Optimization Methods & Software, 2014
Vol. 29, No. 2, 274–285, http://dx.doi.org/10.1080/10556788.2013.788650

Taylor & Francis
Taylor & Francis Group

A relaxed fixed point method for a mean curvature-based denoising model

Fenlin Yang[a,b], Ke Chen[c], Bo Yu[b]* and Donghui Fang[a]

[a]*College of Mathematics and Statistics, Jishou University, Jishou, Hunan 416000, People's Republic of China;* [b]*School of Mathematical Sciences, Dalian University of Technology, Dalian, Liaoning 116024, People's Republic of China;* [c]*Centre for Mathematical Imaging Techniques and Department of Mathematical Sciences, The University of Liverpool, Liverpool L69 7ZL, UK*

(*Received 22 December 2011; final version received 19 March 2013*)

Mean curvature-based energy minimization denoising model by Zhu and Chan offers one approach for restoring both smooth (no edges) and non-smooth (with edges) images. The resulting fourth-order partial differential equations arising from minimization of this model is non-trivial to solve due to appearance of a high nonlinearity and stiffness term, because simple alternative methods such as the fixed point method and the primal dual method do not work. In this paper, we first present a relaxed fixed point method for solving such equations and further to combine with a homotopy algorithm to achieve fast convergence. Numerical experiments show that our method is able to maintain all important information in the image, and at the same time to filter out noise.

Keywords: image denoising; mean curvature model; relaxed fixed point method; homotopy method

1. Introduction

Image denoising is a task to extract a 'quality' image u from the observed image f with η being some noisy image. The common degradation model is

$$f = u + \eta. \tag{1}$$

Here u, f and η are functions defined on a bounded convex region Ω of $\mathbb{R}^d$ ($d = 2$ or 3, in this paper, the two-dimensional case is considered and we will assume Ω is a square in $\mathbb{R}^2$).

The total variation (TV) model proposed by Rudin *et al*. [20] is a well-known model to obtain u, the energy functional represented as

$$\min_u \left\{ E(u) = \alpha \int_\Omega |\nabla u| \, \mathrm{d}x \, \mathrm{d}y + \frac{1}{2}\|u - f\|^2 \right\}. \tag{2}$$

Here $|\cdot|$ is the Euclidean norm in $\mathbb{R}^2$ and $\|\cdot\|$ is the norm in $\mathrm{L}^2(\Omega)$. The Euler–Lagrange equation for the TV model is the second-order equation

$$-\alpha \nabla \cdot \frac{\nabla u}{|\nabla u|_\beta} + (u - f) = 0, \tag{3}$$

*Corresponding author. Email: yubo@dlut.edu.cn

with homogeneous Neumann boundary condition $\partial u/\partial \vec{n} = 0$ along $\partial\Omega$. Here $|\nabla u|_\beta = \sqrt{|\nabla u|^2 + \beta}(\beta > 0)$ and $\vec{n}$ is the normal vector. Much theoretical work on this TV model has been done in more recent years [8,10,20,22,25]. The model can produce desirable results from removing noise while preserving edges for non-smooth images. But for smooth images, the TV model suffers from staircasing effect. To remedy these unfavourable property, some researchers proposed a modified models see [5,11–13,15,17–19,21]; others turned to higher order models [2–4,7,9,16,26].

Mean curvature-based denoising model proposed by Zhu and Chan [26] is another model which can restore effectively both blocky images (of piecewise constant intensities) and smooth images (with no clear jumps), the energy functional is

$$\min_u \left\{ J(u) = \alpha \int_\Omega \Phi(\kappa(u))\,\mathrm{d}x\,\mathrm{d}y + \frac{1}{2}\|u - f\|^2 \right\}. \tag{4}$$

The functional Φ is defined as $\Phi(\kappa(u)) = |\kappa(u)|$, $\Phi(\kappa(u)) = \kappa(u)^2$ or a combination of both, below we shall take $\Phi(\kappa(u)) = \kappa(u)^2$. Here $\kappa(u)$ is the mean curvature of the image defined by

$$\kappa(u) = \nabla \cdot \frac{\nabla u}{|\nabla u|}. \tag{5}$$

To avoid non-smooth, we replace $|\nabla u|$ by $|\nabla u|_\beta$. Thus, we have

$$\kappa_\beta(u) = \nabla \cdot \frac{\nabla u}{|\nabla u|_\beta}, \tag{6}$$

and the energy functional (4) becomes

$$\min_u \left\{ J_\beta(u) = \alpha \int_\Omega \Phi(\kappa_\beta(u))\,\mathrm{d}x\,\mathrm{d}y + \frac{1}{2}\|u - f\|^2 \right\}. \tag{7}$$

The corresponding Euler–Lagrange partial differential equation (PDE) for (7) is, for $(x, y) \in \Omega$,

$$g_\beta(u) = \alpha\nabla \cdot \left(\frac{1}{|\nabla u|_\beta} \left(I_2 - \frac{\nabla u \nabla u^{\mathrm{T}}}{|\nabla u|_\beta^2} \right) \nabla\Phi'(\kappa_\beta(u)) \right) + (u - f) = 0, \tag{8}$$

with homogeneous Neumann boundary condition $\partial u/\partial \vec{n} = 0$ and $\kappa_\beta(u) = 0$ along $\partial\Omega$. Here $I_2 \in \mathbb{R}^{2\times 2}$ is an identity operator, and $\Phi'(\kappa_\beta(u))$ is the derivative of $\Phi(\kappa_\beta(u))$. Since $\Phi(\kappa_\beta(u)) = \kappa_\beta(u)^2$, so $\Phi'(\kappa_\beta(u)) = 2\kappa_\beta(u)$ and $\nabla\Phi'(\kappa_\beta(u)) = 2\nabla\kappa_\beta(u)$. Due to the appearance of a high nonlinearity and stiffness term, simple alternative methods (which worked for the TV model) such as fixed point (FP) method [22] and primal dual method [8] do not work for this newer model, as shown in [3].

Only two methods have been proposed to solve Equation (8).

(1) *Gradient descent method*. As used in [26], instead of the elliptic PDE, a parabolic PDE with time as an evolution parameter is solved by the gradient descent method

$$u_t = -\alpha\nabla \cdot \left(\frac{1}{|\nabla u|_\beta} \left(I_2 - \frac{\nabla u \nabla u^{\mathrm{T}}}{|\nabla u|_\beta^2} \right) \nabla\Phi'(\kappa_\beta(u)) \right) - (u - f), \tag{9}$$

with $u(x, y, 0) = f$. This method is preferred in many situations for its simplicity and fast initial convergence, but its overall convergence is slow (or the time step must be small for stability) to reach steady state since the parabolic term is nearly singular for small gradients.

(2) *Stabilized FP method and nonlinear multigrid*. The idea of stabilized fixed point method is to split the Euler–Lagrange equation (8) in two parts (convex and non-convex) and treat the convex part implicitly and the non-convex part explicitly, after adding suitable stabilizing terms. Thus, the stabilized FP method of [3] takes the form

$$u^{k+1} - \gamma\nabla\cdot\frac{\nabla u^{k+1}}{|\nabla u^k|_\beta} - \alpha\nabla\cdot\left(\frac{\nabla u^k\cdot\nabla\Phi'(\kappa_\beta(u^k))}{|\nabla u^k|_\beta^3}\nabla u^{k+1}\right)$$
$$= -\gamma\nabla\cdot\frac{\nabla u^k}{|\nabla u^k|_\beta} - \alpha\nabla\cdot\frac{\nabla\Phi'(\kappa_\beta(u^k))}{|\nabla u^k|_\beta} + f. \tag{10}$$

Further an efficient nonlinear multigrid for (8) is developed using this stabilized FP method as a smoother. Although stabilization helps to derive a convergent FP method, the smoothing parameter β in (10) cannot be small, and was typically chosen to be, say, 10^{-2} or larger.

2. FP curvature method

Before our relaxed FP method is presented, we introduce an FP curvature method to solve the Euler–Lagrange equation (8) directly. The idea is motivated by the observation that the fourth-order nonlinear operator of Euler–Lagrange equation (8) can be considered as a second-order nonlinear operator compounding another second-order nonlinear operator, therefore, (8) can be rewritten as

$$g_\beta(u) = 2\alpha\left(\nabla\cdot\left(\frac{1}{|\nabla u|_\beta}\left(I_2 - \frac{\nabla u\nabla u^{\mathrm{T}}}{|\nabla u|_\beta^2}\right)\nabla\right)\right)\left(\nabla\cdot\frac{\nabla u}{|\nabla u|_\beta}\right) + (u-f)$$
$$= 2\alpha M_\beta^N(u)\left(-\nabla\cdot\frac{\nabla u}{|\nabla u|_\beta}\right) + (u-f) = 0. \tag{11}$$

The FP equation is obtained by freezing the second-order semi-definite nonlinear operator

$$M_\beta^N(u) = -\nabla\cdot\left(\frac{1}{|\nabla u|_\beta}\left(I_2 - \frac{\nabla u\nabla u^{\mathrm{T}}}{|\nabla u|_\beta^2}\right)\nabla\right)$$

at a known iterate $u^{(l,0)}$, and the formulation is

$$g_\beta^{l,0} = 2\alpha M_\beta^{\mathrm{N}}(u^{(l,0)})\left(-\nabla\cdot\frac{\nabla u}{|\nabla u|_\beta}\right) + (u-f) = 0. \tag{12}$$

Observed that the nonlinearity of the FP equation (12) is formed from multiplying a second-order semi-definite linear operator $2\alpha M_\beta^N(u^{(l,0)})$ to $-\nabla\cdot(\nabla u/|\nabla u|_\beta)$. Clearly Equation (12) is similar to the TV equation (3), and it is also a fourth-order nonlinear PDE, but the nonlinearity is reduced, so smaller β (e.g. $\beta < 10^{-6}$) is allowed. Many methods for TV equation (3) can be used to solve (12). The procedure of FP curvature method for (11) contains outer iteration (FP iteration) and inner iteration (solving FP equation). The whole procedure consists of a succession of the following two steps.

(1) Solve the FP equation (12) to get its solution by $u^{(l,*)}$ and set $u^{(l+1,0)} = u^{(l,*)}$.
(2) Set $l := l + 1$, use $u^{(l,0)}$ to update the second-order semi-definite linear operator $M_\beta^N(u)$, and obtain a new FP equation (12).

Here we use $u^{(0,0)} = f$ at the very first FP equation, and the succession terminates by some tolerance or the maximum number of iteration.

3. The relaxed FP method

Due to high singularity of FP equation, solving (12) with small β is very time consuming. The idea of relaxed FP method is motivated by reducing the singularity of FP equation (12). Observed that if the second term of $M_\beta^N(u^{(l,0)})$ is dropped, the singularity is reduced and the remain term

$$M_\beta^{\mathrm{FP}}(u^{(l,0)}) = -\nabla \cdot \left(\frac{1}{|\nabla u^{(l,0)}|_\beta} \nabla \right)$$

still is a second-order semi-definite linear operator. Therefore, we get a relaxed FP equation

$$\tilde{g}_\beta^{l,0}(u) = 2\alpha M_\beta^{\mathrm{FP}}(u^{(l,0)}) \left(-\nabla \cdot \frac{\nabla u}{|\nabla u|_\beta} \right) + (u - f) = 0. \tag{13}$$

The above equation has the same structure as FP equation (12), but the singularity is lower. So it can be solved by the TV methods, such as the gradient descent method, the FP method and the primal dual method and so on. Our recommendation is the following primal dual method.

$$\begin{aligned} &- 2\alpha M_\beta^{\mathrm{FP}}(u^{(l,0)}) \nabla \cdot w + (u - f) = 0, \\ &h(u, w) = w|\nabla u|_\beta - \nabla u = 0. \end{aligned} \tag{14}$$

The linearization of the above (u, w) system is

$$\begin{bmatrix} -2\alpha M_\beta^{\mathrm{FP}}(u^{(l,0)}) \nabla \cdot & 1 \\ |\nabla u|_\beta I_2 & -\left(I_2 - \dfrac{w(\nabla u)^{\mathrm{T}}}{|\nabla u|_\beta} \right) \nabla \end{bmatrix} \begin{bmatrix} \delta w \\ \delta u \end{bmatrix} = - \begin{bmatrix} \tilde{g}_\beta^{l,0}(u, w) \\ h(u, w) \end{bmatrix}. \tag{15}$$

Equation (15) can be solved by first eliminating δw and solving the resulting equation for δu:

$$\left[-2\alpha M_\beta^{\mathrm{FP}}(u^{(l,0)}) \nabla \cdot \left(\frac{1}{|\nabla u|_\beta} \left(I_2 - \frac{w(\nabla u)^{\mathrm{T}}}{|\nabla u|_\beta} \right) \nabla \right) + 1 \right] \delta u = -\tilde{g}_\beta^{l,0}(u)$$

and

$$\delta w = \frac{1}{|\nabla u|_\beta} \left(I_2 - \frac{w(\nabla u)^{\mathrm{T}}}{|\nabla u|_\beta} \right) \nabla \delta u - w + \frac{\nabla u}{|\nabla u|_\beta}.$$

We simply describe how relaxed FP equation will be solved by primal dual method in preparation for a relaxed FP method.

ALGORITHM 1 $[u, k, \mathrm{flag}] \leftarrow \mathrm{RFP\_solver}(u^{(l,0)}, f, \mathrm{tol}, \mathrm{maxit}, \alpha, \beta)$

Step 1. Set $k := 0$, flag $:= 1$, *compute* $M_\beta^{\mathrm{FP}}(u^{(l,0)})$, $\tilde{g}_\beta^{l,0}(u^{(l,0)})$, *and* $\mathrm{res}_0 := \|\tilde{g}_\beta^{l,0}(u^{(l,0)})\|_2$.
Step 2. Compute update direction $(\delta u^{(l,k)}, \delta w^{(l,k)})$ *by primal dual method, set*

$$u^{(l,k+1)} = u^{(l,k)} + \delta u^{(l,k)}, \quad w^{(l,k+1)} = w^{(l,k)} + \delta w^{(l,k)}$$

and $k = k + 1$.

Step 3. Compute $\tilde{g}_\beta^{l,0}(u^{(l,k)})$*, then set*

$$\text{relres} := \frac{\|\tilde{g}_\beta^{l,0}(u^{(l,k)})\|_2}{\text{res}_0}.$$

Step 4. If relres $\leq$ *tol, then*
record the iteration, then return $u = u^{(l,k)}$ *(solution has been found).*
Else
if the maxit *iterations have been performed, then*
set $u = u^{(l,0)}$ *and* flag $= 0$.
Else return to step 2.

Just like the FP curvature method for (11), the relaxed FP method contains outer iteration (FP iteration) and inner iteration (solving relaxed FP equation).

ALGORITHM 2
$[u, \text{iter}, \text{flag}] \leftarrow \text{RFP\_method}(u^{(0,0)}, f, \text{tol}_1, \text{tol}, \text{maxit}_1, \text{maxit}, \alpha, \beta)$

Step 1. Compute $J_\beta(u^{(0,0)})$ *and set* $l := 1$ *and* iter $:= 0$.
Step 2. $[u^{(l,0)}, \text{it}, \text{flag}] \leftarrow \text{RFP\_solver}(u^{(l-1,0)}, f, \text{tol}, \text{maxit}, \alpha, \beta)$.
Compute the energy function $J_\beta(u^{(l,0)})$ *and* iter $=$ iter $+$ it.
Step 3. If $|J_\beta(u^{(l,0)}) - J_\beta(u^{(l-1,0)})| < \text{tol}_1$ *or* $l = \text{maxit}_1$*, then*
return with $u = u^{(l,0)}$.
Else set $l = l + 1$*, then return to step 2.*
Step 4. iter $=$ iter$/l$.

For the TV equation (3), we know that both the FP method and the primal dual method work well for any non-zero β, but the convergence of the iteration slows down as β decreases, moreover, the inner linear system is difficult to solve. Returning to relaxed FP equation (13), the same conclusions remain. To improve the convergence difficulty with small β, we introduced a homotopy method based on gradually reducing the parameter β. We will use this procedure to achieve fast convergence for our relaxed FP method.

4. Homotopy method

A homotopy method offers a convergent solution for a large class of nonlinear equations. As a globally convergent method, it has versatility and robustness, and it has become an important tool for solving nonlinear problems; see [1,14,23]. The basic idea of a homotopy algorithm is to construct a continuous map $H(u,t)$ with parameter t which deforms a simple function $H(u,0)$ to the given function $H(u,1)$ as t varies from 0 to 1.

To find a required solution from tracking the solution curve Γ emanating from the solution of $H(u,0)$, the homotopy algorithm requires Γ obey strict smoothness conditions. There are many continuous maps that can satisfy these conditions.

Since we hope our homotopy equation is able to reduce the level of singularity and nonlinearity for (8) when t is not near 1. We construct the homotopy map as follows:

$$H(u,t) = 2\alpha\nabla\cdot\left(\frac{1}{|\nabla u|_{\beta(t)}}\left(I_2 - \frac{\nabla u\nabla u^T}{|\nabla u|^2_{\beta(t)}}\right)\nabla\right)\left(\nabla\cdot\frac{\nabla u}{|\nabla u|_{\beta(t)}}\right) + (u - f) = 0, \tag{16}$$

where $\beta(t) = (1-t)/t^2$, $t \in (0,1]$. When $t > 0$ is small, $\beta(t)$ is large and the level of singularity and nonlinearity of $H(u,t) = 0$ is lower than (8), and so it is easier to solve. Of course, other

choices of $\beta(t)$ may be permitted as long as they ensure a positive and monotonically decreasing $\beta(t)$ such that $\beta(1) = 0$ and $\beta(t_0) > 0$ is large when $t_0 \approx 0$. Since

$$\frac{1}{|\nabla u|_{\beta(t)}} = \frac{1}{\sqrt{|\nabla u|^2 + (1-t)/t^2}} = \frac{t}{\sqrt{t^2|\nabla u|^2 + (1-t)}}.$$

To allow $t = 0$, we can rewrite Equation (16) as

$$2\alpha\nabla \cdot \left(\frac{t}{\sqrt{t^2|\nabla u|^2 + (1-t)^2}} \left(I_2 - \frac{t^2 \nabla u \nabla u^{\mathrm{T}}}{t^2|\nabla u|^2 + (1-t)^2} \right) \nabla \right) \times \left(\nabla \cdot \frac{t\nabla u}{\sqrt{t^2|\nabla u|^2 + (1-t)^2}} \right) + (u - f) = 0. \tag{17}$$

Here $H(u, 0) = u - f$ implies that our initial solution is the given image (as commonly used in denoising algorithms) and $H(u, 1) = 0$ will give the solution for (8) with $\beta = 0$.

We can solve a sequence of equations $H(u, t_k) = 0$ for adaptively increasing t_k up to 1 (in practice, we will stop at certain $t_* < 1$ near 1 such that $\beta(t_*) = \beta$, a prescribed small enough smooth parameter). The solution of $H(u, t_{k-1}) = 0$ serves as the good initial guess for iteratively solving $H(u, t_k) = 0$. The whole process is a special predictor–corrector path following procedure with a staircase predictor. It consists of the following two phases.

Predictor step. For $t_0 = 0$, the solution of $H(u, t_0) = 0$ is known, namely $u_0 = f$. After we have obtained an approximate solution u_{k-1} of $H(u, t_{k-1})$ for some $t_{k-1} \in [0, 1)$, increase t with some predictor steplength h_{k-1} to reach $t_k = t_{k-1} + h_{k-1}$ and the solution of $H(u, t_k) = 0$ is provided with the initial guess u_{k-1}.

Corrector steps. From the initial point u_{k-1}, approximately solve $H(u, t_k) = 0$ by the relaxed FP method.

As we know, the prediction phases and the correction phases mutually affect each other. The predictor steplength h is adjusted according to the performance of the corrector procedure as done below in Algorithm 3. When a corrector step terminates within prescribed steps of iter1, h is considered too small for the next predictor and is increased, when the iterations terminate over some iter2>iter1 steps and converge, h is considered too large and will be decreased, while if the iterations diverge, the predictor–corrector step is abandoned and then is restarted starting with a smaller h. The predictor–corrector path following procedure is shown as follows:

ALGORITHM 3 (Homotopy method)
$u \leftarrow \text{homotopy}(f, \text{tol}_1, \text{tol}, \text{maxit}_1, \text{maxit}, h, \alpha, \beta)$

Step 1. *Set* $k = 1$, $u_{k-1} := f$ *and* $t_{k-1} := 0$.
Step 2. *Set* $t_k := t_{k-1} + h$.
If $t_k \geq 2/(1 + \sqrt{1 + 4\beta})$, *then*
$t_k = 2/(1 + \sqrt{1 + 4\beta})$,
$[u_k, \text{iter}, \text{flag}] \leftarrow \text{RFP\_method}(u_{k-1}, f, \text{tol}_1, \text{tol}, \text{maxit}_1, \text{maxit}, \alpha, t_k)$.
If success then
return $u = u_k$ (*solution has been found*).
Else set $h = h/2$, $t_k = t_{k-1} + h(1 - t_{k-1})$.
Step 3. $[u_k, \text{iter}, \text{flag}] \leftarrow \text{RFP\_method}(u_{k-1}, f, \text{tol}_1, \text{tol}, \text{maxit}_1, \text{maxit}, \alpha, t_k)$.
If success, then
set $k = k + 1$.
If the iteration count iter *is less than* iter1, *then*
increase h *by* $h := 1.2h$.

If the iteration count iter *is more than* iter2*, then*
reduce h slightly by $h := h/1.2$.
Return to step 2.
Else reduce h by $h := h/2$.
If h is unreasonably small, then
return with an error Flag.
Else return to step 2.

5. Numerical results

In this section we test our denoising algorithms on several images, 128×128 and 256×256 pixels, with an intensity range of $[0, 255]$. the signal to noise ratio (SNR), the peak signal to noise ratio (PSNR), and the difference between a digital image and its denoised version are used to measure the quality of the restored images, and we define them by

$$\mathrm{SNR} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} u_{i,j}^2}{\sum_{i=1}^{n}\sum_{j=1}^{n}(u_{i,j}-\tilde{u}_{i,j})^2}, \quad \mathrm{PSNR} = 10\log_{10}\frac{255^2}{(1n^2)\sum_{i=1}^{n}\sum_{j=1}^{n}(u_{i,j}-\tilde{u}_{i,j})^2},$$

and

$$\mathrm{diff}(\tilde{u}) = \tilde{u} - f,$$

where u, $\tilde{u}$ and f are, respectively, the original image, the restored image and the noisy image. For ease of comparison, we shall denote by FP for FP curvature method and by RFP for relaxed FP method.

5.1 *Parameters test*

α-Dependence test. We use the 'triangular' image with random noisy as the first test image (Figure 1). Here we analyse how sensitive the performance of the FP curvature method and our relaxed FP method is when $\alpha = 20,\ 50,\ 100,\ 150$, while $\beta = 10^{-2}$ is unchanged. We can see a clear process of the changes of SNR using FP curvature method and relaxed FP method with different α in Figure 2. Although both of them improve the quality of the image for the different values of α, we see that the performance of the FP curvature method for smaller α is less efficient while the relaxed FP method is more consistently behaved. Moreover, the relaxed FP method can get a quality image within one or two outer iterations. Therefore, we can choose $\mathrm{maxit}_1 = 1$ or 2 in Algorithm 3.

β-Dependence test. After the analysis of the effect of α, we analyse how β affects the performance of FP curvature method, relaxed FP method and Algorithm 3. To give an indication of the level of the nonlinearity and singularity reduced by each method with respect to β. Here we take suitable large α for each method respectively and vary $\beta = 1, 10^{-2}, 10^{-4}, 10^{-6}$. The numerical results for both methods are shown in Table 1. Here 'iter' means the total number of the primal dual iterations, and we choose $\mathrm{maxit}_1 = 20$ and $\mathrm{maxit} = 50$. As far as small β is concerned, the number of the primal dual iterations and CPU time needed for Algorithm 3 and relaxed FP method are less than FP method, and Algorithm 3 speeds up the convergence of relaxed FP method.

5.2 *Comparisons of the TV model with Algorithm 3*

The classical TV model is known to yield satisfactory results for removing noise while preserving edges and contours of non-smooth objects. In this section, we conduct numerical experiments to

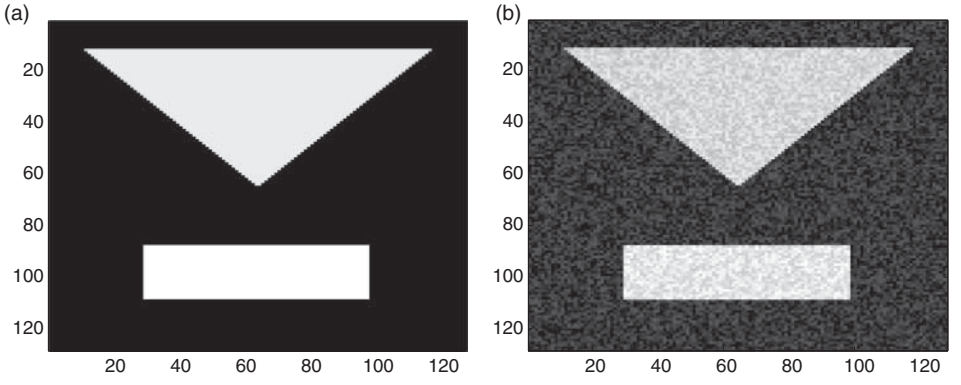

Figure 1. The original 'triangle' image (left) and noisy image (right).

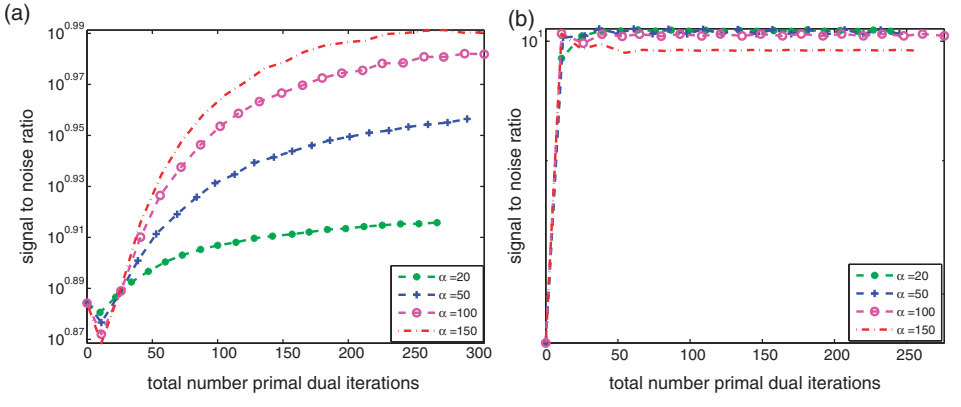

Figure 2. The history of SNR for FP curvature method (left) and relaxed FP method (right) with different α.

Table 1. The numerical results for FP method and RFP method.

β	Method	iter	PSNR	Time
1	FP method	180	24.5708	108.73
	Relaxed FP method	132	24.5884	67.01
	Algorithm 3	78	24.5877	40.59
10^{-2}	FP method	304	24.4520	187.91
	Relaxed FP method	239	24.5868	135.01
	Algorithm 3	83	24.6023	44.07
10^{-4}	FP method	452	24.4547	1176.59
	Relaxed FP method	418	24.5707	220.50
	Algorithm 3	86	24.6027	44.28
10^{-6}	FP method	639	24.4451	2928.32
	Relaxed FP method	633	24.5694	441.79
	Algorithm 3	92	24.6026	47.74

compare this method with our Algorithm 3. Test results obtained from comparing the classical TV model and Algorithm 3 using images of 'Pepper' (Figure 3) and 'Barbara' (Figure 4) with random noise are now shown in Figures 5 and 6 and Figures 7 and 8, respectively.

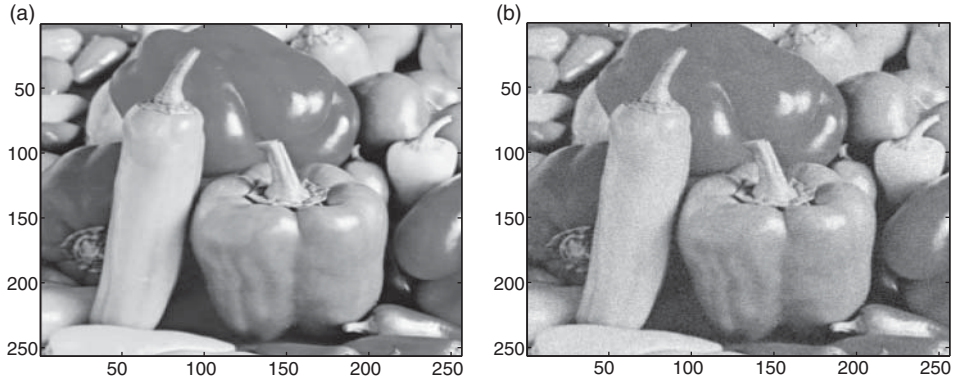

Figure 3. Original 'Pepper' image (left) and noisy image (right).

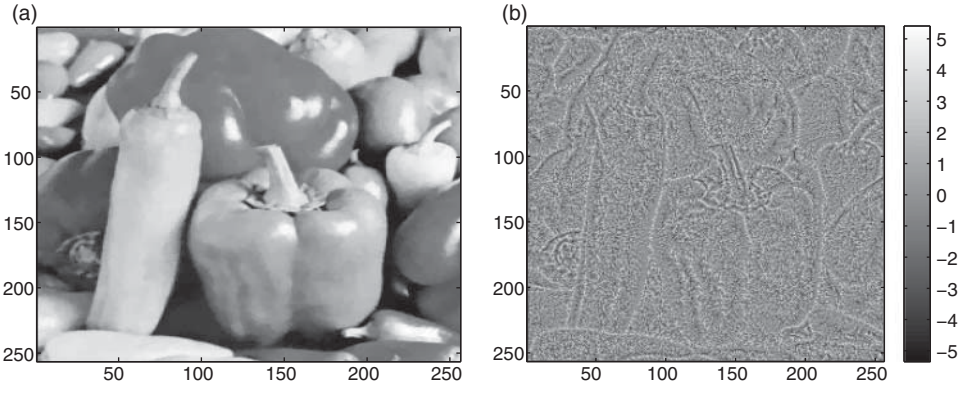

Figure 4. Original 'Barbara' image (left) and noisy image (right).

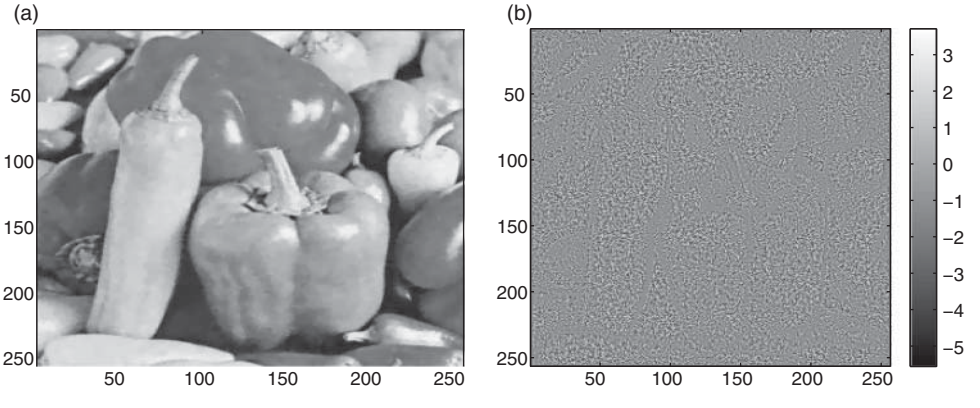

Figure 5. Image denoised by Algorithm the TV model (left) and the difference image (right).

As seen from the recovered images, both methods are able to maintain edges in the images, and at the same time to filter out noise. Algorithm 3 is good at reducing staircasing effect of the TV model. As far as the difference images are concerned, one observes that Algorithm 3 contains less structures in the difference images than TV method and the maximum of the difference for our Algorithm 3 decreases.

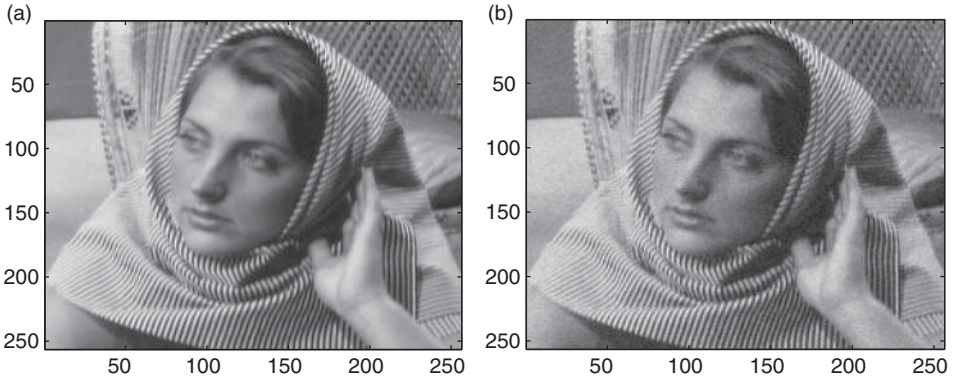

Figure 6. Image denoised by Algorithm 3 (left) and the difference image (right).

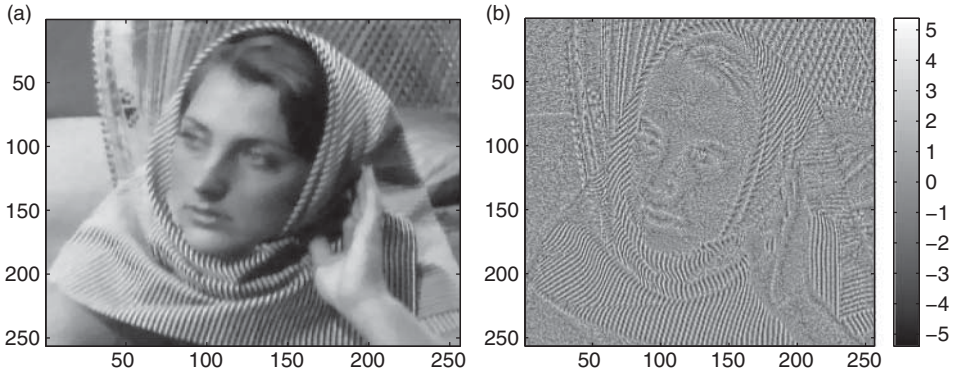

Figure 7. Image denoised by Algorithm the TV model (left) and the difference image (right).

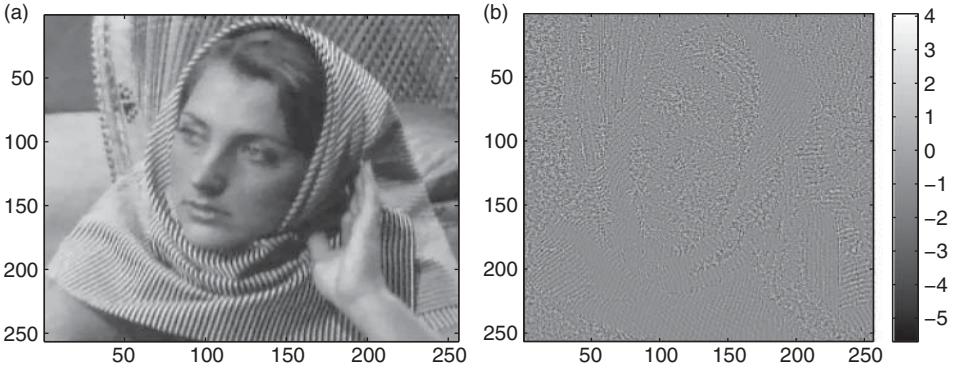

Figure 8. Image denoised by Algorithm 3 (left) and the difference image (right).

6. Conclusions

Curvature-based variational denoising models can restore effectively both blocky images (of piecewise constant intensities) and smooth images (with no clear jumps). For the former case, the restored quality is similar to that from a TV model and for the latter it is much better than the

TV. However, the resulting fourth-order PDE of a curvature model is much more complicated, presenting difficulties for many numerical techniques. Firstly, no simple FP methods can be constructed, a working FP method from [3] requires stabilization and a relatively large regularization parameter β. Secondly, reformulation methods such as from [6,24] can provide an elegant solution approach only by solving an approximated problem. This paper first gave an FP curvature method (directly) for solving the fourth-order PDE, and then proposed a relaxed FP method and a homotopy method with curve tracking to choose the regularizing parameter β adaptively. The resulting method turns out to be able to drive the Newton type method (as a corrector) to convergence for a range of test images and for any small parameter β. Numerical experiments can demonstrate advantages of our methods over the TV model. Future work will consider how to make use of our methods in a multigrid context as well as for other applications where curvature is used.

Acknowledgements

The research was supported by the Fundamental Research Funds for the Central Universities (Jdzd12009), the natural science foundation of Hunan Province (13JJB014) and the National Natural Science Foundation of China (11171051) and (11101186).

References

[1] E.L. Allgower and K. Georg, *Numerical path following*, in *Handbook of Numerical Analysis*, North-Holland, Amsterdam, 1997.
[2] K. Bredies, K. Kunisch, and T. Pock, *Total generalized variation*, SIAM J. Image Sci. 3(3) (2010), pp. 492–526.
[3] C. Brito-Loeza and K. Chen, *Multigrid algorithm for high order denoising*, SIAM J. Image Sci. 3(3) (2010), pp. 363–389.
[4] C. Brito-Loeza and K. Chen, *On high-order denoising models and fast algorithms for vector-valued images*, IEEE Trans. Image Process. 19(6) (2010), pp. 1518–1527.
[5] A. Buades, B. Coll, and J. Morel, *A review of image denoising algorithms, with a new one*, SIAM Multiscale Model. Simul. 4(2) (2005), pp. 490–530.
[6] J. Cai, B. Dong, S. Osher, and Z. Shen, *Image restorations: Total variation, wavelet frames and beyond*, J. Amer. Math. Soc. 25 (2012), pp. 1033–1089.
[7] A. Chambolle and P.L. Lions, *Image recovery via total variation minimization and related problems*, Numer. Math. 76(2) (1997), pp. 167–188.
[8] T.F. Chan, G.H. Golub, and P. Mulet, *A nonlinear primal-dual method for total variation-based image restoration*, SIAM J. Sci. Comput. 20 (1999), pp. 1964–1977.
[9] T.F. Chan, A. Marquina, and P. Mulet, *High-order total variation-based image restoration*, SIAM J. Sci. Comput. 22(2) (2000), pp. 503–516.
[10] T.F. Chan, H. Zhou, and R. Chan, *Advanced signal processing algorithms*, in *Proceedings of the International Society of Photo-Optical Instrumentation Engineers*, F.T. Luk, ed., SPIE, 1995, pp. 314–325.
[11] G. Gilboa and S. Osher, *Nonlocal operators with applications to image processing*, SIAM Multiscale Model. Simul. 7(3) (2008), pp. 1005–1028.
[12] M. Grasmair and F. Lenzen, *Anisotropic total variation filtering*, Appl. Math. Optim. 62 (2010), pp. 323–339.
[13] S. Kindermann, S. Osher, and P. Jones, *Deblurring and denoising of images by nonlocal functionals*, SIAM Multiscale Model. Simul. 4 (2005), pp. 1091–1115.
[14] Z. Lin, B. Yu, and D. Zhu, *A continuation method for solving fixed points of self-mappings in general nonconvex sets*, Nonlinear Anal. 52 (2003), pp. 905–915.
[15] Y. Lou, X. Zhang, S. Osher, and A. Bertozzi, *Image recovery via nonlocal operators*, J. Sci. Comput. 42(2) (2010), pp. 185–197.
[16] M. Lysaker, A. Lundervold, and X.C. Tai, *Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time*, IEEE Trans. Image Process. 12(12) (2003), pp. 1579–1590.
[17] M. Lysaker, S. Osher, and X.C. Tai, *Noise removal using smoothed normals and surface fitting*, IEEE Trans. Image Process. 13(10) (2004), pp. 1345–1357.
[18] A. Marquina and S. Osher, *Explicit algorithms for a new time dependent model based on level set motion for nonlinear deblurring and noise removal*, SIAM J. Sci. Comput. 22(2) (2000), pp. 387–405.
[19] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, *An iterative regularization method for total variation-based image restoration*, SIAM J. Multiscale Model. Simul. 4(2) (2005), pp. 460–489.
[20] L. Rudin, S. Osher, and C. Fatemi, *Nonlinear total variation based noise removal algorithms*, Phys. D 60 (1992), pp. 259–268.

[21] S. Setzer, G. Steidl, and T. Teuber, *Infimal convolution regularizations with discrete L^1-type functionals*, Commun. Math. Sci. 9(3) (2010), pp. 797–827.
[22] C.R. Vogel and M.E. Oman, *Iterative methods for total variation denoising*, SIAM J. Sci. Comput. 17 (1996), pp. 227–238.
[23] L.T. Watson, C. Billups, and P. Morgan, *Algorithm 652 Hompack: A suite of codes for globally convergent homotopy algorithms*, ACM Trans. Math. Softw. 13(3) (1987), pp. 281–310.
[24] C. Wu and X.C. Tai, *Augmented Lagrangian method, dual methods, and split Bregman iteration for ROF, vectorial TV, and high order models*, SIAM J. Image Sci. 3(3) (2010), pp. 300–339.
[25] F. Yang, K. Chen, and B. Yu, *Homotopy curve tracking for total variation image restoration*, J. Comput. Math. 30(2) (2012), pp. 177–196.
[26] W. Zhu and T.F. Chan, *Image denoising using mean curvature of image surface*, SIAM J. Image Sci. 5(1) (2012), pp. 1–32.