Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2013, Article ID 797239, 14 pages
http://dx.doi.org/10.1155/2013/797239

Research Article

Efficient Algorithm for Isotropic and Anisotropic Total Variation Deblurring and Denoising

Yuying Shi[1] and Qianshun Chang[2]

[1] *Department of Mathematics and Physics, North China Electric Power University, Beijing 102206, China*
[2] *Institute of Applied Mathematics, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, China*

Correspondence should be addressed to Yuying Shi; yyshi@amss.ac.cn

Received 17 October 2012; Accepted 24 January 2013

Academic Editor: Changbum Chun

A new deblurring and denoising algorithm is proposed, for isotropic total variation-based image restoration. The algorithm consists of an efficient solver for the nonlinear system and an acceleration strategy for the outer iteration. For the nonlinear system, the split Bregman method is used to convert it into linear system, and an algebraic multigrid method is applied to solve the linearized system. For the outer iteration, we have conducted formal convergence analysis to determine an auxiliary linear term that significantly stabilizes and accelerates the outer iteration. Numerical experiments demonstrate that our algorithm for deblurring and denoising problems is efficient.

1. Introduction

The purpose of image restoration is to recover original image u from an observed data z (noisy and blurred image) from the relation

$$z = Ku + n, \tag{1}$$

where K is a known linear blurring operator and u is a Gaussian white noise. The well-known minimization problem for image denoising and deblurring based on isotropic total variation (TV) is proposed by Rudin et al. [1]:

$$\min_u \left(\alpha \int_\Omega \sqrt{|\nabla u|^2 + \beta} dxdy + \frac{1}{2} \|Ku - z\|_{L_2}^2 \right). \tag{2}$$

Here $\alpha > 0$ is the penalty parameter, and $\beta > 0$ is the diffusion regularization parameter and is typically small. The functional in (2) is strictly convex with a unique global minimizer. The paper [2] showed the well-posedness of problem (2) as $\beta \to 0^+$ and the existence of a unique global minimizer. The corresponding Euler-Lagrange equation for (2) is given by

$$-\alpha \nabla \cdot \left(\frac{\nabla u}{\sqrt{|\nabla u|^2 + \beta}} \right) + K^* (Ku - z) = 0, \tag{3}$$

where K^* is the adjoint operator of K with respect to standard L_2 inner product.

In addition to the isotropic models, the anisotropic models for a qualitative improvement at corners are shown in [3] with that a transfer of the discretization from the anisotropic model to the isotropic setting results in an improvement of rotational invariance. The anisotropic TV defined in [3] is

$$\mathrm{TV}_2 (u) = \int_\Omega \left(|\nabla_x u| + |\nabla_y u| \right) dxdy. \tag{4}$$

The existence and uniqueness of solutions of the anisotropic total variation flow are shown in [4]. The explicit time-marching schemes are applied in [1, 5]. Zuo and Lin [6] proposed a generalized accelerated proximal gradient (GAPG) approach for solving TV-based image restoration problems by replacing the Lipschitz constant with an appropriate positive-definite matrix, resulting in faster convergence, and the TV regularization can be either isotropic or anisotropic. The convergence rate of $O(k^2)$ is maintained by GAPG, where k is the number of iterations. The introduction of an anisotropy to TV regularization [7] indeed leads to improved denoising: the staircasing effect is reduced while at the same time the creation of artifacts is suppressed.

There are different algorithms to solve (3). Earlier works include the time-marching scheme [1], the affine scaling algorithm [8], the Newton's method with a continuation procedure on β [9], a fixed point iteration with multigrid method [10–12], the combination of fixed point iteration and preconditioned conjugate gradient which method is proposed in [13], a fast algorithm based on the discrete cosine transform [14, 15], a new modular solver [16], a proximity algorithm [17], and a nonlinear primal-dual method [18]. Though the corresponding linear system of the nonlinear primal-dual method is twice as large as the primal system, it is shown to be better conditioned. In [19], the combination of the algebraic multigrid (AMG) methods [20, 21], Krylov subspace acceleration, and extrapolation of initial data are successfully combined to give an efficient algorithm for the purely denoising problem. In [22], the authors proposed the split Bregman method to solve image denoising and compressed sensing problem rapidly. Chang et al. [23] added a linear matrix to speed up the computational process.

In this paper, we propose an efficient and rapid algorithm for minimization problem (2) rather than solving the Euler-Lagrange equation (3) directly which combines the split Bregman method, the algebraic multigrid method, and Krylov subspace acceleration. Our algorithm consists of two steps. One uses the split Bregman method to convert (2) into linear system in the outer iteration, and the other adopts an AMG method previously developed by one of the authors for other applications [23, 24] to solve a linear system in the inner iteration. Moreover, the outer iteration is accelerated by Krylov subspace extrapolation. One V-cycle per inner iteration is sufficient in all our simulations. For the outer iteration, we have developed a stabilizing technique adapted to the blur operator K. This is done by adding a linear term on both sides of the equation [23]. Motivated by that, a wider linear term different from that in [23] is considered here. This linear stabilizing term is obtained via formal convergence analysis and is expressed explicitly in terms of the mask of K. The inclusion of the linear stabilizing term plays a crucial role in our scheme. The outer iteration indeed may diverge without a proper linear stabilizing term (Table 6, Section 5).

The rest of the paper is organized as follows. In Section 2, we discuss the purely image denoising problem by describing briefly the AMG method and the split Bregman method for the isotropic TV model and the anisotropic TV model. In Section 3, we discuss the image denoising and deblurring problem and present the formal convergence analysis and the framework of our scheme. In Section 4, we give out explicit formulae of the linear stabilizing term, together with other implementation details. Numerical results are also compared among several algorithms in Section 5. We end this paper with a brief summary in Section 6.

2. The Algebraic Multigrid

In this section, we will show how to directly use AMG in solving (3). We first show some denotations to discretize (3) (cf. [19]). We partition the domain $\Omega = (0,1) \times (0,1)$ into $L \times L$ uniform cells with mesh size $h = 1/L$. The cell centers are

$$(x_l, y_k) = \left(\left(l - \frac{1}{2}\right)h, \left(k - \frac{1}{2}\right)h\right), \quad l, k = 1, \ldots, L. \tag{5}$$

Following the ideas of [12, 19], we discretize (3) by standard five-point finite difference scheme to get

$$\begin{aligned} &-\frac{\alpha}{h}\left(c_{l+1/2,k}\frac{u_{l+1,k} - u_{l,k}}{h} - c_{l-1/2,k}\frac{u_{l,k} - u_{l-1,k}}{h}\right) \\ &-\frac{\alpha}{h}\left(c_{l,k+1/2}\frac{u_{l,k+1} - u_{l,k}}{h} - c_{l,k-1/2}\frac{u_{l,k} - u_{l,k-1}}{h}\right) \\ &+\left(K^*\left(Ku - z\right)\right)_{l,k} = 0, \quad l, k = 1, \ldots, L, \end{aligned} \tag{6}$$

with homogeneous Neumann boundary condition:

$$\begin{aligned} u_{0,k} &= u_{1,k}, \qquad u_{L+1,k} = u_{L,k}, \\ u_{l,0} &= u_{l,1}, \qquad u_{l,L+1} = u_{l,L}, \end{aligned} \tag{7}$$

where

$$c_{l+1/2,k} = \frac{1}{\sqrt{\left(\left(D_x u\right)_{l+1/2,k}\right)^2 + \left(\left(D_y u\right)_{l+1/2,k}\right)^2 + \beta}}, \tag{8}$$

with

$$\begin{aligned} \left(D_x u\right)_{l+1/2,k} &= \frac{u_{l+1,k} - u_{l,k}}{h}, \\ \left(D_y u\right)_{l+1/2,k} &= \frac{1}{2}\left(\frac{u_{l+1,k+1} - u_{l+1,k-1}}{2h}\right. \\ &\qquad \left. + \frac{u_{l,k+1} - u_{l,k-1}}{2h}\right), \end{aligned} \tag{9}$$

and so forth.

To simplify the notation, we abbreviate (6) as

$$\alpha L(u)\, u = -K^*\left(Ku - z\right), \tag{10}$$

where

$$L(v)\, w = -\nabla_h \cdot \left(\frac{\nabla_h w}{\sqrt{\left|\nabla_h v\right|^2 + \beta}}\right), \tag{11}$$

which is fully nonlinear with wildly varying coefficient.

Consider the general system (10) in an $L^2 \times L^2$:

$$AU = F, \tag{12}$$

with the matrix $A = \alpha L(u)$ in (10). In general, A varies wildly near areas of high contrast of the image and need not be diagonally dominant. Nevertheless, A is symmetric and positive definite. Moreover, the matrix K^*K is wide-banded, and the spectra of the matrices $L(u)$ and K^*K are quite differently distributed. It is difficult to compute (10). In [13], Vogel and Oman adopted the combination of a fixed point iteration and a product PCG to handle the nonlinear term and the linear system, respectively. In [15], Chan et al.

proposed another preconditioner based on the fast cosine transform.

Algebraic multigrid method is considered in [19, 24, 25] as a linear solver. In contrast to geometric multigrid method, the construction of the coarse grids Ω^m is solely based on the algebraic information provided by the matrix A in an AMG setting. The details and effectiveness can be found, for example, in [23, 24, 26]. For the restriction and coarse grid operators, a simple and popular approach is the Galerkin type construction. Namely, $I_m^{m+1} = (I_{m+1}^m)^*$ and $A^{m+1} = I_m^{m+1} A^m I_{m+1}^m$. How to construct the interpolation operators I_{m+1}^m is the most important part in the method. The convergence proof for this improved AMG method was given in [25] when A^m is symmetric positive definite. The robustness of these interpolation formulae is supported by plenty of numerical evidence [24, 25]. The detailed algorithm using the AMG method to solve (10) is in Algorithm 1.

2.1. The Split Bregman Method for Isotropic TV Model. In [22], the split Bregman method is applied to the ROF model for image denoising problem. For completeness, we show the main progress about the split Bregman method applied to the ROF model. Firstly, consider discretizing ROF model into

$$\begin{aligned}\int_\Omega |\nabla u|\, dxdy &+ \frac{\mu}{2}\|Ku - z\|_{L_2}^2 \\ &\approx \sum_{i,j}\sqrt{(\nabla_x u)_{i,j}^2 + (\nabla_y u)_{i,j}^2} + \frac{\mu}{2}\|Ku - z\|_{L_2}^2.\end{aligned} \tag{13}$$

Setting $d_x \approx \nabla_x u$ and $d_y \approx \nabla_y u$, the split Bregman formulation of the isotropic problem then becomes

$$\begin{aligned}\min_{u,d_x,d_y} \left\|(d_x, d_y)\right\|_2 &+ \frac{\mu}{2}\|Ku - z\|_2^2 + \frac{\lambda}{2}\left\|d_x - \nabla_x u - b_x\right\|_2^2 \\ &+ \frac{\lambda}{2}\left\|d_y - \nabla_y u - b_y\right\|_2^2,\end{aligned} \tag{14}$$

where

$$\left\|(d_x, d_y)\right\|_2 = \sum_{i,j}\sqrt{d_{x,i,j}^2 + d_{y,i,j}^2}. \tag{15}$$

Consider it into the following three subproblems:

$$\begin{aligned}u^{k+1} &= \min_u \frac{\mu}{2}\|Ku - z\|_2^2 + \frac{\lambda}{2}\left\|d_x^k - \nabla_x u - b_x^k\right\|_2^2 \\ &\quad + \frac{\lambda}{2}\left\|d_y^k - \nabla_y u - b_y^k\right\|_2^2,\end{aligned} \tag{16}$$

$$\begin{aligned}\left(d_x^{k+1}, d_y^{k+1}\right) &= \min_{d_x,d_y}\left\|(d_x, d_y)\right\|_2 \\ &\quad + \frac{\lambda}{2}\left\|d_x^k - \nabla_x u^{k+1} - b_x^k\right\|_2^2 \\ &\quad + \frac{\lambda}{2}\left\|d_y^k - \nabla_y u^{k+1} - b_y^k\right\|_2^2,\end{aligned} \tag{17}$$

$$\begin{aligned}b_x^{k+1} &= b_x^k + \left(\nabla_x u^{k+1} - d_x^{k+1}\right), \\ b_y^{k+1} &= b_y^k + \left(\nabla_y u^{k+1} - d_y^{k+1}\right).\end{aligned} \tag{18}$$

The formula (16) is l_2-norm; it can be solved using the variation approach. The corresponding Euler-Lagrange equation for (16) is

$$\begin{aligned}(\mu K^* K - \lambda\Delta)\, u^{k+1} &= \mu K^* Z + \lambda\nabla_x^T\left(d_x^k - b_x^k\right) \\ &\quad + \lambda\nabla_y^T\left(d_y^k - b_y^k\right).\end{aligned} \tag{19}$$

The difference scheme is

$$\begin{aligned}&\mu\left(K^* K u^{k+1}\right)_{ij} + \lambda\left(4u_{ij}^{k+1} - u_{i+1,j}^{k+1} - u_{i-1,j}^{k+1} - u_{i,j+1}^{k+1} - u_{i,j-1}^{k+1}\right) \\ &\quad = F^k := \mu(K^* z)_{ij} + \lambda\left[\left(d_x^k - b_x^k\right)_{i-1,j} - \left(d_x^k - b_x^k\right)_{i,j}\right. \\ &\qquad \left. + \left(d_y^k - b_y^k\right)_{i,j-1} - \left(d_y^k - b_y^k\right)_{i,j}\right].\end{aligned} \tag{20}$$

When $K = I$, the difference scheme of the previous formula is simplified as

$$\begin{aligned}&(\mu + 4\lambda)\, u_{ij}^{k+1} - \lambda\left(u_{i+1,j}^{k+1} + u_{i-1,j}^{k+1} + u_{i,j+1}^{k+1} + u_{i,j-1}^{k+1}\right) \\ &\quad = \mu z_{ij} + \lambda\left(d_{x,i-1,j}^k - d_{x,i,j}^k + d_{y,i,j-1}^k - d_{y,i,j}^k - b_{x,i-1,j}^k\right. \\ &\qquad \left. + b_{x,i,j}^k - b_{y,i,j-1}^k + b_{y,i,j}^k\right).\end{aligned} \tag{21}$$

Using Gauss-Seidel method for (21), we have

$$\begin{aligned}u_{ij}^{k+1} &= G_{i,j}^k \\ &:= \frac{\lambda}{\mu + 4\lambda}\left(u_{i+1,j}^k + u_{i-1,j}^{k+1} + u_{i,j+1}^k + u_{i,j-1}^{k+1} + d_{x,i-1,j}^k\right. \\ &\qquad - d_{x,i,j}^k + d_{y,i,j-1}^k - d_{y,i,j}^k - b_{x,i-1,j}^k + b_{x,i,j}^k \\ &\qquad \left. - b_{y,i,j-1}^k + b_{y,i,j}^k\right) + \frac{\mu}{\mu + 4\lambda} z_{ij}.\end{aligned} \tag{22}$$

Problem (17) can be solved using the fast shrinkage operators:

$$\begin{aligned}d_x^{k+1} &= \operatorname{shrink}\left(\nabla_x u^k + b_x^k, \frac{1}{\lambda}\right) \\ &= \max\left(s^k - \frac{1}{\lambda}, 0\right)\cdot\frac{\nabla_x u^k + b_x^k}{s^k}, \\ d_y^{k+1} &= \operatorname{shrink}\left(\nabla_y u^k + b_y^k, \frac{1}{\lambda}\right) \\ &= \max\left(s^k - \frac{1}{\lambda}, 0\right)\cdot\frac{\nabla_y u^k + b_y^k}{s^k}, \\ s^k &= \sqrt{\left|\nabla_x u^k + b_x^k\right|^2 + \left|\nabla_y u^k + b_y^k\right|^2}.\end{aligned} \tag{23}$$

```
Initialize: $u^0 = z$
While $\frac{\|u^k - u^{k-1}\|_2}{\|u^k\|_2} > \epsilon$,
        Compute $u$ by $\alpha L(u^k)u^{k+1} = -K^*(Ku^k - z)$.
end
```

Algorithm 1

The optimal value of d can be explicitly gotten using the above shrinkage operators. The formula (18) can be discretized directly. Thus the split Bregman algorithm of the isotropic TV denoising is as in Algorithm 2.

2.2. The Split Bregman Method Based on AMG for Isotropic TV Model. It is easy to find that the accuracy of the split Bregman method is not high because the Gauss-Seidel iteration of u is executed only once per loop. At the same time, the AMG algorithm has fast convergence and high precision. It can be applied to large signal-to-noise ratio of image denoising, which has strong stability. So, we consider the combination of the two methods, and get a new denoising algorithm which is the split Bregman method based on AMG (here we call it as Algorithm 3). This algorithm contains four steps.

Moreover, we use Krylov technique to accelerate subspace extrapolation.

2.3. The Split Bregman Method for Anisotropic TV Model. The anisotropic total variation of u [27] is represented by

$$\left\|\nabla_x u\right\|_1 + \left\|\nabla_y u\right\|_1. \tag{24}$$

The anisotropic TV model for denoising can be formulated as the following minimization problem:

$$\min_{u,d_x,d_y} \left\|\nabla_x u\right\|_1 + \left\|\nabla_y u\right\|_1 + \frac{\mu}{2}\|u - z\|_2^2, \tag{25}$$

where μ is an appropriately chosen positive parameter. Define an operator [27]

$$\begin{aligned} \operatorname{cut}\left(c, \frac{1}{\lambda}\right) &= c - \max\left(|c| - \frac{1}{\lambda}, 0\right) \cdot \frac{c}{|c|} \\ &= \begin{cases} \frac{1}{\lambda}, & c > \frac{1}{\lambda}, \\ c, & -\frac{1}{\lambda} \le c \le \frac{1}{\lambda}, \\ -\frac{1}{\lambda}, & c < -\frac{1}{\lambda}. \end{cases} \end{aligned} \tag{26}$$

With the previous operator, we have Algorithm 4 to solve the anisotropic TV denoising problem.

It should be noticed that there is no need to solve any equations like the isotropic model. This algorithm need not solve any elliptic equation of u in [27], so the procedure is rapid, simple, and not costly.

3. Convergence Analysis

We consider the discretized system with general noise and blur:

$$\begin{aligned} &\mu\left(K^* K u^{k+1}\right)_{ij} + \lambda\left(4u_{ij}^{k+1} - u_{i+1,j}^{k+1} - u_{i-1,j}^{k+1} - u_{i,j+1}^{k+1} - u_{i,j-1}^{k+1}\right) \\ &\quad = \mu(K^* z)_{ij} + \lambda\left[\left(d_x^k - b_x^k\right)_{i-1,j} - \left(d_x^k - b_x^k\right)_{i,j}\right. \\ &\qquad \left. + \left(d_y^k - b_y^k\right)_{i,j-1} - \left(d_y^k - b_y^k\right)_{i,j}\right] = F_{ij}. \end{aligned} \tag{27}$$

To simplify the notation, we abbreviate (27) as

$$-\lambda \Delta u = -\mu K^* K u + F. \tag{28}$$

In general, the matrix $K^* K$ would be dense. There is one way to completely avoid the matrix inversion. A natural approach would then be

$$-\lambda \Delta u^{(s+1)} = -\mu K^* K u^{(s)} + F. \tag{29}$$

It turns out that the iteration (29) is not robust and may diverge even for weak blur operator corresponding to the mask $(1/64)(1,1,4,1,1)^T(1,1,4,1,1)$ with $\sigma = 10$. To overcome this instability, we propose to add a linear term $B + \eta I$ on both sides of (29) and get

$$\left(-\lambda\Delta + B + \eta I\right) u^{(s+1)} = -\left(\mu K^* K u^{(s)} - B - \eta I\right) u^{(s)} + F, \tag{30}$$

where η is arbitrary small positive number and B is a symmetric and positive definite matrix to be determined through the following formal analysis. Now we will analyze the convergence property of (29).

Theorem 1. *If the matrix B satisfies*

$$2B - \mu K^* K \ge 0, \tag{31}$$

the equation (30) *is locally convergent.*

Let u^{∞} be the solution of (29); that is,

$$-\lambda \Delta u^{(\infty)} = -\mu K^* K u^{(\infty)} + F. \tag{32}$$

We then define $e^{(\infty)} = u^{(s)} - u^{(\infty)}$, and the error equation is

$$\left(-\lambda\Delta + B + \eta I\right) e^{(s+1)} = -\left(\mu K^* K u^{(s)} - B - \eta I\right) e^{(s)}, \tag{33}$$

```
Initialize: $u^0 = z, d_x^0 = 0, d_y^0 = 0, b_x^0 = 0, b_y^0 = 0$.
While $\frac{\|u^k - u^{k-1}\|_2}{\|u^k\|_2} > \epsilon$,
        $u^{k+1} = G^k$ by (22),
        $d_x^{k+1} = \max(s^k - \frac{1}{\lambda}, 0) \cdot \frac{\nabla_x u^{k+1} + b_x^k}{s^k}$,
        $d_y^{k+1} = \max(s^k - \frac{1}{\lambda}, 0) \cdot \frac{\nabla_y u^{k+1} + b_y^k}{s^k}$,
        $b_x^{k+1} = b_x^k + (\nabla_x u^{k+1} - d_x^{k+1})$,
        $b_y^{k+1} = b_y^k + (\nabla_y u^{k+1} - d_y^{k+1})$.
End
```

Algorithm 2

```
Step 1: Decomposing the denoising model of ROF to three subproblems (16)–(18) using the split
        Bregman method.
Step 2: To get $u$ by solving (22) using the AMG method.
Step 3: To get $d_x, d_y$, using the shrinkage operator (23).
Step 4: To directly discretize $b_x, b_y$, using difference scheme.
```

Algorithm 3

which is equivalent to

$$\eta\left(e^{(s+1)} - e^{(s)}\right) = -\left(Pe^{(s+1)} - Qe^{(s)}\right) \tag{34}$$

with

$$P = (-\lambda\Delta + B), \qquad Q = \left(B - \mu K^* K\right). \tag{35}$$

It is easy to show that the matrices P and Q are symmetric. Taking the inner product on both sides of (34) with $Pe^{(s+1)} - Qe^{(s)}$ gives

$$\eta\left(e^{(s+1)} - e^{(s)}, Pe^{(s+1)} - Qe^{(s)}\right) = -\left\|Pe^{(s+1)} - Qe^{(s)}\right\|_2^2 \leq 0, \tag{36}$$

and we will now rearrange the formula (36) as

$$\eta\left(e^{(s+1)} - e^{(s)}\right)^T \times \left(\frac{P-Q}{2}\left(e^{(s+1)} + e^{(s)}\right) + \frac{P+Q}{2}\left(e^{(s+1)} - e^{(s)}\right)\right) \leq 0. \tag{37}$$

Rewrite the previous formula as

$$\begin{aligned}&\left(e^{(s+1)}\right)^T (P-Q)\, e^{(s+1)} - \left(e^{(s)}\right)^T (P-Q)\, e^{(s)} \\ &\quad + \left(e^{(s+1)} - e^{(s)}\right)^T (P+Q)\left(e^{(s+1)} - e^{(s)}\right) \leq 0.\end{aligned} \tag{38}$$

If $(P-Q)$ and $(P+Q)$ are symmetric and positive definite, it follows from (38) that

$$\left\|\left(e^{(s+1)}\right)\right\|_{P-Q}^2 - \left\|\left(e^{(s)}\right)\right\|_{P-Q}^2 + \left\|e^{(s+1)} - e^{(s)}\right\|_{P+Q}^2 \leq 0, \tag{39}$$

where $\|v\|_A^2 = v^T A v$ with positive definite matrix A. The behavior of (39) leads to

$$\left\|\left(e^{(s+1)}\right)\right\|_{P-Q}^2 \leq \left\|\left(e^{(s)}\right)\right\|_{P-Q}^2. \tag{40}$$

In other words, the iteration (30) is locally convergent if both

$$\begin{aligned}P - Q &= -\lambda\Delta + \mu K^* K, \\ P + Q &= -\lambda\Delta + 2B - \mu K^* K\end{aligned} \tag{41}$$

are symmetric and positive definite which means

$$\begin{aligned}P - Q &= -\lambda\Delta + \mu K^* K > 0, \\ P + Q &= -\lambda\Delta + 2B - \mu K^* K > 0.\end{aligned} \tag{42}$$

Since $-\lambda\Delta$ and $\mu K^* K$ are symmetric and positive definite, a sufficient condition for (42) is given by (31).

Remark. We can see that (31) indeed is a good guideline for devising the iteration of u. We will elaborate on the choice of B better suited for numerical purposes in Section 4. Indeed, the iteration (30) may fail to converge when (31) is not satisfied. See Section 4.2 for more details. In addition, the preprocessing of initial data proposed in [19] is no longer required when (31) is satisfied.

4. Numerical Experiments and Discussion

4.1. Images and Blurring Operators. The numerical examples given in this paper are based on the three images shown in Figure 1. The first one is a satellite image (Image I), the

```
Initialize: $u^1 = z, b_x^0 = 0, b_y^0 = 0.$
While $\frac{\left\|u^k - u^{k-1}\right\|_2}{\left\|u^k\right\|_2} < \epsilon,$
        $b_x^k = \mathrm{cut}(\nabla_x u^k + b_x^{k-1}, \frac{1}{\lambda}),$
        $b_y^k = \mathrm{cut}(\nabla_y u^k + b_y^{k-1}, \frac{1}{\lambda}),$
        $u^{k+1} = z - \frac{\lambda}{\mu}(\nabla_x^T b^k + \nabla_y^T b^k).$
End
```

Algorithm 4

second one is made up of simple shapes (Image II) used in the literature [15], and the last one is Lena image (Image III). We have experiments on restoring the three images blurred by the following three different blurring operators [23].

Blur operator I:

$$\frac{1.5}{784}(1,2,3,16,3,2,1)^T(1,2,3,16,3,2,1) = \frac{1.5}{784}\begin{bmatrix} 1 & 2 & 3 & 16 & 3 & 2 & 1 \\ 2 & 4 & 6 & 32 & 6 & 4 & 2 \\ 3 & 6 & 9 & 48 & 9 & 6 & 3 \\ 16 & 32 & 48 & 256 & 48 & 32 & 16 \\ 3 & 6 & 9 & 48 & 9 & 6 & 3 \\ 2 & 4 & 6 & 32 & 6 & 4 & 2 \\ 1 & 2 & 3 & 16 & 3 & 2 & 1 \end{bmatrix}. \qquad (43)$$

Blur operator II: an out-of-focus blur with the kernel of convolution given by the kernel

$$v_{k,l} = \begin{cases} \frac{c}{13}, & \text{if } k^2 + l^2 \le 4, \\ 0, & \text{otherwise,} \end{cases} \qquad (44)$$

where $c = 1.2188$.

Blur operator III: a truncated Gaussian blur given by

$$v_{k,l} = \begin{cases} ce^{-\tau(k^2+l^2)} & \text{if } |k|, |l| \le 5, \\ 0, & \text{otherwise,} \end{cases} \qquad (45)$$

where $c = 0.176$ and $\tau = 0.38$.

4.2. The Linear Stabilizing Term. The condition (31) serves as a general guideline for the choice of the matrix. In this section, we will consider a quinta-diagonal matrix B to be compared with a diagonal matrix B [23].

4.2.1. Diagonal Matrix B. In order to solve (30) efficiently, an obvious candidate is diagonal matrices of the form

$$B = \varepsilon I. \qquad (46)$$

The condition (31) then reads

$$\varepsilon \ge \varepsilon^* = \frac{1}{2}\lambda_{\max}\left(\mu K^* K\right) = \frac{1}{2}\lambda_{\max}\left(\mu K^2\right). \qquad (47)$$

To proceed with the estimate of $\lambda_{\max}(K)$, we notice that our sample blurring operators I–III can be represented by a mask of the form

$$V = \begin{bmatrix} v_{-r,-r} & v_{-r,-r+1} & \cdots & v_{-r,r} \\ v_{-r+1,-r} & v_{-r+1,-r+1} & \cdots & v_{-r+1,r} \\ \vdots & \vdots & \vdots & \vdots \\ v_{r,-r} & v_{r,-r+1} & \cdots & v_{r,r} \end{bmatrix}_{(2r+1)*(2r+1)}, \qquad (48)$$

with $v_{k,l} = v_{l,k} = v_{-k,l} = v_{k,-l}$. It is easy to see that, under the Neumann boundary conditions, the largest eigenvalue for such K is given by

$$\lambda_{\max}(K) = \sum_{i=1}^{L^2} K_{i,j} = \sum_{k,l=-r}^{r} v_{k,l}, \qquad (49)$$

where the second term of (49) is independent and the corresponding eigenfunction is a constant. In summary, we have the following sufficient condition for (31):

$$B = \varepsilon I, \qquad \varepsilon \ge \varepsilon^* = \frac{1}{2}\mu\left(\sum_{k,l=-r}^{r} v_{k,l}\right)^2. \qquad (50)$$

Similar conditions can be derived for variants of (50), such as

$$\begin{aligned} B &= \frac{1}{2}\operatorname{diag}\left(K^* K\right) + \frac{1}{2}\delta I, \\ \delta \ge \delta^* &= \max_i \sum_{j \ne i} \mu\left(K^* K\right)_{i,j} \\ &= \mu\left(\sum_{k,l=-r}^{r} v_{k,l}\right)^2 - \sum_{k,l=-r}^{r} v_{k,l}^2, \\ \delta \ge \delta^* &= \max_i \left(\sum_{j \ne i} \mu\left(K^* K\right)_{i,j} - \left(K^* K\right)_{i,i}\right) \\ &= \mu\left(\sum_{k,l=-r}^{r} v_{k,l}\right)^2 - \sum_{k,l=-r}^{r} v_{k,l}^2, \end{aligned} \qquad (51)$$

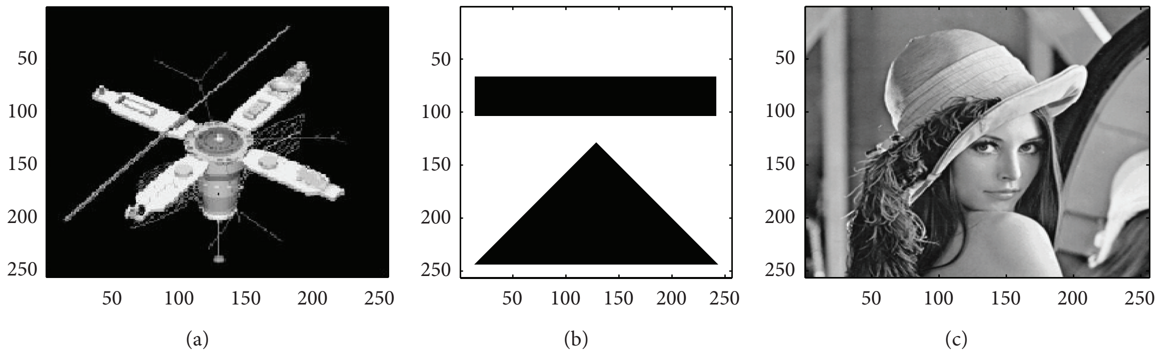


Figure 1: Original images. Image I, Image II, and Image III.

Table 1: Critical values of the parameters, $\varepsilon^*, \delta^*, \gamma^*$ and the solution to (55).

Blur operator	b^*	δ^*	γ^*	$(a,b)^*$
I	1.13	1.95	0.83	(0.72, 0.17)
II	0.74	1.37	0.63	(0.49, 0.09)
III	1.06	1.98	0.93	(0.65, 0.15)

or

$$B = \operatorname{diag}\left(K^*K\right) + \gamma I,$$

$$\gamma \geq \gamma^* = \frac{1}{2}\max_i \left(\sum_{j \neq i} \mu \left(K^*K\right)_{i,j} - 2\left(K^*K\right)_{i,i} \right) = \frac{1}{2}\mu \left(\sum_{k,l=-r}^{r} v_{k,l} \right)^2 - \sum_{k,l=-r}^{r} v_{k,l}^2. \quad (52)$$

In our examples, the diagonal entries of K^*K are constant-valued except for the near-boundary pixels. We expect the performances of (50), (51), and (52) to be comparable to each other. In this paper, most of our numerical experiments are conducted using (52) with $\gamma = 1$. See Section 5 for more details.

4.2.2. Quinta-Diagonal Matrix B. In addition to diagonal matrices, we have also explored quinta-diagonal matrices $B_{a,b}$ given by the mask

$$\begin{bmatrix} 0 & b & 0 \\ b & a & b \\ 0 & b & 0 \end{bmatrix}. \quad (53)$$

$B_{a,b}$ has the same support as laplace operator Δ; therefore the computational cost in the AMG step is comparable to diagonal B. In view of (33) and (31), it is clear that the optimal $B_{(a,b)^*}$ is the one that solves

$$\min_{a,b} \rho \left(\left(-\lambda\Delta + B_{(a,b)} \right)^{-1} \left(B_{(a,b)} - \mu K^*K \right) \right) \quad \text{subject to } 2B_{(a,b)} - \mu K^*K \geq 0, \quad (54)$$

Table 2: Results of Figure 2.

Algorithm	(σ, μ, λ)	Iteration step	$e2$	$e0$	Time
Algorithm 2	(10, 0.1, 0.2)	$N = 3$	6.88	87.36	0.0625
		$N = 50$	1.16	30.64	0.9218
	(40, 0.05, 0.1)	$N = 3$	11.98	93.46	0.0468
		$N = 50$	10.04	106.16	0.6562
Algorithm 4	(10, 0.1, 0.2)	$N = 50$	4.88	46.25	0.5156
	(40, 0.1, 0.2)	$N = 50$	19.24	156.89	0.5156
Algorithm 3	(10, 0.2, 0.2)	$N = 3$	4.40	73.23	0.3125
	(40, 0.03, 0.06)	$N = 3$	9.29	111.18	0.3125

where $\rho(A)$ is the spectral radius of A. The actual minimizer of (54) depends on the true image $u^{(\infty)}$ and there is no simple way of finding it. An accessible approximation is given by the following modified minimization problem:

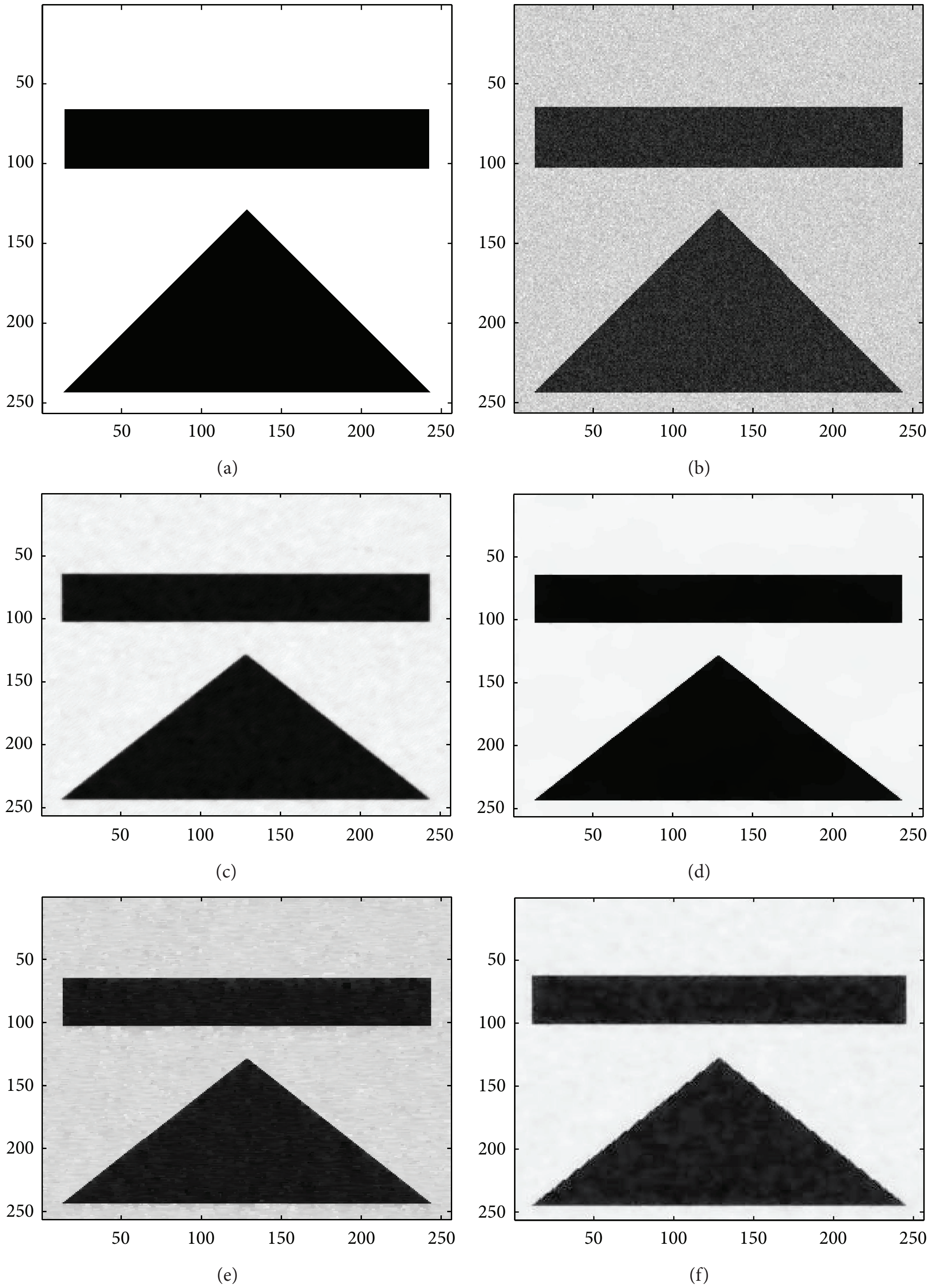
$$\min_{a,b} \rho \left(B_{(a,b)} - \mu K^*K \right) \quad \text{subject to } 2B_{(a,b)} - \mu K^*K \geq 0. \quad (55)$$

In our examples, the common eigenbasis of $B_{a,b}$ and K is given by

$$e_{m,n}\left(x_k, y_l\right) = \cos\left(m\pi x_k\right)\cos\left(n\pi y_l\right), \quad m, n = 1, \ldots, L. \quad (56)$$

It is straightforward to compute the corresponding eigenvalues for $(B_{(a,b)} - \mu K^*K)$ and the numerical solution of (55) by varying a and b over a suitable range. The optimal $(a,b)^*$, together with the critical values of ε, δ, and γ, is given in Table 1. It is not clear a priori why (55) should result in better performance. Nevertheless, we find from our experiment that $B_{(a,b)^*}$ indeed performs no worse than diagonal B and can be significantly faster in some cases. See more details in Section 5.

4.3. The Stopping Criterion and Acceleration Technique. The outer iteration is stopped upon a relative decrease of the



(a)

(b)

(c)

(d)

(e)

(f)

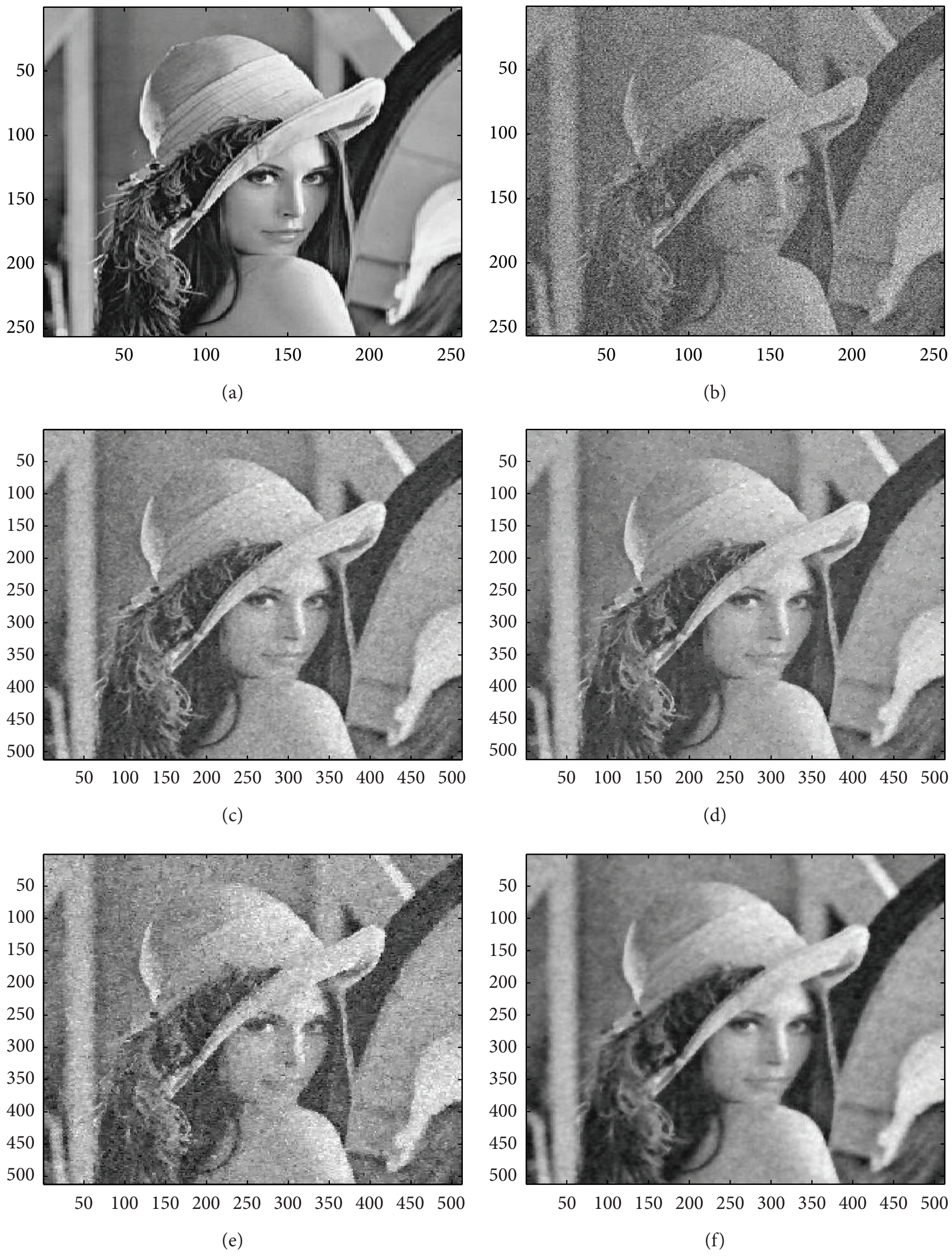
Figure 2: (a) Original image. (b) Noisy image contaminated by $\sigma = 10$ (SNR =14.90%). (c) Denoised image with 3 iterations of Algorithm 2. (d) Denoised image with 50 iterations of Algorithm 2. (e) Denoised image with 50 iterations of Algorithm 4. (f) Denoised image with 3 iterations of Algorithm 3.

(normalized) residual by a factor of 10^{-4} for the blurring operators, namely, when the condition

$$\frac{\left\|r^{(N)}\right\|_{l_2}}{\left\|r^{(1)}\right\|_{l_2}} \leq 10^{-4} \tag{57}$$

is reached. Here the normalization of the residual is given by

$$\begin{aligned} r^{(s+1)} = {} & \left(D^{(s+1)}\right)^{-1} \\ & \times \left(\left(-\lambda\Delta + B\right) v^{(s+1)} + \left(\mu K^{*} K - B\right) v^{(s+1)} - F\right), \end{aligned} \tag{58}$$



(a)

(b)

(c)

(d)

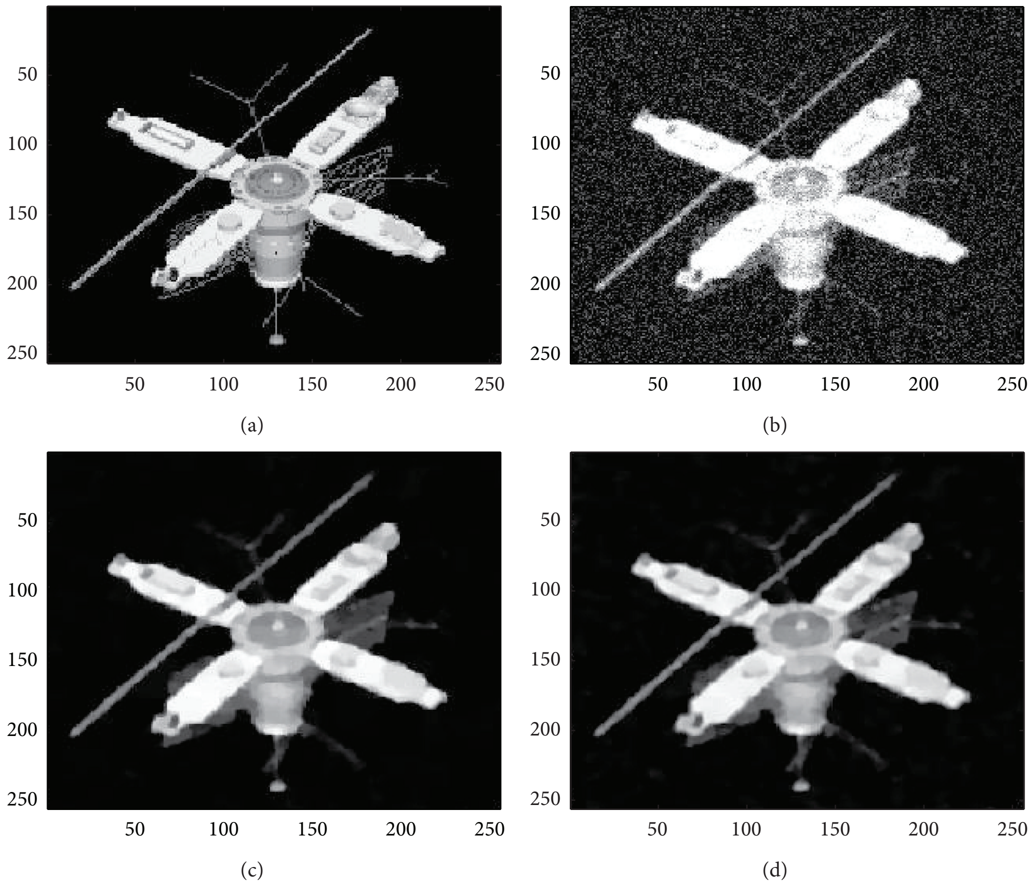
(e)

(f)

Figure 3: (a) Original image. (b) Noisy image contaminated by $\sigma = 40$ (SNR = 26.725%). (c) Denoised image with 3 iterations of Algorithm 2. (d) Denoised image with 50 iterations of Algorithm 2. (e) Denoised image with 50 iterations of Algorithm 4. (f) Denoised image with 3 iterations of Algorithm 3.

where $v^{(s+1)}$ is an approximate solution of (28) obtained with one V-cycle iteration (see also Section 5) and $D^{(s+1)} = \text{diag}(-\lambda\Delta + B)$. Note that the larger entries of the normalizing factor $(D^{(s+1)})^{-1}$ correspond to regions where $v^{(s+1)}$ is less smooth. It amplifies the (unnormalized) residual on regions where $v^{(s+1)}$ either has a jump or requires further denoising, therefore doing a better job measuring the image quality. Numerical experiments confirmed that the normalized residual is indeed a good indicator for the quality of denoising and deblurring.

To accelerate and stabilize the outer iteration, we have incorporated the Krylov subspace method [19, 28, 29] in our implementation. The approximate solution is optimized every p steps on a subspace of dimension $M \leq p$. To be more



(a) (b) (c) (d)

Figure 4: (a) Original image. (b) Blurred and noisy image. (c) Denoised and deblurred image with 20 iterations of Algorithm 1. (d) Denoised and deblurred image with 5 iterations of Algorithm 3.

Table 3: Results of Figure 3.

Algorithm	(σ, μ, λ)	Iteration step	$e2$	$e0$	Time
Algorithm 2	$(10, 0.1, 0.2)$	$N = 3$	5.94	133.24	0.2500
		$N = 50$	5.96	139.42	3.7187
	$(40, 0.05, 0.1)$	$N = 3$	12.04	146.55	0.1875
		$N = 50$	12.12	147.62	2.750
Algorithm 4	$(10, 0.1, 0.2)$	$N = 50$	8.27	56.69	2.0625
	$(40, 0.03, 0.00375)$	$N = 50$	19.71	139.02	2.0781
Algorithm 3	$(10, 0.2, 0.2)$	$N = 2$	5.64	112.07	1.156
	$(40, 0.03, 0.06)$	$N = 2$	10.51	109.00	1.156

precise, we take two fixed integers $0 \le M \le p$, and for any integer $n > 0$, let

$$\tilde{u}\left(c_1, \ldots, c_M\right) = u^{(4n)} + \sum_{m=1}^{M} c_m \left(u^{(4n+1-m)} - u^{(4n-m)}\right). \quad (59)$$

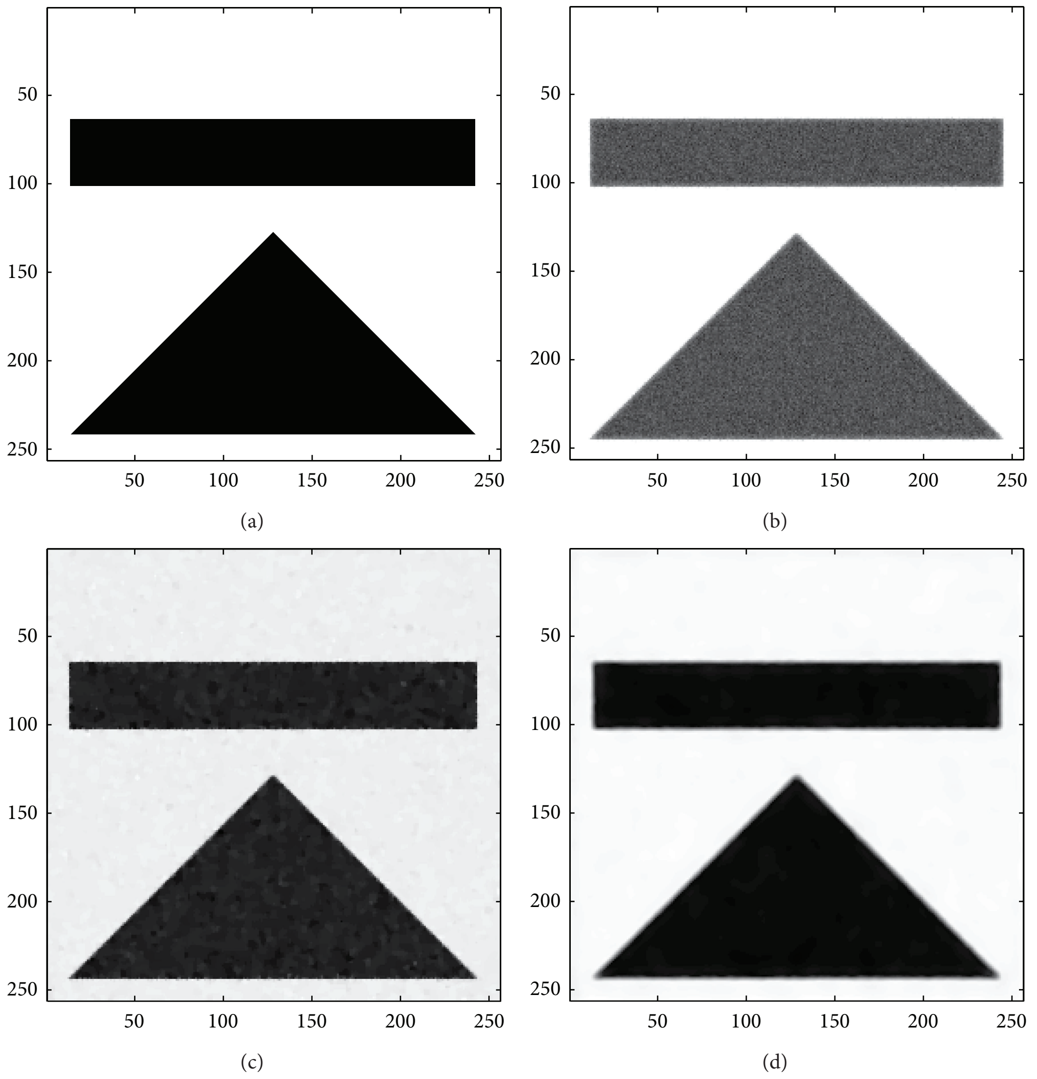
The residual of $\tilde{u}(c_1, \ldots, c_M)$ can be approximated by

$$\tilde{r}\left(c_1, \ldots, c_M\right) \overset{\text{def}}{=} r^{(4n)} + \sum_{m=1}^{M} c_m \left(r^{(4n+1-m)} - r^{(4n-m)}\right). \quad (60)$$

One can minimize $\tilde{r}(c_1, \ldots, c_M)$ with respect to $(c_1, \ldots, c_M)$ to get

$$\min_{c_1,\ldots,c_M} \left\|\tilde{r}\left(c_1, \ldots, c_M\right)\right\|_{l^2} = \left\|\tilde{r}\left(c_1^*, \ldots, c_M^*\right)\right\|_{l^2} \quad (61)$$

and reset $u^{(pn)}$ to $\tilde{u}(c_1^*, \ldots, c_M^*)$ before going to $(pn+1)$ the outer iteration. The numerical results demonstrate that the Krylov acceleration method is an efficient way to accelerate the outer iteration. See Section 5.



(a)

(b)

(c)

(d)

Figure 5: (a) Original image. (b) Blurred and noisy image. (c) Denoised and deblurred image with 10 iterations of Algorithm 1. (d) Denoised and deblurred image with 5 iterations of Algorithm 3.

5. Numerical Results

To generate an observed image z, a Gaussian white noise with mean 0 and variance σ is added to the blurred image. The level of noise can be measured by the signal-to-noise ratio (SNR):

$$\mathrm{SNR} = \frac{\|Ku - z\|_{l^2}}{\|u\|_{l^2}}. \tag{62}$$

The signal-to-blur ratio (SBR)

$$\mathrm{SBR} = \frac{\|Ku - u\|_{l^2}}{\|u\|_{l^2}} \tag{63}$$

measures the strength of the blurring operator K. The two kinds of error $e2 = \|u^{k+1} - u^k\|_{L^2}$ and $e0 = \|u^{k+1} - u^k\|_{L^\infty}$ are used to judge the iterative error as the iteration ends. The cost of the iterative process is denoted by "time." These indexes are captioned in each example for readers' convenience.

In all our examples, the iteration begins with the observed image $u^{(0)} = z$ and all images are contaminated with small noise $\sigma = 10$ and large noise $\sigma = 40$. We choose B according to (52) with $\gamma = 1.0$ except in Table 6 where we demonstrate the necessity of the linear stabilizing term and the overall efficiency of our scheme by varying γ. It is enough to apply one single V-cycle in the AMG step with the Gauss-Seidel iteration as the smoother. We apply the Krylov acceleration procedure every 4 steps with the Krylov dimension $M = 2$. The rate of convergence of the outer iteration can be significantly improved by the Krylov subspace technique.

5.1. Image Denoising. Table 2 shows the contrast results of Image II with the split Bregman method for isotropic TV model (Algorithm 2), the split Bregman method based on AMG (Algorithm 3), and the split Bregman method for anisotropic TV model (Algorithm 4). Figure 2 shows the denoising image for Image II with noise contamination of

Table 4: Results of Figure 4.

Algorithm	$(\sigma, \text{ite}, \alpha)/(\sigma, \text{ite}, \mu, \lambda)$	Blur operator	$e0$	$e2$	Time
Algorithm 1	(10, 10, 0.02)	I	93.91	8.47	5.8906
		II	155.95	12.22	5.1250
		III	149.45	12.09	8.2500
	(40, 20, 0.15)	I	167.84	14.67	7.2500
		II	167.72	15.61	6.8906
		III	168.29	15.08	11.1093
Algorithm 3	(10, 5, 0.04, 0.08)	I	161.52	13.34	1.7500
		II	172.19	15.61	1.2812
		III	172.62	14.96	2.9062
	(40, 5, 0.04, 0.08)	I	151.96	15.17	1.7343
		II	170.75	16.24	1.2968
		III	170.94	15.62	2.9531

Table 5: Results of Figure 5.

Algorithm	$(\sigma, \text{ite}, \alpha)/(\sigma, \text{ite}, \mu, \lambda)$	Blur operator	$e0$	$e2$	Time
Algorithm 1	(10, 10, 0.05)	I	82.67	3.30	5.4375
		II	104.79	4.36	5.2187
		III	104.68	4.99	8.3437
	(40, 20, 1.0)	I	122.00	5.86	5.4843
		II	126.02	6.99	6.9375
		III	123.50	6.61	11.0468
Algorithm 3	(10, 5, 0.04, 0.08)	I	100.84	6.51	1.7500
		II	113.03	8.29	1.2812
		III	111.03	8.01	3.0156
	(40, 5, 0.04, 0.08)	I	109.85	9.79	1.7656
		II	117.62	9.40	1.2812
		III	117.85	9.09	2.9375

$\sigma = 10$ (SNR = 14.90%). Table 3 shows the contrast results of Image III containing $512 * 512$ pixels. Figure 3 shows the denoising image for Image III ($512 * 512$) with noise contamination $\sigma = 40$ (SNR = 26.725%). It is clear that our algorithm (Algorithm 3) is efficient and suitable for elliptic equation.

5.2. Image Denoising and Deblurring. The compared results are given in Tables 4-5. Figure 4 shows the denoising and deblurring images for Image I contaminated by noise with $\sigma = 40$ and blurring operator III (SNR = 51.34%, SBR = 45.31%). Figure 5 shows the denoising and deblurring images for Image II contaminated by noise with $\sigma = 10$ and blurring operator I (SNR = 5.96%, SBR = 48.79%).

We continue our numerical experiments by varying the parameter γ and compare their performance. The number of iterations of a typical example is shown in Table 6. The combination of $\gamma \approx \gamma^*$ and $M = 2$ gives the optimal result among possible choices of γ and M in general. The number of iterations is insensitive to the variation of γ near γ^* and gradually grows as γ increases. For simplicity of presentation, we take $\gamma = 1$ and $M = 2$ in all other numerical examples. We have also included the results for $B = B_{(a,b)^*}$ in Table 6 for comparison. In general, $B = B_{(a,b)^*}$ performs no worse than (52) with $\gamma = \gamma^*$. In some cases, $B = B_{(a,b)^*}$ can be up to 25% faster. In addition, if the blur operator is mild, our method can be used to solve $Ku = z$ directly.

According to the previous figures and tables, we get some conclusions here. First of all, in the second part of (42), we have neglected the $-\lambda\Delta$ term to obtain a sufficient condition (31), from which we further derived the critical value γ^* in (52). We therefore expect γ^* to be slightly overestimated. In other words, the outer iteration should converge for $\gamma \geq \gamma^*$, but not necessarily for $\gamma < \gamma^*$. This is in accordance with our numerical result. See the row with $M = 0$ in Table 6. Secondly, the Krylov subspace technique helps to stabilize the outer iteration for $\gamma < \gamma^*$, but eventually fails if γ becomes too small. This result demonstrates the necessity of the linear stabilizing term.

6. Conclusion

In this paper, we propose a new algorithm by adding a linear term for the total variation-based image denoising and deblurring which combine the split Bregman method, the algebraic multigrid method, and Krylov subspace acceleration. Through formal convergence analysis, we derived an explicit formula for a linear stabilizing term. Numerical

Table 6: Number of outer nonlinear iterations N for various γ. Here $\alpha = 0.05$, $\sigma = 29.72$ for Image I and $\alpha = 2.5$, $\sigma = 89.16$ for Image II.

Blur	Image	M	0.2	0.4	0.6	0.8	0.9	1.2	2	4	10	B_{a^*,b^*}
III	I	0	∞	∞	∞	∞	53	60	81	136	297	46
		1	∞	∞	148	50	48	28	32	48	93	23
		2	∞	∞	37	28	28	27	29	41	78	22
	II	0	∞	∞	109	109	109	113	119	131	162	109
		1	∞	39	37	36	36	37	40	44	65	36
		2	∞	34	32	32	33	33	35	42	57	32

experiments demonstrate that our algorithm is efficient and robust for deblurring and denoising problems.

Acknowledgments

The authors thank the referees for very useful remarks and suggestions. The research is supported by NSFC (no. 10801049 and no. 11271126) and Foundation of North China Electric Power University.

References

[1] L. I. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” *Physica D*, vol. 60, no. 1–4, pp. 259–268, 1992.

[2] R. Acar and C. R. Vogel, “Analysis of bounded variation penalty methods for ill-posed problems,” *Inverse Problems*, vol. 10, no. 6, pp. 1217–1229, 1994.

[3] H. Birkholz, “A unifying approach to isotropic and anisotropic total variation denoising models,” *Journal of Computational and Applied Mathematics*, vol. 235, no. 8, pp. 2502–2514, 2011.

[4] J. S. Moll, “The anisotropic total variation flow,” *Mathematische Annalen*, vol. 332, no. 1, pp. 177–218, 2005.

[5] C. A. Z. Barcelos and Y. Chen, “Heat flows and related minimization problem in image restoration,” *Computers & Mathematics with Applications*, vol. 39, no. 5-6, pp. 81–97, 2000.

[6] W. Zuo and Z. Lin, “A generalized accelerated proximal gradient approach for total-variation-based image restoration,” *IEEE Transactions on Image Processing*, vol. 20, no. 10, pp. 2748–2759, 2011.

[7] M. Grasmair and F. Lenzen, “Anisotropic total variation filtering,” *Applied Mathematics and Optimization*, vol. 62, no. 3, pp. 323–339, 2010.

[8] Y. Li and F. Santosa, “An affine scaling algorithm for minimizing total variation in image enhancement,” Tech. Rep. 12/94, Center for Theory and Simulation in Science and Engineering, Cornell University, 1994.

[9] R. Chan, T. Chan, and H. Zhou, “Advanced signal processing algorithms,” in *Proceedings of the International Society of Photo-Optical Instrumentation Engineers*, F. Luk, Ed., pp. 314–325, SPIE, 1995.

[10] M. E. Oman, “Fast multigrid techniques in total variation-based image reconstruction,” in *Proceedings of the Copper Mountain Conference on Multigrid Methods*, 1995.

[11] C. R. Vogel, “A multigrid method for total variation-based image denoising,” in *Computation and Control, IV*, vol. 20 of *Progress in Systems and Control Theory*, pp. 323–331, Birkhäuser, Boston, Mass, USA, 1995.

[12] C. R. Vogel and M. E. Oman, “Iterative methods for total variation denoising,” *SIAM Journal on Scientific Computing*, vol. 17, no. 1, pp. 227–238, 1996.

[13] C. R. Vogel and M. E. Oman, “Fast, robust total variation-based reconstruction of noisy, blurred images,” *IEEE Transactions on Image Processing*, vol. 7, no. 6, pp. 813–824, 1998.

[14] M. K. Ng, R. H. Chan, and W.-C. Tang, “A fast algorithm for deblurring models with Neumann boundary conditions,” *SIAM Journal on Scientific Computing*, vol. 21, no. 3, pp. 851–866, 1999.

[15] R. H. Chan, T. F. Chan, and C. K. Wong, “Cosine transform based preconditioned for total variation deblurring,” *IEEE Transactions on Image Processing*, vol. 8, no. 10, pp. 1472–1478, 1999.

[16] P. Blomgren and T. F. Chan, “Modular solvers for image restoration problems using the discrepancy principle,” *Numerical Linear Algebra with Applications*, vol. 9, no. 5, pp. 347–358, 2002.

[17] C. A. Micchelli, L. Shen, and Y. Xu, “Proximity algorithms for image models: denoising,” *Inverse Problems*, vol. 27, no. 4, Article ID 045009, 2011.

[18] T. F. Chan, G. H. Golub, and P. Mulet, “A nonlinear primal-dual method for total variation-based image restoration,” *SIAM Journal on Scientific Computing*, vol. 20, no. 6, pp. 1964–1977, 1999.

[19] Q. Chang and I.-L. Chern, “Acceleration methods for total variation-based image denoising,” *SIAM Journal on Scientific Computing*, vol. 25, no. 3, pp. 982–994, 2003.

[20] Y. Shi and Q. Chang, “Remark on convergence of algebraic multigrid in the form of matrix decomposition,” *Applied Mathematics and Computation*, vol. 170, no. 2, pp. 1170–1184, 2005.

[21] Y. Shi and Q. Chang, “Acceleration methods for image restoration problem with different boundary conditions,” *Applied Numerical Mathematics*, vol. 58, no. 5, pp. 602–614, 2008.

[22] T. Goldstein and S. Osher, “The split Bregman method for L1-regularized problems,” *SIAM Journal on Imaging Sciences*, vol. 2, no. 2, pp. 323–343, 2009.

[23] Q. Chang, L. Chien, W. Wang, and J. Xu, “A robust algorithm variation deblurring and denoising,” in *Proceedings of the 2nd International Conference on Scale Space and Variational Methods in Computer Vision (SSVM ’09)*, vol. 5567 of *Lecture Notes in Computer Science*, Springer, 2009.

[24] Q. Chang, Y. S. Wong, and H. Fu, “On the algebraic multigrid method,” *Journal of Computational Physics*, vol. 125, no. 2, pp. 279–292, 1996.

[25] Q. Chang and Z. Huang, “Efficient algebraic multigrid algorithms and their convergence,” *SIAM Journal on Scientific Computing*, vol. 24, no. 2, pp. 597–618, 2002.

[26] J. W. Ruge and K. Stüben, “Algebraic multigrid,” in *Multigrid Methods*, S. F. McCormick, Ed., vol. 3 of *Frontiers Appl. Math.*, pp. 73–130, SIAM, Philadelphia, Pa, USA, 1987.

[27] R.-Q. Jia, H. Zhao, and W. Zhao, “Convergence analysis of the Bregman method for the variational model of image denoising,” *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 367–379, 2009.

[28] A. Brandt and V. Mikulinsky, “On recombining iterants in multigrid algorithms and problems with small islands,” *SIAM Journal on Scientific Computing*, vol. 16, no. 1, pp. 20–28, 1995.

[29] C. W. Oosterlee and T. Washio, “Krylov subspace acceleration of nonlinear multigrid with application to recirculating flows,” *SIAM Journal on Scientific Computing*, vol. 21, no. 5, pp. 1670–1690, 2000.