Augmented Lagrangian Method for A Mean Curvature Based Image Denoising Model

Wei Zhu [*], Xue-Cheng Tai [†], and Tony Chan [‡]

Abstract

High order derivative information has been widely used in developing variational models in image processing to accomplish more advanced tasks. However, it is a nontrivial issue to construct efficient numerical algorithms to deal with the minimization of these variational models due to the associated high order Euler-Lagrange equations. In this paper, we propose an efficient numerical method for a mean curvature based image denoising model using the augmented Lagrangian method. We describe how to construct an augmented Lagrangian functional for the model and detail the procedures of finding the related saddle-points of the functional. We present numerical experiments to illustrate the effectiveness and efficiency of the proposed numerical method, and show a few important features of the image denoising model such as keeping corners and image contrast. Moreover, a comparison with the gradient descent method further demonstrates the efficiency of the proposed augmented Lagrangian method.

1 Introduction

Image denoising provides a crucial step towards the capturing image signals from noisy images with lots of applications in medical imaging and video monitoring etc. During the last three decades, numerous methods have been applied to deal with this problem, such as variational methods, partial differential equations, statistical methods and so on [3, 4, 8, 23, 24, 25, 27, 28, 29, 33]. Among these variational method is one of the most popular methods. A classical variational model on image denoising was developed by Rudin, Osher, and Fatemi (ROF) [29], in which achieving a clean image from a noisy one amounts to the minimization of the following functional:

$$E(u) \quad = \quad \lambda \int_\Omega |\nabla u| + \int_\Omega (f-u)^2, \tag{1}$$

where $f : \Omega \to \mathbb{R}$ is a given noisy image defined on Ω (always a rectangle in $\mathbb{R}^2$) and $\lambda > 0$ is a positive constant. The ROF model has proved to be very efficient in the noise removal

[*] Department of Mathematics, University of Alabama, Box 870350, Tuscaloosa, AL 35487. E-mail: wzhu7@bama.ua.edu.

[†] Department of Mathematics, University of Bergen, Norway and Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, 637371, Singapore. Email: tai@math.uib.no.

[‡] Office of the President, Hong Kong University of Science and Technology (HKUST), Clear Water Bay, Kowlon, Hong Kong. Email: tonyfchan@ust.hk.

tasks, and it has also been widely used in other tasks of image processing such as image deblurring, image inpainting etc. Besides these, the idea of the total variation has broadly been utilized in many other disciplines.

Even though the ROF model is very powerful in removing noise while preserving edges and contours of objects, it also has some unfavorable properties. For instance, it yields the staircase effect, smears object corners and leads to the loss of image contrasts. To remedy these drawbacks, quite a few high order variational models have been proposed [1, 2, 21, 22, 10, 19, 38]. For instance, in [1], the authors used the Euler's elastica of level curves of a smooth function as a regularizer and proposed the following energy functional:

$$E(u) \;=\; \int \left[a + b\left(\nabla \cdot \frac{\nabla u}{|\nabla u|}\right)^2 \right] |\nabla u| + \frac{\eta}{2}\int (f-u)^2. \tag{2}$$

In [19], Lysaker et al. directly incorporated second order derivative information into the image denoising process and their energy functional reads:

$$E(u) \;=\; \lambda \int \sqrt{u_{xx}^2 + u_{xy}^2 + u_{yx}^2 + u_{yy}^2} + \frac{1}{2}\int (f-u)^2. \tag{3}$$

In [38], two of the authors proposed a variational model that uses the mean curvature of the induced image surface $(x, y, f(x, y))$ to remove noise. Specifically, the model employs the total variation of mean curvature as a regularized term and the functional is written as follows:

$$E(u) \;=\; \lambda \int \left| \nabla \cdot \left(\frac{\nabla u}{\sqrt{1+|\nabla u|^2}} \right) \right| + \frac{1}{2}\int (f-u)^2, \tag{4}$$

where λ is a tuning parameter. The term $\nabla \cdot (\frac{\nabla u}{\sqrt{1+|\nabla u|^2}})$ is the mean curvature of the surface $\phi(x, y, z) = u(x, y) - z = 0$. The model tries to fit the given noisy image surface $(x, y, f(x, y))$ with a surface $(x, y, u(x, y))$ that bears small amount of mean curvature. The model can sweep noise while keeping object edges, and it also ameliorates the staircase effect. More importantly, as discussed in [38], the model is also capable of preserving image contrasts as well as object corners.

While all the models can effectively accomplish the noise removal task, the associated functionals are not easy to minimize. Since they involve high order derivatives, the related Euler-Lagrange equations are often fourth-order, which raises a nontrivial issue of developing effective and efficient numerical algorithms to solve them. Indeed, there are lots of high order model existing in the literature of image processing [5, 8, 9, 12, 13, 14, 15, 26, 32, 36, 37]. Therefore, there is a pressing need to develop efficient numerical methods for these models.

In this paper, we plan to develop a fast algorithm to minimize the functional (4). We employ the idea of augmented Lagrangian methods that have already been successfully utilized in the minimization of the ROF model [34] and Euler's elastica based functionals [31]. In these works, the augmented Lagrangian methods achieve much higher speed than other numerical methods, such as the gradient descent method, primal-dual methods. The reason of how the method is so efficient mainly lies in the following fact: the augmented Lagrangian methods proposed in [34, 31] decomposed the original nontrivial minimization problem to be a few simple ones, some of which can be solved using the Fast Fourier transformation

(FFT), while the other ones can be quickly dealt with due to their existing explicit solutions. Therefore, the construction of an efficient augmented Lagrangian method for the minimization of a given functional depends on whether one can break down the original functional into simple ones. In the functional (4), note that the mean curvature $\nabla \cdot (\frac{\nabla u}{\sqrt{1+|\nabla u|^2}})$ is not homogeneous in the variable u, how to introduce a new variable that can commute freely with ∇u is a key problem for developing the associated augmented Lagrangian method of the functional (4). To solve this problem, just as what we regard a $2D$ image function as a surface in $\mathbb{R}^3$, we introduce a new variable $\mathbf{p} =< \nabla u, 1 >$, instead of $\mathbf{p} = \nabla u$, to connect with ∇u. In this way, we can apply the augmented Lagrangian method to the minimization of the functional (4).

The rest of this paper is organized as follows. In Section 2, we will first review the mean curvature based image denoising model, recall the augmented Lagrangian method for Euler's elastica based variational models [31], and then will detail how to apply the augmented Lagrangian method to the mean curvature denoising model. In Section 3, we will discuss how to carry out the numerical implementation. In Section 4, we present numerical experiments to illustrate the effectiveness and efficiency of the proposed numerical method, which is followed by a conclusion in Section 5.

2 Augmented Lagrangian Method

In this section, we first review the mean curvature image denoising model and the augmented Lagrangian method to the Euler's elastica functional, then detail how to treat the mean curvature based image denoising model using the augmented Lagrangian method.

2.1 The Mean Curvature Based Image Denoising Model

In [38], two of the authors proposed a variational model that uses the mean curvature of image surface for image denoising. For simplicity, we will call it as MC model in the following context. The functional can be written as follows:

$$E(u) \quad = \quad \lambda \int |\kappa_u| + \frac{1}{2} \int (f-u)^2, \tag{5}$$

where λ is a tuning parameter and the term $\kappa_u = \nabla \cdot (\frac{\nabla u}{\sqrt{1+|\nabla u|^2}})$ is the mean curvature of the image surface $\phi(x,y,z) = u(x,y) - z = 0$. This model aims for obtaining a surface that is close to the original induced surface $(x, y, f(x,y))$ and has a small amount of mean curvature. Therefore, noise part of an image can be removed to reduce the amount of mean curvature.

The motivation of the MC model is to remedy some unfavorable features of the classical ROF model [29] such as the staircase effect. To avoid this effect, as discussed above, numerous variational models that use high order derivative information have already developed [1, 2, 21, 22, 10, 19]. The MC model regards a given $2D$ image as a surface in $3D$ space, and uses the total variation of mean curvature of the surface as the regularizer. Therefore, the model suppresses the oscillation of mean curvature and prefers small mean curvature that lead to piecewisely smooth image surface. This effectively ameliorates the staircase effect.

Besides removing noise, the MC model is also capable of preserving image contrasts. In fact, as discussed in [38], for a region $R \subset \Omega$ with a C^2 boundary, one gets $\int_\Omega |\kappa_{h\chi_R}| = H^1(R)$ for the image $h\chi_R$, where χ_R and $H^1(R)$ are the characteristic function and the perimeter of the region R, respectively, and h depicts the height. Note that the regularizer is independent of the height h, which results in the preservation of image contrasts. It demonstrates an important difference between this model and the ROF model whose regularizer $\int_\Omega |\nabla(h\chi_R)|$ equals to $hH^1(R)$ that relies on the height h.

Moreover, the MC model is capable of preserving object corners. This suggests that the MC model can keep lots of visually meaningful clues. It is well-known that object corners will be smeared when the ROF model is applied.

Even though the MC model possesses the above merits, it is a challenging issue to develop numerical algorithms to minimize its functional as it involves high order derivatives and the non-differentiable regularization term. In fact, due to the non-differentiability of the function $|x|$, one cannot obtain the associated Euler-Lagrangian equation. Instead, as in [26], one may consider the following smoothed version of the functional (5)

$$E(u) \quad = \quad \lambda \int |\Phi(\kappa_u)| + \frac{1}{2}\int (f-u)^2, \tag{6}$$

with

$$\Phi(x) = \begin{cases} x^2, & |x| \leq 1 \\ |x|, & |x| > 1. \end{cases} \tag{7}$$

and the associated Euler-Lagrangian equation can then be written as follows [38]:

$$\lambda \nabla \cdot [\frac{1}{\sqrt{1+|\nabla u|^2}}(\mathbf{I}-\mathbf{P})\nabla\Phi'(\kappa_u)] - (f-u) \quad = \quad 0, \tag{8}$$

where $\mathbf{I}, \mathbf{P} : \mathbb{R}^2 \to \mathbb{R}^2$ with $\mathbf{I}(\mathbf{x}) = \mathbf{x}$ and $\mathbf{P}(\mathbf{x}) = (\mathbf{x} \cdot \frac{\nabla u}{\sqrt{1+|\nabla u|^2}})\frac{\nabla u}{\sqrt{1+|\nabla u|^2}}$, and the associated gradient descent equation of the functional (6) can be given as follows:

$$\frac{\partial u}{\partial t} \quad = \quad -\lambda \nabla \cdot [\frac{1}{\sqrt{1+|\nabla u|^2}}(\mathbf{I}-\mathbf{P})\nabla\Phi'(\kappa_u)] + (f-u). \tag{9}$$

with time t being an evolution parameter.

2.2 The Augmented Lagrangian Method to the Euler's Elastica functional

In [31], the augmented Lagrangian method was successfully applied to the Euler's elastica based image denoising model that was supposed to remove Gaussian white noise [1, 2]. The functional reads

$$E(u) \quad = \quad \int_\Omega \left[a + b\left(\nabla \cdot \frac{\nabla u}{|\nabla u|}\right)^2\right]|\nabla u| + \frac{\eta}{2}\int_\Omega (f-u)^2. \tag{10}$$

As discussed in [31], one may convert the minimization of this functional as a constrained minimization problem by introducing two new variables $\mathbf{p} = \nabla u$ and $\mathbf{n} = \nabla u/|\nabla u|$, and the corresponding augmented Lagrangian functional can be written as follows:

$$\begin{aligned}\mathcal{L}(u,\mathbf{p},\mathbf{n};\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2) &= \int \left[a+b\,(\nabla\cdot n)^2\right]|\mathbf{p}| + \frac{\eta}{2}\int (f-u)^2 \\ &+ \frac{r_1}{2}\int_\Omega (|\mathbf{p}|\mathbf{n}-\mathbf{p})^2 + \int_\Omega \boldsymbol{\lambda}_1\cdot(|\mathbf{p}|\mathbf{n}-\mathbf{p}) \\ &+ \frac{r_2}{2}\int_\Omega (\mathbf{p}-\nabla u)^2 + \int_\Omega \boldsymbol{\lambda}_2\cdot(\mathbf{p}-\nabla u). \end{aligned} \tag{11}$$

However, instead of dealing with this functional, in [31], the authors considered a more relaxed and easily handled functional that is given as follows:

$$\begin{aligned}\mathcal{L}(u,\mathbf{m},\mathbf{p},\mathbf{n};\lambda_1,\boldsymbol{\lambda}_2,\boldsymbol{\lambda}_3) &= \int \left[a+b\,(\nabla\cdot n)^2\right]|\mathbf{p}| + \frac{\eta}{2}\int (f-u)^2 \\ &+ \frac{r_1}{2}\int_\Omega (|\mathbf{p}|-\mathbf{m}\cdot\mathbf{p}) + \int_\Omega \lambda_1(|\mathbf{p}|-\mathbf{m}\cdot\mathbf{p}) \\ &+ \frac{r_2}{2}\int_\Omega (\mathbf{p}-\nabla u)^2 + \int_\Omega \boldsymbol{\lambda}_2\cdot(\mathbf{p}-\nabla u) \\ &+ \frac{r_3}{2}\int_\Omega (\mathbf{n}-\mathbf{m})^2 + \int_\Omega \boldsymbol{\lambda}_3\cdot(\mathbf{p}-\nabla u) + \delta_{\mathcal{R}}(\mathbf{m}), \end{aligned} \tag{12}$$

where $\mathcal{R} = \{\mathbf{m} \in L^2(\Omega) : |\mathbf{m}| \leq 1 \text{ a.e. in } \Omega\}$ and $\delta_{\mathcal{R}}(\cdot)$ is the characteristic function on $\mathcal{R}$ and can be expressed as follows:

$$\delta_{\mathcal{R}}(\mathbf{m}) = \begin{cases} 0, & \mathbf{m} \in \mathcal{R}; \\ +\infty, & \text{otherwise.} \end{cases}$$

One then needs to find its saddle points as these points correspond to the minimizers of the original functional (10). To this end, as in [31], one just needs to solve the associated subproblems of each variable by fixing other ones iteratively with new updated Lagrangian multipliers λ_1, $\boldsymbol{\lambda}_2$, $\boldsymbol{\lambda}_3$, and $\boldsymbol{\lambda}_4$. There are totally four subproblems related to the functional, each of which involves only one variable of u, $\mathbf{m}$, $\mathbf{p}$, and $\mathbf{n}$. Specifically, the equation for the variable u is a linear equation that can be efficiently solved using FFT, and the one for $\mathbf{n}$ can be locally solved. Moreover, the other two subproblems have explicit minimizers using some shrinkages. Therefore, the new augmented Lagrangian functional (12) will be solved very efficiently.

From the above discussion, whether an augmented Lagrangian method can be efficiently employed is highly determined by the original functional. Basically, a functional could be easily converted to be a constrained problem by introducing new variables. However, the searching for saddle points of the associated augmented Lagrangian functional may require quite a lot of effort. The success of the above augmented Lagrangian method is benefited by the simpleness of the associated subproblems.

2.3 Description of the Augmented Lagrangian Method to the Mean Curvature Based Image Denoising Model

In this section, we discuss how a similar augmented Lagrangian method will be applied to the MC model (5).

To develop an augmented Lagrangian functional for the mean curvature based image denoising model, we may naturally introduce three new variables q, $\mathbf{w}$, and p and consider the following constrained minimization problem:

$$\begin{aligned} &\min_{u,q,\mathbf{w},\mathbf{p}} \left[\lambda \int_\Omega |q| + \frac{1}{2} \int (f-u)^2 \right], \\ &\text{with } q = \nabla \cdot \mathbf{n},\ \mathbf{n} = \frac{\mathbf{p}}{\sqrt{1+|\mathbf{p}|^2}},\ \mathbf{p} = \nabla u. \end{aligned} \tag{13}$$

We then write down the associated augmented Lagrangian functional:

$$\begin{aligned} E(u, q, \mathbf{n}, \mathbf{p}, \lambda_1, \boldsymbol{\lambda}_2, \boldsymbol{\lambda}_3) &= \lambda \int |q| + \frac{1}{2} \int (f-u)^2 \\ &+ \frac{r_1}{2} \int (q - \nabla \cdot \mathbf{n})^2 + \int \lambda_1 (q - \nabla \cdot \mathbf{n}) \\ &+ \frac{r_2}{2} \int (\mathbf{n} - \frac{\mathbf{p}}{\sqrt{1+|\mathbf{p}|^2}})^2 + \int \boldsymbol{\lambda}_2 \cdot (\mathbf{n} - \frac{\mathbf{p}}{\sqrt{1+|\mathbf{p}|^2}}) \\ &+ \frac{r_3}{2} \int (\mathbf{p} - \nabla u)^2 + \int \boldsymbol{\lambda}_3 \cdot (\mathbf{p} - \nabla u). \end{aligned} \tag{14}$$

However, due to the term $\mathbf{p}/\sqrt{1+|\mathbf{p}|^2}$, this functional is also not easy to handle. This differs from the Euler's elastica functional where $\mathbf{p} = \nabla u$ and $\mathbf{n} = \nabla u/|\nabla u|$ can be nicely and mutually converted. Therefore, a key issue of applying the augmented Lagrangian method to functionals involving the term $\sqrt{1+|\nabla u|^2}$ or $\nabla u/\sqrt{1+|\nabla u|^2}$ is to find a way that can easily solve $\mathbf{p}$ from $\sqrt{1+|\mathbf{p}|^2}$. In fact, the term $\sqrt{1+|\nabla u|^2}$ also appears in other occasions such as the surface area based image denoising model [17]:

$$E(u) = \lambda \int_\Omega \sqrt{1+|\nabla u|^2} + \frac{1}{2} \int_\Omega (f-u)^2. \tag{15}$$

Before stating our idea of dealing with this issue, let's recall the idea of introducing the mean curvature denoising model. In this model, a given $2D$ image $f(x)$ is regarded as a surface $(x, y, f(x, y))$ in $\mathbb{R}^3$. One thus considers the surface $\phi(x, y, z) = u(x, y) - z = 0$ and the mean curvature $\kappa = \nabla \cdot (\nabla \phi/|\nabla \phi|) = \nabla \cdot (< \nabla u, -1 > /| < \nabla u, -1 > |)$. Note that one introduces two variables $\mathbf{p} = \nabla u$ and $\mathbf{n} = \nabla u/|\nabla u|$ to tackle the Euler's elastica for its curvature term $\kappa = \nabla \cdot (\nabla u/|\nabla u|)$. This gives us a hint to treat the curvature term in our case, that is, we may introduce a variable $\mathbf{p} =< \nabla u, -1 >$ instead of $\mathbf{p} = \nabla u$ and $\mathbf{n} =< \nabla u, -1 > /| < \nabla u, -1 > |$ accordingly.

As there is no difference if $\mathbf{p} =< \nabla u, 1 >$ is used, we then propose the following constrained minimization problem that is equivalent to the minimization of the functional (5):

$$\begin{aligned} &\min_{u,q,\mathbf{n},\mathbf{p}} \left[\lambda \int_\Omega |q| + \frac{1}{2} \int (f-u)^2 \right], \\ &\text{with } q = \nabla \cdot \mathbf{n},\ \mathbf{n} = \frac{\mathbf{p}}{|\mathbf{p}|},\ \mathbf{p} =< \nabla u, 1 >, \end{aligned} \tag{16}$$

and the associated augmented Lagrangian functional:

$$\begin{aligned}
\mathcal{L}(u,q,\mathbf{p},\mathbf{n},\mathbf{m};\lambda_1,\boldsymbol{\lambda}_2,\lambda_3,\boldsymbol{\lambda}_4) &= \lambda\int|q|+\frac{1}{2}\int(f-u)^2 \\
&+ r_1\int(|\mathbf{p}|-\mathbf{p}\cdot\mathbf{m})+\int\lambda_1(|\mathbf{p}|-\mathbf{p}\cdot\mathbf{m}) \\
&+ \frac{r_2}{2}\int|\mathbf{p}-<\nabla u,1>|^2+\int\boldsymbol{\lambda}_2\cdot(\mathbf{p}-<\nabla u,1>) \\
&+ \frac{r_3}{2}\int(q-\partial_x n_1-\partial_y n_2)^2+\int\lambda_3(q-\partial_x n_1-\partial_y n_2) \\
&+ \frac{r_4}{2}\int|\mathbf{n}-\mathbf{m}|^2+\int\boldsymbol{\lambda}_4\cdot(\mathbf{n}-\mathbf{m})+\delta_{\mathcal{R}}(\mathbf{m}), \qquad (17)
\end{aligned}$$

where $\mathbf{n},\mathbf{m},\mathbf{p}\in\mathbb{R}^3$, $\lambda_1,\lambda_3\in\mathbb{R}$, $\boldsymbol{\lambda}_2,\boldsymbol{\lambda}_4\in\mathbb{R}^3$. All these terms are similar to the ones discussed in [31]. For the sake of the completeness of presentation, we want to emphasize important points of the above augmented Lagrangian functional as follows.

The introduction of the variable $\mathbf{m}$ is to relax the variable $\mathbf{n}$ that is supposed to connect with the variable $\mathbf{p}$ in terms of $\mathbf{n}=\mathbf{p}/|\mathbf{p}|$, and the variable $\mathbf{m}$ is required to lie in the set $\mathcal{R}$ so that the term $|\mathbf{p}|-\mathbf{p}\cdot\mathbf{m}$ is always non-negative. As discussed in [31], the benefit of this non-negativeness is that the L^2 penalization is unnecessary and we just use $|\mathbf{p}|-\mathbf{p}\cdot\mathbf{m}$ as a penalization.

As the saddle points of the augmented Lagrangian functional (17) corresponds to the minimizers of the constrained minimization problem (16), one just needs to find the saddle points of (17). To this end, as in [31], we apply an iterative algorithm. Specifically, for each variable in (17), we fix all the other variables and seek a critical point of the induced functional to update this variable. Once all the variables are updated, the Lagrangian multipliers will also be advanced. Then we repeat all the process until the variables are convergent. The algorithm is summarized in Table 1 and Table 2.

Therefore, we need to consider the following subproblems and obtain critical points for each of them.

$$\begin{aligned}
\varepsilon_1(u) &= \frac{1}{2}\int(f-u)^2+\frac{r_2}{2}\int|\mathbf{p}-<\nabla u,1>|^2+\int\boldsymbol{\lambda}_2\cdot(\mathbf{p}-<\nabla u,1>), \qquad (24)\\
\varepsilon_2(q) &= \lambda\int|q|+\frac{r_3}{2}\int(q-\partial_x n_1-\partial_y n_2)^2+\int\lambda_3(q-\partial_x n_1-\partial_y n_2), \qquad (25)\\
\varepsilon_3(\mathbf{p}) &= r_1\int(|\mathbf{p}|-\mathbf{p}\cdot\mathbf{m})+\int\lambda_1(|\mathbf{p}|-\mathbf{p}\cdot\mathbf{m})+\frac{r_2}{2}\int|\mathbf{p}-<\nabla u,1>|^2\\
&\quad+\int\boldsymbol{\lambda}_2\cdot(\mathbf{p}-<\nabla u,1>), \qquad (26)\\
\varepsilon_4(\mathbf{n}) &= \frac{r_3}{2}\int(q-\partial_x n_1-\partial_y n_2)^2+\int\lambda_3(q-\partial_x n_1-\partial_y n_2)+\frac{r_4}{2}\int|\mathbf{n}-\mathbf{m}|^2\\
&\quad+\int\boldsymbol{\lambda}_4\cdot(\mathbf{n}-\mathbf{m}), \qquad (27)\\
\varepsilon_5(\mathbf{m}) &= r_1\int(|\mathbf{p}|-\mathbf{p}\cdot\mathbf{m})+\int\lambda_1(|\mathbf{p}|-\mathbf{p}\cdot\mathbf{m})+\frac{r_4}{2}\int|\mathbf{n}-\mathbf{m}|^2\\
&\quad+\int\boldsymbol{\lambda}_4\cdot(\mathbf{n}-\mathbf{m})+\delta_{\mathcal{R}}(\mathbf{m}). \qquad (28)
\end{aligned}$$

Table 1: Augmented Lagrangian method for the mean curvature model.

1. Initialization: u^0, q^0, $\mathbf{p}^0$, $\mathbf{m}^0$, $\mathbf{n}^0$, and λ_1^0, $\boldsymbol{\lambda}_2^0$, λ_3^0, $\boldsymbol{\lambda}_4^0$.

2. For $k \geq 1$, compute an approximate minimizer $(u^k, q^k, \mathbf{p}^k, \mathbf{m}^k, \mathbf{n}^k)$ of the augmented Lagrangian functional with the fixed Lagrangian multiplier λ_1^{k-1}, $\boldsymbol{\lambda}_2^{k-1}$, λ_3^{k-1}, $\boldsymbol{\lambda}_4^{k-1}$:

$$(u^k, q^k, \mathbf{p}^k, \mathbf{m}^k, \mathbf{n}^k) \quad \approx \quad \text{argmin } \mathcal{L}(u, q, \mathbf{p}, \mathbf{m}, \mathbf{n}, \lambda_1^{k-1}, \boldsymbol{\lambda}_2^{k-1}, \lambda_3^{k-1}, \boldsymbol{\lambda}_4^{k-1}). \tag{18}$$

3. Update the Lagrangian multipliers

$$\begin{aligned}
\lambda_1^k &= \lambda_1^{k-1} + r_1(|\mathbf{p}^k| - \mathbf{p}^k \cdot \mathbf{m}^k) \\
\boldsymbol{\lambda}_2^k &= \boldsymbol{\lambda}_2^{k-1} + r_2(|\mathbf{p}|^k - < \nabla u^k, 1 >) \\
\lambda_3^k &= \lambda_3^{k-1} + r_3(q^k - \partial_x n_1^k - \partial_y n_2^k) \\
\boldsymbol{\lambda}_4^k &= \boldsymbol{\lambda}_4^{k-1} + r_4(\mathbf{n}^k - \mathbf{m}^k),
\end{aligned}$$

where $\mathbf{n} = (n_1, n_2, n_3)$.

4. Measure the relative residuals and stop the iteration if they are small than a threshold ϵ_r.

Table 2: Alternating minimization method to solve the subproblems.

1. Initialization: $\widetilde{u}^0 = u^{k-1}$, $\widetilde{q}^0 = q^{k-1}$, $\widetilde{\mathbf{p}}^0 = \mathbf{p}^{k-1}$, $\widetilde{\mathbf{m}}^0 = \mathbf{m}^{k-1}$, and $\widetilde{\mathbf{m}}^0 = \mathbf{n}^{k-1}$.

2. For $l = 1, \cdots, L$ and fixed Lagrangian multiplier $\lambda_1 = \lambda_1^{k-1}$, $\boldsymbol{\lambda}_2 = \boldsymbol{\lambda}_2^{k-1}$, $\lambda_3 = \lambda_3^{k-1}$, and $\boldsymbol{\lambda}_4 = \boldsymbol{\lambda}_4^{k-1}$, solve the following subproblems :

$$\begin{aligned}
\widetilde{u}^l &= \text{argmin } \mathcal{L}(u, \widetilde{q}^{l-1}, \widetilde{\mathbf{p}}^{l-1}, \widetilde{\mathbf{m}}^{l-1}, \widetilde{\mathbf{n}}^{l-1}, \lambda_1, \boldsymbol{\lambda}_2, \lambda_3, \lambda_{\mathbf{4}}) && (19) \\
\widetilde{q}^l &= \text{argmin } \mathcal{L}(\widetilde{u}^l, q, \widetilde{\mathbf{p}}^{l-1}, \widetilde{\mathbf{m}}^{l-1}, \widetilde{\mathbf{n}}^{l-1}, \lambda_1, \boldsymbol{\lambda}_2, \lambda_3, \lambda_{\mathbf{4}}) && (20) \\
\widetilde{\mathbf{p}}^l &= \text{argmin } \mathcal{L}(\widetilde{u}^l, \widetilde{q}^l, \mathbf{p}, \widetilde{\mathbf{m}}^{l-1}, \widetilde{\mathbf{n}}^{l-1}, \lambda_1, \boldsymbol{\lambda}_2, \lambda_3, \lambda_{\mathbf{4}}) && (21) \\
\widetilde{\mathbf{m}}^l &= \text{argmin } \mathcal{L}(\widetilde{u}^l, \widetilde{q}^l, \mathbf{p}^l, \mathbf{m}, \widetilde{\mathbf{n}}^{l-1}, \lambda_1, \boldsymbol{\lambda}_2, \lambda_3, \lambda_{\mathbf{4}}) && (22) \\
\widetilde{\mathbf{n}}^l &= \text{argmin } \mathcal{L}(\widetilde{u}^l, \widetilde{q}^l, \mathbf{p}^l, \mathbf{m}^l, \mathbf{n}, \lambda_1, \boldsymbol{\lambda}_2, \lambda_3, \boldsymbol{\lambda}_4) && (23)
\end{aligned}$$

3. $(u^k, q^k, \mathbf{p}^k, \mathbf{m}^k, \mathbf{n}^k) = (\widetilde{u}^l, \widetilde{q}^l, \widetilde{\mathbf{p}}^l, \widetilde{\mathbf{m}}^l, \widetilde{\mathbf{n}}^l)$.

In what follows, we will discuss the minimizers of all the above functionals and the update of the Lagrangian Multipliers.

Similarly as what discussed in [31], the minimizers of the functionals $\varepsilon_2(q)$, $\varepsilon_3(\mathbf{p})$, and $\varepsilon_5(\mathbf{m})$ can be expressed explicitly, while the minimizers of the functionals $\varepsilon_2(u)$ and $\varepsilon_3(\mathbf{n})$ are determined by the associated Euler-Lagrange equations. For the sake of completeness of presentation, we provide the details here.

To obtain the minimizers of the functionals $\varepsilon_2(q)$ and $\varepsilon_3(\mathbf{p})$, we need the following lemmas [31].

Lemma 1. *Let $\mathbf{x}_0$ be a given scalar or vector, μ, r be two parameters with $r > 0$, then the minimizer of the following functional*

$$min_{\mathbf{x}} \int_\Omega \mu|\mathbf{x}| + \frac{r}{2}|\mathbf{x} - \mathbf{x}_0|^2,$$

reads

$$\mathbf{x} \quad = \quad max\left\{0, 1 - \frac{\mu}{r|\mathbf{x}_0|}\right\}\mathbf{x}_0,$$

Lemma 2. *(Lions and Mercier [18]) Let $\mathbf{m}_0$ be a given vector, then the minimization problem*

$$min_{\mathbf{m}} \int_\Omega |\mathbf{m} - \mathbf{m}_0|^2 + \delta_{\mathcal{R}}(m)$$

has an explicit solution

$$\mathbf{m} \quad = \quad \begin{cases} \mathbf{m}_0, & |\mathbf{m}_0| \leq 1; \\ \mathbf{m}_0/|\mathbf{m}_0|, & otherwise. \end{cases}$$

Note that

$$\varepsilon_2(q) \quad = \quad \int_\Omega \lambda|q| + \frac{r_3}{2}\left(q - (\partial_x n_1 + \partial_y n_2 - \frac{\lambda_3}{r_3})\right)^2 + \widetilde{c}_1, \tag{29}$$

where $\widetilde{c}_1$ is a constant independent of the variable q, and

$$\varepsilon_3(\mathbf{p}) \quad = \quad \int_\Omega (r_1 + \lambda_1)|\mathbf{p}| + \frac{r_2}{2}|\mathbf{p} - (< \nabla u, 1 > -\frac{\boldsymbol{\lambda}_2}{r_2} + \frac{r_1 + \lambda_1}{r_2}\mathbf{m})|^2 + \widetilde{c}_2,$$

where $\widetilde{c}_2$ is a constant independent of the variable $\mathbf{p}$, based on Lemma 1, we obtain the minimizers of $\varepsilon_2(q)$ and $\varepsilon_3(\mathbf{p})$ as follows:

$$\text{Argmin}_q \varepsilon_2(q) \quad = \quad \max\left\{0, 1 - \frac{\lambda}{r_3|\widetilde{q}|}\right\}\widetilde{q}, \qquad \widetilde{q} = \partial_x n_1 + \partial_y n_2 - \frac{\lambda_3}{r_3}, \tag{30}$$

$$\text{Argmin}_{\mathbf{p}} \varepsilon_3(\mathbf{p}) \quad = \quad \max\left\{0, 1 - \frac{r_1 + \lambda_1}{r_2|\widetilde{\mathbf{p}}|}\right\}\widetilde{\mathbf{p}}, \qquad \widetilde{\mathbf{p}} =< \nabla u, 1 > -\frac{\boldsymbol{\lambda}_2}{r_2} + \frac{(r_1 + \lambda_1)\mathbf{m}}{r_2}. \tag{31}$$

As

$$\varepsilon_5(\mathbf{m}) = \frac{r_4}{2}\int |\mathbf{m} - \left(\mathbf{n} + \frac{r_1+\lambda_1}{r_4}\mathbf{p} + \frac{\boldsymbol{\lambda}_4}{r_4}\right)|^2 + \delta_{\mathcal{R}}(\mathbf{m}) + \widetilde{c}_3,$$

where $\widetilde{c}_3$ is a constant independent of the variable $\mathbf{m}$, based on Lemma 2, we get

$$\text{Argmin}_{\mathbf{m}}\varepsilon_5(\mathbf{m}) = \begin{cases} \widetilde{\mathbf{m}}, & |\widetilde{\mathbf{m}}| \le 1; \\ \widetilde{\mathbf{m}}/|\widetilde{\mathbf{m}}|, & |\widetilde{\mathbf{m}}| > 1. \end{cases} \qquad \widetilde{\mathbf{m}} = \mathbf{n} + \frac{\boldsymbol{\lambda}_4}{r_4} + \frac{(r_1+\lambda_1)\mathbf{p}}{r_4}. \tag{32}$$

As for the functionals $\varepsilon_1(u)$, $\varepsilon_4(\mathbf{n})$, we need to get the associated Euler-Lagrange equations. Standard procedures lead to the following equations of u and $\mathbf{n}$:

$$-r_2\Delta u + u = f - (r_2 p_1 + \lambda_{21})_x - (r_2 p_2 + \lambda_{22})_y, \tag{33}$$

and

$$\begin{aligned} -r_3(\partial_x n_1 + \partial_y n_2)_x + r_4 n_1 &= r_4 m_1 - \lambda_{41} - (r_3 q + \lambda_3)_x, \\ -r_3(\partial_x n_1 + \partial_y n_2)_y + r_4 n_2 &= r_4 m_2 - \lambda_{42} - (r_3 q + \lambda_3)_y, \\ n_3 &= m_3 - \lambda_{43}/r_4, \end{aligned} \tag{34}$$

where $\mathbf{p} =< p_1, p_2, p_3 >$, $\mathbf{m} =< m_1, m_2, m_3 >$, $\mathbf{n} =< n_1, n_2, n_3 >$ and the Lagrangian multiplier $\boldsymbol{\lambda}_2 =< \lambda_{21}, \lambda_{22}, \lambda_{23} >$, $\boldsymbol{\lambda}_4 =< \lambda_{41}, \lambda_{42}, \lambda_{43} >$. To update the variables u and $\mathbf{n}$, one needs to solve these Euler-Lagrangian equations.

Finally, the updates of the variables $\lambda_1, \boldsymbol{\lambda}_2, \lambda_3, \boldsymbol{\lambda}_4$ are given as follows:

$$\begin{aligned} \lambda_1^{new} &= \lambda_1^{old} + r_1(|\mathbf{p}| - \mathbf{p}\cdot\mathbf{m}), \\ \boldsymbol{\lambda}_2^{new} &= \boldsymbol{\lambda}_2^{old} + r_2(\mathbf{p} - < \nabla u, 1 >), \\ \lambda_3^{new} &= \lambda_3^{old} + r_3(q - \partial_x n_1 - \partial_y n_2), \\ \boldsymbol{\lambda}_4^{new} &= \boldsymbol{\lambda}_4^{old} + r_4(\mathbf{n} - \mathbf{m}). \end{aligned} \tag{35}$$

2.4 Discussion of the Smoothed Mean Curvature Based Image Denoising Model

As discussed previously, in [38], due to the non-differentiability of the MC model (5), the gradient descent equation of the smoothed version of the MC model (6) had to be employed. However, by using the augmented Lagrangian method, the difficulty can be easily overcome [34, 31]. This is one appealing feature of the augmented Lagrangian methods.

In this work, to fairly get a comparison of efficiency for the proposed augmented Lagrange method and the gradient descent method, we also consider the functional (6). In fact, to minimize this functional, we may use the same augmented Lagrangian functional (17) except that the term $\lambda\int|q|$ that will be replaced by $\lambda\int\Phi_\epsilon(q)$ with Φ_ϵ defined as follows:

$$\Phi_\epsilon(x) = \begin{cases} x^2/\epsilon, & |x| \le \epsilon, \\ |x|, & |x| > \epsilon. \end{cases} \tag{36}$$

where $\epsilon > 0$ is a parameter. This modification only leads to the change of the subproblem related to the variable q, and it becomes

$$\varepsilon_2(q) \quad = \quad \lambda \int \Phi_\epsilon(q) + \frac{r_3}{2} \int (q - \partial_x n_1 - \partial_y n_2)^2 + \int \lambda_3 (q - \partial_x n_1 - \partial_y n_2). \qquad (37)$$

Therefore, the subsequent problem is how to determine its minimizer. To this end, we propose the following lemma.

Lemma 3. *Let q_0 be a given scalar, μ, r be two parameters with $r > 0$, then the following functional*

$$\int_\Omega \mu \Phi_\epsilon(q) + \frac{r}{2}(q - q_0)^2,$$

with Φ_ϵ being defined as (36) attains its minimizer at

$$q \quad = \quad sgn(q_0) \begin{cases} q_2, & q_1 \geq \epsilon \text{ and } q_2 \geq \epsilon, \\ \epsilon, & q_1 \geq \epsilon \text{ and } q_2 < \epsilon, \\ q_1, & q_1 < \epsilon \text{ and } q_2 \leq \epsilon, \\ argmin\{f(q_1), f(q_2)\}, & q_1 < \epsilon \text{ and } q_2 > \epsilon. \end{cases} \qquad (38)$$

where $f(q) = \mu \Phi_\epsilon(q) + \frac{r}{2}(q - q_0)^2$, and q_1, q_2 are given as follows:

$$\begin{aligned} q_1 &= \frac{r|q_0|}{2\mu/\epsilon + r}, \\ q_2 &= |q_0| - \frac{\mu}{r}. \end{aligned}$$

Proof. Since no derivatives involved in the functional, one just needs to find the minimum point of the following scalar function:

$$f(q) \quad = \quad \mu \Phi_\epsilon(q) + \frac{r}{2}(q - q_0)^2.$$

Note that f is piecewisely defined on the intervals $(-\infty, -\epsilon)$, $[-\epsilon, \epsilon]$, and $(\epsilon, +\infty)$, to find the minimum, one just needs to calculate the critical points for each interval. As f is an even function, only the case with $q_0 > 0$ is considered below. The critical points of f defined in $[-\epsilon, \epsilon]$ and $(\epsilon, +\infty)$ are given as

$$\begin{aligned} q_1 &= \frac{r q_0}{2\mu/\epsilon + r}, \\ q_2 &= q_0 - \frac{\mu}{r}. \end{aligned}$$

respectively.

As f is upward convex in each of these intervals, f attains the minimum at the critical point if it is inside the interval where f is defined. Therefore, if $q_1 \geq \epsilon$, the minimum is either at ϵ or at q_2, which can be determined by whether q_2 is inside $(\epsilon, +\infty)$; if $q_1 < \epsilon$, the minimum may be attained either at q_1 or q_2. In summary, we get the minimum point of f as stated in this lemma. □

3 Numerical Implementation

In this section, we present the details of how to solve the equations (33) and (34) and update the variables q, $\mathbf{p}$ and $\mathbf{m}$ for each iteration. Since the numerics are almost the same as what discussed in [31], we here only present the key points.

Before discussing the discretizations of the above equations, we need to note that the regularizer of the MC model, $|\kappa_u| = |\nabla \cdot (\frac{\nabla u}{\sqrt{1+|\nabla u|^2}})|$ or $\Phi_\epsilon(\kappa_u)$, is not homogeneous in u. This is one of the most important features that differentiate it from other image denoising ones such as the ROF model [29] and the Euler's elastica based model [1, 2]. Therefore, one shall introduce a spatial mesh size for the discretization of the terms involving derivatives.

Numerically, the spatial mesh size plays an important role in the performance of the model, as it determines the magnitude of gradient. Note that any given image is only defined on a finite number of grids, therefore, once the spatial mesh size is small enough, any tiny variation of u in a neighborhood of a grid will lead to large κ_u at that point and may be preserved as a jump by the model. This surely fails to remove the noise but also suggests that a relative small spatial mesh size helps to preserve fine structures of images. On the other hand, if the spatial mesh size is large enough, and as the image intensity only ranges in $[0, 255]$, then even visually salient jumps will still lead to a small κ_u and be removed.

We then define the discrete backward and forward differential operators with periodic boundary condition and the spatial mesh size h as follows:

$$
\begin{aligned}
\partial_1^- u(i,j) &= \begin{cases} (u(i,j) - u(i-1,j))/h, & 1 < i \le M; \\ (u(1,j) - u(M,j))/h, & i = 1. \end{cases} \\
\partial_1^+ u(i,j) &= \begin{cases} (u(i+1,j) - u(i,j))/h, & 1 \le i < M-1; \\ (u(1,j) - u(M,j))/h, & i = M. \end{cases} \\
\partial_2^- u(i,j) &= \begin{cases} (u(i,j) - u(i,j-1))/h, & 1 < j \le N; \\ (u(i,1) - u(i,N))/h, & j = 1. \end{cases} \\
\partial_2^+ u(i,j) &= \begin{cases} (u(i,j+1) - u(i,j))/h, & 1 \le j < N; \\ (u(i,1) - u(i,N))/h, & j = N. \end{cases}
\end{aligned}
$$

and the central difference operators and the gradient operators are defined accordingly as

$$
\begin{aligned}
\partial_1^c u(i,j) &= (\partial_1^- u(i,j) + \partial_1^+ u(i,j))/2, \\
\partial_2^c u(i,j) &= (\partial_2^- u(i,j) + \partial_2^+ u(i,j))/2, \\
\nabla^\pm u(i,j) &= (\partial_1^\pm u(i,j), \partial_2^\pm u(i,j)).
\end{aligned}
$$

Let's first discuss how to solve the equation (33). As periodic condition is imposed, we may use FFT to solve this linear equation. We employ the following discretization:

$$
-r_2 \text{div}^- \nabla^+ u + u = f - \partial_1^- (r_2 p_1 + \lambda_{21}) - \partial_2^- (r_2 p_2 + \lambda_{22}), \tag{39}
$$

or

$$
-r_2(\partial_1^- \partial_1^+ + \partial_2^- \partial_2^+) u + u = f - \partial_1^- (r_2 p_1 + \lambda_{21}) - \partial_2^- (r_2 p_2 + \lambda_{22}). \tag{40}
$$

We then apply the discrete Fourier transform $\mathcal{F}$ for both sides. Note that

$$\begin{aligned}\mathcal{F}\partial_1^{\pm}u(i,j) &= \pm(e^{\pm\sqrt{-1}z_i}-1)\mathcal{F}u(i,j),\\ \mathcal{F}\partial_2^{\pm}u(i,j) &= \pm(e^{\pm\sqrt{-1}z_j}-1)\mathcal{F}u(i,j),\end{aligned}$$

where $z_i = 2\pi i/M, i = 1, \cdots, M$, $z_j = 2\pi j/N, j = 1, \cdots, N$, we get

$$(-2r_2(\cos z_i + \cos z_j - 2) + 1)\mathcal{F}u(i,j) = \mathcal{F}g(i,j), \tag{41}$$

where $g(i,j) = f(i,j) - \partial_1^-(r_2p_1 + \lambda_{21})(i,j) - \partial_2^-(r_2p_2 + \lambda_{22})(i,j)$. Therefore, once $\mathcal{F}u$ is calculated, u can be obtained using the discrete inverse Fourier transform.

We also use FFT to solve the equations (34). To this end, we first discretize the first two equations of (34) as follows:

$$\begin{aligned}-r_3(\partial_1^+\partial_1^-n_1 + \partial_1^+\partial_2^-n_2) + r_4n_1 &= r_4m_1 - \lambda_{41} - \partial_1^+(r_3q + \lambda_3) &(42)\\ -r_3(\partial_2^+\partial_1^-n_1 + \partial_2^+\partial_2^-n_2) + r_4n_2 &= r_4m_2 - \lambda_{42} - \partial_2^+(r_3q + \lambda_3), &(43)\end{aligned}$$

then apply the discrete Fourier transform to both sides and get the following 2×2 system for each grid (i,j):

$$\begin{pmatrix} a_{11} & a_{12}\\ a_{21} & a_{22}\end{pmatrix}\begin{pmatrix}\mathcal{F}n_1(i,j)\\ \mathcal{F}n_2(i,j)\end{pmatrix} = \begin{pmatrix}\mathcal{F}h_1(i,j)\\ \mathcal{F}h_2(i,j)\end{pmatrix} \tag{44}$$

with

$$\begin{aligned}a_{11} &= r_4 - r_3(e^{2\pi i\sqrt{-1}/M} - 1)(1 - e^{-2\pi i\sqrt{-1}/M}),\\ a_{12} &= -r_3(e^{2\pi i\sqrt{-1}/M} - 1)(1 - e^{-2\pi j\sqrt{-1}/N}),\\ a_{21} &= -r_3(e^{2\pi j\sqrt{-1}/N} - 1)(1 - e^{-2\pi i\sqrt{-1}/M}),\\ a_{11} &= r_4 - r_3(e^{2\pi j\sqrt{-1}/N} - 1)(1 - e^{-2\pi j\sqrt{-1}/N}),\end{aligned}$$

and

$$\begin{aligned}h_1(i,j) &= r_4m_1(i,j) - \lambda_{41}(i,j) - \partial_1^+(r_3q + \lambda_3)(i,j)\\ h_2(i,j) &= r_4m_2(i,j) - \lambda_{42}(i,j) - \partial_2^+(r_3q + \lambda_3)(i,j).\end{aligned}$$

As the determinant of the above 2×2 matrix equals to $r_4^2 - 2r_4r_3(\cos 2\pi i/M + \cos 2\pi j/N - 2)$, which is always positive when $r_4 > 0$, we can easily solve the above system to get $\mathcal{F}n_1$ and $\mathcal{F}n_2$, then apply the discrete inverse Fourier transform to them, and the real parts of the inverse Fourier give us the new n_1 and n_2. The third component n_3 of $\mathbf{n}$ can be calculated directly.

In what follows, we discuss the update of the variables q, $\mathbf{p}$, $\mathbf{m}$ as well as the Lagrangian multipliers.

As q is scalar defined on the grid points (i,j), based on the formulation (30), one gets

$$q(i,j) = \max\left\{0, 1 - \frac{\lambda}{r_3|\widetilde{q}_{i,j}|}\right\}\widetilde{q}_{i,j},$$

with $\widetilde{q}_{i,j} = \partial_1^-n_1(i,j) + \partial_2^-n_2(i,j) - \lambda_3/r_3$.

As for the variable $\mathbf{p}$, we first calculate the three components of $\widetilde{\mathbf{p}}$, and then calculate the length of $\widetilde{\mathbf{p}}$ and the updated $\mathbf{p}$. Specifically,

$$\begin{aligned}\widetilde{\mathbf{p}}(i,j) &= <\partial_1^+ u(i,j), \partial_2^+ u(i,j), 1> - \frac{<\lambda_{21}, \lambda_{22}, \lambda_{23}>(i,j)}{r_2} \\ &\quad + \frac{(r_1+\lambda_1) <m_1, m_2, m_3>(i,j)}{r_2},\end{aligned}$$

and based on the formulation (31)

$$\mathbf{p}(i,j) = \max\left\{0, 1 - \frac{r_1 + \lambda_1(i,j)}{r_2 |\widetilde{\mathbf{p}}(i,j)|}\right\} \widetilde{\mathbf{p}}(i,j).$$

Similarly, we calculate

$$\widetilde{\mathbf{m}}(i,j) = \mathbf{n}(i,j) + \frac{\boldsymbol{\lambda}_4(i,j)}{r_4} + \frac{(r_1 + \lambda_1(i,j))\mathbf{p}(i,j)}{r_4},$$

and get the new $\mathbf{m}(i,j)$ using the formulation (32).

Moreover, based on the formulations (35), we may update all the Lagrangian multipliers:

$$\lambda_1^{new}(i,j) = \lambda_1^{old}(i,j) + r_1(|\mathbf{p}|(i,j) - \mathbf{p}(i,j)\cdot\mathbf{m}(i,j)),$$

with $|\mathbf{p}|(i,j) = \sqrt{p_1^2(i,j) + p_2^2(i,j) + p_3^2(i,j)}$, and

$$\begin{aligned}\lambda_{21}^{new}(i,j) &= \lambda_{21}^{old}(i,j) + r_2(p_1(i,j) - \partial_1^- u(i,j)),\\ \lambda_{22}^{new}(i,j) &= \lambda_{22}^{old}(i,j) + r_2(p_2(i,j) - \partial_2^- u(i,j)),\\ \lambda_{23}^{new}(i,j) &= \lambda_{23}^{old}(i,j) + r_2(p_3(i,j) - 1),\\ \lambda_3^{new}(i,j) &= \lambda_3^{old}(i,j) + r_3(q(i,j) - \partial_1^- n_1(i,j) - \partial_2^- n_2(i,j)),\\ \lambda_{41}^{new}(i,j) &= \lambda_{41}^{old}(i,j) + r_4(n_1(i,j) - m_1(i,j)),\\ \lambda_{42}^{new}(i,j) &= \lambda_{42}^{old}(i,j) + r_4(n_2(i,j) - m_2(i,j)),\\ \lambda_{43}^{new}(i,j) &= \lambda_{43}^{old}(i,j) + r_4(n_3(i,j) - m_3(i,j)).\end{aligned}$$

4 Numerical Experiments

In this section, we apply the proposed augmented Lagrangian method (ALM) for both the original MC model (5) and the smoothed MC model (6), and then compare the algorithm with the gradient descent method (GDM) in their efficiency for the smoothed MC model.

To check whether the ALM converges to some saddle point, we monitor the following relative residuals as in [31]:

$$(R_1^k, R_2^k, R_3^k, R_4^k) = \frac{1}{|\Omega|}(\widetilde{R}_1^k, \widetilde{R}_2^k, \widetilde{R}_3^k, \widetilde{R}_4^k), \tag{45}$$

with

$$\begin{aligned}\widetilde{R}_1^k &= |\mathbf{p}^k| - \mathbf{p}^k \cdot \mathbf{m}^k,\\ \widetilde{R}_2^k &= \mathbf{p}^k - <\nabla u^k, 1>,\\ \widetilde{R}_3^k &= q^k - \partial_x n_1^k - \partial_y n_2^k,\\ \widetilde{R}_4^k &= \mathbf{n}^k - \mathbf{m}^k.\end{aligned}$$

In all the following numerical experiments, we use the relative residuals (45) as the stopping criterion, that is, given a small threshold ϵ_r, once $R_i^k < \epsilon_r$ for $i = 1, ..., 4$ and for some k, the iteration process will be terminated. To check the convergence of the iteration process, as in [31], we also check the relative errors of Lagrange multipliers:

$$(L_1^k, L_2^k, L_3^k, L_4^k) = \left(\frac{\| \lambda_1^k - \lambda_1^{k-1} \|_{L^1}}{\| \lambda_1^{k-1} \|_{L^1}}, \frac{\| \boldsymbol{\lambda}_2^k - \boldsymbol{\lambda}_2^{k-1} \|_{L^1}}{\| \boldsymbol{\lambda}_2^{k-1} \|_{L^1}}, \frac{\| \lambda_3^k - \lambda_3^{k-1} \|_{L^1}}{\| \lambda_3^{k-1} \|_{L^1}}, \frac{\| \boldsymbol{\lambda}_4^k - \boldsymbol{\lambda}_4^{k-1} \|_{L^1}}{\| \boldsymbol{\lambda}_4^{k-1} \|_{L^1}} \right), \tag{46}$$

and the relative error of the solution u^k

$$\frac{\| u^k - u^{k-1} \|_{L^1}}{\| u^{k-1} \|_{L^1}}. \tag{47}$$

Besides all these quantities, we also consider how the energy (5) is evolving during the iteration by tracking the amount $E(u^k)$. For the presentation purpose, all the above quantities are shown in *log*-scale in the following results. Moreover, to illustrate what signals are removed as noise, we also present the associated residual image $f - u$, besides the given noisy image f and the cleaned one u.

We first consider some real images. In Fig 1, we present the results for the images "Cameraman" and "Peppers", including the original noisy images, cleaned ones and the residual images. The cleaned images demonstrate that noise part and small scale signals have been removed, which can be observed in the related residual images. One can easily find that the cleaned images are composed of piecewisely smooth surface patches, that is, they are free from the stair case effect that is a typical unfavorable feature of the well-known ROF model. In Fig 2, the plots of the relative residuals (45), relative errors in Lagrange multipliers (46), relative error in u^k (47), and the energy versus iteration are listed for these two examples. These series of plots illustrate the convergence of the iteration process using the proposed Augmented Lagrange method, and also show that the process leads to some saddle points of the constrained minimization problems.

We then apply the proposed Augmented Lagrange method for some synthetic images. In Fig 3, the results of two synthetic images are presented. These two images have some special visual clues. For instance, the image "lattice" has corners and sharp edges, and the image "rings" consists of a few surface patches with both flat patches and a curved one inside the image. From the two cleaned images, one can check that the noise has been effectively removed and the corresponding residual images nearly contain no meaningful signals for both examples. A careful check of the cleaned lattice image shows that the corners are preserved very well, which is an important feature of the mean curvature model [38]. Moreover, in the plots (**a4**) and (**b4**), we present the slices of the original noise free images, noisy images and the cleaned ones. These slices provide another perspective to check the denoising results. In the plot (**a4**), we can see that the cleaned slice curve (blue) almost overlaps with the original noise free slice curve (red), which demonstrates the preservation of image contrast, which is another feature of the mean curvature model. The plot (**b4**) shows that the original noise free "rings" has a smooth arch in its center, which is also almost fully restored by the model.

We have applied the proposed ALM for real and synthetic images. To explicit how efficiency the proposed method is, in Table 3, we present the cpu time for all the above

Table 3: The presentation of the sizes, the SNRs, the numbers of total outer iteration, and the computational time for the images in Figures 1 and 3.

image	size	SNR	Number of iterations	time (sec)
Fig. 1-(a2)	256×256	8.93	583	155.22
Fig. 1-(b2)	256×256	10.37	433	112.01
Fig. 3-(a2)	128×128	10.88	268	18.80
Fig. 3-(b2)	100×100	11.40	579	25.58

experiments.

As mentioned in the introduction, a few other higher order image denoising models have been existed in the literature such as Euler's elastica image denoising model [1] and the Lysaker et al.'s model [19]. In Figure 4, we apply the MC model, Euler's elastica model and the ROF model for a synthetic image with sharp edge and corners. The results show that the MC model keeps corners better than the other two models. This can be observed in the related residual images where some corner signals exist explicitly for the two models. Moreover, from the cleaned images, one can see that the MC model also preserves image contrast better than the ROF model. In fact, the result obtained by the ROF shows less darkness of the object than that of the MC model.

The above three examples demonstrate that the proposed ALM can efficiently accomplish the task of sweeping noise and keeping corners and image contrasts that are assumed by the MC model.

In the next experiment, we demonstrate an important feature of the MC model, that is, the clean image function u depends discontinuously on the regularization parameter λ. To see this point clearly, we compare the results using the ROF model and the MC model respectively in Figure 5. In this experiment, we consider a noise-free synthetic image with a bunch of squares of different sizes. We first list the results of the MC model for two different regularization parameters. One can see that when $\lambda = 2.5 \times 10^4$, the squares with the smallest size are swept as noise and show up in the residual image, while the other squares are well preserved in the cleaned image. When we increase the parameter to be $\lambda = 4.0 \times 10^4$, then the squares of larger size will be removed as shown in the residual image. This phenomenon demonstrates that the results of the MC model do not continuously depend on the regularization parameter λ. In contrast, the ROF model depends on its regularization parameter continuously. In fact, for either a relatively small parameter $\lambda = 10.0$ and a relatively larger one $\lambda = 100.0$, all the squares, regardless of the sizes, are partially swept as noise, and the removed part depends on the square size.

This specific feature of the MC model suggests a data-driven scale selection approach, that is, as the regularization parameter λ gradually increases, objects of small sizes will be removed from the given image, while large size objects will be almost completely preserved. This particular property is also possessed by the TV-L1 model [6] with lots of applications [7, 35]. However, the MC model is able to keep corners, while the TV-L1 model cannot, which suggests that the MC model might be more suited than the TV-L1 model to applications for which the preservation of geometry is crucial.

In the last experiment, we compare the efficiency of the proposed ALM and the GDM by applying them to the smoothed MC model (6). To make the comparison fairly, for each method, we stop the iteration process once the following L^2 error in u^k with satisfies:

$$\frac{1}{|\Omega|} \parallel u^k - u_{exact} \parallel_{L^2} \quad < \quad \epsilon_e, \tag{48}$$

where $\epsilon_e = 5.5 \times 10^{-3}$, u_{exact} denotes the convergent state of u^k. Note that the MC model is not convex, the two algorithms may converge to two different local minimizers. Therefore, for each algorithm, we calculate its own u_{exact}. For instance, for ALM, we get the u_{exact} using a far large number of iterations; while for GDM, besides the huge iteration number, we choose a much small time step. By using the quantity $\frac{1}{|\Omega|} \parallel u^k - u_{exact} \parallel_{L^2}$, one can check which algorithm can converge to u_{exact} more quickly. In Figure 6, we list the results for both algorithms. The two cleaned images show almost no difference, however, the energy associated with ALM drops dramatically within one hundred iterations. From the efficiency comparison Table 4, we also see that ALM takes much fewer iterations and less cost time than GDM, which demonstrate that ALM is much more efficient than GDM.

Table 4: Efficiency comparison between ALM and GDM for Figure 6

	Augmented Lagrangian Method	Gradient Descent Method
Iterations needed	866	53,635
Time needed (seconds)	233.78	7,153.88

5 Conclusion

Recently, the augmented Lagrangian method has been successfully used to minimize the classical ROF functional [34] and Euler's elastica based functionals [31]. In this paper, we apply this method to deal with a mean curvature based image denoising model, which is nontrivial to develop an efficient algorithm due to higher order derivative information and non-differentiability of the model. The numerical experiments demonstrate that the proposed augmented Lagrangian method is much faster than the gradient descent method in converging to minimizers of the mean curvature based functional.

Acknowledgments

The authors thank Dr. Rongjie Lai for his Matlab code for the LLT model and helpful discussion on this project.

References

[1] L. Ambrosio and S. Masnou, *A direct variational approach to a problem arising in image reconstruction,* Interfaces Free Bound., 5, pp. 63-81, 2003.

[2] L. Ambrosio and S. Masnou, *On a variational problem arising in image reconstruction,* Vol. 147, Birkhauser, Basel, 2004.

[3] L. Alvarez, F. Guichard, P.L. Lions, and J.M. Morel, *Axioms and fundamental equations of image-processing,* Archive for Rational Mechanics and Analysis, Vol 123, pp. 199-257, 1993.

[4] G. Aubert and L. Vese, *A variational method in image recovery,* SIAM J. Numer. Anal. Vol 34(5), pp. 1948-1979, 1997.

[5] A.L. Bertozzi and J.B. Greer, *Low curvature image simplifiers: Global regularity of smooth solutions and Laplacian limiting schemes,* Comm. Pure Appl. Math., Vol 57, pp. 764-790, 2004.

[6] T. Chan and S. Esedoglu, *Aspects of total variation regularized L^1 function approximation,* SIAM J. Appl. Math. 65(5), pp. 1817-1837, 2005.

[7] T. Chen, W. Yin, X.S. Zhou, D. Comaniciu, and T. Huang. *Total variation models for variable lighting face recognition,* IEEE. Trans. Pattern Anal. Mach. Intell., Vol 28(9), pp. 15191524, 2006.

[8] A. Chambolle and P.L. Lions, *Image recovery via total variation minimization and related problems,* Numer. Math. Vol 76, pp. 167-188, 1997.

[9] T. Chan, S.H. Kang, and J.H. Shen, *Euler's elastica and curvature based inpaintings,* SIAM J. Appl. Math, Vol. 63(2), pp. 564-592, 2002.

[10] T. Chan, A. Marquina, and P. Mulet, *High-order total variation-based image restoration,* SIAM J. Sci. Comput., Vol. 22(2), pp. 503-516, 2000.

[11] M.P. do Carmo, *Differential geometry of curves and surfaces,* Prentice-Hall, Inc., 1976.

[12] M. Elsey and S. Esedoḡlu, *Analogue of the total variation denoising model in the context of geometry processing,* SIAM J. Multiscale Modeling and Simulation Vol. 7(4), pp. 1549-1573, 2009.

[13] S. Esedoḡlu and J. Shen, *Digital inpainting based on the Mumford-Shah-Euler image model,* European J. Appl. Math., Vol. 13, pp. 353-370, 2002.

[14] J.B. Greer and A.L. Bertozzi, *Traveling Wave Solutions of Fourth Order PDEs for Image Processing,* SIAM J. Math. Anal., Vol. 36(1), pp. 38-68, 2004.

[15] J.B. Greer, A.L. Bertozzi, and G. Sapiro, *Fourth order partial differential equations on general geometries,* J. Comp. Phys., Vol. 216(1), pp. 216-246, 2006.

[16] R. Kimmel, R. Malladi, and N. Sochen, *Image Processing via the Beltrami Operator,* Proceedings of Asian Conference on Computer Vision, LNCS Vol. 1351, pp. 574-581, Hong Kong, January 1998.

[17] R. Kimmel, R. Malladi, and N. Sochen, *Images as Embedded Maps and Minimal Surfaces: Movies, Color, Texture and Volumetric Medical Images,* International Journal of Computer Vision, Vol. 39 (2), pp. 111-129, 2000.

[18] P. L. Lions and B. Mercier, *Splitting algorithms for the sume of two nonlinear opertors,* SIAM J. Numer. Amal., 16, pp. 964-979, 1979.

[19] 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., Vol. 12, pp. 1579-1590, 2003.

[20] M. Lysaker, S. Osher, and X.C. Tai, *Noise removal using smoothed normals and surface fitting,* IEEE. Trans. Image Process., Vol. 13(10), pp. 1345-1457, 2004.

[21] S. Masnou, *Disocclusion: A variational approach using level lines,* IEEE Trans. Image Process., 11, pp. 68-76, 2002.

[22] S. Masnou and J.M. Morel, *Level lines based disocclusion,* Proc. IEEE Int. Conf. on Image Processing, Chicago, IL, pp. 259-263, 1998.

[23] Y. Meyer, *Oscillating patterns in image processing and nonlinear evolution equations,* University Lecture Series, Vol 22, Amer. Math. Soc.

[24] D. Mumford and J. Shah, *Optimal approximation by piecewise smooth functions and associated variational problems,* Comm. Pure Appl. Math., Vol. 42 pp. 577-685, 1989.

[25] J.M. Morel and S. Solimini, *Variational methods in image segmentation,* Birkhauser, Boston, 1995.

[26] M. Nitzberg, D. Mumford, and T. Shiota, *Filering, segmentation, and depth,* Lecture Notes in Computer Science, Vol. 662, Springer Verlag, Berlin, 1993.

[27] S. Osher, A. Sole, and L. Vese, *Image decomposition and restoration using total variation minimization and the H^{-1}norm,* SIAM Multiscale Model. Simul. Vol 1, pp. 349-370, 2003.

[28] P. Perona and J. Malik, *Scale-space and edge-detection using anisotropic diffusion,* IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 12(7), pp. 629-639, 1990.

[29] L. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithm,* Physica D, Vol. 60, pp. 259-268.

[30] N. Sochen, R. Kimmel, and R. Malladi, *A geometrical framework for low level vision,* IEEE Trans. on Image Process., Vol. 7(3), pp. 310-318, 1998.

[31] X.C. Tai, J. Hahn, and G.J. Chung, *A fast algorithm for Euler's Elastica model using augmented Lagrangian method,* SIAM J. Imaging Sciences, Vol 4(1), pp. 313-344, 2011.

[32] T. Tasdizen, R. Whitaker, P. Burchard, and S. Osher, *Geometric surface processing via normal maps,* ACM Transactions on Graphics, Vol 22, pp. 1012 - 1033, 2003.

[33] L. Vese and S. Osher, *Modeling textures with total variation minimization and oscillatory patterns im image processing,* SINUM. Vol 40, pp. 2085-2104, 2003.

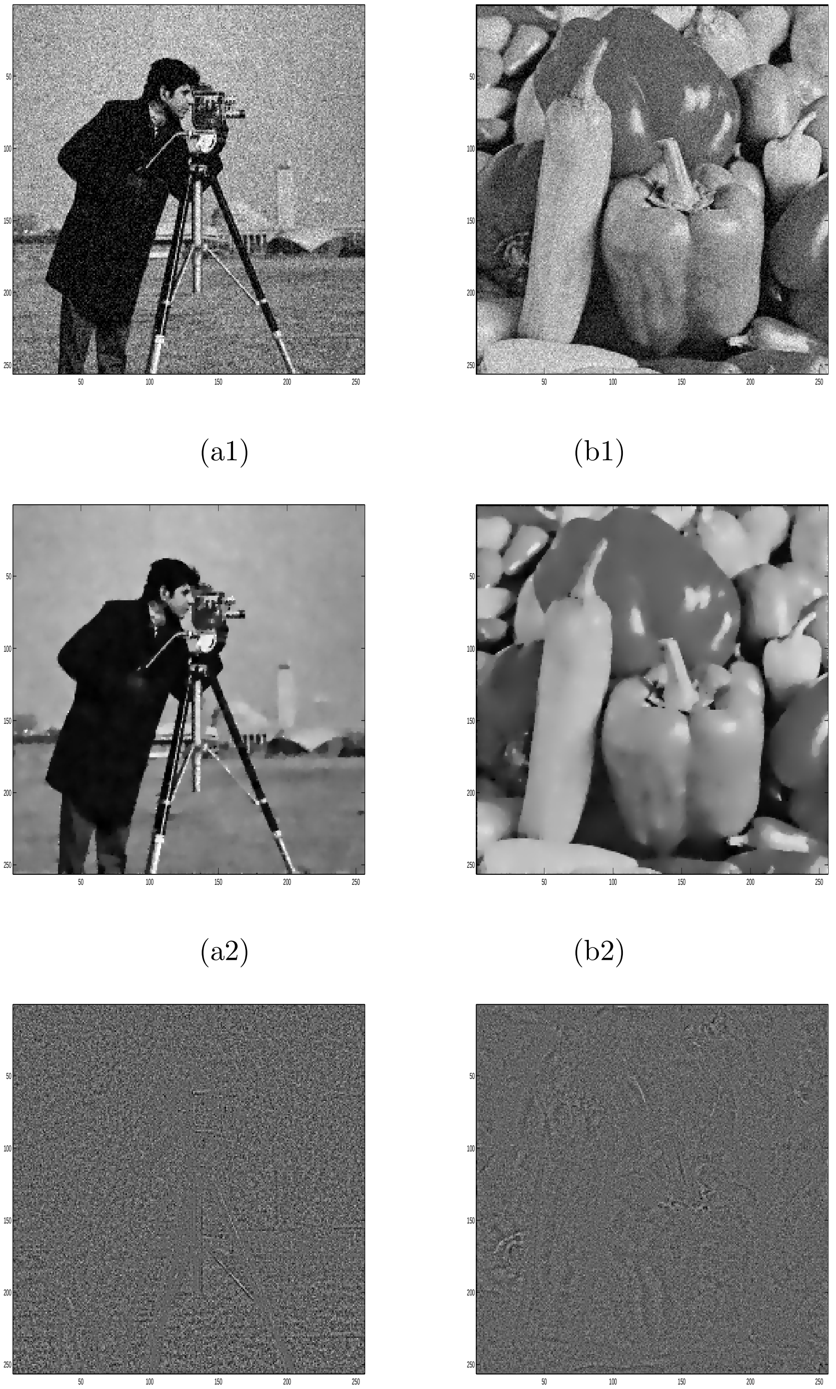
[34] C. Wu and X.C. Tai, *Augmented Lagrangian method, dual methods, and split Bregman iteration for ROF, Vectorial TV, and high order models,* SIAM J. Imaging Sciences, Vol 3(3), pp. 300-339, 2010.

[35] W. Yin, T. Chen, X.S. Zhou, and A. Chakraborty. *Background correction for cDNA microarray image using the TV+L1 model,* Bioinformatics, Vol 21(10), pp.24102416, 2005.

[36] W. Zhu and T. Chan, *A variational model for capturing illusory contours using curvature,* J. Math. Imaging Vision, 27, pp. 29-40, 2007.

[37] W. Zhu, T. Chan, and S. Esedoḡlu, *Segmentation with depth: a level set approach,* SIAM J. Sci. Comput. 28, pp. 1957-1973, 2006.

[38] W. Zhu and T. Chan, *Image denoising using mean curvature of image surface,* Accepted by SIAM J. Imaging Sciences.



(a1) (b1)

(a2) (b2)

(a3) (b3)

Figure 1: The denoising results for the images ”Cameraman” and ”peppers”. The noisy, cleaned, and residual images are listed from the first row to the third row respectively. For both examples, we set the spatial step size $h = 5.0$, choose $r_1 = 40$, $r_2 = 40$, $r_3 = 1 \times 10^5$, and the remaining parameters $r_4 = 1.5 \times 10^5$ and $\epsilon_r = 6 \times 10^{-3}$ for ”Cameraman” while $r_4 = 10^5$ and $\epsilon_r = 6 \times 10^{-3}$ for ”peppers”.

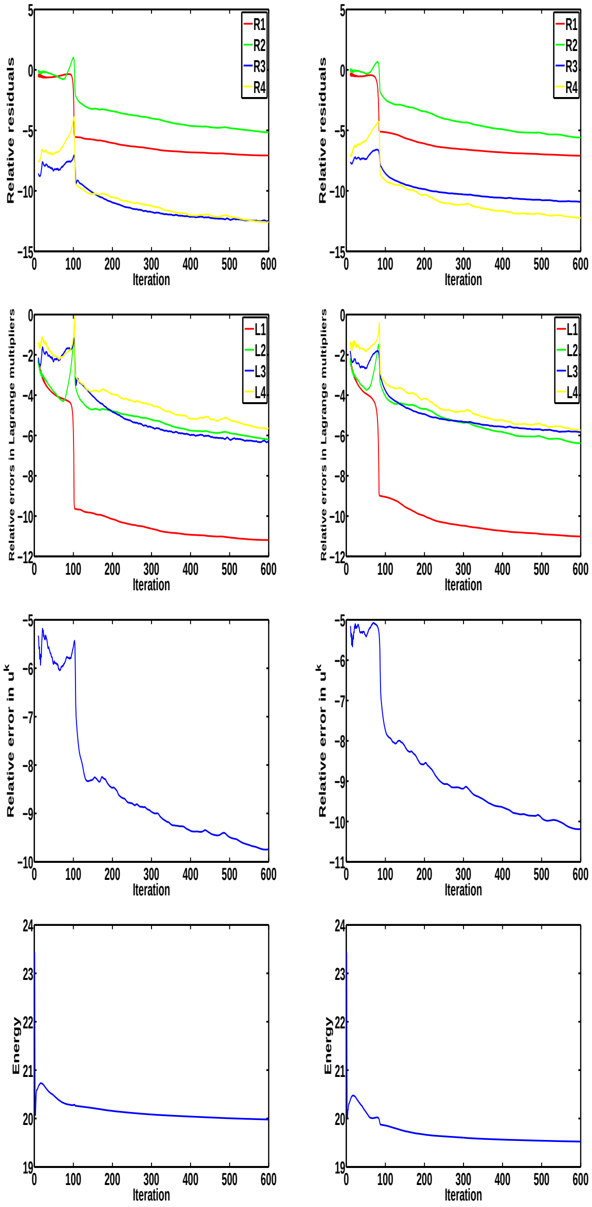
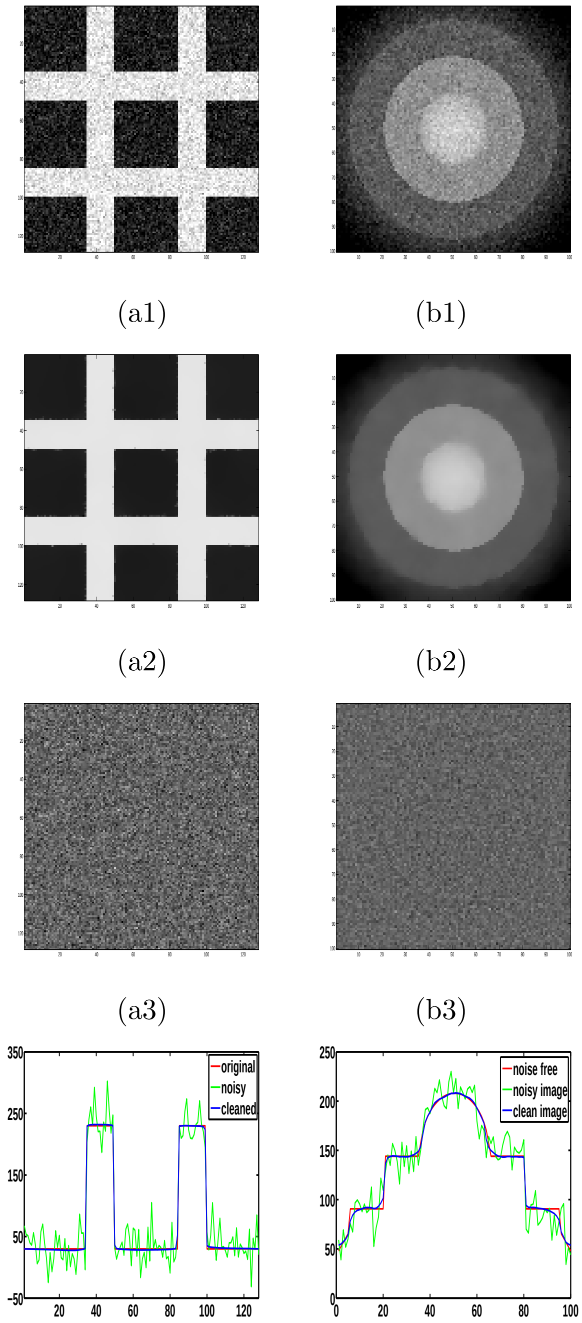


Figure 2: The plots of relative residuals, relative errors in Lagrange multipliers, relative error in u^k and energy for the previous examples "Cameraman" and "peppers". The left column lists the plots for "Cameraman" while the right one for "peppers". The plots demonstrate the converges of the iteration process, showing that some saddle points will be achieved.



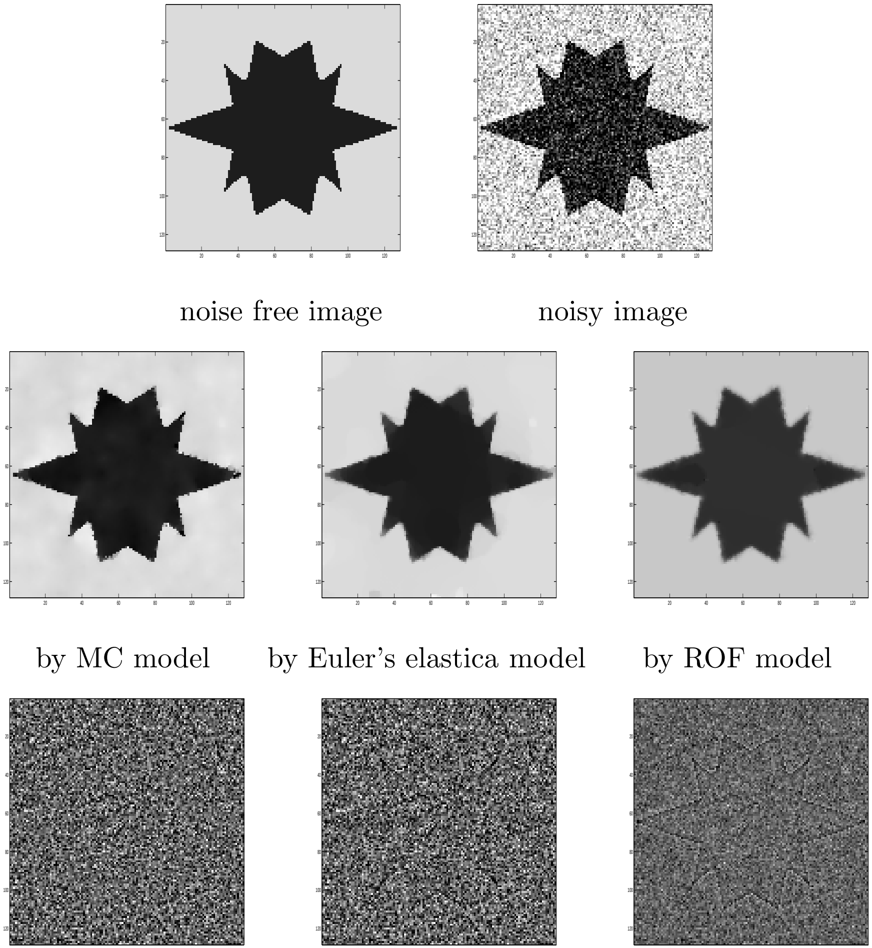
(a1) (b1)

(a2) (b2)

(a3) (b3)

(a4) (b4)

Figure 3: The denosing results for two synthetic images ”lattices” and ”rings”. The noisy, cleaned, residual images are listed from the first row to the third row respectively. In the fourth row, a slice of the original noise free image, noisy and the cleaned images for each example is presented. For the ”lattices” example, we choose the parameters as follows: $h = 1.0$, $r_1 = 20$, $r_2 = 20$, $r_3 = 1 \times 10^4$, $r_4 = 1 \times 10^5$ and $\epsilon_r = 5 \times 10^{-3}$, while for the ”rings” example, $h = 5.0$, $r_1 = 20$, $r_2 = 20$, $r_3 = 1 \times 10^5$, $r_4 = 2 \times 10^5$ and $\epsilon_r = 8 \times 10^{-3}$.

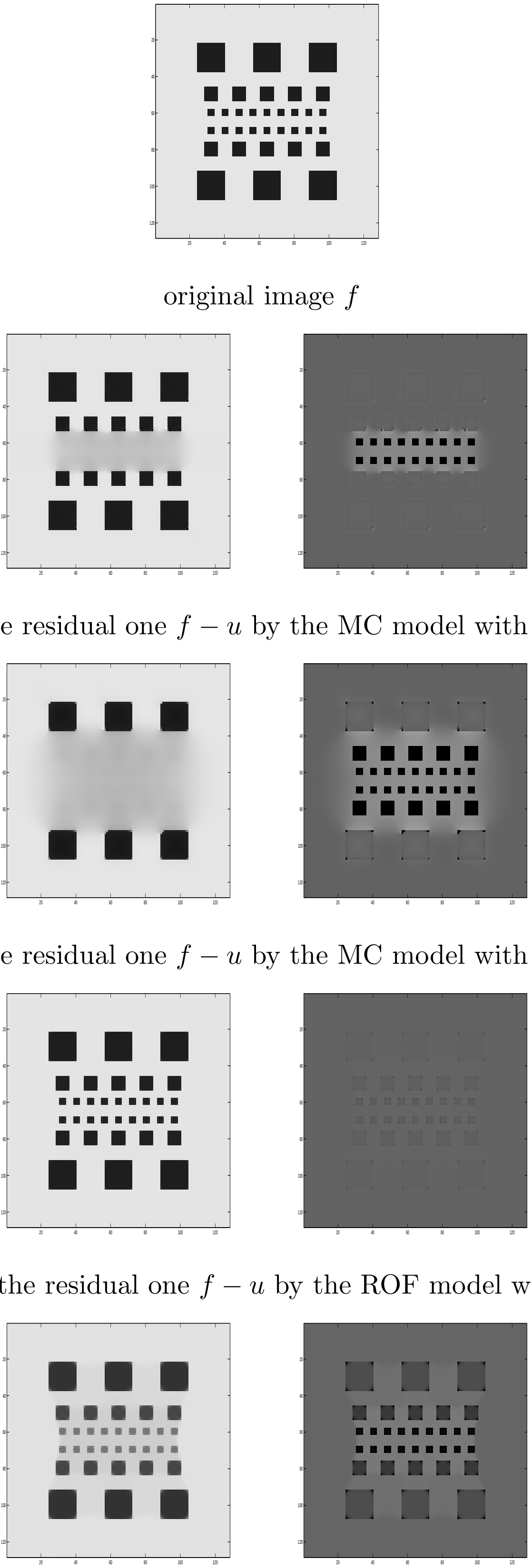


noise free image   noisy image

by MC model   by Euler's elastica model   by ROF model

by MC model   by Euler's elastica model   by ROF model

Figure 4: The comparison of three models: the mean curvature (MC) model, the Euler's elastica model and the ROF model. The first row lists the noise free image and the noisy one. The second row presents the results by applying the above three models respectively, while the associated residual images are shown in the third row. The results illustrate that the MC model keeps corners better than the other two models, which can be observed in the related residual images where some corner signals exist explicitly for the two models. Moreover, the cleaned image by the ROF shows less darkness of the object than that of the MC model, which suggests that the MC model preserves image contrast better than the ROF model.



original image f

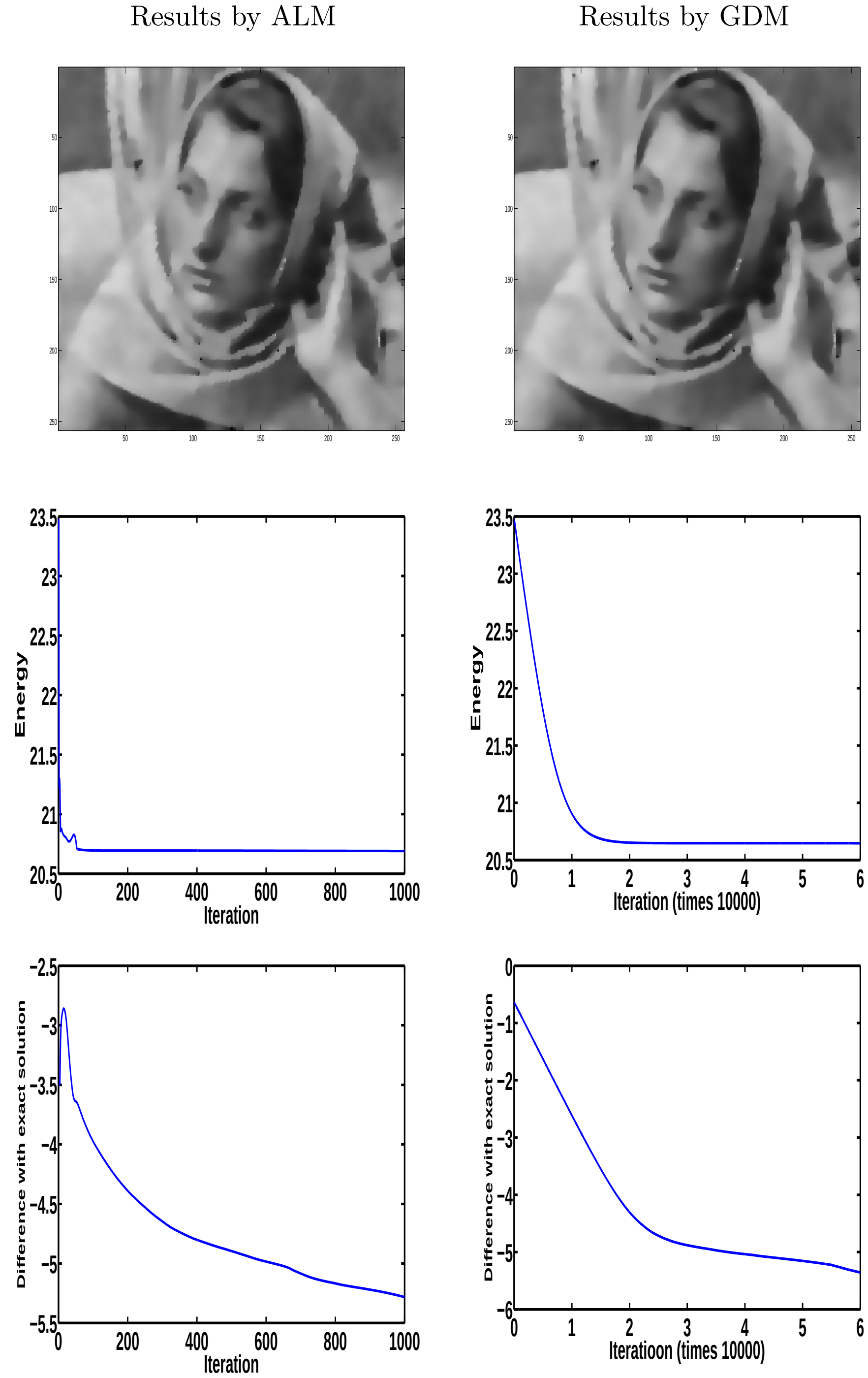
clean u and the residual one $f - u$ by the MC model with $\lambda = 2.5 \times 10^4$

clean u and the residual one $f - u$ by the MC model with $\lambda = 5.0 \times 10^4$

clean u and the residual one $f - u$ by the ROF model with $\lambda = 10.0$

clean u and the residual one $f - u$ by the ROF model with $\lambda = 100.0$

Figure 5: The dependence of the regularization parameter for the MC model and the ROF model. The results demonstrate that the MC model is lack of continuous dependence on the regularization parameter, which is not possessed by the ROF model.


Results by ALM

Results by GDM

Figure 6: The efficiency comparison of ALM and GDM for the smoothed MC model. The first row lists the two results by both algorithms; the second row shows how the energy is evolving for the two algorithms; and the third row presents the L^2-norm of the difference between the calculated solution and the exact solution, which is calculated with high accuracy. The common parameters used for these two methods are: $\lambda = 5 \times 10^3$, $h = 5.0$, and $\epsilon = 0.2$. The parameters for ALM are $r_1 = 50, r_2 = 100, r_3 = 5 \times 10^4$, and $r_4 = 5 \times 10^4$. The time step size for GDM is $dt = 2 \times 10^{-4}$.