



# A regularization model with adaptive diffusivity for variational image denoising



Po-Wen Hsieh<sup>a</sup>, Pei-Chiang Shao<sup>b</sup>, Suh-Yuh Yang<sup>b,c,\*</sup>

<sup>a</sup> Department of Applied Mathematics, National Chung Hsing University, South District, Taichung City 40227, Taiwan

<sup>b</sup> Department of Mathematics, National Central University, Zhongli District, Taoyuan City 32001, Taiwan

<sup>c</sup> National Center for Theoretical Sciences, National Taiwan University, Da'an District, Taipei City 10617, Taiwan

## ARTICLE INFO

### Article history:

Received 29 July 2017

Revised 13 November 2017

Accepted 6 December 2017

Available online 7 December 2017

### MSC:

68U10

65K10

### Keywords:

Image denoising

Total variation

Regularization

Adaptivity

Split Bregman iteration

## ABSTRACT

In this paper, motivated by approximating the Euler-Lagrange equation of the  $p$ th-order regularization for  $0 < p \leq 1$ , we propose a new regularization model with adaptive diffusivity for variational image denoising. The model is equipped with a regularization controller which is introduced to adaptively adjust the diffusivity from pixel to pixel according to the magnitude of image gradient. The associated energy functional is convex and thus the minimization problem can be efficiently solved using a modified split Bregman iterative scheme. A convergence analysis of the iterative scheme is established. Numerical experiments are performed to demonstrate the good performance of the proposed model. Comparisons with some other image denoising models are also made.

© 2017 Elsevier B.V. All rights reserved.

## 1. Introduction

The study of image denoising has a rather long history, which can be traced back to more than forty years ago. The goal is to remove or reduce the noise, that corrupts a digital image, and to make the features of image clearer and more discernible for improving the quality and ability of human or machine vision identifications. Nowadays, denoising is still an attractive and challenging subject in image processing field. Denoising methods arising from many different disciplines of mathematics and statistics have been extensively studied, such as Wiener filters [2,18], wavelet shrinkage denoising [8,13,15,33], Bayesian approaches [26,32], level set methods [25,27], nonlinear isotropic and anisotropic diffusions [5,21,29,37,38], and total variation based methods [7,30,31,34], just to name a few. For more details, we refer the reader to a recent review by Jain and Tyagi [19], see also Lee et al. [20], and many references cited therein.

In this paper, we will focus on the variational approach [1] for image denoising, which seeks the desired image  $u$  defined in  $\Omega$ , in a suitable function space, as the minimizer of an energy functional

composed of a regularization term with a data fidelity term,

$$\min_u \left\{ E_\phi(u) := \int_\Omega \left( \phi(|\nabla u|) + \frac{\lambda}{2} (u - f)^2 \right) dx \right\}, \quad (1.1)$$

where  $f$  is the given noisy image degraded by, e.g., the Gaussian white noise,  $\phi$  is a regularization function and  $|\cdot|$  denotes the usual  $\ell^2$ -norm, and  $\lambda > 0$  is a constant regularization parameter. A breakthrough in this direction is the so-called total variation (TV) regularization model [31] proposed by Rudin, Osher, and Fatemi (ROF) in 1992, where the regularizer is chosen as the  $L^1$  norm of the image gradient, i.e.,  $\phi(s) = s$  in (1.1). The search space of the ROF model is the bounded variation space  $BV(\Omega)$ , which includes piecewise smooth functions as the limit points of functions in the Sobolev space  $W^{1,1}(\Omega)$ , so that image features such as edges or other sharp structures can be successfully preserved.

Although the ROF model can suppress the variation caused by noise without penalizing edge gradient too much, it may bring the undesirable staircasing effect [4] into the ramp regions; that is, it replaces smooth transitions in the image with piecewise constant regions, which is visually unpleasant to human eyes and may be falsely identified as edges in computer vision related tasks. In fact, deriving the functional derivative of Eq. (1.1) and changing the basis to the local normal direction  $N(x) := \nabla u / |\nabla u|$  and the local tangent direction  $T(x) := \nabla u^\perp / |\nabla u|$ , we obtain the following Euler-

\* Corresponding author.

E-mail addresses: [pwhsieh@nchu.edu.tw](mailto:pwhsieh@nchu.edu.tw) (P.-W. Hsieh), [shaopj823@gmail.com](mailto:shaopj823@gmail.com) (P.-C. Shao), [syyang@math.ncu.edu.tw](mailto:syyang@math.ncu.edu.tw) (S.-Y. Yang).

Lagrange equation:

$$\frac{\phi'(|\nabla u|)}{|\nabla u|} u_{TT} + \phi''(|\nabla u|) u_{NN} + \lambda(f - u) = 0 \quad \text{in } \Omega, \quad (1.2)$$

where  $u_{TT}$  and  $u_{NN}$  represent the second derivatives in the  $T$  and  $N$  directions, respectively. For  $\phi(s) = s$ , we have  $\phi''(s) = 0$  for all  $s$ , which means that the diffusivity in the normal direction is always zero. This is the main reason why TV model can preserve edges, but may cause staircases. A possible idea for overcoming the staircasing problem is to use higher-order regularizer at pixels of moderate gradient (such as ramps). In [4], Blomgren et al. proposed a denoising model which minimizes the following adaptive energy functional,

$$\min_u \int_{\Omega} \left( |\nabla u|^{p(|\nabla u|)} + \frac{\lambda}{2} (u - f)^2 \right) dx, \quad (1.3)$$

where the exponent  $p$  is not a constant parameter but an adaptive function of  $|\nabla u(x)|$  which is used as a detector to distinguish between edges and ramps. A suitable exponent  $p$  can be designed as a decreasing function from 2 to 1, which is large at ramps and small at edges. This means that the adaptive model behaves like the TV model when  $|\nabla u(x)|$  is large and more and more like the  $H^1$  model when  $|\nabla u(x)|$  becomes small. In this direction, other detectors have been proposed in [11,42] to replace the  $|\nabla u(x)|$  in  $p(\cdot)$  for better detection.

In contrast to using higher-order regularization model such as  $\phi(s) = s^p$  in (1.1) for constant parameter  $1 < p \leq 2$  to eliminate the staircasing effect, lower-order model for  $0 < p < 1$  was shown more effective than TV ( $p = 1$ ) for preserving edges [9,23,41]. The idea is similar to using  $\ell^p$  regularization with  $0 < p < 1$  to enhance the sparsity of the basis pursuit based  $\ell^1$  regularization in the compressed sensing community [10]. Here, instead of transforming images to wavelet domain, the sparsity is considered directly on the image gradient. To overcome the NP-hard nature of the  $L^0$  minimization, the convex relaxation ( $L^1$  minimization) is often used, which is exactly the TV minimization. Although the  $L^p$  minimization with  $0 < p < 1$  can further promote the sparsity of image gradient than TV, the main difficulty lies in that  $L^p$  problem with  $0 < p < 1$  is itself non-convex. Computationally, some iteratively re-weighted type algorithm [12] should be employed for  $0 < p < 1$  such that the original non-convex problem can be approximated by a series of convex ones.

In this paper, we will develop a new denoising model with adaptive diffusivity. Motivated by approximating the Euler-Lagrange equation of the  $p$ th-order model for  $0 < p \leq 1$ , the proposed regularization model relaxes the  $p$ th-order model to a  $q$ -order one, with  $q = 1$  or 2, through a regularization controller which is introduced to adaptively adjust the diffusivity from pixel to pixel according to the magnitude of image gradient. One of the most advantageous features of the developed model is that the associated energy functional is always convex and thus we can develop a simple fast algorithm based on the split Bregman iteration [16] to accelerate the computation. In this paper, a standard convergence analysis of the iterative scheme will be given. Moreover, we will present some numerical examples to demonstrate the good performance of the newly proposed model. Comparisons with several other image denoising models are also carried out by considering the ROF model [31] implemented by the split Bregman method [16], the variants [28,40] of the Lysaker–Osher–Tai (LOT) model [24], the total generalized variation (TGV) model [6] implemented by the fast Fourier transform [35], the block-matching and 3D filtering (BM3D) approach [14], and the nonlocal diffusion (NLD) approach [17], where the LOT and TGV models mainly focus on reducing the staircasing problem and both the BM3D and NLD approaches process images non-locally. Indeed, we find that the newly proposed model is comparable to the above-mentioned

methods for restoring natural images and it seems to be better than the others in denoising synthetic images.

The remainder of this paper is organized as follows. In Section 2, we introduce the new denoising model with an adaptive controller. In Section 3, we show that this regularization model can be solved efficiently by a modified split Bregman iterative scheme. In Section 4, we give a convergence analysis of the iteration scheme. Numerical experiments are performed in Section 5 to illustrate the good performance of the newly proposed regularization model. Finally, some concluding remarks are given in Section 6.

## 2. The regularization model with an adaptive controller

In this paper, we consider the following regularization model for variational image denoising:

$$\min_u \left\{ E_{p,q}(u) := \int_{\Omega} \left( \frac{\alpha_{p,q}}{q} |\nabla u|^q + \frac{\lambda}{2} (u - f)^2 \right) dx \right\}, \quad (2.1)$$

where  $u$  and  $f$  are the recovered and noisy images, respectively;  $q = 1$  or 2 and  $\alpha_{p,q} = \alpha_{p,q}(|\nabla u^*(x)|) > 0$  is a spatially variable controller depending on the given quantity  $|\nabla u^*(x)|$  that will be defined later; the notion  $|\cdot|$  is always considered as the  $\ell^2$  norm;  $p$  is a parameter with  $0 < p \leq 1$  which should be specified in the controller  $\alpha_{p,q}$ ;  $\lambda > 0$  is a constant regularization parameter. We remark that  $|\nabla u^*(x)|$  is considered as a quantity that approximates the magnitude of the gradient of the original image at  $x$ .

In this section, we first consider  $q = 2$  (see Remark 2.2 below for  $q = 1$ ) that makes the energy functional  $E_{p,q}(u)$  in (2.1) always a convex functional in  $u$  (see Section 4) and the corresponding  $d$ -subproblem in the split Bregman-type iteration has a closed-form solution (see Section 3). Our goal is to design an effective regularization controller  $\alpha_{p,2}$  which has the adaptive capability to adjust the diffusivity from pixel to pixel according to the magnitude of image gradient. First, note that the Euler-Lagrange equation associated with (2.1) can be derived as follows:

$$\text{div}(\alpha_{p,2} \nabla u) + \lambda(f - u) = 0 \quad \text{in } \Omega. \quad (2.2)$$

From another perspective, in order to keep the edge-preserving property, in general we should consider the lower-order energy functional with  $0 < p \leq 1$ :

$$E_p(u) := \int_{\Omega} \left( \frac{1}{p} |\nabla u|^p + \frac{\lambda}{2} (u - f)^2 \right). \quad (2.3)$$

Even though (2.3) is non-convex when  $0 < p < 1$ , the Euler-Lagrange equation of (2.3) is still given by

$$\text{div}(|\nabla u|^{p-2} \nabla u) + \lambda(f - u) = 0 \quad \text{in } \Omega. \quad (2.4)$$

As a result, in comparison with (2.2), Eq. (2.4) suggests us to define the adaptive diffusivity  $\alpha_{p,2}$  in (2.1) with  $q = 2$  as

$$\alpha_{p,2}(|\nabla u^*(x)|) := \begin{cases} |\nabla u^*(x)|^{p-2} & \text{if } |\nabla u^*(x)| \neq 0, \\ \varepsilon^{-1} & \text{if } |\nabla u^*(x)| = 0, \end{cases} \quad (2.5)$$

where  $0 < p \leq 1$  and  $\varepsilon > 0$  is a prescribed small number.

Intuitively, we may expect that the model (2.1) for  $q = 2$  and  $0 < p \leq 1$  with the adaptive diffusivity (2.5) will effectively preserve edges just like the lower-order regularization model (2.3) with  $0 < p \leq 1$ . Indeed, we will show that this novel regularization model exhibits a high adaptive capability for image denoising in the following sections.

**Remark 2.1.** For the image pixels where  $|\nabla u^*(x)| \approx 0$ , the model (2.1) for  $q = 2$  and  $0 < p \leq 1$  with the adaptive diffusivity (2.5) has extra smoothing effect since  $\alpha_{p,2}$  is large. Therefore, the newly proposed model not only inherits the advantages of edge-preserving models, but also exhibits the smoothness of  $H^1$  model to alleviate

the staircasing effect. This observation will be confirmed by the numerical studies in Section 5.

**Remark 2.2.** For the case  $q = 1$  and  $0 < p \leq 1$ , following the similar process described above, we obtain the adaptive controller  $\alpha_{p,1}$ ,

$$\alpha_{p,1}(|\nabla u^*(x)|) := \begin{cases} |\nabla u^*(x)|^{p-1} & \text{if } |\nabla u^*(x)| \neq 0, \\ \varepsilon^{-1} & \text{if } |\nabla u^*(x)| = 0. \end{cases} \quad (2.6)$$

In this case, the minimization problem,

$$\min_u \left\{ E_{p,1}(u) := \int_{\Omega} \left( \alpha_{p,1} |\nabla u| + \frac{\lambda}{2} (u - f)^2 \right) dx \right\}, \quad (2.7)$$

is similar to the TV model except the adaptive diffusion coefficient  $\alpha_{p,1}(|\nabla u^*(x)|)$ . Since the regularization term will not penalize the edge too much, we believe that the model (2.7) with (2.6) can deal with more complicated boundary geometry such as snowflakes whose small-scale features of the edges can not be preserved easily. However, the staircasing effect will probably be strengthened such that the model (2.7) with (2.6) seems not suitable for denoising natural images.

**Remark 2.3.** For the general  $p$ th-order regularization model associated with the energy functional (2.3), one may design an adaptive exponent  $p = p(|\nabla u^*|) \in [1, 2]$  for adaptive smoothing, that is, using  $p$  near 1 to preserve edges and using  $p$  near 2 to smooth flat regions. However, this adaptive regularization model cannot be implemented easily by the split Bregman algorithm just like TV model. In such case, the  $d$ -subproblem in the split Bregman iteration is very time-consuming, since the solution formula at each pixel is not the same such that one cannot solve it in the vector sense and the roots of high-order polynomials should be solved numerically for  $1 < p < 2$ . In contrast, the newly proposed model (2.1) can also integrate the adaptive regularization model into the unified format and the corresponding  $d$ -subproblem can be solved very efficiently (see Section 3).

### 3. A split Bregman-type iterative scheme

In this section, we will show that the newly proposed model (2.1) can be solved numerically by a split Bregman-type iteration [16]. We remark that the main purpose of designing such a split Bregman-type iteration is to introduce a unified approach to solve the proposed model (2.1) for both  $q = 1$  and  $q = 2$ , even though the case  $q = 2$  could be solved in another way due to the better differentiability.

Below we focus on describing the algorithmic details for the case of  $q = 2$ . The other case of  $q = 1$  can be derived in a similar way with a slight modification. Without loss of generality, grayscale images are considered as  $N \times N$  matrices. Introducing the discrete gradient operator as  $(\nabla u)_{i,j} = ((\nabla_x^+ u)_{i,j}, (\nabla_y^+ u)_{i,j})$ ,

$$(\nabla_x^+ u)_{i,j} := \begin{cases} u_{i,j+1} - u_{i,j}, & 1 \leq j \leq N-1, \\ 0, & j = N, \end{cases}$$

$$(\nabla_y^+ u)_{i,j} := \begin{cases} u_{i+1,j} - u_{i,j}, & 1 \leq i \leq N-1, \\ 0, & i = N, \end{cases}$$

we discretize the model (2.1) for  $q = 2$  and  $0 < p \leq 1$  as

$$\min_u \left\{ \bar{E}_{p,2}(u) := \sum_{i,j} \left( \frac{\alpha_{p,2,i,j}}{2} |(\nabla u)_{i,j}|^2 + \frac{\lambda}{2} (u_{i,j} - f_{i,j})^2 \right) \right\}. \quad (3.1)$$

Applying the operator splitting technique to (3.1), we obtain the equivalent minimization problem,

$$\min_{u,d} \left\{ \bar{E}_{p,2}(u, d) := \sum_{i,j} \left( \frac{\alpha_{p,2,i,j}}{2} |d_{i,j}|^2 + \frac{\lambda}{2} (u_{i,j} - f_{i,j})^2 \right) \right\}$$

subject to  $d = \nabla u$ . (3.2)

The splitted problem (3.2) can be solved by using the Bregman iteration. Introducing a penalty parameter  $\gamma > 0$ , we arrive at the following unconstrained minimization problem:

$$\min_{u,d} \left\{ \bar{E}_{p,2}(u, d) := \sum_{i,j} \left( \frac{\alpha_{p,2,i,j}}{2} |d_{i,j}|^2 + \frac{\lambda}{2} (u_{i,j} - f_{i,j})^2 + \frac{\gamma}{2} |(\nabla u)_{i,j} - d_{i,j} - b_{i,j}|^2 \right) \right\}, \quad (3.3)$$

where  $b$  is a variable related to the Bregman iteration algorithm. Furthermore, (3.3) can be solved by alternating the search directions of  $u$  and  $d$  as follows:

•  **$u$ -subproblem:**

$$u^{n+1} = \arg \min_u \left\{ \sum_{i,j} \left( \frac{\lambda}{2} (u_{i,j} - f_{i,j})^2 + \frac{\gamma}{2} |(\nabla u)_{i,j} - d_{i,j}^n - b_{i,j}^n|^2 \right) \right\}; \quad (3.4)$$

•  **$d$ -subproblem:**

$$d^{n+1} = \arg \min_d \left\{ \sum_{i,j} \left( \frac{\alpha_{p,2,i,j}}{2} |d_{i,j}|^2 + \frac{\gamma}{2} |(\nabla u^{n+1})_{i,j} - d_{i,j} - b_{i,j}^n|^2 \right) \right\}, \quad (3.5)$$

where  $b^n := b^{n-1} + d^n - \nabla u^n$ .

Next we give more details about these two subproblems. For the  $u$ -subproblem, differentiating the objective function given in (3.4) and then setting it to zero, we obtain the discrete screened Poisson equation:

$$\lambda u_{i,j}^{n+1} - \gamma (\Delta u^{n+1})_{i,j} = \lambda f_{i,j} - \gamma (\text{div}(d^n + b^n))_{i,j}, \quad (3.6)$$

which is commonly seen in the standard split Bregman iteration. The discrete operators  $\text{div}$  and  $\Delta$  with Neumann boundary conditions are defined as follows [39]: given  $g = (g^1, g^2)$  with  $g^1, g^2 \in \mathbb{R}^{N \times N}$ , we define

$$(\text{div}g)_{i,j} := (\nabla_x^- g^1)_{i,j} + (\nabla_y^- g^2)_{i,j} := (g_{i,j}^1 - g_{i,j-1}^1) + (g_{i,j}^2 - g_{i-1,j}^2),$$

where  $\nabla_x^-$  and  $\nabla_y^-$  are backward difference operators with Neumann boundary conditions  $g_{1,0}^1 = g_{1,1}^1$  and  $g_{0,j}^2 = g_{1,j}^2$ . The discrete Laplacian is then defined as the composite of  $\nabla$  and  $\text{div}$  as  $\Delta u = \text{div}(\nabla u)$ . Since (3.6) produces a symmetric and diagonally dominant linear system, some iterative solvers such as Jacobi or Gauss-Seidel method can be applied for efficiently solving  $u$ .

For the  $d$ -subproblem, the objective function given in (3.5) is strictly convex and differentiable, thus  $d^{n+1}$  has the closed-form solution,

$$d_{i,j}^{n+1} = \frac{\gamma}{\alpha_{p,2,i,j} + \gamma} (\nabla u^{n+1} - b^n)_{i,j}. \quad (3.7)$$

To sum up, the split Bregman-type algorithm can be summarized as below:

$$\begin{array}{l} \text{split Bregman-type algorithm for solving (2.1) with } q = 2 \text{ and} \\ 0 < p \leq 1 \text{ initialize } u = f, d = 0, b = 0 \\ \text{while } \|u - u_{\text{prev}}\|_2^2 / \|u\|_2^2 > \text{tol do} \\ \quad \text{solve the } u \text{ - subproblem defined in (3.4) using (3.6)} \\ \quad \text{solve the } d \text{ - subproblem defined in (3.5) using (3.7)} \\ \quad b = b + d - \nabla u \end{array} \quad (3.8)$$

We give the following remark to complete this section.

**Remark 3.1.** The constant penalty parameter  $\gamma > 0$  should not be chosen extremely small nor large in order to ensure good convergence rate and numerical stability. In addition, the solution in the  $u$ -subproblem need not be too accurate, an approximate solution is almost good enough. In general, to speed up the computation, only one simple iteration of Jacobi or Gauss Seidel method is used. The split Bregman-type iteration will rectify the approximate solution quickly. Therefore, based on this implementation viewpoint, we can employ the gradient information  $|(\nabla u^{n+1})_{i,j}|$  obtained from the  $n + 1$  step to approximate  $\alpha_{p,q}(|(\nabla u^*)_{i,j}|)$  by simply setting

$$\alpha_{p,q,i,j} := \begin{cases} |(\nabla u^{n+1})_{i,j}|^{p-q} & \text{if } |(\nabla u^{n+1})_{i,j}| \neq 0, \\ \varepsilon^{-1} & \text{if } |(\nabla u^{n+1})_{i,j}| = 0. \end{cases} \quad (3.9)$$

Moreover, letting  $\varepsilon \rightarrow 0^+$ , the solution of  $d$ -subproblem can be directly written as follows:

$$d_{i,j}^{n+1} = \frac{\gamma |(\nabla u^{n+1})_{i,j}|^{2-p}}{1 + \gamma |(\nabla u^{n+1})_{i,j}|^{2-p}} (\nabla u^{n+1} - b^n)_{i,j} \quad \text{for } q = 2,$$

and

$$d_{i,j}^{n+1} = \frac{(\nabla u^{n+1})_{i,j} - b_{i,j}^n}{|(\nabla u^{n+1})_{i,j} - b_{i,j}^n|} \times \max \left\{ |(\nabla u^{n+1})_{i,j} - b_{i,j}^n| - \frac{1}{\gamma |(\nabla u^{n+1})_{i,j}|^{1-p}}, 0 \right\} \quad \text{for } q = 1,$$

where  $0 < p \leq 1$ .

#### 4. Convergence analysis

In this section, we will give a convergence analysis of the model (2.1). We only consider the case  $q = 2$  and  $0 < p \leq 1$ , but all the results still hold for  $q = 1$  and  $0 < p \leq 1$  with a slight modification (cf. [39]). First, the convexity of the energy functional  $E_{p,2}$ , which guarantees the existence of minimizer of the minimization problem (2.1), can be proved directly.

**Lemma 4.1.** *The energy functional  $E_{p,2}$  is convex. That is, for any  $u$  and  $v$ , we have*

$$E_{p,2}(tu + (1-t)v) \leq tE_{p,2}(u) + (1-t)E_{p,2}(v) \quad \forall 0 \leq t \leq 1,$$

where

$$E_{p,2}(u) := \int_{\Omega} \left( \frac{\alpha_{p,2}}{2} |\nabla u|^2 + \frac{\lambda}{2} (u-f)^2 \right) dx, \quad \alpha_{p,2} > 0. \quad (4.1)$$

**Proof.** By a direct computation, we have

$$\begin{aligned} & E_{p,2}(tu + (1-t)v) \\ &= \int_{\Omega} \frac{\alpha_{p,2}}{2} \left( t^2 |\nabla u|^2 + 2t(1-t) \nabla u \cdot \nabla v + (1-t)^2 |\nabla v|^2 \right) \\ & \quad + \frac{\lambda}{2} \left( t^2 (u-f)^2 + (1-t)^2 (v-f)^2 + 2t(1-t)(u-f)(v-f) \right) dx, \end{aligned}$$

which implies

$$\begin{aligned} & E_{p,2}(tu + (1-t)v) \\ & \leq \int_{\Omega} \frac{\alpha_{p,2}}{2} \left( t^2 |\nabla u|^2 + t(1-t)(|\nabla u|^2 + |\nabla v|^2) + (1-t)^2 |\nabla v|^2 \right) \\ & \quad + \frac{\lambda}{2} \left( t^2 (u-f)^2 + (1-t)^2 (v-f)^2 + t(1-t)((u-f)^2 + (v-f)^2) \right) dx \\ & = \int_{\Omega} \frac{\alpha_{p,2}}{2} \left( t |\nabla u|^2 + (1-t) |\nabla v|^2 \right) + \frac{\lambda}{2} \left( t(u-f)^2 + (1-t)(v-f)^2 \right) dx \\ & = tE_{p,2}(u) + (1-t)E_{p,2}(v). \end{aligned}$$

This completes the proof.  $\square$

Secondly, the uniqueness of minimizer can be ensured easily.

**Lemma 4.2.** *If  $u_1$  and  $u_2$  are two minimizers of problem (2.1) with  $q = 2$  and  $0 < p \leq 1$ , then we have  $u_1 = u_2$ .*

**Proof.** From the definition of the energy functional  $E_{p,2}(u)$  in (4.1), we have

$$\begin{aligned} E_{p,2}\left(\frac{u_1 + u_2}{2}\right) &= \int_{\Omega} \frac{\alpha_{p,2}}{2} \left| \frac{\nabla u_1}{2} + \frac{\nabla u_2}{2} \right|^2 + \frac{\lambda}{2} \left( \frac{u_1 - f}{2} + \frac{u_2 - f}{2} \right)^2 dx \\ &\leq \int_{\Omega} \frac{\alpha_{p,2}}{2} \left( \frac{|\nabla u_1|^2}{2} + \frac{|\nabla u_2|^2}{2} \right) \\ & \quad + \frac{\lambda}{2} \left( \frac{(u_1 - f)^2}{2} + \frac{(u_2 - f)^2}{2} - \left( \frac{(u_1 - f) - (u_2 - f)}{2} \right)^2 \right) dx \\ &= \frac{1}{2} E_{p,2}(u_1) + \frac{1}{2} E_{p,2}(u_2) - \int_{\Omega} \frac{\lambda}{2} \left( \frac{u_1 - u_2}{2} \right)^2 dx \\ &= E_{p,2}(u_1) - \int_{\Omega} \frac{\lambda}{2} \left( \frac{u_1 - u_2}{2} \right)^2 dx. \end{aligned}$$

If  $u_1 \neq u_2$ , then the above equation gives a contradiction that  $u_1$  is not a minimizer.  $\square$

The well-posedness of the discrete counterpart (3.1) can be proved in a similar way. Next, we are going to show that the split Bregman-type iterative scheme proposed in Section 3 will converge to the solution of the discretized model (3.1). We first give some notations and definitions. Let  $U$  denote the Euclidean space  $\mathbb{R}^{N \times N}$  and  $D := U \times U$ . Define the usual inner products and norms on  $U$  and  $D$ , respectively,

$$(u, v)_U = \sum_{i,j} u_{i,j} v_{i,j}, \quad \|u\|_U = \sqrt{(u, u)_U}, \quad (4.2)$$

$$(d, e)_D = \sum_{k=1}^2 (d_k, e_k)_U, \quad \|d\|_D = \sqrt{(d, d)_D}, \quad (4.3)$$

where  $d := (d_1, d_2)$  and  $e := (e_1, e_2)$ . With these notations, problem (3.3) can be rewritten as

$$\min_{u,d} \left\{ \bar{E}_{p,2}(u, d) := \frac{1}{2} \|\sqrt{\alpha_{p,2}} d\|_D^2 + \frac{\lambda}{2} \|u - f\|_U^2 + \frac{\gamma}{2} \|\nabla u - d - b\|_D^2 \right\}, \quad (4.4)$$

or in an equivalent form,

$$\begin{aligned} \min_{u,d} \left\{ \bar{L}_{p,2}(u, d) := \frac{1}{2} \|\sqrt{\alpha_{p,2}} d\|_D^2 + \frac{\lambda}{2} \|u - f\|_U^2 + \gamma (b, d - \nabla u)_D \right. \\ \left. + \frac{\gamma}{2} \|d - \nabla u\|_D^2 \right\}, \end{aligned} \quad (4.5)$$

provided we expand the last term in (4.4) and drop the constant term  $\|b\|_D^2$ . In order to convert problem (4.5) into a saddle-point problem, we claim the following lemma:

**Lemma 4.3.** *Assume that  $(u, d)$  is the solution of problem (4.5). Then we have*

$$\begin{aligned} & \frac{1}{2} \|\sqrt{\alpha_{p,2}} e\|_D^2 - \frac{1}{2} \|\sqrt{\alpha_{p,2}} d\|_D^2 \\ & \quad + \gamma (b, e - d)_D + \gamma (d - \nabla u, e - d)_D \geq 0, \end{aligned} \quad (4.6)$$

$$\begin{aligned} & \frac{\lambda}{2} \|v - f\|_U^2 - \frac{\lambda}{2} \|u - f\|_U^2 + \gamma (\text{div} b, v - u)_U \\ & \quad + \gamma (\text{div}(d - \nabla u), v - u)_U \geq 0, \end{aligned} \quad (4.7)$$

for all  $(v, e) \in U \times D$ .

**Proof.** Given any  $e \in D$ , since  $\bar{L}_{p,2}(u, d) \leq \bar{L}_{p,2}(u, d + t(e - d))$  for all  $0 \leq t \leq 1$ , we have

$$\begin{aligned} & \frac{1}{2} \|\sqrt{\alpha_{p,2}}(d + t(e - d))\|_D^2 - \frac{1}{2} \|\sqrt{\alpha_{p,2}} d\|_D^2 + \gamma (b, t(e - d))_D \\ & \quad + \frac{\gamma}{2} \left( \|d + t(e - d) - \nabla u\|_D^2 - \|d - \nabla u\|_D^2 \right) \geq 0, \end{aligned} \quad (4.8)$$

or equivalently,

$$\frac{1}{2} \|\sqrt{\alpha_{p,2}}(te + (1-t)d)\|_D^2 - \frac{1}{2} \|\sqrt{\alpha_{p,2}}d\|_D^2 + \gamma(b, t(e-d))_D + \frac{\gamma}{2} (\|t(e-d)\|_D^2 + 2(t(e-d), d - \nabla u)_D) \geq 0. \quad (4.9)$$

Using the convexity of the norm  $\|\cdot\|_D$ , i.e.,

$$\frac{1}{2} \|\sqrt{\alpha_{p,2}}(te + (1-t)d)\|_D^2 \leq \frac{t}{2} \|\sqrt{\alpha_{p,2}}e\|_D^2 + \frac{1-t}{2} \|\sqrt{\alpha_{p,2}}d\|_D^2, \quad (4.10)$$

or equivalently,

$$\frac{1}{2} \|\sqrt{\alpha_{p,2}}(te + (1-t)d)\|_D^2 - \frac{1}{2} \|\sqrt{\alpha_{p,2}}d\|_D^2 \leq \frac{t}{2} (\|\sqrt{\alpha_{p,2}}e\|_D^2 - \|\sqrt{\alpha_{p,2}}d\|_D^2), \quad (4.11)$$

we obtain

$$\frac{t}{2} (\|\sqrt{\alpha_{p,2}}e\|_D^2 - \|\sqrt{\alpha_{p,2}}d\|_D^2) + \gamma(b, t(e-d))_D + \frac{\gamma}{2} (\|t(e-d)\|_D^2 + 2(t(e-d), d - \nabla u)_D) \geq 0, \quad (4.12)$$

or equivalently,

$$\frac{1}{2} (\|\sqrt{\alpha_{p,2}}e\|_D^2 - \|\sqrt{\alpha_{p,2}}d\|_D^2) + \gamma(b, e-d)_D + \frac{\gamma}{2} (\|\sqrt{t}(e-d)\|_D^2 + 2(e-d, d - \nabla u)_D) \geq 0. \quad (4.13)$$

Taking  $t \rightarrow 0^+$ , we have (4.6). The inequality (4.7) can be proved in a similar way.  $\square$

We remark that with the above Lemmas 4.1.4.2.4.3 and Theorem 4.1 given in [39], one can show that problem (4.5) is equivalent to the following saddle-point problem: Find  $(u, d, b) \in U \times D \times D$  such that

$$\tilde{L}_{p,2}(u, d, c) \leq \tilde{L}_{p,2}(u, d, b) \leq \tilde{L}_{p,2}(v, e, b), \quad (4.14)$$

for all  $(v, e, c) \in U \times D \times D$ , where the augmented Lagrangian functional  $\tilde{L}_{p,2} : U \times D \times D \rightarrow \mathbb{R}$  is defined by

$$\tilde{L}_{p,2}(v, e, c) := \frac{1}{2} \|\sqrt{\alpha_{p,2}}e\|_D^2 + \frac{\lambda}{2} \|v - f\|_U^2 + \gamma(c, e - \nabla v)_D + \frac{\gamma}{2} \|e - \nabla v\|_D^2. \quad (4.15)$$

In the following theorem, we show that the sequence generated by the proposed split Bregman-type algorithm (3.8) converges to the solution of (4.14).

**Theorem 4.1.** Let  $(u, d, b)$  be the saddle point of the functional  $\tilde{L}_{p,2}$  defined by (4.14). Then the sequence  $(u^n, d^n, b^n)$  generated by (3.8) satisfies  $\lim_{n \rightarrow \infty} u^n = u$  and  $\lim_{n \rightarrow \infty} d^n = d$ .

**Proof.** We follow the techniques given in [39]. Define  $\tilde{u}^n := u^n - u$ ,  $\tilde{d}^n := d^n - d$ , and  $\tilde{b}^n := b^n - b$ , then we have  $\tilde{b}^{n+1} = \tilde{b}^n + \tilde{d}^{n+1} - \nabla \tilde{u}^{n+1}$  and

$$\|\tilde{b}^n\|_D^2 - \|\tilde{b}^{n+1}\|_D^2 = -2(\tilde{b}^n, \tilde{d}^{n+1} - \nabla \tilde{u}^{n+1})_D - \|\tilde{d}^{n+1} - \nabla \tilde{u}^{n+1}\|_D^2. \quad (4.16)$$

From Lemma 4.3, we have

$$\frac{1}{2} \|\sqrt{\alpha_{p,2}}d^{n+1}\|_D^2 - \frac{1}{2} \|\sqrt{\alpha_{p,2}}d\|_D^2 + \gamma(b, d^{n+1} - d)_D + \gamma(d - \nabla u, d^{n+1} - d)_D \geq 0 \quad (4.17)$$

and

$$\frac{\lambda}{2} \|u^{n+1} - f\|_U^2 - \frac{\lambda}{2} \|u - f\|_U^2 + \gamma(\operatorname{div} b, u^{n+1} - u)_U + \gamma(\operatorname{div}(d - \nabla u), u^{n+1} - u)_U \geq 0. \quad (4.18)$$

On the other hand, since  $(u^{n+1}, d^{n+1})$  is the minimizer at each iteration of (3.8), we have

$$\frac{1}{2} \|\sqrt{\alpha_{p,2}}d\|_D^2 - \frac{1}{2} \|\sqrt{\alpha_{p,2}}d^{n+1}\|_D^2 + \gamma(b^n, d - d^{n+1})_D + \gamma(d^{n+1} - \nabla u^{n+1}, d - d^{n+1})_D \geq 0 \quad (4.19)$$

and

$$\frac{\lambda}{2} \|u - f\|_U^2 - \frac{\lambda}{2} \|u^{n+1} - f\|_U^2 + \gamma(\operatorname{div} b^n, u - u^{n+1})_U + \gamma(\operatorname{div}(d^n - \nabla u^{n+1}), u - u^{n+1})_U \geq 0. \quad (4.20)$$

Adding (4.17)–(4.20) together and using the integration by parts, we obtain

$$(\tilde{b}^n, \tilde{d}^{n+1} - \nabla \tilde{u}^{n+1})_D + \|\tilde{d}^{n+1} - \nabla \tilde{u}^{n+1}\|_D^2 + (\nabla \tilde{u}^{n+1}, \tilde{d}^{n+1} - \tilde{d}^n)_D \leq 0. \quad (4.21)$$

It then follows from (4.16) and (4.21) that

$$\|\tilde{b}^n\|_D^2 - \|\tilde{b}^{n+1}\|_D^2 \geq \|\tilde{d}^{n+1} - \nabla \tilde{u}^{n+1}\|_D^2 + 2(\nabla \tilde{u}^{n+1}, \tilde{d}^{n+1} - \tilde{d}^n)_D. \quad (4.22)$$

The inner product  $(\nabla \tilde{u}^{n+1}, \tilde{d}^{n+1} - \tilde{d}^n)_D$  in (4.22) can be further expressed as

$$(\nabla \tilde{u}^{n+1}, \tilde{d}^{n+1} - \tilde{d}^n)_D = (\nabla \tilde{u}^{n+1} - \nabla \tilde{u}^n, \tilde{d}^{n+1} - \tilde{d}^n)_D + (\nabla \tilde{u}^n - \tilde{d}^n, \tilde{d}^{n+1} - \tilde{d}^n)_D + (\tilde{d}^n, \tilde{d}^{n+1} - \tilde{d}^n)_D. \quad (4.23)$$

Using the construction of  $d^n$  and  $d^{n+1}$ , it follows that

$$\frac{1}{2} \|\sqrt{\alpha_{p,2}}d^{n+1}\|_D^2 - \frac{1}{2} \|\sqrt{\alpha_{p,2}}d^n\|_D^2 + \gamma(b^{n-1}, d^{n+1} - d^n)_D + \gamma(d^n - \nabla u^n, d^{n+1} - d^n)_D \geq 0 \quad (4.24)$$

and

$$\frac{1}{2} \|\sqrt{\alpha_{p,2}}d^n\|_D^2 - \frac{1}{2} \|\sqrt{\alpha_{p,2}}d^{n+1}\|_D^2 + \gamma(b^n, d^n - d^{n+1})_D + \gamma(d^{n+1} - \nabla u^{n+1}, d^n - d^{n+1})_D \geq 0. \quad (4.25)$$

Adding (4.24) and (4.25), we obtain

$$\|\tilde{d}^{n+1} - \tilde{d}^n\|_D^2 + (\tilde{d}^{n+1} - \tilde{d}^n, \tilde{b}^n - \tilde{b}^{n-1})_D - (\tilde{d}^{n+1} - \tilde{d}^n, \nabla \tilde{u}^{n+1} - \nabla \tilde{u}^n)_D \leq 0. \quad (4.26)$$

Replacing  $\tilde{b}^n - \tilde{b}^{n-1}$  in (4.26) with  $\tilde{d}^n - \nabla \tilde{u}^n$ , we have

$$(\tilde{d}^{n+1} - \tilde{d}^n, \nabla \tilde{u}^{n+1} - \nabla \tilde{u}^n)_D + (\tilde{d}^{n+1} - \tilde{d}^n, \nabla \tilde{u}^n - \tilde{d}^n)_D \geq \|\tilde{d}^{n+1} - \tilde{d}^n\|_D^2. \quad (4.27)$$

Combining (4.23), (4.27), and the identity

$$(\tilde{d}^n, \tilde{d}^{n+1} - \tilde{d}^n)_D = \frac{1}{2} (\|\tilde{d}^{n+1}\|_D^2 - \|\tilde{d}^n\|_D^2 - \|\tilde{d}^{n+1} - \tilde{d}^n\|_D^2)$$

gives

$$(\nabla \tilde{u}^{n+1}, \tilde{d}^{n+1} - \tilde{d}^n)_D \geq \frac{1}{2} (\|\tilde{d}^{n+1}\|_D^2 - \|\tilde{d}^n\|_D^2 + \|\tilde{d}^{n+1} - \tilde{d}^n\|_D^2). \quad (4.28)$$

From (4.22) and (4.28), it follows that

$$\|\tilde{b}^n\|_D^2 + \|\tilde{d}^n\|_D^2 - (\|\tilde{b}^{n+1}\|_D^2 + \|\tilde{d}^{n+1}\|_D^2) \geq \|\tilde{d}^{n+1} - \nabla \tilde{u}^{n+1}\|_D^2 + \|\tilde{d}^{n+1} - \tilde{d}^n\|_D^2, \quad (4.29)$$

which indicates that

$$\lim_{n \rightarrow \infty} \|\tilde{d}^n - \nabla \tilde{u}^n\|_D = \lim_{n \rightarrow \infty} \|d^{n+1} - d^n\|_D = 0.$$

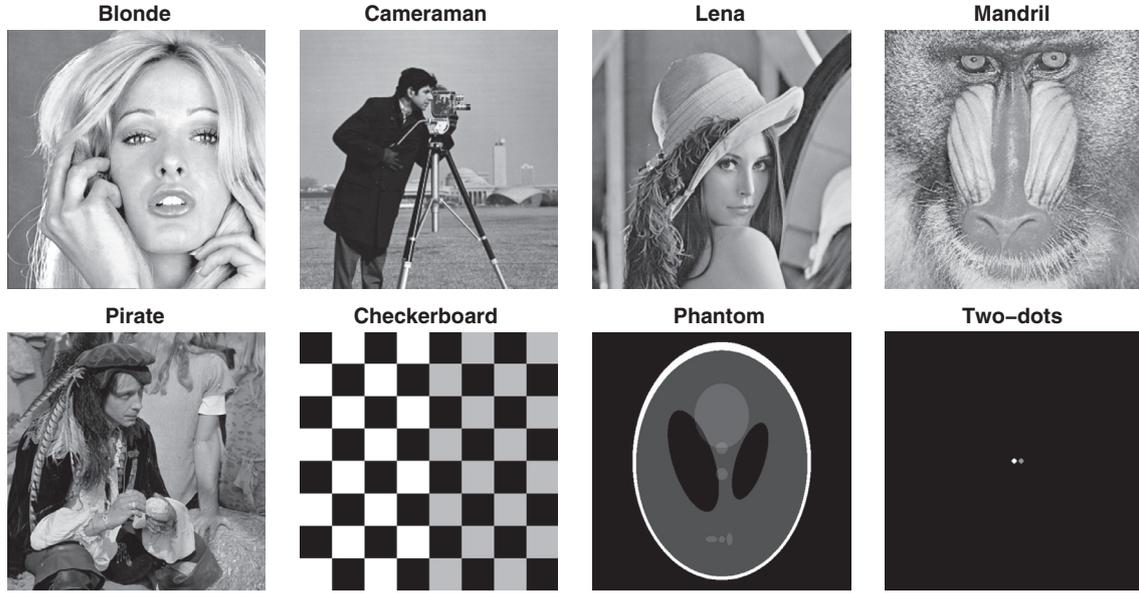


Fig. 1. Test images used in this paper: “Blonde”, “Cameraman”, “Lena”, “Mandril”, and “Pirate” are natural images, while “Checkerboard”, “Phantom”, and “Two-dots” are synthetic images. All these test images are of size  $512 \times 512$  except that the image “Two-dots” is only of size  $300 \times 300$ .

Table 1  
Index values of present algorithm with  $(p, q) = (1, 2)$  and TV-SB ( $p = 1$ ) for Example 5.1.

Image	$\sigma$	$\lambda$	SNR TV-SB	SNR Present	SSIM TV-SB	SSIM Present	# Iter. TV-SB	# Iter. Present
Lena	15	0.085	17.80	17.82	0.9697	0.9693	13	14
		0.070	17.55	17.73	0.9684	0.9696	14	14
		0.055	16.95	17.27	0.9642	0.9670	16	15
		0.040	15.85	16.24	0.9554	0.9593	19	17
Lena	20	0.025	14.14	14.46	0.9396	0.9430	25	23
		0.060	16.50	16.53	0.9615	0.9613	15	16
		0.050	16.28	16.44	0.9601	0.9616	17	17
		0.040	15.66	15.93	0.9546	0.9575	19	18
Lena	25	0.030	14.71	15.02	0.9455	0.9490	23	21
		0.020	13.29	13.61	0.9314	0.9344	29	26
		0.045	15.55	15.60	0.9541	0.9544	17	18
		0.040	15.37	15.50	0.9528	0.9542	19	19
Blonde	20	0.035	15.09	15.29	0.9501	0.9525	20	20
		0.030	14.61	14.86	0.9450	0.9480	23	21
Cameraman	20	0.025	14.07	14.34	0.9393	0.9423	26	24
		0.050	13.67	13.80	0.9424	0.9443	16	16
Mandril	20	0.050	19.06	19.19	0.9690	0.9702	16	17
Pirate	20	0.050	10.03	10.25	0.8943	0.9008	16	17
Pirate	20	0.050	14.44	14.61	0.9366	0.9396	17	18

Then using the fact that  $\tilde{L}_{p,2}(u, d, c) \leq \tilde{L}_{p,2}(u, d, b)$  for all  $c \in U$  implies  $d = \nabla u$  and we have  $\lim_{n \rightarrow \infty} \|d^n - \nabla u^n\|_D = 0$ . The remainder of the proof is similar to that of Theorem 4.3 and Theorem 4.4 in [39]. Thus, we can prove that  $\lim_{n \rightarrow \infty} u^n = u$  and  $\lim_{n \rightarrow \infty} d^n = d$ . This completes the proof.  $\square$

### 5. Numerical experiments

In this section, we will give some numerical examples to demonstrate the effective performance of the newly proposed approach. We first introduce two commonly used indices to measure the quality of images and to evaluate the denoising performance. Let  $u, \tilde{u}$  and  $\bar{u}$  denote the processed image, the clean image, and the mean intensity of the clean image, respectively. Let

$$\text{MSE}(v, w) := \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N (v_{i,j} - w_{i,j})^2$$

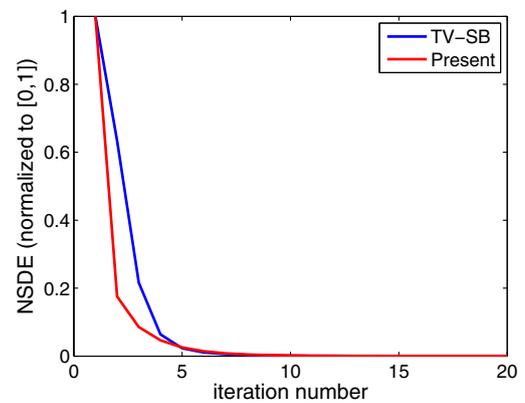
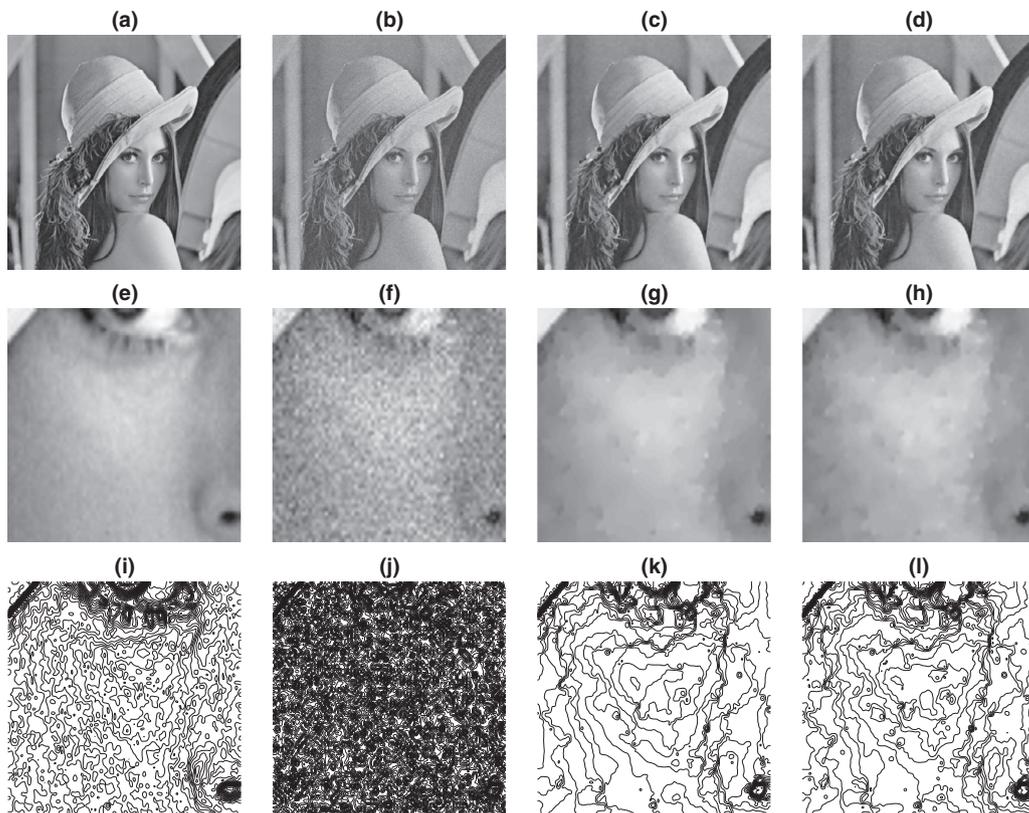
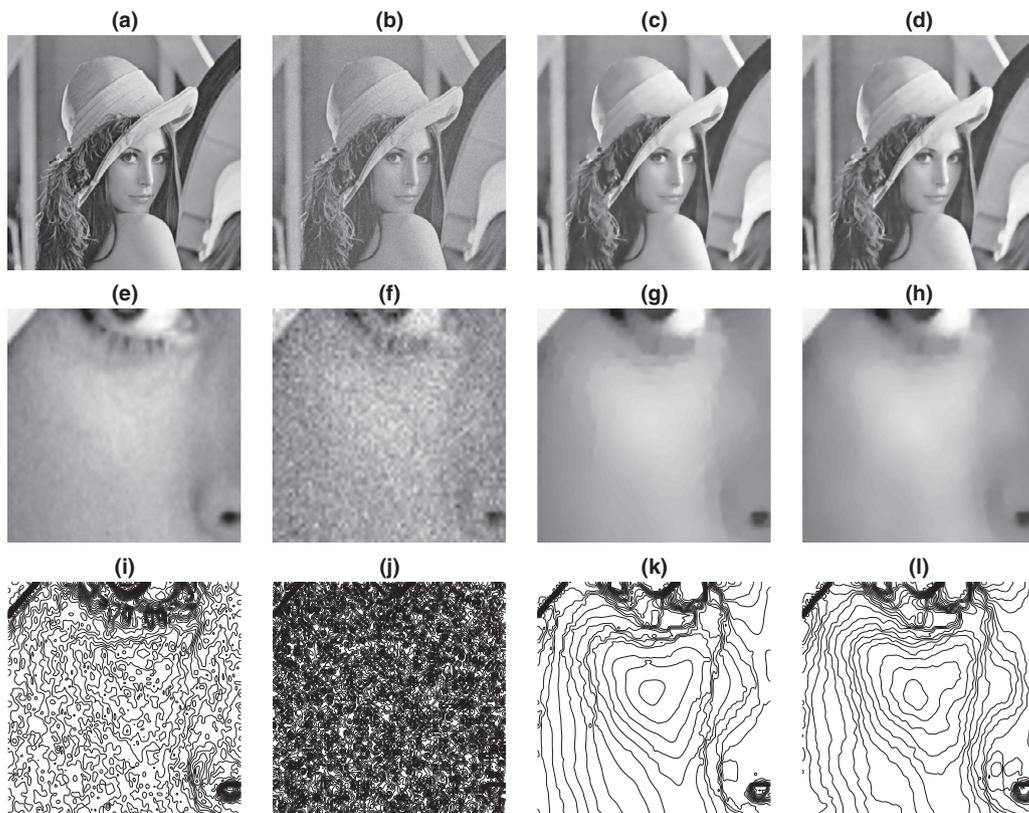


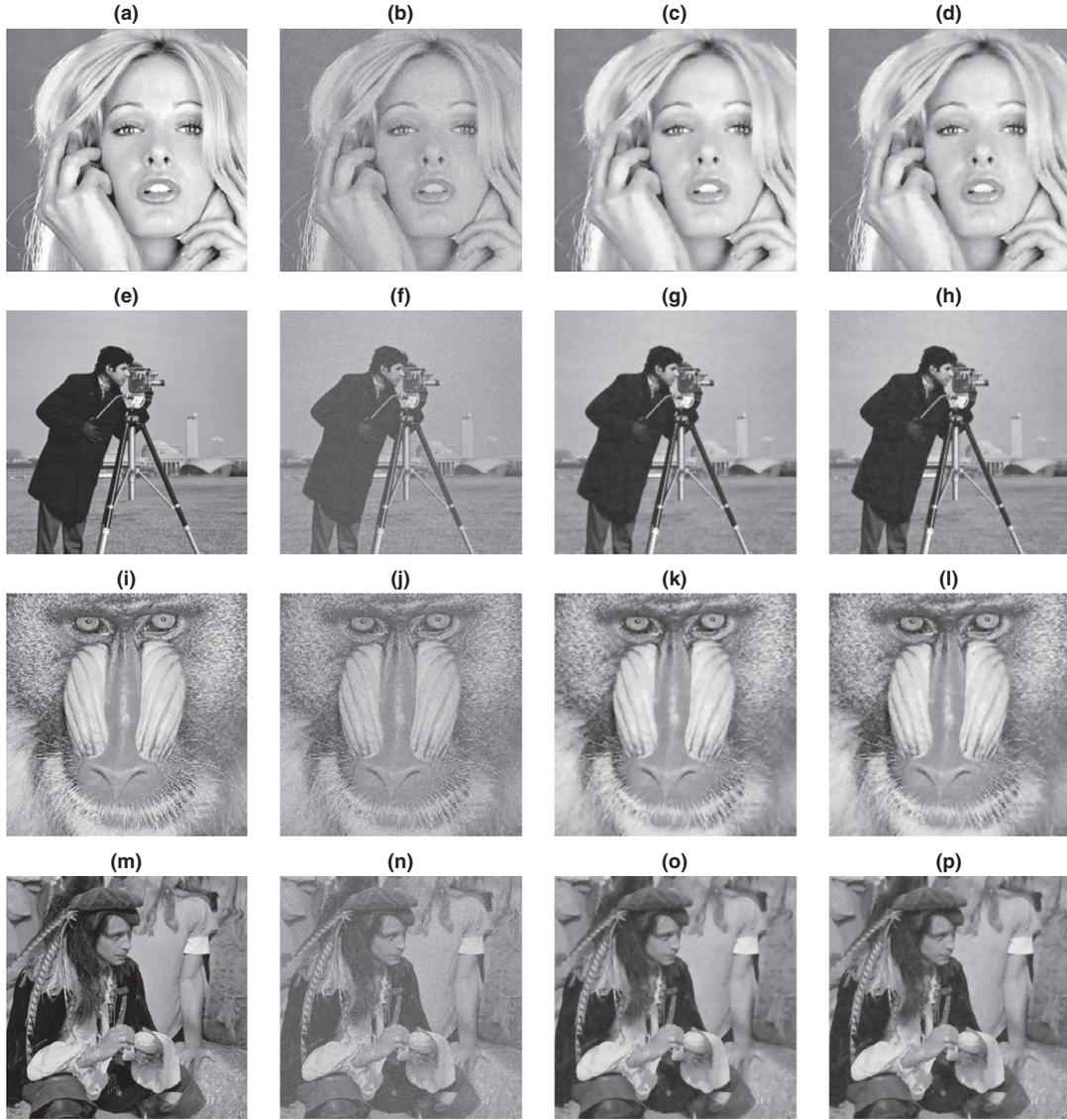
Fig. 2. Convergence plots for the test image “Lena” with  $\sigma = 15$  and  $\lambda = 0.085$ , where NSDE is calculated at each iteration. With the same error tolerance  $10^{-6}$ , present algorithm stops after 14 iterations and TV-SB stops after 13 iterations.



**Fig. 3.** Numerical results of Example 5.1 (“Lena”), where the noise level is  $\sigma = 15$  and the best denoising parameter is  $\lambda = 0.085$  for both TV-SB and present algorithm. (a), (e), and (i) are the global, local, and contour plot of the original “Lena” image; (b), (f), and (j) correspond to the noisy version; (c), (g), and (k) are the denoising results by TV-SB; (d), (h), and (l) are the denoising results by present algorithm with  $(p, q) = (1, 2)$ . The corresponding SNR, SSIM, iteration number are (17.80, 0.9697, 13) for TV-SB and (17.82, 0.9693, 14) for present algorithm.



**Fig. 4.** Numerical results of Example 5.1 (“Lena”), where the noise level is  $\sigma = 15$  and the denoising parameter is  $\lambda = 0.040$  for both TV-SB and present algorithm. (a), (e), and (i) are the global, local, and contour plot of the original “Lena” image; (b), (f), and (j) correspond to the noisy version; (c), (g), and (k) are the denoising results by TV-SB; (d), (h), and (l) are the denoising results by present algorithm with  $(p, q) = (1, 2)$ . The corresponding SNR, SSIM, iteration number are (15.85, 0.9554, 19) for TV-SB and (16.24, 0.9593, 17) for present algorithm.



**Fig. 5.** Numerical results of Example 5.1, where the noise level is  $\sigma = 20$  and the denoising parameter is fixed at  $\lambda = 0.050$  for both TV-SB and present algorithm. (a), (e), (i), (m) are original images; (b), (f), (j), (n) correspond to the noisy version; (c), (g), (k), (o) are the denoising results by TV-SB; (d), (h), (l), (p) are the denoising results by present algorithm with  $(p, q) = (1, 2)$ . The corresponding SNR, SSIM, iteration number are summarized in Table 1.

denote the mean squared error of any two images  $v$  and  $w$ . The signal-to-noise ratio (SNR) and the structural similarity (SSIM) [36] are respectively defined as follows:

$$\text{SNR} := 10 \log_{10} \frac{\text{MSE}(\tilde{u}, \bar{u})}{\text{MSE}(\tilde{u}, u)}, \tag{5.1}$$

$$\text{SSIM} := \frac{1}{M} \sum_{m=1}^M \text{ssim}(\tilde{u}_m, u_m), \tag{5.2}$$

where  $M$  is the total window number,  $\tilde{u}_m$  and  $u_m$  are the restriction of  $\tilde{u}$  and  $u$  to the  $m$ th window, with the local similarity index  $\text{ssim}$  defined by

$$\text{ssim}(v, w) := \frac{(2\mu_v\mu_w + c_1)(2\sigma_{vw} + c_2)}{(\mu_v^2 + \mu_w^2 + c_1)(\sigma_v^2 + \sigma_w^2 + c_2)}, \tag{5.3}$$

where  $\mu_v$  and  $\mu_w$  are the mean intensity of  $v$  and  $w$ ;  $\sigma_v$  and  $\sigma_w$  are the variance,  $\sigma_{vw}$  is the covariance of  $v$  and  $w$ ; and  $c_1$  and  $c_2$  are two numbers to stabilize the division. To measure the convergency, the so-called normalized step difference energy (NSDE)

is calculated at each iteration,

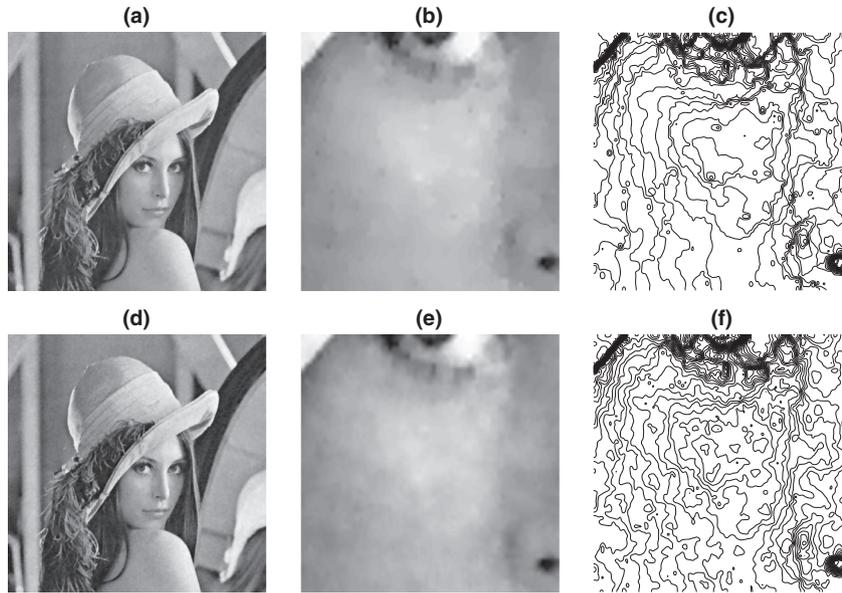
$$\text{NSDE} := \frac{\|u^n - u^{n-1}\|^2}{\|u^n\|^2}, \tag{5.4}$$

where  $u^n$  and  $u^{n-1}$  are the obtained image vectors at the  $n$ th and  $(n - 1)$  th iterations, respectively.

For all test problems, the intensity values of test images are in  $[0, 255]$ . We take the stopping tolerance  $\text{tol} = 10^{-6}$ , two stabilizing numbers  $c_1 = c_2 = 0.05$ , and consider windows of size  $8 \times 8$  for SSIM. In each  $u$ -subproblem, only one simple iteration of the Jacobi method is used to obtain an approximation of  $u^{n+1}$ . In each  $d$ -subproblem, we adopt formulas given in Remark 3.1. The test images are shown in Fig. 1, where “Blonde”, “Cameraman”, “Lena”, “Mandrill”, and “Pirate” are natural images downloaded from the website

[http://www.imageprocessingplace.com/root\\_files\\_V3/image\\_databases.htm](http://www.imageprocessingplace.com/root_files_V3/image_databases.htm).

while “Checkerboard”, “Phantom”, and “Two-dots” are synthetic images generated directly by MATLAB. All test images are



**Fig. 6.** Numerical results of the “Lena” image (cf. Remark 5.1), where the noise level is  $\sigma = 15$  and the best denoising parameter is  $\lambda = 0.085$  for both the TV-SB and present algorithm. (a), (b), and (c) are the denoising results by TV-SB; (d), (e), and (f) are the denoising results by present algorithm using the exact gradient of the original image as the adaptive controller  $\alpha_{1,2}$  defined in (5.5). The SNR, SSIM, iteration number are (17.76, 0.9694, 13) for TV-SB and (19.07, 0.9766, 10) for present algorithm.

of size  $512 \times 512$  except that the image “Two-dots” is only of size  $300 \times 300$ .

**Example 5.1** (natural images). In this example, we take  $(p, q) = (1, 2)$ ,  $\gamma = 5/255$  and study the efficiency of the proposed algorithm for natural images. The image “Lena” is contaminated with three different levels of Gaussian noise ( $\sigma = 15, 20$ , and  $25$ ), and the others are corrupted with median level  $\sigma = 20$ . We compare the SNR, SSIM, and image quality with those generated by the TV model using the split Bregman algorithm (abbreviated as TV-SB) [16]. The indices for various images, noise levels, and denoising parameters are summarized in Table 1, where the iteration numbers are presented as well. The denoising results are shown in Figs. 3–5. From the numerical results presented in Table 1 and Figs. 2–4, we can conclude the followings:

- The SNR and SSIM of the present algorithm are generally slightly greater than those of the TV-SB, which means that our algorithm shows at least comparable performance to the TV-SB (see Table 1 and Fig. 3).
- Though these two methods have very similar indices, they have rather different image quality. Because the proposed model is like the  $H^1$  model, we can expect the proposed model to exhibit more smoothing characteristics. Indeed, numerical results show that present algorithm always puts extra smoothness into flat regions so that some artifacts are prevented and the image looks more natural. In addition, it can still retain the edge-preserving property similar to TV-SB, no matter how we decrease the parameter  $\lambda$ , see Fig. 4(g) and (h). We note that the density of the contour plots was used in [3,22] to identify the smoothness of images.
- The NSDE normalized to  $[0, 1]$  is shown in Fig. 2, which shows that present algorithm has a similar convergence behavior to the TV-SB.

The good performance shows that our model may be more suitable for practical use, since the best denoising parameter  $\lambda$  for TV-SB seems not easy to estimate.

**Remark 5.1.** It is worth mentioning that if we take

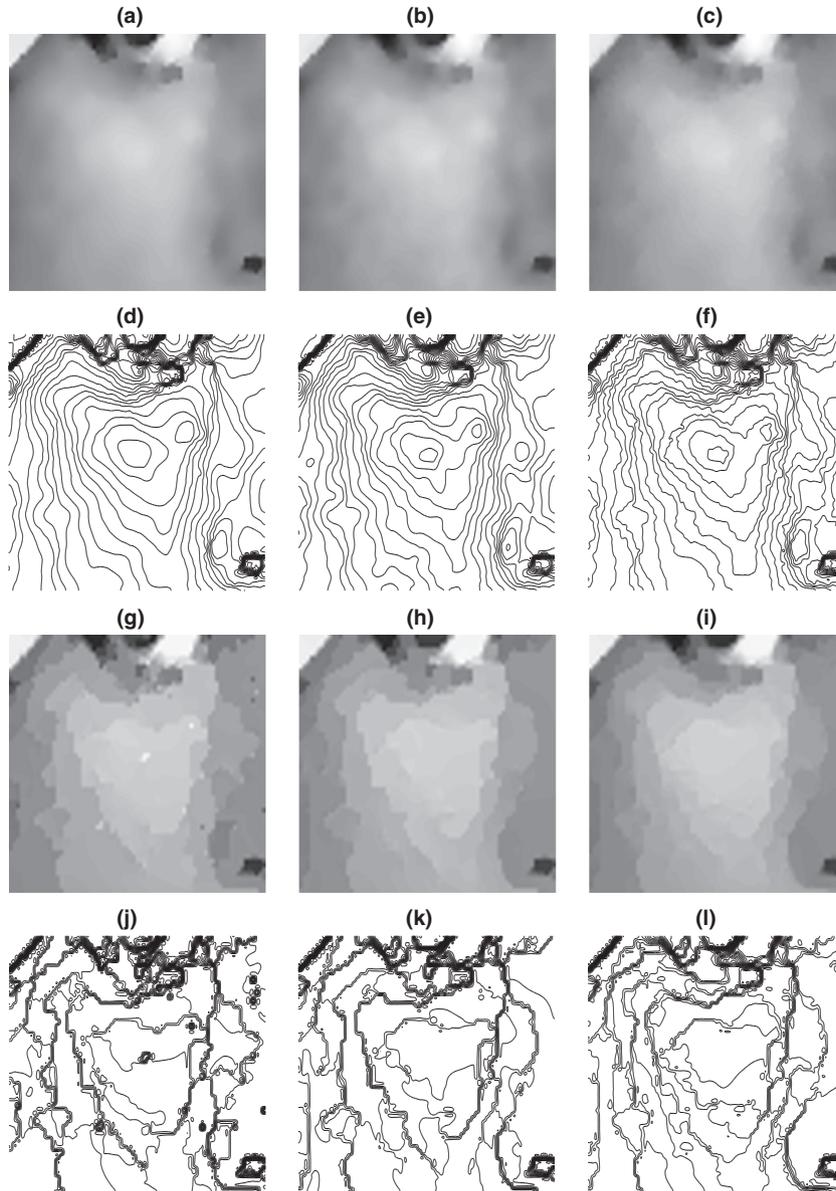
$$\alpha_{p,2}(|\nabla u^*|) := |\nabla u_e|^{p-2}, \quad (5.5)$$

where  $\nabla u_e$  represents the gradient of the original image, the numerical simulation shows that our model has potential to produce a wonderful result, see Fig. 6. It not only gives higher index values, SNR= 19.07 and SSIM = 0.9766, but also captures the characteristics of natural images very well.

**Example 5.2.** (ordered pair  $(p, q)$ ). In this example, we discuss the role that the ordered pair  $(p, q)$  plays in the model (2.1) by showing the performances of various  $(p, q)$  for the “Lena” image with  $\sigma = 15$ . We first consider  $q = 2$  and compare the image quality of different  $p$  values, namely,  $p = 0.4$ ,  $p = 0.6$  and  $p = 1$ . The  $\lambda$  and  $\gamma$  parameters for these three cases are individually tuned to achieve SNR value being close to 17.1. The results are depicted in Figs. 7(a)–(f). In a similar fashion, we also consider  $q = 1$  with  $p = 0.4$ ,  $p = 0.6$ , and  $p = 0.8$ , where the parameters are tuned to have SNR value close to 16.2. The results are shown in Fig. 7(g)–(l). We have the following findings:

- Comparing Fig. 7(b) and (c), and their corresponding contour plots Fig. 7(e) and (f), one can find that for  $q = 2$  (i.e., the adaptive  $H^1$  model) with a fixed SNR level, the case  $p = 0.6$  results in a little smoother image quality than that of  $p = 1$  so that the staircasing effect is more likely to be reduced. Similarly,  $p = 0.4$  seems has more smoothing property than  $p = 0.6$  and the resulting effect may be a little bit over-smoothed, cf. Fig. 7(a) and (d).
- On the other hand, for  $q = 1$  (i.e., the adaptive TV model) with a fixed SNR level, a smaller  $p$  exhibits a more severe staircasing effect but the contrast of the resulting image is high (comparing Fig. 7(g) with (h) and (i), and their contour plots as well). We can also find that a smaller  $p$  tends to falsely retain noises in the image. Consequently, taking  $q = 1$  seems not to be favorable in natural image denoising.

To sum up briefly, with a careful choice of  $0 < p \leq 1$ ,  $q = 1$  seems more suitable for dealing with the synthetic image denoising, while  $q = 2$  may be more favorable for the natural image denoising. This observation will be further verified in the next two examples.



**Fig. 7.** Numerical results of Example 5.2, where  $\sigma = 15$ , and the parameters  $\lambda$  and  $\gamma$  are individually tuned to have the same SNR for each ordered pair  $(p, q)$  with  $(p, q) = (0.4, 2)$  in (a) and (d),  $(p, q) = (0.6, 2)$  in (b) and (e),  $(p, q) = (1.2)$  in (c) and (f),  $(p, q) = (0.4, 1)$  in (g) and (j),  $(p, q) = (0.6, 1)$  in (h) and (k), and  $(p, q) = (0.8, 1)$  in (i) and (l).

**Example 5.3** (natural images). In this example, we are going to show that our approach can increase more smoothness for flat regions without losing a large value of SNR. We consider two different ways to achieve the goal as follows:

- The first way is to consider an originally adaptive regularization model. As we have mentioned in Remark 2.3, our algorithm can also be applied to implement adaptive regularization models easily. For example, we consider the *difference curvature*  $D$  proposed in [11] as the detector for determining suitable model order,

$$p(D) := 2 - \sqrt{D}, \quad \bar{D} := \frac{D}{D_{max}} \quad \text{and} \quad D := \left| |u_{NN}| - |u_{TT}| \right|,$$

where

$$u_{NN} = \frac{u_x^2 u_{xx} + 2u_x u_y u_{xy} + u_y^2 u_{yy}}{u_x^2 + u_y^2} \quad \text{and}$$

$$u_{TT} = \frac{u_y^2 u_{xx} - 2u_x u_y u_{xy} + u_x^2 u_{yy}}{u_x^2 + u_y^2}$$

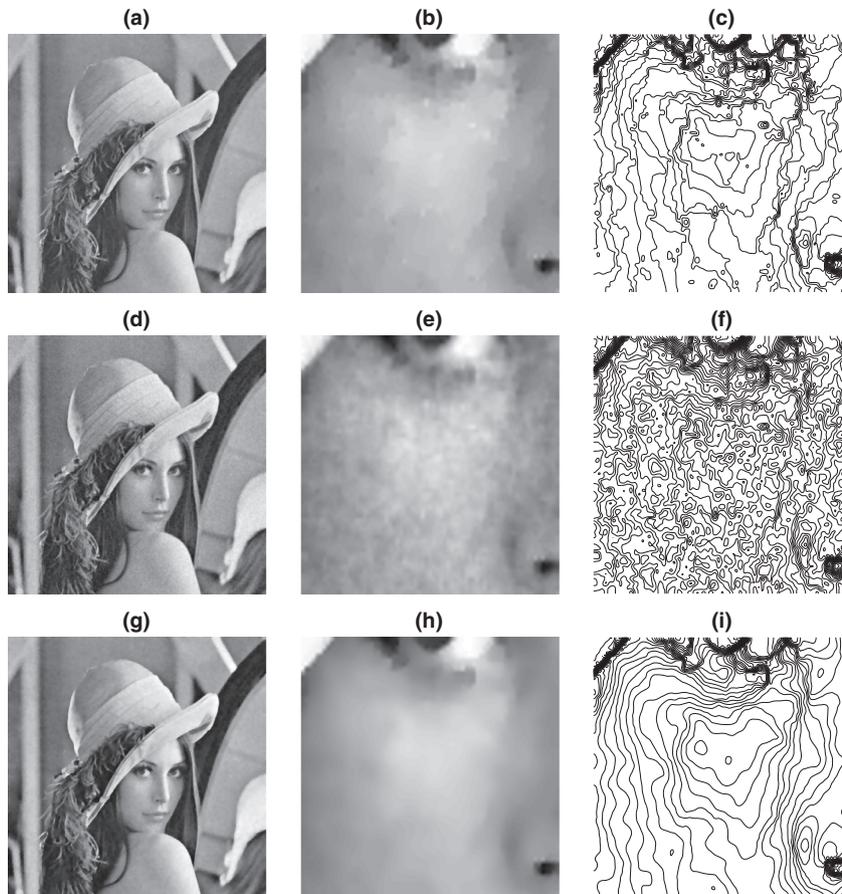
represent the second derivatives in the direction of  $N$  and  $T$ , respectively. The energy functional of the originally adaptive model is given by

$$E_{ATV} := \int_{\Omega} \left( |\nabla u|^{p(D)} + \frac{\lambda}{2} (u - f)^2 \right) dx. \tag{5.6}$$

We can apply the proposed new model (2.1) with  $q = 2$  to the above energy functional. This implies that

$$d_{i,j}^{n+1} = \frac{\gamma |\nabla u^{n+1}|_{i,j}^{2-p(D)}}{1 + \gamma |\nabla u^{n+1}|_{i,j}^{2-p(D)}} (\nabla u^{n+1} - b^n)_{i,j}$$

in the split Bregman-type algorithm (3.8).



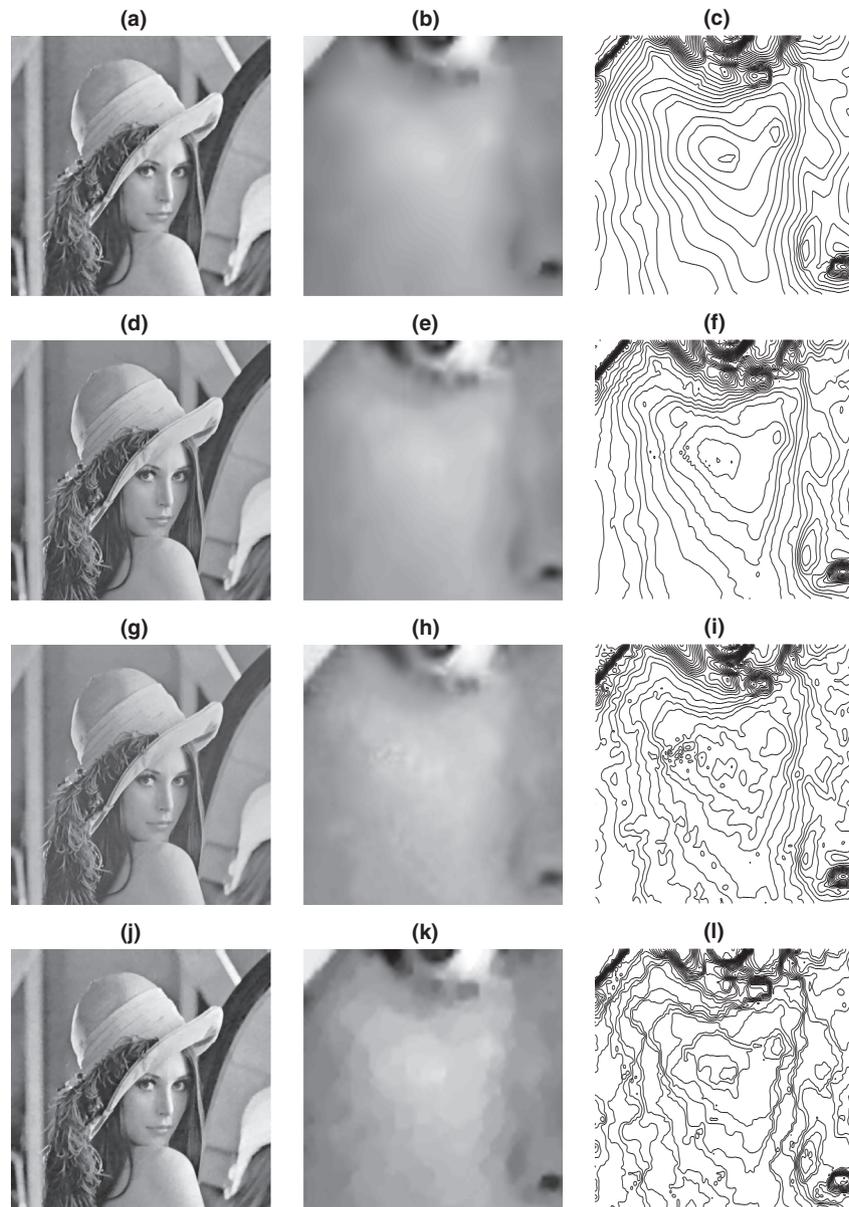
**Fig. 8.** Numerical results of [Example 5.3](#) (“Lena”), where the noise level is  $\sigma = 15$ . The best denoising parameter is  $\lambda = 0.085$  for TV-SB,  $\lambda = 0.45$ ,  $\gamma = 5/255$  for the adaptive  $p(D)$ , and  $\lambda = 0.015$ ,  $\gamma = 0.5/255$  for  $(p, q) = (0.6, 2)$  in our model. The denoising results are: (a), (b), (c) for TV-SB, (d), (e), (f) for the adaptive  $p(D)$ , and (g), (h), (i) for  $(p, q) = (0.6, 2)$ . The corresponding SNR, SSIM and iteration number are (17.78, 0.9696, 13) for TV-SB, (17.02, 0.9627, 21) for adaptive  $p(D)$ , and (17.13, 0.9660, 21) for our model, respectively.

**Table 2**  
Index values of present algorithm with various  $(p, q)$  and TV-SB ( $p = 1$ ) for [Example 5.4](#).

Image	$\sigma$	Order ( $p, q$ )	SNR	SNR	SSIM	SSIM	# Iter.	# Iter.
			TV-SB	Present	TV-SB	Present	TV-SB	Present
Two-dots	15	(0.6, 1)	15.16	19.68	0.9988	0.9994	90	106
		(0.6, 2)		19.82		0.9994		103
		(0.4, 1)		19.92		0.9993		108
		(0.4, 2)		20.19		0.9993		108
	20	(0.4, 2)	13.02	18.37	0.9983	0.9989	94	177
	25	(0.4, 2)	11.79	17.76	0.9984	0.9988	102	224
Checkerboard	15	(0.6, 1)	31.28	37.32	0.9961	0.9988	14	26
		(0.6, 2)		36.80		0.9987		24
		(0.4, 1)		37.08		0.9988		32
		(0.4, 2)		36.62		0.9988		29
	20	(0.6, 1)	29.27	35.20	0.9948	0.9982	17	35
	25	(0.6, 1)	27.86	33.39	0.9934	0.9969	20	47
Phantom	15	(0.6, 1)	26.26	31.96	0.9964	0.9988	19	34
		(0.6, 2)		31.46		0.9987		31
		(0.4, 1)		32.85		0.9990		50
		(0.4, 2)		32.45		0.9990		45
	20	(0.4, 1)	24.40	29.54	0.9954	0.9980	22	76
	25	(0.4, 1)	22.98	27.88	0.9943	0.9972	26	89

- As we have observed in [Example 5.2](#), a possible way to enhance the smoothness of flat regions in a natural image is to consider  $0 < p < 1$  and  $q = 2$ . Compared with the first way, we don't need to design a function  $p$  varying with pixels, but have to choose a suitable parameter  $\gamma$ . We find that for flat regions, the smaller  $p$ , the smoother the restoration. In this example, we take  $p = 0.6$ .

The test image is “Lena”, which is contaminated with Gaussian noise ( $\sigma = 15$ ). The denoising parameter for best SNR value is  $\lambda = 0.45$  and  $\gamma = 5/255$  for adaptive  $p(D)$ , and  $\lambda = 0.015$  and  $\gamma = 0.5/255$  for  $(p, q) = (0.6, 2)$  in the proposed model. The denoising result is shown in [Fig. 8](#). It can be seen that the staircasing effect is significantly reduced by both models, making the image more natural; however, the SNR, SSIM will drop a little bit compared to



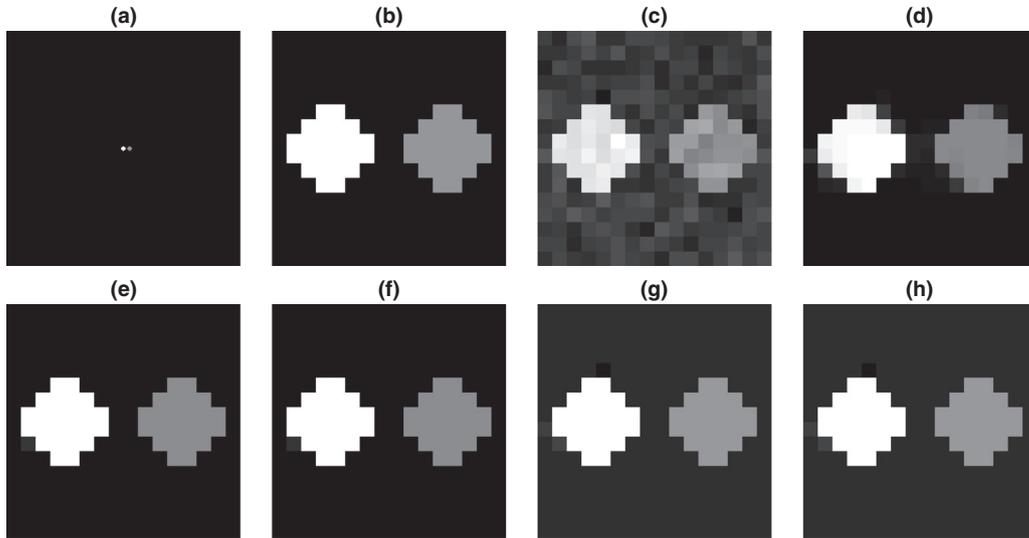
**Fig. 9.** Numerical results of Example 5.3 (“Lena”), where the experimental setting is the same as that in Fig. 8. The denoising results are: (a), (b), (c) for TGV, (d), (e), (f) for BM3D, (g), (h), (i) for NLD, and (j), (k), (l) for LOT. The corresponding SNR and SSIM are (17.75, 0.9703) for TGV, (19.73, 0.9797) for BM3D, (18.66, 0.9745) for NLD, and (17.62, 0.9689) for LOT.

those of the TV model. More specifically, we have  $(\text{SNR}, \text{SSIM}) = (17.02, 0.9627)$  in the first way,  $(17.13, 0.9660)$  in the second way, and  $(17.78, 0.9696)$  in the method of TV-SB. The iteration numbers of the TV-SB and our algorithm are 13 and 21, respectively.

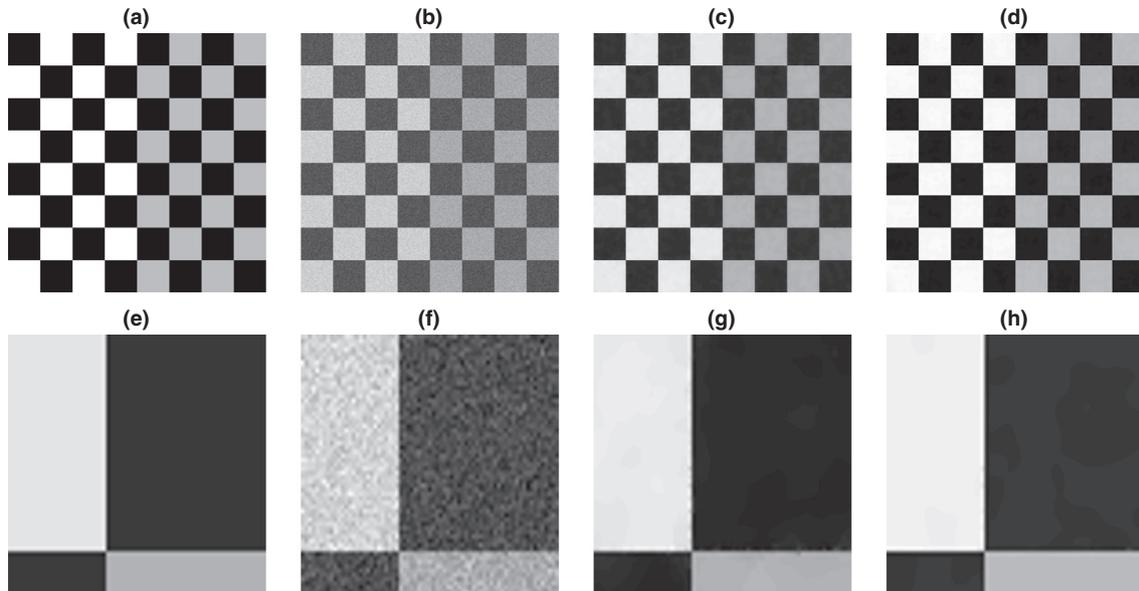
In addition to the TV-SB method, we give some other methods for the numerical comparison, namely, the TGV model in [35], the BM3D approach in [14], the NLD approach in [17], and the variant [28] of LOT method [24]. Under the same experimental condition described above, the numerical results for the four methods are shown in Fig. 9. The indices  $(\text{SNR}, \text{SSIM})$  are  $(17.75, 0.9703)$  for TGV,  $(19.73, 0.9797)$  for BM3D,  $(18.66, 0.9745)$  for NLD, and  $(17.62, 0.9689)$  for LOT. All the indices are satisfactory, but it can be seen from the zoomed-in face of “Lena” that the results of TGV and BM3D are over-smoothed, and the results of NLD and LOT still have some unnatural artifacts. We remark that the BM3D and NLD are non-local methods while the TV-SB, TGV, LOT and the

present methods are local ones. Non-local methods usually have higher indices than local ones but their computational complexity are also high. As a non-local method, BM3D has the highest indices among the all; however, the exact performance still depends on the estimation of the noise level, which is directly set to the true noise level in this paper for convenience. In contrast to the other non-local methods discussed in this example, the newly proposed method compromises between the indices and the smoothness to have a more natural image quality.

**Example 5.4** (synthetic images). In this example, we study the effectiveness of our method for synthetic images. Here we fix the penalty parameter  $\gamma = 5/255$ , and the parameter  $\lambda$  is always tuned to have best SNR value. We consider four cases with  $p = 0.6, 0.4$  and  $q = 1, 2$  for the three synthetic images. The numerical results are reported in Table 2.



**Fig. 10.** Numerical results of Example 5.4 (“Two-dots”), where the noise level is  $\sigma = 15$  and the denoising parameter  $\lambda$  is tuned to have best SNR value. (a) the original image; (b) the locally zoomed-in; (c) the noisy version of locally zoomed-in; (d) the denoising result by TV-SB; (e), (f), (g), (h) are the denoising results by present algorithm with  $(p, q) = (0.6, 1)$ ,  $(0.6, 2)$ ,  $(0.4, 1)$ ,  $(0.4, 2)$ , respectively. The (SNR, SSIM) of (d)–(h) are (15.16, 0.9988), (19.68, 0.9994), (19.82, 0.9994), (19.92, 0.9993), and (20.19, 0.9993), respectively.

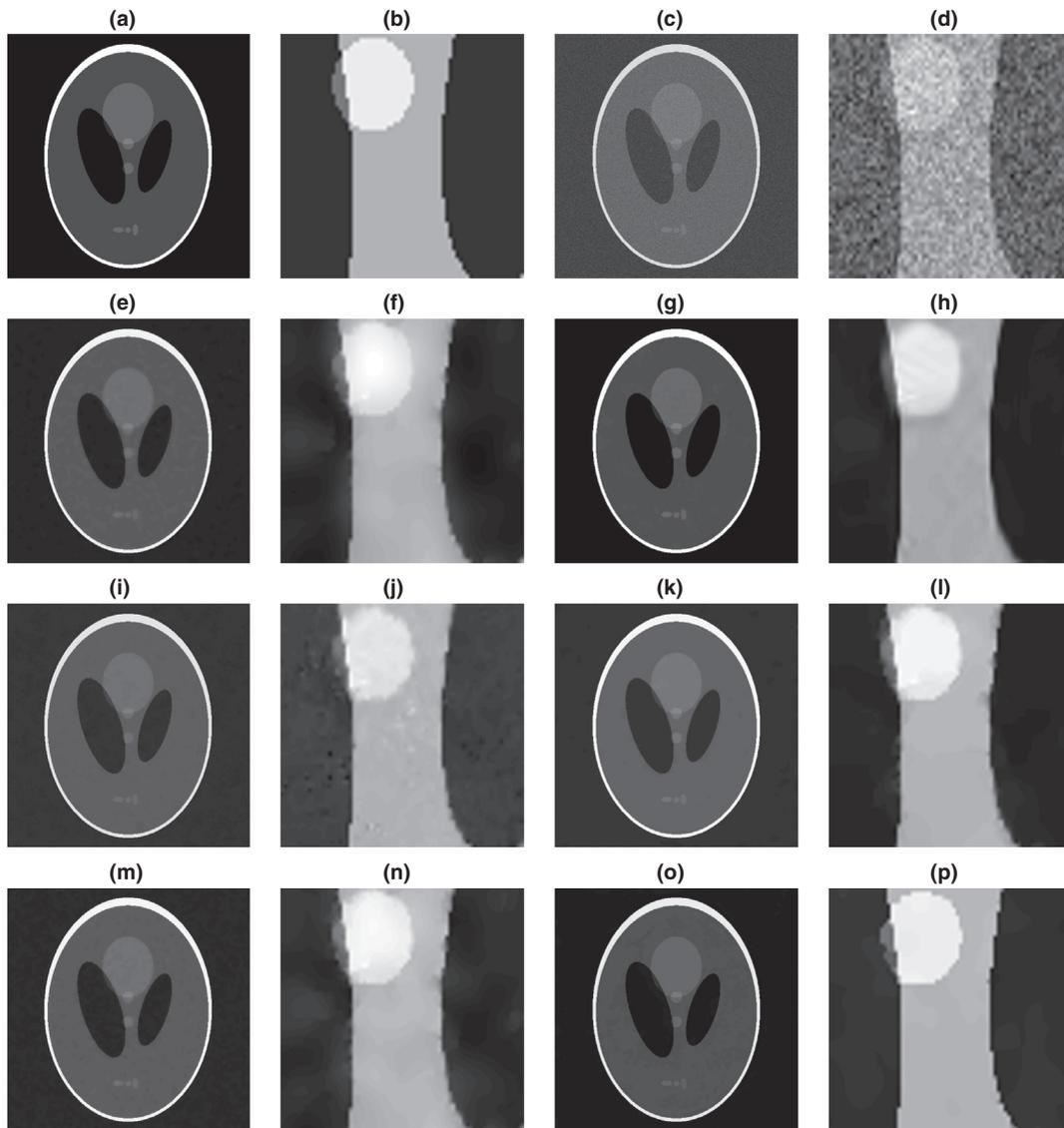


**Fig. 11.** Numerical results of Example 5.4 (“Checkerboard”), where the noise level is  $\sigma = 25$  and the denoising parameter  $\lambda$  is tuned to have best SNR value. (a), (e) are the global, local part of the original image; (b), (f) correspond to the noisy version; (c), (g) are the denoising results by TV-SB; (d), (h) are the denoising results by present algorithm with  $(p, q) = (0.6, 1)$ . The corresponding SNR, SSIM, iteration number are (27.86, 0.9934, 20) for TV-SB and (33.39, 0.9969, 47) for present algorithm.

- From Table 2, we can find that present algorithm exhibits a good performance for denoising synthetic images. The SNR value of the present approach is far greater than that of the TV-SB, but with more iterations. We note that only the pair  $(p, q)$  of best performance for  $\sigma = 15$  is employed for the cases of  $\sigma = 20, 25$ , and other pairs have very similar results both in performance indices and iteration numbers. We omit the details.
- Consider the test image “Two-dots”, which is contaminated with Gaussian noise ( $\sigma = 15$ ). The denoising result is shown in Fig. 10, where one can find that the performance of our method is much better than that of TV-SB. Although, there are winners and losers in the four cases, they all perfectly reconstruct the two small dots. In general,  $q = 1$  has better edge-preserving

ability than  $q = 2$  since  $q = 1$  amounts to the adaptive TV model in (2.7) while  $q = 2$  amounts to the adaptive  $H^1$  model. However, for this specific image,  $q = 2$  has larger SNR values than  $q = 1$  since there are few edges in the image, only around the two small dots.

- The setting of the test image “Checkerboard” is same with the “Two-dots” image. The numerical result is shown in Fig. 11. Comparing Fig. 11(g) and (h), one can find that our method has distinguished edge-preserving ability than TV-SB, where the borderline between any two areas of different intensity is perfectly recovered by our method.
- We also compare our method with the TGV, BM3D, NLD, and LOT methods using the “Phantom” image. The numerical result



**Fig. 12.** Numerical results of Example 5.4 (“Phantom”), where the experimental setting is the same as that in Fig. 10. (a), (b) are the global, local part of the original image; (c), (d) correspond to the noisy version. The denoising results are: (e), (f) for TGV, (g), (h) for BM3D, (i), (j) for NLD, (k), (l) for LOT, (m), (n) for TV-SB, and (o), (p) for the present algorithm with  $(p, q) = (0.4, 1)$ . The corresponding SNR and SSIM are (25.37, 0.9960) for TGV, (30.62, 0.9983) for BM3D, (27.92, 0.9962) for NLD, (27.44, 0.9982) for LOT, (26.26, 0.9964) for TV-SB, and (32.85, 0.9990) for the present algorithm.

is shown in Fig. 12. The present method seems to have both the highest indices and the most satisfactory result.

The better performance shows that our model seems more effective than many typical methods for restoring synthetic images.

## 6. Summary and conclusions

In this paper, we have developed a simple regularization model with adaptive diffusivity for variational image denoising. The model is motivated by approximating the Euler-Lagrange equation of the usual  $p$ th-order regularization with  $0 < p \leq 1$ . This model can integrate all the usual lower-order models into a single unified framework through a regularization controller which is introduced to adaptively adjust the diffusivity from pixel to pixel according to the magnitude of image gradient. A split Bregman-type iterative scheme is constructed for efficiently solving the new model. Moreover, a convergence analysis of the iterative scheme is established. Numerical examples show that the proposed model and algorithm are able (i) to implement the usual adaptive regularization

models with pixel-dependent power  $1 \leq p \leq 2$  for reducing staircasing effect, (ii) to have a similar edge-preserving performance to TV when  $p = 1$  while additionally putting more smoothness into flat regions, (iii) to outperform TV when  $0 < p < 1$  for denoising synthetic images, but with more iterations. Moreover, numerical results obtained in this paper also show that the performance of the proposed model is comparable to several typical methods such as the TGV, BM3D, NLD, and LOT for restoring natural images, and it seems to be better than the others for restoring synthetic images.

The most distinguished features of the proposed approach are twofold. Firstly, one single algorithm can efficiently handle several types of regularization models for different purposes. Secondly, in addition to the edge preserving ability just like the TV approach, our model puts extra smoothness into flat regions to enhance the image quality. On the other hand, the disadvantage of this approach may be that more iterations will be needed for the convergence of the split Bregman-type iterative scheme when  $0 < p < 1$ .

We remark that the magnitude of image gradient in this paper is considered in  $\ell^2$ -norm, which leads to isotropic diffusions.

For anisotropic diffusions, this work may be generalized to the  $\ell^1$ -norm case for better performance. Finally, the proposed approach can be employed for image deblurring as well.

## Acknowledgments

The authors would like to thank an anonymous referee for his/her valuable comments and suggestions that led to a substantial improvement of the original manuscript. This work was partially supported by the Ministry of Science and Technology of Taiwan under the grants MOST 104-2811-M-008-049 and MOST 106-2115-M-005-005-MY2 (Po-Wen Hsieh), MOST 103-2115-M-008-009-MY3 and MOST 106-2115-M-008-014-MY2 (Suh-Yuh Yang).

## References

- [1] G. Aubert, P. Kornprobst, *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*, Springer Verlag, New York, 2002.
- [2] J. Benesty, J. Cheng, Y.A. Huang, Study of widely linear wiener filter for noise reduction, in: *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2010, pp. 205–208.
- [3] A. Buades, B. Coll, J.M. Morel, The staircasing effect in neighborhood filters and its solution, *IEEE Trans. Image Process.* 15 (2006) 1499–1505.
- [4] P. Blomgren, T.F. Chan, P. Mulet, C.K. Wong, Total variation image restoration: numerical methods and extensions, in: *Proceedings of IEEE International Conference on Image Processing (ICIP)*, vol. 3, 1997, pp. 384–387.
- [5] M.J. Black, G. Sapiro, D.H. Marimont, D. Heeger, Robust anisotropic diffusion, *IEEE Trans. Image Process.* 7 (1998) 421–432.
- [6] K. Bredies, K. Kunisch, T. Pock, Total generalized variation, *SIAM J. Imag. Sci.* 3 (2010) 492–526.
- [7] A. Chambolle, P.L. Lions, Image recovery via total variation minimization and related problems, *Numerische Mathematik* 76 (1997) 167–188.
- [8] R.H. Chan, S. Setzer, G. Steidl, Inpainting by flexible haar-wavelet shrinkage, *SIAM J. Imag. Sci.* 1 (2008) 273–293.
- [9] R. Chartrand, Nonconvex regularization for shape preservation, in: *Proceedings of IEEE International Conference on Image Processing (ICIP)*, 2007, pp. 293–296.
- [10] S.S. Chen, D.L. Donoho, M.A. Saunders, Atomic decomposition by basis pursuit, *SIAM Rev.* 43 (2001) 129–159.
- [11] Q. Chen, P. Montesinos, Q.S. Sun, P.A. Heng, Adaptive total variation denoising based on difference curvature, *Image Vis. Comput.* 28 (2010) 298–306.
- [12] E.J. Candes, M.B. Wakin, S.P. Boyd, Enhancing sparsity by reweighted  $\ell_1$  minimization, *J. Fourier Anal. Appl.* 14 (2008) 877–905.
- [13] S.G. Chang, B. Yu, M. Vetterli, Adaptive wavelet thresholding for image denoising and compression, *IEEE Trans. Image Process.* 9 (2000) 1532–1546.
- [14] K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, Image denoising with block-matching and 3d filtering, in: *Proc. SPIE 6064, Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning*, vol. 6064, 2006. 606414-1-12.
- [15] D.L. Donoho, I.M. Johnstone, Adapting to unknown smoothness via wavelet shrinkage, *J. Am. Stat. Assoc.* 90 (1995) 1200–1224.
- [16] T. Goldstein, S. Osher, The split bregman method for  $l^1$  regularized problems, *SIAM J. Imag. Sci.* 2 (2009) 323–343.
- [17] G. Gilboa, S. Osher, Nonlocal operators with applications to image processing, *Multiscale Model. Simul.* 7 (2008) 1005–1028.
- [18] A.K. Jain, *Fundamentals of Digital Image Processing*, Prentice-Hall, Upper Saddle River, NJ, 1989.
- [19] P. Jain, V. Tyagi, A survey of edge-preserving image denoising methods, *Inf. Syst. Frontiers* 18 (2016) 159–170.
- [20] Y.J. Lee, S. Lee, J. Yoon, A framework for moving least squares method with total variation minimizing regularization, *J. Math. Imag. Vis.* 48 (2014) 566–582.
- [21] F. Li, L. Pi, T. Zeng, Explicit coherence enhancing filter with spatial adaptive elliptical kernel, *IEEE Signal Process. Lett.* 19 (2012) 555–558.
- [22] C. Louchet, L. Moisan, Total variation as a local filter, *SIAM J. Imag. Sci.* 4 (2011) 651–694.
- [23] Y. Lou, T. Zeng, S. Osher, J. Xin, A weighted difference of anisotropic and isotropic total variation model for image processing, *SIAM J. Imag. Sci.* 8 (2015) 1798–1823.
- [24] M. Lysaker, S. Osher, X.C. Tai, Noise removal using smoothed normals and surface fitting, *IEEE Trans. Image Process.* 13 (2004) 1345–1357.
- [25] R. Malladi, J.A. Sethian, Image processing via level set curvature flow, *Proc. Natl. Acad. Sci.* 92 (1995) 7046–7050.
- [26] H. Oh, *Bayesian ensemble learning for image denoising*, 2013. arXiv:1308.1374.
- [27] S. Osher, *Level set methods, Geometric Level Set Methods in Imaging, Vision, and Graphics*, Springer Verlag, New York, 2003.
- [28] Z.F. Pang, Y.F. Yang, A two-step model for image denoising using a duality strategy and surface fitting, *J. Comput. Appl. Math.* 235 (2010) 82–90.
- [29] P. Perona, J. Malik, Scale-space and edge detection using anisotropic diffusion, *IEEE Trans. Pattern Anal. Mach. Intel.* 12 (1990) 629–639.
- [30] Z. Ren, C. He, Q. Zhang, Fractional order total variation regularization for image super-resolution, *Signal Process.* 93 (2013) 2408–2421.
- [31] L.I. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, *Physica D* 60 (1992) 259–268.
- [32] J. Salvador, M. Borsum, A. Kochale, A bayesian approach for natural image denoising, in: *Proceedings of IEEE International Conference on Image Processing (ICIP)*, 2013, pp. 1095–1099.
- [33] L. Sendur, I.W. Selesnick, Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency, *IEEE Trans. Signal Process.* 50 (2002) 2744–2756.
- [34] M. Shi, T. Han, S. Liu, Total variation image restoration using hyper-laplacian prior with overlapping group sparsity, *Signal Process.* 126 (2016) 65–76.
- [35] K. Shirai, M. Okuda, FFT based solution for multivariable l2 equations using KKT system via FFT and efficient pixel-wise inverse calculation, in: *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2014, pp. 2629–2633.
- [36] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, *IEEE Trans. Image Process.* 13 (2004) 600–612.
- [37] J. Weickert, *Anisotropic Diffusion in Image Processing*, Teubner, Stuttgart, 1998.
- [38] J. Weickert, B.T.H. Romeny, M.A. Viergever, Efficient and reliable schemes for nonlinear diffusion filtering, *IEEE Trans. Image Process.* 7 (1998) 398–410.
- [39] C. Wu, X.C. Tai, Augmented lagrangian method, dual methods, and split bregman iteration for ROF, vectorial TV, and high order models, *SIAM J. Imag. Sci.* 3 (2010) 300–339.
- [40] J. Xu, Y. Hao, H. Song, A modified LOT model for image denoising, *Multimed. Tools Appl.* 76 (2017) 8131–8144.
- [41] J. Yan, W.S. Lu, Image denoising by generalized total variation regularization and least squares fidelity, *Multidimension. Syst. Signal Process.* 26 (2015) 243–266.
- [42] W. Zhang, Y. Cao, R. Zhang, Y. Wang, Image denoising using total variation model guided by steerable filter, *Math. Probl. Eng.* (2014). Article ID 423761.