem 5 also gives an idea of how to stop Algorithm 3. Thus, Algorithm 3 has the same stopping criterion as Algorithm 2.

6. Numerical Experiments

In this section, we provide numerical experiments for several test problems to evaluate the efficiency of two fixed-point-like methods, called Algorithms 3 and 4, proposed in this paper. Performances of these two algorithms are analyzed by comparing their numerical results with those of Algorithms 1 and 2.

In Section 2, it was shown that the isotropic TV of $u \in \mathbb{R}^m$ can be represented by $||u||_{\mathrm{TV}} = (\varphi \circ B)(u)$, where φ and B are defined the same as in Equation (10). Let $D_x = D_y$ be the first order backward finite difference matrix of order N defined by

$$D_x = D_y = \begin{pmatrix} 0 & 0 & \cdots & \cdots & 0 \\ -1 & 1 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & -1 & 1 & 0 \\ 0 & \cdots & 0 & -1 & 1 \end{pmatrix}.$$

Then, B can be expressed as a $d \times m$ matrix given by

$$B = \begin{pmatrix} I_N \otimes D_x \\ D_y \otimes I_N \end{pmatrix},$$

where $\otimes$ denotes the Kronecker product, I_N denotes the identity matrix of order N, $m = N^2$, and $d = 2m$. The discrete negative Laplacian operator $D = -\Delta$ can be represented by an $m \times m$ matrix $(I_N \otimes D_{xx} + D_{yy} \otimes I_N)$, where $D_{xx} = D_{yy}$ is the second order finite difference matrix of order N defined by

$$D_{xx} = D_{yy} = \begin{pmatrix} 1 & -1 & \cdots & \cdots & 0 \\ -1 & 2 & -1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & -1 & 2 & -1 \\ 0 & \cdots & 0 & -1 & 1 \end{pmatrix}.$$

In order to illustrate the efficiency and reliability of two fixed-point-like methods, called Algorithms 3 and 4, for solving the new proposed TVL1 problems Equations (6) and (7), we provide numerical results for five test images such as Cameraman, Lena, House, Boat, and Pepper (see Figure 1). The pixel size of five test images is 256×256. All numerical tests have been performed using Matlab R2019a (Mathworks, Natick, MA, USA) on a personal computer with 3.2 GHz CPU and 8 GB RAM. *maxit* is set to 3500 (for Algorithms 1 and 2) or 500 (for Algorithms 3 and 4), and *tol* is set to 1×10^{-5} (for Algorithm 1), 1.5×10^{-4} (for Algorithm 2), 2×10^{-4} (for Algorithm 3) or 2×10^{-3} (for Algorithm 4). For the CGLS method which is used to solve a linear system every iteration of Algorithms 3 and 4, the tolerance for stopping criterion is set to 5×10^{-3} (for Algorithm 3) or 1×10^{-2} (for Algorithm 4), and the maximum number of iterations is set to 25. Note that the CGLS uses a symmetric preconditioner to accelerate its convergence. To see how to construct the symmetric preconditioner, we refer to [18,19].

To evaluate the quality of the restored images, we use the PSNR (peak signal-to-noise ratio) value between the restored image and original image which is defined by

$$\mathrm{PSNR} = 10 \log_{10} \left(\frac{N^2 \max_{i,j} |u_{ij}|^2}{\|U - \tilde{U}\|_F^2} \right)$$

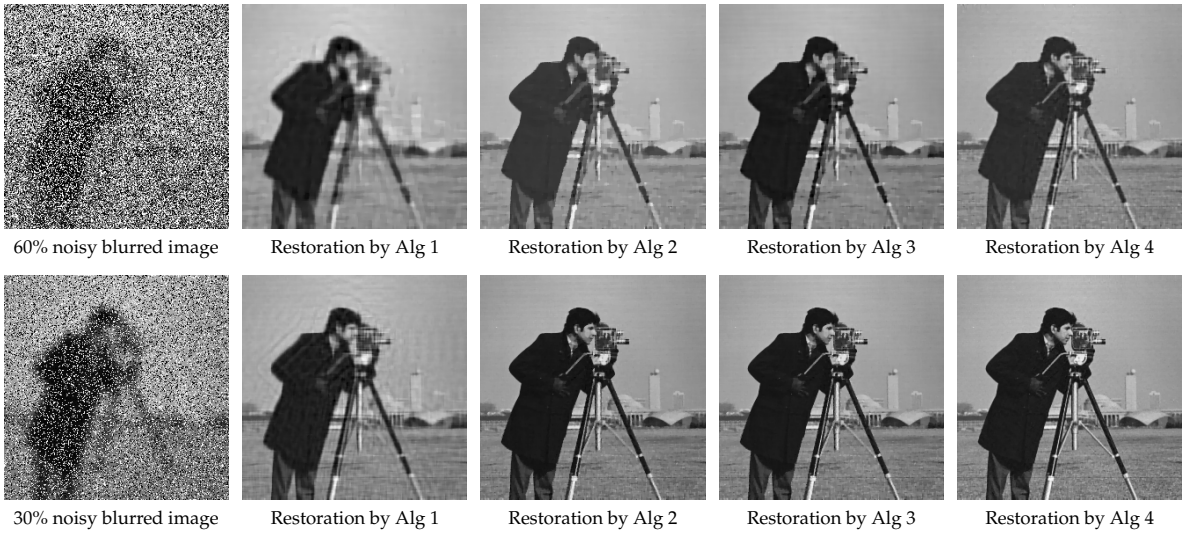
where $\|\cdot\|_F$ refers to the Frobenius norm, U and $\tilde{U}$ are the original and restored images with size $N \times N$, respectively. In addition, u_{ij} stands for the value of original image U at the pixel point (i, j) and N^2 is the total number of pixels. It is generally true that the larger PSNR value stands for the better quality of restored image.

Cameraman image Lena image House image Boat image Pepper image

Figure 1. True images for Cameraman, Lena, House, Boat, and Pepper.

For all numerical experiments, we have used the test images with an intensity range of $[0, 1]$. For all test problems, we choose the degraded test images which are resulting images blurred by Gaussian kernel of size 15×15 with standard deviation 9 under the reflexive boundary condition, and then corrupted by salt-and-pepper impulse noise with noise levels 20%, 30%, 60%, or 70%. In Tables 1–4, P_0 represents the PSNR values for the blurred and noisy images f, Alg denotes the algorithm to be used, Cam denotes the Cameraman image, PSNR represents the PSNR value for the restored image, Iter denotes the number of iterations, and CPU denotes the elapsed CPU time in seconds. All parameters α, β, γ, λ, and ρ are chosen as the best one by numerical tries.

Tables 1–4 contain numerical results of Algorithms 1–4 for degraded test images with 20%, 30%, 60%, or 70% salt-and-pepper impulse noises. In Table 1, we do not provide numerical results for Algorithms 1 and 2 since it does not converges within $maxit = 3500$. Figure 1 contains the true images for Cameraman, Lena, House, Boat, and Pepper. In Figures 2–6, the first row contains the images restored by Algorithms 1–4 for blurred images with 60% salt-and-pepper noise, and the second row contains the images restored by Algorithms 1–4 for blurred images with 30% salt-and-pepper noise.



60% noisy blurred image Restoration by Alg 1 Restoration by Alg 2 Restoration by Alg 3 Restoration by Alg 4

30% noisy blurred image Restoration by Alg 1 Restoration by Alg 2 Restoration by Alg 3 Restoration by Alg 4

Figure 2. Image restoration for blurred Cameraman image with 60% or 30% salt-and-pepper noise.

Figure 3. Image restoration for blurred Lena image with 60% or 30% salt-and-pepper noise.

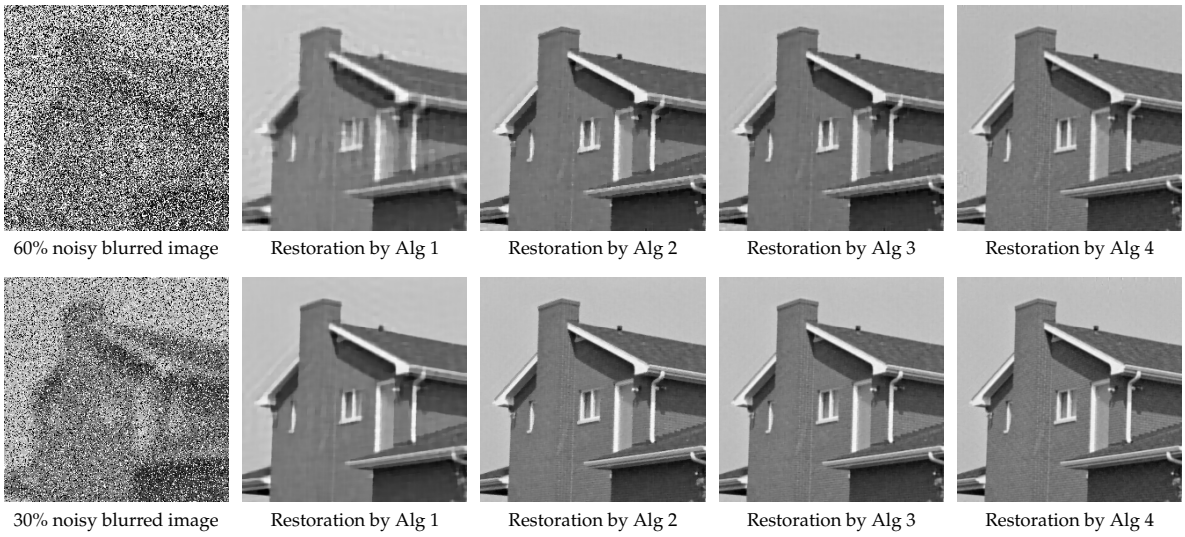


Figure 4. Image restoration for blurred House image with 60% or 30% salt-and-pepper noise.

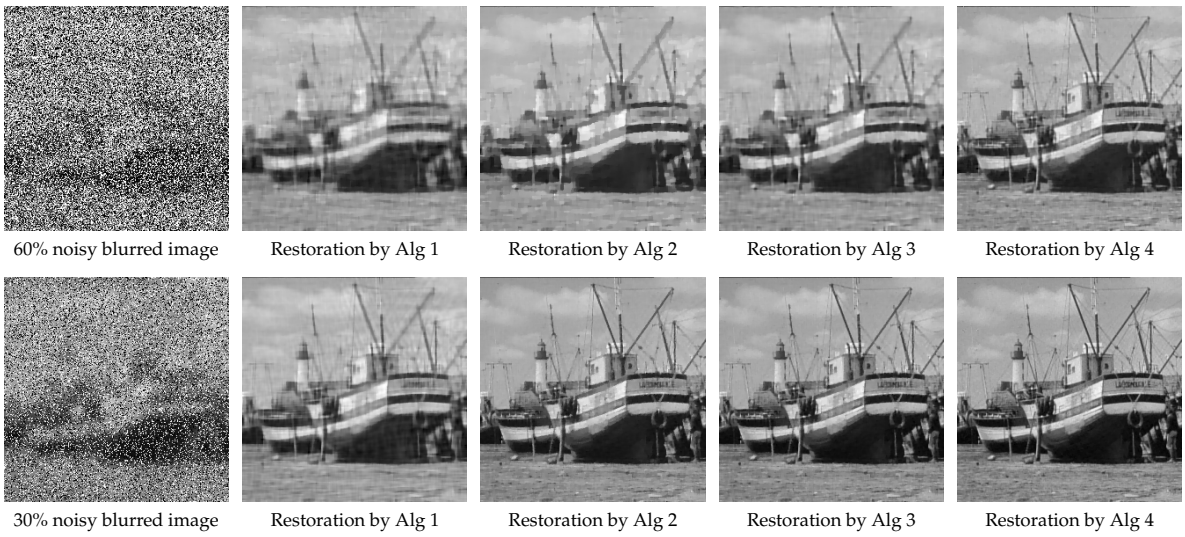


Figure 5. Image restoration for blurred Boat image with 60% or 30% salt-and-pepper noise.

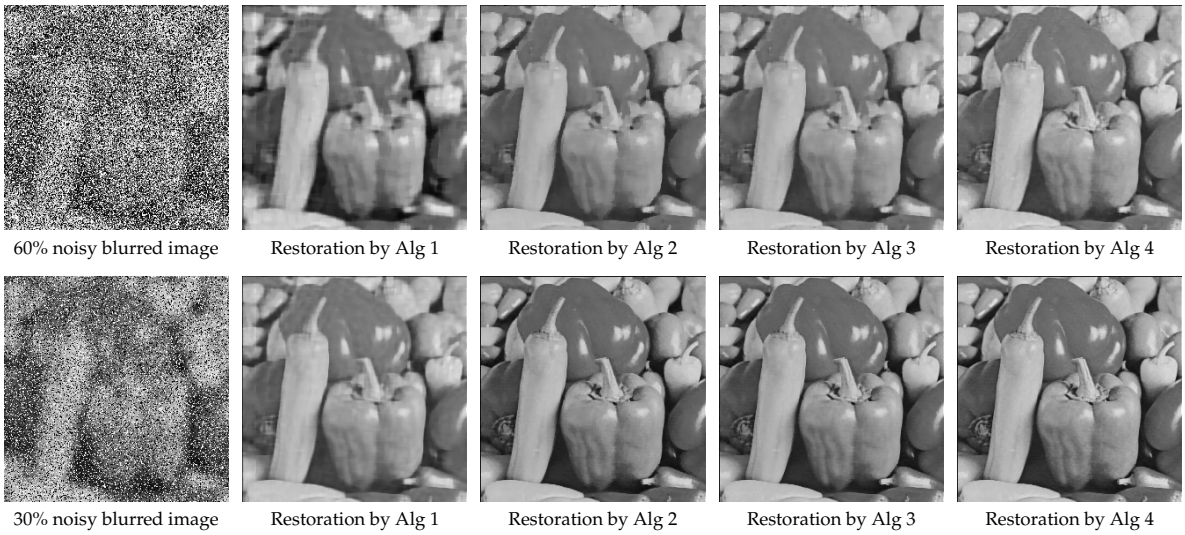


Figure 6. Image restoration for blurred Pepper image with 60% or 30% salt-and-pepper noise.

Table 1. Numerical results for TVL1 problems with 20% salt-and-pepper impulse noise.

Image	P_0	Alg	α	β	γ	λ	ρ	tol	PSNR	Iter	CPU
Cam	11.42	3	130	0.09		0.0001	0.0017	2×10^{-4}	32.71	146	81
		4	90	0.001	0.0003	0.004	0.00007	2×10^{-3}	35.52	257	107
Lena	11.52	3	270	0.09		0.0002	0.0015	2×10^{-4}	32.33	133	71
		4	150	0.0008	0.0003	0.004	0.00007	2×10^{-3}	34.38	317	106
House	11.54	3	500	0.2		0.002	0.0015	2×10^{-4}	38.74	123	61
		4	170	0.0004	0.0006	0.004	0.00005	2×10^{-3}	40.53	339	90
Boat	11.11	3	480	0.2		0.0008	0.0017	2×10^{-4}	33.14	163	83
		4	160	0.0004	0.0007	0.004	0.00004	2×10^{-3}	35.88	290	77
Pepper	11.65	3	270	0.09		0.0002	0.0015	2×10^{-4}	36.57	136	71
		4	170	0.0004	0.0006	0.004	0.00005	2×10^{-3}	38.01	316	82

Table 2. Numerical results for TVL1 problems with 30% salt-and-pepper impulse noise.

Image	P_0	Alg	α	β	γ	λ	ρ	tol	PSNR	Iter	CPU
Cam	9.86	1	2.2	0.1		0.14	0.004	1×10^{-5}	25.12	2323	868
		2	6.3	0.001	1.3	1.03	0.003	1.5×10^{-4}	30.03	1843	678
		3	450	0.008		0.0001	0.003	2×10^{-4}	30.49	243	80
		4	115.0	0.0004	0.0004	0.003	0.0002	2×10^{-3}	33.46	359	101
Lena	9.92	1	2.2	0.07		0.13	0.01	1×10^{-5}	26.65	3122	1172
		2	6.3	0.0003	1.3	1.1	0.003	1.5×10^{-4}	30.09	1845	679
		3	50.0	0.0002		0.0008	0.003	2×10^{-4}	30.42	270	44
		4	40.0	0.0006	0.0006	0.005	0.00004	2×10^{-3}	32.78	195	85
House	9.91	1	2.2	0.2		0.15	0.02	1×10^{-5}	31.20	2195	829
		2	6.6	0.00003	1.3	1.1	0.003	1.5×10^{-4}	36.57	2332	857
		3	460	0.002		0.005	0.003	2×10^{-4}	37.02	161	21
		4	37.0	0.0005	0.0007	0.004	0.00005	2×10^{-3}	39.03	191	79
Boat	9.55	1	2.2	0.06		0.14	0.007	1×10^{-5}	26.23	2907	1082
		2	6.6	0.00008	1.3	1.1	0.003	1.5×10^{-4}	30.43	1668	624
		3	450	0.002		0.001	0.003	2×10^{-4}	30.75	228	41
		4	36.0	0.0006	0.0004	0.005	0.00005	2×10^{-3}	33.99	206	94
Pepper	10.13	1	2.2	0.07		0.13	0.01	1×10^{-5}	28.11	3060	1123
		2	6.3	0.0003	1.3	1.1	0.003	1.5×10^{-4}	34.15	1680	628
		3	50.0	0.0002		0.0008	0.003	2×10^{-4}	34.58	230	37
		4	37.0	0.0005	0.0007	0.004	0.00005	2×10^{-3}	36.51	178	74

Table 3. Numerical results for TVL1 problems with 60% salt-and-pepper impulse noise.

Image	P_0	Alg	α	β	γ	λ	ρ	tol	PSNR	Iter	CPU
Cam	7.11	1	2.6	0.2		0.29	0.02	1×10^{-5}	22.11	1433	536
		2	6.3	0.002	1.1	1.1	0.014	1.5×10^{-4}	24.86	1096	409
		3	6.70	0.002		0.004	0.014	2×10^{-4}	24.95	287	58
		4	23.5	0.03	0.003	0.01	0.0011	2×10^{-3}	26.96	227	118
Lena	7.12	1	2.6	0.2		0.29	0.03	1×10^{-5}	23.89	1716	643
		2	5.9	0.0008	1.1	1.1	0.015	1.5×10^{-4}	26.38	1143	428
		3	20.0	0.002		0.009	0.013	2×10^{-4}	26.54	246	40
		4	27.5	0.0019	0.0017	0.0081	0.0007	2×10^{-3}	28.55	320	110
House	7.07	1	2.5	0.3		0.25	0.02	1×10^{-5}	27.17	1313	513
		2	5.3	0.0008	0.7	1.3	0.015	1.5×10^{-4}	31.15	958	356
		3	20.0	0.003		0.009	0.013	2×10^{-4}	31.53	175	34
		4	24.0	0.0018	0.003	0.0082	0.0004	2×10^{-3}	34.44	250	86
Boat	6.76	1	2.5	0.2		0.24	0.01	1×10^{-5}	23.13	1553	582
		2	1.5	0.001	1.4	2.2	0.015	1.5×10^{-4}	24.72	1546	578
		3	10.0	0.001		0.020	0.011	2×10^{-4}	25.00	242	41
		4	38.5	0.002	0.002	0.008	0.001	2×10^{-3}	28.43	320	101
Pepper	7.40	1	2.5	0.3		0.25	0.02	1×10^{-5}	23.79	1507	555
		2	5.3	0.0008	0.7	1.3	0.015	1.5×10^{-4}	27.77	877	328
		3	20.0	0.002		0.009	0.013	2×10^{-4}	27.88	258	41
		4	27.5	0.0019	0.0017	0.0081	0.0007	2×10^{-3}	31.22	309	105

Table 4. Numerical results for TVL1 problems with 70% salt-and-pepper impulse noise.

Image	P_0	Alg	α	β	γ	λ	ρ	tol	PSNR	Iter	CPU
Cam	6.47	1	2.0	0.77		0.10	0.05	1×10^{-5}	22.26	2258	835
		2	6.0	0.30	5.0	1.50	0.03	1.5×10^{-4}	23.40	709	266
		3	2.6	0.03		0.01	0.03	2×10^{-4}	23.49	169	46
		4	8.5	0.017	0.09	0.018	0.022	2×10^{-3}	23.55	93	34
Lena	6.47	1	2.0	0.77		0.10	0.05	1×10^{-5}	24.11	2605	960
		2	6.0	0.50	5.0	1.50	0.03	1.5×10^{-4}	24.84	649	243
		3	24.0	0.006		0.03	0.03	2×10^{-4}	25.11	247	40
		4	20.0	0.017	0.02	0.05	0.0065	2×10^{-3}	25.48	166	46
House	6.40	1	1.9	0.80		0.09	0.05	1×10^{-5}	27.12	2095	786
		2	10.0	1.00	7.0	1.90	0.03	1.5×10^{-4}	28.71	546	204
		3	26.0	0.03		0.02	0.027	2×10^{-4}	29.18	154	35
		4	18.0	0.018	0.02	0.05	0.0065	2×10^{-3}	29.54	154	44
Boat	6.11	1	2.0	0.77		0.10	0.05	1×10^{-5}	22.54	2563	940
		2	5.0	0.50	5.0	1.90	0.03	1.5×10^{-4}	23.24	570	213
		3	7.0	0.003		0.04	0.024	2×10^{-4}	23.48	278	46
		4	13.0	0.019	0.02	0.05	0.0053	2×10^{-3}	24.04	135	44
Pepper	6.75	1	1.9	0.80		0.09	0.05	1×10^{-5}	23.47	2662	978
		2	6.0	0.50	4.0	1.50	0.03	1.5×10^{-4}	24.66	506	189
		3	26.0	0.03		0.02	0.027	2×10^{-4}	24.87	169	40
		4	12.0	0.02	0.02	0.05	0.01	2×10^{-3}	25.20	109	36

As can be seen in Tables 1–4, Algorithm 4 for the TVL1 problem Equation (7) restores the true image best, and Algorithm 1 for the TVL1 problem Equation (4) restores the true image worst. That is, Algorithm 4 yields the highest PSNR values, and Algorithm 1 yields the lowest PSNR values. Algorithm 4 for the TVL1 problem Equation (7) restores the true image much better than Algorithm 3

for the TVL1 problem Equation (6), while Algorithm 4 takes much more CPU time than Algorithm 3 in most cases. In addition, Algorithms 3 and 4 for the new proposed TVL1 problems Equations (6) and (7) perform better than Algorithms 1 and 2 for the TVL1 problems Equations (4) and (5) in both PSNR values and CPU time. Based on numerical results, Algorithm 4 for solving the new TVL1 problem Equation (7) is preferred over Algorithm 3 for solving the new TVL1 problem Equation (6), though it may take more CPU time than Algorithm 3 in most cases.

7. Conclusions

In this paper, we proposed two new TVL1 variational problems Equations (6) and (7) for restoring images degraded by blurring and impulse noise, and then we proposed two fixed-point-like methods, called Algorithms 3 and 4, for solving the new TVL1 problems Equations (6) and (7). Numerical experiments showed that Algorithms 3 and 4 for solving the new proposed TVL1 problems Equations (6) and (7) perform better than Algorithms 1 and 2 for solving the existing TVL1 problems Equations (4) and (5) in both PSNR values and CPU time, and Algorithm 4 restores the true image much better than Algorithm 3 at the expense of some increase in CPU time. Hence, it can be concluded that Algorithms 3 and 4 are preferred over Algorithms 1 and 2, and Algorithm 4 for the new TVL1 problem Equation (7) is strongly recommended to use even though it may take more CPU time than Algorithm 3 for the new TVL1 problem Equation (6).

The fixed-point-like methods and TVL1 problems Equations (6) and (7) proposed in this paper can be applied to an image inpainting problem or image denoising problem with impulse noise. Future work will study these kinds of problems. Notice that the problem of finding optimal parameters for fixed-point-like methods is a very challenging problem. Future work will also study how to choose optimal or near optimal parameters for the fixed-point-like methods proposed in this paper.

Author Contributions: Conceptualization, J.H.Y.; Formal analysis, J.H.Y.; Methodology, J.H.Y. and H.J.L.; Software, J.H.Y. and H.J.L.; Writing—original draft, J.H.Y. and H.J.L.; Writing—review and editing, J.H.Y. and H.J.L. Both authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) funded by the Korea government (MSIT) (No. 2019R1F1A1060718).

Acknowledgments: The authors would like to thank the anonymous reviewers for their valuable comments and suggestions which greatly improved the quality of the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ma, L.; Yu, J.; Zeng, T. Sparse representation prior and total variation-based image deblurring under impulse noise. *SIAM J. Imaging Sci.* **2013**, *6*, 2258–2284. [CrossRef]
2. Chen, C.L.P.; Liu, L.; Chen, L.; Tang, Y.Y.; Zhou, Y. Weighted couple sparse representation with classified regularization for impulse noise removal. *IEEE Trans. Image Process.* **2015**, *24*, 4014–4026. [CrossRef] [PubMed]
3. Chambolle, A.; Pock, T. A first order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vis.* **2011**, *40*, 120–145. [CrossRef]
4. Chan, T.F.; Esedoglu, S. Aspects of total variation regularized L1 function approximation. *SIAM J. Appl. Math.* **2005**, *65*, 1817–1837. [CrossRef]
5. Darbon, J.; Sigelle, M. Image restoration with discrete constrained total variation Part I: Fast and exact optimization. *J. Math. Imaging Vis.* **2006**, *26*, 261–276. [CrossRef]
6. Dong, Y.; Hintermüller, M.; Neri, M. An efficient primal–dual method for L1-TV image restoration. *SIAM J. Imaging Sci.* **2009**, *2*, 1168–1189. [CrossRef]
7. Duval, V.; Aujol, J.-F.; Gousseau, Y. The TVL1 model: A geometric point of view. *SIAM J. Multiscale Model. Simul.* **2009**, *8*, 154–189. [CrossRef]
8. Li, Q.; Micchelli, C.A.; Shen, L.; Xu, Y. A proximity algorithm accelerated by Gauss–Seidel iterations for L1/TV denoising models. *Inverse Probl.* **2012**, *28*, 095003. [CrossRef]

9. Micchelli, C.A.; Shen, L.; Xu, Y.; Zeng, X. Proximity algorithms for the L1/TV image denoising model. *Adv. Comput. Math.* **2013**, *38*, 401–426. [CrossRef]
10. Nikolova, M. A variational approach to remove outliers and impulse noise. *J. Math. Imaging Vis.* **2004**, *20*, 99–120. [CrossRef]
11. Yang, J.F.; Zhang, Y.; Yin, W.T. An efficient TVL1 algorithm for deblurring multichannel images corrupted by impulsive noise. *SIAM J. Sci. Comput.* **2008**, *31*, 2842–2865. [CrossRef]
12. Yin, W.; Goldfarb, D.; Osher, S. The total variation regularized L1 model for multiscale decomposition. *SIAM J. Multiscale Model. Simul.* **2007**, *6*, 190–211. [CrossRef]
13. Lu, J.; Qiao, K.; Shen, L.; Zou, Y. Fixed-point algorithms for a TVL1 image restoration model. *Int. J. Comput. Math.* **2018**, *95*, 1829–1844. [CrossRef]
14. Han, Y.D.; Yun, J.H. Performance of fixed-point-like methods for a TVL1 problem with impulse noise. *Int. J. Eng. Res. Technol. Comput. Math.* **2019**, *12*, 2431–2439.
15. Moreau, J.J. Proximité et dualité dans un espace Hilbertien. *Bull. Soc. Math. France* **1965**, *93*, 273–299. [CrossRef]
16. Beck, A. *First-Order Methods in Optimization*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2017.
17. Bjorck, A. *Numerical Methods for Least Squares Problems*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 1996.
18. Hansen, P.C.; Nagy, J.G.; O'Leary, D.P. *Deblurring Images: Matrices, Spectra, and Filtering*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2006.
19. Yun, J.H. Performance of relaxed iterative methods for image deblurring problems. *J. Algorithms Comput. Technol.* **2019**, *13*, 861732. [CrossRef]