รายงานวิจัยฉบับสมบูรณ์

โครงการ ตัวแบบเชิงการแปรผันอันดับสูงชนิดใหม่ สำหรับการประมวลผลภาพ

โดย ผู้ช่วยศาสตราจารย์ ดร.นพดล ชุมชอบ

เมษายน 2562

สัญญาเลขที่ MRG6080169

รายงานวิจัยฉบับสมบูรณ์

โครงการ ตัวแบบเชิงการแปรผันชนิดใหม่
สำหรับการประมวลผลภาพ

ผู้ช่วยศาสตราจารย์ ดร.นพดล ชุมชอบ (หัวหน้าโครงการ)
ภาควิชาคณิตศาสตร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

รองศาสตราจารย์ ดร. จันทนา จันทราพรชัย (นักวิจัยที่ปรึกษา)
ภาควิชาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเกษตรศาสตร์

สนับสนุนโดยสำนักงานกองทุนสนับสนุนการวิจัย
และมหาวิทยาลัยศิลปากร

กิตติกรรมประกาศ

กระผมขอแสดงความขอบคุณสำนักงานกองทุนสนับสนุนการวิจัยและมหาวิทยาลัยศิลปากรเป็นอย่างสูงที่ได้สนับสนุนทุนวิจัยให้กระผมตามสัญญาเลขที่ MRG6080169 ขอบคุณรองศาสตราจารย์ ดร. จันทนา จันทราพรชัย นักวิจัยที่ปรึกษาซึ่งได้สนับสนุนให้กระผมมีโอกาสได้ทำงานวิจัยชิ้นนี้รวมไปถึงกำลังใจและคำแนะนำที่เป็นประโยชน์ตลอดระยะเวลาการทำวิจัยจนได้ผลการวิจัยสำเร็จเป็นไปดังเป้าหมายที่กำหนดไว้ นอกจากนี้กระผมขอขอบคุณ นางสาวพรพิมล สร้อยสังวาลย์ ผู้ช่วยวิจัยและนักศึกษาในที่ปรึกษาวิทยานิพนธ์ระดับบัณฑิตศึกษา หลักสูตร วิทยาศาสตรมหาบัณฑิต สาชาวิชาคณิตศาสตร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร ที่มีส่วนช่วยเหลือในการทำการทดลองเชิงตัวเลขให้เป็นไปได้โดยสะดวกและมีประสิทธิภาพ

นพดล ชุมชอบ

Abstract

Project Code : MRG6080169

Project Title : A novel high-order variational model for image processing

Investigator : Assistant Professor Noppadol Chumchob, Ph.D.
Department of Mathematics, Faculty of Science, Silpakorn University

E-mail Address : chumchob_n@silpakorn.edu, chumchob@gmail.com

Project Period : 2 years (April 3, 2017 – April 3, 2019)

The classical total variation (TV) model has made great successes in image denoising due to the edge-preservation property of the TV regularization. However, it is well-know that the TV model suffers from the staircase effects and loss of image details. In order to overcome these problems, this project presents a new curvature-based model for image denoising. To find a minimizer of our proposed model, we develop an efficient numerical algorithm in the split Bregman framework for solving the associated minimization problem. We experimentally demonstrate the effectiveness and efficiency of the proposed techniques over existing high-order variational models.

Keywords: image restoration, mean curvature, Gaussian curvature, high-order regularization, split Bregman

บทคัดย่อ

รหัสโครงการ: MRG6080169

ชื่อโครงการ: ตัวแบบเชิงการแปรผันอันดับสูงชนิดใหม่สำหรับการประมวลผลภาพ

ชื่อนักวิจัย: ผู้ช่วยศาสตราจารย์ ดร.นพดล ชุมชอบ
ภาควิชาคณิตศาสตร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

E-mail Address : chumchob_n@silpakorn.edu, chumchob@gmail.com

ระยะเวลาโครงการ: 2 ปี (3 เมษายน 2560 – 3 เมษายน 2562)

ตัวแบบการแปรผันรวมแบบดั้งเดิมประสบความสำเร็จอย่างสูงในการกำจัดสัญญาณรบกวนออกจากภาพ เนื่องจากสมบัติการอนุรักษ์เส้นของวิธีการเร็กกิวลาร์ไรซ์เซชันแบบแปรผันรวม อย่างไรก็ตาม เป็นที่ทราบกันดีว่าตัวแบบการแปรผันรวมมีจุดด้อยจากผลกระทบขั้นบันได ซึ่งทำให้สูญเสียรายละเอียดของภาพ เพื่อแก้ปัญหาเหล่านี้ โครงการวิจัยนี้ได้นำเสนอตัวแบบที่ใช้เคริฟเวเจอร์ชนิดใหม่สำหรับการกำจัดสัญญาณรบกวนออกจากภาพ ในการหามินิไมซ์เซอร์ของตัวแบบใหม่ที่นำเสนอ โครงการวิจัยนี้ได้พัฒนาขั้นตอนวิธีเชิงตัวเลขที่มีประสิทธิภาพในกรอบการทำงานแบบสปริทเบรกแมนสำหรับแก้ปัญหาการหาค่าต่ำสุดที่เกี่ยวข้อง เราได้ทำการทดลองเพื่อแสดงประสิทธิผลและประสิทธิภาพของวิธีการที่นำเสนอโดยการเปรียบเทียบกับตัวแบบเชิงแปรผันอันดับสูงที่มีอยู่แล้ว

คำหลัก : การซ่อมแซมภาพ, เคริฟเวเจอร์, เคริฟเวเจอร์แบบเกาส์เซียน, เร็กกิวลาร์ไรซ์เซชันอันดับสูง, สปริมเบรกแมน

1. Introduction

Images generated by imaging devices are frequently contaminated by noise during image acquisition and transmission processes. Thus the task of image denoising, consisting of removing electronic noise, which is a random variation of brightness or color information, from the image signal, has become a highly desirable and essential pre-processing technique in several areas of multimedia applications. Especially in medical application, for example, it has been used routinely to enhance the quality and information content in clinical images by estimating the original clean image from a recorded noisy image without losing significant features such as edges and texture details.

Over the last decades, a variety of methods have been developed to deal with the image denoising problems. Roughly speaking, these denoising methods fall into five categories: 1) filtering based methods in spatial domain [1, 2, 3, 4] and transformed domain, e.g. wavelet domain [5, 6, 7]; 2) nonlocal filtering methods [8, 9, 10]; 3) anisotropic diffusion methods [11, 12]; 4) deep convolution neural network methods [13, 14, 15, 16]; and 5) variational methods [17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31]. Among these methods, the variational methods are one of the most powerful techniques to offer high quality of denoised images. Accordingly, this work focuses on the variational method based on high-order regularizations.

In the next subsections, we present a variational formulation and review briefly several existing variational models for image denoising problems.

1.1. Variational formulation

Let Ω be a bounded domain of $\mathbb{R}^2$ (a rectangle in practice). We consider the problem of recovering the original image $u : \Omega \to \mathbb{R}$ from a recorded noisy image $z : \Omega \to \mathbb{R}$. Generally, most of the noise obtained during acquisition and transmission of the natural images is assumed to be additive noise (AN). Thus, a recorded noisy image z can be mathematically modeled with the equation

$$z = u + \eta, \tag{1}$$

where $\eta : \Omega \to \mathbb{R}$ is the zero-mean Gaussian white noise with a given variance σ.

It is well known that the problem of recovering u from z by using the AN model (1) is an ill-posed inverse problem because the solution is very sensitive (unstable) to the noise. In this case, a widely common procedure in inverse problems is to use a regularization for computing the approximate solution that is less sensitive to the noise than the naive solution. From the variational point of view, the task of removing noise can be accomplished by solving

a minimization problem such as

$$\min_u \{\mathcal{J}(u) = \mathcal{D}_{\mathrm{AN}}(u,z) + \gamma\mathcal{R}(u)\}, \tag{2}$$

where $\mathcal{D}_{\mathrm{AN}}(u,z) = \frac{1}{2}\int_\Omega (u-z)^2 d\Omega$ is the fidelity term, which is derived from the AN model (1) and used to measure the similarity between the denoised image u and the recorded noisy image z, $\mathcal{R}(u) = \int_\Omega F_{\mathcal{R}}(x,y,u,D_1u,D_2u,\cdots,D_iu,\cdots)d\Omega$ is the regularization term, which is used to filter out the noise from the recorded noisy image z as well as to preserve significant features such as edges and textures of the denoised image u, and $\gamma > 0$ is the regularization parameter which compromises the fidelity term $\mathcal{D}_{\mathrm{AN}}(u,z)$ and the regularization term $\mathcal{R}(u)$. Note that $D_iu = \left[\frac{\partial^i u}{\partial x^i}, \frac{\partial^i u}{\partial y^i}\right]^\top$ and fractional derivatives of u may occur in $F_{\mathcal{R}}$.

1.2. A review of existing high-order variational models

Rudin, Osher, and Fatemi [32] proposed one of the most popular variational models. Their variational model, also known as the total variation (TV) model, is given by

$$\min_u \left\{\mathcal{J}^{\mathrm{TV}}(u) = \mathcal{D}_{\mathrm{AN}}(u,z) + \gamma\mathcal{R}^{\mathrm{TV}}(u)\right\}, \tag{3}$$

involving the TV regularization $\mathcal{R}^{\mathrm{TV}}(u) = \int_\Omega |\nabla u| \, d\Omega$. Although TV regularization is effective in preserving sharp edges while removing unwanted oscillations and noise, it has some undesired effects. In particular, it transforms smooth signals or signals that are not necessarily piecewise constant into piecewise constants. This phenomenon is known as staircase effect. From a practical point of view, staircase solutions of (3) fail to satisfy the evaluation of visual quality, and they can develop false edges that do not exist in the true image. The main challenge is thus to deal with the staircase effect while preserving image sharpness.

To overcome the drawback of the TV regularization, several high-order regularizations have been proposed so far in the literature to yield better quality of denoised images. These high-order regularizations usually involve second-order differential operators in dealing with the staircase effect because piecewise-vanishing second-order derivatives yield piecewise-linear solutions that better fit than those by the TV regularization for smooth intensity changes. To the best of our knowledge, there are two main classes of high-order regularizations for image denoising problems. The first class combines a second-order regularization with the TV regularization. The second class employs a second-order regularization in a standalone way. Below, we present a bird's eye view of specific variational models involving the high-order regularizations.

You and Kaveh [17] proposed the total Laplace (TL) model

$$\min_u \left\{\mathcal{J}^{\mathrm{TL}}_{\mathrm{AN}}(u) = \mathcal{D}_{\mathrm{AN}}(u,z) + \gamma\mathcal{R}^{\mathrm{TL}}(u)\right\} \tag{4}$$

with the TL regularization $\mathcal{R}^{\mathrm{TL}}(u) = \int_\Omega |\triangle u| d\Omega = \int_\Omega \sqrt{|u_{xx}|^2 + |u_{yy}|^2} d\Omega$, where $\triangle u$ denotes the Laplacian operator. Scherzer [18], Lysaker et al. [19], Hinterberger and Scherzer [20], Bergounioux and Piffet [21] and Lai et al. [22] considered the bounded Hessian (BH) model

$$\min_u \left\{ \mathcal{J}_{\mathrm{AN}}^{\mathrm{BH}}(u) = \mathcal{D}_{\mathrm{AN}}(u, z) + \gamma \mathcal{R}^{\mathrm{BH}}(u) \right\} \tag{5}$$

with the BH regularization $\mathcal{R}^{\mathrm{BH}}(u) = \int_\Omega |\nabla^2 u| d\Omega$, where

$$\nabla^2 u = \begin{pmatrix} u_{xx} & u_{yx} \\ u_{xy} & u_{yy} \end{pmatrix}$$

represents the Hessian matrix of u and $|\nabla^2 u| = \sqrt{u_{xx}^2 + u_{yx}^2 + u_{xy}^2 + u_{yy}^2}$. Chan et al. [23] introduced the CEP-L^2 model

$$\min_{u_1, u_2} \left\{ \mathcal{J}^{\text{CEP-L}^2}(u_1, u_2) = \bar{\mathcal{D}}_{\mathrm{AN}}(u_1, u_2, z) + \mathcal{R}^{\text{CEP-L}^2}(u_1, u_2) \right\} \tag{6}$$

with the CEP-L^2 regularization $\mathcal{R}^{\text{CEP-L}^2_{\mathrm{AN}}}(u_1, u_2) = \gamma_1 \mathcal{R}^{\mathrm{TV}}(u_1) + \gamma_2 \mathcal{R}^{\mathrm{TL}}(u_2)$ where $\gamma_1 > 0$ and $\gamma_2 > 0$ are the weighting parameters that balance the first- and second-order derivative terms. Note that the noisy image z is decomposed into three parts, i.e. $z = u_1 + u_2 + \eta$ and the associated fidelity term is given by

$$\bar{\mathcal{D}}_{\mathrm{AN}}(u_1, u_2, z) = \frac{1}{2} \int_\Omega (z - u_1 - u_2)^2 \, d\Omega.$$

Zheng et al. [24], Wang et al. [25] and Chan et al. [26] considered the TVL model

$$\min_u \left\{ \mathcal{J}_{\mathrm{AN}}^{\mathrm{TVL}}(u) = \mathcal{D}_{\mathrm{AN}}(u, z) + \mathcal{R}^{\mathrm{TVL}}(u) \right\} \tag{7}$$

with the TVL regularization $\mathcal{R}^{\mathrm{TVL}}(u) = \gamma_1 \mathcal{R}^{\mathrm{TV}}(u) + \gamma_2 \mathcal{R}^{\mathrm{TL}}(u)$. Papafitsoros and Schönlieb [27] introduced the TVBH model

$$\min_u \left\{ \mathcal{J}_{\mathrm{AN}}^{\mathrm{TVBH}}(u) = \mathcal{D}_{\mathrm{AN}}(u, z) + \mathcal{R}^{\mathrm{TVBH}}(u) \right\} \tag{8}$$

with the TVBH regularization $\mathcal{R}^{\mathrm{TVBH}}(u) = \gamma_1 \mathcal{R}^{\mathrm{TV}}(u) + \gamma_2 \mathcal{R}^{\mathrm{BH}}(u)$. Chambolle and Lions [28] proposed the INFCON model

$$\min_{u_1, u_2} \left\{ \mathcal{J}_{\mathrm{AN}}^{\mathrm{TVBH}}(u_1, u_2) = \bar{\mathcal{D}}_{\mathrm{AN}}(u_1, u_2, z) + \mathcal{R}^{\mathrm{TVBH}}(u_1, u_2) \right\}. \tag{9}$$

with the INFCON regularization $\mathcal{R}^{\mathrm{TVBH}}(u_1, u_2) = \gamma_1 \mathcal{R}^{\mathrm{TV}}(u_1) + \gamma_2 \mathcal{R}^{\mathrm{BH}}(u_2)$ deriving from an inf-convolution of two convex regularizations. Bredies and et al. [29] proposed the total generalized variation (TGV) model

$$\min_{u, \tilde{\boldsymbol{p}}} \left\{ \mathcal{J}_{\mathrm{AN}}^{\mathrm{TGV}}(u, \tilde{\boldsymbol{p}}) = \mathcal{D}_{\mathrm{AN}}(u, z) + \mathcal{R}^{\mathrm{TGV}}(u, \tilde{\boldsymbol{p}}) \right\} \tag{10}$$

with the TGV regularization

$$\mathcal{R}^{\mathrm{TGV}}(u,\tilde{\boldsymbol{p}}) = \gamma_1 \int_\Omega |\nabla u - \tilde{\boldsymbol{p}})|\ d\Omega + \gamma_2 \int_\Omega |\epsilon(\tilde{\boldsymbol{p}})|\ d\Omega,$$

where $\epsilon(\tilde{\boldsymbol{p}})$ is the symmetrized derivatives defined as

$$\epsilon(\tilde{\boldsymbol{p}}) = \begin{pmatrix} \tilde{p}_{1_x} & \frac{\tilde{p}_{1_y}+\tilde{p}_{2_x}}{2} \\ \frac{\tilde{p}_{1_y}\tilde{p}_{2_x}}{2} & \tilde{p}_{2_y} \end{pmatrix}.$$

Note that the minimum of (11) is taken over all gradients of the deformation field $\tilde{\boldsymbol{p}} = (\tilde{p}_1, \tilde{p}_2)$ on image surface Ω. Here $\tilde{p}_x = \partial_x \tilde{p}_l$ and $\tilde{p}_y = \partial_y \tilde{p}_l$ for $l = 1, 2$. Zhu and Chan [30] proposed the mean curvature (MC) model

$$\min_u \left\{ \mathcal{J}^{\mathrm{MC}}_{\mathrm{AN}}(u) = \mathcal{D}_{\mathrm{AN}}(u,z) + \gamma \mathcal{R}^{\mathrm{MC}}(u) \right\}, \tag{11}$$

with the MC regularization $\mathcal{R}^{\mathrm{MC}}(u) = \int_\Omega |\kappa_{\mathrm{M}}(u)|\ d\Omega$ where

$$\kappa_{\mathrm{M}}(u) = \nabla \cdot \left(\frac{\nabla u}{\sqrt{1+|\nabla u|^2}}\right) \tag{12}$$

is the mean curvature of image surface $\phi(x,y,z) = u(x,y) - z = 0$. This model tries to fit the given noisy image surface $(x, y, z(x, y))$ with a surface $(x, y, u(x, y))$ that bears small magnitude of mean curvature. As demonstrated in [30], the MC model is able to sweep noise while keeping object edges, and it also ameliorates the staircase effect. More importantly, the MC model is also capable of preserving image contrasts as well as geometry of object shapes, especially object corners. Brito-Loeza and Chen [31] proposed the Gaussian curvature (GC) model

$$\min_u \left\{ \mathcal{J}^{\mathrm{GC}}_{\mathrm{AN}}(u) = \mathcal{D}_{\mathrm{AN}}(u,z) + \gamma \mathcal{R}^{\mathrm{GC}}(u) \right\} \tag{13}$$

with the GC regularization $\mathcal{R}^{\mathrm{GC}}(u) = \int_\Omega |\kappa_{\mathrm{G}}(u)|\ d\Omega$ where

$$\kappa_{\mathrm{G}}(u) = \frac{u_{xx}u_{yy} - u_{xy}u_{yx}}{(1+|\nabla u|^2)^2} \tag{14}$$

is the Gaussian curvature of image surface $\phi(x,y,z) = u(x,y) - z = 0$. This model is able to preserve image contrast, edges, and corners such as the MC model does.

Non-surprisingly, the choice of the regularization significantly affects the quality of the denoised image. In particular, the high-order regularizations mentioned above are better than the TV one in reducing staircase effects and achieving good trade-off between noise removal and edge preservation. Some comparative results of high-order regularizations can be seen in [33, 34, 31].

1.3. Contribution

The main contribution of this work is to propose a new variational model for image denoising. Our new model is designed to utilize the favorable denoising properties of the high-order regularization deriving from the sum of squared principal curvatures of the image surface. As a result, our variational model is difficult to solve by standard techniques such as gradient descent methods. To simplify the associated minimization problem, we propose to use the split Bregman (SB) method. The main idea is to introduce several auxiliary variables for converting the complex operation into an alternating iterative process by several simple operations, which are easy to implement and have high computational efficiency. We note that the new variational model has several advantages. Firstly, it is more effective than existing high-order variational models mentioned above to remove noise and preserve edge. Secondly, as we shall demonstrate, our denoising algorithm to be introduced in Section 3 is able to converge in a reasonable number of iterations. Finally, we found what we expected in numerical experiments on real noisy images that it is able to deliver visually pleasing results.

The remainder of this report is organized as follows. In Section 2, we introduce a new curvature-based regularization. Then, our new high-order variational model for image denoising proposed. The denoising algorithm using the SB iteration is introduced in Section 3. Experiments results from synthetic and real images are given in Section 4. In Section 5, the concluding remarks are drawn.

2. The proposed model

Inspired by the attractive property of the high-order regularizations mentioned in Section 1.2 for dealing with the staircase effects from the noise reduction process, we propose our new high-order variational model for image denoising as follows.

Let $\kappa_1(u) = \kappa_1(\mathbf{x}, u(\mathbf{x}))$ and $\kappa_2(u) = \kappa_2(\mathbf{x}, u(\mathbf{x}))$ be the principal curvatures of the image surface induced by a given image $u = u(\mathbf{x})$ through the mapping $(\mathbf{x}, u(\mathbf{x}))$. According to theory of surfaces, the mean curvature $\kappa_{\mathrm{M}}(u)$ in (12) and the Gaussian curvature $\kappa_{\mathrm{G}}(u)$ in (14) are related to the principal curvatures $\kappa_1(u)$ and $\kappa_1(u)$ by

$$\kappa_{\mathrm{M}}(u) = \kappa_1(u) + \kappa_2(u) \text{ and } \kappa_{\mathrm{G}}(u) = \kappa_1(u)\kappa_2(u).$$

Therefore, the sum of the squares of the principal curvatures of the image surface are related

to the mean and Gaussian curvatures as follows.

$$\begin{aligned}\kappa_1^2(u)+\kappa_2^2(u) &= (\kappa_1(u)+\kappa_2(u))^2 - 2\kappa_1(u)\kappa_2(u)\\ &= \kappa_{\mathrm{M}}^2(u) - 2\kappa_{\mathrm{G}}(u)\\ &= \left(\frac{(1+u_x^2)u_{yy} - 2u_xu_yu_{xy} + (1+u_y^2)u_{xx}}{(1+|\nabla u|^2)^{3/2}}\right)^2 - 2\frac{u_{xx}u_{yy}-u_{xy}u_{yx}}{(1+|\nabla u|^2)^2}. \end{aligned} \tag{15}$$

We note that the authors in [30] use

$$\kappa_{\mathrm{M}}(u) = \frac{\kappa_1(u)+\kappa_2(u)}{2} = \frac{(1+u_x^2)u_{yy} - 2u_xu_yu_{xy} + (1+u_y^2)u_{xx}}{2(1+|\nabla u|^2)^{3/2}},$$

justifying the name 'mean' curvature. We also note that the sum of the squares of the principal curvatures of the image surface for the case $|\nabla u| \approx 0$ can be approximated by

$$\kappa_1^2(u)+\kappa_2^2(u) \approx \triangle^2 u - 2\left(u_{xx}u_{yy} - u_{xy}^2\right).$$

Carefully considering, we have $\kappa_{\mathrm{M}}(u_0) = 0$ and $\kappa_{\mathrm{G}}(u_0) = 0$ for piecewise constant or piecewise affine images u_0. This motivates us to propose the following functional as the new regularization

$$\mathcal{R}^{\mathrm{NewCv}}(u) = \frac{1}{2}\int_\Omega \left(\kappa_1^2(u)+\kappa_2^2(u)\right) d\Omega = \frac{1}{2}\int_\Omega \left(\kappa_{\mathrm{M}}^2(u) - 2\kappa_{\mathrm{G}}(u)\right) d\Omega. \tag{16}$$

As can be seen, the kernel of $\mathcal{R}^{\mathrm{NewCv}}$ denoted by $\ker\left(\mathcal{R}^{\mathrm{NewCv}}\right) = \left\{u : \Omega \to \mathbb{R} | \mathcal{R}^{\mathrm{New}}(u) = 0\right\}$ is given by

$$\ker\left(\mathcal{R}^{\mathrm{NewCv}}(u)\right) = \left\{u(\mathbf{x}) = \mathbf{A}\mathbf{x} + b \text{ a.e. in } \Omega \text{ for some } \mathbf{A} \in \mathbb{R}^{1\times 2},\ \mathbf{x} = (x,y)^\top \in \mathbb{R}^2,\ b \in \mathbb{R}\right\}.$$

It follows that $\mathcal{R}^{\mathrm{NewCv}}$ shares this attractive property with the MC and GC models in dealing with different image characteristics such as piecewise constant and piecewise affine images. We note that $\mathcal{R}^{\mathrm{NewCv}}$ is related to the bending energy used to measure surface fairing and the strain energy of a thin elastic plate, which has lots of applications in computer graphics [35].

In summary, the new high-order variational model with the regularization (16) is then given by

$$\min_u \left\{\mathcal{J}_{\mathrm{AN}}^{\mathrm{NewCv}}(u) = \mathcal{D}_{\mathrm{AN}}(u,z) + \gamma\mathcal{R}^{\mathrm{NewCv}}(u)\right\}. \tag{17}$$

3. Solution of the proposed model

To find the solution of the new model (17), ones may directly solve the corresponding Euler-Lagrange (EL) equation, which can be derived similarly as in [36] (see in Appendix A) and given by

$$f(u,z) + \gamma\nabla\cdot\boldsymbol{V} = 0 \tag{18}$$

subject to the natural boundary conditions

$$\kappa_{\mathrm{M}}(u) = 0, \tag{19}$$

$$-\frac{\partial \kappa_{\mathrm{M}}(u)}{\partial \boldsymbol{n}} + \frac{\nabla \kappa_{\mathrm{M}}(u) \cdot \nabla u}{(1+|\nabla u|^2)^2} \frac{\partial u}{\partial \boldsymbol{n}} = 0, \tag{20}$$

$$(u_{xy}, -u_{xx}) \cdot \boldsymbol{n} = 0, \tag{21}$$

$$(-u_{yy}, u_{yx}) \cdot \boldsymbol{n} = 0, \tag{22}$$

$$((\frac{u_{yy}}{(1+|\nabla u|^2)^2})_x, -(\frac{u_{xy}}{(1+|\nabla u|^2)^2}))_x \cdot \boldsymbol{n} = 0, \tag{23}$$

$$((\frac{u_{yx}}{(1+|\nabla u|^2)^2})_y, -(\frac{u_{xx}}{(1+|\nabla u|^2)^2}))_y \cdot \boldsymbol{n} = 0, \tag{24}$$

$$\frac{4(u_{xy}u_{yx} - u_{xx}u_{yy})}{(1+|\nabla u|^2)^3} \frac{\partial u}{\partial \boldsymbol{n}} = 0. \tag{25}$$

Here $f(u,z) = (u-z)$ and $\boldsymbol{V} = \boldsymbol{V}_1 + \boldsymbol{V}_2 + \boldsymbol{V}_3 + \boldsymbol{V}_4 + \boldsymbol{V}_5$, where

$$\boldsymbol{V}_1 = \frac{1}{\sqrt{1+|\nabla u|^2}} \nabla \kappa_{\mathrm{M}}(u), \boldsymbol{V}_2 = -\frac{\nabla u \cdot \nabla \kappa_{\mathrm{M}}(u)}{(\sqrt{1+|\nabla u|^2})^3} \nabla u,$$
$$\boldsymbol{V}_3 = \frac{4(u_{xy}u_{yx} - u_{xx}u_{yy})}{(1+|\nabla u|^2)^3} \nabla u, \boldsymbol{V}_4 = ((-\frac{u_{yy}}{(1+|\nabla u|^2)^2})_x, (\frac{u_{xy}}{(1+|\nabla u|^2)^2})_x),$$
$$\boldsymbol{V}_5 = ((\frac{u_{yx}}{(1+|\nabla u|^2)^2})_y, (-\frac{u_{xx}}{(1+|\nabla u|^2)^2})_y).$$

A simple numerical technique for solving the nonlinear and fourth-order partial differential equation in (18) is to update u using the gradient descent method as follows:

$$u_t = -\left(f(u,z) + \gamma \nabla \cdot \boldsymbol{V}\right), \tag{26}$$

where $u = u(\mathbf{x}, t)$ and $t \in [0, \infty)$ is the artificial time variable. However, this technique requires a small time step and is not efficient because the time step is required to be very small for stability reasons. One way to solve it effectively is to develop a multigrid method as in [37]. Here, we consider an alternative approach in a unilevel framework.

3.1. Discretization

In order to provide the numerical algorithm for the proposed variational model (17), the finite difference discretizations of the first-oder and second-order derivative operators are required. They are respectively ∂_x^+, ∂_x^-, ∂_y^+, ∂_y^-, $\partial_x^-\partial_x^+$, $\partial_x^+\partial_x^-$, $\partial_y^-\partial_y^+$, and $\partial_y^+\partial_y^-$. Note that we apply periodic boundary condition with these discretizations. It is often used for the image denoising problem since it is able to allow fast numerical computations.

From now on, we assume $\Omega = [0, M] \times [0, N]$. Let $\bar{\Omega} \subset \Omega$ be the discretized image domain as given by

$$\bar{\Omega} = \{(x,y) \in \Omega | (x,y) = (x_i, y_j),\ x_i = i,\ y_j = j,\ 1 \le i \le M \text{ and } 1 \le j \le N)\}.$$

For simplicity, each grid point $(x_i, y_j) \in \bar{\Omega}$ is denoted by (i, j), where the coordinates x and y are oriented along columns and rows, respectively. All the variables are defined on these grid points. Let $\bar{\Omega} \to \mathbb{R}^{M\times N}$ be the 2D grayscale image space with size MN. So, the first-order forward differences of the grayscale image u at grid point (i, j) along x and y directions are respectively

$$\partial_x^+(u)_{i,j} = \begin{cases} (u)_{i,j+1} - (u)_{i,j} & \text{if } 1 \le i \le M, 1 \le j \le N \\ (u)_{i,1} - (u)_{i,j} & \text{if } 1 \le i \le M, j = N \end{cases} \tag{27}$$

and

$$\partial_y^+(u)_{i,j} = \begin{cases} (u)_{i+1,j} - (u)_{i,j} & \text{if } 1 \le i \le M, 1 \le j \le N \\ (u)_{1,j} - (u)_{i,j} & \text{if } i = M, 1 \le j \le N \end{cases}. \tag{28}$$

The first-order backward differences are respectively

$$\partial_x^-(u)_{i,j} = \begin{cases} (u)_{i,j} - (u)_{i,j-1} & \text{if } 1 \le i \le M, 1 \le j \le N \\ (u)_{i,j} - (u)_{i,N} & \text{if } 1 \le i \le M, j = 1 \end{cases} \tag{29}$$

and

$$\partial_y^-(u)_{i,j} = \begin{cases} (u)_{i,j} - (u)_{i-1,j} & \text{if } 1 \le i \le M, 1 \le j \le N \\ (u)_{i,j} - (u)_{M,j} & \text{if } i = 1, 1 \le j \le N \end{cases}. \tag{30}$$

For $\boldsymbol{p} = (p_1, p_2) \in (\mathbb{R}^{M\times N})^2$ and $u \in \mathbb{R}^{M\times N}$, the discrete version of the first-order adjoint divergence operator satisfies

$$\sum_{\substack{1\le i\le M \\ 1\le j\le M}} -(\nabla\cdot\boldsymbol{p})\cdot u = \sum_{\substack{1\le i\le M \\ 1\le j\le M}} \boldsymbol{p}\cdot(\nabla u)$$

Therefore, according to the definition of discrete gradient operator

$$(\nabla u)_{i,j} = \begin{pmatrix} \partial_x^+(u)_{i,j} \\ \partial_y^+(u)_{i,j} \end{pmatrix},$$

the discrete divergence is given by

$$(\nabla\cdot\boldsymbol{p})_{i,j} = \partial_x^+(p_1)_{i,j} + \partial_y^+(p_2)_{i,j}. \tag{31}$$

The second-order derivative operators of u at grid point (i, j) are given by

$$\partial_x^+\partial_x^-(u)_{i,j} = \partial_x^-\partial_x^+(u)_{i,j} = \begin{cases} (u)_{i,N} - 2(u)_{i,j} + (u)_{i,j+1} & \text{if } 1 \le i \le M, j = 1 \\ (u)_{i,j-1} - 2(u)_{i,j} + (u)_{i,j+1} & \text{if } 1 \le i \le M, 1 < j < N \\ (u)_{i,j-1} - 2(u)_{i,j} + (u)_{i,1} & \text{if } 1 \le i \le M, j = N \end{cases} \tag{32}$$

$$\partial_y^+\partial_y^-(u)_{i,j} = \partial_y^-\partial_y^+(u)_{i,j} = \begin{cases} (u)_{M,j} - 2(u)_{i,j} + (u)_{i+1,j} & \text{if } i = 1, 1 \le j \le N \\ (u)_{i-1,j} - 2(u)_{i,j} + (u)_{i+1,j} & \text{if } 1 < i < M, 1 \le j \le N \\ (u)_{i-1,j} - 2(u)_{i,j} + (u)_{1,j} & \text{if } i = M, 1 \le j \le N \end{cases} \tag{33}$$

$$\partial_x^+\partial_y^-(u)_{i,j} = \begin{cases} (u)_{i,j+1} - (u)_{i,j} - (u)_{M,j+1} + (u)_{M,j} & \text{if } i = 1, 1 \le j < N \\ (u)_{i,1} - (u)_{i,j} - (u)_{M,1} + (u)_{M,j} & \text{if } i = 1, j = N \\ (u)_{i,j+1} - (u)_{i,j} - (u)_{i-1,j+1} + (u)_{i-1,j} & \text{if } 1 < i \le M, 1 \le j < N \\ (u)_{i,1} - (u)_{i,j} - (u)_{i-1,1} + (u)_{i-1,j} & \text{if } 1 < i \le M, j = N \end{cases} \tag{34}$$

and

$$\partial_y^+\partial_x^-(u)_{i,j} = \begin{cases} (u)_{i+1,j} - (u)_{i,j} - (u)_{i+1,N} + (u)_{i,N} & \text{if } 1 \le i < M, j = 1 \\ (u)_{1,j} - (u)_{i,j} - (u)_{1,N} + (u)_{i,N} & \text{if } i = M, j = 1 \\ (u)_{i+1,j} - (u)_{i,j} - (u)_{i+1,j-1} + (u)_{i,j-1} & \text{if } 1 \le i < M, 1 < j \le N \\ (u)_{1,j} - (u)_{i,j} - (u)_{1,j-1} + (u)_{i,j-1} & \text{if } i = M, 1 < j \le N \end{cases}. \tag{35}$$

Based on (32) and (33), the discrete Laplace operator is given by

$$(\triangle u)_{i,j} = \partial_x^-\partial_x^+(u)_{i,j} + \partial_y^-\partial_y^+(u)_{i,j}. \tag{36}$$

3.2. Numerical algorithm based on split Bregman method

Alternating direction method of multipliers (ADMM) has been successfully applied in the field of image denoising for solving variational models, which require the minimization of nonlinear and non-differentiable functionals. For instance, the authors in [38] developed the split Bregman (SB) iteration method to solve the TV model, while the augmented Lagrangian method (ALM) has been developed for the TV model in [39, 40]. Note that both methods employ ADMM to minimize their energy functionals and the convergence of ADMM is always guaranteed. In [39, 40], the authors have proven that the two methods are in fact equivalent. For applying ADMM to minimize the energy functionals, we refer to [41, 42, 43, 44, 45].

In this subsection, our aim is to show how SB method can be used to solve our proposed variational model (17). Before starting our discussion on SB method, we note that our proposed model in (17) can be expressed as

$$\min_u \left\{ \mathcal{D}_{\mathrm{AN}}(u, z) + \frac{\gamma}{2}\int_\Omega \kappa_{\mathrm{M}}^2(u)d\Omega - \gamma\int_\Omega \kappa_{\mathrm{G}}(u)d\Omega \right\}. \tag{37}$$

Introducing the auxiliary variables $\boldsymbol{p} = (p_1, p_2)$, $\boldsymbol{m} = (m_1, m_2)$, $\boldsymbol{n} = (n_1, n_2)$, and q leads to the following constrained minimization problem:

$$\min_{u,q,\boldsymbol{m},\boldsymbol{n},\boldsymbol{p}} \left\{ \mathcal{D}_{\mathrm{AN}}(u, z) + \frac{\gamma}{2}\int_\Omega q^2 d\Omega + \gamma\int_\Omega \frac{p_{1_y}p_{2_x} - p_{1_x}p_{2_y}}{(1 + p_1^2 + p_2^2)^2} d\Omega \right\}, \tag{38}$$

$$\text{s.t. } \boldsymbol{p} = \nabla u,\ \boldsymbol{m} = \frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2 + 1}},\ \boldsymbol{m} = \boldsymbol{n},\ q = \nabla\cdot\boldsymbol{n}. \tag{39}$$

Therefore the solution of the constrained minimization problem (39) can be obtained by minimizing the following functional

$$\begin{aligned}\mathcal{E}(u,q,\boldsymbol{m},\boldsymbol{n},\boldsymbol{p};\boldsymbol{b}_1,\boldsymbol{b}_2,\boldsymbol{b}_3,b_4) = \mathcal{D}_{\mathrm{AN}}(u,z) + \frac{\gamma}{2}\int_\Omega q^2 d\Omega + \gamma\int_\Omega \frac{p_{1_y}p_{2_x} - p_{1_x}p_{2_y}}{(1+p_1^2+p_2^2)^2}d\Omega \\ + \frac{\theta_1}{2}\int_\Omega (\boldsymbol{p} - \nabla u - \boldsymbol{b_1})^2 d\Omega + \frac{\theta_2}{2}\int_\Omega (\boldsymbol{m} - \frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2+1}} - \boldsymbol{b_2})^2 d\Omega \\ + \frac{\theta_3}{2}\int_\Omega (\boldsymbol{n} - \boldsymbol{m} - \boldsymbol{b_3})^2 d\Omega + \frac{\theta_4}{2}\int_\Omega (q - \nabla\cdot\boldsymbol{n} - b_4)^2 d\Omega, \end{aligned} \tag{40}$$

where $\boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}, b_4$ are the Bregman parameters and $\theta_1, \theta_2, \theta_3, \theta_4$ are the positive penalty parameters.

Since all variables u, q, $\boldsymbol{m}$, $\boldsymbol{n}$, $\boldsymbol{p}$ are very difficult to solve simultaneously, we then separate the minimization problem of $\mathcal{E}$ into five subproblems and propose an alternating minimization procedure as shown in Algorithm 1 to approximate the solution. The process is repeated until one of the following stopping rules is satisfied:

$$\frac{\left\|u^{(m)} - u^{(m-1)}\right\|^2}{\left\|u^{(m-1)}\right\|^2} < \epsilon_1^{\mathrm{SB}}, \tag{41}$$

$$m \geq \epsilon_2^{\mathrm{SB}}, \tag{42}$$

where ϵ_1^{SB} denotes the predefined small positive number and ϵ_2^{SB} denotes the maximum iteration of the SB method. Here m represents the index of the current iteration.

Algorithm 1 SB iteration for the proposed model

1) Initialization: set $m = 1$, choose $\epsilon_1^{\mathrm{SB}}, \epsilon_2^{\mathrm{SB}}, \theta_1, \theta_2, \theta_3, \theta_4 > 0$ and $\boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}, b_4$.

2) **Repeat**

2.1) Compute $(u^{(m)}, p^{(m)}, \boldsymbol{m}^{(m)}, \boldsymbol{n}^{(m)}, \boldsymbol{p}^{(m)})$ as an *approximate* minimizer of the functional $\mathcal{E}$ in (40) with the Bregman parameter $\boldsymbol{b}_1^{(m-1)}$, $\boldsymbol{b}_2^{(m-1)}$, , $\boldsymbol{b}_3^{(m-1)}$, $b_4^{(m-1)}$, i.e.,

$$\begin{aligned}(u^{(m)}, p^{(m)}, &\boldsymbol{m}^{(m)}, \boldsymbol{n}^{(m)}, \boldsymbol{p}^{(m)}) \approx \\ &\underset{u,q,\boldsymbol{m},\boldsymbol{n},\boldsymbol{p}}{\arg\min}\ \mathcal{E}(u,q,\boldsymbol{m},\boldsymbol{n},\boldsymbol{p};\boldsymbol{b}_1^{(m-1)},\boldsymbol{b}_2^{(m-1)},\boldsymbol{b}_3^{(m-1)},b_4^{(m-1)}).\end{aligned} \tag{43}$$

2.2) Update Bregman parameters

$$\begin{cases} \boldsymbol{b}_1^{(m)} = \boldsymbol{b}_1^{(m-1)} + \nabla u^{(m)} - \boldsymbol{p}^{(m)}, \\ \boldsymbol{b}_2^{(m)} = \boldsymbol{b}_2^{(m-1)} + \frac{\boldsymbol{p}^{(m)}}{\sqrt{|\boldsymbol{p}^{(m)}|^2+1}} - \boldsymbol{m}^{(m)}, \\ \boldsymbol{b}_3^{(m)} = \boldsymbol{b}_3^{(m-1)} + \boldsymbol{m}^{(m)} - \boldsymbol{n}^{(m)}, \\ b_4^{(m)} = b_4^{(m-1)} + \nabla\cdot\boldsymbol{n}^{(m)} - q^{(m)}. \end{cases} \tag{44}$$

until a stopping rule for SB iteration is satisfied.

The five subproblems associated with the minimization of $\mathcal{E}$ in (40) are as follows:

u-subproblem. We fix the variables q, $\boldsymbol{m}$, $\boldsymbol{n}$, $\boldsymbol{p}$ and derive the EL equation respect to u:

$$u - \theta_1 \triangle u = G, \tag{45}$$

where $G = z + \theta_1 \nabla \cdot (\boldsymbol{b_1} - \boldsymbol{p})$. To solve the linear partial differential equation in (45), we apply the discrete divergence operator in (31) and the discrete Laplace operator in (36) as follows

$$(u)_{i,j} - \theta_1 \left(\partial_x^- \partial_x^+ (u)_{i,j} + \partial_y^- \partial_y^+ (u)_{i,j}\right) = (G)_{i,j}, \tag{46}$$

where $(G)_{i,j} = (z)_{i,j} - \theta_1 \left(\partial_x^- ((b_{11})_{i,j} - (p_1)_{i,j}) + \partial_y^- ((b_{12})_{i,j} - (p_2)_{i,j})\right)$. As periodic boundary conditions have been imposed on the discrete derivatives in Section 3.1, the discrete Fourier transform can be directly applied to the both sides of (46)

$$\mathcal{F}\left\{(u)_{i,j} - \theta_1 \left(\partial_x^- \partial_x^+ (u)_{i,j} + \partial_y^- \partial_y^+ (u)_{i,j}\right)\right\} = \mathcal{F}\left\{(G)_{i,j}\right\}, \tag{47}$$

where $\mathcal{F}$ denotes the discrete Fourier transform. For $r \in [0, M)$ and $s \in [0, N)$ in the discrete frequency domain, we have

$$\underbrace{\left(1 - 2\theta_1 \left(\cos\frac{2\pi s}{N} + \cos\frac{2\pi r}{N} - 2\right)\right)}_{\zeta} \mathcal{F}\left\{(u)_{i,j}\right\} = \mathcal{F}\left\{(G)_{i,j}\right\}, \tag{48}$$

where $i \in [1, M]$ and $j \in [1, N]$ are the indexes in discrete time domain. Thus a closed-form solution of u is given by

$$(u)_{i,j} = \text{Real}\left(\mathcal{F}^{-1}\left\{\frac{\mathcal{F}\left\{(G)_{i,j}\right\}}{\zeta}\right\}\right), \tag{49}$$

where $\mathcal{F}^{-1}$ denotes the discrete inverse Fourier transform. We note that 'Real' is the real part of a complex number and '—' stands for point-wise division of matrices.

q-subproblem. Fixing the variables u, $\boldsymbol{m}$, $\boldsymbol{n}$, $\boldsymbol{p}$ leads to the EL equation respect to q as given by

$$\gamma q + \theta_4 (q - \nabla \cdot \boldsymbol{n} - b_4) = 0. \tag{50}$$

As can be seen, the exact solution is given by

$$(q)_{i,j} = \frac{\theta_4((\partial_x^- (n_1)_{i,j} + \partial_y^- (n_2)_{i,j}) + (b_4)_{i,j})}{\gamma + \theta_4}. \tag{51}$$

***m**-subproblem.* Fixing the variables u, q, $\boldsymbol{n}$, $\boldsymbol{p}$ yields the EL equation respect to $\boldsymbol{m}$ as follows:

$$\theta_2(\boldsymbol{m} - \frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2 + 1}} - \boldsymbol{b_2}) - \theta_3(\boldsymbol{n} - \boldsymbol{m} - \boldsymbol{b_3}) = 0, \tag{52}$$

which can be determined using the closed-form formula

$$(\boldsymbol{m})_{i,j} = \frac{\theta_2((\frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2+1}})_{i,j} + (\boldsymbol{b_2})_{i,j}) + \theta_3((\boldsymbol{n})_{i,j} - (\boldsymbol{b_3})_{i,j})}{\theta_2 + \theta_3}. \tag{53}$$

***n**-subproblem.* By fixing the variables u, q, $\boldsymbol{m}$, $\boldsymbol{p}$, the EL equation respect to $\boldsymbol{n}$ is given by

$$\left(\theta_3 - \theta_4\partial_x^+\partial_x^-\right)(n_1)_{i,j} - \theta_4\partial_x^+\partial_y^-(n_2)_{i,j} = (h_1)_{i,j}, \tag{54}$$

$$\left(\theta_3 - \theta_4\partial_y^+\partial_y^-\right)(n_2)_{i,j} - \theta_4\partial_y^+\partial_x^-(n_1)_{i,j} = (h_2)_{i,j}, \tag{55}$$

where

$$(h_1)_{i,j} = \theta_3(m_1 + b_{31}) - \theta_4\partial_x^+(q - b_4), \tag{56}$$

$$(h_2)_{i,j} = \theta_3(m_2 + b_{32}) - \theta_4\partial_y^+(q - b_4). \tag{57}$$

Taking the DFT with (54) and (55) leads to the following system of linear equations

$$\mathbf{Ax} = \mathbf{b},$$

where

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}, \mathbf{x} = \begin{pmatrix} \mathcal{F}\{(n_1)_{i,j}\} \\ \mathcal{F}\{(n_2)_{i,j}\} \end{pmatrix}, \mathbf{b} = \begin{pmatrix} \mathcal{F}\{(h_1)_{i,j}\} \\ \mathcal{F}\{(h_2)_{i,j}\} \end{pmatrix},$$

$$a_{11} = \theta_3 - 2\theta_4\left(\cos\frac{2\pi s}{N} - 1\right),$$

$$a_{12} = -\theta_4\left(-1 - \cos\frac{2\pi r}{M} + \sqrt{-1}\sin\frac{2\pi r}{M}\right)\left(1 - \cos\frac{2\pi s}{N} + \sqrt{-1}\sin\frac{2\pi s}{N}\right),$$

$$a_{21} = -\theta_4\left(-1 - \cos\frac{2\pi s}{N} + \sqrt{-1}\sin\frac{2\pi s}{N}\right)\left(1 - \cos\frac{2\pi r}{M} + \sqrt{-1}\sin\frac{2\pi r}{M}\right),$$

$$a_{22} = \theta_3 - 2\theta_4\left(\cos\frac{2\pi r}{M} - 1\right).$$

For all discrete frequencies, the determinant of the coefficient matrix $\mathbf{A}$ is given by

$$D = \theta_3^2 - 2\theta_3\theta_4\left(\cos\frac{2\pi s}{N} + \cos\frac{2\pi r}{M} - 2\right), \tag{58}$$

which is always positive. Thus, the closed-form solution of n_1 and n_2 can be expressed as

$$\begin{cases} (n_1)_{i,j} = \text{Real}\left(\mathcal{F}^{-1}\left\{\frac{a_{22}\mathcal{F}\{(h_1)_{i,j}\} - a_{12}\mathcal{F}\{(h_2)_{i,j}\}}{D}\right\}\right), \\ (n_2)_{i,j} = \text{Real}\left(\mathcal{F}^{-1}\left\{\frac{a_{11}\mathcal{F}\{(h_2)_{i,j}\} - a_{21}\mathcal{F}\{(h_1)_{i,j}\}}{D}\right\}\right). \end{cases} \tag{59}$$

***p**-subproblem.* Fixing the variable u, q, $\boldsymbol{m}$, $\boldsymbol{n}$, the EL equation respect to $\boldsymbol{p}$ is given by (see Appendix B)

$$\begin{cases} -\gamma\nabla\cdot\boldsymbol{w}_1 + \gamma\Gamma p_1 + \theta_1(p_1 - u_x - b_{11}) + \theta_2(m_1 - \psi p_1 - b_{21})(\psi^3 p_1^2 - \psi) = 0, \\ -\gamma\nabla\cdot\boldsymbol{w}_2 + \gamma\Gamma p_2 + \theta_1(p_2 - u_y - b_{12}) + \theta_2(m_2 - \psi p_2 - b_{22})(\psi^3 p_2^2 - \psi) = 0, \end{cases} \tag{60}$$

where $\boldsymbol{w}_1 = \left(-\frac{(p_2)_y}{\beta_2^2}, \frac{(p_2)_x}{\beta_2^2}\right)$, $\boldsymbol{w}_2 = \left(\frac{(p_1)_y}{\beta_2^2}, -\frac{(p_1)_x}{\beta_2^2}\right)$, $\Gamma_1 = (p_1)_x(p_2)_y - (p_1)_y(p_2)_x$, $\Gamma_2 = p_1^2 + p_2^2 + 1$, $\Gamma = \frac{4\Gamma_1}{\Gamma_2^3}$, $\psi = \frac{1}{\sqrt{\Gamma_2}}$.

To solve the nonlinear partial differential equation in (60), we develop a semi-implicit fixed-point iteration method. We start from an initial solution $\boldsymbol{p}^{[0]}$ and compute a sequence of approximate solutions $\boldsymbol{p}^{[1]}, \boldsymbol{p}^{[2]}, \cdots, \boldsymbol{p}^{[\nu]}, \boldsymbol{p}^{[\nu+1]}, \cdots$ by solving the discrete linear system

$$\begin{cases} (p_1^{[\nu+1]})_{i,j} = \dfrac{\gamma(\nabla \cdot \boldsymbol{w}_1^{[\nu]})_{i,j} + \theta_1(\partial_x^+(u)_{i,j} + (b_{11})_{i,j}) + \theta_2\psi(m_1 - b_{21})_{i,j}}{\gamma(\Gamma^{[\nu]})_{i,j} + \theta_1 + \theta_2(\psi^{[\nu]})_{i,j}((m_1 - \psi^{[\nu]}p_1^{[\nu]} - b_{21})_{i,j}(\psi^{[\nu]})_{i,j}(p_1^{[\nu]})_{i,j} + 1)}, \\ (p_2^{[\nu+1]})_{i,j} = \dfrac{\gamma(\nabla \cdot \boldsymbol{w}_2^{[\nu]})_{i,j} + \theta_1(\partial_y^+(u)_{i,j} + (b_{12})_{i,j}) + \theta_2\psi(m_2 - b_{22})_{i,j}}{\gamma(\Gamma^{[\nu]})_{i,j} + \theta_1 + \theta_2(\psi^{[\nu]})_{i,j}((m_2 - \psi^{[\nu]}p_2^{[\nu]} - b_{22})_{i,j}(\psi^{[\nu]})_{i,j}(p_2^{[\nu]})_{i,j} + 1)}, \end{cases} \tag{61}$$

until the stopping rule $\max\{\frac{\|p_1^{[\nu]} - p_1^{[\nu-1]}\|}{\|p_1^{[\nu]}\|}, \frac{\|p_2^{[\nu]} - p_2^{[\nu-1]}\|}{\|p_2^{[\nu]}\|}\} < \epsilon_1^{\mathrm{FP}}$ or $\nu \geq \epsilon_2^{\mathrm{FP}}$ is met for a given small threshold $\epsilon_1^{\mathrm{FP}} > 0$ and the maximum iteration of the FP method ϵ_2^{FP}.

Now we are at a position to formally present the SB method for our proposed model as given in Algorithm 2.

Algorithm 2 SB iteration for the proposed model – solve the subproblems of (43)

1) Initialization: set $m = 1$, choose $\epsilon_1^{\mathrm{SB}}, \epsilon_2^{\mathrm{SB}}, \epsilon_1^{\mathrm{FP}}, \epsilon_2^{\mathrm{FP}}, \theta_1, \theta_2, \theta_3, \theta_4 > 0$ and $\boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}, b_4$.

2) **Repeat**

   2.1) Compute $u^{(m)}$ by (49) with $q^{(m-1)}$, $\boldsymbol{m}^{(m-1)}$, $\boldsymbol{n}^{(m-1)}$ and $\boldsymbol{p}^{(m-1)}$.

   2.2) Compute $q^{(m)}$ by (51) with $u^{(m)}$, $\boldsymbol{m}^{(m-1)}$, $\boldsymbol{n}^{(m-1)}$ and $\boldsymbol{p}^{(m-1)}$.

   2.3) Compute $\boldsymbol{m}^{(m)}$ by (53) with $u^{(m)}$, $q^{(m)}$, $\boldsymbol{n}^{(m-1)}$ and $\boldsymbol{p}^{(m-1)}$.

   2.4) Compute $\boldsymbol{n}^{(m)}$ by (59) with $u^{(m)}$, $q^{(m)}$, $\boldsymbol{m}^{(m)}$ and $\boldsymbol{p}^{(m-1)}$.

   2.5) Compute $\boldsymbol{p}^{(m)}$ by (61) with $u^{(m)}$, $q^{(m)}$, $\boldsymbol{m}^{(m)}$ and $\boldsymbol{n}^{(m)}$.

   2.6) Update all Bregman parameters by (44) with $u^{(m)}$, $q^{(m)}$, $\boldsymbol{m}^{(m)}$, $\boldsymbol{n}^{(m)}$ and $\boldsymbol{p}^{(m)}$.

   until a stopping rule for SB iteration is satisfied.

4. Experimental results

In this section, we present a number of numerical experiments to show the performance of our proposed model (the NewCv model) in comparison with the existing variational models mentioned in Section 1.2, which are the TV, TL, BH, CEP-L^2, TVL, INFCON, TVBH, TGV, MC and GC models.

We use the SB algorithms of Lu et al. [34] to solve the TV, TL, BH, CEP-L^2, TVL, INFCON, TVBH, TGV, and MC models, while we apply the SB algorithm developed in our previous work [46] to solve the GC model. All eleven SB algorithms are started with $u^{(0)} = z$ and terminated with the same stopping rules as given in (41) and (42), where $\epsilon_1^{\mathrm{SB}} = 5.5 \times 10^{-5}$

and $\epsilon_2^{\text{SB}} = 1000$. For the proposed SB algorithm in Algorithm 2, we apply $\epsilon_1^{\text{FP}} = 10^{-2}$ and $\epsilon_2^{\text{FP}} = 10$ to stop the iterative procedure in solving $\boldsymbol{p}$-subproblem. We note that all eleven SB algorithms consist of regularization and penalty parameters. In each model, we carry out many experiments with different values of these parameters and we select manually the ones with the best denoising result.

In total, we apply twelve images to demonstrate the denoising performance of eleven different models. The test images are 256×256 as shown in Figure 1: i) three synthetic images: (a) "Piecewise constant", (b) "Piecewise linear", and (c) "Piecewise smooth"; ii) six real natural images: (d) "Barbara", (e) "Boat", (f) "Cameraman", (g) "Castle", (h) "Lena" and (i) "Peppers"; and iii) three real noisy images: (j) "Abdomen", (k) "Brain", (l) "Knee".

For measuring the denoised image quality, Peak signal-to-noise ratio (PSNR) in decibels (dB):

$$\text{PNSR} = 10 \log_{10} \left(\frac{255^2}{\frac{1}{256^2} \|\hat{u} - u\|^2} \right), \tag{62}$$

and structure similarity index map (SSIM):

$$\text{SSIM} = \frac{(2\mu_1\mu_2 + C_1)(2\sigma_{12} + C_2)}{(\mu_1^2 + \mu_2^2 + C_1)(\sigma_1^2 + \sigma_2^2 + C_2)}, \tag{63}$$

are used to perform effectiveness evaluation for all variational models, where μ_1 and μ_2 are averages of $\hat{u}$ and u, respectively, σ_1 and σ_2 are variance of $\hat{u}$ and u, respectively, and σ_{12} is the covariance of $\hat{u}$ and u. The positive constants C_1 and C_2 can be thought of as stabilizing constants for near-zero denominator values. Here $\hat{u}$ and u denote the denoised image and the original clean image, respectively. Recall that the higher the PSNR, the better denoised image has been recovered to match the original clean image. For SSIM in (63), it is a well-known quality metric used to asses the the conservation of the structural information of the denoised image. Note that a perfect restoration has SSIM value equal to 1. We refer [47] for more details in image quality assessment.

Note that in all experiments, each denoising algorithm is performed using MATLAB R2018a and run on a machine configured with Intel(R) Core(TM) i7 Quad-core 4.2GHz and 32GB of RAM.

4.1. Comparison 1: edge-preservation ability test

In Figure 2, we illustrate the edge-preservation ability of eleven different models on a noisy image, which is piecewise constant, in the sub-figure (a) through visual inspection. From the sub-figures (b) to (l), it is clear that our NewCv model preserves the edges of the strips perfectly. Its denoised results look almost the same as the original clean one in Figure

1 (a). TV, TGV, MC and GC models also give very good edge-preservation results. Similar visual effect can be observed on results by INFCON and TVBH models. The denoised images in the sub-figures (c)-(f) show that applying TL, BH, CEP-L^2 and TVL can blur the edges of the objects, where TL model is the worst method for edge preservation. By incorporating the TV regularization into BH and TL regularizations, TVL and TVBH model can improve the denoised quality as can see from the sub-figures (f) and (h). However, the edges of the objects are still smeared by these two models.

In Figure 3, we present the associated residual images $z - u$ for all variational models, where z is the noisy image in Figure 2 (a) and u represents each corresponding denoised image in Figure 2. The residual image in the sub-figure (l) contains almost all noise, whereas the other ten residual images contain more or less structure information of the original image in Figure 1 (a). This means that some contrast information has been lost in the denoised images handled by other ten models. Figure 3 illustrates that our NewCv model is the best in preserving image contrast.

In Figure 4, the plots of the middle slices of the denoised images in Figure 2 are presented. We can see that the denoised slice curve (black) in the sub-figure (l) almost overlaps with the original noise free slice curve (red), which demonstrates the preservation of images contrast as well as corners of NewCv model. Compared with TGV and NewCv models, the other nine models lead to less pleasant fitting results.

4.2. Comparison 2: smoothness-preservation ability test

In this subsection, we test the smoothness-preservation ability of eleven different models on the "Piecewise linear" image in Figure 1 (b) and the "Piecewise smooth" in Figures 1 (c). Both test images are corrupted with Gaussian noise with the noise variance 0.07.

Figure 5 and 7 show the denoised images on the test images by eleven different models. One can easily see in the sub-figure (l) that our NewCv model produces the denoised results with the highest values of PSNR and SSIM. Moreover, the plots of the middle slices of the denoised images as shown in Figures 6 and 8 show that the denoised signal in the sub-figure (b) by TV model is composed of jagged appearance which is staircase effect. In addition, the second-order models involving TV regularization, CEP-L^2, TVL, INFCON and TVBH models, deliver staircase artifact in the denoised signals as represented by the sub-figures (e)-(h), whereas the pure second-order models, TL, BH, TGV and NewCv models, do not produce unfavorable staircase effect in the denoised signals as shown in the sub-figures (c), (d), (i) and (l). The staircase artifact can be observed from the sub-figure (j), whereas we see the oscillation in the smooth region of the denoised signal by GC model. According to Figures

6 and 8, the visual effects show our NewCv model gives the best smoothness-preservation result.

4.3. Comparison 3: signal-dependent test

In this subsection we compare the denoising performance of eleven different models in removing noise with different noise variances. The six real natural images in Figures 1 (d)-(i) are corrupted with five variance values from 0.01, 0.03, 0.05, 0.07 and 0.1.

Tables 1 - 5 provide the comparative results, which contain the two evaluation metrics for quantitative comparison of eleven different models on the six real natural images with the five different noise variances. In order to easily assess these metrics in Tables 1 - 5, we draw the plots in Figure 9, where each plot in the sub-figures (a) and (b) corresponds to the average values of the denoised results produced by its corresponding model on different noise levels. The results in Figure 9 demonstrate that our NewCv model is the most effective

Table 1: Comparison of eleven different models in restoration quality on the six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with the noise variance 0.01.

	PSNR							SSIM						
	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average
TV	32.71	32.89	33.26	34.89	35.53	36.04	34.22	0.8997	0.9041	0.9083	0.9437	0.9377	0.9534	0.9245
TL	31.39	31.42	31.22	33.22	34.41	34.28	32.66	0.8781	0.8862	0.8905	0.9241	0.9252	0.9446	0.9081
BH	31.68	32.05	31.72	34.15	34.84	35.21	33.27	0.8801	0.8954	0.8922	0.9360	0.9310	0.9497	0.9141
CEP-L^2	33.36	33.87	34.16	35.98	36.43	36.89	35.11	0.9103	0.9210	0.9208	0.9512	0.9457	0.9606	0.9349
TVL	33.32	33.63	33.92	35.62	36.24	36.81	34.92	0.9096	0.9161	0.9178	0.9498	0.9445	0.9584	0.9327
INFCON	34.32	35.03	35.32	37.02	37.64	37.98	36.22	0.9238	0.9373	0.9335	0.9593	0.9558	0.9655	0.9459
TVBH	34.90	36.08	35.72	37.95	38.52	38.99	37.03	0.9305	0.9497	0.9395	0.9651	0.9624	0.9698	0.952
TGV	42.86	43.59	43.37	44.77	45.04	45.00	44.11	0.9861	0.9886	0.9804	0.9889	0.9881	0.9886	0.9868
MC	39.27	40.94	40.46	42.26	42.75	43.29	41.49	0.9719	0.9821	0.9710	0.9836	0.9830	0.9848	0.9794
GC	38.84	38.03	35.95	39.00	40.70	40.11	38.77	0.9619	0.9578	0.9341	0.9573	0.9673	0.9664	0.9575
NewCv	**47.18**	**46.21**	**45.57**	**47.42**	**47.89**	**47.41**	**46.95**	**0.9949**	**0.9946**	**0.9906**	**0.9941**	**0.9938**	**0.9936**	**0.9936**

Table 2: Comparison of eleven different models in restoration quality on the six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with the noise variance 0.03.

	PSNR							SSIM						
	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average
TV	32.59	32.81	33.22	34.75	35.38	35.73	34.08	0.8972	0.9031	0.9082	0.9408	0.9352	0.9497	0.9224
TL	31.29	31.33	31.15	33.09	34.26	34.07	32.53	0.8757	0.8850	0.8896	0.9218	0.9228	0.9419	0.9061
BH	31.64	32.02	31.70	34.09	34.79	35.10	33.22	0.8796	0.8955	0.8925	0.9355	0.9308	0.9490	0.9138
CEP-L^2	33.19	33.70	34.04	35.74	36.17	36.48	34.88	0.9077	0.9196	0.9203	0.9492	0.9430	0.9576	0.9329
TVL	33.14	33.53	33.80	35.46	36.07	36.48	34.75	0.9066	0.9154	0.9167	0.9473	0.9425	0.9554	0.9306
INFCON	34.22	34.89	35.21	36.83	37.45	37.67	36.05	0.9235	0.9371	0.9339	0.9584	0.9551	0.9644	0.9454
TVBH	34.74	35.88	35.59	37.71	38.27	38.62	36.80	0.9299	0.9492	0.9399	0.9643	0.9616	0.9688	0.9523
TGV	37.84	39.06	38.76	39.84	39.95	40.19	39.04	0.9639	0.9624	0.9628	0.9743	0.9720	0.9752	0.9675
MC	36.25	37.37	37.33	38.89	39.21	39.49	38.09	0.9480	0.9619	0.9532	0.9683	0.9655	0.9703	0.9612
GC	37.21	37.64	36.70	38.71	39.19	38.83	38.16	0.9520	0.9587	0.9412	0.9551	0.9584	0.9595	0.9551
NewCv	**39.29**	**39.36**	**39.73**	**40.68**	**41.14**	**40.72**	**40.15**	**0.9722**	**0.9760**	**0.9684**	**0.9750**	**0.9745**	**0.9768**	**0.9738**

Table 3: Comparison of eleven different models in restoration quality on the six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with the noise variance 0.05.

	PSNR							SSIM						
	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average
TV	32.39	32.65	33.13	34.53	35.12	35.26	33.84	0.8941	0.9017	0.9084	0.9371	0.9323	0.9451	0.9198
TL	31.39	31.61	31.47	33.20	34.16	33.28	32.52	0.8785	0.8937	0.8937	0.9186	0.9194	0.9321	0.9060
BH	31.58	31.95	31.64	33.98	34.68	34.91	33.12	0.8790	0.8955	0.8929	0.9347	0.9300	0.9478	0.9133
CEP-L^2	32.91	33.39	33.79	35.32	35.72	35.87	34.50	0.9043	0.9173	0.9189	0.9453	0.9386	0.9532	0.9296
TVL	32.91	33.27	33.67	35.13	35.72	35.90	34.43	0.9036	0.9129	0.9167	0.9431	0.9390	0.9506	0.9276
INFCON	34.00	34.58	34.93	35.89	36.48	36.65	35.42	0.9226	0.9358	0.9334	0.9527	0.9487	0.9595	0.9421
TVBH	34.40	35.41	35.22	36.22	36.80	37.07	35.85	0.9274	0.9464	0.9385	0.9538	0.9503	0.9604	0.9461
TGV	35.07	35.09	35.03	36.27	37.22	37.19	35.98	0.9381	0.9412	0.9340	0.9534	0.9538	0.9598	0.9467
MC	34.53	34.90	35.12	35.16	36.12	36.28	35.35	0.9297	0.9378	0.9352	0.9406	0.9401	0.9492	0.9388
GC	34.67	34.72	34.61	35.88	35.93	35.66	35.25	0.9246	0.9300	0.9236	0.9280	0.9211	0.9296	0.9262
NewCv	**35.16**	**35.75**	**35.87**	**37.01**	**37.73**	**37.22**	**36.46**	**0.9351**	**0.9448**	**0.9412**	**0.9550**	**0.9552**	**0.9565**	**0.9480**

Table 4: Comparison of eleven different models in restoration quality on the six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with the noise variance 0.07.

	PSNR							SSIM						
	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average
TV	32.12	32.40	32.94	34.18	34.75	34.66	33.51	0.8907	0.8995	0.9082	0.9319	0.9279	0.9391	0.9162
TL	31.12	31.18	31.01	32.87	34.00	33.73	32.32	0.8724	0.8834	0.8882	0.9182	0.9191	0.9381	0.9032
BH	31.47	31.82	31.53	33.77	34.48	34.51	32.76	0.8779	0.8947	0.8924	0.9325	0.9279	0.9354	0.9101
CEP-L^2	32.63	33.01	33.58	34.24	34.64	34.61	33.79	0.9007	0.9118	0.9175	0.9327	0.9273	0.9426	0.9221
TVL	32.41	32.75	33.17	34.35	34.65	34.71	33.67	0.8960	0.9062	0.9119	0.9353	0.9278	0.9423	0.9199
INFCON	32.72	33.09	33.65	34.28	34.54	34.42	33.78	0.9032	0.9134	0.9190	0.9359	0.9268	0.9390	0.9229
TVBH	32.95	33.44	33.80	34.43	34.60	34.71	33.99	0.9056	0.9187	0.9209	0.9373	0.9274	0.9423	0.9254
TGV	32.91	33.94	34.35	35.48	34.90	34.91	34.41	0.9073	0.9280	0.9235	0.9476	0.9314	0.9450	0.9288
MC	32.79	33.41	33.68	34.90	34.62	34.73	34.02	0.9027	0.9176	0.9084	0.9269	0.9197	0.9320	0.9179
GC	32.12	31.90	30.70	33.04	34.67	33.33	32.63	0.8952	0.9087	0.9003	0.9085	0.9047	0.9245	0.9070
NewCv	**33.71**	**34.30**	**34.71**	**35.68**	**35.68**	**35.67**	**34.95**	**0.9157**	**0.9295**	**0.9266**	**0.9413**	**0.9353**	**0.9459**	**0.9324**

Table 5: Comparison of eleven different models in restoration quality on the six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with the noise variance 0.1.

	PSNR							SSIM						
	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average
TV	29.46	29.20	30.56	30.96	32.54	31.93	30.77	0.8279	0.8202	0.8667	0.8884	0.8953	0.9077	0.8677
TL	29.28	28.91	28.62	30.35	31.73	31.03	29.99	0.8269	0.8288	0.8442	0.8753	0.8790	0.9037	0.8596
BH	29.45	28.96	28.76	30.84	31.91	31.66	30.26	0.8285	0.8234	0.8420	0.8894	0.8853	0.9139	0.8638
CEP-L^2	29.73	29.67	31.27	31.45	33.04	31.93	31.18	0.8323	0.8352	0.8810	0.8956	0.9028	0.9123	0.8765
TVL	29.79	29.54	30.83	31.31	33.01	32.09	31.09	0.8364	0.8298	0.8725	0.8950	0.9034	0.9116	0.8748
INFCON	29.88	29.79	31.30	31.64	33.05	32.59	31.38	0.8381	0.8395	0.8826	0.9008	0.9046	0.9246	0.8817
TVBH	29.86	29.56	31.74	31.45	33.11	32.74	31.41	0.8366	0.8306	0.8908	0.8990	0.9059	0.9215	0.8807
TGV	29.89	29.73	31.88	31.57	33.47	33.00	31.59	0.8397	0.8397	0.8931	0.8987	0.9119	0.9288	0.8853
MC	29.69	29.49	31.61	30.85	33.09	32.74	31.24	0.8344	0.8414	0.8784	0.8834	0.8946	0.8978	0.8717
GC	29.28	28.31	29.30	29.41	32.03	30.02	29.72	0.8384	0.8394	0.8221	0.8525	0.8688	0.8861	0.8512
NewCv	**30.22**	**30.12**	**32.14**	**33.10**	**33.72**	**33.30**	**32.10**	**0.8554**	**0.8614**	**0.8989**	**0.9174**	**0.9116**	**0.9216**	**0.8944**

for removing noise and preserving structural information with high values for all the quality metrics.

4.4. Comparison 4: computational efficiency test

In this experiment, the six real natural images in Figures 1(d)-(i) are used further to evaluate the computational efficiency of eleven different models. The test images are corrupted with the five different noise variances from 0.01, 0.03, 0.05, 0.07 and 0.1.

Table 6 demonstrates computational efficiency of different models on the noisy versions of the test images with the five different noise levels. In order to clearly compare the data in Table 6, we draw the bar charts in Figures 10, where each bar in the sub-figures (a)-(c) corresponds to the average value of the denoised results delivering by its corresponding model on the noisy test images.

In terms of computation times per iteration, we can see that our NewCv model requires to solve five sub-problems in each step. Thus, in accordance with our measurements, we expect that more computation time per iteration is required for the proposed model than those of the other ten models. As can be seen from the sub-figure (a), our numerical experiments reflect this expectation.

In terms of total CPU times, the sub-figure (b) illustrates that the SB algorithm of INFCON model is the slowest method, while that of TV is the fastest method. Moreover, the SB algorithms of CEP-L^2, TVL, INFCON, TVBH, TGV, MC, GC and NewCv models are slower than those of TL and BH models.

In terms of total iterations, the sub-figure (c) shows that the SB algorithms of CEP-L^2 and INFCON take more than 175 iterations to converge, while the other nine algorithms take less than 100 iterations. It is important to note that our SB algorithm for the proposed model is slightly better than those of TGV, MC and GC models.

4.5. Comparison 5: real noisy image test

In the last test, in order to further show the denoising performance of our proposed model, we extend to a real world application, which is clinical image, and compare our NewCv model against the other eleven models on three real clinical images as shown in Figures 1 (j)-(l).

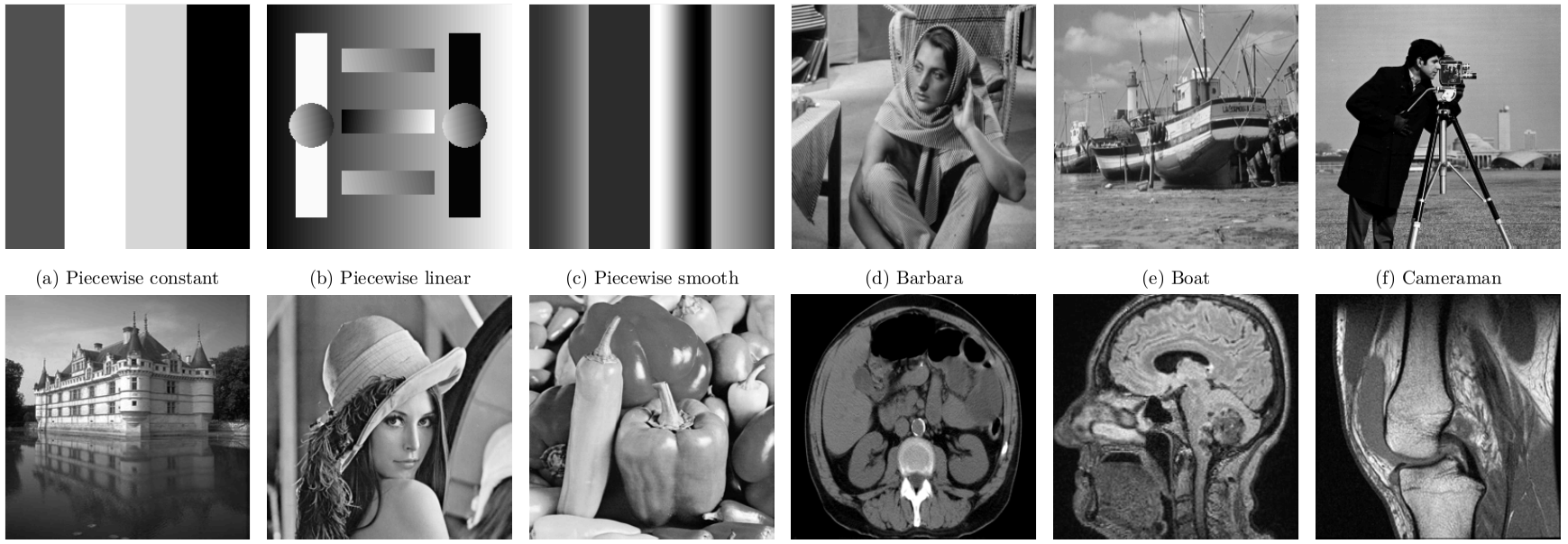
In Figures 11, 13 and 15 we present the denoised images by applying eleven different models. For better visualization, we include the corresponding plots of the middle slices of the denoised images in Figures 12, 14 and 16.

As the clean image is unknown, the denoising performance is evaluated by visual inspection. First, as expected, one can observe the staircase effect in the denoised results by TV model as shown in Figures 11 - 16. Next, the denoised images by TL and BH models in the sub-figures (c) and (d) of Figures 11, 13 and 15 seem to be good in term of visual quality, but TL and BH models tend to oversmooth certain features, which wipe out important details

Table 6: Comparison of computational efficiency by eleven different models on the six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with five different noise variances from0.01, 0.03, 0.05, 0.07 and 0.1.

Noise variance		**TV**	**TL**	**BH**	**CEP-L^2**	**TVL**	**INFCON**	**TVBH**	**TGV**	**MC**	**GC**	**NewCv**
	Time per iteration (s)	0.0057	0.0050	0.0085	0.0082	0.0070	0.0118	0.0104	0.0144	0.0119	0.0092	0.0212
0.01	Total CPU time (s)	0.2082	0.2871	0.4541	1.4637	0.5508	2.1377	0.7350	0.5277	0.3808	0.3772	0.7420
	Total iterations	37	58	54	179	78	181	71	37	32	41	35
	Time per iteration (s)	0.0057	0.0048	0.0081	0.0080	0.0068	0.0114	0.0101	0.0130	0.0111	0.0099	0.0199
0.03	Total CPU time (s)	0.1957	0.2824	0.4348	1.4001	0.5355	1.9838	0.7142	0.9540	0.7770	0.8217	0.8557
	Total iterations	35	59	54	175	79	176	71	75	70	83	43
	Time per iteration (s)	0.0059	0.0051	0.0085	0.0083	0.0071	0.0121	0.0110	0.0135	0.0115	0.0081	0.0199
0.05	Total CPU time (s)	0.1930	0.3149	0.4538	1.4179	0.5691	2.2812	0.7882	0.6365	0.4750	0.7371	1.5366
	Total iterations	33	61	54	170	80	187	71	47	41	91	78
	Time per iteration (s)	0.0057	0.0047	0.0082	0.0079	0.0067	0.0112	0.0099	0.0118	0.0107	0.0081	0.0214
0.07	Total CPU time (s)	0.1762	0.3038	0.4374	2.0557	1.0614	1.8395	1.4814	1.4568	1.0914	0.9234	2.0758
	Total iterations	31	65	54	262	159	165	151	125	102	114	97
	Time per iteration (s)	0.0055	0.0047	0.0081	0.0080	0.0069	0.0112	0.0103	0.0120	0.0108	0.0087	0.0201
0.1	Total CPU time (s)	0.2442	0.2762	0.4314	1.2337	0.5283	2.1278	0.5327	1.4183	1.1180	0.9396	2.2310
	Total iterations	45	58	53	155	77	190	51	118	110	108	111

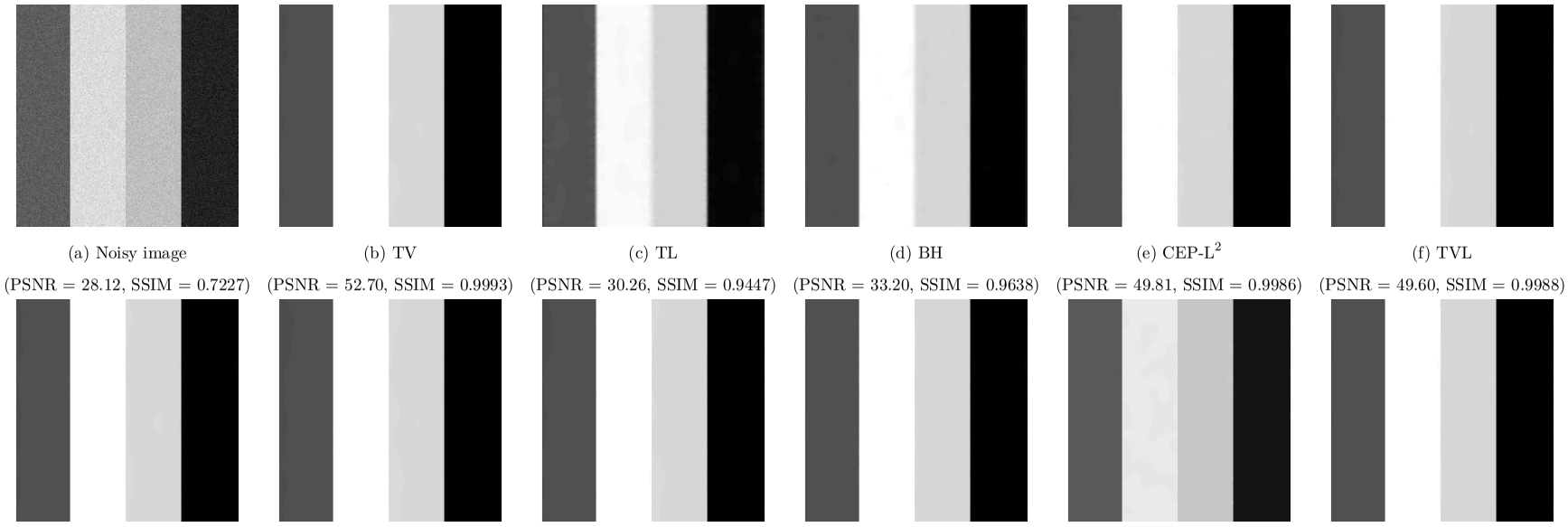
of the image structure. Moreover, the sub-figures (e)-(k) of Figures 11, 13 and 15 present a better performance compared with those shown in the sub-figures (c) and (d). Finally, the denoised results by our NewCv model shown in Figures 11 - 16, especially in the zoomed parts, are superior to the other eleven models for preserving edges and reducing the staircase effect.



(a) Piecewise constant (b) Piecewise linear (c) Piecewise smooth (d) Barbara (e) Boat (f) Cameraman

(g) Castle (h) Lena (i) Peppers (j) Abdomen (k) Brain (l) Knee

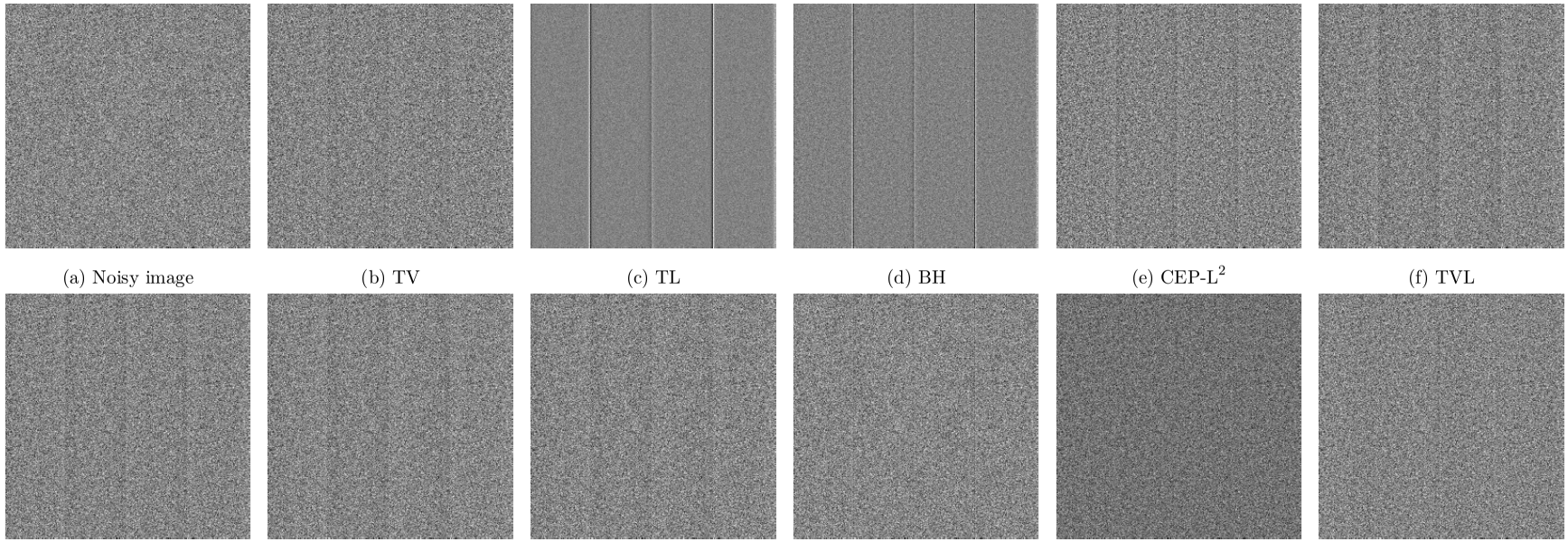
Figure 1: Original images: i) three synthetic images (a)-(c); ii) six real natural images (d)-(i); iii) three real noisy images (j)-(l).



(a) Noisy image (PSNR = 28.12, SSIM = 0.7227) (b) TV (PSNR = 52.70, SSIM = 0.9993) (c) TL (PSNR = 30.26, SSIM = 0.9447) (d) BH (PSNR = 33.20, SSIM = 0.9638) (e) CEP-L^2 (PSNR = 49.81, SSIM = 0.9986) (f) TVL (PSNR = 49.60, SSIM = 0.9988)

(g) INFCON (PSNR = 51.05, SSIM = 0.9991) (h) TVBH (PSNR = 50.85, SSIM = 0.9992) (i) TGV (PSNR = 51.88, SSIM = 0.9993) (j) MC (PSNR = 51.32, SSIM = 0.9988) (k) GC (PSNR = 51.79, SSIM = 0.9986) (l) NewCv (PSNR = 53.18, SSIM = 0.9993)

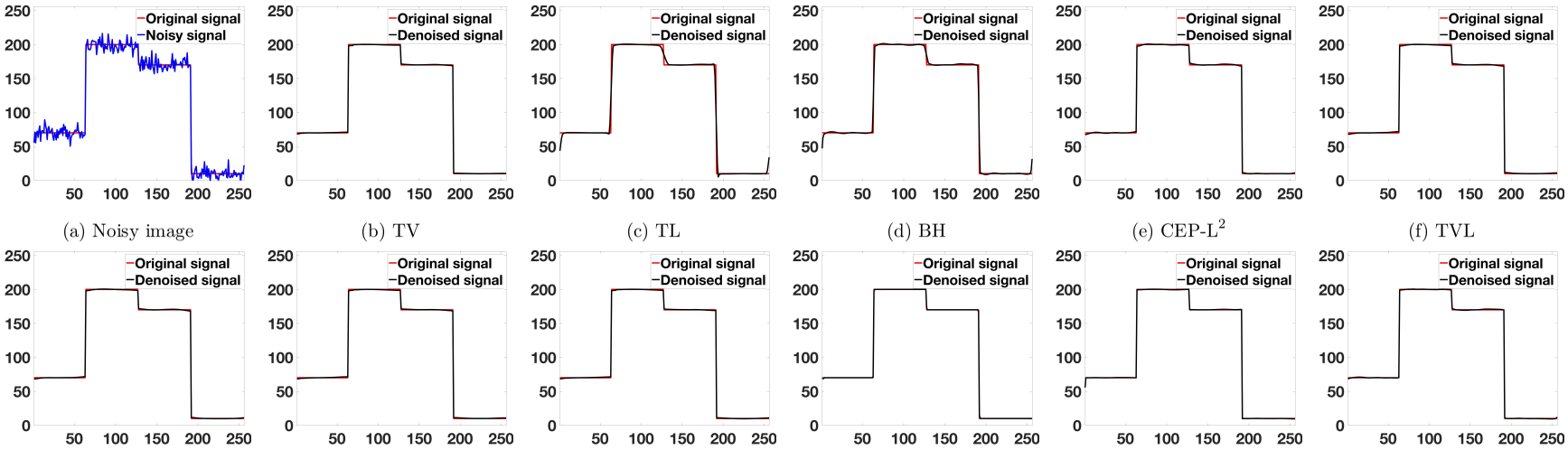
Figure 2: Edge-preservation ability test. The "Piecewise constant" image in Figure 1 (a) is distorted by the Gaussian noise with the noise variance 0.07. (a) Noisy image; (b)-(l) denoised images by eleven different models.



(a) Noisy image (b) TV (c) TL (d) BH (e) CEP-L^2 (f) TVL

(g) INFCON (h) TVBH (i) TGV (j) MC (k) GC (l) NewCv

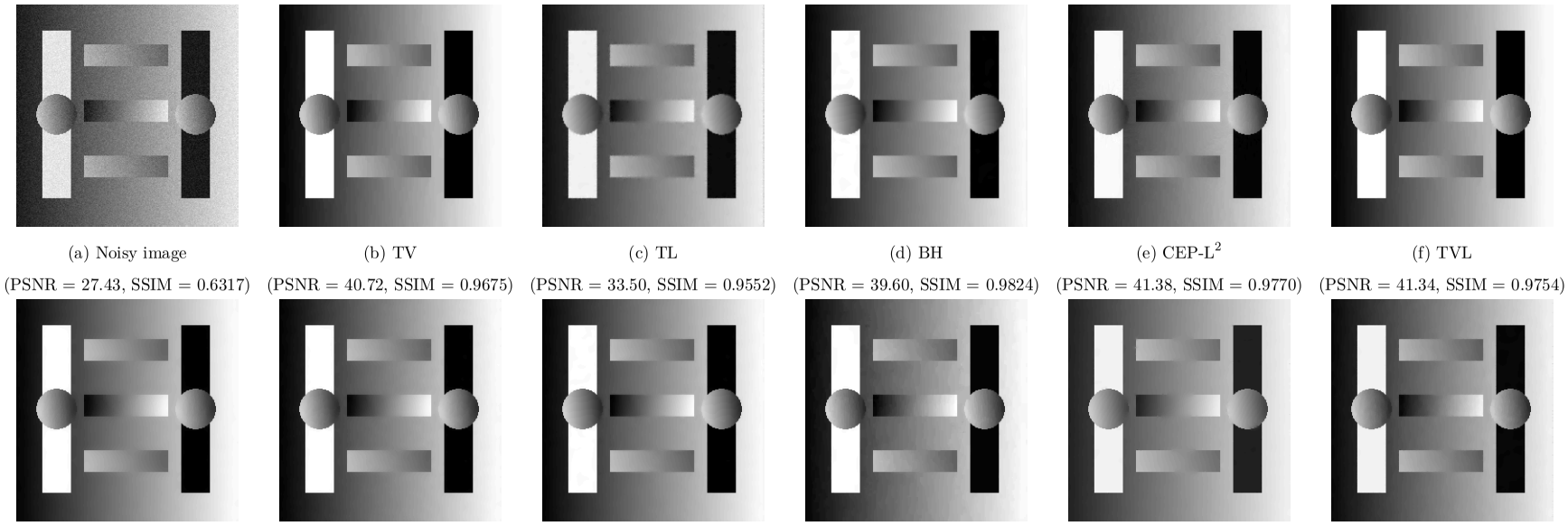
Figure 3: Comparison of residual images between the original and denoised images in Figure 2 by eleven different models.



(a) Noisy image (b) TV (c) TL (d) BH (e) CEP-L^2 (f) TVL

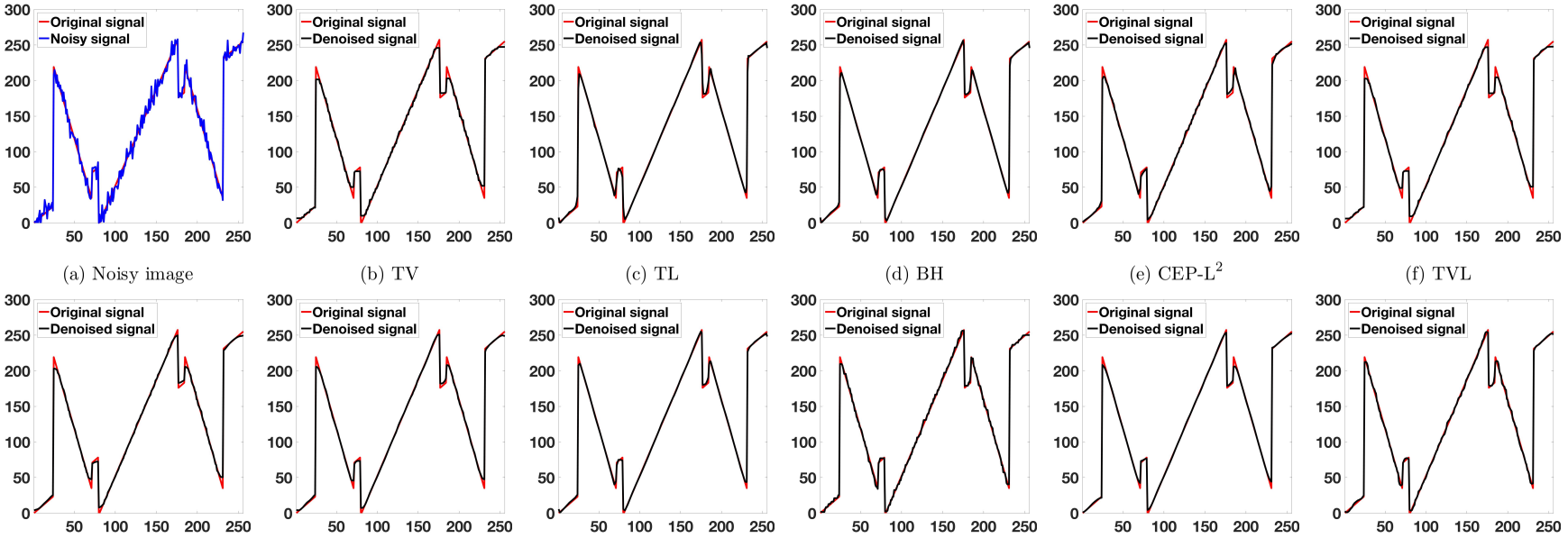
(g) INFCON (h) TVBH (i) TGV (j) MC (k) GC (l) NewCv

Figure 4: Plots of the middle slices of the denoised images in Figure 2 by eleven different models.



(a) Noisy image (PSNR = 27.43, SSIM = 0.6317)
(b) TV (PSNR = 40.72, SSIM = 0.9675)
(c) TL (PSNR = 33.50, SSIM = 0.9552)
(d) BH (PSNR = 39.60, SSIM = 0.9824)
(e) CEP-L^2 (PSNR = 41.38, SSIM = 0.9770)
(f) TVL (PSNR = 41.34, SSIM = 0.9754)
(g) INFCON (PSNR = 41.34, SSIM = 0.9802)
(h) TVBH (PSNR = 41.99, SSIM = 0.9823)
(i) TGV (PSNR = 43.40, SSIM = 0.9874)
(j) MC (PSNR = 42.29, SSIM = 0.9756)
(k) GC (PSNR = 43.04, SSIM = 0.9898)
(l) NewCv (PSNR = 43.93, SSIM = 0.9865)

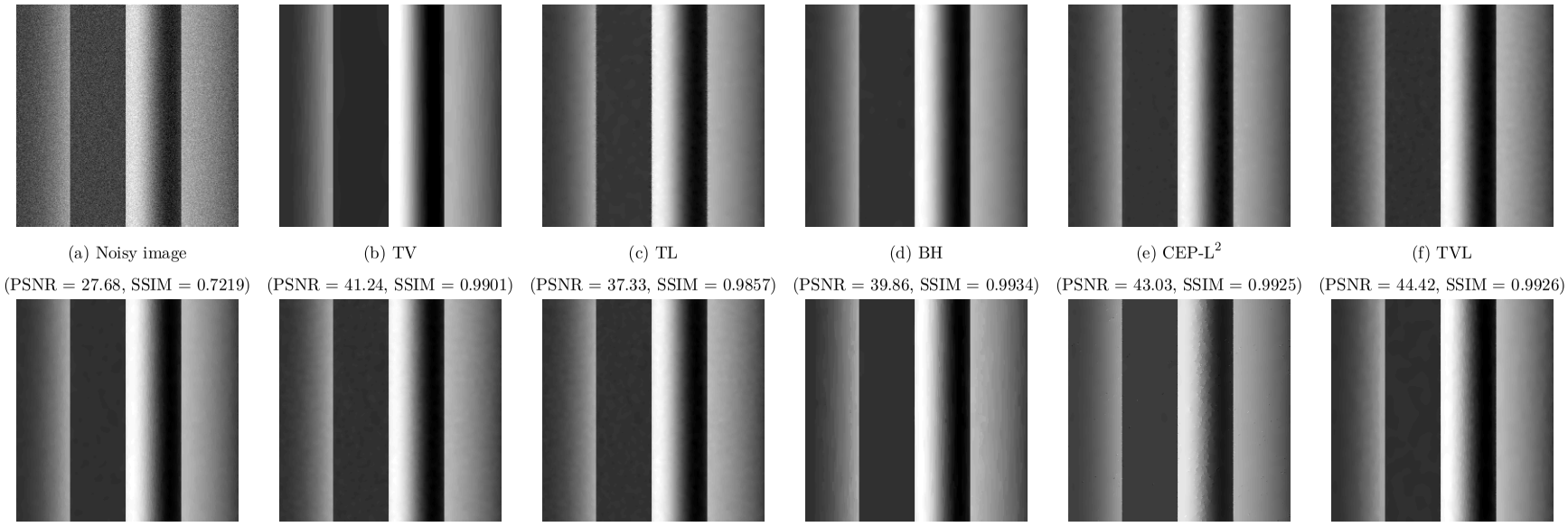
Figure 5: Smoothness-preservation ability test. The "Piecewise linear" image in Figure 1 (b) is distorted by the Gaussian noise with the noise variance 0.07. (a) noisy image; (b)-(l) denoised images by eleven different models.



(a) Noisy image (b) TV (c) TL (d) BH (e) CEP-L^2 (f) TVL

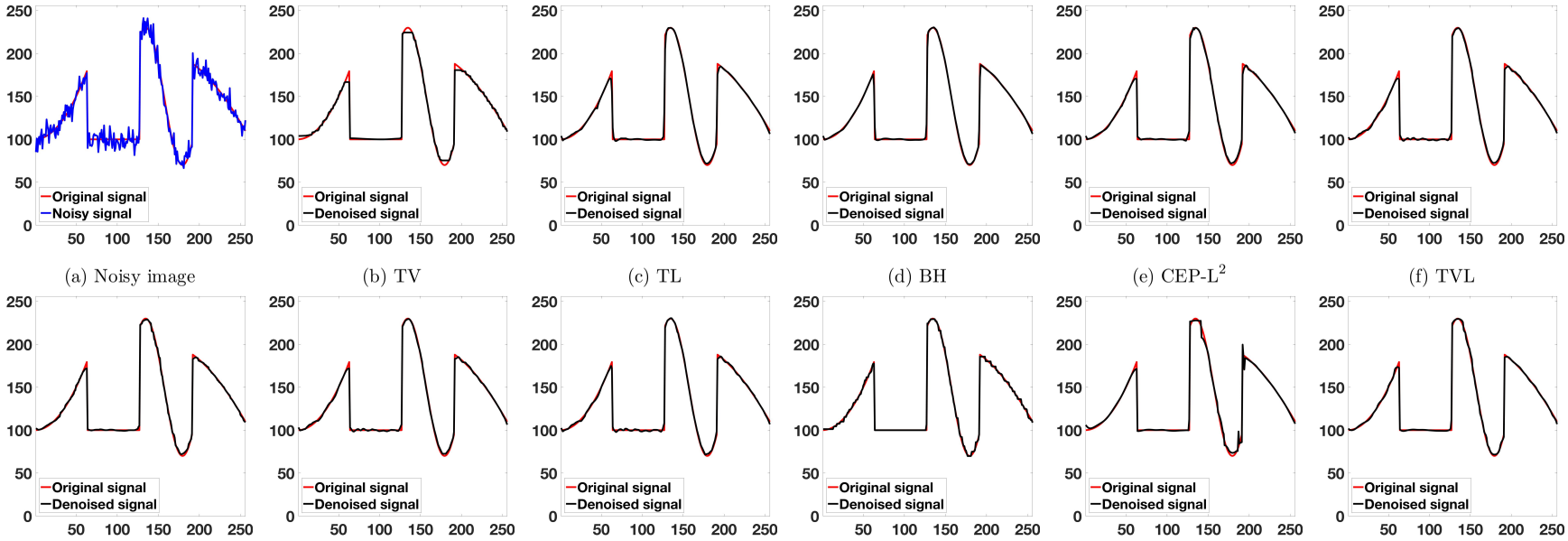
(g) INFCON (h) TVBH (i) TGV (j) MC (k) GC (l) NewCv

Figure 6: Plots of the middle slices of the denoised images in Figure 5 "Piecewise linear" by eleven different models.



(a) Noisy image (PSNR = 27.68, SSIM = 0.7219)
(b) TV (PSNR = 41.24, SSIM = 0.9901)
(c) TL (PSNR = 37.33, SSIM = 0.9857)
(d) BH (PSNR = 39.86, SSIM = 0.9934)
(e) CEP-L^2 (PSNR = 43.03, SSIM = 0.9925)
(f) TVL (PSNR = 44.42, SSIM = 0.9926)
(g) INFCON (PSNR = 45.08, SSIM = 0.9923)
(h) TVBH (PSNR = 45.09, SSIM = 0.9931)
(i) TGV (PSNR = 45.20, SSIM = 0.9922)
(j) MC (PSNR = 45.18, SSIM = 0.9866)
(k) GC (PSNR = 40.37, SSIM = 0.9808)
(l) NewCv (PSNR = 45.46, SSIM = 0.9935)

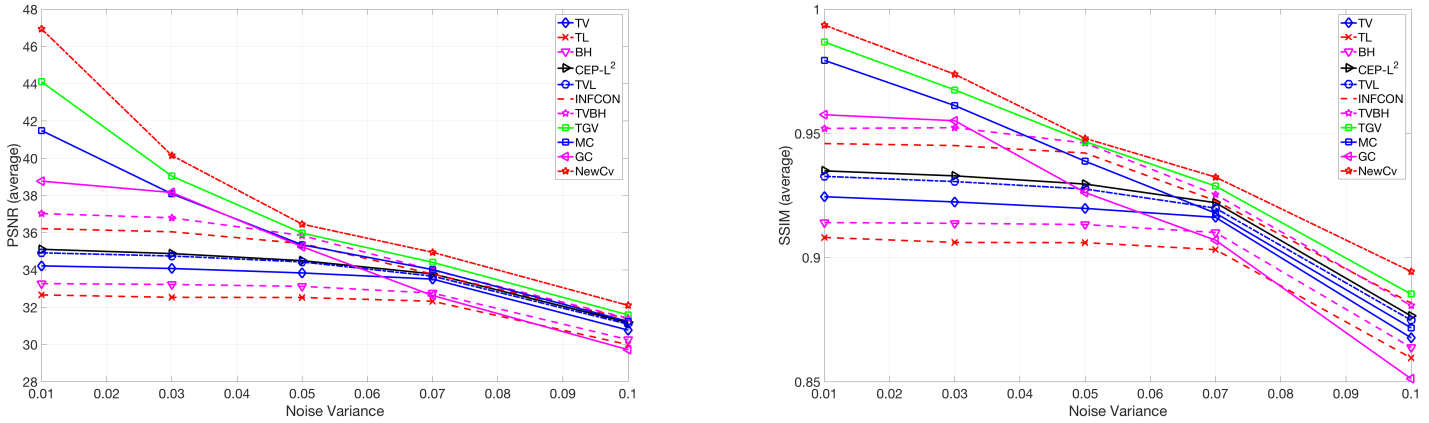
Figure 7: Smoothness-preservation ability test. The "Piecewise smooth" image in Figure 1 (c) is distorted by the Gaussian noise with the noise variance 0.07. (a) noisy image; (b)-(l) denoised image by eleven different models.



(a) Noisy image (b) TV (c) TL (d) BH (e) CEP-L^2 (f) TVL

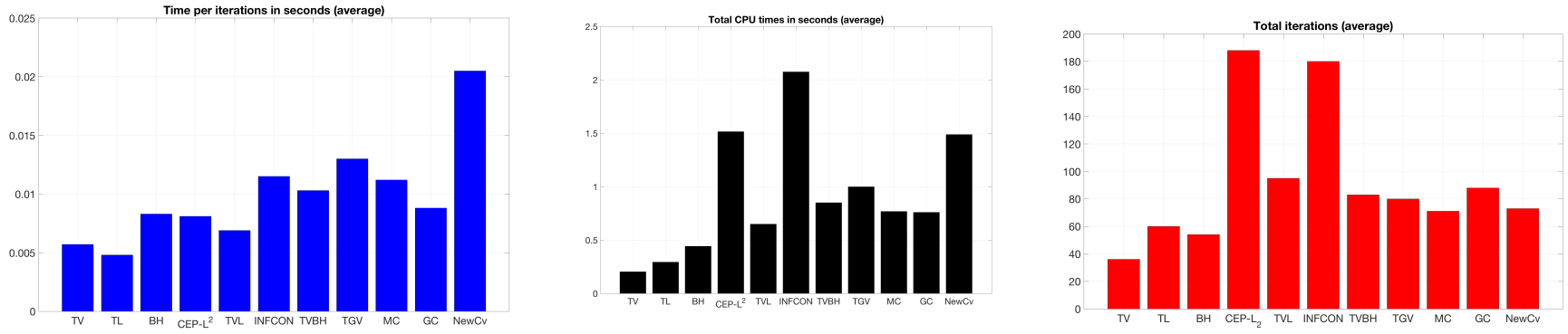
(g) INFCON (h) TVBH (i) TGV (j) MC (k) GC (l) NewCv

Figure 8: Plots of the middle slices of the denoised images in Figure 7 “Piecewise smooth” by eleven different models.



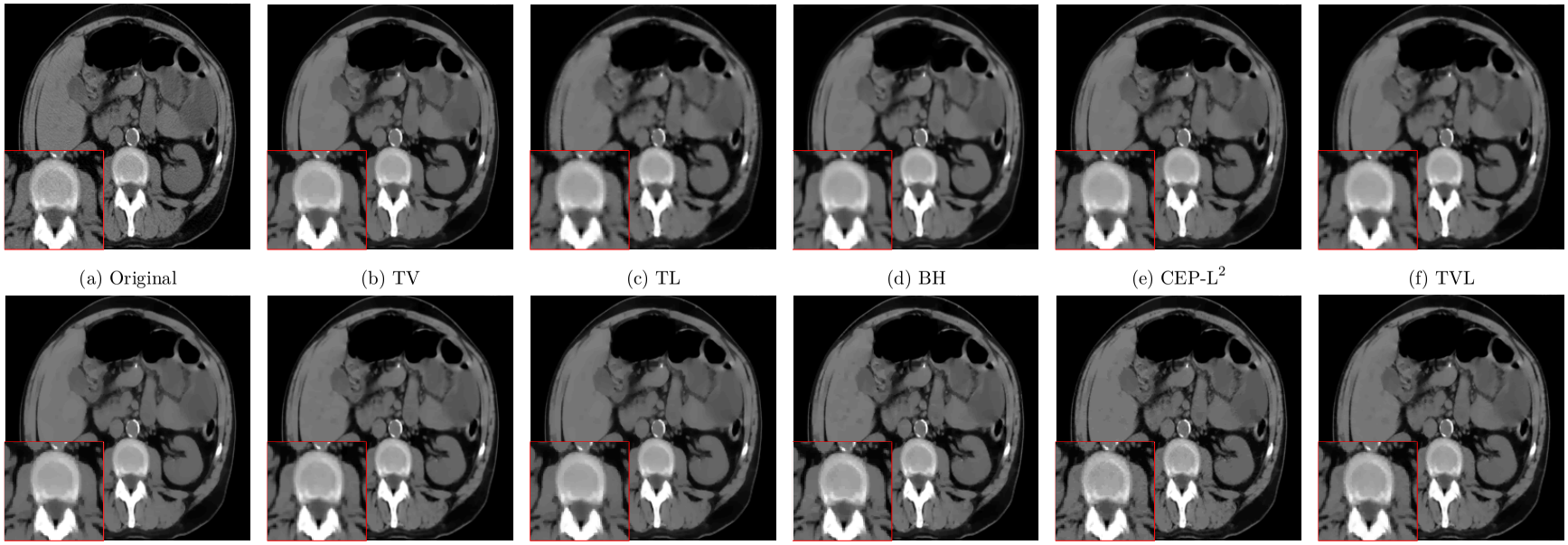
(a) (b)

Figure 9: Average denoising results by eleven different models on six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with five different noise variance (0.01, 0.03, 0.05, 0.07 and 0.1). (a) PSNR (average); (b) SSIM (average).



(a) (b) (c)

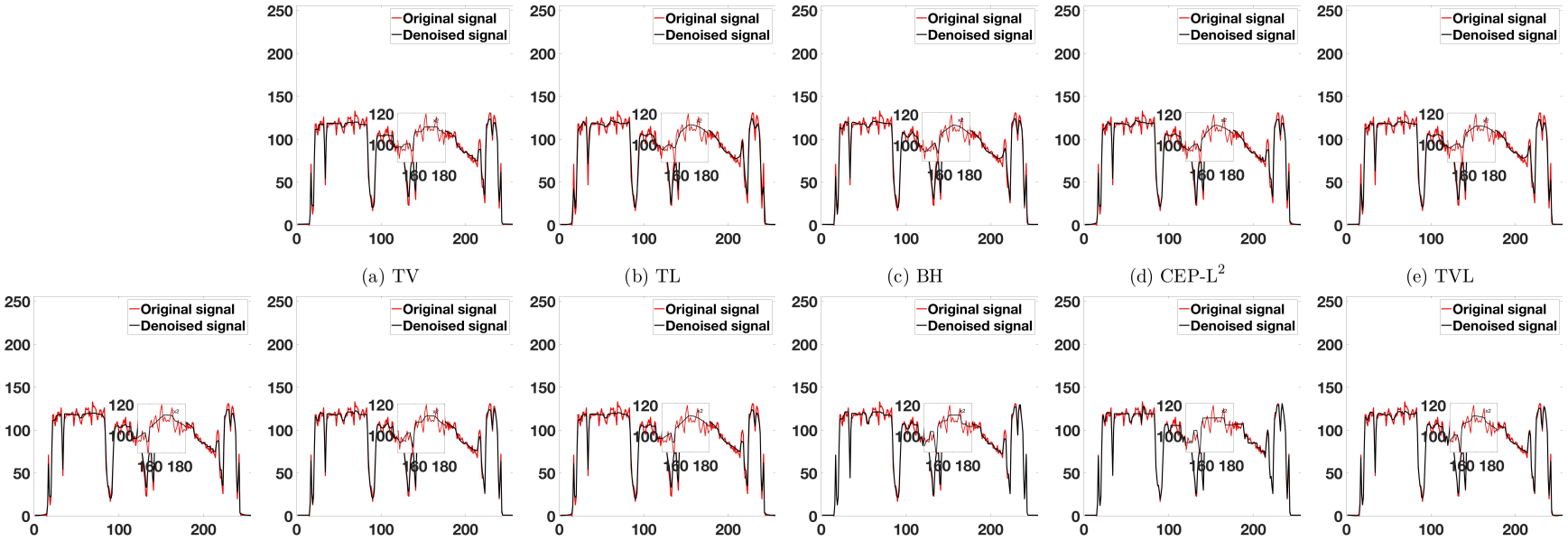
Figure 10: Bar charts of the experimental data in Table 6 (average denoising results by eleven different models on the noisy images of the six real natural images in Figure 1 (d)-(i) with five different noise variance (0.01, 0.03, 0.05, 0.07 and 0.1). (a) Time per iteration in seconds (average); (b) total CPU times in seconds (average); (c) total iterations (average).



(a) Original (b) TV (c) TL (d) BH (e) CEP-L^2 (f) TVL

(g) INFCON (h) TVBH (i) TGV (j) MC (k) GC (l) NewCv

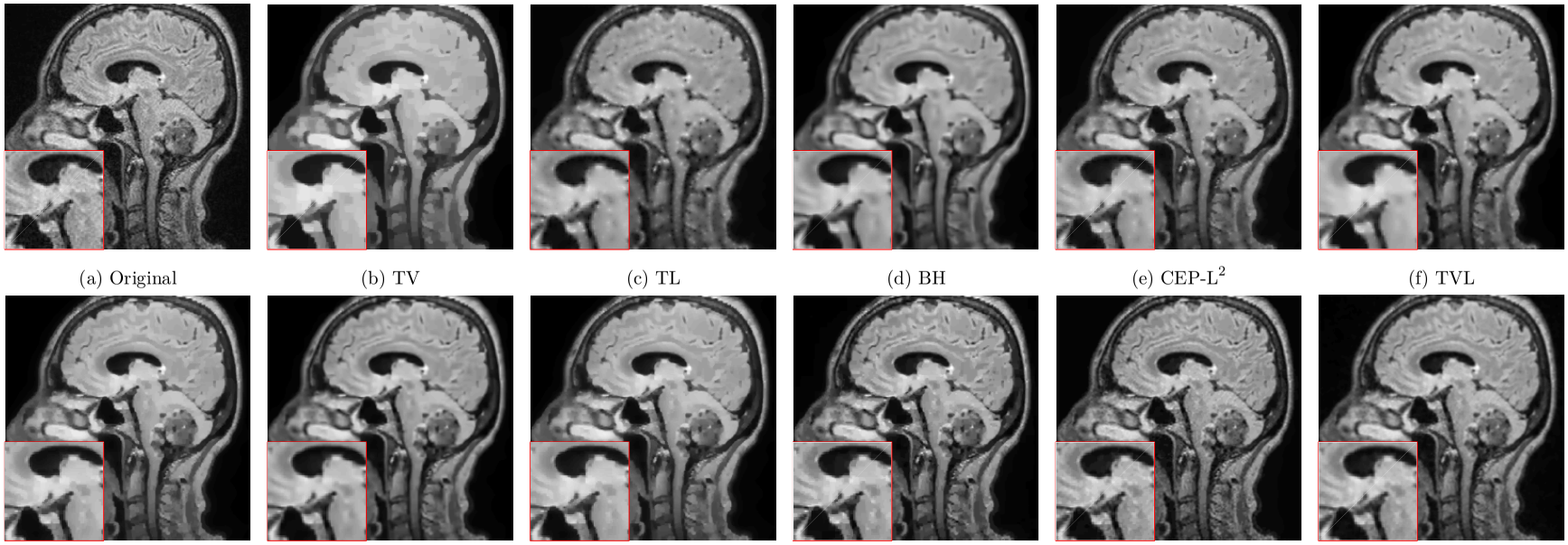
Figure 11: Comparison of the real “Abdomen” image by using 11 different models.



(a) TV (b) TL (c) BH (d) CEP-L^2 (e) TVL

(f) INFCON (g) TVBH (h) TGV (i) MC (j) GC (k) NewCv

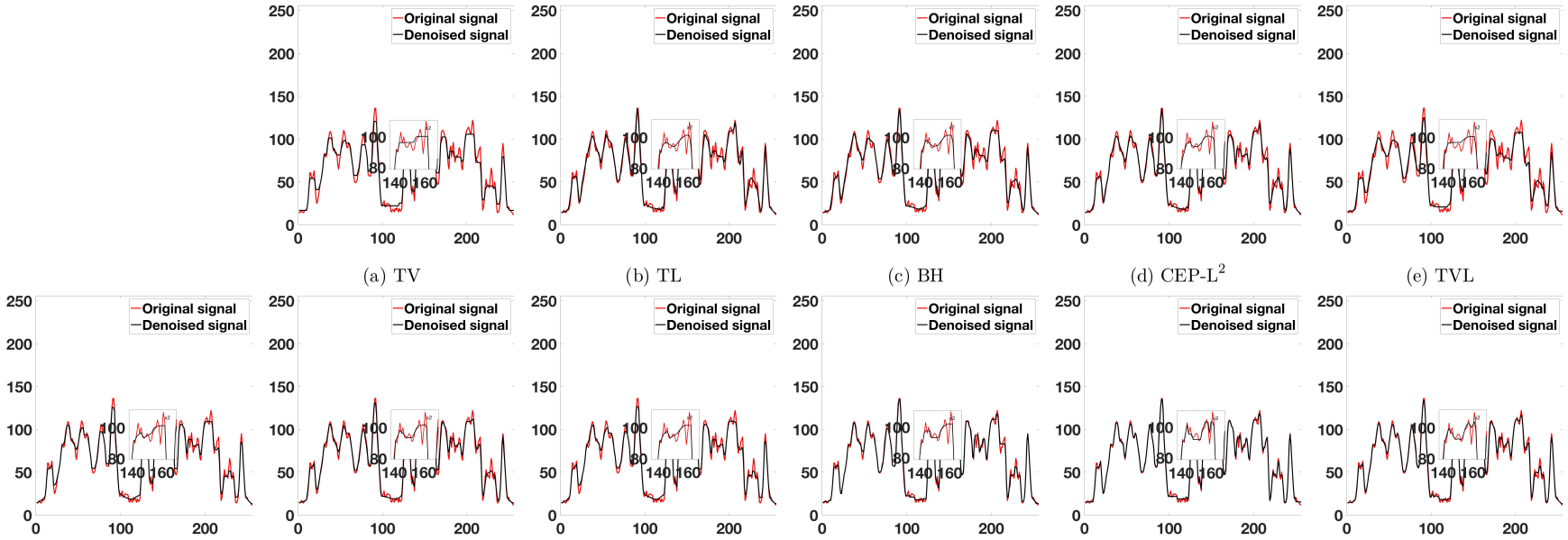
Figure 12: Plots of middle slices of the denoised images in Figure 11 “Abdomen” by eleven different model.



(a) Original (b) TV (c) TL (d) BH (e) CEP-L^2 (f) TVL

(g) INFCON (h) TVBH (i) TGV (j) MC (k) GC (l) NewCv

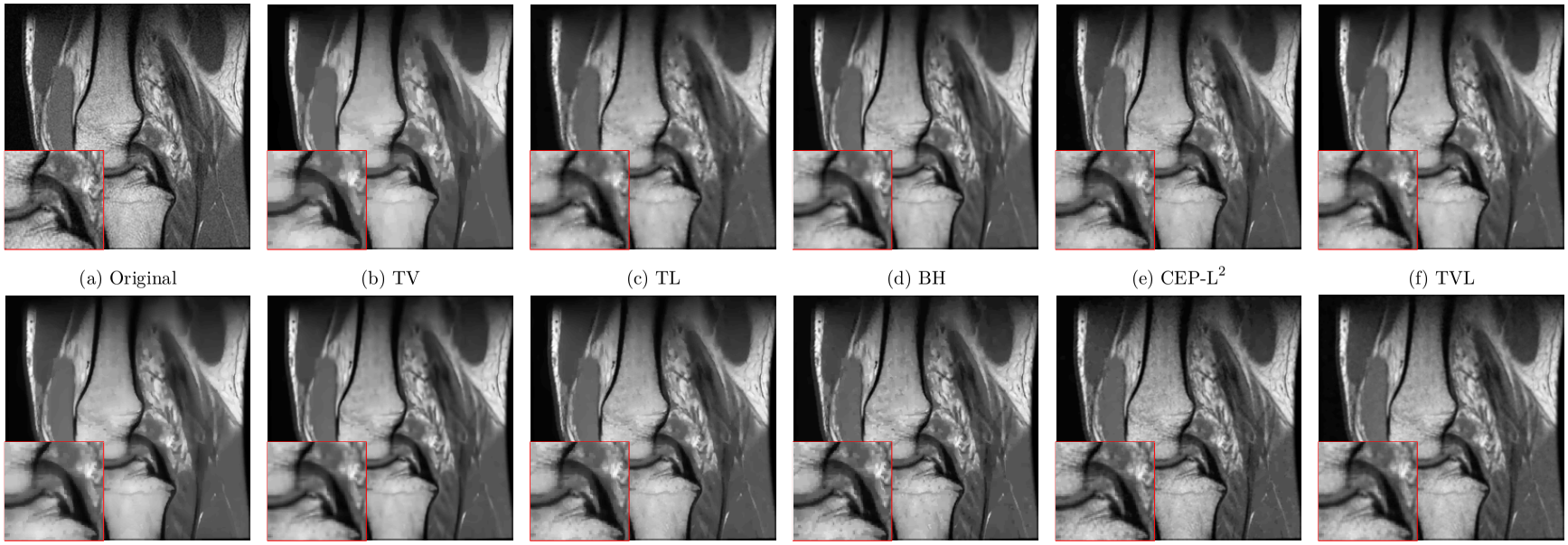
Figure 13: Comparison of the real “Brain” image by using 11 different models.



(a) TV (b) TL (c) BH (d) CEP-L^2 (e) TVL

(f) INFCON (g) TVBH (h) TGV (i) MC (j) GC (k) NewCv

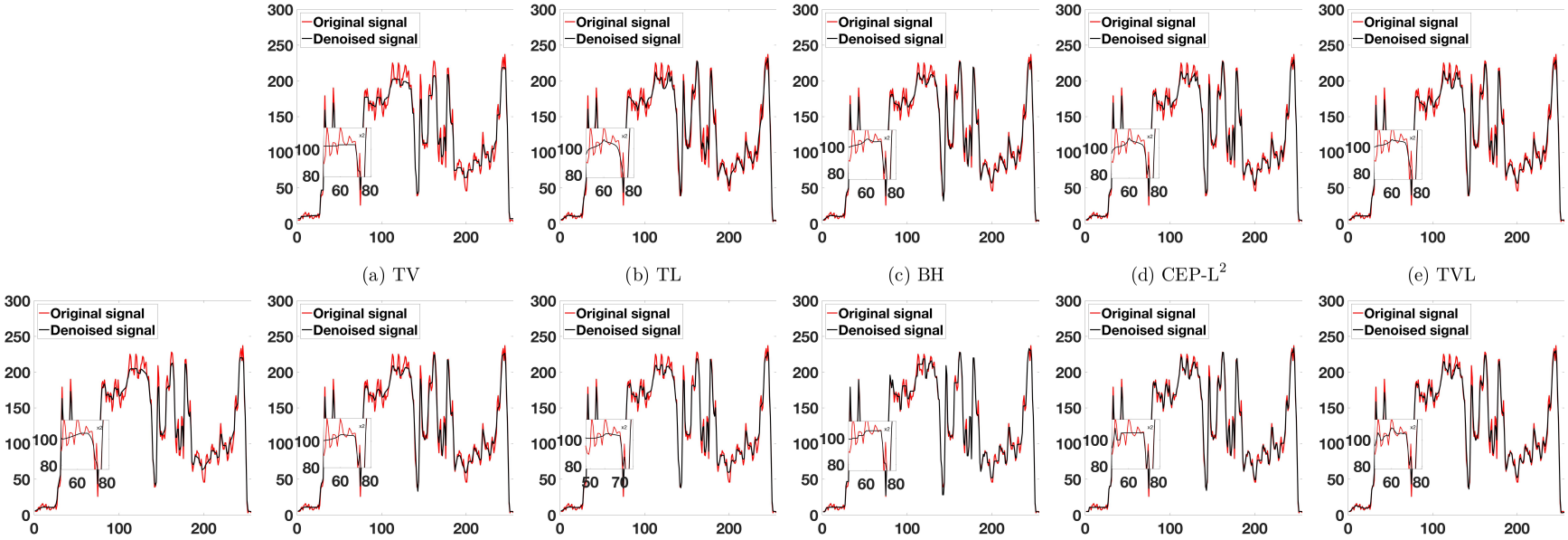
Figure 14: Plots of middle slices of the denoised images in Figure 13 “Brain” by eleven different models.



(a) Original (b) TV (c) TL (d) BH (e) CEP-L^2 (f) TVL

(g) INFCON (h) TVBH (i) TGV (j) MC (k) GC (l) NewCv

Figure 15: Comparison of the real "Knee" image by using eleven different models.



(a) TV (b) TL (c) BH (d) CEP-L^2 (e) TVL

(f) INFCON (g) TVBH (h) TGV (i) MC (j) GC (k) NewCv

Figure 16: Plots of middle slices of the denoised images in Figure 15 “Knee” by eleven different models.

5. Conclusion and future direction

In this work, a new high-order variational model to restore noisy images corrupted by Gaussian noise was proposed. Based on the good feature of high-order regularization, we proposed a curvature-based regularization. In order to solve the new model, the split Bregman algorithm was developed. Compared with some state-of-the-art models, experimental results revealed that the new model effectively preserves edges and reduces the staircase effect. The idea from this project can be extended to remove speckle noise from real ultrasound images.

Appendix A – Derivation of the Euler-Lagrange Equation for (17)

Consider the proposed model in (17)

$$\min_u \mathcal{J}_{\mathrm{AN}}^{\mathrm{NewCv}}(u)$$

where

$$\mathcal{J}_{\mathrm{AN}}^{\mathrm{NewCv}}(u) = \mathcal{D}_{\mathrm{AN}}(u,z) + \gamma\mathcal{R}^{\mathrm{NewCv}}(u).$$

In order to compute the first variation of $\mathcal{J}_{\mathrm{AN}}^{\mathrm{NewCv}}$ for u in a direction of v in a test space $\mathcal{V}$, we will find the first variations of $\mathcal{D}_{\mathrm{AN}}$, $\mathcal{R}_1$ and $\mathcal{R}_2$ respect to u, and apply the sum rule.

The first variation of $\mathcal{D}_{\mathrm{AN}}$ for u in the direction of v is given by

$$\begin{aligned}
\frac{\delta\mathcal{D}_{\mathrm{AN}}(u,z;v)}{\delta u} &= \left.\frac{d\mathcal{D}_{\mathrm{AN}}((u+\varepsilon v),z)}{d\varepsilon}\right|_{\epsilon=0} \\
&= \left.\frac{d}{d\varepsilon}\left\{\frac{1}{2}\int_\Omega ((u+\varepsilon v)-z)^2 \; d\Omega\right\}\right|_{\varepsilon=0}, \\
&= \left.\frac{1}{2}\int_\Omega \frac{d}{d\varepsilon}\left((u+\varepsilon v)-z\right)^2\right|_{\varepsilon=0} d\Omega, \\
&= \int_\Omega (u-z)\, v \; d\Omega. \qquad (64)
\end{aligned}$$

Let $\mathcal{R}_1(u) = \frac{1}{2}\int_\Omega \kappa_{\mathrm{M}}^2(u)d\Omega$ and $\mathcal{R}_2(u) = \frac{1}{2}\int_\Omega -2\kappa_{\mathrm{G}}(u)d\Omega$. Therefore

$$\mathcal{R}^{\mathrm{NewCv}}(u) = \mathcal{R}_1(u) + \mathcal{R}_2(u).$$

Thus, we have

$$\begin{aligned}
\frac{\delta\mathcal{R}_1(u;v)}{\delta u} &= \frac{d}{d\varepsilon}\left(\frac{1}{2}\int_\Omega \kappa_{\mathrm{M}}^2(u+\varepsilon v)\; d\Omega\right)\Big|_{\varepsilon=0} \\
&= \frac{1}{2}\int_\Omega \left.\frac{d}{d\varepsilon}\kappa_{\mathrm{M}}^2(u+\varepsilon v)\right|_{\varepsilon=0} d\Omega \\
&= \frac{1}{2}\int_\Omega 2\kappa_{\mathrm{M}}(u)\; \left.\frac{d}{d\varepsilon}(\kappa_{\mathrm{M}}(u+\varepsilon v))\right|_{\varepsilon=0} d\Omega \\
&= \int_\Omega \kappa_{\mathrm{M}}(u)\; \left.\frac{d}{d\varepsilon}(\nabla\cdot\frac{\nabla(u+\varepsilon v)}{\sqrt{1+|\nabla(u+\varepsilon v)|^2}})\right|_{\varepsilon=0} d\Omega \\
&= \int_\Omega \kappa_{\mathrm{M}}(u)(\nabla\cdot\left.\frac{d}{d\varepsilon}\;\frac{\nabla(u+\varepsilon v)}{\sqrt{1+|\nabla(u+\varepsilon v)|^2}})\right|_{\varepsilon=0} d\Omega
\end{aligned}$$

or

$$\begin{aligned}
\frac{\delta \mathcal{R}_1(u,v)}{\delta u} &= \int_\Omega \kappa_{\mathrm{M}}(u) \left(\nabla \cdot \left(\frac{1}{\sqrt{1+|\nabla(u+\varepsilon v)|^2)}} \frac{d}{d\epsilon} \nabla(u+\epsilon v) \Big|_{\epsilon=0} \right.\right. \\
&\qquad \left.\left. + \nabla(u+\epsilon v) \frac{d}{d\epsilon} \frac{1}{\sqrt{1+|\nabla(u+\varepsilon v)|^2)}} \Big|_{\epsilon=0} \right) \right) d\Omega \\
&= \int_\Omega \kappa_{\mathrm{M}}(u) \left(\nabla \cdot \left(\frac{\nabla v}{\sqrt{1+|\nabla u|^2)}} \right.\right. \\
&\qquad + \nabla u \sum_{m=1}^{2} \frac{\partial [1+(u_{x_1}+\epsilon v_{x_1})^2 + (u_{x_2}+\epsilon v_{x_2})^2]^{-1/2}}{\partial (u_{x_m}+\epsilon v_{x_m})} \times \\
&\qquad \left.\left. \frac{\partial (u_{x_m}+\epsilon v_{x_m})^2}{\partial (u_{x_m}+\epsilon v_{x_m})} \frac{\partial (u_{x_m}+\epsilon v_{x_m})}{\partial \epsilon} \Big|_{\epsilon=0} \right) \right) d\Omega \\
&= \int_\Omega \kappa_{\mathrm{M}}(u) (\nabla \cdot (\frac{\nabla v}{\sqrt{1+|\nabla u|^2}} - \nabla u \sum_{m=1}^{2} \frac{u_{x_m} v_{x_m}}{\sqrt{1+|\nabla u|^2}^3})) d\Omega \\
&= \int_\Omega \kappa_{\mathrm{M}}(u) (\nabla \cdot (\frac{\nabla v}{\sqrt{1+|\nabla u|^2}} - (\nabla u \frac{\nabla u}{(\sqrt{1+|\nabla u|^2})^3}) \nabla v)) d\Omega.
\end{aligned}$$

After applying the divergence theorem, we get

$$\begin{aligned}
\frac{\delta \mathcal{R}_1(u,v)}{\delta u} &= \int_{\partial\Omega} \kappa_{\mathrm{M}}(u) \langle \frac{\nabla v}{\sqrt{1+|\nabla u|^2}} - (\nabla u \frac{\nabla u}{(\sqrt{1+|\nabla u|^2})^3}) \nabla v, \boldsymbol{n} \rangle d\boldsymbol{s} \\
&\quad - \int_\Omega \langle \kappa_{\mathrm{M}}(u), \frac{\nabla v}{\sqrt{1+|\nabla u|^2}} - (\nabla u \frac{\nabla u}{(\sqrt{1+|\nabla u|^2})^3}) \nabla v \rangle d\Omega \\
&= \int_{\partial\Omega} \kappa_{\mathrm{M}}(u) \langle \frac{1}{\sqrt{1+|\nabla u|^2}} (\mathbf{I} - (\nabla u \frac{\nabla u}{|\nabla u|^2})) \nabla v, \boldsymbol{n} \rangle d\boldsymbol{s} \\
&\quad - \int_\Omega \langle \nabla(\kappa_{\mathrm{M}}(u), \frac{1}{\sqrt{1+|\nabla u|^2}} (\mathbf{I} - (\nabla u \frac{\nabla u}{(\sqrt{1+|\nabla u|^2})^2})) \nabla v \rangle d\Omega \\
&= \int_{\partial\Omega} \kappa_{\mathrm{M}}(u) \langle \frac{1}{\sqrt{1+|\nabla u|^2}} \left(\mathbf{I} - \mathbf{P}\right) \nabla v, \boldsymbol{n} \rangle d\boldsymbol{s} \\
&\quad - \int_\Omega \langle \frac{1}{\sqrt{1+|\nabla u|^2}} \left(\mathbf{I} - \mathbf{P}\right) \nabla(\kappa_{\mathrm{M}}(u)), \nabla v \rangle d\mathbf{x}.
\end{aligned}$$

Here $\mathbf{I}(\mathbf{x}) = \mathbf{x}$ is the identity transform and $\mathbf{P}(\mathbf{x}) = (\mathbf{x} \cdot \frac{\nabla u}{\sqrt{1+|\nabla u|^2}}) \frac{\nabla u}{\sqrt{1+|\nabla u|^2}}$ is the orthogonal projection onto the normal direction. By using the divergence theorem with the second term, we have

$$\begin{aligned}
\frac{\delta \mathcal{R}_1(u,v)}{\delta u} &= \int_{\partial\Omega} \kappa_{\mathrm{M}}(u) \langle \frac{1}{\sqrt{1+|\nabla u|^2}} \left(\mathbf{I} - \mathbf{P}\right) \nabla v, \boldsymbol{n} \rangle \; d\boldsymbol{s} \\
&\quad + \int_\Omega (\nabla \cdot (\frac{1}{\sqrt{1+|\nabla u|^2}} \left(\mathbf{I} - \mathbf{P}\right) \nabla \kappa_{\mathrm{M}}(u)))) v \; d\Omega \\
&\quad - \int_{\partial\Omega} \langle \frac{1}{\sqrt{1+|\nabla u|^2}} \left(\mathbf{I} - \mathbf{P}\right) \nabla(\kappa_{\mathrm{M}}(u), \boldsymbol{n} \rangle v d\boldsymbol{s}
\end{aligned}$$

or

$$\frac{\delta \mathcal{R}_1(u;v)}{\delta u} = \int_\Omega (\nabla \cdot (\frac{1}{\sqrt{1+|\nabla u|^2}} \nabla \kappa_{\mathrm{M}}(u) - \frac{\nabla u \cdot \nabla \kappa_{\mathrm{M}}(u)}{(\sqrt{1+|\nabla u|^2})^3} \nabla u)) v d\Omega + \int_{\partial\Omega} \kappa_{\mathrm{M}}(u) \langle \frac{1}{\sqrt{1+|\nabla u|^2}} (\mathbf{I} - \mathbf{P}) \nabla v, \boldsymbol{n} \rangle d\boldsymbol{s} - \int_{\partial\Omega} \langle \frac{1}{\sqrt{1+|\nabla u|^2}} (\mathbf{I} - \mathbf{P}) \nabla (\kappa_{\mathrm{M}}(u), \boldsymbol{n} \rangle v d\boldsymbol{s}. \tag{65}$$

For $\mathcal{R}_2$, we have

$$\begin{aligned}
\frac{\delta \mathcal{R}_2(u;v)}{\delta u} &= -\frac{d}{d\varepsilon} \int_\Omega \kappa_{\mathrm{G}}(u+\varepsilon v)\, d\Omega \Big|_{\varepsilon=0} \\
&= -\int_\Omega \frac{d}{d\varepsilon} \kappa_{\mathrm{G}}(u+\varepsilon v) \Big|_{\varepsilon=0} d\Omega \\
&= -\int_\Omega \frac{d}{d\varepsilon} \frac{(u+\varepsilon v)_{x_1x_1}(u+\varepsilon v)_{x_2x_2} - (u+\varepsilon v)_{x_1x_2}(u+\varepsilon v)_{x_2x_1}}{(1+|\nabla(u+\varepsilon v)|^2)^2} \Big|_{\varepsilon=0} d\Omega \\
&= \int_\Omega \frac{d}{d\varepsilon} \frac{(u+\varepsilon v)_{x_1x_2}(u+\varepsilon v)_{x_2x_1} - (u+\varepsilon v)_{x_1x_1}(u+\varepsilon v)_{x_2x_2}}{(1+|\nabla(u+\varepsilon v)|^2)^2} \Big|_{\varepsilon=0} d\Omega \\
&= \Big[\int_\Omega \frac{u_{x_1x_2}v_{x_2x_1} - u_{x_1x_1}v_{x_2x_2} + u_{x_2x_1}v_{x_1x_2} - u_{x_2x_2}v_{x_1x_1}}{(1+|\nabla(u)|^2)^2}\, d\Omega \\
&\quad - \int_\Omega \frac{4(u_{x_1x_2}u_{x_2x_1} - u_{x_1x_1}u_{x_2x_2})(u_{x_1}v_{x_1} + u_{x_2}v_{x_2})}{(1+|\nabla(u)|^2)^3}\, d\Omega \Big] \\
&= \Big[\int_\Omega \frac{u_{x_1x_2}v_{x_2x_1} - u_{x_1x_1}v_{x_2x_2}}{(1+|\nabla(u)|^2)^2}\, d\Omega + \int_\Omega \frac{u_{x_2x_1}v_{x_1x_2} - u_{x_2x_2}v_{x_1x_1}}{(1+|\nabla(u)|^2)^2}\, d\Omega \\
&\quad - \int_\Omega \frac{4(u_{x_1x_2}u_{x_2x_1} - u_{x_1x_1}u_{x_2x_2})(u_{x_1}v_{x_1} + u_{x_2}v_{x_2})}{(1+|\nabla(u)|^2)^3}\, d\Omega \Big].
\end{aligned} \tag{66}$$

Applying the divergence theorem and integrating by parts yields

$$\begin{aligned}
&\int_\Omega \frac{u_{x_1x_2}v_{x_2x_1} - u_{x_1x_1}v_{x_2x_2}}{(1+|\nabla(u)|^2)^2}\, d\Omega + \int_\Omega \frac{u_{x_2x_1}v_{x_1x_2} - u_{x_2x_2}v_{x_1x_1}}{(1+|\nabla(u)|^2)^2}\, d\Omega \\
&= \Big[\int_{\partial\Omega} \langle (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2}, \frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2}) v_{x_2}, \boldsymbol{n} \rangle d\boldsymbol{s} \\
&\quad - \int_\Omega ((\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1} + (\frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2}) v_{x_2} d\Omega \\
&\quad + \int_{\partial\Omega} \langle (\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2}, \frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2}) v_{x_1}, \boldsymbol{n} \rangle d\boldsymbol{s} \\
&\quad - \int_\Omega ((\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1} + (\frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2}) v_{x_1} d\Omega \Big]
\end{aligned}$$

or

$$\begin{aligned}
&\int_\Omega \frac{u_{x_1x_2}v_{x_2x_1} - u_{x_1x_1}v_{x_2x_2}}{(1+|\nabla(u)|^2)^2}\, d\Omega + \int_\Omega \frac{u_{x_2x_1}v_{x_1x_2} - u_{x_2x_2}v_{x_1x_1}}{(1+|\nabla(u)|^2)^2}\, d\Omega \\
&= \Big[\int_{\partial\Omega} \langle (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2}, \frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})v_{x_2}, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad + \int_{\partial\Omega} \langle (\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2}, \frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})v_{x_1}, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad - \int_\Omega ((\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1})v_{x_1} + ((\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1})v_{x_2} d\Omega \\
&\quad + \int_\Omega ((\frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2})v_{x_1} + ((\frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2})v_{x_2} d\Omega\Big]
\end{aligned} \tag{67}$$

Applying the divergence theorem with the last two terms of (67) lead to

$$\begin{aligned}
&\int_\Omega \frac{u_{x_1x_2}v_{x_2x_1} - u_{x_1x_1}v_{x_2x_2}}{(1+|\nabla(u)|^2)^2}\, d\Omega + \int_\Omega \frac{u_{x_2x_1}v_{x_1x_2} - u_{x_2x_2}v_{x_1x_1}}{(1+|\nabla(u)|^2)^2}\, d\Omega \\
&= \Big[\int_{\partial\Omega} \langle (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2}, \frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})v_{x_2}, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad + \int_{\partial\Omega} \langle (\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2}, \frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})v_{x_1}, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad + \int_\Omega ((\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1x_1}) + (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1x_2})v d\Omega \\
&\quad - \int_{\partial\Omega} \langle ((\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1}, (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1})v, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad + \int_\Omega ((\frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2x_1}) + (\frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2x_2})v d\Omega \\
&\quad - \int_{\partial\Omega} \langle ((\frac{-u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2}, (\frac{u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2})v, \boldsymbol{n}\rangle d\boldsymbol{s}\Big]
\end{aligned}$$

or

$$\begin{aligned}
&\int_\Omega \frac{u_{x_1x_2}v_{x_2x_1} - u_{x_1x_1}v_{x_2x_2}}{(1+|\nabla(u)|^2)^2}\, d\Omega + \int_\Omega \frac{u_{x_2x_1}v_{x_1x_2} - u_{x_2x_2}v_{x_1x_1}}{(1+|\nabla(u)|^2)^2}\, d\Omega \\
&= \Big[\int_{\partial\Omega} \langle (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2}, \frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})v_{x_2}, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad + \int_{\partial\Omega} \langle (\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2}, \frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})v_{x_1}, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad - \int_{\partial\Omega} \langle ((\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1}, (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1})v, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad - \int_{\partial\Omega} \langle ((\frac{-u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2}, (\frac{u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2})v, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad + \int_\Omega \langle (\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1x_1} + (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1x_2}, v\rangle d\Omega \\
&\quad + \int_\Omega \langle (\frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2x_1} + (\frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2x_2}, v\rangle d\Omega\Big].
\end{aligned} \tag{68}$$

Similarly, we have

$$
\begin{aligned}
&-\int_{\Omega}\frac{4(u_{x_1x_2}u_{x_2x_1}-u_{x_1x_1}u_{x_2x_2})(u_{x_1}v_{x_1}+u_{x_2}v_{x_2})}{(1+|\nabla(u)|^2)^3}\,d\Omega\\
&\quad=\Big[\int_{\Omega}\langle\nabla\cdot(\frac{4(u_{x_1x_2}u_{x_2x_1}-u_{x_1x_1}u_{x_2x_2})}{(1+|\nabla(u)|^2)^3})\nabla u,v\rangle d\Omega\\
&\quad-\int_{\partial\Omega}\langle(\frac{4(u_{x_1x_2}u_{x_2x_1}-u_{x_1x_1}u_{x_2x_2})}{(1+|\nabla(u)|^2)^3}\nabla u)v,\boldsymbol{n}\rangle d\boldsymbol{s}\Big].
\end{aligned}\tag{69}
$$

Substituting (68) and (69) into (66) and rearranging the terms yield

$$
\begin{aligned}
\frac{\delta}{\delta u}\mathcal{R}_2(u;v)=&\int_{\partial\Omega}\frac{1}{(1+|\nabla(u)|^2)^2}v_{x_2}\langle(u_{x_1x_2},-u_{x_1x_1}),\boldsymbol{n}\rangle d\boldsymbol{s}\\
&+\int_{\partial\Omega}\frac{1}{(1+|\nabla(u)|^2)^2}v_{x_1}\langle(-u_{x_2x_2},u_{x_2x_1}),\boldsymbol{n}\rangle d\boldsymbol{s}\\
&+\int_{\partial\Omega}\langle((\frac{u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1},(\frac{-u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1}),\boldsymbol{n}\rangle v d\boldsymbol{s}\\
&+\int_{\partial\Omega}\langle((\frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2},-(\frac{u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2}),\boldsymbol{n}\rangle v d\boldsymbol{s}\\
&+\int_{\Omega}\nabla\cdot((\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1},(\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1})v d\Omega\\
&+\int_{\Omega}\nabla\cdot((\frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2},(\frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2})v d\Omega\\
&+\int_{\Omega}(\nabla\cdot\left(\frac{4(u_{x_1x_2}u_{x_2x_1}-u_{x_1x_1}u_{x_2x_2})}{(1+|\nabla(u)|^2)^3}\nabla u\right)v d\Omega\\
&-\int_{\partial\Omega}\langle\frac{4(u_{x_1x_2}u_{x_2x_1}-u_{x_1x_1}u_{x_2x_2})}{(1+|\nabla(u)|^2)^3}\nabla u,\boldsymbol{n}\rangle v d\boldsymbol{s}.
\end{aligned}\tag{70}
$$

By dropping the boundary terms, a function u minimizing the joint functional $\mathcal{J}_{\mathrm{AN}}^{\mathrm{NewCv}}(u)$ in (17) has to satisfy the following partial differential equation

$$f(u,z)+\gamma\nabla\cdot\boldsymbol{V}=0,$$

subject to the natural boundary conditions

$$
\begin{aligned}
\kappa_{\mathrm{M}}(u)&=0,\\
-\frac{\partial\kappa_{\mathrm{M}}(u)}{\partial\boldsymbol{n}}+\frac{\nabla\kappa_{\mathrm{M}}(u)\cdot\nabla u}{(1+|\nabla u|^2)^2}\frac{\partial u}{\partial\boldsymbol{n}}&=0,\\
(u_{x_1x_2},-u_{x_1x_1})\cdot\boldsymbol{n}&=0,\\
(-u_{x_2x_2},u_{x_2x_1})\cdot\boldsymbol{n}&=0,\\
((\frac{u_{x_2x_2}}{(1+|\nabla u|^2)^2})_{x_1},-(\frac{u_{x_1x_2}}{(1+|\nabla u|^2)^2}))_{x_1}\cdot\boldsymbol{n}&=0,\\
((\frac{u_{x_2x_1}}{(1+|\nabla u|^2)^2})_{x_2},-(\frac{u_{x_1x_1}}{(1+|\nabla u|^2)^2}))_{x_2}\cdot\boldsymbol{n}&=0,\\
\frac{4(u_{x_1x_2}u_{x_2x_1}-u_{x_1x_1}u_{x_2x_2})}{(1+|\nabla u|^2)^3}\frac{\partial u}{\partial\boldsymbol{n}}&=0,
\end{aligned}
$$

where $\boldsymbol{V} = \boldsymbol{V}_1 + \boldsymbol{V}_2 + \boldsymbol{V}_3 + \boldsymbol{V}_4 + \boldsymbol{V}_5$,

$$\boldsymbol{V}_1 = \frac{1}{\sqrt{1+|\nabla u|^2}}\nabla\kappa_{\mathrm{M}}(u),\ \boldsymbol{V}_2 = -\frac{\nabla u\cdot\nabla\kappa_{\mathrm{M}}(u)}{(\sqrt{1+|\nabla u|^2})^3}\nabla u,$$
$$\boldsymbol{V}_3 = \frac{4(u_{x_1x_2}u_{x_2x_1} - u_{x_1x_1}u_{x_2x_2})}{(1+|\nabla u|^2)^3}\nabla u,\ \boldsymbol{V}_4 = ((-\frac{u_{x_2x_2}}{(1+|\nabla u|^2)^2})_{x_1}, (\frac{u_{x_1x_2}}{(1+|\nabla u|^2)^2})_{x_1}),$$
$$\boldsymbol{V}_5 = ((\frac{u_{x_2x_1}}{(1+|\nabla u|^2)^2})_{x_2}, (-\frac{u_{x_1x_1}}{(1+|\nabla u|^2)^2})_{x_2})$$

Appendix B – Derivation of the Euler-Lagrange Equation for p-subproblem

Here our aim is to derive the Euler-Lagrange equation for $\boldsymbol{p}$-subproblem.

$$\min_{\boldsymbol{p}} \overline{\mathcal{E}}\left(\boldsymbol{p}\right),$$

where

$$\begin{aligned}\overline{\mathcal{E}}\left(\boldsymbol{p}\right) = \overline{\mathcal{E}}\left(p_1, p_2\right) = -\gamma \int_\Omega \frac{p_{1_x} p_{2_y} - p_{1_y} p_{2_x}}{(p_1^2 + p_2^2 + 1)^2} d\Omega + \frac{\theta_1}{2} \int_\Omega (\boldsymbol{p} - \nabla u - \boldsymbol{b_1})^2 d\Omega \\ + \frac{\theta_2}{2} \int_\Omega (\boldsymbol{m} - \frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2 + 1}} - \boldsymbol{b_2})^2 d\Omega,\end{aligned}$$

$\boldsymbol{p} = (p_1, p_2)$, $\boldsymbol{m} = (m_1, m_2)$, $\boldsymbol{b}_1 = (b_{11}, b_{12})$ and $\boldsymbol{b}_2 = (b_{21}, b_{22})$.

Let $\mathcal{V}$ be a test space. For any perturbation $\varphi_1 \in \mathcal{V}$, the first variation of $\overline{\mathcal{E}}$ for p_1 in the direction of φ_1 is given by

$$\begin{aligned}\frac{\delta \overline{\mathcal{E}}(p_1, p_2; \varphi_1)}{\delta p_1} &= \frac{d\overline{\mathcal{E}}((p_1 + \varepsilon_1 \varphi_1), p_2)}{d\varepsilon_1}\Big|_{\epsilon_1 = 0} \\ &= \frac{d}{d\varepsilon_1} \Big\{ -\gamma \int_\Omega \frac{(p_1 + \varepsilon_1\varphi_1)_x p_{2_y} - (p_1 + \varepsilon_1\varphi_1)_y p_{2_x}}{((p_1 + \varepsilon_1\varphi_1)^2 + p_2^2 + 1)^2} d\Omega \\ &\quad + \frac{\theta_1}{2} \int_\Omega ((p_1 + \varepsilon_1\varphi_1) - u_x - b_{11})^2 d\Omega \\ &\quad + \frac{\theta_2}{2} \int_\Omega (m_1 - \frac{(p_1 + \varepsilon_1\varphi_1)}{\sqrt{(p_1 + \varepsilon_1\varphi_1)^2 + p_2^2 + 1}} - b_{21})^2 d\Omega \Big\} \Big|_{\varepsilon_1 = 0}.\end{aligned}$$

Thus, we have

$$\begin{aligned}\frac{\delta \overline{\mathcal{E}}(p_1, p_2; \varphi_1)}{\delta p_1} &= -\gamma \int_\Omega \frac{d}{d\varepsilon_1} \frac{(p_1 + \varepsilon_1\varphi_1)_x p_{2_y} - (p_1 + \varepsilon_1\varphi_1)_y p_{2_x}}{((p_1 + \varepsilon_1\varphi_1)^2 + p_2^2 + 1)^2} \Big|_{\varepsilon_1 = 0} d\Omega \\ &\quad + \frac{\theta_1}{2} \int_\Omega \frac{d}{d\varepsilon_1} ((p_1 + \varepsilon_1\varphi_1) - u_x - b_{11})^2 \Big|_{\varepsilon_1 = 0} d\Omega \\ &\quad + \frac{\theta_2}{2} \int_\Omega \frac{d}{d\varepsilon_1} (m_1 - \frac{(p_1 + \varepsilon_1\varphi_1)}{\sqrt{(p_1 + \varepsilon_1\varphi_1)^2 + p_2^2 + 1}} - b_{21})^2 \Big|_{\varepsilon_1 = 0} d\Omega, \\ &= -\gamma \int_\Omega \frac{(p_1^2 + p_2^2 + 1)^2 (\varphi_{1_x} p_{2_y} - \varphi_{1_y} p_{2_x})}{(p_1^2 + p_2^2 + 1)^4} d\Omega \\ &\quad + \gamma \int_\Omega \frac{p_{1_x} p_{2_y} - p_{1_y} p_{2_x}}{(p_1^2 + p_2^2 + 1)^4} \cdot 2(p_1^2 + p_2^2 + 1)(2p_1\varphi_1) d\Omega \\ &\quad + \theta_1 \int_\Omega (p_1 - u_x - b_{11})\varphi_1 \, d\Omega + \theta_2 \int_\Omega (m_1 - \frac{p_1}{\sqrt{p_1^2 + p_2^2 + 1}} - b_{21}) \times \\ &\quad (\frac{p_1^2 \varphi_1}{(\sqrt{p_1^2 + p_2^2 + 1})^3} - \frac{\varphi_1}{\sqrt{p_1^2 + p_2^2 + 1}}) \, d\Omega.\end{aligned}$$

Therefore,

$$\begin{aligned}\frac{\delta\overline{\mathcal{E}}(p_1,p_2;\varphi_1)}{\delta p_1} &= -\gamma\int_\Omega \frac{\varphi_{1_x}p_{2_y}-\varphi_{1_y}p_{2_x}}{\left(p_1^2+p_2^2+1\right)^2}\,d\Omega + 4\gamma\int_\Omega \frac{p_{1_x}p_{2_y}-p_{1_y}p_{2_x}}{\left(p_1^2+p_2^2+1\right)^3}(p_1\varphi_1)\,d\Omega\\ &\quad+\theta_1\int_\Omega (p_1-u_x-b_{11})\varphi_1\,d\Omega+\theta_2\int_\Omega (m_1-\frac{p_1}{\sqrt{p_1^2+p_2^2+1}}-b_{21})\times\\ &\quad(\frac{p_1^2\varphi_1}{(\sqrt{p_1^2+p_2^2+1})^3}-\frac{\varphi_1}{\sqrt{p_1^2+p_2^2+1}})\,d\Omega,\\ &=\gamma\int_\Omega\frac{\varphi_{1_y}p_{2_x}-\varphi_{1_x}p_{2_y}}{\Gamma_2^2}\,d\Omega+\gamma\int_\Omega\Gamma p_1\varphi_1\,d\Omega+\theta_1\int_\Omega(p_1-u_x-b_{11})\varphi_1\,d\Omega\\ &\quad+\theta_2\int_\Omega(m_1-\psi p_1-b_{21})\psi^3p_1^2\varphi_1\,d\Omega-\theta_2\int_\Omega(m_1-\psi p_1-b_{21})\psi\varphi_1\,d\Omega,\end{aligned}$$

where $\Gamma=\frac{4\Gamma_1}{\Gamma_2^3}$, $\Gamma_1=p_{1_x}p_{2_y}-p_{1_y}p_{2_x}$, $\Gamma_2=p_1^2+p_2^2+1$ and $\psi=\dfrac{1}{\sqrt{\Gamma_2}}$.

Let $\boldsymbol{w_1}=(-\frac{p_{2_y}}{\Gamma_2^2},\frac{p_{2_x}}{\Gamma_2^2})$. Thus,

$$\int_\Omega\nabla\varphi_1\cdot\boldsymbol{w_1}\,d\Omega=\int_\Omega\frac{\varphi_{1_y}p_{2_x}-\varphi_{1_x}p_{2_y}}{\Gamma_2^2}\,d\Omega.$$

By Green's Theorem, we have

$$\int_\Omega\nabla\varphi_1\cdot\boldsymbol{w_1}\,d\Omega=\int_{\partial\Omega}\varphi_1(\boldsymbol{w_1}\cdot\boldsymbol{n}_1)\,d\boldsymbol{s}-\int_\Omega\varphi_1(\nabla\cdot\boldsymbol{w_1})\,d\Omega.$$

Therefore,

$$\int_\Omega\frac{\varphi_{1_y}p_{2_x}-\varphi_{1_x}p_{2_y}}{\Gamma_2^2}\,d\Omega=\int_{\partial\Omega}(\boldsymbol{w_1}\cdot\boldsymbol{n}_1)\varphi_1\,d\boldsymbol{s}-\int_\Omega(\nabla\cdot\boldsymbol{w_1})\varphi_1\,d\Omega.$$

It follows that

$$\begin{aligned}\frac{\delta\overline{\mathcal{E}}(p_1,p_2;\varphi_1)}{\delta p_1} &= -\gamma\int_\Omega(\nabla\cdot\boldsymbol{w_1})\varphi_1\,d\Omega+\gamma\int_{\partial\Omega}(\boldsymbol{w_1}\cdot\boldsymbol{n})\varphi_1\,d\boldsymbol{s}+\gamma\int_\Omega\Gamma p_1\varphi_1\,d\Omega\\ &\quad+\theta_1\int_\Omega(p_1-u_x-b_{11})\varphi_1\,d\Omega+\theta_2\int_\Omega(m_1-\psi p_1-b_{21})\psi^3p_1^2\varphi_1\,d\Omega\\ &\quad+\theta_2\int_\Omega\psi^2p_1\varphi_1\,d\Omega-\theta_2\int_\Omega(m_1-b_{21})\psi\varphi_1\,d\Omega\\ &=\gamma\int_\Omega(\Gamma p_1-\nabla\cdot\boldsymbol{w_1})\varphi_1\,d\Omega+\gamma\int_{\partial\Omega}(\boldsymbol{w_1}\cdot\boldsymbol{n})\varphi_1\,d\boldsymbol{s}+\theta_1\int_\Omega(p_1-u_x-b_{11})\varphi_1\,d\Omega\\ &\quad+\theta_2\int_\Omega((m_1-\psi p_1-b_{21})\psi p_1+1)\psi^2p_1\varphi_1\,d\Omega-\theta_2\int_\Omega(m_1-b_{21})\psi\varphi_1\,d\Omega.\end{aligned}$$

Similarly, the first variation of $\overline{\mathcal{E}}$ for p_2 in the direction of $\varphi_2\in\mathcal{V}$ is given by

$$\begin{aligned}\frac{\delta\overline{\mathcal{E}}(p_1,p_2;\varphi_2)}{\delta p_2} &= \gamma\int_\Omega(\Gamma p_2-\nabla\cdot\boldsymbol{w_2})\varphi_2\,d\Omega+\gamma\int_{\partial\Omega}(\boldsymbol{w_2}\cdot\boldsymbol{n})\varphi_2\,dS+\theta_1\int_\Omega(p_2-u_y-b_{12})\varphi_2\,d\Omega\\ &\quad+\theta_2\int_\Omega((m_2-\psi p_2-b_{22})\psi p_2+1)\psi^2p_2\varphi_2\,d\Omega-\theta_2\int_\Omega(m_2-b_{22})\psi\varphi_2\,d\Omega,\end{aligned}$$

where

$$\boldsymbol{w_2}=(\frac{p_{1_y}}{\Gamma_2^2},-\frac{p_{1_x}}{\Gamma_2^2}).$$

Therefore, a function $\boldsymbol{p} = (p_1, p_2)$ minimizing the energy functional $\overline{\mathcal{E}}$ for $\boldsymbol{p}$-subproblem has to satisfy the following Euler-Lagrange equation

$$\begin{cases} -\gamma\nabla\cdot\boldsymbol{w}_1 + \gamma\Gamma p_1 + \theta_1(p_1 - u_x - b_{11}) + \theta_2(m_1 - \psi p_1 - b_{21})(\psi^3 p_1^2 - \psi) = 0 \\ -\gamma\nabla\cdot\boldsymbol{w}_2 + \gamma\Gamma p_2 + \theta_1(p_2 - u_y - b_{12}) + \theta_2(m_2 - \psi p_2 - b_{22})(\psi^3 p_2^2 - \psi) = 0 \end{cases}, \tag{71}$$

subject to the natural boundary condition $\boldsymbol{w_1}\cdot\boldsymbol{n} = 0$ and $\boldsymbol{w_2}\cdot\boldsymbol{n} = 0$.

References

[1] T. Loupas, W. McDicken, P. Allan, An adaptive weighted median filter for speckle suppression in medical ultrasound images, IEEE Transactions on Circuits and Systems 36 (1) (1989) 129–135.

[2] J. Koo, S. Oark, Speckle reduction with edge preservation in medical ultrasonic images using a homogenous region growing mean filter (HRGMF), Ultrasonic Imaging 13 (1990) 629–639.

[3] O. Michailovich, A. Tannenbaum, An accurate and efficient baysian model for automatic segmetation of brain MRI, IEEE Trans. Ultrason., Ferroelect., Freq. Contr. 53 (1) (2006) 64–78.

[4] P. Hellier, C. Kervrann, C. Barillot, Nonlocal means-based speckle filtering for ultrasound images, IEEE Trans. Image Process. 18 (10) (2009) 2221–2229.

[5] X. Hao, S. Gao, X. Gao, A novel multiscale nonlinear thresholding method for ultrasonic speckle suppressing, IEEE Trans. Med. Imaging. 18 (9) (1999) 787–794.

[6] J. Koo, S. Oark, Despeckling of medical ultrasound images using daubechies complex wavelet transform, Signal Processing 90 (2010) 428–439.

[7] F. Adamo, G. Andria, F. Attivissimo, A. Lanzolla, M. Spadavecchia, A comparative study on mother wavelet selection in ultrasound image denoising, Measurement 46 (2013) 2447–2456.

[8] A. Buades, B. Coll, J. M. Morel, A review of image denoising algorithms, with a new one, Multiscale Model. Simul. 4 (2) (2005) 490–530.

[9] P. Coupé, P. Hellier, C. Kervrann, C. Barillot, Nonlocal means-based speckle filtering for ultrasound images, IEEE Trans. Image Process. 18 (10) (2009) 2221–2229.

[10] P. Sudeep, P. Palanisamy, J. Rajan, H. Baradaran, L. Saba, A. Gupta, J. Suri, Speckle reduction in medical ultrasound images using an unbiased non-local means method, Biomedical Signal Processing and Control 28 (2016) 1–8.

[11] Q. Chen, P. Montesinos, Q. Sun, D. Xia, Ramp preserving Perona-Malik model, Signal Processing 90 (2010) 1963–1975.

[12] H. Yua, J.L.Tana, Y.Y.Wang, Ultrasound speckle reduction by a SUSAN-controlled anisotropic diffusion method, Pattern Recognition 43 (2010) 3083–3092.

[13] M. Islama, S. Mahbubur-Rahman, M. Omair-Ahmad, M. Swamy, Mixed Gaussian-impulse noise reduction from images using convolutional neural network, Signal Processing: Image Communication 68 (2018) 26–41.

[14] F. Zhang, N. Cai, J. Wu, G. Cen, H. Wang, X. Chen, Image denoising method based on a deep convolution neural network, IET Image Processing 12 (4) (2018) 485–493.

[15] W. Zuo, K. Zhang, L. Zhang, Convolutional Neural Networks for Image Denoising and Restoration, in Advances in Computer Vision and Pattern Recognition Series: Denoising of Photographic Images and Video, edited by M. Bertalmío , Springer, Cham, pp. 93-123, 2018.

[16] H. Shahdoost, Z. Rahemi, Edge-preserving image denoising using a deep convolutional neural network, Signal Processing 159 (2019) 20–32.

[17] Y. You, M. Kaveh, Fourth-order partial differential equations for noise removal, IEEE Transactions on Image Processing 9 (10) (2000) 1723–1730.

[18] O. Scherzer, Denoising with higher order derivatives of bounded variation and an application to parameter estimation, Computing 60 (1) (1998) 1–27.

[19] M. Lysaker, A. Lundervold, X.-C. Tai, Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time, IEEE Transactions on Image Processing 12 (12) (2003) 1579–1590.

[20] W. Hinterberger, O. Scherzer, Variational methods on the space of functions of bounded hessian for convexification and denoising, Computing 76 (1) (2006) 109–133.

[21] M. Bergounioux, L.Piffet, A second-order model for image denoising, Set-Valued and Variational Analysis 18 (3-4) (2010) 277–306.

[22] R. Lai, X.-C. Tai, T. Chan, A ridge and corner preserving model for surface restoration, SIAM Journal on Scientific Computing 35 (2) (2013) A675–A695.

[23] T. Chan, S. Esedoglu, F. Park, A fourth order dual method for staircase reduction in texture extraction and image restoration problems, 2010 17th IEEE International Conference on Image Processing (ICIP), Hongkong, China (2010) 4137–4140.

[24] S. Zheng, Z. Pan, C. Jiang, G. Wang, A new fast algorithm for image denoising, 3rd International Conference on Multimedia Technology (2013) 682–689.

[25] G. Wang, J. Xu, Q. Dong, Z. Pan, Active contour model coupling with higher order diffusion for medical image segmentation, International Journal of Biomedical Imaging 2014 (2014) 1–8.

[26] R. Chan, H. Liang, S. Wei, M. Nikolova, X.-C. Tai, High-order total variation regularization approach for axially symmetric object tomography from a single radiograph, Inverse Problems and Imaging 9 (1) (2015) 55–77.

[27] K. Papafitsoros, C. Schönlieb, A combined first and second order variational approach for image reconstruction, Journal of Mathematical Imaging and Vision 48 (2) (2014) 308–338.

[28] A. Chambolle, P. Lions, Image recovery via total variation minimization and related problems, Numerische Mathematik 76 (2) (1997) 167–188.

[29] K. Bredies, K. Kunisch, T. Pock., Color TV: Total variation methods for restoration of vector-valued images, SIAM J. Imaging Sci. 3 (3) (2010) 492–526.

[30] W. Zhu, T. Chan, Image denoising using mean curvature of image surface, SIAM J. Imaging Sci. 5 (1) (2012) 1–32.

[31] C. Brito-Loeza, K. Chen, V. Uc-Cetina, Image denoising using the gaussian curvature of the image surface, Numer. Meth. Part. Differ. Equ. 32 (3) (2016) 1066–1089.

[32] L. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Physica D 60 (1992) 259–268.

[33] J. Zhang, K. Chen, A total fractional-order variation model for image restoration with nonhomogeneous boundary conditions and its numerical solution, SIAM J. Imaging Sci. 8 (4) (2015) 2487–2518.

[34] W. Lu, J. Duan, Z. Qiu, Z. Pan, R. Lid, L. Bai, Implementation of high-order variational models made easy for image processing, Mathematical Methods in the Applied Sciences 39 (2016) 4208–4233.

[35] C. Brito-Loeza, K. Chen, On fast iterative algorithms for solving the minimisation of curvature-related functionals in surface fairing, International Journal of Computer Mathematics 90 (1) (2013) 92–110.

[36] T.F.Chan, S. Kang, J. Shen, Euler's elastica and curvature based inpaintings, SIAM J. Appl. Math. 63 (2) (2002) 564–592.

[37] C. Brito-Loeza, K. Chen, Multigrid algorithm for high order denoising, SIAM J.Imaging Sci. 3 (3) (2010) 363–389.

[38] T. Goldstein, S. Osher., The split bregman method for l1-regularized problems., SIAM Journal on Imaging Sciences 2 (2) (2009) 323–343.

[39] X.-C. Tai., C. Wu., Augmented Lagrangian method, dual methods and split Bregman iteration for ROF model, Scale Space and Variational Methods in Computer Vision, LNCS 5567 (556) (2009) 502–513.

[40] C. Wu., X.-C. Tai., Augmented lagrangian method, dual methods, and split bregman iteration for rof, vectorial tv, and high order models, SIAM J. Imaging Sci. 3 (3) (2010) 300–339.

[41] Y. Han, X.-C. Feng, G. Baciu, W.-W. Wang., Nonconvex spare regularizer based speckle noise removal, Pattern recognition 46 (2013) 989–1001.

[42] Y. Huang, M. Ng, Y. Wen, A new total variation method for multiplicative noise removal, SIAM J. Imaging Sci. 2 (1) (2009) 20–40.

[43] J.Huang, X.Yang, Fast reduction of speckle noise in real ultrasound images, Signal Processing 93 (2013) 684–694.

[44] L. Xiao, L.-L. Huang, Z.-H. Wei., A weberized total variation regularization-based image multiplicative noise removal algorithm, EURASIP Journal on Advances in Signal Processing 490384 (2010) :DOI:0.1155/2010/490384.

[45] Z.-F. P. B. Shi, L.Huang, Fast algorithm for multiplicative noise removal, J. Vis. Commun. Image R. 23 (2012) 126–133.

[46] P. Sroisangwan, N. Chumchob, A New Numerical Method for Gaussian Curvatuve Based Image Restoration, in Proceedings of Annual Pure and Applied Mathematics Conference 2018, Chulalongkorn University, Bangkok, Thailand, pp. 54-66, 2018.

[47] Z. Wang, A. Bovik, H. Sheikh, E. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612.

Output ที่ได้จากโครงการ

1. ผลงานตีพิมพ์ในวารสารวิชาการนานาชาติ

1.1 **N. Chumchob** and I. Prakit, *An Improved Variational Model and Its Numerical Solutions for Speckle Noise Removal from Real Ultrasound Images*, Journal of Computational Mathematics, Vol.37(2019), pp 201-239. (IF 2017: 1.026).

1.2 P. Sroisangwan and **N. Chumchob**, *A New Curvature-based Model for Image Denoising*, Submitted to Applied Mathematical Modelling. (IF 2017: 2.617)

1.3 .**N. Chumchob**, *An Augmented Lagrangian Method for Solving Total Variation (TV)-based Image Registration Model*, Submitted to Computers and Mathematics with Applications. (IF 2017: 1.860)

1.4 P. Sroisangwan and **N. Chumchob**, *A New Numerical Method of Gaussian Curvature Based Image* Restoration, The Proceedings of Annual Pure and Applied Mathematics Conference 2018, pp. 54-66, May 30-June 1.

1.5 P. Sroisangwan and **N. Chumchob**, *An Improved Variational Model for Image Restoration*, The Proceedings of 24th Annual Meeting in Mathematics 2019. Accepted.

2. การนำผลงานวิจัยไปใช้ประโยชน์

2.1. ใช้ในเชิงวิชาการสำหรับการจัดการเรียนการสอนในรายวิชา Variational techniques and partial differential equations in image sciences และ Numerical analysis ระดับปริญญาศึกษาและระดับบัณฑิตศึกษา

2.2.ใช้ในเชิงวิชาการสำหรับเป็นแนวทางในการทำวิจัยเพื่อสร้างนักวิจัยระดับบัณฑิตศึกษาในอนาคต

Journal of Computational Mathematics
Vol.37, No.2, 2019, 201–239.

http://www.global-sci.org/jcm
doi:10.4208/jcm.1711-m2015-0477

AN IMPROVED VARIATIONAL MODEL AND ITS NUMERICAL SOLUTIONS FOR SPECKLE NOISE REMOVAL FROM REAL ULTRASOUND IMAGES*

Noppadol Chumchob[1)] and Isararat Prakit
Department of Mathematics, Faculty of Science, Silpakorn University, Nakorn Pathom 73000, Thailand and Centre of Excellence in Mathematics, CHE, Si Ayutthaya Rd., Bangkok 10400, Thailand
Email: chumchob_n@silpakorn.edu, prakit_i@silpakorn.edu

Abstract

Ultrasound imaging technique is one of the most non-invasive, practically harmless to the human body, accurate, cost effective and real-time techniques in medical diagnosis. However, ultrasound images suffer from the so-called speckle noise because of the imaging principle. The speckle noise reduces the quality and visibility of ultrasound images, thereby decreasing overall reliability of the images and interfering with the clinical diagnosis. In this paper, we propose a novel variational model under a combination of total variation regularization and Weberized total variation regularization and prove the existence and uniqueness of the minimizer for the variational problem. In order to efficiently solve the associated Euler-Lagrange equation consisting of nonlinear partial differential equation, we apply a finite difference method and develop several numerical techniques for solving the resulting discrete system. Numerical experiments on various synthetic and real ultrasound images not only confirm that our improved model is effective, but also it can provide significant improvement over evaluated models. Moreover, they also show that our proposed multigrid method has great potential applications to medical ultrasound imaging technique in delivering fast, accurate, and visually pleasing restoration results.

Mathematics subject classification: 68U10, 65F10, 65K10.
Key words: Image restoration, Multigrid method, Speckle noise, Ultrasound image, Weber's law.

1. Introduction

Research in the field of medical imaging has produced several different techniques for clinical diagnosis such as ultrasound (US), computed tomography (CT), magnetic resonance imaging (MRI), and so on. Each technique has its own advantages and disadvantages. One may be more appropriate than other depending on syndrome and/or disease severity. In particular, US imaging has been considered as one of the most non-invasive, practically harmless to the human body, portable, accurate, cost effective and real-time techniques for visualizing the human body's internal structures (e.g. soft organs such as liver, kidney, spleen, uterus, and heart) and movements (e.g., blood flowing through vessels and fetal development in pregnant women). These features not only help the clinicians in almost all stages of patient care: from disease detection to treatment guidance and monitoring, but also make the US imaging the most prevalent diagnostic tool in nearly all hospitals around the world.

[1)] Corresponding author

Basically, the US imaging is a coherent imaging system. All US images are obtained by using high-frequency sound waves, inaudible to the human ear. As the sound waves are transmitted through body tissues, they are partially reflected by the boundary between two tissue structures back to the US machine in different ways, depending on the difference in acoustic impedance of the two tissues at the interface. The time-of-flight (TOF) and the energy of the reflected echo are recorded and transformed into video or photographic images. Note that the measurement of TOF determines the gray level of each image pixel, whereas the measurement of the echo energy provides coordinates identification of the analyzed tissues. We also note that US imaging cannot be used to create the images of bones because they are too dense to penetrate. In addition, the intestinal tract and normal lung tissue are not easily identified with this medical imaging technique because air or gas can interfere with the production of the US images.

Due to the coherent nature of the US imaging systems, the quality and visibility of the US images are limited by the noise, which originates both from physical phenomena underlying the image acquisition process and imperfections of the US imaging system. The so-called *speckle noise* (SN) is found commonly in the US images of the soft organs whose underlying structures are too small to be resolved by large wavelengths. As pointed out in [42], SN is a type of the multiplicative noise (MN), which is a random granular appearance that marks small differences in gray levels and obscures small structures. The SN occurs when there is large number of scatterers with random phase within the resolution cell of an US beam. This interfering phenomenon arises when two or more waves traveling to the probe from the scatterers interfere with each other, constructively or destructively, producing bright and dark spots. The pattern of the SN depends on the probe characteristics, such as the transducer (the US device making the sound waves and receiving the echoes) frequency and the distance from the maximum-pressure point to the transducer. In particular, the SN at high (acoustic) frequencies is less granular than at low frequencies while, its size rises as the distance of the probe source increases.

In the literature, various methods have been proposed and studied for SN reduction from US images. These methods include adaptive filtering [30, 37, 41, 42], wavelet techniques [1, 29, 38], anisotropic diffusion methods [17, 62], and variational methods [2, 4, 32, 34–36, 45, 50]. Among these SN removal techniques, the variational methods are well-established mathematical theory to offer superior image restoration quality. However, much improvement on computational complexity is a major challenge to develop fast, accurate, and stable numerical algorithms for solving associated variational problems.

In the next section, we present a mathematical framework and briefly introduce a variational formulation of US image denoising problems.

1.1. Mathematical framework

Let $z : \Omega \subset \mathbb{R}^2 \to V \subset \mathbb{R}$ be an observed noisy image, where Ω is a rectangle of $\mathbb{R}^2$. The goal of image denoising problems is to restore (or recover) the original image $u : \Omega \subset \mathbb{R}^2 \to V \subset \mathbb{R}$ from the noisy image z. According to the maximum likelihood principle [25], most image denoising problems involve solving the following variational problem:

$$\min_{u \in \mathcal{U}} \{\mathcal{J}_\alpha(u) = \mathcal{D}(u, z) + \alpha \mathcal{R}(u)\}, \tag{1.1}$$

where $\mathcal{D}(u, z)$ is the fidelity or data fitting term deriving from the assumption on the distribution of the noise in the observed noisy image, which is used to penalize the inconsistency between

the denoised image to be restored and the observed noisy image, $\mathcal{R}(u)$ is the regularization term, which is used to filter out the noise from the observed noisy image as well as to preserve significant features such as edges and textures of the restored image, $\alpha > 0$ is the regularization parameter, which compromises the fidelity term $\mathcal{D}(u, z)$ and the regularization term $\mathcal{R}(u)$, and $\mathcal{U}$ is a set of admissible functions, which minimizes the energy functional $\mathcal{J}_\alpha$. For example, the classical model by Rudin, Osher, and Fatemi (ROF model) [44] considers the additive noise (AN) model

$$z = u + \eta \tag{1.2}$$

and minimizes the total variation (TV) regularization $\mathcal{R}^{\mathrm{TV}}(u) = \int_\Omega |\nabla u| d\Omega$ with the fidelity term $\mathcal{D}^{\mathrm{ROF}}(u, z) = \frac{1}{2}\int_\Omega (u-z)^2 d\Omega$. Here η in (1.2) is the zero-mean Gaussian white noise. Note that the remarkable advantages of using the TV regularization for image denoising problems are to allow discontinuities and preserve edges in the restored images.

1.2. A review of variational models for SN removal

The SN model considered in many previous works is given by a multiplicative form [31,34,36]

$$z = u\zeta, \tag{1.3}$$

where ζ is the MN. Unlike traditionally additive Gaussian noise model in (1.2), the noisy signals in the recorded US images are much more difficult to be removed, mainly not only because of the multiplicative nature between the noise and the original image, which is signal correlated, but also because of the noise distribution, which is generally complicated than that of Gaussian noise, commonly assumed to be Rayleigh distribution for the recorded US images.

Non-surprisingly, the choice of $\mathcal{R}(u)$ is very crucial for restoration results. Different choices of $\mathcal{R}(u)$ lead to different restored images. In the literature, $\mathcal{R}^{\mathrm{TV}}(u)$ has is commonly used as the regularization term in several variational models for SN removal, and the main difference among different variational models only comes from their image fidelity terms. As far as we know, Rudin, Lions, and Osher [45] proposed firstly the TV based variational model (RLO model) which contains a nonconvex fidelity term

$$\mathcal{D}^{\mathrm{RLO}}(u, z) = \alpha_1 \int_\Omega \frac{z}{u} d\Omega + \alpha_2 \int_\Omega \left(\frac{z}{u} - 1\right)^2 d\Omega$$

derived from considering ζ to be Gaussian white noise with mean one and a very small variance, where α_1 and α_2 are the weighted parameters. This model is effective to remove the SN to a certain degree. By using the Gamma-noise assumption on ζ, Aubert and Aujol [2] proposed a SN removal method (AA model) with the nonconvex fidelity term

$$\mathcal{D}^{\mathrm{AA}}(u, z) = \int_\Omega \left(\log u + \frac{z}{u}\right) d\Omega$$

derived from the maximum a posteriori probability (MAP) estimation. Due to the nonconvexity of their fidelity terms, the corresponding algorithms may converge slowly and the computed solutions by some optimization methods are not necessary to be a global solution. To overcome these drawbacks, Shi and Osher [50] applied the logarithmic transformation with the fidelity term in the AA model to present their convex SN removal method (SO model) and developed a nonlinear inverse scale space method for their variational technique. Afterwards, Huang et al. [32] modified differently the fidelity term in the AA model by using an exponential

transformation and proposed an alternative minimization technique to solve their (strictly) convex SN removal model (HNW model). Bioucas-Dias and Figueiredo [4] applied the MAP estimation method in the logarithm domain to propose a convex SN removal method (BF model). The advantage of their BF model is that it can be efficiently solved by a fast iteration algorithm.

In clinical US imaging system, a nonlinear compression algorithm is usually applied before displaying in order to adjust the large echo dynamic range [54]. This nonlinear compression totally changes the gray level statistics of the displayed (real) US "on screen". The authors in [6,24,29,41,54,56] pointed out that the SN in the real US images can be modeled as corrupted with the signal-dependent MN of the form

$$z = u + \sqrt{u}\zeta, \tag{1.4}$$

where ζ is the zero-mean Gaussian noise with the standard deviation σ_n^2. Based on the SN model for the real US images as represented by (1.4), the authors in [34, 36, 40] also proposed the SN removal methods including the TV regularization for SN reduction from the real US images. The authors in [40] presented a new SN removal technique (KKWV model) using the convex fidelity term

$$\mathcal{D}^{\mathrm{KKWV}}(u,z) = \int_\Omega \frac{(u-z)^2}{u} d\Omega$$

resulting from (1.4) and developed an anisotropic diffusion method specially designed preserving and enhancing small vessel structures with a constrained filtering. Afterwards, a rigorous analysis of the KKWV model was discussed in [36]. The authors of this previous work first investigated into the existence and uniqueness of the minimizer for the associated variational problem and derived the existence and uniqueness of the resulting evolution equation. Recently, the authors in [34] proposed a new convex fidelity term using a generalized Kullback-Leibler (KL) distance as given by

$$\mathcal{D}^{\mathrm{HY}}(u,z) = \int_\Omega \left(\frac{z}{\sqrt{u}} \log \frac{z}{u} - \frac{z}{\sqrt{u}} + \sqrt{u} \right) d\Omega$$

and applied the variable splitting and Bregman iterative methods developed by [43] in the logarithm domain to efficiently solve their proposed convex SN removal model (HY model). Their numerical tests show that the HY model marginally improves over the KKWV model in terms of the restoration quality, whereas the numerical algorithm of HY model is much faster than that of KKWV model in delivering the same level of restoration quality.

1.3. A review of numerical techniques for SN removal

Classified by the order of its major ingredients, there are two main types of numerical schemes to compute a numerical solution of the minimization problem (1.1) for a given regularization parameter α. The first is the so-called *optimize-discretize* approach and the second is the so-called *discretize-optimize* approach. On one hand the main idea of the first approach is to compute the associated Euler-Lagrange equation in the continuous domain and then solve its discretized version on the corresponding discrete domain by a method of our choices, e.g. a so-called parabolic and elliptic approach. On the other hand the latter approach aims to discretize the energy functional $\mathcal{J}_\alpha$ and then solve the discrete minimization problem by standard optimization techniques, e.g. steepest descent or Newton-type methods.

1.3.1. The optimize-discretize approach

For the optimize-discretize approach, the main aim is to solve the associated Euler-Lagrange (EL) equation, which generally turns out to be a nonlinear PDE

$$f(u,z)+\alpha\mathcal{A}(u)=0 \tag{1.5}$$

subject to the appropriate boundary conditions. In other words, the approach aims to satisfy the first-order necessary condition for being a local minimizer of the energy functional (1.1). Note that on one hand the first term $f(u,z)$ (usually nonlinear) is related to the Gâteaux derivative of the fidelity term $\mathcal{D}$. On the other hand, the second term $\mathcal{A}$ is the partial differential operator (linear or nonlinear) resulting from the Gâteaux derivative of the regularization term $\mathcal{R}$.

There are various numerical techniques for solving the nonlinear PDE in (1.5). These techniques can be broadly divided into two main categories: the *parabolic* and *elliptic* approaches. A parabolic approach (also known as gradient descent or time marching approach) performs by introducing the artificial time variable t and then determining the steady state solution of the nonlinear time-dependent PDE:

$$\partial_t u(t)=f(u(t),z)+\alpha\mathcal{A}(u(t)), \tag{1.6}$$

where $u(t)=u(\mathbf{x};t)$, typically $u(0)=z$. For example, if f is nonlinear and $\mathcal{A}$ is linear, the semi-implicit scheme for (1.6) can be given by

$$\frac{u(t^{(k+1)})-u(t^{(k)})}{\tau}=f(u(t^{(k)}),z)+\alpha\mathcal{A}(u(t^{(k+1)})) \tag{1.7}$$

where $k\in\mathbb{N}_0$ and $\tau>0$ denotes the time length used to discretize $\partial_t\boldsymbol{u}(t)$; see [2, 4, 44, 45, 50]. For an elliptic approach it performs by directly solving the nonlinear PDE (1.5) with a method of our choice. For example, if both f and $\mathcal{A}$ are nonlinear, the fixed-point (FP) iteration of (1.5) can be defined by

$$f(u^{[\nu]},z)+\alpha\mathcal{A}[u^{[\nu]}](u^{[\nu+1]})=0 \tag{1.8}$$

where both f and $\mathcal{A}$ are globally linearized at the current approximation $u^{[\nu]}$ and $\nu\in\mathbb{N}_0$ denotes the FP step; see [10, 11, 46–48].

1.3.2. The discretize-optimize approach

In this section, we shall briefly give the main idea of the discretize-optimize approach based in the Newton-type schemes. To this end, let us first consider the discrete minimization problem corresponding to (1.1)

$$\min_{u\in\mathcal{U}}\{\widetilde{\mathcal{J}}_\alpha(u)=\widetilde{\mathcal{D}}\,(u,z)+\alpha\widetilde{\mathcal{R}}\,(u)\}. \tag{1.9}$$

The next step is to linearize the discrete energy functional $\widetilde{\mathcal{J}}_\alpha$ around the current approximation $u^{(k)}$ $(k\in\mathbb{N}_0)$ by the Taylor expansion given by

$$\widetilde{\mathcal{J}}_\alpha(u^{(k)}+\delta u^{(k)})=\widetilde{\mathcal{J}}_\alpha(u^{(k)})+\mathbf{J}^{(k)}_{\widetilde{\mathcal{J}}_\alpha}\delta u^{(k)}+\frac{1}{2}(\delta u^{(k)})^\top\mathbf{H}^{(k)}_{\widetilde{\mathcal{J}}_\alpha}\delta u^{(k)} \tag{1.10}$$

and define an (outer) iteration by

$$u^{(k+1)}=u^{(k)}+\zeta^{(k)}\delta u^{(k)}. \tag{1.11}$$

Here $\mathbf{J}^{(k)}_{\widetilde{\mathcal{J}}_\alpha}$, $\mathbf{H}^{(k)}_{\widetilde{\mathcal{J}}_\alpha}$ are the Jacobian and the Hessian of $\widetilde{\mathcal{J}}_\alpha$ at $u^{(k)}$, and $\zeta^{(k)} > 0$ is the line-search parameter used to guarantee the reduction of $\widetilde{\mathcal{J}}_\alpha$ in each outer step k. For Newton-type methods, the perturbation $\delta u^{(k)}$ can be determined by

$$\delta u^{(k)} = -\mathbf{B}^{(k)}_{\widetilde{\mathcal{J}}_\alpha}\mathbf{J}^{(k)}_{\widetilde{\mathcal{J}}_\alpha} \tag{1.12}$$

where $\mathbf{B}^{(k)}_{\widetilde{\mathcal{J}}_\alpha}$ is an approximation of the inverse Hessian by a method of our choices, e.g., DFP and BFGS methods.

This paper prefers the first approach. However, no matter which method is used in practical applications, both approaches should be integrated with a so-called *multilevel* technique in order to provide reliable initial guesses, avoid getting in the trap of unwanted minimizers and save computation times [10–14, 28, 46–48].

1.4. Contributions

The main aim of this paper is to propose an improved variational model and its fast solution for the numerical approximation in removing the SN from real US images. The improved model includes the combined regularization term for removing the SN. In order to efficiently solve the associated EL equation, we apply the optimize-discretize approach implemented in a nonlinear multigrid (NMG) framework as a fast and effective solver. We note that the variational model and its numerical solutions to be proposed in this work are the improvement of those done in the previous works by [34, 40, 49] with several advantages. Firstly, the proposed model is more appropriate than those in [49] for restoring real US images corrupted by the signal-dependent multiplicative noise as represented in (1.4). Secondly, the proposed model can also be considered as the generalization of the variational model in the previous work done by [40]. Next, we found what we expected in numerical experiments that our variational model provides better restoration results than those obtained from [34, 40]. Finally, as we shall demonstrate, our proposed NMG solution to be introduced in Section 2 is fast and efficient in providing visually pleasing SN reduction from both synthetic and real US images.

This work is organized as follows. In Section 2, we propose the improved variational model, followed by its numerical solutions for the associated EL equation. In Section 3, experimental results from synthetic and real US images illustrating the effectiveness of the improved model and the efficiency of the proposed numerical methods are shown. Section 4 contains the conclusion of this work.

2. The Improved Variational Model and Its Numerical Solutions

2.1. The improved variational model

As pointed out in [49], all images are eventually to be perceived and interpreted by the human visual system. Therefore, any ideals in image processing should take into account the effects of vision psychology and psychophysics, such as that of Weber's law. The law reveals the universal influence of the background stimulus u on humans sensitivity to the intensity increment δu, or the so-called JND (just-noticeable-difference), in the perception of both sound and light. It claims that the so-called Weber's fraction is a constant, i.e.

$$\frac{\delta u}{u} = \text{cont.} \tag{2.1}$$

In the context of visual perception, we note that u stands for the background light intensity and δu the intensity fluctuation, respectively.

To the best of our knowledge, most conventional SN removal models do not take into account that our visual sensitivity to the local fluctuation δu depends on the ambient intensity level u. That is, models such as the KKWV model given by

$$\min_{u\in S(\Omega)} \left\{ \mathcal{J}_{\alpha}^{\mathrm{KKWV}}(u) = \mathcal{D}^{\mathrm{KKWV}}(u,z) + \alpha \mathcal{R}^{\mathrm{TV}}(u) \right\}, \tag{2.2}$$

assumes that a local variation, $\delta u = 0.02$ say, should be treated equally independent of the background intensity level u, no matter whether it is $u = 0.1$ or 0.8. But this exactly violates Weber's law, according to which, a fluctuation level of $\delta u = 0.02$ against a background intensity $u = 0.1$ is much more significant than the same amount against $u = 0.8$. In fact it is approximately equivalent to a level of $\delta u = \frac{0.8}{0.1} \times 0.02 = 0.16$ in the latter situation, a noticeable 60% increment. This motivates us to integrate the effects of vision phychology and psychophysics of human's visual sensitivity for improving the qualiy of the restored US images produced by the original KKWV model. Note that $S(\Omega) = \{u \in BV(\Omega), u > 0\}$ and $BV(\Omega)$ denotes the bounded variation space, which is the space of functions $u \in L^1(\Omega)$ such that the following quantity

$$\int_{\Omega} |Du| = \sup \left\{ \int_{\Omega} u \mathrm{div}(\varphi) d\Omega | \varphi \in C_0^1(\Omega; \mathbb{R}^n), |\varphi| \leq 1 \right\}$$

is finite, where Du represents the distributional gradient of u. As can be seen $BV(\Omega)$ is a Banach space with the norm $\|u\|_{BV(\Omega)} = \int_{\Omega} |Du| + \|u\|_{L^1(\Omega)}$.

In order to improve KKWV model by the Weberized local variation $\frac{|\nabla u|}{u}$ which encodes the influence of the background intensity u according to Weber's law (2.1), we add the WTV regularization

$$\mathcal{R}^{\mathrm{WTV}}(u) = \int_{\Omega} \frac{|\nabla u|}{u} d\Omega$$

into the regularization term of the KKWV model as follows:

$$\mathcal{R}^{\text{TV-WTV}}(u) = \alpha_1 \int_{\Omega} |\nabla u| d\Omega + \alpha_2 \int_{\Omega} \frac{|\nabla u|}{u} d\Omega, \tag{2.3}$$

where α_1 and α_2 are the regularization parameters. Therefore, our improved KKWV model is given by

$$\min_{u\in \bar{S}(\Omega)} \left\{ \bar{\mathcal{J}}_{\alpha_1,\alpha_2}^{\mathrm{KKWV}}(u) = \mathcal{D}^{\mathrm{KKWV}}(u,z) + \mathcal{R}^{\text{TV-WTV}}(u) \right\}, \tag{2.4}$$

where

$$\bar{S}(\Omega) = \left\{ u \in BV(\Omega), u > 0, \int_{\Omega} \frac{|Du|}{u} < \infty \right\}$$

denotes the space of admissible functions minimizing the energy functional $\bar{\mathcal{J}}_{\alpha}^{\mathrm{KKWV}}$ in (2.4), resulting from the coarea formula; see [27].

We bridge the connection between our improved model $\bar{\mathcal{J}}_{\alpha_1,\alpha_2}^{\mathrm{KKWV}}$ in (2.4) and the original KKWV model $\mathcal{J}_{\alpha}^{\mathrm{KKWV}}$ in (2.3) as follows. First, if $\phi(u) = \alpha_1 + \alpha_2/u$, then (2.3) can be re-written as

$$\mathcal{R}^{\text{TV-WTV}}(u) = \int_{\Omega} \phi(u)|Du| = \alpha_1 \int_{\Omega} |Du| + \alpha_2 \int_{\Omega} \frac{|Du|}{u} = \int_{\Omega} \left(\alpha_1 + \frac{\alpha_2}{u} \right) |Du|. \tag{2.5}$$

which is a combination of $\mathcal{R}^{\mathrm{TV}}$ and $\mathcal{R}^{\mathrm{WTV}}$. This means that $\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}$ gains the benefits of $\mathcal{R}^{\mathrm{TV}}$ and $\mathcal{R}^{\mathrm{WTV}}$ and should have better performance than $\mathcal{J}^{\mathrm{KKWV}}_{\alpha}$(involving only $\mathcal{R}^{\mathrm{TV}}$) in removing SN for real US images. Note that on one hand $\mathcal{R}^{\mathrm{TV}}$ is effective to reduce SN without losing important features (i.e. edges and contrast), but on the other hand $\mathcal{R}^{\mathrm{WTV}}$ is able to distribute the minimum amount of irregularity adaptively over the image domain according Weber's law. Therefore, in the restored real US image, the minimum fluctuation δu is allows to be larger on regions where the background intensity u is higher and human's visual sensitivity is weaker. Second, our improved KKWV model $\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}$ clearly reduces to the original KKWV model $\mathcal{J}^{\mathrm{KKWV}}_{\alpha}$ when $\alpha_2 = 0$.

We note that Total generalized variation (TGV) regularization by [5] is also a potential tool in removing SN from the real US images. It involves higher-order derivatives of u to deal with several undesirable side effects, one of which is the staircase effect. However, it does not take into account the consequences of vision psychology and psychophysics to produce good denoised US images.

2.2. Mathematical analysis for the improved model

This section aims to investigate the existence and uniqueness of the minimizer to problem (2.4).

Theorem 2.1 (Existence). *Suppose that $z \in L^\infty(\Omega)$ with $\inf\limits_{\Omega} z > 0$. Then the variational problem* (2.4) *has at least one minimizer $\bar{u}$ in the admissible space $\bar{S}(\Omega)$ satisfying*

$$\inf_{\Omega} \ z \le \bar{u} \le \sup_{\Omega} \ z.$$

Proof. Let us denote that

$$\gamma = \inf_{\Omega} z, \quad \beta = \sup_{\Omega} z, \quad h(s) = \frac{(s-z)^2}{s}.$$

It is obvious that $\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u) \ge 0$ for all $u \in \bar{S}(\Omega)$. This implies that $\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u)$ has a lower bound for all $u \in \bar{S}(\Omega)$. Therefore, we consider a minimizing sequence $\{u_n\} \subset \bar{S}(\Omega)$ for (2.4).

First, we show that $\gamma \le u_n \le \beta$. Since $z \in L^\infty(\Omega)$ with $\inf\limits_{\Omega} z > 0$, we can choose a sequence $\{z_n\} \subset C^\infty(\bar{\Omega})$ such that $z_n \to z$ in $L^1(\Omega)$ and a.e. in Ω as $n \to \infty$, and

$$\inf_{\Omega} \ z \le z_n \le \sup_{\Omega} \ z.$$

Replacing z in (2.4) by z_n yields $h(s)$ is decreasing as $s \in (0, z_n)$ and increasing as $s \in (z_n, \infty)$ for $n \in \mathbb{N}$. Therefore, if $A \ge z_n$, we have

$$\frac{(\min(s,A)-z)^2}{\min(s,A)} \le \frac{(s-z)^2}{s} \tag{2.6}$$

for $\mathbf{x} \in \Omega$ and $n \in \mathbb{N}$. Hence, if we let $A = \beta$, we have

$$\int_\Omega \frac{\left(\min(u,\beta)-z_n\right)^2}{\min(u,\beta)} d\Omega \le \int_\Omega \frac{\left(u-z_n\right)^2}{u} d\Omega.$$

Letting $n \to \infty$ in the above inequality and using Lebesgue Convergence Theorem and (2.6) yield

$$\int_\Omega \frac{\left(\min(u,\beta)-z\right)^2}{\min(u,\beta)} d\Omega \le \int_\Omega \frac{\left(u-z\right)^2}{u} d\Omega. \tag{2.7}$$

By using the results of [39] (see Lemma 1 in Section 4.3) and [49] (see Lemma 1 Section 3.2), we obtain

$$\int_\Omega |D(\min(u,\beta))| \le \int_\Omega |Du|, \tag{2.8}$$

$$\int_\Omega \frac{|D(\min(u,\beta))|}{\min(u,\beta)} \le \int_\Omega \frac{|Du|}{u}, \tag{2.9}$$

respectively. Combining (2.7)–(2.9) implies that

$$\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(\min(u,\beta)) \le \bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u).$$

On the other hand, we get in the same way that

$$\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(\max(u,\beta)) \le \bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u).$$

Therefore we can assume without restriction that $\gamma \le u_n \le \beta$.

Next, we prove that there exists $u \in \bar{S}(\Omega)$ such that

$$\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u) = \min_{v\in\bar{S}(\Omega)} \left\{\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(v)\right\}.$$

Without loss of generality, we assume that $\alpha_1 = \alpha_2 = 1$. As can be seen, the above proof implies that u_n is bounded in $L^1(\Omega)$. Moreover, by the definition of $\{u_n\}$, we get that there exists a constant C such that

$$\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u_n) \le C. \tag{2.10}$$

Since $\gamma \le u_n \le \beta$ and $h \in C[\gamma,\beta]$, we get that $h(u_n)$ is bounded. Therefore, by using (2.10), we obtain

$$\int_\Omega |Du_n| \le C.$$

Hence, $\{u_n\}$ is bounded in $BV(\Omega)$. By the weak compactness, $\{u_n\}$ has a subsequence, still denoted by $\{u_n\}$ for simplicity, that converges strongly in $L^1(\Omega)$ to some $\bar{u}$, i.e. $u_n \to \bar{u}$. After a refinement of the subsequence if necessary, we can assume that

$$u_n(x) \to \bar{u}(x), \text{a.e. } \mathbf{x} \in \Omega.$$

Thus by Lebesgue's dominated convergence theorem, we obtain

$$\int_\Omega \frac{(\bar{u}-z)^2}{\bar{u}} d\Omega = \lim_{n\to\infty} \int_\Omega \frac{(u_n-z)^2}{u_n} d\Omega. \tag{2.11}$$

Applying the lower semicontinuity of the total variation term and Fatou's lemma leads to

$$\int_\Omega |D\bar{u}| \le \liminf_{n\to\infty} \int_\Omega |Du_n|. \tag{2.12}$$

The lower semicontinuity of the Weberized total variation term can be obtained by [49] (Theorem 1 in Section 3.2):

$$\int_\Omega |D\bar{v}| \le \liminf_{n\to\infty} \int_\Omega |Dv_n|, \tag{2.13}$$

where $v_n = \log u_n$ and $\bar{v} = \log \bar{u}$.

Combining (2.11), (2.12), and (2.13), we have

$$\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(\bar{u}) \leq \liminf_{n\to\infty} \bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u_n).$$

It is obvious to see that $\bar{u} \in \bar{S}(\Omega)$. Due to $\{u_n\}$ is the minimizing sequence, we therefore have shown that $\bar{u}$ is in fact a minimizer of the problem (2.4). □

Unlike the several variational models discussed in Section 1, our improved KKWV model (2.4) is not convex due to the Weberized TV regularization term. As a result, uniqueness is no longer a direct product of convexity. To this end, we start with a computational lemma for the associated EL equation of the variational problem (2.4). Recall that $\phi(u) = \alpha_1 + \alpha_2/u$ in $\mathcal{R}^{\text{TV-WTV}}$.

Lemma 2.1. *Let $\phi(u) : (0,\infty) \to (0,\infty)$ be a C^1 function and*

$$\bar{\mathcal{J}}^{KKWV}_{\alpha_1,\alpha_2}(u) = \mathcal{D}^{KKWV}(u,z) + \mathcal{R}^{TV\text{-}WTV}(u)$$

then the formal equilibrium EL equation of $\bar{\mathcal{J}}^{KKWV}_{\alpha_1,\alpha_2}(u)$ is given by

$$-\phi(u)\nabla\cdot\left(\frac{\nabla u}{|\nabla u|}\right) + \left(1 - \frac{z^2}{u^2}\right) = 0, \tag{2.14a}$$

$$\left.\frac{\partial u}{\partial \boldsymbol{n}}\right|_{\partial\Omega} = 0. \tag{2.14b}$$

where $\boldsymbol{n}$ is the unit outward normal vector on the image boundary $\partial\Omega$.

Proof. Applying the standard computation of Calculus of Variation

$$\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2} \to \bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2} + \delta\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}$$

leads to

$$\begin{aligned}
\delta\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2} &= \int_\Omega \left(\phi'(u)|\nabla u|\delta u + \phi(u)\frac{\nabla u}{|\nabla u|}\nabla(\delta u)\right) d\Omega \int_\Omega \left(1 - \frac{z^2}{u^2}\right)\delta u d\Omega,\\
&= \int_\Omega \left(\phi'(u)|\nabla u| - \nabla\cdot\left(\phi(u)\frac{\nabla u}{|\nabla u|}\right)\delta u\right) d\Omega\\
&\qquad + \int_{\partial\Omega} \frac{\phi(u)}{|\nabla u|}\frac{\partial u}{\partial \boldsymbol{n}}\delta u dS + \int_\Omega \left(1 - \frac{z^2}{u^2}\right)\delta u d\Omega,\\
&= \int_\Omega \left(-\phi(u)\nabla\cdot\left(\frac{\nabla u}{|\nabla u|}\right)\right)\delta u d\Omega\\
&\qquad + \int_{\partial\Omega} \frac{\phi(u)}{|\nabla u|}\frac{\partial u}{\partial \boldsymbol{n}}\delta u dS + \int_\Omega \left(1 - \frac{z^2}{u^2}\right)\delta u d\Omega,
\end{aligned}$$

where dS denotes the arc-length element of the boundary. Since $u > 0$, the EL equation for $\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u)$ can be re-written equivalently as (2.14). □

Theorem 2.2 (Uniqueness). *Assume that $\alpha_1 > 0$, $\alpha_2 > 0$ and $z > 0$ is in $L^\infty(\Omega)$, and u is a minimizer of the energy functional $\bar{\mathcal{J}}^{KKWV}_{\alpha_1,\alpha_2}(u)$. Then u is a unique solution in $\bar{S}(\Omega)$.*

Proof. Applying ideas in [49,61], let us denote that

$$F'(u;\alpha_1,\alpha_2)=\frac{1}{\phi(u)}\left(1-\frac{z^2}{u^2}\right)=\frac{u^2-z^2}{\alpha_1u^2+\alpha_2u}.$$

Define a new energy function as follow:

$$\bar{\bar{\mathcal{J}}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u)=\int_\Omega\left(|\nabla u|+F(u;\alpha_1,\alpha_2)\right)d\Omega. \tag{2.15}$$

As can be seen, (2.14a) is exactly the associated EL equation for $\bar{\bar{\mathcal{J}}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u)$. It is easy to find that

$$F''(u;\alpha_1,\alpha_2)=\frac{\alpha_2u^2+2\alpha_1uz^2+\alpha_2z^2}{\left(\alpha_1u^2+\alpha_2u\right)^2}$$

is strictly convex as $u>0$. Therefore, the uniqueness of the minimizer follow from the strict convexity of the energy functional $\bar{\bar{\mathcal{J}}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u)$. □

2.3. Numerical solutions of the EL equation

In this section, we introduce possible numerical solutions that could be considered for solving the EL equation (2.14a). To start with, we note that applying Lemma 2.1 with $\phi(u)=\alpha_1+\alpha_2/u$ yields

$$\underbrace{-\mathcal{K}(u)+\frac{1}{\alpha_1u+\alpha_2}\left(u-\frac{z^2}{u}\right)}_{\mathcal{N}(u)}=g, \tag{2.16}$$

where $\mathcal{K}(u)=\nabla\cdot(\frac{\nabla u}{|\nabla u|_\epsilon})$, $|\nabla u|_\epsilon=\sqrt{|\nabla u|^2+\epsilon}$, and $0<\epsilon\ll 1$ is a small constant to avoid division by zero. Here $g=0$ on the finest grid for the multigrid setting.

2.3.1. Finite difference discretization

Let $\left(u^h\right)_{i,j}=u^h\left(x_{1_i},x_{2_j}\right)$ be the grid function with the uniform grid spacing $h=1/n$. Here the integer $n=1/h$ is the number of uniform intervals in the x_1 and x_2 coordinate directions. Each grid point $\mathbf{x}$ in the discretized domain Ω^h is cell-centered and given by

$$\mathbf{x}=\left(x_{1_i},x_{2_j}\right)^\top=\left((2i-1)\,h/2,(2j-1)\,h/2\right)^\top$$

for $1\le i,j\le n$. The partial derivatives in (2.16) are approximated by the standard second-order finite difference schemes. Therefore, the discrete nonlinear system is given by

$$\underbrace{-\mathcal{K}^h(u^h)_{i,j}+\alpha^h_\star(u^h)_{i,j}\left((u^h)_{i,j}-\frac{(z^h)^2_{i,j}}{(u^h)_{i,j}}\right)}_{\mathcal{N}^h(u^h)_{i,j}}=(g^h)_{i,j}, \tag{2.17}$$

where

$$\mathcal{K}^h(u^h)_{i,j} = -\frac{1}{h^2}\Big((\Sigma^h)_{i,j}(u^h)_{i,j} - (\bar{\Sigma}^h)_{i,j}(u^h)_{i,j}\Big),$$

$$(\Sigma^h)_{i,j}(u^h)_{i,j} = \Big(D_1(u^h)_{i,j} + D_2(u^h)_{i,j} + 2D_3(u^h)_{i,j}\Big)(u^h)_{i,j},$$

$$(\bar{\Sigma}^h)_{i,j}(u^h)_{i,j} = D_1(u^h)_{i,j}(u^h)_{i-1,j} + D_2(u^h)_{i,j}(u^h)_{i,j-1} + D_3(u^h)_{i,j}(u^h)_{i+1,j} + D_3(u^h)_{i,j}(u^h)_{i,j+1},$$

$$D_1(u^h)_{i,j} = D(u^h)_{i-1,j}, D_2(u^h)_{i,j} = D(u^h)_{i,j-1}, D_3(u^h)_{i,j} = D(u^h)_{i,j},$$

$$D(u^h)_{i,j} = \frac{1}{\sqrt{(\delta_x^+(u^h)_{i,j})^2 + (\delta_y^+(u^h)_{i,j})^2 + \epsilon}},$$

$$\delta_{x_1}^+(u^h)_{i,j} = ((u^h)_{i+1,j} - (u^h)_{i,j})/h, \qquad \delta_{x_2}^+(u^h)_{i,j} = ((u^h)_{i,j+1} - (u^h)_{i,j})/h,$$

$$\delta_{x_1}^-(u^h)_{i,j} = ((u^h)_{i,j} - (u^h)_{i-1,j})/h, \qquad \delta_{x_2}^-(u^h)_{i,j} = ((u^h)_{i,j} - (u^h)_{i,j-1})/h,$$

$$\alpha_\star^h(u^h)_{i,j} = \frac{1}{\alpha_1(u^h)_{i,j} + \alpha_2}.$$

We note that the approximations in (2.17) need to be adjusted at the image boundary $\partial\Omega_h$ using the discrete boundary conditions

$$(u^h)_{i,1} = (u^h)_{i,2}, \quad (u^h)_{i,n} = (u^h)_{i,n-1}, \quad (u^h)_{1,j} = (u^h)_{2,j}, \quad (u^h)_{n,j} = (u^h)_{n-1,j}.$$

In the following subsections the symbol 'h' will sometimes drop for simplicity.

2.3.2. Method 1–Fixed-point (FP) method

A FP method is a possible option in solving the EL equations related to the TV minimization; see e.g. [3,8–11,15,19–23,31,46–48,58]. Before introducing our three proposed FP methods, we will briefly explain the semi-implicit time marching (SITM) method.

As discussed in Section 1.3.1, the main idea of time marching approaches is to introduce an artificial time variable t with (2.16) and drive it to a steady state. In order to overcome the nonlinear terms, we may linearize the associated equation respect to the $(k+1)$th time step using the method of frozen coefficients as well known for variational approaches related to the TV operator (see e.g. [46–48,58], and obtain the semi-implicit scheme as follows:

$$(u^{(k+1)})_{i,j} - \tau\bar{\mathcal{K}}(u^{(k+1)})_{i,j} = (u^{(k)})_{i,j} - \tau\alpha_\star(u^{(k)})_{i,j}\left((u^{(k)})_{i,j} - \frac{(z)_{i,j}^2}{(u^{(k)})_{i,j}}\right), \tag{2.18}$$

where

$$\bar{\mathcal{K}}(u^{(k+1)})_{i,j} = -(1/h^2)\left((\Sigma^{(k)})_{i,j}(u^{(k+1)})_{i,j} - (\bar{\Sigma}^{(k)})_{i,j}(u^{(k+1)})_{i,j}\right),$$

$$(\Sigma^{(k)})_{i,j}(u^{(k+1)})_{i,j} = \left(D_1(u^{(k)})_{i,j} + D_2(u^{(k)})_{i,j} + 2D_3(u^{(k)})_{i,j}\right)(u^{(k+1)})_{i,j},$$

$$(\bar{\Sigma}^{(k)})_{i,j}(u^{(k+1)})_{i,j} = D_1(u^{(k)})_{i,j}(u^{(k+1)})_{i-1,j} + D_2(u^{(k)})_{i,j}(u^{(k+1)})_{i,j-1}$$
$$+ D_3(u^{(k)})_{i,j}(u^{(k+1)})_{i+1,j} + D_3(u^{(k)})_{i,j}(u^{(k+1)})_{i,j+1},$$

$(u^{(k)})_{i,j} = u(x_{1_i}, x_{2_j}, t_k)$, $\alpha_\star(u^{(k)})_{i,j} = \frac{1}{\alpha_1(u^{(k)})_{i,j}+\alpha_2}$ and $\tau > 0$.

Although the above SITM method is simple for numerical implementation, this method is slow due to strict stability constraints in the time step. Usually $\tau \approx \mathcal{O}(h^2)$ for the discretize PDEs like (2.18), which in our case implies that standard time marching methods are practically of no use for processing large images.

Global fixed-point (GFP) method. For the first FP method, the nonlinear terms $1/|\nabla u|_\epsilon$, $\alpha_\star(u)$ and $1/u$ represented in (2.16) or $D\,(u)_{i,j}$, $\alpha_\star(u)_{i,j}$ and $1/(u)_{i,j}$ in (2.17) may be linearized or frozen globally at a previous FP step ν. This yields the resulting linearized system

$$\mathbf{N}[u^{[\nu]}]u^{[\nu+1]} = \mathbf{G}[u^{[\nu]}], \quad \nu = 0, 1, 2, ... \tag{2.19}$$

where

$$\mathbf{N}[u^{[\nu]}] = -\nabla \cdot (\tfrac{\nabla}{|\nabla u^{[\nu]}|_\epsilon}) + \alpha_\star(u^{[\nu]}) \text{ and } \mathbf{G}[u^{[\nu]}] = g + \frac{\alpha_\star(u^{[\nu]})z^2}{u^{[\nu]}},$$

typically $u^{[0]} = z$. Classified by its ingredients, we shall name this FP method the global FP (GFP) method.

As a common way to solve (2.19) for each GFP or outer step ν, we use the successive over-relaxation (SOR) method with the relaxation parameter $\omega \in (0, 2)$ as the linear solver and then the new step at a grid point (x_{1_i}, x_{2_j}) is given by

$$(u^{[\nu+1,\bar{k}+1]})_{i,j} = (1-\omega)\, u^{[\nu+1,\bar{k}]})_{i,j} + \omega(\mathbf{N}[u^{[\nu]}])^{-1}_{i,j}(\mathbf{G}[u^{[\nu,\bar{k}+1/2]}])_{i,j}, \tag{2.20}$$

where

$$(\mathbf{N}[u^{[\nu]}])^{-1}_{i,j} = \frac{1}{(1/h^2)(\Sigma^{[\nu]})_{i,j} + \alpha_\star(u^{[\nu]})_{i,j}}, \tag{2.21}$$

$$(\mathbf{G}[u^{[\nu,\bar{k}+1/2]}])_{i,j} = (g)_{i,j} + \frac{\alpha_\star(u^{[\nu]})_{i,j}(z)^2_{i,j}}{(u)^{[\nu]}_{i,j}} + \frac{1}{h^2}(\overline{\Sigma}^{[\nu]})_{i,j}(u^{[\nu+1,\bar{k}+1/2]})_{i,j}, \tag{2.22}$$

$$(\overline{\Sigma}^{[\nu]})_{i,j}(u^{[\nu+1,\bar{k}+1/2]})_{i,j} = D_1(u^{[\nu]})_{i,j}(u^{[\nu+1,\bar{k}+1]})_{i-1,j} + D_2(u^{[\nu]})_{i,j}(u^{[\nu+1,\bar{k}+1]})_{i,j-1}$$
$$+ D_3(u^{[\nu]})_{i,j}(u^{[\nu+1,\bar{k}]})_{i+1,j} + D_3(u^{[\nu]})_{i,j}(u^{[\nu+1,\bar{k}]})_{i,j+1}.$$

Here the superscripts $\bar{k}$, $\bar{k}+1/2$, and $\bar{k}+1$ denote the current, intermediate and new approximations computed by the SOR method, respectively.

We note first that solving the linearized system (2.19) with very high precision is not necessary. The SOR method can perform only a few inner steps to obtain a good approximate solution. Second, the relaxation parameter ω has a strong influence on the convergence speed. We usually use $\omega > 1$, typically $\omega = 1.2$, because it results in speeding up the convergence. Finally, the numerical implementation to compute one iteration of the proposed GFP method (2.19) based on the SOR method (2.20) can be summarized in Algorithm 1.

Algorithm 1 (Our Proposed GFP method)

Denote by

u	the restored US image
z	the noisy US image
g	the RHS (right-hand side) term of (2.17)
α_1	the regularization parameter of the TV regularization term in (2.3)
α_2	the regularization parameter of the WTV regularization term in (2.3)
ω	relaxation parameter ($\omega = 1.2$)
$\max_{\mathrm{SOR}}$	the maximum number of SOR iterations

$$[u] \leftarrow \mathrm{GFP}\,(u, z, g, \alpha_1, \alpha_2, \omega, \max_{\mathrm{SOR}})$$

- Use input arguments to compute $(\mathbf{N}[u])^{-1}_{i,j}$ (2.21) and the first two terms of $(\mathbf{G}[u])_{i,j}$ in (2.22) for all $1 \le i, j \le n$
- Perform SOR steps for solving (2.19)
  - for $k = 1 : \max_{\mathrm{SOR}}$ do
    - for $i = 1 : n$ do
      - for $j = 1 : n$ do
        - Compute the last term of $(\mathbf{G}[u])_{i,j}$ in (2.22)
        - Updating $(u^{[k+1]})_{i,j}$ using (2.20)
      - end
    - end
  - end

Local fixed-point (LFP) method. Apart from global linearization, the alternative approach for solving the nonlinear discrete systems like (2.17) is to use methods using only *local* linearization; see, e.g., [3, 8–11, 23, 46, 47] and references therein. The main idea is to solve a single nonlinear equation in the given nonlinear system for a (single) unknown using a numerical method of nonlinear equations in one variable.

More precisely, consider the corresponding nonlinear equation for the unknowns $(u)_{i,j-1}$, $(u)_{i-1,j}$, $(u)_{i,j}$, $(u)_{i+1,j}$, $(u)_{i,j+1}$ given by (2.17) as follows:

$$(1/h^2)((\Sigma)_{i,j}(u)_{i,j} - (\overline{\Sigma})_{i,j}(u)_{i,j}) + \alpha_\star(u)_{i,j}\left((u)_{i,j} - \frac{(z)^2_{i,j}}{(u)_{i,j}}\right) = (g)_{i,j}\,.$$

Therefore, at the $\bar{k}$th iteration a nonlinear GS step is given by

$$(1/h^2)((\Sigma^{[\bar{k}+1]})_{i,j}(u^{[\bar{k}+1]})_{i,j} - (\overline{\Sigma}^{[\bar{k}+1]})_{i,j}(u^{[\bar{k}+1]})_{i,j})$$

$$+\alpha_\star(u^{[\bar{k}+1]})_{i,j}\left((u^{[\bar{k}+1]})_{i,j} - \frac{(z)^2_{i,j}}{(u^{[\bar{k}+1]})_{i,j}}\right) = (g)_{i,j}\,, \tag{2.23}$$

where

$$(\Sigma^{[\bar{k}+1]})_{i,j}(u^{[\bar{k}+1]})_{i,j} = (D_1(u^{[\bar{k}+1]})_{i,j} + D_2(u^{[\bar{k}+1]})_{i,j} + 2D_3(u^{[\bar{k}+1]})_{i,j})(u^{[\bar{k}+1]})_{i,j},$$

$$\begin{aligned}(\overline{\Sigma}^{[\bar{k}+1]})_{i,j}(u^{[\bar{k}+1]})_{i,j} = {} & D_1(u^{[\bar{k}+1]})_{i,j}(u^{[\bar{k}+1]})_{i-1,j} + D_2(u^{[\bar{k}+1]})_{i,j}(u^{[\bar{k}+1]})_{i,j-1}) \\ & + D_3(u^{[\bar{k}+1]})_{i,j}((u^{[\bar{k}]})_{i+1,j} + (u^{[\bar{k}]})_{i,j+1}).\end{aligned}$$

If the nonlinear terms $D_*(u^{[\bar{k}+1]})_{i,j}$, $\alpha_\star(u^{[\bar{k}+1]})_{i,j}$ and $1/(u^{[\bar{k}+1]})_{i,j}$ are simply replaced by $D_*(u^{[\bar{k}]})_{i,j}$, $\alpha_\star(u^{[\bar{k}]})_{i,j}$ and $1/(u^{[\bar{k}]})_{i,j}$, we obtain the so-called *Gauss-Seidel-fixed point* relaxation and we shall name this numerical scheme the local FP (LFP) method. As a result, we found experimentally that this relaxation method is inefficient in leading to fast convergence. An improvement can be simply obtained by using a few more steps of FP iterations with respect to the relaxation parameter $\omega \neq 1$ (typically $\omega = 1.3$ and $\max_{\mathrm{FP}} = 2$, where $\max_{\mathrm{FP}}$ denotes the maximum number of FP iterations) as follows:

$$(u^{[\bar{k}+1]})_{i,j} = (1-\omega)\,(u^{[\bar{k}]})_{i,j} + \omega(\overline{u}^{[\bar{k}+1]})_{i,j}, \tag{2.24}$$

where

$$(\overline{u}^{[\bar{k}+1]})_{i,j} = \frac{(\mathbf{G}[u^{[\nu]}])_{i,j} + (1/h^2)((\overline{\Sigma}^{[\nu]})_{i,j}(u^{[\bar{k}+1/2;\nu]})_{i,j})}{(1/h^2)(\Sigma^{[\nu]})_{i,j} + \alpha_\star(u^{[\nu]})_{i,j}}, \tag{2.25}$$

$$(\mathbf{G}[u^{[\nu]}])_{i,j} = (g)_{i,j} + \frac{\alpha_\star(u^{[\nu]})_{i,j}(z)^2_{i,j}}{(u)^{[\nu]}_{i,j}}, \tag{2.26}$$

$$\begin{aligned}(\overline{\Sigma}^{[\nu]})_{i,j}(u^{[\bar{k}+1/2;\nu]})_{i,j} = {} & D_1(u^{[\nu]})_{i,j}(u^{[\bar{k}+1]})_{i-1,j} + D_2(u^{[\nu]})_{i,j}(u^{[\bar{k}+1]})_{i,j-1} \\ & + D_3(u^{[\nu]})_{i,j}((u^{[\bar{k}]})_{i+1,j} + (u^{[\bar{k}]})_{i,j+1}),\end{aligned} \tag{2.27}$$

$$(\Sigma^{[\nu]})_{i,j} = D_1(u^{[\nu]})_{i,j} + D_2(u^{[\nu]})_{i,j} + 2D_3(u^{[\nu]})_{i,j}. \tag{2.28}$$

We note that

$$D_1(u^{[\nu]})_{i,j} = \frac{1}{\sqrt{\left(\frac{(u^{[\nu]})_{i,j}-(u^{[\bar{k}+1]})_{i-1,j}}{h}\right)^2 + \left(\frac{(u^{[\bar{k}]})_{i-1,j+1}-(u^{[\bar{k}+1]})_{i-1,j}}{h}\right)^2 + \epsilon}},$$

$$D_2(u^{[\nu]})_{i,j} = \frac{1}{\sqrt{\left(\frac{(u^{[\bar{k}]})_{i+1,j-1}-(u^{[\bar{k}+1]})_{i,j-1}}{h}\right)^2 + \left(\frac{(u^{[\nu]})_{i,j}-(u^{[\bar{k}+1]})_{i,j-1}}{h}\right)^2 + \epsilon}},$$

$$D_3(u^{[\nu]})_{i,j} = \frac{1}{\sqrt{\left(\frac{(u^{[\bar{k}]})_{i+1,j}-(u^{[\nu]})_{i,j}}{h}\right)^2 + \left(\frac{(u^{[\bar{k}]})_{i,j+1}-(u^{[\nu]})_{i,j}}{h}\right)^2 + \epsilon}}.$$

Finally our proposed method for performing one GS iteration with $\omega \neq 1$ (SOR iteration) can be summarized as given in Algorithm 2.

Algorithm 2 (Our Proposed LFP method)

Denote by

u	the restored US image
z	the noisy US image
g	the RHS (right-hand side) term of (2.17)
α_1	the regularization parameter of the TV regularization term in (2.3)
α_2	the regularization parameter of the WTV regularization term in (2.3)
ω	relaxation parameter ($\omega = 1.3$)
$\max_{\mathrm{FP}}$	the maximum number of inner FP iterations ($\max_{\mathrm{FP}} = 2$)

$$[u] \leftarrow \mathrm{LFP}\,(u, z, g, \alpha_1, \alpha_2, \omega, \max_{\mathrm{FP}})$$

- for $i = 1 : n$ do
  - for $j = 1 : n$ do
    - Set $(u^{[\nu=0]})_{i,j} = (u)_{i,j}$
    - for $\nu = 0 : \max_{\mathrm{FP}}$ do
      - Compute $(\mathbf{G}[u^{[\nu]}])_{i,j}$ using (2.26)
      - Compute $(\overline{\Sigma}^{[\nu]})_{i,j}(u^{[\nu]})_{i,j}$ using (2.27)
      - Compute $(\Sigma^{[\nu]})_{i,j}$ using (2.28)
      - Compute $(\overline{u})_{i,j}$ using (2.25)
      - Set $(u^{[\nu+1]})_{i,j} = (\overline{u})_{i,j}$
    - end
    - Use $(u^{[\nu=0]})_{i,j}$, $(\overline{u})_{i,j}$ and (2.24) to compute $(v)_{i,j}$
  - end
- end

Combined global-local fixed-point (CGLFP) method. In this section we introduce a FP scheme combining the GFP and LFP methods. This FP method has two steps as follows.

In the first (outer) step, we apply the GFP method. The nonlinear terms $\alpha_\star(u)_{i,j}$ and $1/\left(u\right)_{i,j}$ in (2.17) are linearized globally at a previous FP step, denoted by '$*$'. This yields

$$(1/h^2)((\Sigma)_{i,j}(u)_{i,j} - (\overline{\Sigma})_{i,j}(u)_{i,j}) + \alpha_\star(u^*)_{i,j}(u)_{i,j} = (\mathbf{G})_{i,j},$$

where

$$(\mathbf{G})_{i,j} = g_{i,j} + \frac{\alpha_\star(u^*)_{i,j}(z)^2_{i,j}}{(u^*)_{i,j}}. \tag{2.29}$$

In the second (inner) step, we apply the LFP method. The resulting nonlinear system is solved by the nonlinear GS method with a few inner FP steps (2 or 3) and the relaxation parameter $1 \neq \omega \in (0, 2)$ (typically, $\omega = 0.9$ and $\max_{\mathrm{FP}} = 2$) as given by

$$(u^{[\bar{k}+1]})_{i,j} = (1-\omega)\,(u^{[\bar{k}]})_{i,j} + \omega(\overline{u}^{[\bar{k}+1]})_{i,j}, \tag{2.30}$$

where

$$(\overline{u}^{[\bar{k}+1]})_{i,j} = \frac{1}{(1/h^2)(\Sigma^{[\nu]})_{i,j} + \alpha_\star (u^*)_{i,j}} \times \left[(\mathbf{G})_{i,j} + (1/h^2)((\overline{\Sigma}^{[\nu]})_{i,j}(u^{[\bar{k}+1/2;\nu]})_{i,j}) \right], \quad (2.31)$$

$$(\overline{\Sigma}^{[\nu]})_{i,j}(u^{[\bar{k}+1/2;\nu]})_{i,j} = D_1(u^{[\nu]})_{i,j}(u^{[\bar{k}+1]})_{i-1,j} + D_2(u^{[\nu]})_{i,j}(u^{[\bar{k}+1]})_{i,j-1} + D_3(u^{[\nu]})_{i,j}((u^{[\bar{k}]})_{i+1,j} + (u^{[\bar{k}]})_{i,j+1}), \quad (2.32)$$

$$(\Sigma^{[\nu]})_{i,j} = D_1(u^{[\nu]})_{i,j} + D_2(u^{[\nu]})_{i,j} + 2D_3(u^{[\nu]})_{i,j}, \quad (2.33)$$

$$D_1(u^{[\nu]})_{i,j} = \frac{1}{\sqrt{\left(\frac{(u^{[\nu]})_{i,j} - (u^{[\bar{k}+1]})_{i-1,j}}{h}\right)^2 + \left(\frac{(u^{[\bar{k}]})_{i-1,j+1} - (u^{[\bar{k}+1]})_{i-1,j})}{h}\right)^2 + \epsilon}},$$

$$D_2(u^{[\nu]})_{i,j} = \frac{1}{\sqrt{\left(\frac{(u^{[\bar{k}]})_{i+1,j-1} - (u^{[\bar{k}+1]})_{i,j-1}}{h}\right)^2 + \left(\frac{(u^{[\nu]})_{i,j} - (u^{[\bar{k}+1]})_{i,j-1}}{h}\right)^2 + \epsilon}},$$

$$D_3(u^{[\nu]})_{i,j} = \frac{1}{\sqrt{\left(\frac{(u^{[\bar{k}]})_{i+1,j} - (u^{[\nu]})_{i,j}}{h}\right)^2 + \left(\frac{(u^{[\bar{k}]})_{i,j+1} - (u^{[\nu]})_{i,j}}{h}\right)^2 + \epsilon}},$$

and $\nu = 0, 1, 2, \ldots$ represent the inner FP steps. Here $\bar{k}$, $\bar{k} + 1/2$, and $\bar{k} + 1$ denote respectively the current, intermediate and new approximations computed by the nonlinear GS method. We shall name this numerical scheme the combined global-local FP (CGLFP) method. Finally our proposed CGLFP method can be summarized in Algorithm 3.

In order to reduce computational costs of the above three FP methods, they should involve solution on multiple grids using a so-called coarse-to-fine approach. The advantage of such an approach is that a good initial solution for the fine grid problem is obtained by solving coarse grid problems. Since the coarse grid problems can be solved cheaply, it is possible to quickly find an approximate solution to the fine grid problem; using this approximate solution as an initial solution, only a few iterations are needed to provide visually pleasing restoration results on the fine grid.

Suppose that we operate with L levels in total with $l = 1$ the coarsest level (where the image size is the smallest) and $l = L$ the finest level (where the image size is the same as the original one). Here the size of the coarsest level 1 is chosen as 32×32. As a starting point, the standard coarsening is used in computing the coarse-grid domain Ω^H by doubling the grid size in each space direction, i.e. $h \to 2h = H$. In order to transfer grid functions between different levels, our coarse-to-fine method uses the four-point average operator $u_l^H = I_h^H u_l^h$ for restriction and the bi-linear interpolation operator $u_l^h = I_H^h u_l^H$ for prolongation. Here the operators I_h^H and I_H^h are given by

$$(u_l^H)_{i,j} = \frac{1}{4}[(u_l^h)_{2i-1,2j-1} + (u_l^h)_{2i-1,2j} + (u_l^h)_{2i,2j-1} + (u_l^h)_{2i,2j}],$$

$$(u_l^h)_{2i,2j} = \frac{1}{16}[9(u_l^H)_{i,j} + 3((u_l^H)_{i+1,j} + (u_l^H)_{i,j+1}) + (u_l^H)_{i+1,j+1})],$$

$$(u_l^h)_{2i-1,2j} = \frac{1}{16}[9(u_l^H)_{i,j} + 3((u_l^H)_{i-1,j} + (u_l^H)_{i,j+1}) + (u_l^H)_{i-1,j+1})],$$

$$(u_l^h)_{2i,2j-1} = \frac{1}{16}[9(u_l^H)_{i,j} + 3((u_l^H)_{i+1,j} + (u_l^H)_{i,j-1}) + (u_l^H)_{i+1,j-1})],$$

$$(u_l^h)_{2i-1,2j-1} = \frac{1}{16}[9(u_l^H)_{i,j} + 3((u_l^H)_{i-1,j} + (u_l^H)_{i,j-1}) + (u_l^H)_{i-1,j-1})],$$

respectively.

Algorithm 3 (Our Proposed CGLFP Method)

Denote by

u	the restored US image
z	the noisy US image
g	the RHS (right-hand side) term of (2.17)
α_1	the regularization parameter of TV regularization term in (2.3)
α_2	the regularization parameter of WTV regularization term in (2.3)
ω	relaxation parameter ($\omega = 0.9$)
$\max_{\mathrm{FP}}$	the maximum number of inner FP iterations ($\max_{\mathrm{FP}} = 2$)

$$[u] \leftarrow \mathrm{CGLFP}\,(u, z, g, \alpha_1, \alpha_2, \omega, \max_{\mathrm{FP}})$$

- Compute $\alpha_\star(u)_{i,j} = \frac{1}{\alpha_1(u)_{i,j}+\alpha_2}$ and $(\mathbf{G})_{i,j}$ using (2.29) for all $1 \le i, j \le n$
- for $i = 1 : n$ do
  - for $j = 1 : n$ do
    - Set $(u^{[\nu=0]})_{i,j} = (u)_{i,j}$
    - for $\nu = 0$:$\max_{\mathrm{FP}}$ do
      - Compute $(\overline{\Sigma}^{[\nu]})_{i,j}(u^{[\nu]})_{i,j}$ using (2.32)
      - Compute $(\Sigma^{[\nu]})_{i,j}$ using (2.35)
      - Compute $(\overline{u})_{i,j}$ using (2.31)
      - Set $(u^{[\nu+1]})_{i,j} = (\overline{u})_{i,j}$
    - end
    - Use $(u^{[\nu=0]})_{i,j}$, $(\overline{u})_{i,j}$ and (2.30) to compute $(u)_{i,j}$
  - end
- end

Finally, the whole procedure of Method 1 may be summarized with a recursion step as follows:

$$[u^h] \leftarrow \mathrm{CF}(u^h, z^h, g^h, \alpha_1, \alpha_2, \omega, \max_{\mathrm{C}}, \max_{\mathrm{F}})$$

1) If Ω^h = coarsest grid $\left(\left|\Omega^h\right| = 32 \times 32\right)$, solve (2.17) using a FP method as introduced in Section 2.3.2 with $\max_{\mathrm{C}}$ steps and then stop. Else continue with following step.
2) Restriction to the coarse grid:
   $u^H \leftarrow I_h^H u^h$, $z^H \leftarrow I_h^H z^h$, $g^H \leftarrow I_h^H g^h$.
3) Implement the multi-resolution step on the next coarser grid:
   $u^H \leftarrow \mathrm{CF}(u^H, z^H, g^H, \alpha_1, \alpha_2, \omega, \max_{\mathrm{C}}, \max_{\mathrm{F}})$.
4) Interpolation to the next finer grid:
   $u^h \leftarrow I_H^h u^H$.
5) Solve (2.17) using the FP method in 1) with $\max_{\mathrm{F}}$ steps.

2.3.3. Method 2–Full approximation scheme based nonlinear multigrid (FAS-NMG) method

The basic idea of a multigrid (MG) method is to solve the problems on a series of coarse grids and interpolates coarse grid correction back to the fine grids. Performing major computational work on the coarse grids reduces significantly computation time.

One iteration of a standard MG algorithm consists of smoothing high frequency components of the error using a few steps of a smoother (an iterative relaxation technique, e.g., Jacobi, GS, and successive over relaxation (SOR) methods), solving an approximation to the smooth error equation on a coarse grid, interpolating the error correction to the fine grid, and finally adding the error correction into the current approximation (coarse-grid correction step).

An important aspect of the MG method is that the coarse grid solution can be approximated by recursively using the MG idea. That is, on the coarse grid, the smoother is performed to reduce the high frequency component of the errors followed by the projection of a residual equation on yet a coarser grid, and so on. Thus, the MG method requires a series of problems to be solved on a hierarchy of grids with different meshsizes. A multigrid V-cycle is the process that goes from the finest grid down to the coarsest grid and moves back from the coarsest up to the finest. A V($\nu_1;\nu_2$)-cycle is a MG V-cycle algorithm that performs ν_1 steps of the smoother on each level before projecting the residual back to the coarse grid (pre-smoothing step), and performs ν_2 steps of the smoother after interpolating the coarse grid correction back to the fine grid (post-smoothing step). For other MG cycling algorithms and more details, we refer to [7, 55, 60].

Full approximation scheme based NMG (FAS-NMG) method has become an efficient approach for solving nonlinear problems; see e.g. [11, 12, 16, 18, 20–23, 57]. Here we have to solve the nonlinear PDE (2.17), i.e.

$$\underbrace{-\mathcal{K}^h(u^h)_{i,j} + \alpha^h_\star(u^h)_{i,j}\left((u^h)_{i,j} - \frac{(z^h)^2_{i,j}}{(u^h)_{i,j}}\right)}_{\mathcal{N}^h(u^h)_{i,j}} = (g^h)_{i,j}.$$

Recall that $g^h = 0$ on the finest grid.

Let $\overline{u}^h$ be the current approximation of u^h after a few smoothing iterations in a pre-smoothing step on a fine-grid problem where we denote by u^h the exact solution of (2.17). Then, the algebraic error $\boldsymbol{e}^h$ of the solution is given by $e^h = u^h - \overline{u}^h$. The nonlinear residual equation is given by

$$\mathcal{N}^h(\overline{u}^h + e^h) - \mathcal{N}^h(\overline{u}^h) = g^h - \mathcal{N}^h(\overline{u}^h) = r^h.$$

In order to correct the approximated solution $\overline{u}^h$ on the fine grid, one needs to compute the error e^h. However, the computation of e^h is prohibitively expensive and cannot be computed directly on the fine grid. Since high frequency components of the error in the pre-smoothing step have already been removed by the smoother, we can transfer the following nonlinear system to the coarse grid as follows:

$$\underbrace{\mathcal{N}^h(\overline{u}^h + e^h)}_{\mathcal{N}^h(u^h)} = \underbrace{r^h + \mathcal{N}^h(\overline{u}^h)}_{g^h} \rightarrow \underbrace{\mathcal{N}^H(\overline{u}^H + e^H)}_{\mathcal{N}^H(u^H)} = \underbrace{r^H_{\widehat{l}} + \mathcal{N}^H(\overline{u}^H)}_{g^H} \tag{2.34}$$

where $H = 2h$ is the new cell size $H \times H$ and $g^H \neq 0$ on the coarse grid. After the nonlinear residual equation on the coarse grid (2.34) has been solved with a method of our choice, the coarse-grid correction $e^H = u^H - \overline{u}^H$ is then interpolated back to the fine grid e^h that can now be used for updating the approximated solution $\overline{u}^h$ of the original system on the fine grid $\overline{u}^h_{new} = \overline{u}^h + e^h$. The last step for a FAS-NMG method is to perform the smoother again to remove high frequency parts of the interpolated error.

The FAS-NMG components for solving (2.17) are as follows.

1. The *MG smoother* is obtained from the FP method discussed in Section 2.3.2; GFP, LFP or GLFP method.

2. The *standard coarsening method* is used in the coarse-grid domain Ω^H by doubling the grid size in each space direction − i.e. $h \to 2h = H$.

3. The *intergrid transfer operators* are determined by averaging and bilinear interpolation techniques, for the restriction and interpolation operators denoted respectively by I_h^H and I_H^h as represented in the previous section.

4. The *discretization coarse grid approximation* (DCA) method is applied to compute the coarse-grid operator of $\mathcal{N}^h(u^h)$, where the EL system is re-discretized directly.

5. SITM method as given by (2.18) is used as the *coarsest grid solver* for solving the nonlinear residual equation on the coarsest grid, typically the 4×4 grid.

6. The *MG cycle* is $V(\nu_1;\nu_2)$-cycle for solving the discrete nonlinear system (2.17).

For practical applications our FAS-NMG approach is stopped if the maximum number of V−cycles ε_1 is reached (usually $\varepsilon_1 = 20$), the relative residual obtained from the EL equation (2.17) is smaller than a small number $\varepsilon_2 > 0$ (typically $\varepsilon_2 = 10^{-4}$), the change in two consecutive steps of PSNR (see the meaning for PSNR in Section 3) is smaller than some $\varepsilon_3 > 0$ (typically $\varepsilon_3 = 10^{-2}$), the change in two consecutive steps of J_{α_1,α_2} is smaller than a small number $\varepsilon_4 > 0$ (typically $\varepsilon_4 = 10^{-4}$) or the relative difference between two consecutive solutions is smaller than a small number $\varepsilon_5 > 0$ (typically $\varepsilon_5 = 10^{-3}$). Finally, the pseudo-code implementation of our FAS-NMG method can be summarized in Algorithm 4.

2.3.4. Method 3–Split Bregman (SB) iteration

Alternating direction method of multipliers (ADMM) has been successfully applied to minimize the energy functional involving TV regularization and quadratic or non-quadratic data fitting term. For example, the authors in [26] used the split Bregman (SM) iteration to solve the ROF model, while the augmented Lagrangian method (ALM) has been developed for the ROF model in [52, 53]. Note that both SB and ALM employ ADMM to minimize their energy functionals and the convergence of ADMM is always guaranteed. In [52, 53], the authors have proven that the two methods are in fact equivalent. For applying ADMM to minimize the energy functionals, we refer to [32–34, 51, 61].

In this section, our aim is to show how SB method can be used to solve our proposed SN removal model. Before starting our discussion on SB method, we note that the minimizations of $\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}$ in (2.4) and $\bar{\bar{\mathcal{J}}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u)$ in (2.15) lead to the same EL equation (2.14a); see the proof in Theorem 2.2. Instead of solving the minimization of $\bar{\mathcal{J}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}$ directly, we therefore consider the minimization of $\bar{\bar{\mathcal{J}}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u)$. We start by introducing an auxiliary splitting vector variable $\boldsymbol{w} = (w_1, w_2)$, a Bregman iterative parameter $\boldsymbol{b} = (b_1, b_2)$, and a positive penalty parameter θ to convert $\bar{\bar{\mathcal{J}}}^{\mathrm{KKWV}}_{\alpha_1,\alpha_2}(u)$ into following form

Algorithm 4 (FAS-NMG Algorithm)

Denote the FAS-NMG parameters as follows:

u^h	the restored image
z^h	the noisy image
g^h	the RHS (right-hand side) term of (2.17)
α_1	the regularization parameter of TV regularization term in (2.3)
α_2	the regularization parameter of WTV regularization term in (2.3)
ω	relaxation parameter
$\max_S$	the maximum number of iterations using by a smoother
ν_1	pre-smoothing steps on each level
ν_2	post-smoothing steps on each level
$\overrightarrow{\varepsilon}$	the tolerance ($\overrightarrow{\varepsilon} = (\varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon_4, \varepsilon_5) = (20, 10^{-4}, 10^{-2}, 10^{-4}, 10^{-3})$)

$$u^h \leftarrow \text{FASNMG}\left(u^h, z^h, g^h, \alpha_1, \alpha_2, \omega, \max_S, \nu_1, \nu_2, \overrightarrow{\varepsilon}\right)$$

- Select initial guess solution $\widetilde{u}^h_{\text{initial}}$ on the finest grid
- Set $K = 0$, $(u^h)^{[K]} = \widetilde{u}^h_{\text{initial}}$, $\widetilde{\varepsilon}_2 = \varepsilon_2 + 1$, $\widetilde{\varepsilon}_3 = \varepsilon_3 + 1$, $\widetilde{\varepsilon}_4 = \varepsilon_4 + 1$ and $\widetilde{\varepsilon}_5 = \varepsilon_5 + 1$
- While ($K < \varepsilon_1$ AND $\widetilde{\varepsilon}_2 \geq \varepsilon_2$ AND $\widetilde{\varepsilon}_3 \geq \varepsilon_3$ AND $\widetilde{\varepsilon}_4 \geq \varepsilon_4$ AND $\widetilde{\varepsilon}_5 \geq \varepsilon_5$)
  - ► $(u^h)^{[K+1]} \leftarrow \text{FASCYC}((u^h)^{[K]}, z^h, g^h, \alpha_1, \alpha_2, \omega, \max_S, \nu_1, \nu_2)$
  - ► $\widetilde{\varepsilon}_2 = ||g^h - \mathcal{N}^h((u^h)^{[K+1]})||_{L^2(\Omega^h)} / ||g^h - \mathcal{N}^h(\widetilde{u}^h_{\text{initial}})||_{L^2(\Omega^h)}$
  - ► $\widetilde{\varepsilon}_3 = |\text{PSNR}(u^h)^{[K+1]} - \text{PSNR}(u^h)^{[K]}|$, [PSNR is given by (3.1)]
  - ► $\widetilde{\varepsilon}_4 = |J^h_{\alpha_1,\alpha_2}(u^h)^{[K+1]} - J^h_{\alpha_1,\alpha_2}(u^h)^{[K]}|$
  - ► $\widetilde{\varepsilon}_5 = ||(u^h)^{[K+1]} - (u^h)^{[K]}||_{L^2(\Omega^h)} / ||(u^h)^{[K]}||_{L^2(\Omega^h)}$
  - ► $K = K + 1$
- end

where

$$\left[u^h\right] \leftarrow \text{FASCYC}(u^h, z^h, g^h, \alpha_1, \alpha_2, \omega, \max_S, \nu_1, \nu_2)$$

- If Ω_h = coarsest grid ($|\Omega_h| = 4 \times 4$), solve (2.17) using (2.18) and then stop. Else continue with the following steps.
- Pre-smoothing:
  For $k = 1$ to ν_1, $\left[u^h\right] \leftarrow \text{Smoother}(u^h, z^h, g^h, \alpha_1, \alpha_2, \omega, \max_S)$
- Restriction to the coarse grid:
  $u^H \leftarrow I^H_h u^h$, $z^H \leftarrow I^H_h z^h$
- Set the initial solution for the coarse-grid problem:
  $\widetilde{u}^H \leftarrow u^H$
- Compute the new right-hand side for the coarse-grid problem:
  $g^H \leftarrow I^H_h(g^h - \mathcal{N}^h(u^h)) + \mathcal{N}^H\left(u^H\right)$
- Implement the FAS-NMG method on the coarse-grid problem:
  $\left[u^H\right] \leftarrow \text{FASCYC}\left(u^H, z^H, g^H, \alpha_1, \alpha_2, \omega, \max_S, \nu_1, \nu_2\right)$
- Add the coarse-grid corrections:
  $u^h \leftarrow u^h + I^h_H\left(u^h - \widetilde{u}^H\right)$
- Post-smoothing:
  For $k = 1$ to ν_2, $\left[u^h\right] \leftarrow \text{Smoother}(u^h, z^h, g^h, \alpha_1, \alpha_2, \omega, \max_S)$

$$\mathcal{E}(u,\boldsymbol{w};\boldsymbol{b}) = \int_\Omega |\boldsymbol{w}|\, d\Omega + \frac{\theta}{2}\int_\Omega (\boldsymbol{w}-\nabla u-\boldsymbol{b})^2 d\Omega + \int_\Omega F(u;\alpha_1,\alpha_2)d\Omega. \tag{2.35}$$

In order to find the minimizer of (2.35), we use an alternating optimization procedure to find the optimal values of u and $\boldsymbol{w}$ where the process involves only three main steps.

Step 1. For the first step we fix $\boldsymbol{w}$ and update u by minimizing the following functional

$$\mathcal{E}_1(u;\boldsymbol{b}) = \frac{\theta}{2}\int_\Omega (\boldsymbol{w}-\nabla u-\boldsymbol{b})^2 d\Omega + \int_\Omega F(u;\alpha_1,\alpha_2)d\Omega. \tag{2.36}$$

This minimization problem can be solved using the associated EL equation

$$-\triangle u + \alpha_\star(u)u = \frac{\alpha_\star(u)z^2}{u} - \theta\nabla\cdot(\boldsymbol{w}-\boldsymbol{b}), \tag{2.37}$$

with Neumann boundary conditions. Recall that $\alpha_\star(u) = \frac{1}{\alpha_1 u+\alpha_2}$.

To solve the nonlinear PDE in (2.37), we apply a coupled outer-inner iteration method as follows. In the outer iteration, we start from an initial solution $u^{[0]}$ (typically $u^{[0]} = z$) and compute a sequence of approximate solutions $u^{[1]}, u^{[2]}, \cdots, u^{[\nu]}, u^{[\nu+1]}, \cdots$ by solving the discrete EL equation

$$\bar{\mathbf{N}}[u^{[\nu]}][u^{[\nu+1]}] = \bar{\mathbf{G}}[u^{[\nu]}] \tag{2.38}$$

until the stopping rule $\left\|u^{[\nu+1]}-u^{[\nu]}\right\|^2_{L^2(\Omega)} / \left\|u^{[\nu]}\right\|^2_{L^2(\Omega)} < \epsilon_{\text{SB\_out}}$ is met.

We note first that (2.38) is derived from the finite difference approximations in Section 2.3.1 and the GFP method in Section 2.3.2, where the symbols 'h' and '$(\cdot)_{i,j}$' are dropped for simplicity, ν denotes the index for the outer iteration step,

$$\begin{aligned}
&\bar{\mathbf{N}}[u^{[\nu]}] = -\mathcal{L} + \alpha_\star(u^{[\nu]}),\\
&\mathbf{G}[u^{[\nu]}] = \frac{\alpha_\star(u^{[\nu]})z^2}{u^{[\nu]}} - \theta\nabla^\dagger\cdot(\boldsymbol{w}-\boldsymbol{b}),\\
&\mathcal{L}u = \delta^-_{x_1}\delta^+_{x_1}u + \delta^-_{x_2}\delta^+_{x_2}u,\\
&\nabla^\dagger\cdot\boldsymbol{v} = \delta^-_{x_1}v_1 + \delta^-_{x_2}v_2, (\boldsymbol{v}=(v_1,v_2)).
\end{aligned}$$

Here $\mathcal{L}$ and $\nabla^\dagger\cdot$ represent the discrete versions of $\triangle$ (Laplace operator) and $\nabla\cdot$ (divergence operator), respectively. Second, we note that for each outer step ν the discrete linear system in (2.38) is solved by the preconditioned conjugate gradient (PCG) method as the inner iteration. We stop the inner iterations when the size of the relative residual at the $\tilde{k}$ iteration $\left\|r_{\tilde{k}}\right\|^2_{L^2(\Omega)} / \left\|r_0\right\|^2_{L^2(\Omega)} < \epsilon_{\text{SB\_in}}$. Finally, we note that $\mathcal{E}_1(u;\boldsymbol{b})$ is convex, which implies that the corresponding equation (2.38) has a unique solution. By experiments, we find the GFP method yields an optimal solution by running about ten iterations. That is to say the optimal solution can be found by the GFP method very efficiently. Thus, Step 1 does not have to be solved exactly. In all our experiments we take $\epsilon_{\text{SB\_out}} = \epsilon_{\text{SB\_In}} = 0.01$ to reduce computation time.

Step 2. For the second step we fix u and update $\boldsymbol{w}$ by minimizing the following functional

$$\mathcal{E}_2(\boldsymbol{w};\boldsymbol{b}) = \int_\Omega |\boldsymbol{w}|\, d\Omega + \frac{\theta}{2}\int_\Omega (\boldsymbol{w}-\nabla u-\boldsymbol{b})^2 d\Omega. \tag{2.39}$$

We can obtain its EL as follows

$$\frac{\boldsymbol{w}}{|\boldsymbol{w}|} + \theta(\boldsymbol{w}-\nabla u-\boldsymbol{b}) = 0 \tag{2.40}$$

which can be solved component wisely through following discrete 2D analytical generalized soft thresholding equation with the convention that $0/0 = 0$

$$(\boldsymbol{w})_{i,j} = \max\left(|\nabla(u)_{i,j} + (\boldsymbol{b})_{i,j}| - \frac{1}{\theta}, 0\right) \frac{\nabla(u)_{i,j} + (\boldsymbol{b})_{i,j}}{|\nabla(u)_{i,j} + (\boldsymbol{b})_{i,j}|}. \tag{2.41}$$

Step 3. For the last step we apply u in Step 1 and $\boldsymbol{w}$ in Step 2 to update the new Bregman iterative parameter $\boldsymbol{b}$; see Algorithm 5.

Finally the whole procedure is stopped if the maximum number of iteration ε_1 is reached (usually $\varepsilon_1 = 1000$) and the relative difference between two consecutive solutions is smaller than a small number $\varepsilon_5 > 0$ (typically $\varepsilon_5 = 10^{-3}$). Algorithm 5 summarizes SB method.

Algorithm 5 (Split Bregman Method)

Denote by

u	the restored US image
z	the noisy US image
α_1	the regularization parameter of TV regularization term in (2.3)
α_2	the regularization parameter of WTV regularization term in (2.3)
θ	penalty parameter ($\theta = 1$)
$\overrightarrow{\varepsilon_\star}$	the tolerance ($\overrightarrow{\varepsilon_\star} = (\varepsilon_1, \varepsilon_5)$)

$$[u] \leftarrow \mathrm{SB}\,(u, z, \alpha_1, \alpha_2, \theta, \overrightarrow{\varepsilon_\star})$$

- Initialize $K = 0$, $u^{[0]} = z$, $\boldsymbol{w}^{[K]} = \boldsymbol{b}^{[K]} = \mathbf{0}$ $\widetilde{\varepsilon}_5 = \varepsilon_5 + 1$
- While ($K < \varepsilon_1$ AND $\widetilde{\varepsilon}_5 \geq \varepsilon_5$)
  - ▶ Use $\boldsymbol{w}^{[K]}$ and $\boldsymbol{b}^{[K]}$ to compute $u^{[K+1]}$ according to (2.38)
  - ▶ Use $u^{[K+1]}$ and $\boldsymbol{b}^{[K]}$ to compute $\boldsymbol{w}^{[K+1]}$ according to (2.41)
  - ▶ Use $u^{[K+1]}$ and $\boldsymbol{w}^{[K+1]}$ to update Bregman iterative parameter $\boldsymbol{b}^{[K+1]} \leftarrow \boldsymbol{b}^{[K]} + \nabla u^{[K+1]} - \boldsymbol{w}^{[K+1]}$
  - ▶ $\widetilde{\varepsilon}_5 = ||u^{[K+1]} - u^{[K]})||_{L^2(\Omega^h)} / ||u^{[K]}||_{L^2(\Omega^h)}$
  - ▶ $K = K + 1$
- end
- $u = u^{[K]}$

We have so far presented three numerical methods for solving our proposed SN removal model (2.4). So it remains to test and compare the overall performances of these numerical techniques. Their performances will be tested and reported in the next section.

3. Numerical Results and Discussion

In this section, we carry out numerical experiments from several test cases for both synthetic and real US images to

(i) compare the restoration results of the proposed model in (2.4) with the state-of-the-art models for SN removal in the literature;

(ii) illustrate the overall performances of the three numerical methods discussed in Section 2; and

(iii) to assess the accuracy and efficiency of our proposed FAS-NMG technique with regard to parameter changes.

We note first that all numerical algorithms for Methods $1-3$ were implemented in MATLAB (version R2011a) and run on an iMac under OS X EI Capitan 10.11.6, at a 4 GHz clock speed and equipped with an Intel Core i7 and 32 GB of RAM. Second, a peak signal to noise ratio (PSNR) and a relative error (ReErr) of a restored image are used for the evaluated experiments by measuring the quality of the restored images or the ability to reduce SN from the given noisy images. The PSNR and ReErr are defined as follows:

$$\mathrm{PSNR} = 20\log_{10}\left[\frac{\max_{u_0}}{\sqrt{\mathrm{MSE}}}\right], \tag{3.1}$$

$$\mathrm{ReErr} = \frac{\|\boldsymbol{u}-\boldsymbol{u}_0\|_2^2}{\|\boldsymbol{u}_0\|_2^2}, \tag{3.2}$$

where

$$\mathrm{MSE} = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\left((u)_{i,j}-(u_0)_{i,j}\right)_2^2,$$

$\boldsymbol{u}_0$ and $\boldsymbol{u}$ represent respectively the vectors of the original (clean) image u_0 and the restored image u. Here $\max_{u_0}$ represents the maximum intensity value that exists in the original image. $(u_0)_{i,j}$ and $(u)_{i,j}$ are the sample values of the original image and the restored image at a sample point $(x_{1_i}, x_{2_j}) \in \Omega^h$.

3.1. Comparison with other SN removal models

The standard test images: "ring", "Lena" and "pepper" are used in this section to compare the performance between two competing SN removal models in the literature and the one proposed in Section 2 as part of this paper. The sizes of the three images are all the same which is 256×256 pixels. The noise formation model in (1.4) is being assumed for creating the noisy versions of the images. The stopping criterion of the SN removal models under consideration is that the relative difference between the successive iterates of the restored image should satisfy the following inequality:

$$\frac{\left\|\boldsymbol{u}^{[m+1]}-\boldsymbol{u}^{[m]}\right\|_2^2}{\left\|\boldsymbol{u}^{[m]}\right\|_2^2} < 10^{-3}. \tag{3.3}$$

We remark that there are two regularization parameters α_1 and α_2 in the proposed model. In order to reduce the computation time in the search for good regularization parameters, we fix α_1 in all the tests, typically $\alpha_1 = 10^{-4}$. Therefore, we need only to determine the best value of α_2 for their tested values such that the relative error of the restored image $u(\alpha_2)$ with respect to the original image u is the smallest, i.e.,

$$\frac{\|\boldsymbol{u}(\alpha_2)-\boldsymbol{u}_0\|_2^2}{\|\boldsymbol{u}_0\|_2^2} \tag{3.4}$$

is the smallest among all tested values of α_2.

3.1.1. Experiments on synthetic images

Fig. 3.1 shows the original test images to be used here for the performance test in removing SN by various models.

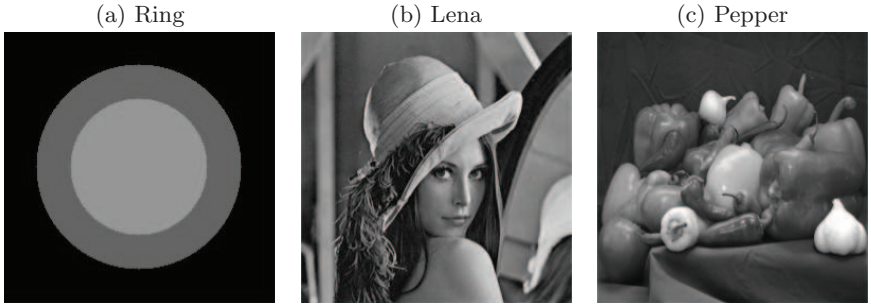


Fig. 3.1. The original test images.

In Fig. 3.2, we compare the performance of the proposed model with those by KKWV and HY models using the test image "ring" degraded with the standard deviation $\sigma_n^2 = 10, 15$

Table 3.1: SN removal models evaluated using PSNR. Test images with the size 256×256 corrupted by the SN generated by (1.4) with the noise standard deviation $\sigma_n^2 = 10, 15$, and 20.

Image	σ_n^2	PSNR KKWV	PSNR HY	PSNR **Proposed Model**
Ring	10	37.96	38.37	**39.30**
	15	35.06	36.10	**37.98**
	20	31.07	32.22	**33.37**
Lena	10	25.50	26.10	**28.97**
	15	21.95	23.84	**25.82**
	20	20.30	21.43	**23.31**
Pepper	10	28.18	30.16	**32.02**
	15	26.90	27.93	**29.05**
	20	24.14	25.52	**26.75**

Table 3.2: SN removal models evaluated using ReErr. Test images with the size 256×256 corrupted by the SN generated by (1.4) with the noise standard deviation $\sigma_n^2 = 10, 15$, and 20.

Image	σ_n^2	ReErr KKWV	ReErr HY	ReErr **Proposed Model**
Ring	10	0.0015	0.0013	**0.0011**
	15	0.0096	0.0053	**0.0029**
	20	0.0581	0.0250	**0.0139**
Lena	10	0.0145	0.0126	**0.0065**
	15	0.0490	0.0267	**0.0143**
	20	0.0991	0.0374	**0.0289**
Pepper	10	0.0111	0.0071	**0.0046**
	15	0.0407	0.0224	**0.0114**
	20	0.0836	0.0305	**0.0251**

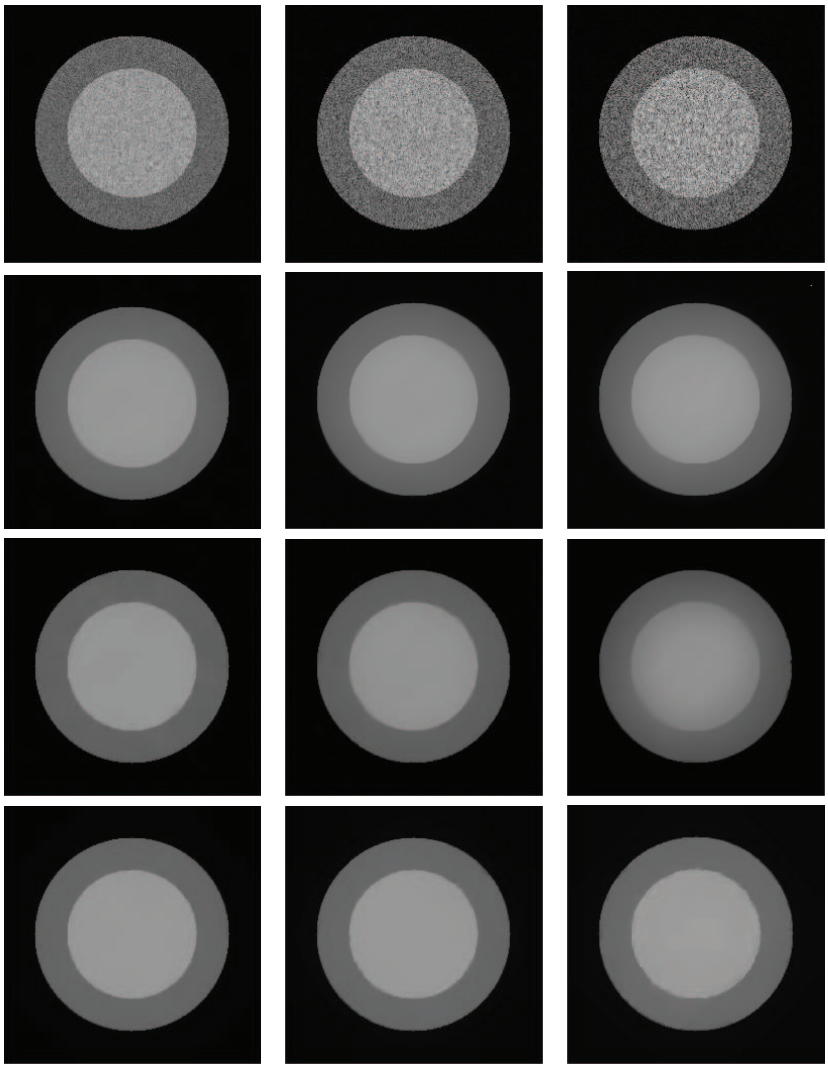


Fig. 3.2. Noisy and restored images (image: ring): ROW 1: image corrupted by SN with the standard deviation $\sigma_n^2 = 10, 15$ and 20, respectively; ROW 2: images restored using KKWV model ($\alpha = 0.010, 0.0125, 0.02$); ROW 3: images restored using HY model ($\alpha = 0.25, 0.50, 0.75$); ROW 4: images restored using the proposed model ($\alpha_2 = 0.001, 0.00125, 0.002$).

and 20, respectively. The first row in the figure shows the degraded images with different SN strength. The second row shows the SN removal capability of KKWV model for the noisy images shown in the first row of the figure. The third row illustrates the performance of HY model and finally the last row shows the restoration results by the proposed model under various SN noise strength.

Tables 3.1 and 3.2 show respectively the PSNR and ReErr of the restored ring images produced by the proposed model and other SN removal models. In this experiment, the ring image is corrupted by the SN with the noise standard deviation $\sigma_n^2 = 10$, 15, and 20. As expected, the restoration results show that as the SN strength increases the PSNR and ReErr of

Fig. 3.3. Noisy and restored images (image: Lena): ROW 1: image corrupted by SN with the standard deviation $\sigma_n^2 = 10, 15$ and 20, respectively; ROW 2: images restored using KKWV model ($\alpha = 0.02, 0.05, 0.10$); ROW 3: images restored using HY model ($\alpha = 0.30, 0.50, 0.75$); ROW 4: images restored using the proposed model ($\alpha_2 = 0.0005, 0.001, 0.002$).

the restored images by the proposed model are better than those of other two models considered here.

Similarly the SN removal capacity of the models considered here is further tested and compared in Figures 3.3−3.4 and Tables 3.1−3.2 for the test images "Lena" and "pepper". The performance (in terms of SN reduction) of the proposed model is further well presented in these results.

Fig. 3.5 shows a one-dimensional profile of the image "ring" (128th column with 256 rows) treated with other models and the proposed one. The original, noisy and restored images are shown in the figure with the noise standard deviation $\sigma_n^2 = 10, 15$ and 20. It can be observed from Fig. 3.5 that all models can be used to remove SN from the noisy images. However, as

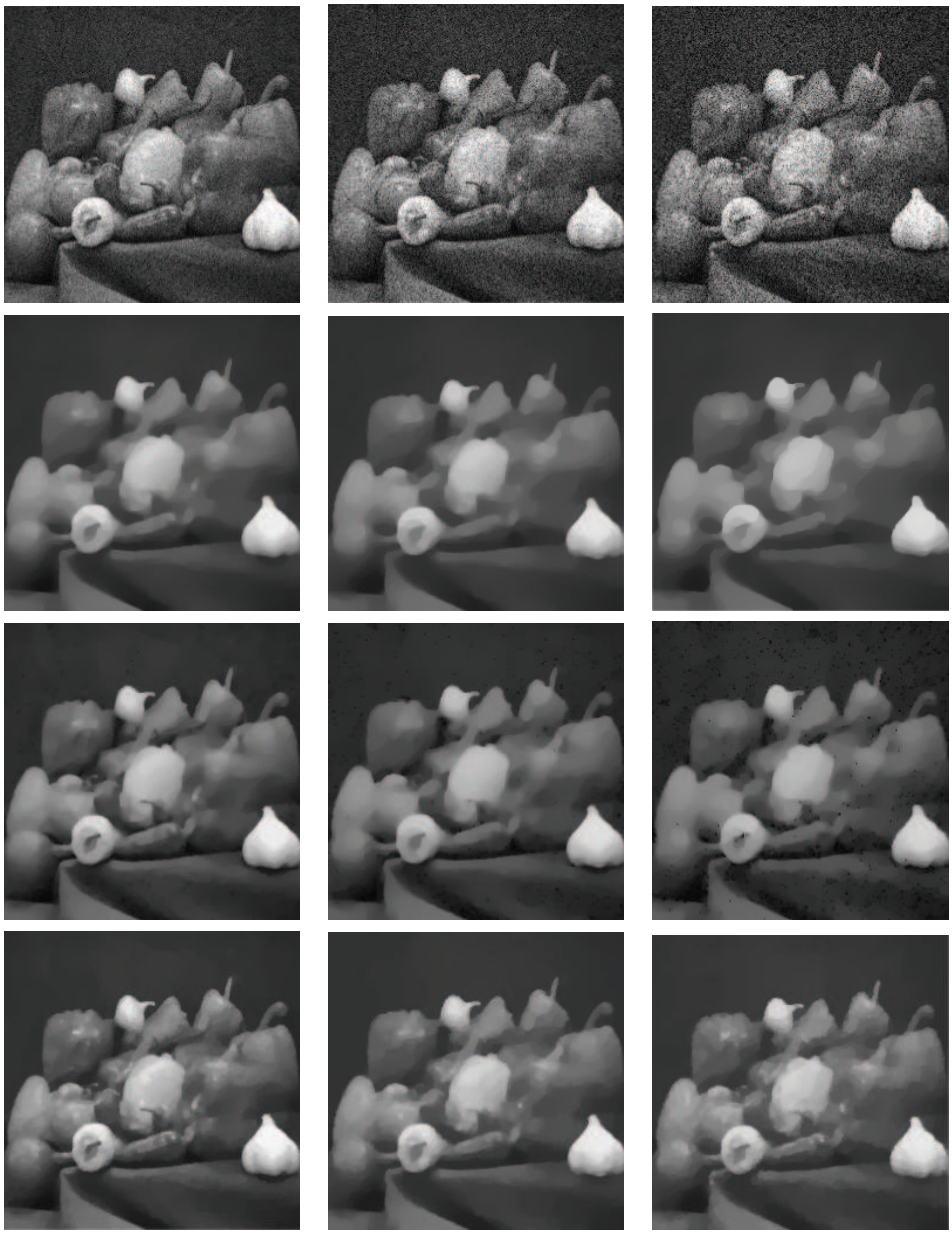


Fig. 3.4. Noisy and restored images (image: pepper): ROW 1: image corrupted by SN with the standard deviation $\sigma_n^2 = 10, 15$ and 20, respectively; ROW 2: images restored using KKWV model ($\alpha = 0.02, 0.05, 0.10$); ROW 3: images restored using HY model ($\alpha = 0.30, 0.50, 0.75$); ROW 4: images restored using the proposed model ($\alpha_2 = 0.0005, 0.001, 0.002$).

the SN strength increases the proposed model yields better restoration quality for reducing SN.

Fig. 3.6 shows the enlarged portions of the Lena image in the homogeneous of regions. The first row shows the enlarged portion in the image Lena and its noisy version with the noise standard deviation $\sigma_n^2 = 10$ in the selected region. The second row shows the restoration results by KKWV and HY models and the proposed model. We can see that while KKWV and HY

models produce the undesirable blocky (staircase) images, the proposed model has recovered the image without causing any staircase effect. This fact is evident from the homogeneous regions presented in these enlarged images.

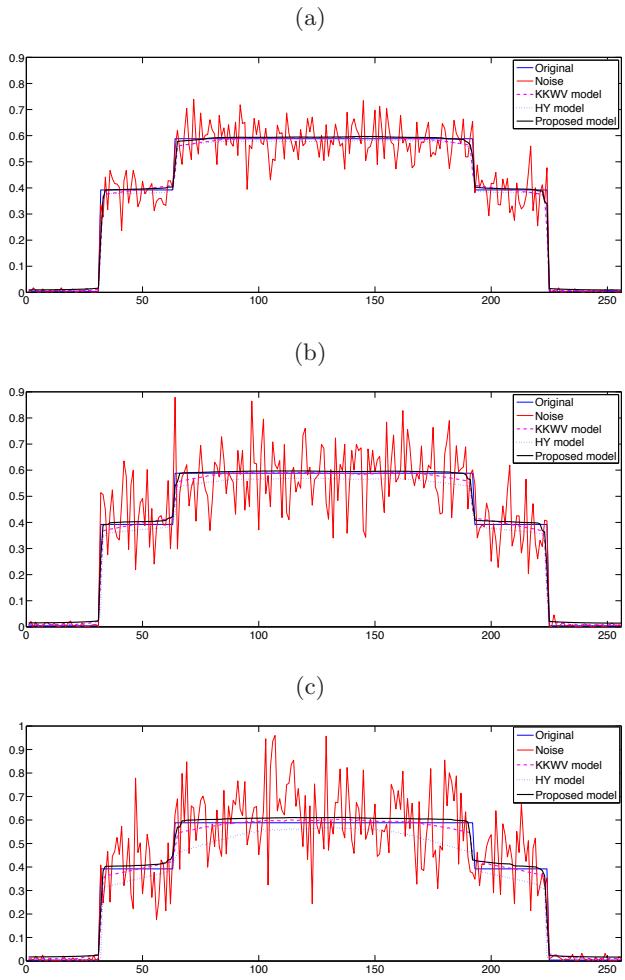


Fig. 3.5. Row profiles plotted for image "ring" in Figs. 3.1−3.2 is selected at the 128th column with 256 rows, for display at (a) $\sigma_n^2 = 10$; (b) $\sigma_n^2 = 15$; and (c) $\sigma_n^2 = 20$.

3.1.2. Experiments on real US images

In this part, we test the performance of the proposed model and compared with other two models using three real US images, "baby", "liver" and "kidney". These real US images from different medical applications are the same size which is 256×256 pixels and corrupted with unknown SN strength.

The first rows in Figures 3.7−3.9 show the real US images and the restored images by the proposed model. The last rows in Figures. 3.7−3.9 show the restored images by KKWV and HY models.

Figures 3.10−3.12 (from top-to-bottom) show the corresponding signal of one column 128

Table 3.3: Experts evaluation on the restoration quality for the clinical US images as shown in Figs. 3.7−3.9.

Expert	Image	Grading on restored results		
		KKWV	HY	**Proposed Model**
1	Baby	3	4	**5**
	Liver	2	3	**4**
	Kidney	3	4	**5**
2	Baby	2	3	**5**
	Liver	2	4	**5**
	Kidney	3	4	**5**

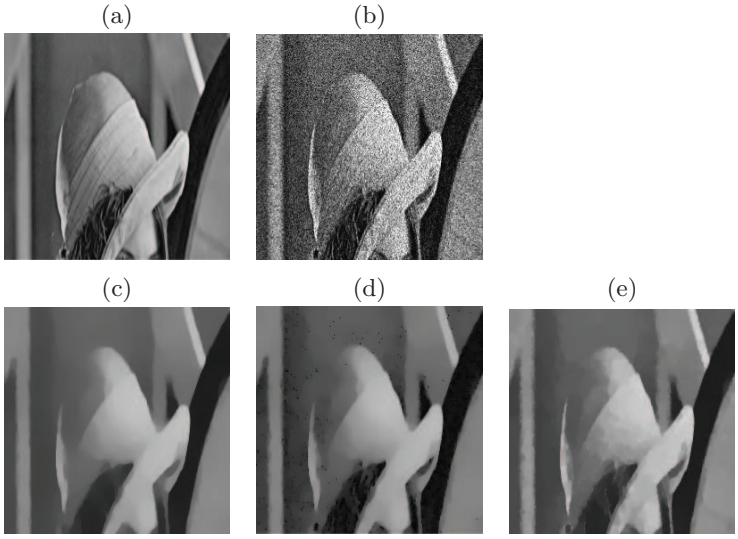


Fig. 3.6. Enlarged portions of the image Lena: (a) original image; (b) noisy image with $\sigma_n^2 = 10$; (c) results by KKWV model ($\alpha = 0.02$); (d) results by HY model ($\alpha = 0.3$); and (e) results by the proposed model ($\alpha_2 = 0.0005$).

(named by the highlight line) and the restored signal by KKWV model, HY model and the proposed model. The horizontal axis is the pixel along the highlight line and the vertical axis is the gray value. Digits 1−3 are marked along the highlight line. Here the areas in Digit 1 (the local changes of intensity) and Digit 2 (the strong reflection points) should be well preserved and the areas in Digit 3 (the homogeneous regions) should be smoothed in SN reduction. In the aspect of protection Digit 1, KKWV and HY models are not successful as the proposed model, however, all models are able to smooth the homogeneous area in Digit 3. Besides, the gray value in Digit 2 is 0.95 in the baby image, 0.26 in the liver image and 0.77 in the kidney image after performing SN reduction with our proposed model, which is the highest (or lowest) and clearest to the original signal among three models, that is, our proposed model outperforms to protect the high reflection point in Digit 2. Moreover, it is also be observed through Table 3.3 by two experts grading on the restoration quality of various models for SN reduction from the real US images. The experts have graded the restored images in the range of $[1, 5]$ of which 1 and 5 are used to represent the poor and best results respectively. It is clear that our proposed model is the most effective variational model among the three for clinical US images.

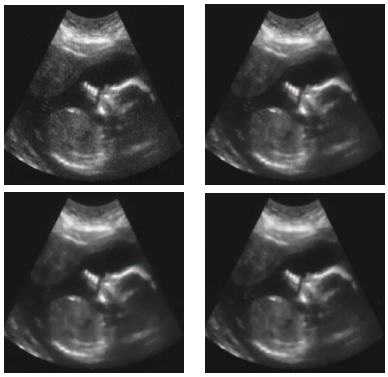


Fig. 3.7. Comparison with different SN removal models on a real US image of a baby. Top row (from left to right): original image; restored image by proposed model ($\alpha_2 = 0.0001$); bottom row: restored image by KKWV model ($\alpha = 0.01$); restored image by HY model ($\alpha = 0.1$).

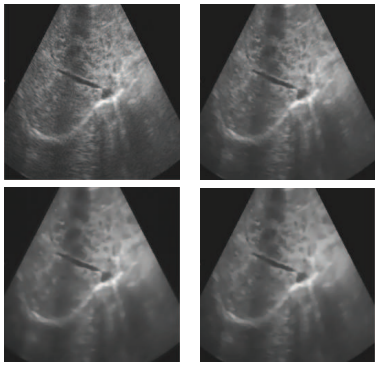


Fig. 3.8. Comparison with different SN removal models on a real US image of a human liver. Top row (from left to right): original image; restored image by proposed model ($\alpha_2 = 0.0001$); bottom row: restored image by KKWV model ($\alpha = 0.01$); restored image by HY model ($\alpha = 0.1$).

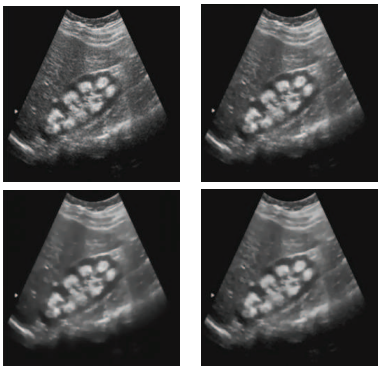


Fig. 3.9. Comparison with different SN removal models on a real US image of a human kidney. Top row (from left to right): original image; restored image by proposed mode ($\alpha_2 = 0.0001$)l; bottom row: restored image by KKWV model ($\alpha = 0.01$); restored image by HY model ($\alpha = 0.1$).

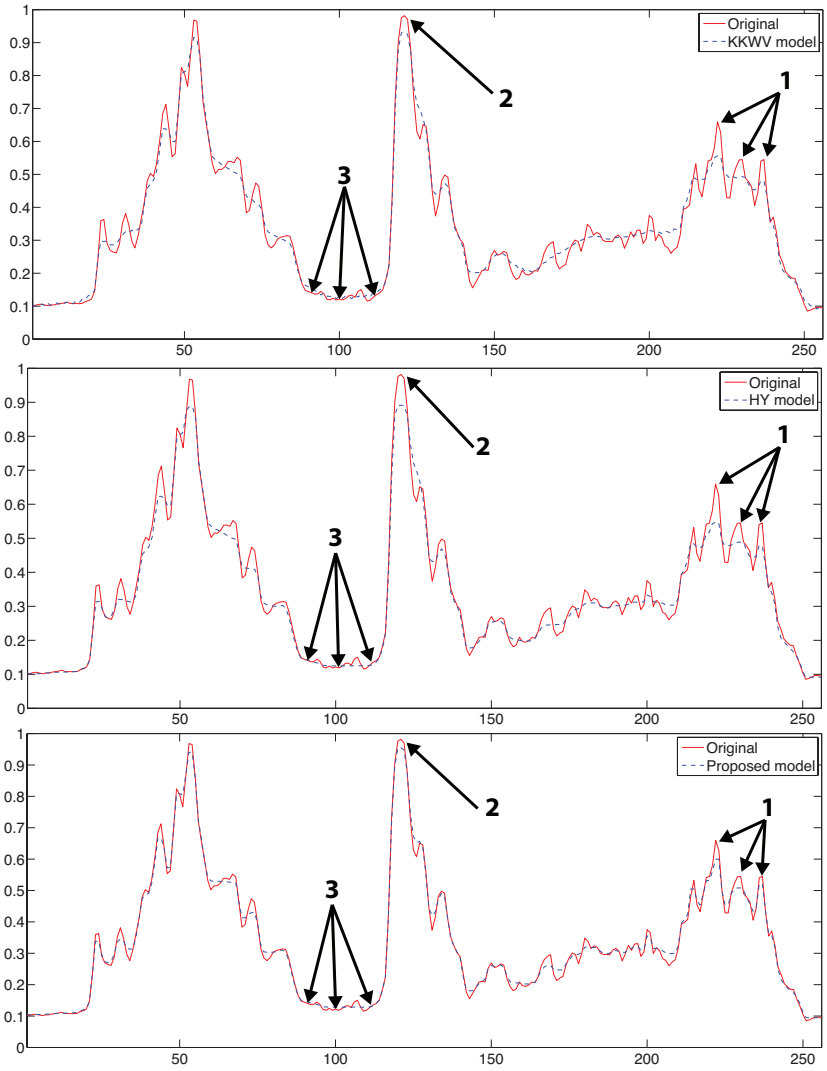


Fig. 3.10. Corresponding signals of one column 128 of the baby image as shown in Fig. 3.7. Top row: recovered signal by KKWV model; middle row: recovered signal by HY model; bottom row: recovered by the proposed model.

3.2. Performance tests among Methods 1–3

To illustrate the performance of Methods 1–3, the three test images with the 256×256 pixels as shown in Fig. 3.1 are used here and distorted by a SN with $\sigma_n^2 = 10$. All methods starts with the same initial solution $u^{[0]} = z$. In this section we will use the same stopping rule as shown in (3.3).

By experiments we note first that the set of the FP parameters $(\omega, \max_{\mathrm{SOR}}) = (1.2, 2)$ for GFP method, $(\omega, \max_{\mathrm{FP}}) = (1.3, 2)$ for LFP method, and $(\omega, \max_{\mathrm{FP}}) = (0.9, 2)$ for CGLFP method, is recommended in leading to fast convergence and visually pleasing restoration results for Method 1. We also note that setting $\max_{\mathrm{c}} = 1000$ and $\max_{\mathrm{F}} = 100$ in our coarse-to-fine framework for the FP methods outperforms other choices of these two parameters in terms of the quality of the restored images and the amount of run time. Next, we note that applying

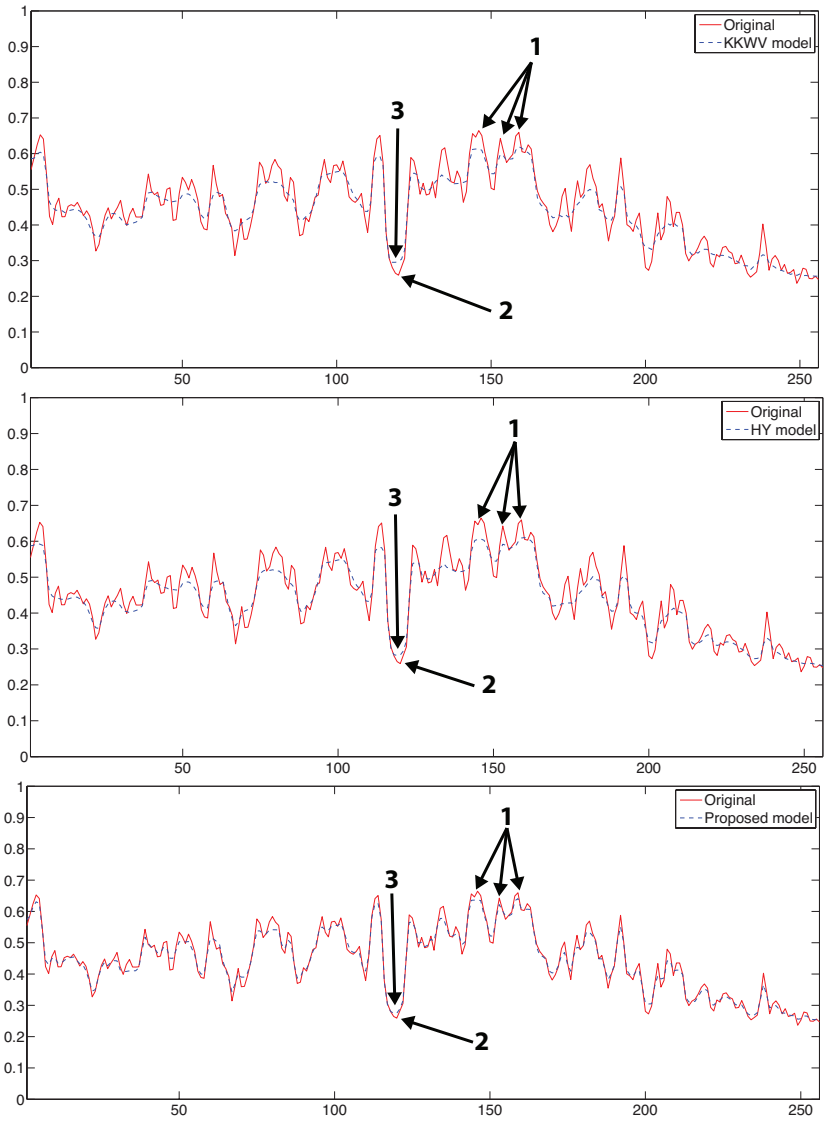


Fig. 3.11. Corresponding signals of one column 128 of the liver image as shown in Fig. 3.8. Top row: recovered signal by KKWV model; middle row: recovered signal by HY model; bottom row: recovered by the proposed model.

this set of the FP parameters with Method 2 leads to convergent MG methods for all different sizes of images, where $\nu_1 = \nu_2 = 7$. Finally we note that using simple penalty parameter $\theta = 1$ in Method 3 usually works out well.

As can be seen, Table 3.4 shows first that GFP method performs slightly better than other two FP methods in terms of the restoration quality, while as the image size increases GFP method is much more better than the two other FP methods in terms of the amount of run time. Thus, GFP method can be viewed as the optimal choice of Methods 1 and 2 in being the efficient inner FP solver. Second, Method 3 is more efficient than Method 1 in delivering higher quality of the restoration, while Method 3 is more expensive than Method 1. Third, Method 3 usually takes less than 13 steps. So, the proposed numerical scheme for Method 3 is

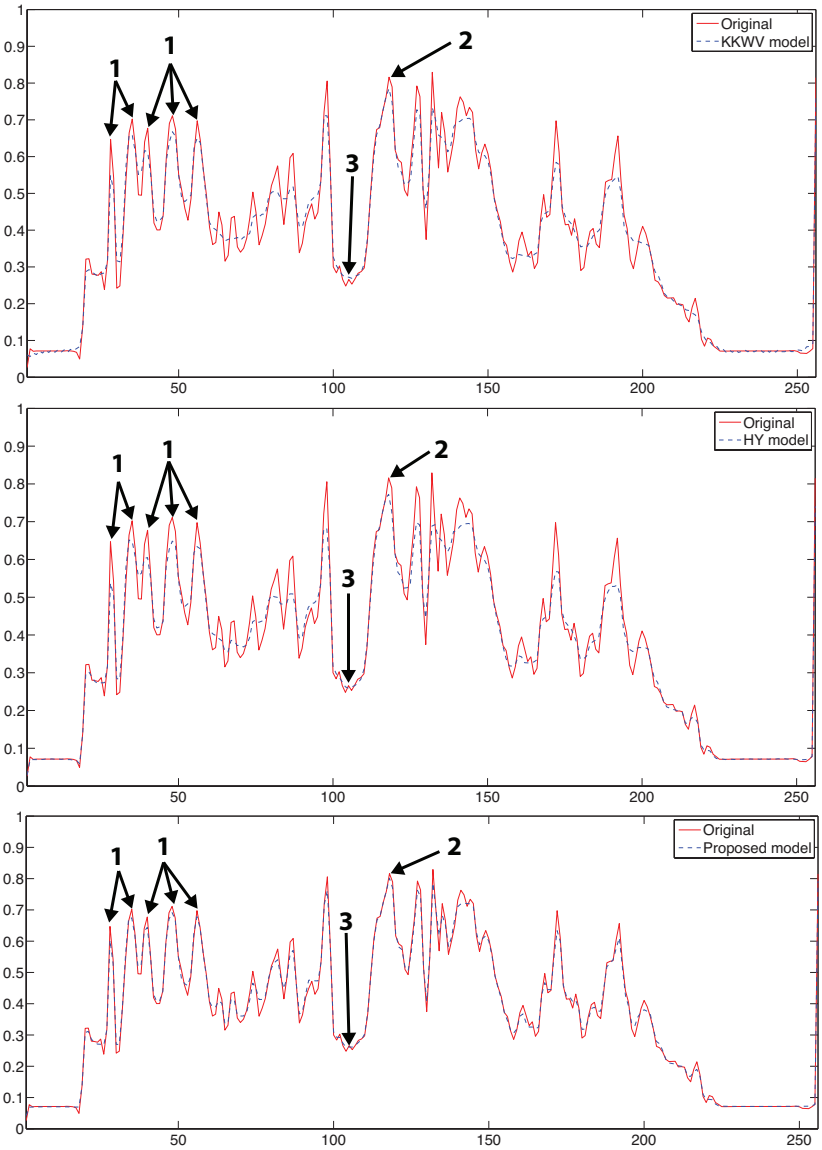


Fig. 3.12. Corresponding signals of one column 128 of the kidney image as shown in Fig. 3.9. Top row: recovered signal by KKWV model; middle row: recovered signal by HY model; bottom row: recovered by the proposed model.

quite efficient. Next, we can clearly see that Method 2 is the fastest and best way in producing the good quality of the restored images. In particular, as the image size increases Method 2 is much more efficient than Method 1 and Method 3. So, Method 2 should be recommended to solve our proposed SN removal model (2.4). Finally, the quality of the restored results by Methods 1–3 looks slightly different and almost identical in several cases, while the amount of run time used by these methods is clearly different. Therefore, this implies first that the different numerical methods for our proposed SM removal model do not significantly influence on the final image restoration results. Second a MG method like Method 3 has a major impact on runtime performance compared with that of a non-MG method.

Table 3.4: Performance tests ... (in seconds).

Image	Parameter	Method 1			Method 2			Method 3
		GFP	LFP	CGLFP	MG-GFP	MG-LFP	MG-CGLFP	SB
		N/CPUs/PSNR	N/CPUs/PSNR	N/CPUs/PSNR	N/CPUs/PSNR	N/CPUs/PSNR	N/CPUs/PSNR	N/CPUs/PSNR
Ring								
$h = \frac{1}{256}$	$\alpha_1 = 10^{-4}$	10/0.67/38.46	8/1.07/38.14	10/0.96/38.35	**4/0.27/39.30**	4/0.64/39.28	4/0.51/39.28	10/2.44/39.25
$h = \frac{1}{512}$	$\alpha_2 = 10^{-3}$	10/2.77/40.12	8/4.38/40.11	10/3.97/40.23	**4/0.92/41.32**	4/2.62/41.32	4/2.09/41.32	9/10.33/40.94
$h = \frac{1}{1024}$	$\epsilon = 10^{-2}$	10/11.82/42.05	8/18.66/42.01	10/16.89/42.05	**3/3.37/43.99**	4/11.12/43.99	4/8.89/43.99	9/41.53/42.93
$h = \frac{1}{2048}$		10/48.77/45.34	8/77.54/45.32	10/69.67/45.32	**3/14.78/46.93**	4/45.84/46.88	4/36.67/46.88	9/178.65/46.53
$h = \frac{1}{4096}$		10/202.62/48.02	8/319.14/48.00	10/289.42/48.00	**3/63.94/49.84**	4/190.44/49.79	4/152.35/49.79	9/733.47/49.32
Lena								
$h = \frac{1}{256}$	$\alpha_1 = 10^{-4}$	10/0.69/29.12	8/1.10/29.07	10/0.98/29.12	**2/0.16/29.57**	2/0.38/29.41	2/0.28/29.57	12/3.02/29.37
$h = \frac{1}{512}$	$\alpha_2 = 5\alpha_1$	10/2.42/29.88	8/4.53/29.75	10/3.43/29.88	**2/0.49/30.39**	2/1.35/30.23	2/1.05/30.34	11/12.11/30.10
$h = \frac{1}{1024}$	$\epsilon = 10^{-2}$	10/11.04/33.67	8/18.73/33.66	10/16.06/33.66	**2/2.17/34.31**	2/5.53/34.15	2/4.48/34.29	10/46.34/34.03
$h = \frac{1}{2048}$		10/45.47/36.10	8/78.60/36.11	10/70.06/36.05	**2/9.81/37.49**	2/22.77/37.33	2/18.59/37.40	10/200.05/37.30
$h = \frac{1}{4096}$		10/200.37/39.83	8/321.13/39.80	10/287.83/39.77	**2/42.63/41.06**	2/94.20/40.90	2/75.55/41.02	9/730.65/41.00
Pepper								
$h = \frac{1}{256}$	$\alpha_1 = 10^{-4}$	10/0.66/32.31	8/1.24/32.20	10/0.91/32.30	**2/0.16/32.73**	2/0.37/32.71	2/0.29/32.70	11/2.73/32.66
$h = \frac{1}{512}$	$\alpha_2 = 5\alpha_1$	10/2.56/33.83	8/4.42/33.81	10/3.94/33.82	**2/0.48/34.36**	2/1.34/34.34	2/1.07/34.30	10/11.55/34.23
$h = \frac{1}{1024}$	$\epsilon = 10^{-2}$	10/11.34/37.66	8/18.85/37.62	10/16.55/37.63	**2/2.20/38.27**	2/5.55/38.15	2/4.44/38.12	10/46.19/38.08
$h = \frac{1}{2048}$		10/47.98/39.82	8/77.12/39.81	10/69.42/39.82	**2/9.71/40.46**	2/22.74/40.44	2/18.19/40.43	10/198.22/40.39
$h = \frac{1}{4096}$		10/201.18/44.20	8/319.32/44.15	10/288.31/44.14	**2/42.34/44.61**	2/96.31/44.56	2/77.04/44.54	10/811.55/44.48

3.3. Further performance test on Method 2

As shown in Table 3.4, GFP method is recommendable as a potential smoother for Method 2 or our FAS-NMG method. Thus, in the following tests we designed our experiments to investigate its convergence with regard to parameter changes.

3.3.1. Signal−dependent tests

Table 3.5 shows the robustness of our SN removal model and the proposed FAS-NMG method in Algorithm 4 for different noise levels.

Here we tested our proposed FAS-NMG method with the GFP method u2sing the parameters $h = 1/256$, $\epsilon = 10^{-2}$, $\nu_1 = \nu_2 = 7$, $\omega = 1.2$, $\max_{\text{SOR}} = 2$ and $\overrightarrow{\varepsilon} = (20, 10^{-4}, 10^{-3}, 10^{-4}, 10^{-3})$ for all tests on the image ring.

As can be clearly seen, although convergence is slower for noisier images, the PSNR and ReErr values show that the restored images would come with good quality. Moreover, the number of MG steps does not increase very much.

Table 3.5: Restoration results for signal−dependent tests of Algorithm 4 (Proposed FAS-NMG method) with the GFP method for the image ring shown in Fig. 3.1. Note the the regularization parameters for all tests were well-selected.

σ_n^2	1	5	10	15	20
M	3	4	4	5	5
PSNR	57.19	45.75	39.30	36.11	34.20
ReErr	1.81×10^{-5}	2.52×10^{-4}	1.11×10^{-3}	2.32×10^{-3}	3.60×10^{-3}

3.3.2. α_1/α_2−dependent test

Next we evaluate to show how our FAS-NMG method in Algorithm 4 is affected with varying $\bar{\alpha} = \alpha_1/\alpha_2$.

To this end, the MG algorithm based on the GFP method was tested on the image ring with the results shown in Table 3.6. Here the following parameters are used: $\epsilon = 10^{-2}$, $h = 1/256$,

Table 3.6: Restoration results for α_1/α_2-dependent tests of Algorithm 4 (Proposed FAS-NMG method) with the GFP method for the image ring shown in Fig. 3.1. Here "n.c." means that our proposed MG method did not converge.

$\frac{\alpha_1}{\alpha_2}$	$\frac{10^{-4}}{10^{-2}}$	$\frac{10^{-4}}{10^{-3}}$	$\frac{10^{-4}}{0.5\times10^{-3}}$	$\frac{10^{-4}}{0.2\times10^{-3}}$	$\frac{10^{-4}}{10^{-4}}$	$\frac{10^{-4}}{0.5\times10^{-4}}$	$\frac{10^{-4}}{0.2\times10^{-4}}$	$\frac{10^{-4}}{10^{-5}}$	$\frac{10^{-4}}{10^{-6}}$
$\bar{\alpha}$	0.01	**0.1**	**0.2**	**0.5**	**1**	**2**	5	10	100
M	15	**4**	**4**	**4**	**3**	**3**	n.c.	n.c.	n.c.
PSNR	28.68	**39.30**	**39.22**	**34.98**	**31.38**	**29.69**	*	*	*
ReErr	1.28×10^{-2}	$\mathbf{1.11\times10^{-3}}$	$\mathbf{8.90\times10^{-4}}$	$\mathbf{3.00\times10^{-3}}$	$\mathbf{6.89\times10^{-3}}$	$\mathbf{1.01\times10^{-2}}$	*	*	*

Table 3.7: Restoration results for $\epsilon-$dependent tests of Algorithm 4 (Proposed FAS-NMG method) with the GFP method for the image ring shown in Fig. 3.1. Clearly, $\epsilon = 10^{-2}$ is enough to remove this kind of SN with the good PSNR results in a few MG steps.

ϵ	5×10^{-1}	2.5×10^{-1}	10^{-1}	$\mathbf{10^{-2}}$	10^{-3}	10^{-4}
M	4	4	4	**4**	5	5
PSNR	38.08	38.40	38.75	**39.30**	39.51	39.57
ReErr	1.47×10^{-3}	1.37×10^{-3}	1.26×10^{-3}	$\mathbf{1.11\times10^{-3}}$	1.05×10^{-3}	1.04×10^{-3}

$\nu_1 = \nu_2 = 7$, $\omega = 1.2$, $\max_{\text{SOR}} = 2$, $\sigma_n^2 = 10$ and $\overrightarrow{\epsilon} = (20, 10^{-4}, 10^{-3}, 10^{-4}, 10^{-3})$ for all experiments and $\bar{\alpha}$ is varied from 0.01 to 100.

As can be seen from Table 3.6, decreasing the values of α_2 leads to the best restoration result at $\bar{\alpha} = 0.2$. Moreover, we can also see that large $\bar{\alpha}$ is not needed as small ones give better results, typically $\bar{\alpha} = 0.1, 0.2, 0.5, 1$, and 2. We note that the process to select the optimal value of $\bar{\alpha}$ is a separate but important issue because it is in general unknown a priori and it significantly affects on the qualities of restored images as well as the MG performance.

3.3.3. $\epsilon-$dependent test

As is well known, the qualities of restoration results and the performance of the MG techniques in solving the nonlinear systems related to the TV regularization method are affected significantly by the values of ϵ.

Here our aim for this test is to see how the proposed variational model and FAS-NMG algorithm are affected with varying the values of ϵ. To see this, the MG algorithm based on the GFP method was tested on the image ring using the following parameters $h = 1/256$, $\alpha_1 = 10^{-4}$, $\alpha_2 = 10^{-3}$, $\sigma_n^2 = 10$, $\nu_1 = \nu_2 = 7$, $\omega = 1.2$, $\max_{\text{SOR}} = 2$ and $\overrightarrow{\epsilon} = (20, 10^{-4}, 10^{-3}, 10^{-4}, 10^{-3})$ for all experiments and ϵ is varied from 10^{-4} to 5×10^{-1}.

As can be seen, Table 3.7 shows that our FAS-NMG method with GFP method converges within a few MG steps for $\epsilon \in [10^{-4}, 5\times10^{-1}]$. Theoretically, ϵ should be selected to be very small as much as possible. However our experimental results indicate that very small ϵ is not necessary and not recommendable. As clearly shown in Table 3.4, $\epsilon = 10^{-2}$ is enough to remove this kind of noise with the good PSNR results. We note GFP method can lead to better MG convergence for the case $\epsilon < 10^{-2}$ when the number of pre- and post-smoothing steps ν_1 and ν_2 are increased.

4. Conclusion

The main focus of this work is to present an improved variational model for removing SN from real US images based on the TV and WTV regularization methods. The existence and uniqueness of the solution for the improved variational model have also discussed. In order

to solve the associated EL equation, we have proposed several numerical methods. Numerical tests confirmed that the model delivers better restoration results than the competing models and very importantly its MG solution is fast, robust and reliable in providing visually pleasing restoration results.

Acknowledgments. The authors express their thanks to the referees for a number of useful suggestions. The first author's work was partially supported by the Thailand Research Fund Grant MRG6080169.

References

[1] F. Adamo, G. Andria, F. Attivissimo, A.M.L. Lanzolla, and M. Spadavecchia, A comparative study on mother wavelet selection in ultrasound image denoising, *Measurement*, **46** (2013), 2447-2456.

[2] G. Aubert and J.F. Aujol, A variational approach to removing multiplicative noise, *SIAM J. Appl. Math.*, **68**:4 (2008), 925-946.

[3] N. Badshah and K. Chen, Multigrid method for the Chan-Vese model in variational segmentation, *Commun. Comput. Phys.*, **4**:2 (2008), 294-316.

[4] J. Bioucas-Dias and M.A.T. Figueiredo, Multiplicative noise removal using variable splitting and constrained optimization, *IEEE Trans. Image Process.*, **19**:7 (2010), 1720-1730.

[5] K. Bredies, K. Kunisch and T. Pock, Total Generalized Variation, *SIAM J. Imaging Sci.*, **3**:3 (2010), 492-526.

[6] H. Brezis, Operateures Maximaux Monotone, North-Holland, Amsterdam, 1993.

[7] W.L. Briggs, V.E. Henson, and S.F. McCormick, A Multigrid Tutorial (2nd Edition), SIAM Publications, Philadelphia, USA, 2000.

[8] C. Brito-Loeza and K. Chen, Multigrid method for a modified curvature driven diffusion model for image inpainting, *J. Comput. Math.*, **26**:6 (2008), 856-875.

[9] C. Brito-Loeza and K. Chen, Fast numerical algorithms for Euler's Elastica digital inpainting model, *Int. J. Mod. Math.*, **5**:2 (2010), 157-182.

[10] C. Brito-Loeza and K. Chen, Multigrid algorithm for high order denoising, *SIAM J.Imaging Sci.*, **3**:3 (2010), 363-389.

[11] 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), 1518-1527.

[12] T.F. Chan and K. Chen, On a nonlinear multigrid algorithm with primal relaxation for the image total variation minimization *Numer. Algorithms*, **41** (2006), 387-411.

[13] T.F. Chan and K. Chen, An optimization-based multilevel algorithm for total variation image denoising, *Multiscale Model. Simul.*, **5**:2 (2006), 615-645.

[14] T.F. Chan, K. Chen, and X.C. Tai, Nonlinear multilevel scheme for solving the total variation image minimization problem, in *Proceedings of the International Conference on PDE-Based Image Processing and Related Inverse Problems Series: Mathematics and Visualization*, X.-C. Tai, K.-A. Lie, T.F. Chan and S. Osher (Eds.), Springer Verlag, 2006, pp. 1-27.

[15] T.F. Chan and P. Mulet, On the convergence of the lagged diffusivity fixed point method in total variation image restoration, *SIAM J. Numer. Analysis*, **36**:2 (2007), 354-367.

[16] K. Chen and X.C. Tai, A nonlinear multigrid method for total variation minimization from image restoration, *J. Sci. Comput.*, **32**:2 (2007), 115-138.

[17] Q. Chen, P. Montesinos, Q.S. Sun, and D.S. Xia, Ramp preserving Perona-Malik model, *Signal Processing*, **90** (2010), 1963-1975.

[18] N. Chumchob, Vectorial total variation-based regularization for variational image registration, *IEEE Trans. Image Process.*, **22**:11 (2013), 4551-4559.

[19] N. Chumchob and K. Chen, A variational approach for discontinuity-preserving image registration, *East-West Journal of Mathematics*, (2010), 266-282.

[20] N. Chumchob and K. Chen, A robust multigrid approach for variational image registration models, *J. Comput. Appl. Math.*, **236** (2011), 653-674.

[21] N. Chumchob and K. Chen, An improved variational image registration model and a fast algorithm for its numerical approximation, *Numer. Methods Partial Differential Equations*, **28**:6 (2012), 1966-1995.

[22] N. Chumchob, K. Chen, and C. Brito-Loeza, A fourth order variational image registration model and its fast multigrid algorithm, *Multiscale Model. Simul.*, **9** (2011), 89-128.

[23] N. Chumchob, K. Chen, and C. Brito-Loeza, A new variational model for removal of combined additive and multiplicative noise and a fast algorithm for its numerical approximation, *Int. J. Comput. Math.*, **90**:1 (2013), 140-161.

[24] V. Dutt and J. Greenleaf, Adaptive speckle reduction filter for log-compressed B-scan images, *IEEE Trans. Med. Imag*, **15**:6 (1996), 802-813.

[25] S. Geman and D. Geman, Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images, *IEEE Trans. Pattern Anal. Mach Intel.*, **6**:6 (1984), 721-741.

[26] T. Goldstein and S. Osher, The Split Bregman Method for L1-Regularized Problems, *SIAM J. Imaging Sci.*, **2**:2 (2009), 323-343.

[27] E. Giusti, Minimal Surfaces and Functions of Bounded Variation, Monographs in Mathematics, Vol. 80, Birkhauser, Boston, 1984.

[28] E. Haber and J. Modersitzki, A multilevel method for image registration, *SIAM J. Sci. Comput.*, **27**:5 (2006), 1594-1607.

[29] X.H. Hao, S.K. Gao, and X.R. Gao, A novel multiscale nonlinear thresholding method for ultrasonic speckle suppressing, *IEEE Trans. Med. Imaging.*, **18**:9 (1999), 787-794.

[30] P. Hellier, C. Kervrann, and C. Barillot, Nonlocal means-based speckle filtering for ultrasound images, *IEEE Trans. Image Process.*, **18**:10 (2009), 2221-2229.

[31] L.L. Huang, L. Xiao, and Z.H. Wei, Multiplicative noise removal via a novel variational model, *EURASIP J. Image Vide.*, **250768** (2010), DOI:10.1155/2010/250768.

[32] Y. Huang, M. Ng, and Y. Wen, A new total variation method for multiplicative noise removal, *SIAM J. Imaging Sci.*, **2**:1 (2009), 20-40.

[33] Y. Han, X.C. Feng, G. Baciu, and W.W. Wang, Nonconvex sparse regularizer based speckle noise removal, *Pattern Recogn.*, **46** (2013), 989-1001.

[34] J. Huang and X. Yang, Fast reduction of speckle noise in real ultrasound images, *Signal Processing*, **93** (2013), 684-694.

[35] Z. Jin and X. Yang, Analysis of a new variational model for multiplicative noise removal, *J. Math. Anal. Appl.*, **362** (2010), 415-426.

[36] Z. Jin and X. Yang, A variational model to remove the multiplicative noise in ultrasound images, *J. Math Imaging Vis.*, **39** (2011), 62-74.

[37] J. Koo and S.B. Oark, Speckle reduction with edge preservation in medical ultrasonic images using a homogeneous region growing mean filter (HRGMF), *Ultrasonic Imaging*, **13** (1990), 629-639.

[38] J. Koo and S.B. Oark, Despeckling of medical ultrasound images using Daubechies complex wavelet transform, *Signal Processing*, **90** (2010), 428-439.

[39] P. Kornprobst, R. Deriche, and G. Aubert, Image sequence analysis via partial differential equations, *J. Math. Imaging Vis.*, **11**:1 (1999), 5-26.

[40] K. Krissian, R. Kikinis, C.F. Westin, and K. Vosburgh, Speckle constrained filtering of ultrasound images, *IEEE Comput. Vis. Pattern Recogn.*, **15** (2005), 547-552.

[41] T. Loupas, W. McDicken, and P. Allan, An adaptive weighted median filter for speckle suppression in medical ultrasound images, *IEEE Trans. Circuits Syst.*, **36**:1 (1989), 129-135.

[42] O.V. Michailovich and A. Tannenbaum, An accurate and efficient Bayesian model for automatic segmentation of brain MRI, *IEEE Trans. Ultrason., Ferroelect., Freq. Contr.*, **53**:1 (2006), 64-78.

[43] S. Osher, M. Burger, D.Goldfarb, J. Xu, and W.T. Yin, An iterative regularization method for total variation-based image restoration, *Multiscale Model. Simul.*, **4** (2005), 460-489.

[44] L. Rudin, S. Osher, and E. Fatemi, Nonlinear total variation based noise removal algorithms, *Physica D*, **60** (1992), 259-268.

[45] L. Rudin, P. L. Lions, and S. Osher, Multiplicative denoising and deblurring: theory and algorithms, in *Geometric Level Set Methods in Imaging, Vision, and Graphics*, S. Osher and N. Paragios (Eds.), Springer, Berlin, Germany, 2003, 103-120.

[46] J. Savage and K. Chen, An improved and accelerated nonlinear multigrid method for total-variation denoising, *Int. J. Comput. Math.*, **82**:8 (2005), 1001-1015.

[47] J. Savage and K. Chen, On multigrids for solving a class of improved total variation based staircasing reduction models, in *Proceedings of the International Conference on PDE-Based Image Processing and Related Inverse Problems Series: Mathematics and Visualization*, X.C. Tai, K.A. Lie, T.F. Chan and S. Osher (Eds.), Springer Verlag, 2006, 69-94.

[48] C. F.-Schauf, S.Henn, and K. Witsch, Nonlinear multigrid methods for total variation image denoising, *Comput. Visual. Sci.*, **7** (2004), 199-206.

[49] J. Shen, On the foundations of vision modeling: I. Weber's law and Weberized TV restoration, *Physica D: Nonlinear Phenomena*, **175**:3-4 (2003), 241-251.

[50] J.N. Shi and S. Osher, A nonlinear inverse scale space method for a convex multiplicative noise model, *SIAM J. Imaging Sci.*, **1**:3 (2008), 294-321.

[51] B. Shi, L.Huang, Z.F. Pang, Fast algorithm for multiplicative noise removal, *J. Vis. Commun. Image R.*, **23** (2012), 126-133.

[52] X.C. Tai. and C. Wu, Augmented Lagrangian Method, Dual Methods and Split Bregman Iteration for ROF Model, *Scale Space and Variational Methods in Computer Vision, LNCS*, **5567**:556 (2009), 502-513.

[53] 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 Sci.*, **3**:3 (2010), 300-339.

[54] J.M. Thijssen, Ultrasonic speckle formation, analysis and processing applied to tissue characterization, *Pattern Recognition Letters*, **24** (2003), 659-675.

[55] U. Trottenberg, C. Oosterlee, and A. Schuller, Multigrid, Academic Press, 2001.

[56] T.A. Tuthill, R.H. Sperry, and K.J. Parker, Deviations from Rayleigh statistics in ultrasonic speckle, *Ultrason. Imag.*, **10** (1988), 81-90.

[57] C.R. Vogel, A multigrid method for total variation-based image denoising, in *Computation and control IV*, 1995.

[58] C.R. Vogel and M.E. Oman, Iterative methods for total variation denoising, *SIAM J. Sci. Comput.*, **17**:1 (1996), 227-238.

[59] J. Weickert, B.M. ter Haar Romeny, and M. Viergever, Efficient and reliable schemes for nonlinear diffusion filtering, *IEEE Trans. Image Process.*, **7** (1998), 398-410.

[60] R. Wienands and W. Joppich, Practical Fourier Analysis for Multigrid Method, Chapman & Hall/CRC, U.S.A., 2005.

[61] L. Xiao, L.L. Huang, and Z.H. Wei, A Weberized Total Variation Regularization-Based Image Multiplicative Noise Removal Algorithm, *EURASIP J. Adv. Signal Process.*, **490384** (2010), DOI:0.1155/2010/490384.

[62] H. Yua, J.L. Tana, and Y.Y. Wang, Ultrasound speckle reduction by a SUSAN-controlled anisotropic diffusion method, *Pattern Recogn.*, **43** (2010), 3083-3092.

A New Curvature-based Model for Image Denoising

Pornpimon Sroisangwan[a,b], Noppadol Chumchob[a,b,*]

[a] *Department of Mathematics, Faculty of Science, Silpakorn University, Nakorn Pathom 73000, Thailand*
[b] *Centre of Excellence in Mathematics, CHE, Si Ayutthaya Rd., Bangkok 10400, Thailand*

Abstract

The classical total variation (TV) model has made great successes in image denoising due to the edge-preserving property of the TV regularization. However, it is well known that the TV model suffers from the staircase effects and the loss of image details. In order to overcome these problems, we present a new curvature-based model for image denoising. To find a minimizer of our proposed model, we develop an efficient numerical algorithm in the split Bregman framework for solving the associated minimization problem. We experimentally demonstrate the effectiveness and efficiency of the proposed techniques over existing high-order variational models.

Keywords: Image restoration, mean curvature, Gaussian curvature, high-order regularization, split Bregman

2010 MSC: 93C10, 35Q94, 94A08, 68U10, 65K10, 65N55

1. Introduction

Images generated by imaging devices are frequently contaminated by noise during image acquisition and transmission processes. Thus the task of image denoising, consisting of removing electronic noise, which is a random variation of brightness or color information, from the image signal, has become a highly desirable and essential pre-processing technique in several areas of multimedia applications. Especially in medical application, for example, it has been used routinely to enhance the quality and information content in clinical images by estimating the original clean image from a recorded noisy image without losing significant features such as edges and texture details.

Over the last decades, a variety of methods have been developed to deal with the image denoising problems. Roughly speaking, these denoising methods fall into five categories: 1) filtering based methods in spatial domain [1, 2, 3, 4] and transformed domain, e.g. wavelet

*Correspondence author.
Email addresses: sroisangwan_p@silpakorn.edu (Pornpimon Sroisangwan), chumchob_n@silpakorn.edu (Noppadol Chumchob)

domain [5, 6, 7]; 2) nonlocal filtering methods [8, 9, 10]; 3) anisotropic diffusion methods [11, 12]; 4) deep convolution neural network methods [13, 14, 15, 16]; and 5) variational methods [17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31]. Among these methods, the variational methods are one of the most powerful techniques to offer high quality of denoised images. Accordingly, this paper focuses on the variational method based on high-order regularizations.

In the next subsections, we present a variational formulation and review briefly several existing variational models for image denoising problems.

1.1. Variational formulation

Let Ω be a bounded domain of $\mathbb{R}^2$ (a rectangle in practice). We consider the problem of recovering the original image $u : \Omega \to \mathbb{R}$ from a recorded noisy image $z : \Omega \to \mathbb{R}$. Generally, most of the noise obtained during acquisition and transmission of the natural images is assumed to be additive noise (AN). Thus, a recorded noisy image z can be mathematically modeled with the equation

$$z = u + \eta, \tag{1}$$

where $\eta : \Omega \to \mathbb{R}$ is the zero-mean Gaussian white noise with a given variance σ.

It is well known that the problem of recovering u from z by using the AN model (1) is an ill-posed inverse problem because the solution is very sensitive (unstable) to the noise. In this case, a widely common procedure in inverse problems is to use a regularization for computing the approximate solution that is less sensitive to the noise than the naive solution. From the variational point of view, the task of removing noise can be accomplished by solving a minimization problem such as

$$\min_u \left\{ \mathcal{J}(u) = \mathcal{D}_{\mathrm{AN}}(u, z) + \gamma \mathcal{R}(u) \right\}, \tag{2}$$

where $\mathcal{D}_{\mathrm{AN}}(u, z) = \frac{1}{2} \int_\Omega (u - z)^2 d\Omega$ is the fidelity term, which is derived from the AN model (1) and used to measure the similarity between the denoised image u and the recorded noisy image z, $\mathcal{R}(u) = \int_\Omega F_{\mathcal{R}}(x, y, u, D_1 u, D_2 u, \cdots, D_i u, \cdots) d\Omega$ is the regularization term, which is used to filter out the noise from the recorded noisy image z as well as to preserve significant features such as edges and textures of the denoised image u, and $\gamma > 0$ is the regularization parameter which compromises the fidelity term $\mathcal{D}_{\mathrm{AN}}(u, z)$ and the regularization term $\mathcal{R}(u)$. Note that $D_i u = \left[\frac{\partial^i u}{\partial x^i}, \frac{\partial^i u}{\partial y^i} \right]^\top$ and fractional derivatives of u may occur in $F_{\mathcal{R}}$.

1.2. A review of existing high-order variational models

Rudin, Osher, and Fatemi [32] proposed one of the most popular variational models. Their variational model, also known as the total variation (TV) model, is given by

$$\min_u \left\{ \mathcal{J}^{\mathrm{TV}}(u) = \mathcal{D}_{\mathrm{AN}}(u,z) + \gamma \mathcal{R}^{\mathrm{TV}}(u) \right\}, \tag{3}$$

involving the TV regularization $\mathcal{R}^{\mathrm{TV}}(u) = \int_\Omega |\nabla u| \, d\Omega$. Although TV regularization is effective in preserving sharp edges while removing unwanted oscillations and noise, it has some undesired effects. In particular, it transforms smooth signals or signals that are not necessarily piecewise constant into piecewise constants. This phenomenon is known as staircase effect. From a practical point of view, staircase solutions of (3) fail to satisfy the evaluation of visual quality, and they can develop false edges that do not exist in the true image. The main challenge is thus to deal with the staircase effect while preserving image sharpness.

To overcome the drawback of the TV regularization, several high-order regularizations have been proposed so far in the literature to yield better quality of denoised images. These high-order regularizations usually involve second-order differential operators in dealing with the staircase effect because piecewise-vanishing second-order derivatives yield piecewise-linear solutions that better fit than those by the TV regularization for smooth intensity changes. To the best of our knowledge, there are two main classes of high-order regularizations for image denoising problems. The first class combines a second-order regularization with the TV regularization. The second class employs a second-order regularization in a standalone way. Below, we present a bird's eye view of specific variational models involving the high-order regularizations.

You and Kaveh [17] proposed the total Laplace (TL) model

$$\min_u \left\{ \mathcal{J}^{\mathrm{TL}}_{\mathrm{AN}}(u) = \mathcal{D}_{\mathrm{AN}}(u,z) + \gamma \mathcal{R}^{\mathrm{TL}}(u) \right\} \tag{4}$$

with the TL regularization $\mathcal{R}^{\mathrm{TL}}(u) = \int_\Omega |\triangle u| d\Omega = \int_\Omega \sqrt{|u_{xx}|^2 + |u_{yy}|^2} d\Omega$, where $\triangle u$ denotes the Laplacian operator. Scherzer [18], Lysaker et al. [19], Hinterberger and Scherzer [20], Bergounioux and Piffet [21] and Lai et al. [22] considered the bounded Hessian (BH) model

$$\min_u \left\{ \mathcal{J}^{\mathrm{BH}}_{\mathrm{AN}}(u) = \mathcal{D}_{\mathrm{AN}}(u,z) + \gamma \mathcal{R}^{\mathrm{BH}}(u) \right\} \tag{5}$$

with the BH regularization $\mathcal{R}^{\mathrm{BH}}(u) = \int_\Omega |\nabla^2 u| d\Omega$, where

$$\nabla^2 u = \begin{pmatrix} u_{xx} & u_{yx} \\ u_{xy} & u_{yy} \end{pmatrix}$$

represents the Hessian matrix of u and $|\nabla^2 u| = \sqrt{u_{xx}^2 + u_{yx}^2 + u_{xy}^2 + u_{yy}^2}$. Chan et al. [23] introduced the CEP-L^2 model

$$\min_{u_1,u_2} \left\{ \mathcal{J}^{\text{CEP-L}^2}(u_1,u_2) = \bar{\mathcal{D}}_{\text{AN}}(u_1,u_2,z) + \mathcal{R}^{\text{CEP-L}^2}(u_1,u_2) \right\} \tag{6}$$

with the CEP-L^2 regularization $\mathcal{R}^{\text{CEP-L}^2_{\text{AN}}}(u_1,u_2) = \gamma_1 \mathcal{R}^{\text{TV}}(u_1) + \gamma_2 \mathcal{R}^{\text{TL}}(u_2)$ where $\gamma_1 > 0$ and $\gamma_2 > 0$ are the weighting parameters that balance the first- and second-order derivative terms. Note that the noisy image z is decomposed into three parts, i.e. $z = u_1 + u_2 + \eta$ and the associated fidelity term is given by

$$\bar{\mathcal{D}}_{\text{AN}}(u_1,u_2,z) = \frac{1}{2} \int_\Omega (z - u_1 - u_2)^2 \, d\Omega.$$

Zheng et al. [24], Wang et al. [25] and Chan et al. [26] considered the TVL model

$$\min_u \left\{ \mathcal{J}_{\text{AN}}^{\text{TVL}}(u) = \mathcal{D}_{\text{AN}}(u,z) + \mathcal{R}^{\text{TVL}}(u) \right\} \tag{7}$$

with the TVL regularization $\mathcal{R}^{\text{TVL}}(u) = \gamma_1 \mathcal{R}^{\text{TV}}(u) + \gamma_2 \mathcal{R}^{\text{TL}}(u)$. Papafitsoros and Schönlieb [27] introduced the TVBH model

$$\min_u \left\{ \mathcal{J}_{\text{AN}}^{\text{TVBH}}(u) = \mathcal{D}_{\text{AN}}(u,z) + \mathcal{R}^{\text{TVBH}}(u) \right\} \tag{8}$$

with the TVBH regularization $\mathcal{R}^{\text{TVBH}}(u) = \gamma_1 \mathcal{R}^{\text{TV}}(u) + \gamma_2 \mathcal{R}^{\text{BH}}(u)$. Chambolle and Lions [28] proposed the INFCON model

$$\min_{u_1,u_2} \left\{ \mathcal{J}_{\text{AN}}^{\text{TVBH}}(u_1,u_2) = \bar{\mathcal{D}}_{\text{AN}}(u_1,u_2,z) + \mathcal{R}^{\text{TVBH}}(u_1,u_2) \right\}. \tag{9}$$

with the INFCON regularization $\mathcal{R}^{\text{TVBH}}(u_1,u_2) = \gamma_1 \mathcal{R}^{\text{TV}}(u_1) + \gamma_2 \mathcal{R}^{\text{BH}}(u_2)$ deriving from an inf-convolution of two convex regularizations. Bredies and et al. [29] proposed the total generalized variation (TGV) model

$$\min_{u,\tilde{\boldsymbol{p}}} \left\{ \mathcal{J}_{\text{AN}}^{\text{TGV}}(u,\tilde{\boldsymbol{p}}) = \mathcal{D}_{\text{AN}}(u,z) + \mathcal{R}^{\text{TGV}}(u,\tilde{\boldsymbol{p}}) \right\} \tag{10}$$

with the TGV regularization

$$\mathcal{R}^{\text{TGV}}(u,\tilde{\boldsymbol{p}}) = \gamma_1 \int_\Omega |\nabla u - \tilde{\boldsymbol{p}})| \, d\Omega + \gamma_2 \int_\Omega |\epsilon(\tilde{\boldsymbol{p}})| \, d\Omega,$$

where $\epsilon(\tilde{\boldsymbol{p}})$ is the symmetrized derivatives defined as

$$\epsilon(\tilde{\boldsymbol{p}}) = \begin{pmatrix} \tilde{p}_{1_x} & \frac{\tilde{p}_{1_y} + \tilde{p}_{2_x}}{2} \\ \frac{\tilde{p}_{1_y} \tilde{p}_{2_x}}{2} & \tilde{p}_{2_y} \end{pmatrix}.$$

Note that the minimum of (11) is taken over all gradients of the deformation field $\tilde{\boldsymbol{p}} = (\tilde{p}_1, \tilde{p}_2)$ on image surface Ω. Here $\tilde{p}_x = \partial_x \tilde{p}_l$ and $\tilde{p}_y = \partial_y \tilde{p}_l$ for $l = 1, 2$. Zhu and Chan [30] proposed the mean curvature (MC) model

$$\min_u \left\{ \mathcal{J}_{\mathrm{AN}}^{\mathrm{MC}}(u) = \mathcal{D}_{\mathrm{AN}}(u, z) + \gamma \mathcal{R}^{\mathrm{MC}}(u) \right\}, \tag{11}$$

with the MC regularization $\mathcal{R}^{\mathrm{MC}}(u) = \int_\Omega |\kappa_{\mathrm{M}}(u)| \, d\Omega$ where

$$\kappa_{\mathrm{M}}(u) = \nabla \cdot \left(\frac{\nabla u}{\sqrt{1 + |\nabla u|^2}} \right) \tag{12}$$

is the mean curvature of image surface $\phi(x, y, z) = u(x, y) - z = 0$. This model tries to fit the given noisy image surface $(x, y, z(x, y))$ with a surface $(x, y, u(x, y))$ that bears small magnitude of mean curvature. As demonstrated in [30], the MC model is able to sweep noise while keeping object edges, and it also ameliorates the staircase effect. More importantly, the MC model is also capable of preserving image contrasts as well as geometry of object shapes, especially object corners. Brito-Loeza and Chen [31] proposed the Gaussian curvature (GC) model

$$\min_u \left\{ \mathcal{J}_{\mathrm{AN}}^{\mathrm{GC}}(u) = \mathcal{D}_{\mathrm{AN}}(u, z) + \gamma \mathcal{R}^{\mathrm{GC}}(u) \right\} \tag{13}$$

with the GC regularization $\mathcal{R}^{\mathrm{GC}}(u) = \int_\Omega |\kappa_{\mathrm{G}}(u)| \, d\Omega$ where

$$\kappa_{\mathrm{G}}(u) = \frac{u_{xx} u_{yy} - u_{xy} u_{yx}}{(1 + |\nabla u|^2)^2} \tag{14}$$

is the Gaussian curvature of image surface $\phi(x, y, z) = u(x, y) - z = 0$. This model is able to preserve image contrast, edges, and corners such as the MC model does.

Non-surprisingly, the choice of the regularization significantly affects the quality of the denoised image. In particular, the high-order regularizations mentioned above are better than the TV one in reducing staircase effects and achieving good trade-off between noise removal and edge preservation. Some comparative results of high-order regularizations can be seen in [33, 34, 31].

1.3. Contribution

The main contribution of this paper is to propose a new variational model for image denoising. Our new model is designed to utilize the favorable denoising properties of the high-order regularization deriving from the sum of squared principal curvatures of the image surface. As a result, our variational model is difficult to solve by standard techniques such as gradient descent methods. To simplify the associated minimization problem, we propose to use the split Bregman (SB) method. The main idea is to introduce several auxiliary variables

for converting the complex operation into an alternating iterative process by several simple operations, which are easy to implement and have high computational efficiency. We note that the new variational model has several advantages. Firstly, it is more effective than existing high-order variational models mentioned above to remove noise and preserve edge. Secondly, as we shall demonstrate, our denoising algorithm to be introduced in Section 3 is able to converge in a reasonable number of iterations. Finally, we found what we expected in numerical experiments on real noisy images that it is able to deliver visually pleasing results.

The remainder of this paper is organized as follows. In Section 2, we introduce a new curvature-based regularization. Then, our new high-order variational model for image denoising proposed. The denoising algorithm using the SB iteration is introduced in Section 3. Experiments results from synthetic and real images are given in Section 4. In Section 5, the concluding remarks are drawn.

2. The proposed model

Inspired by the attractive property of the high-order regularizations mentioned in Section 1.2 for dealing with the staircase effects from the noise reduction process, we propose our new high-order variational model for image denoising as follows.

Let $\kappa_1(u) = \kappa_1(\mathbf{x}, u(\mathbf{x}))$ and $\kappa_2(u) = \kappa_2(\mathbf{x}, u(\mathbf{x}))$ be the principal curvatures of the image surface induced by a given image $u = u(\mathbf{x})$ through the mapping $(\mathbf{x}, u(\mathbf{x}))$. According to theory of surfaces, the mean curvature $\kappa_{\mathrm{M}}(u)$ in (12) and the Gaussian curvature $\kappa_{\mathrm{G}}(u)$ in (14) are related to the principal curvatures $\kappa_1(u)$ and $\kappa_1(u)$ by

$$\kappa_{\mathrm{M}}(u) = \kappa_1(u) + \kappa_2(u) \text{ and } \kappa_{\mathrm{G}}(u) = \kappa_1(u)\kappa_2(u).$$

Therefore, the sum of the squares of the principal curvatures of the image surface are related to the mean and Gaussian curvatures as follows.

$$\begin{aligned}
\kappa_1^2(u) + \kappa_2^2(u) &= (\kappa_1(u) + \kappa_2(u))^2 - 2\kappa_1(u)\kappa_2(u) \\
&= \kappa_{\mathrm{M}}^2(u) - 2\kappa_{\mathrm{G}}(u) \\
&= \left(\frac{(1+u_x^2)u_{yy} - 2u_x u_y u_{xy} + (1+u_y^2)u_{xx}}{(1+|\nabla u|^2)^{3/2}}\right)^2 - 2\frac{u_{xx}u_{yy} - u_{xy}u_{yx}}{(1+|\nabla u|^2)^2}. \qquad (15)
\end{aligned}$$

We note that the authors in [30] use

$$\kappa_{\mathrm{M}}(u) = \frac{\kappa_1(u) + \kappa_2(u)}{2} = \frac{(1+u_x^2)u_{yy} - 2u_x u_y u_{xy} + (1+u_y^2)u_{xx}}{2(1+|\nabla u|^2)^{3/2}},$$

justifying the name 'mean' curvature. We also note that the sum of the squares of the principal curvatures of the image surface for the case $|\nabla u| \approx 0$ can be approximated by

$$\kappa_1^2(u) + \kappa_2^2(u) \approx \triangle^2 u - 2\left(u_{xx}u_{yy} - u_{xy}^2\right).$$

Carefully considering, we have $\kappa_{\mathrm{M}}(u_0) = 0$ and $\kappa_{\mathrm{G}}(u_0) = 0$ for piecewise constant or piecewise affine images u_0. This motivates us to propose the following functional as the new regularization

$$\mathcal{R}^{\mathrm{NewCv}}(u) = \frac{1}{2}\int_\Omega \left(\kappa_1^2(u) + \kappa_2^2(u)\right) d\Omega = \frac{1}{2}\int_\Omega \left(\kappa_{\mathrm{M}}^2(u) - 2\kappa_{\mathrm{G}}(u)\right) d\Omega. \tag{16}$$

As can be seen, the kernel of $\mathcal{R}^{\mathrm{NewCv}}$ denoted by $\ker\left(\mathcal{R}^{\mathrm{NewCv}}\right) = \left\{u : \Omega \to \mathbb{R} | \mathcal{R}^{\mathrm{New}}(u) = 0\right\}$ is given by

$$\ker\left(\mathcal{R}^{\mathrm{NewCv}}(u)\right) = \left\{u(\mathbf{x}) = \mathbf{A}\mathbf{x} + b \text{ a.e. in } \Omega \text{ for some } \mathbf{A} \in \mathbb{R}^{1\times 2},\ \mathbf{x} = (x,y)^\top \in \mathbb{R}^2,\ b \in \mathbb{R}\right\}.$$

It follows that $\mathcal{R}^{\mathrm{NewCv}}$ shares this attractive property with the MC and GC models in dealing with different image characteristics such as piecewise constant and piecewise affine images. We note that $\mathcal{R}^{\mathrm{NewCv}}$ is related to the bending energy used to measure surface fairing and the strain energy of a thin elastic plate, which has lots of applications in computer graphics [35].

In summary, the new high-order variational model with the regularization (16) is then given by

$$\min_u \left\{\mathcal{J}_{\mathrm{AN}}^{\mathrm{NewCv}}(u) = \mathcal{D}_{\mathrm{AN}}(u,z) + \gamma\mathcal{R}^{\mathrm{NewCv}}(u)\right\}. \tag{17}$$

3. Solution of the proposed model

To find the solution of the new model (17), ones may directly solve the corresponding Euler-Lagrange (EL) equation, which can be derived similarly as in [36] (see in Appendix A) and given by

$$f(u,z) + \gamma\nabla\cdot\boldsymbol{V} = 0 \tag{18}$$

subject to the natural boundary conditions

$$\kappa_{\mathrm{M}}(u) = 0, \tag{19}$$

$$-\frac{\partial\kappa_{\mathrm{M}}(u)}{\partial\boldsymbol{n}} + \frac{\nabla\kappa_{\mathrm{M}}(u)\cdot\nabla u}{(1+|\nabla u|^2)^2}\frac{\partial u}{\partial\boldsymbol{n}} = 0, \tag{20}$$

$$(u_{xy}, -u_{xx})\cdot\boldsymbol{n} = 0, \tag{21}$$

$$(-u_{yy}, u_{yx}) \cdot \boldsymbol{n} = 0, \tag{22}$$

$$((\frac{u_{yy}}{(1+|\nabla u|^2)^2})_x, -(\frac{u_{xy}}{(1+|\nabla u|^2)^2}))_x \cdot \boldsymbol{n} = 0, \tag{23}$$

$$((\frac{u_{yx}}{(1+|\nabla u|^2)^2})_y, -(\frac{u_{xx}}{(1+|\nabla u|^2)^2}))_y \cdot \boldsymbol{n} = 0, \tag{24}$$

$$\frac{4(u_{xy}u_{yx} - u_{xx}u_{yy})}{(1+|\nabla u|^2)^3}\frac{\partial u}{\partial \boldsymbol{n}} = 0. \tag{25}$$

Here $f(u,z) = (u-z)$ and $\boldsymbol{V} = \boldsymbol{V}_1 + \boldsymbol{V}_2 + \boldsymbol{V}_3 + \boldsymbol{V}_4 + \boldsymbol{V}_5$, where

$$\boldsymbol{V}_1 = \frac{1}{\sqrt{1+|\nabla u|^2}}\nabla\kappa_{\mathrm{M}}(u), \boldsymbol{V}_2 = -\frac{\nabla u \cdot \nabla\kappa_{\mathrm{M}}(u)}{(\sqrt{1+|\nabla u|^2})^3}\nabla u,$$
$$\boldsymbol{V}_3 = \frac{4(u_{xy}u_{yx} - u_{xx}u_{yy})}{(1+|\nabla u|^2)^3}\nabla u, \boldsymbol{V}_4 = ((-\frac{u_{yy}}{(1+|\nabla u|^2)^2})_x, (\frac{u_{xy}}{(1+|\nabla u|^2)^2})_x),$$
$$\boldsymbol{V}_5 = ((\frac{u_{yx}}{(1+|\nabla u|^2)^2})_y, (-\frac{u_{xx}}{(1+|\nabla u|^2)^2})_y).$$

A simple numerical technique for solving the nonlinear and fourth-order partial differential equation in (18) is to update u using the gradient descent method as follows:

$$u_t = -\left(f(u,z) + \gamma\nabla \cdot \boldsymbol{V}\right), \tag{26}$$

where $u = u(\mathbf{x}, t)$ and $t \in [0, \infty)$ is the artificial time variable. However, this technique requires a small time step and is not efficient because the time step is required to be very small for stability reasons. One way to solve it effectively is to develop a multigrid method as in [37]. Here, we consider an alternative approach in a unilevel framework.

3.1. Discretization

In order to provide the numerical algorithm for the proposed variational model (17), the finite difference discretizations of the first-oder and second-order derivative operators are required. They are respectively ∂_x^+, ∂_x^-, ∂_y^+, ∂_y^-, $\partial_x^-\partial_x^+$, $\partial_x^+\partial_x^-$, $\partial_y^-\partial_y^+$, and $\partial_y^+\partial_y^-$. Note that we apply periodic boundary condition with these discretizations. It is often used for the image denoising problem since it is able to allow fast numerical computations.

From now on, we assume $\Omega = [0, M] \times [0, N]$. Let $\bar{\Omega} \subset \Omega$ be the discretized image domain as given by

$$\bar{\Omega} = \{(x,y) \in \Omega | (x,y) = (x_i, y_j),\ x_i = i,\ y_j = j,\ 1 \le i \le M \text{ and } 1 \le j \le N)\}.$$

For simplicity, each grid point $(x_i, y_j) \in \bar{\Omega}$ is denoted by (i, j), where the coordinates x and y are oriented along columns and rows, respectively. All the variables are defined on these grid points. Let $\bar{\Omega} \to \mathbb{R}^{M \times N}$ be the 2D grayscale image space with size MN. So, the first-order

forward differences of the grayscale image u at grid point (i,j) along x and y directions are respectively

$$\partial_x^+(u)_{i,j} = \begin{cases} (u)_{i,j+1} - (u)_{i,j} & \text{if } 1 \le i \le M, 1 \le j \le N \\ (u)_{i,1} - (u)_{i,j} & \text{if } 1 \le i \le M, j = N \end{cases} \tag{27}$$

and

$$\partial_y^+(u)_{i,j} = \begin{cases} (u)_{i+1,j} - (u)_{i,j} & \text{if } 1 \le i \le M, 1 \le j \le N \\ (u)_{1,j} - (u)_{i,j} & \text{if } i = M, 1 \le j \le N \end{cases}. \tag{28}$$

The first-order backward differences are respectively

$$\partial_x^-(u)_{i,j} = \begin{cases} (u)_{i,j} - (u)_{i,j-1} & \text{if } 1 \le i \le M, 1 \le j \le N \\ (u)_{i,j} - (u)_{i,N} & \text{if } 1 \le i \le M, j = 1 \end{cases} \tag{29}$$

and

$$\partial_y^-(u)_{i,j} = \begin{cases} (u)_{i,j} - (u)_{i-1,j} & \text{if } 1 \le i \le M, 1 \le j \le N \\ (u)_{i,j} - (u)_{M,j} & \text{if } i = 1, 1 \le j \le N \end{cases}. \tag{30}$$

For $\boldsymbol{p} = (p_1, p_2) \in (\mathbb{R}^{M\times N})^2$ and $u \in \mathbb{R}^{M\times N}$, the discrete version of the first-order adjoint divergence operator satisfies

$$\sum_{\substack{1\le i\le M \\ 1\le j\le M}} -(\nabla\cdot\boldsymbol{p})\cdot u = \sum_{\substack{1\le i\le M \\ 1\le j\le M}} \boldsymbol{p}\cdot(\nabla u)$$

Therefore, according to the definition of discrete gradient operator

$$(\nabla u)_{i,j} = \begin{pmatrix} \partial_x^+(u)_{i,j} \\ \partial_y^+(u)_{i,j} \end{pmatrix},$$

the discrete divergence is given by

$$(\nabla\cdot\boldsymbol{p})_{i,j} = \partial_x^+(p_1)_{i,j} + \partial_y^+(p_2)_{i,j}. \tag{31}$$

The second-order derivative operators of u at grid point (i,j) are given by

$$\partial_x^+\partial_x^-(u)_{i,j} = \partial_x^-\partial_x^+(u)_{i,j} = \begin{cases} (u)_{i,N} - 2(u)_{i,j} + (u)_{i,j+1} & \text{if } 1 \le i \le M, j = 1 \\ (u)_{i,j-1} - 2(u)_{i,j} + (u)_{i,j+1} & \text{if } 1 \le i \le M, 1 < j < N \\ (u)_{i,j-1} - 2(u)_{i,j} + (u)_{i,1} & \text{if } 1 \le i \le M, j = N \end{cases} \tag{32}$$

$$\partial_y^+\partial_y^-(u)_{i,j} = \partial_y^-\partial_y^+(u)_{i,j} = \begin{cases} (u)_{M,j} - 2(u)_{i,j} + (u)_{i+1,j} & \text{if } i = 1, 1 \le j \le N \\ (u)_{i-1,j} - 2(u)_{i,j} + (u)_{i+1,j} & \text{if } 1 < i < M, 1 \le j \le N \\ (u)_{i-1,j} - 2(u)_{i,j} + (u)_{1,j} & \text{if } i = M, 1 \le j \le N \end{cases} \tag{33}$$

$$\partial_x^+\partial_y^-(u)_{i,j} = \begin{cases} (u)_{i,j+1} - (u)_{i,j} - (u)_{M,j+1} + (u)_{M,j} & \text{if } i = 1, 1 \le j < N \\ (u)_{i,1} - (u)_{i,j} - (u)_{M,1} + (u)_{M,j} & \text{if } i = 1, j = N \\ (u)_{i,j+1} - (u)_{i,j} - (u)_{i-1,j+1} + (u)_{i-1,j} & \text{if } 1 < i \le M, 1 \le j < N \\ (u)_{i,1} - (u)_{i,j} - (u)_{i-1,1} + (u)_{i-1,j} & \text{if } 1 < i \le M, j = N \end{cases} \tag{34}$$

and

$$\partial_y^+\partial_x^-(u)_{i,j} = \begin{cases} (u)_{i+1,j} - (u)_{i,j} - (u)_{i+1,N} + (u)_{i,N} & \text{if } 1 \le i < M, j = 1 \\ (u)_{1,j} - (u)_{i,j} - (u)_{1,N} + (u)_{i,N} & \text{if } i = M, j = 1 \\ (u)_{i+1,j} - (u)_{i,j} - (u)_{i+1,j-1} + (u)_{i,j-1} & \text{if } 1 \le i < M, 1 < j \le N \\ (u)_{1,j} - (u)_{i,j} - (u)_{1,j-1} + (u)_{i,j-1} & \text{if } i = M, 1 < j \le N \end{cases}. \tag{35}$$

Based on (32) and (33), the discrete Laplace operator is given by

$$(\triangle u)_{i,j} = \partial_x^-\partial_x^+(u)_{i,j} + \partial_y^-\partial_y^+(u)_{i,j}. \tag{36}$$

3.2. Numerical algorithm based on split Bregman method

Alternating direction method of multipliers (ADMM) has been successfully applied in the field of image denoising for solving variational models, which require the minimization of nonlinear and non-differentiable functionals. For instance, the authors in [38] developed the split Bregman (SB) iteration method to solve the TV model, while the augmented Lagrangian method (ALM) has been developed for the TV model in [39, 40]. Note that both methods employ ADMM to minimize their energy functionals and the convergence of ADMM is always guaranteed. In [39, 40], the authors have proven that the two methods are in fact equivalent. For applying ADMM to minimize the energy functionals, we refer to [41, 42, 43, 44, 45].

In this subsection, our aim is to show how SB method can be used to solve our proposed variational model (17). Before starting our discussion on SB method, we note that our proposed model in (17) can be expressed as

$$\min_u \left\{ \mathcal{D}_{\mathrm{AN}}(u, z) + \frac{\gamma}{2}\int_\Omega \kappa_{\mathrm{M}}^2(u)d\Omega - \gamma\int_\Omega \kappa_{\mathrm{G}}(u)d\Omega \right\}. \tag{37}$$

Introducing the auxiliary variables $\boldsymbol{p} = (p_1, p_2)$, $\boldsymbol{m} = (m_1, m_2)$, $\boldsymbol{n} = (n_1, n_2)$, and q leads to

the following constrained minimization problem:

$$\min_{u,q,\boldsymbol{m},\boldsymbol{n},\boldsymbol{p}} \left\{ \mathcal{D}_{\mathrm{AN}}(u,z) + \frac{\gamma}{2}\int_\Omega q^2 d\Omega + \gamma \int_\Omega \frac{p_{1_y}p_{2_x} - p_{1_x}p_{2_y}}{(1+p_1^2+p_2^2)^2} d\Omega \right\}, \tag{38}$$

$$\text{s.t. } \boldsymbol{p} = \nabla u,\ \boldsymbol{m} = \frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2+1}},\ \boldsymbol{m} = \boldsymbol{n},\ q = \nabla\cdot\boldsymbol{n}. \tag{39}$$

Therefore the solution of the constrained minimization problem (39) can be obtained by minimizing the following functional

$$\begin{aligned}\mathcal{E}(u,q,\boldsymbol{m},\boldsymbol{n},\boldsymbol{p};\boldsymbol{b}_1,\boldsymbol{b}_2,\boldsymbol{b}_3,b_4) = {} & \mathcal{D}_{\mathrm{AN}}(u,z) + \frac{\gamma}{2}\int_\Omega q^2 d\Omega + \gamma\int_\Omega \frac{p_{1_y}p_{2_x} - p_{1_x}p_{2_y}}{(1+p_1^2+p_2^2)^2} d\Omega \\ & + \frac{\theta_1}{2}\int_\Omega (\boldsymbol{p} - \nabla u - \boldsymbol{b_1})^2 d\Omega + \frac{\theta_2}{2}\int_\Omega (\boldsymbol{m} - \frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2+1}} - \boldsymbol{b_2})^2 d\Omega \\ & + \frac{\theta_3}{2}\int_\Omega (\boldsymbol{n} - \boldsymbol{m} - \boldsymbol{b_3})^2 d\Omega + \frac{\theta_4}{2}\int_\Omega (q - \nabla\cdot\boldsymbol{n} - b_4)^2 d\Omega, \end{aligned} \tag{40}$$

where $\boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}, b_4$ are the Bregman parameters and $\theta_1, \theta_2, \theta_3, \theta_4$ are the positive penalty parameters.

Since all variables u, q, $\boldsymbol{m}$, $\boldsymbol{n}$, $\boldsymbol{p}$ are very difficult to solve simultaneously, we then separate the minimization problem of $\mathcal{E}$ into five subproblems and propose an alternating minimization procedure as shown in Algorithm 1 to approximate the solution. The process is repeated until one of the following stopping rules is satisfied:

$$\frac{\left\|u^{(m)} - u^{(m-1)}\right\|^2}{\left\|u^{(m-1)}\right\|^2} < \epsilon_1^{\mathrm{SB}}, \tag{41}$$

$$m \geq \epsilon_2^{\mathrm{SB}}, \tag{42}$$

where ϵ_1^{SB} denotes the predefined small positive number and ϵ_2^{SB} denotes the maximum iteration of the SB method. Here m represents the index of the current iteration.

The five subproblems associated with the minimization of $\mathcal{E}$ in (40) are as follows:

u-subproblem. We fix the variables q, $\boldsymbol{m}$, $\boldsymbol{n}$, $\boldsymbol{p}$ and derive the EL equation respect to u:

$$u - \theta_1 \triangle u = G, \tag{45}$$

where $G = z + \theta_1 \nabla\cdot(\boldsymbol{b_1} - \boldsymbol{p})$. To solve the linear partial differential equation in (45), we apply the discrete divergence operator in (31) and the discrete Laplace operator in (36) as follows

$$(u)_{i,j} - \theta_1\left(\partial_x^-\partial_x^+(u)_{i,j} + \partial_y^-\partial_y^+(u)_{i,j}\right) = (G)_{i,j}, \tag{46}$$

Algorithm 1 SB iteration for the proposed model

1) Initialization: set $m = 1$, choose $\epsilon_1^{\mathrm{SB}}, \epsilon_2^{\mathrm{SB}}, \theta_1, \theta_2, \theta_3, \theta_4 > 0$ and $\boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}, b_4$.

2) **Repeat**

2.1) Compute $(u^{(m)}, p^{(m)}, \boldsymbol{m}^{(m)}, \boldsymbol{n}^{(m)}, \boldsymbol{p}^{(m)})$ as an *approximate* minimizer of the functional $\mathcal{E}$ in (40) with the Bregman parameter $\boldsymbol{b}_1^{(m-1)}$, $\boldsymbol{b}_2^{(m-1)}$, , $\boldsymbol{b}_3^{(m-1)}$, $b_4^{(m-1)}$, i.e.,

$$(u^{(m)}, p^{(m)}, \boldsymbol{m}^{(m)}, \boldsymbol{n}^{(m)}, \boldsymbol{p}^{(m)}) \approx \underset{u,q,\boldsymbol{m},\boldsymbol{n},\boldsymbol{p}}{\arg\min}\ \mathcal{E}(u, q, \boldsymbol{m}, \boldsymbol{n}, \boldsymbol{p}; \boldsymbol{b}_1^{(m-1)}, \boldsymbol{b}_2^{(m-1)}, \boldsymbol{b}_3^{(m-1)}, b_4^{(m-1)}). \tag{43}$$

2.2) Update Bregman parameters

$$\begin{cases} \boldsymbol{b}_1^{(m)} = \boldsymbol{b}_1^{(m-1)} + \nabla u^{(m)} - \boldsymbol{p}^{(m)}, \\ \boldsymbol{b}_2^{(m)} = \boldsymbol{b}_2^{(m-1)} + \frac{\boldsymbol{p}^{(m)}}{\sqrt{|\boldsymbol{p}^{(m)}|^2+1}} - \boldsymbol{m}^{(m)}, \\ \boldsymbol{b}_3^{(m)} = \boldsymbol{b}_3^{(m-1)} + \boldsymbol{m}^{(m)} - \boldsymbol{n}^{(m)}, \\ b_4^{(m)} = b_4^{(m-1)} + \nabla \cdot \boldsymbol{n}^{(m)} - q^{(m)}. \end{cases} \tag{44}$$

until a stopping rule for SB iteration is satisfied.

where $(G)_{i,j} = (z)_{i,j} - \theta_1 \left(\partial_x^-((b_{11})_{i,j} - (p_1)_{i,j}) + \partial_y^-((b_{12})_{i,j} - (p_2)_{i,j})\right)$. As periodic boundary conditions have been imposed on the discrete derivatives in Section 3.1, the discrete Fourier transform can be directly applied to the both sides of (46)

$$\mathcal{F}\left\{(u)_{i,j} - \theta_1 \left(\partial_x^-\partial_x^+(u)_{i,j} + \partial_y^-\partial_y^+(u)_{i,j}\right)\right\} = \mathcal{F}\left\{(G)_{i,j}\right\}, \tag{47}$$

where $\mathcal{F}$ denotes the discrete Fourier transform. For $r \in [0, M)$ and $s \in [0, N)$ in the discrete frequency domain, we have

$$\underbrace{\left(1 - 2\theta_1\left(\cos\frac{2\pi s}{N} + \cos\frac{2\pi r}{N} - 2\right)\right)}_{\zeta}\mathcal{F}\left\{(u)_{i,j}\right\} = \mathcal{F}\left\{(G)_{i,j}\right\}, \tag{48}$$

where $i \in [1, M]$ and $j \in [1, N]$ are the indexes in discrete time domain. Thus a closed-form solution of u is given by

$$(u)_{i,j} = \mathrm{Real}\left(\mathcal{F}^{-1}\left\{\frac{\mathcal{F}\left\{(G)_{i,j}\right\}}{\zeta}\right\}\right), \tag{49}$$

where $\mathcal{F}^{-1}$ denotes the discrete inverse Fourier transform. We note that 'Real' is the real part of a complex number and '—' stands for point-wise division of matrices.

q-subproblem. Fixing the variables u, $\boldsymbol{m}$, $\boldsymbol{n}$, $\boldsymbol{p}$ leads to the EL equation respect to q as given by

$$\gamma q + \theta_4(q - \nabla \cdot \boldsymbol{n} - b_4) = 0. \tag{50}$$

As can be seen, the exact solution is given by

$$(q)_{i,j} = \frac{\theta_4((\partial_x^-(n_1)_{i,j} + \partial_y^-(n_2)_{i,j}) + (b_4)_{i,j})}{\gamma + \theta_4}. \tag{51}$$

$\boldsymbol{m}$*-subproblem.* Fixing the variables u, q, $\boldsymbol{n}$, $\boldsymbol{p}$ yields the EL equation respect to $\boldsymbol{m}$ as follows:

$$\theta_2(\boldsymbol{m} - \frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2 + 1}} - \boldsymbol{b_2}) - \theta_3(\boldsymbol{n} - \boldsymbol{m} - \boldsymbol{b_3}) = 0, \tag{52}$$

which can be determined using the closed-form formula

$$(\boldsymbol{m})_{i,j} = \frac{\theta_2((\frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2+1}})_{i,j} + (\boldsymbol{b_2})_{i,j}) + \theta_3((\boldsymbol{n})_{i,j} - (\boldsymbol{b_3})_{i,j})}{\theta_2 + \theta_3}. \tag{53}$$

$\boldsymbol{n}$*-subproblem.* By fixing the variables u, q, $\boldsymbol{m}$, $\boldsymbol{p}$, the EL equation respect to $\boldsymbol{n}$ is given by

$$\left(\theta_3 - \theta_4\partial_x^+\partial_x^-\right)(n_1)_{i,j} - \theta_4\partial_x^+\partial_y^-(n_2)_{i,j} = (h_1)_{i,j}, \tag{54}$$

$$\left(\theta_3 - \theta_4\partial_y^+\partial_y^-\right)(n_2)_{i,j} - \theta_4\partial_y^+\partial_x^-(n_1)_{i,j} = (h_2)_{i,j}, \tag{55}$$

where

$$(h_1)_{i,j} = \theta_3(m_1 + b_{31}) - \theta_4\partial_x^+(q - b_4), \tag{56}$$

$$(h_2)_{i,j} = \theta_3(m_2 + b_{32}) - \theta_4\partial_y^+(q - b_4). \tag{57}$$

Taking the DFT with (54) and (55) leads to the following system of linear equations

$$\mathbf{A}\mathbf{x} = \mathbf{b},$$

where

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}, \mathbf{x} = \begin{pmatrix} \mathcal{F}\{(n_1)_{i,j}\} \\ \mathcal{F}\{(n_2)_{i,j}\} \end{pmatrix}, \mathbf{b} = \begin{pmatrix} \mathcal{F}\{(h_1)_{i,j}\} \\ \mathcal{F}\{(h_2)_{i,j}\} \end{pmatrix},$$

$$a_{11} = \theta_3 - 2\theta_4\left(\cos\frac{2\pi s}{N} - 1\right),$$

$$a_{12} = -\theta_4\left(-1 - \cos\frac{2\pi r}{M} + \sqrt{-1}\sin\frac{2\pi r}{M}\right)\left(1 - \cos\frac{2\pi s}{N} + \sqrt{-1}\sin\frac{2\pi s}{N}\right),$$

$$a_{21} = -\theta_4\left(-1 - \cos\frac{2\pi s}{N} + \sqrt{-1}\sin\frac{2\pi s}{N}\right)\left(1 - \cos\frac{2\pi r}{M} + \sqrt{-1}\sin\frac{2\pi r}{M}\right),$$

$$a_{22} = \theta_3 - 2\theta_4\left(\cos\frac{2\pi r}{M} - 1\right).$$

For all discrete frequencies, the determinant of the coefficient matrix $\mathbf{A}$ is given by

$$D = \theta_3^2 - 2\theta_3\theta_4\left(\cos\frac{2\pi s}{N} + \cos\frac{2\pi r}{M} - 2\right), \tag{58}$$

which is always positive. Thus, the closed-form solution of n_1 and n_2 can be expressed as

$$\begin{cases} (n_1)_{i,j} = \text{Real}\left(\mathcal{F}^{-1}\left\{\frac{a_{22}\mathcal{F}\{(h_1)_{i,j}\} - a_{12}\mathcal{F}\{(h_2)_{i,j}\}}{D}\right\}\right), \\ (n_2)_{i,j} = \text{Real}\left(\mathcal{F}^{-1}\left\{\frac{a_{11}\mathcal{F}\{(h_2)_{i,j}\} - a_{21}\mathcal{F}\{(h_1)_{i,j}\}}{D}\right\}\right). \end{cases} \tag{59}$$

***p**-subproblem.* Fixing the variable u, q, $\boldsymbol{m}$, $\boldsymbol{n}$, the EL equation respect to $\boldsymbol{p}$ is given by (see Appendix B)

$$\begin{cases} -\gamma\nabla\cdot\boldsymbol{w}_1 + \gamma\Gamma p_1 + \theta_1(p_1 - u_x - b_{11}) + \theta_2(m_1 - \psi p_1 - b_{21})(\psi^3 p_1^2 - \psi) = 0, \\ -\gamma\nabla\cdot\boldsymbol{w}_2 + \gamma\Gamma p_2 + \theta_1(p_2 - u_y - b_{12}) + \theta_2(m_2 - \psi p_2 - b_{22})(\psi^3 p_2^2 - \psi) = 0, \end{cases} \tag{60}$$

where $\boldsymbol{w}_1 = \left(-\frac{(p_2)_y}{\beta_2^2}, \frac{(p_2)_x}{\beta_2^2}\right)$, $\boldsymbol{w}_2 = \left(\frac{(p_1)_y}{\beta_2^2}, -\frac{(p_1)_x}{\beta_2^2}\right)$, $\Gamma_1 = (p_1)_x(p_2)_y - (p_1)_y(p_2)_x$, $\Gamma_2 = p_1^2 + p_2^2 + 1$, $\Gamma = \frac{4\Gamma_1}{\Gamma_2^3}$, $\psi = \frac{1}{\sqrt{\Gamma_2}}$.

To solve the nonlinear partial differential equation in (60), we develop a semi-implicit fixed-point iteration method. We start from an initial solution $\boldsymbol{p}^{[0]}$ and compute a sequence of approximate solutions $\boldsymbol{p}^{[1]}, \boldsymbol{p}^{[2]}, \cdots, \boldsymbol{p}^{[\nu]}, \boldsymbol{p}^{[\nu+1]}, \cdots$ by solving the discrete linear system

$$\begin{cases} (p_1^{[\nu+1]})_{i,j} = \dfrac{\gamma(\nabla\cdot\boldsymbol{w}_1^{[\nu]})_{i,j} + \theta_1(\partial_x^+(u)_{i,j} + (b_{11})_{i,j}) + \theta_2\psi(m_1 - b_{21})_{i,j}}{\gamma(\Gamma^{[\nu]})_{i,j} + \theta_1 + \theta_2(\psi^{[\nu]})_{i,j}((m_1 - \psi^{[\nu]}p_1^{[\nu]} - b_{21})_{i,j}(\psi^{[\nu]})_{i,j}(p_1^{[\nu]})_{i,j} + 1)}, \\ (p_2^{[\nu+1]})_{i,j} = \dfrac{\gamma(\nabla\cdot\boldsymbol{w}_2^{[\nu]})_{i,j} + \theta_1(\partial_y^+(u)_{i,j} + (b_{12})_{i,j}) + \theta_2\psi(m_2 - b_{22})_{i,j}}{\gamma(\Gamma^{[\nu]})_{i,j} + \theta_1 + \theta_2(\psi^{[\nu]})_{i,j}((m_2 - \psi^{[\nu]}p_2^{[\nu]} - b_{22})_{i,j}(\psi^{[\nu]})_{i,j}(p_2^{[\nu]})_{i,j} + 1)}, \end{cases} \tag{61}$$

until the stopping rule $\max\{\frac{\|p_1^{[\nu]} - p_1^{[\nu-1]}\|}{\|p_1^{[\nu]}\|}, \frac{\|p_2^{[\nu]} - p_2^{[\nu-1]}\|}{\|p_2^{[\nu]}\|}\} < \epsilon_1^{\text{FP}}$ or $\nu \geq \epsilon_2^{\text{FP}}$ is met for a given small threshold $\epsilon_1^{\text{FP}} > 0$ and the maximum iteration of the FP method ϵ_2^{FP}.

Now we are at a position to formally present the SB method for our proposed model as given in Algorithm 2.

4. Experimental results

In this section, we present a number of numerical experiments to show the performance of our proposed model (the NewCv model) in comparison with the existing variational models mentioned in Section 1.2, which are the TV, TL, BH, CEP-L^2, TVL, INFCON, TVBH, TGV, MC and GC models.

We use the SB algorithms of Lu et al. [34] to solve the TV, TL, BH, CEP-L^2, TVL, INFCON, TVBH, TGV, and MC models, while we apply the SB algorithm developed in our previous work [46] to solve the GC model. All eleven SB algorithms are started with $u^{(0)} = z$

Algorithm 2 SB iteration for the proposed model – solve the subproblems of (43)

1) Initialization: set $m = 1$, choose $\epsilon_1^{\text{SB}}, \epsilon_2^{\text{SB}}, \epsilon_1^{\text{FP}}, \epsilon_2^{\text{FP}}, \theta_1, \theta_2, \theta_3, \theta_4 > 0$ and $\boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}, b_4$.

2) **Repeat**

2.1) Compute $u^{(m)}$ by (49) with $q^{(m-1)}$, $\boldsymbol{m}^{(m-1)}$, $\boldsymbol{n}^{(m-1)}$ and $\boldsymbol{p}^{(m-1)}$.

2.2) Compute $q^{(m)}$ by (51) with $u^{(m)}$, $\boldsymbol{m}^{(m-1)}$, $\boldsymbol{n}^{(m-1)}$ and $\boldsymbol{p}^{(m-1)}$.

2.3) Compute $\boldsymbol{m}^{(m)}$ by (53) with $u^{(m)}$, $q^{(m)}$, $\boldsymbol{n}^{(m-1)}$ and $\boldsymbol{p}^{(m-1)}$.

2.4) Compute $\boldsymbol{n}^{(m)}$ by (59) with $u^{(m)}$, $q^{(m)}$, $\boldsymbol{m}^{(m)}$ and $\boldsymbol{p}^{(m-1)}$.

2.5) Compute $\boldsymbol{p}^{(m)}$ by (61) with $u^{(m)}$, $q^{(m)}$, $\boldsymbol{m}^{(m)}$ and $\boldsymbol{n}^{(m)}$.

2.6) Update all Bregman parameters by (44) with $u^{(m)}$, $q^{(m)}$, $\boldsymbol{m}^{(m)}$, $\boldsymbol{n}^{(m)}$ and $\boldsymbol{p}^{(m)}$.

until a stopping rule for SB iteration is satisfied.

and terminated with the same stopping rules as given in (41) and (42), where $\epsilon_1^{\text{SB}} = 5.5\times10^{-5}$ and $\epsilon_2^{\text{SB}} = 1000$. For the proposed SB algorithm in Algorithm 2, we apply $\epsilon_1^{\text{FP}} = 10^{-2}$ and $\epsilon_2^{\text{FP}} = 10$ to stop the iterative procedure in solving $\boldsymbol{p}$-subproblem. We note that all eleven SB algorithms consist of regularization and penalty parameters. In each model, we carry out many experiments with different values of these parameters and we select manually the ones with the best denoising result.

In total, we apply twelve images to demonstrate the denoising performance of eleven different models. The test images are 256×256 as shown in Figure 1: i) three synthetic images: (a) "Piecewise constant", (b) "Piecewise linear", and (c) "Piecewise smooth"; ii) six real natural images: (d) "Barbara", (e) "Boat", (f) "Cameraman", (g) "Castle", (h) "Lena" and (i) "Peppers"; and iii) three real noisy images: (j) "Abdomen", (k) "Brain", (l) "Knee".

For measuring the denoised image quality, Peak signal-to-noise ratio (PSNR) in decibels (dB):

$$\text{PNSR} = 10\log_{10}\left(\frac{255^2}{\frac{1}{256^2}\|\hat{u}-u\|^2}\right), \tag{62}$$

and structure similarity index map (SSIM):

$$\text{SSIM} = \frac{(2\mu_1\mu_2 + C_1)(2\sigma_{12} + C_2)}{(\mu_1^2 + \mu_2^2 + C_1)(\sigma_1^2 + \sigma_2^2 + C_2)}, \tag{63}$$

are used to perform effectiveness evaluation for all variational models, where μ_1 and μ_2 are averages of $\hat{u}$ and u, respectively, σ_1 and σ_2 are variance of $\hat{u}$ and u, respectively, and σ_{12} is

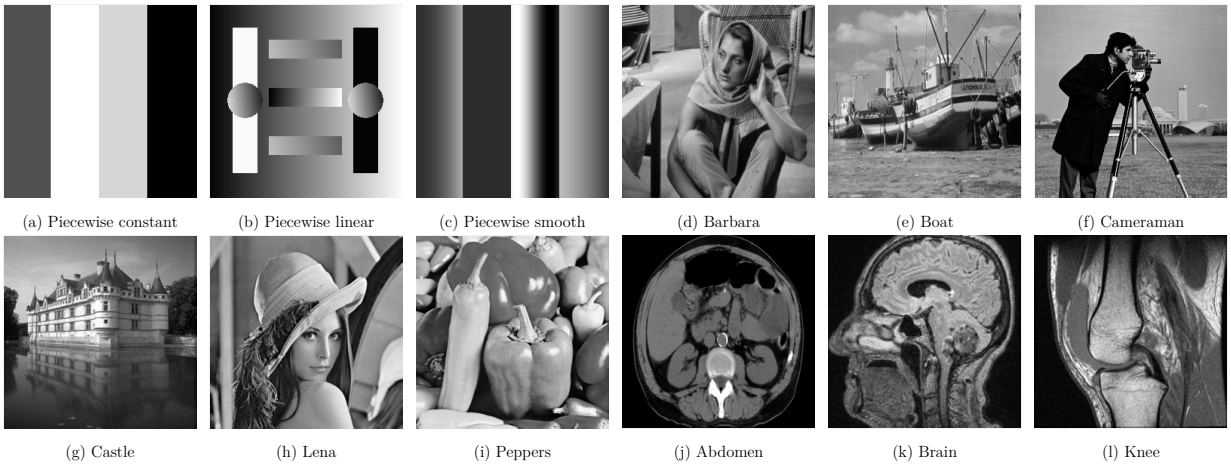


Figure 1: Original images: i) three synthetic images (a)-(c); ii) six real natural images (d)-(i); iii) three real noisy images (j)-(l).

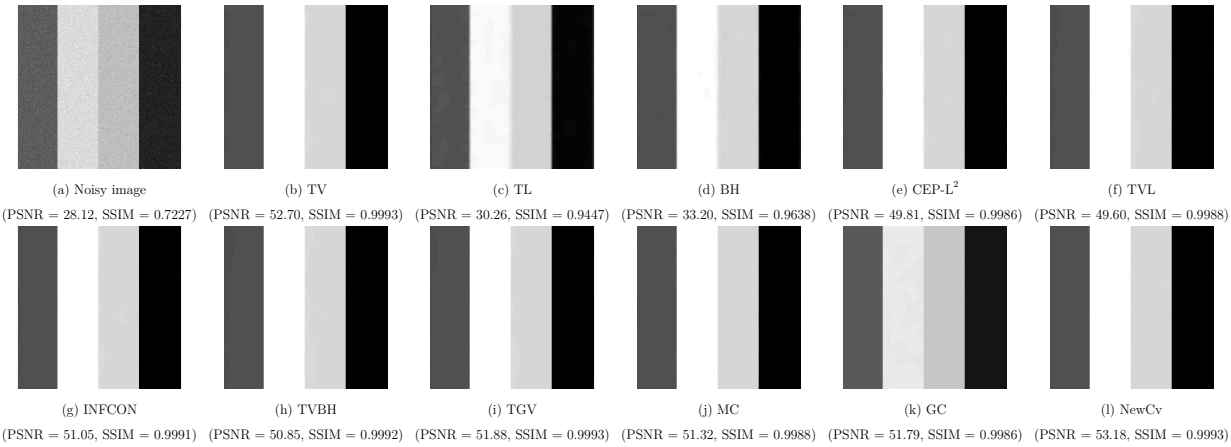
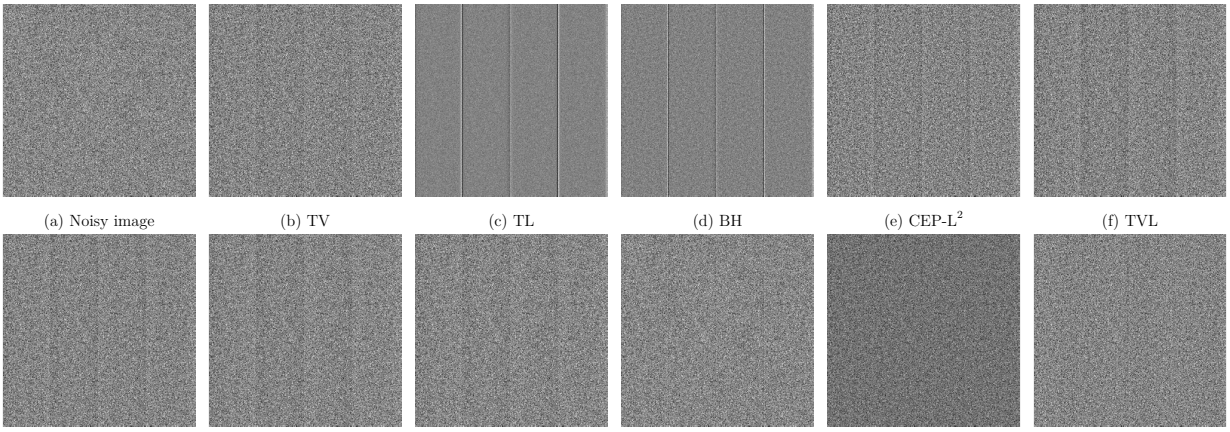


Figure 2: Edge-preservation ability test. The "Piecewise constant" image in Figure 1 (a) is distorted by the Gaussian noise with the noise variance 0.07. (a) Noisy image; (b)-(l) denoised images by eleven different models.

the covariance of $\hat{u}$ and u. The positive constants C_1 and C_2 can be thought of as stabilizing constants for near-zero denominator values. Here $\hat{u}$ and u denote the denoised image and the original clean image, respectively. Recall that the higher the PSNR, the better denoised image has been recovered to match the original clean image. For SSIM in (63), it is a well-known quality metric used to asses the the conservation of the structural information of the denoised image. Note that a perfect restoration has SSIM value equal to 1. We refer [47] for more details in image quality assessment.

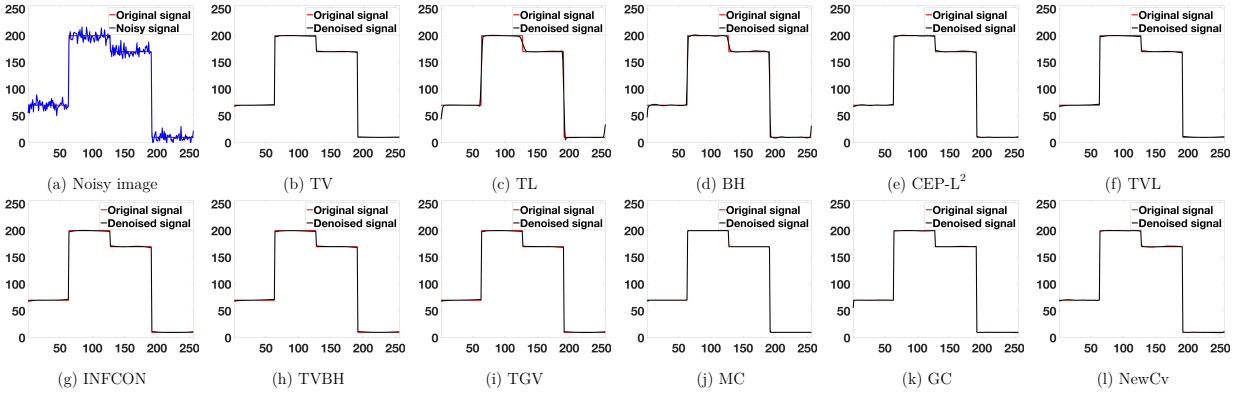
Note that in all experiments, each denoising algorithm is performed using MATLAB R2018a and run on a machine configured with Intel(R) Core(TM) i7 Quad-core 4.2GHz and 32GB of RAM.



(a) Noisy image (b) TV (c) TL (d) BH (e) CEP-L^2 (f) TVL

(g) INFCON (h) TVBH (i) TGV (j) MC (k) GC (l) NewCv

Figure 3: Comparison of residual images between the original and denoised images in Figure 2 by eleven different models.



(a) Noisy image (b) TV (c) TL (d) BH (e) CEP-L^2 (f) TVL

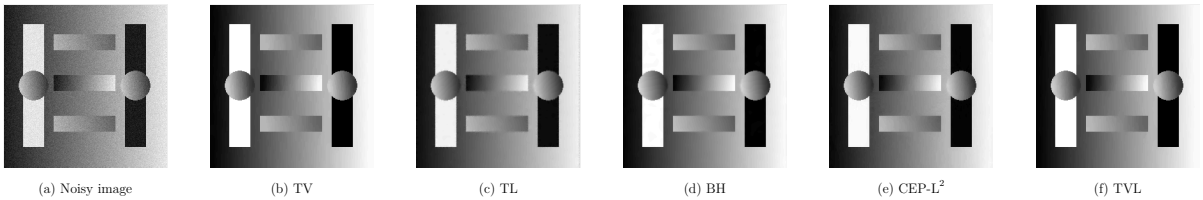
(g) INFCON (h) TVBH (i) TGV (j) MC (k) GC (l) NewCv

Figure 4: Plots of the middle slices of the denoised images in Figure 2 by eleven different models.

4.1. Comparison 1: edge-preservation ability test

In Figure 2, we illustrate the edge-preservation ability of eleven different models on a noisy image, which is piecewise constant, in the sub-figure (a) through visual inspection. From the sub-figures (b) to (l), it is clear that our NewCv model preserves the edges of the strips perfectly. Its denoised results look almost the same as the original clean one in Figure 1 (a). TV, TGV, MC and GC models also give very good edge-preservation results. Similar visual effect can be observed on results by INFCON and TVBH models. The denoised images in the sub-figures (c)-(f) show that applying TL, BH, CEP-L^2 and TVL can blur the edges of the objects, where TL model is the worst method for edge preservation. By incorporating the TV regularization into BH and TL regularizations, TVL and TVBH model can improve the denoised quality as can see from the sub-figures (f) and (h). However, the edges of the objects are still smeared by these two models.

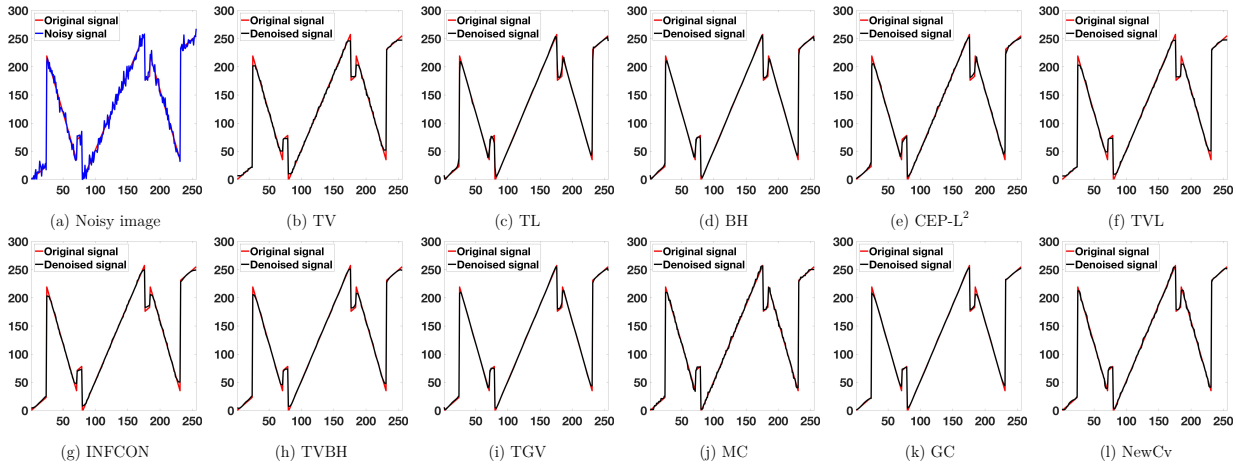
In Figure 3, we present the associated residual images $z - u$ for all variational models,



(a) Noisy image (PSNR = 27.43, SSIM = 0.6317) (b) TV (PSNR = 40.72, SSIM = 0.9675) (c) TL (PSNR = 33.50, SSIM = 0.9552) (d) BH (PSNR = 39.60, SSIM = 0.9824) (e) CEP-L^2 (PSNR = 41.38, SSIM = 0.9770) (f) TVL (PSNR = 41.34, SSIM = 0.9754)

(g) INFCON (PSNR = 41.34, SSIM = 0.9802) (h) TVBH (PSNR = 41.99, SSIM = 0.9823) (i) TGV (PSNR = 43.40, SSIM = 0.9874) (j) MC (PSNR = 42.29, SSIM = 0.9756) (k) GC (PSNR = 43.04, SSIM = 0.9898) (l) NewCv (PSNR = 43.93, SSIM = 0.9865)

Figure 5: Smoothness-preservation ability test. The "Piecewise linear" image in Figure 1 (b) is distorted by the Gaussian noise with the noise variance 0.07. (a) noisy image; (b)-(l) denoised images by eleven different models.



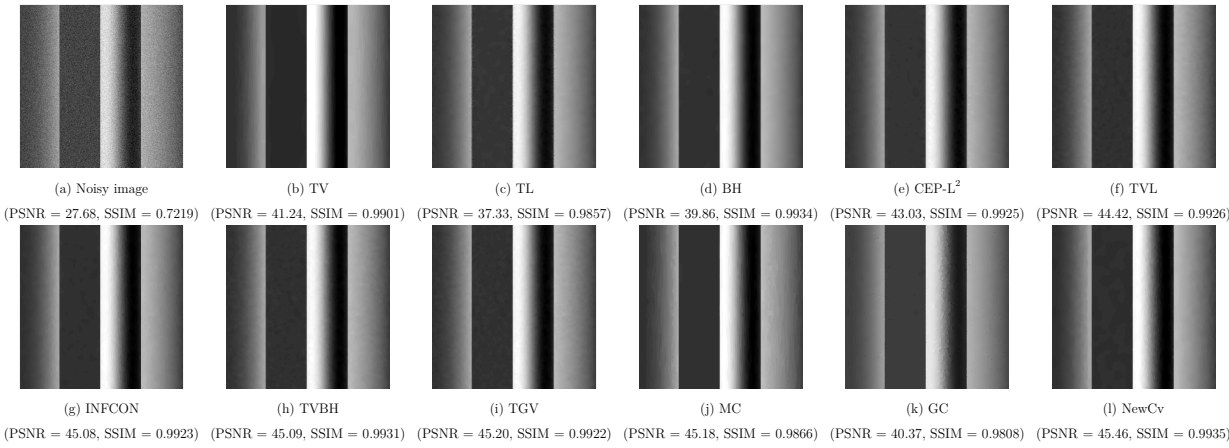
(a) Noisy image (b) TV (c) TL (d) BH (e) CEP-L^2 (f) TVL

(g) INFCON (h) TVBH (i) TGV (j) MC (k) GC (l) NewCv

Figure 6: Plots of the middle slices of the denoised images in Figure 5 "Piecewise linear" by eleven different models.

where z is the noisy image in Figure 2 (a) and u represents each corresponding denoised image in Figure 2. The residual image in the sub-figure (l) contains almost all noise, whereas the other ten residual images contain more or less structure information of the original image in Figure 1 (a). This means that some contrast information has been lost in the denoised images handled by other ten models. Figure 3 illustrates that our NewCv model is the best in preserving image contrast.

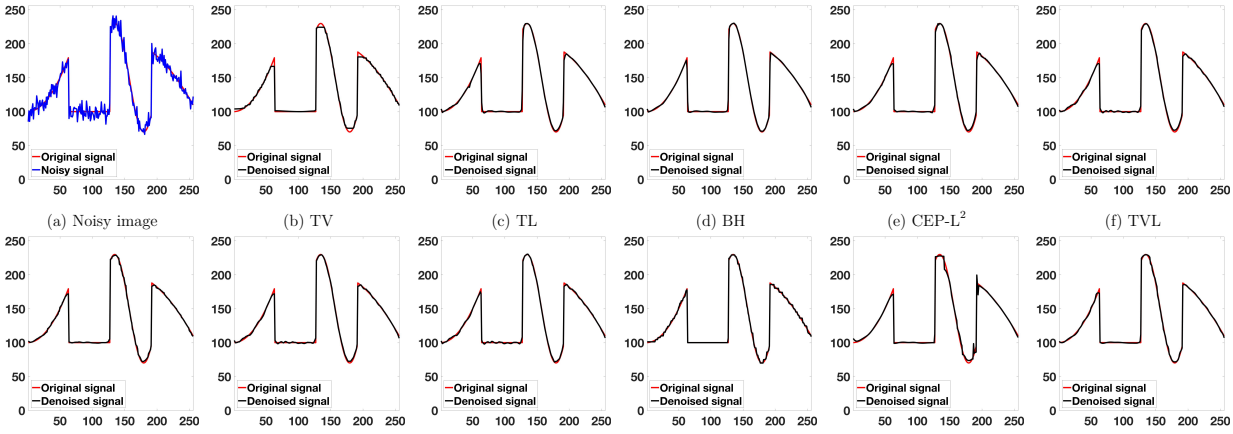
In Figure 4, the plots of the middle slices of the denoised images in Figure 2 are presented. We can see that the denoised slice curve (black) in the sub-figure (l) almost overlaps with the original noise free slice curve (red), which demonstrates the preservation of images contrast as well as corners of NewCv model. Compared with TGV and NewCv models, the other nine models lead to less pleasant fitting results.



(a) Noisy image (PSNR = 27.68, SSIM = 0.7219) (b) TV (PSNR = 41.24, SSIM = 0.9901) (c) TL (PSNR = 37.33, SSIM = 0.9857) (d) BH (PSNR = 39.86, SSIM = 0.9934) (e) CEP-L^2 (PSNR = 43.03, SSIM = 0.9925) (f) TVL (PSNR = 44.42, SSIM = 0.9926)

(g) INFCON (PSNR = 45.08, SSIM = 0.9923) (h) TVBH (PSNR = 45.09, SSIM = 0.9931) (i) TGV (PSNR = 45.20, SSIM = 0.9922) (j) MC (PSNR = 45.18, SSIM = 0.9866) (k) GC (PSNR = 40.37, SSIM = 0.9808) (l) NewCv (PSNR = 45.46, SSIM = 0.9935)

Figure 7: Smoothness-preservation ability test. The "Piecewise smooth" image in Figure 1 (c) is distorted by the Gaussian noise with the noise variance 0.07. (a) noisy image; (b)-(l) denoised image by eleven different models.



(a) Noisy image (b) TV (c) TL (d) BH (e) CEP-L^2 (f) TVL

(g) INFCON (h) TVBH (i) TGV (j) MC (k) GC (l) NewCv

Figure 8: Plots of the middle slices of the denoised images in Figure 7 "Piecewise smooth" by eleven different models.

4.2. Comparison 2: smoothness-preservation ability test

In this subsection, we test the smoothness-preservation ability of eleven different models on the "Piecewise linear" image in Figure 1 (b) and the "Piecewise smooth" in Figures 1 (c). Both test images are corrupted with Gaussian noise with the noise variance 0.07.

Figure 5 and 7 show the denoised images on the test images by eleven different models. One can easily see in the sub-figure (l) that our NewCv model produces the denoised results with the highest values of PSNR and SSIM. Moreover, the plots of the middle slices of the denoised images as shown in Figures 6 and 8 show that the denoised signal in the sub-figure (b) by TV model is composed of jagged appearance which is staircase effect. In addition, the second-order models involving TV regularization, CEP-L^2, TVL, INFCON and TVBH models, deliver staircase artifact in the denoised signals as represented by the sub-figures

(e)-(h), whereas the pure second-order models, TL, BH, TGV and NewCv models, do not produce unfavorable staircase effect in the denoised signals as shown in the sub-figures (c), (d), (i) and (l). The staircase artifact can be observed from the sub-figure (j), whereas we see the oscillation in the smooth region of the denoised signal by GC model. According to Figures 6 and 8, the visual effects show our NewCv model gives the best smoothness-preservation result.

4.3. Comparison 3: signal-dependent test

In this subsection we compare the denoising performance of eleven different models in removing noise with different noise variances. The six real natural images in Figures 1 (d)-(i) are corrupted with five variance values from 0.01, 0.03, 0.05, 0.07 and 0.1.

Tables 1 - 5 provide the comparative results, which contain the two evaluation metrics

Table 1: Comparison of eleven different models in restoration quality on the six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with the noise variance 0.01.

	PSNR							SSIM						
	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average
TV	32.71	32.89	33.26	34.89	35.53	36.04	34.22	0.8997	0.9041	0.9083	0.9437	0.9377	0.9534	0.9245
TL	31.39	31.42	31.22	33.22	34.41	34.28	32.66	0.8781	0.8862	0.8905	0.9241	0.9252	0.9446	0.9081
BH	31.68	32.05	31.72	34.15	34.84	35.21	33.27	0.8801	0.8954	0.8922	0.9360	0.9310	0.9497	0.9141
CEP-L^2	33.36	33.87	34.16	35.98	36.43	36.89	35.11	0.9103	0.9210	0.9208	0.9512	0.9457	0.9606	0.9349
TVL	33.32	33.63	33.92	35.62	36.24	36.81	34.92	0.9096	0.9161	0.9178	0.9498	0.9445	0.9584	0.9327
INFCON	34.32	35.03	35.32	37.02	37.64	37.98	36.22	0.9238	0.9373	0.9335	0.9593	0.9558	0.9655	0.9459
TVBH	34.90	36.08	35.72	37.95	38.52	38.99	37.03	0.9305	0.9497	0.9395	0.9651	0.9624	0.9698	0.952
TGV	42.86	43.59	43.37	44.77	45.04	45.00	44.11	0.9861	0.9886	0.9804	0.9889	0.9881	0.9886	0.9868
MC	39.27	40.94	40.46	42.26	42.75	43.29	41.49	0.9719	0.9821	0.9710	0.9836	0.9830	0.9848	0.9794
GC	38.84	38.03	35.95	39.00	40.70	40.11	38.77	0.9619	0.9578	0.9341	0.9573	0.9673	0.9664	0.9575
NewCv	**47.18**	**46.21**	**45.57**	**47.42**	**47.89**	**47.41**	**46.95**	**0.9949**	**0.9946**	**0.9906**	**0.9941**	**0.9938**	**0.9936**	**0.9936**

Table 2: Comparison of eleven different models in restoration quality on the six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with the noise variance 0.03.

	PSNR							SSIM						
	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average
TV	32.59	32.81	33.22	34.75	35.38	35.73	34.08	0.8972	0.9031	0.9082	0.9408	0.9352	0.9497	0.9224
TL	31.29	31.33	31.15	33.09	34.26	34.07	32.53	0.8757	0.8850	0.8896	0.9218	0.9228	0.9419	0.9061
BH	31.64	32.02	31.70	34.09	34.79	35.10	33.22	0.8796	0.8955	0.8925	0.9355	0.9308	0.9490	0.9138
CEP-L^2	33.19	33.70	34.04	35.74	36.17	36.48	34.88	0.9077	0.9196	0.9203	0.9492	0.9430	0.9576	0.9329
TVL	33.14	33.53	33.80	35.46	36.07	36.48	34.75	0.9066	0.9154	0.9167	0.9473	0.9425	0.9554	0.9306
INFCON	34.22	34.89	35.21	36.83	37.45	37.67	36.05	0.9235	0.9371	0.9339	0.9584	0.9551	0.9644	0.9454
TVBH	34.74	35.88	35.59	37.71	38.27	38.62	36.80	0.9299	0.9492	0.9399	0.9643	0.9616	0.9688	0.9523
TGV	37.84	39.06	38.76	39.84	39.95	40.19	39.04	0.9639	0.9624	0.9628	0.9743	0.9720	0.9752	0.9675
MC	36.25	37.37	37.33	38.89	39.21	39.49	38.09	0.9480	0.9619	0.9532	0.9683	0.9655	0.9703	0.9612
GC	37.21	37.64	36.70	38.71	39.19	38.83	38.16	0.9520	0.9587	0.9412	0.9551	0.9584	0.9595	0.9551
NewCv	**39.29**	**39.36**	**39.73**	**40.68**	**41.14**	**40.72**	**40.15**	**0.9722**	**0.9760**	**0.9684**	**0.9750**	**0.9745**	**0.9768**	**0.9738**

Table 3: Comparison of eleven different models in restoration quality on the six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with the noise variance 0.05.

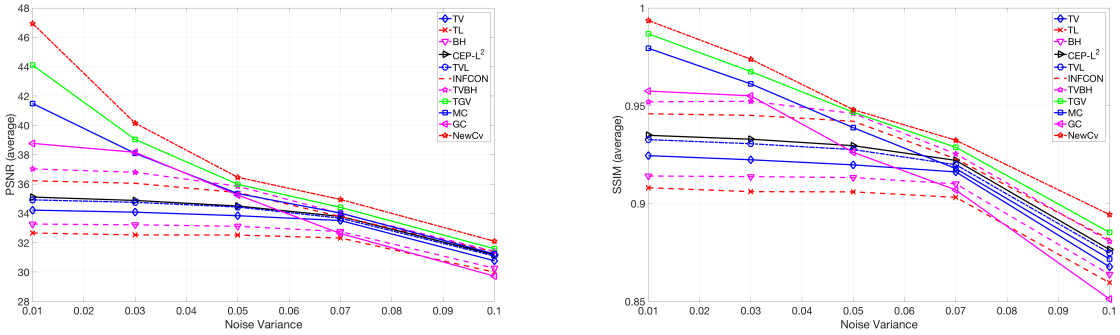
	PSNR							SSIM						
	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average
TV	32.39	32.65	33.13	34.53	35.12	35.26	33.84	0.8941	0.9017	0.9084	0.9371	0.9323	0.9451	0.9198
TL	31.39	31.61	31.47	33.20	34.16	33.28	32.52	0.8785	0.8937	0.8937	0.9186	0.9194	0.9321	0.9060
BH	31.58	31.95	31.64	33.98	34.68	34.91	33.12	0.8790	0.8955	0.8929	0.9347	0.9300	0.9478	0.9133
CEP-L^2	32.91	33.39	33.79	35.32	35.72	35.87	34.50	0.9043	0.9173	0.9189	0.9453	0.9386	0.9532	0.9296
TVL	32.91	33.27	33.67	35.13	35.72	35.90	34.43	0.9036	0.9129	0.9167	0.9431	0.9390	0.9506	0.9276
INFCON	34.00	34.58	34.93	35.89	36.48	36.65	35.42	0.9226	0.9358	0.9334	0.9527	0.9487	0.9595	0.9421
TVBH	34.40	35.41	35.22	36.22	36.80	37.07	35.85	0.9274	0.9464	0.9385	0.9538	0.9503	0.9604	0.9461
TGV	35.07	35.09	35.03	36.27	37.22	37.19	35.98	0.9381	0.9412	0.9340	0.9534	0.9538	0.9598	0.9467
MC	34.53	34.90	35.12	35.16	36.12	36.28	35.35	0.9297	0.9378	0.9352	0.9406	0.9401	0.9492	0.9388
GC	34.67	34.72	34.61	35.88	35.93	35.66	35.25	0.9246	0.9300	0.9236	0.9280	0.9211	0.9296	0.9262
NewCv	**35.16**	**35.75**	**35.87**	**37.01**	**37.73**	**37.22**	**36.46**	**0.9351**	**0.9448**	**0.9412**	**0.9550**	**0.9552**	**0.9565**	**0.9480**

Table 4: Comparison of eleven different models in restoration quality on the six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with the noise variance 0.07.

	PSNR							SSIM						
	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average
TV	32.12	32.40	32.94	34.18	34.75	34.66	33.51	0.8907	0.8995	0.9082	0.9319	0.9279	0.9391	0.9162
TL	31.12	31.18	31.01	32.87	34.00	33.73	32.32	0.8724	0.8834	0.8882	0.9182	0.9191	0.9381	0.9032
BH	31.47	31.82	31.53	33.77	34.48	34.51	32.76	0.8779	0.8947	0.8924	0.9325	0.9279	0.9354	0.9101
CEP-L^2	32.63	33.01	33.58	34.24	34.64	34.61	33.79	0.9007	0.9118	0.9175	0.9327	0.9273	0.9426	0.9221
TVL	32.41	32.75	33.17	34.35	34.65	34.71	33.67	0.8960	0.9062	0.9119	0.9353	0.9278	0.9423	0.9199
INFCON	32.72	33.09	33.65	34.28	34.54	34.42	33.78	0.9032	0.9134	0.9190	0.9359	0.9268	0.9390	0.9229
TVBH	32.95	33.44	33.80	34.43	34.60	34.71	33.99	0.9056	0.9187	0.9209	0.9373	0.9274	0.9423	0.9254
TGV	32.91	33.94	34.35	35.48	34.90	34.91	34.41	0.9073	0.9280	0.9235	0.9476	0.9314	0.9450	0.9288
MC	32.79	33.41	33.68	34.90	34.62	34.73	34.02	0.9027	0.9176	0.9084	0.9269	0.9197	0.9320	0.9179
GC	32.12	31.90	30.70	33.04	34.67	33.33	32.63	0.8952	0.9087	0.9003	0.9085	0.9047	0.9245	0.9070
NewCv	**33.71**	**34.30**	**34.71**	**35.68**	**35.68**	**35.67**	**34.95**	**0.9157**	**0.9295**	**0.9266**	**0.9413**	**0.9353**	**0.9459**	**0.9324**

Table 5: Comparison of eleven different models in restoration quality on the six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with the noise variance 0.1.

	PSNR							SSIM						
	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average	Barbara	Boat	Cameraman	Castle	Lena	Peppers	Average
TV	29.46	29.20	30.56	30.96	32.54	31.93	30.77	0.8279	0.8202	0.8667	0.8884	0.8953	0.9077	0.8677
TL	29.28	28.91	28.62	30.35	31.73	31.03	29.99	0.8269	0.8288	0.8442	0.8753	0.8790	0.9037	0.8596
BH	29.45	28.96	28.76	30.84	31.91	31.66	30.26	0.8285	0.8234	0.8420	0.8894	0.8853	0.9139	0.8638
CEP-L^2	29.73	29.67	31.27	31.45	33.04	31.93	31.18	0.8323	0.8352	0.8810	0.8956	0.9028	0.9123	0.8765
TVL	29.79	29.54	30.83	31.31	33.01	32.09	31.09	0.8364	0.8298	0.8725	0.8950	0.9034	0.9116	0.8748
INFCON	29.88	29.79	31.30	31.64	33.05	32.59	31.38	0.8381	0.8395	0.8826	0.9008	0.9046	0.9246	0.8817
TVBH	29.86	29.56	31.74	31.45	33.11	32.74	31.41	0.8366	0.8306	0.8908	0.8990	0.9059	0.9215	0.8807
TGV	29.89	29.73	31.88	31.57	33.47	33.00	31.59	0.8397	0.8397	0.8931	0.8987	0.9119	0.9288	0.8853
MC	29.69	29.49	31.61	30.85	33.09	32.74	31.24	0.8344	0.8414	0.8784	0.8834	0.8946	0.8978	0.8717
GC	29.28	28.31	29.30	29.41	32.03	30.02	29.72	0.8384	0.8394	0.8221	0.8525	0.8688	0.8861	0.8512
NewCv	**30.22**	**30.12**	**32.14**	**33.10**	**33.72**	**33.30**	**32.10**	**0.8554**	**0.8614**	**0.8989**	**0.9174**	**0.9116**	**0.9216**	**0.8944**



(a) (b)

Figure 9: Average denoising results by eleven different models on six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with five different noise variance (0.01, 0.03, 0.05, 0.07 and 0.1). (a) PSNR (average); (b) SSIM (average).

for quantitative comparison of eleven different models on the six real natural images with the five different noise variances. In order to easily assess these metrics in Tables 1 - 5, we draw the plots in Figure 9, where each plot in the sub-figures (a) and (b) corresponds to the average values of the denoised results produced by its corresponding model on different noise levels. The results in Figure 9 demonstrate that our NewCv model is the most effective for removing noise and preserving structural information with high values for all the quality metrics.

4.4. Comparison 4: computational efficiency test

In this experiment, the six real natural images in Figures 1(d)-(i) are used further to evaluate the computational efficiency of eleven different models. The test images are corrupted with the five different noise variances from 0.01, 0.03, 0.05, 0.07 and 0.1.

Table 6 demonstrates computational efficiency of different models on the noisy versions of the test images with the five different noise levels. In order to clearly compare the data in Table 6, we draw the bar charts in Figures 10, where each bar in the sub-figures (a)-(c) corresponds to the average value of the denoised results delivering by its corresponding model on the noisy test images.

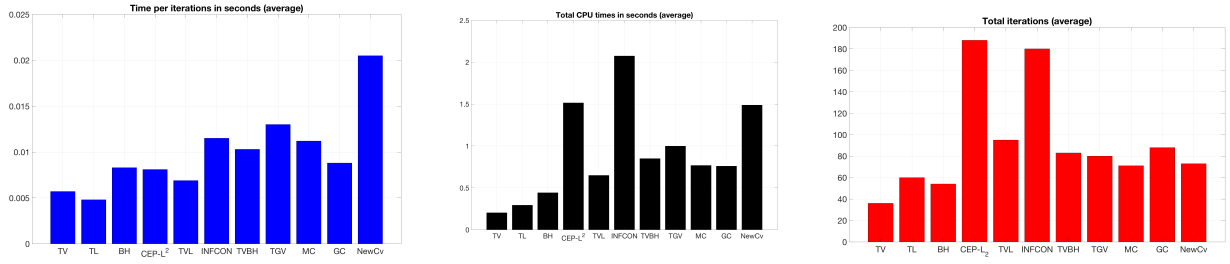
In terms of computation times per iteration, we can see that our NewCv model requires to solve five sub-problems in each step. Thus, in accordance with our measurements, we expect that more computation time per iteration is required for the proposed model than those of the other ten models. As can be seen from the sub-figure (a), our numerical experiments reflect this expectation.

Table 6: Comparison of computational efficiency by eleven different models on the six real natural images in Figure 1 (d)-(i), which are distorted by the Gaussian noise with five different noise variances from0.01, 0.03, 0.05, 0.07 and 0.1.

Noise variance		TV	TL	BH	CEP-L^2	TVL	INFCON	TVBH	TGV	MC	GC	NewCv
	Time per iteration (s)	0.0057	0.0050	0.0085	0.0082	0.0070	0.0118	0.0104	0.0144	0.0119	0.0092	0.0212
0.01	Total CPU time (s)	0.2082	0.2871	0.4541	1.4637	0.5508	2.1377	0.7350	0.5277	0.3808	0.3772	0.7420
	Total iterations	37	58	54	179	78	181	71	37	32	41	35
	Time per iteration (s)	0.0057	0.0048	0.0081	0.0080	0.0068	0.0114	0.0101	0.0130	0.0111	0.0099	0.0199
0.03	Total CPU time (s)	0.1957	0.2824	0.4348	1.4001	0.5355	1.9838	0.7142	0.9540	0.7770	0.8217	0.8557
	Total iterations	35	59	54	175	79	176	71	75	70	83	43
	Time per iteration (s)	0.0059	0.0051	0.0085	0.0083	0.0071	0.0121	0.0110	0.0135	0.0115	0.0081	0.0199
0.05	Total CPU time (s)	0.1930	0.3149	0.4538	1.4179	0.5691	2.2812	0.7882	0.6365	0.4750	0.7371	1.5366
	Total iterations	33	61	54	170	80	187	71	47	41	91	78
	Time per iteration (s)	0.0057	0.0047	0.0082	0.0079	0.0067	0.0112	0.0099	0.0118	0.0107	0.0081	0.0214
0.07	Total CPU time (s)	0.1762	0.3038	0.4374	2.0557	1.0614	1.8395	1.4814	1.4568	1.0914	0.9234	2.0758
	Total iterations	31	65	54	262	159	165	151	125	102	114	97
	Time per iteration (s)	0.0055	0.0047	0.0081	0.0080	0.0069	0.0112	0.0103	0.0120	0.0108	0.0087	0.0201
0.1	Total CPU time (s)	0.2442	0.2762	0.4314	1.2337	0.5283	2.1278	0.5327	1.4183	1.1180	0.9396	2.2310
	Total iterations	45	58	53	155	77	190	51	118	110	108	111

In terms of total CPU times, the sub-figure (b) illustrates that the SB algorithm of INFCON model is the slowest method, while that of TV is the fastest method. Moreover, the SB algorithms of CEP-L^2, TVL, INFCON, TVBH, TGV, MC, GC and NewCv models are slower than those of TL and BH models.

In terms of total iterations, the sub-figure (c) shows that the SB algorithms of CEP-L^2 and INFCON take more than 175 iterations to converge, while the other nine algorithms take less than 100 iterations. It is important to note that our SB algorithm for the proposed model is slightly better than those of TGV, MC and GC models.



(a) (b) (c)

Figure 10: Bar charts of the experimental data in Table 6 (average denoising results by eleven different models on the noisy images of the six real natural images in Figure 1 (d)-(i) with five different noise variance (0.01, 0.03, 0.05, 0.07 and 0.1). (a) Time per iteration in seconds (average); (b) total CPU times in seconds (average); (c) total iterations (average).

4.5. Comparison 5: real noisy image test

In the last test, in order to further show the denoising performance of our proposed model, we extend to a real world application, which is clinical image, and compare our NewCv model against the other eleven models on three real clinical images as shown in Figures 1 (j)-(l).

In Figures 11, 13 and 15 we present the denoised images by applying eleven different models. For better visualization, we include the corresponding plots of the middle slices of the denoised images in Figures 12, 14 and 16.

As the clean image is unknown, the denoising performance is evaluated by visual inspection. First, as expected, one can observe the staircase effect in the denoised results by TV model as shown in Figures 11 - 16. Next, the denoised images by TL and BH models in the sub-figures (c) and (d) of Figures 11, 13 and 15 seem to be good in term of visual quality, but TL and BH models tend to oversmooth certain features, which wipe out important details of the image structure. Moreover, the sub-figures (e)-(k) of Figures 11, 13 and 15 present a better performance compared with those shown in the sub-figures (c) and (d). Finally, the denoised results by our NewCv model shown in Figures 11 - 16, especially in the zoomed parts, are superior to the other eleven models for preserving edges and reducing the staircase effect.

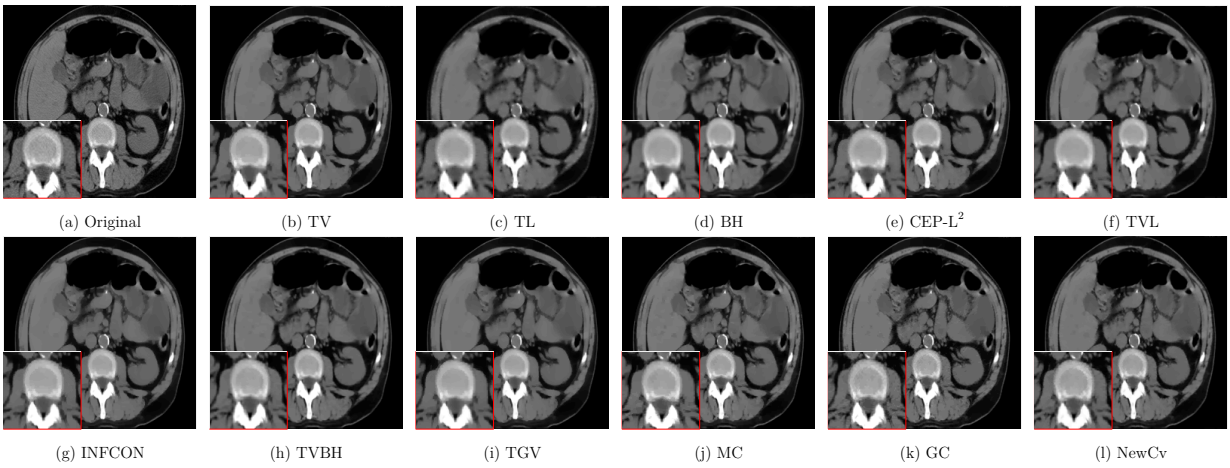
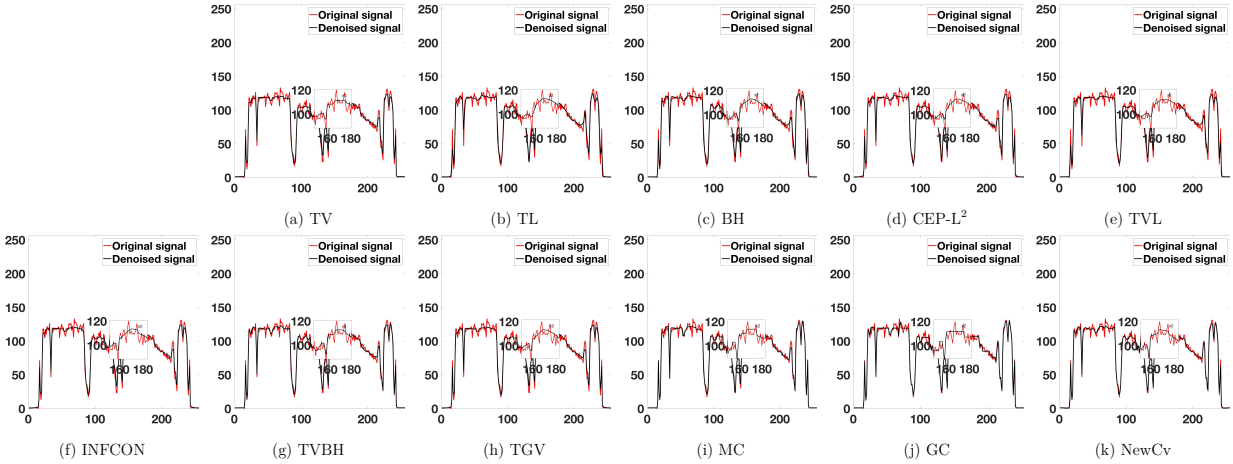


Figure 11: Comparison of the real "Abdomen" image by using 11 different models.

5. Conclusion

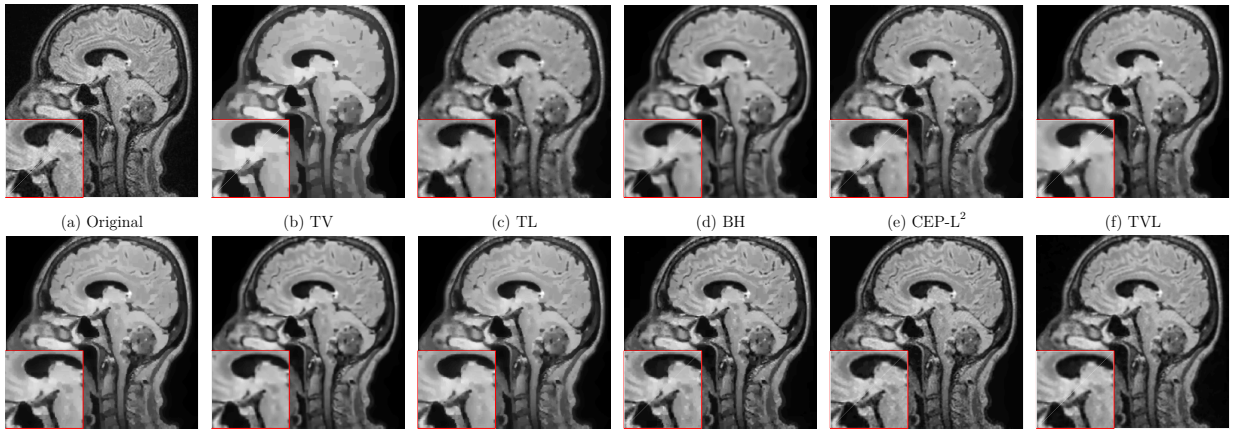
In this paper, a new high-order variational model to restore noisy images corrupted by Gaussian noise was proposed. Based on the good feature of high-order regularization, we proposed a curvature-based regularization. In order to solve the new model, the split Bregman algorithm was developed. Compared with some state-of-the-art models, experimental



(a) TV (b) TL (c) BH (d) CEP-L^2 (e) TVL

(f) INFCON (g) TVBH (h) TGV (i) MC (j) GC (k) NewCv

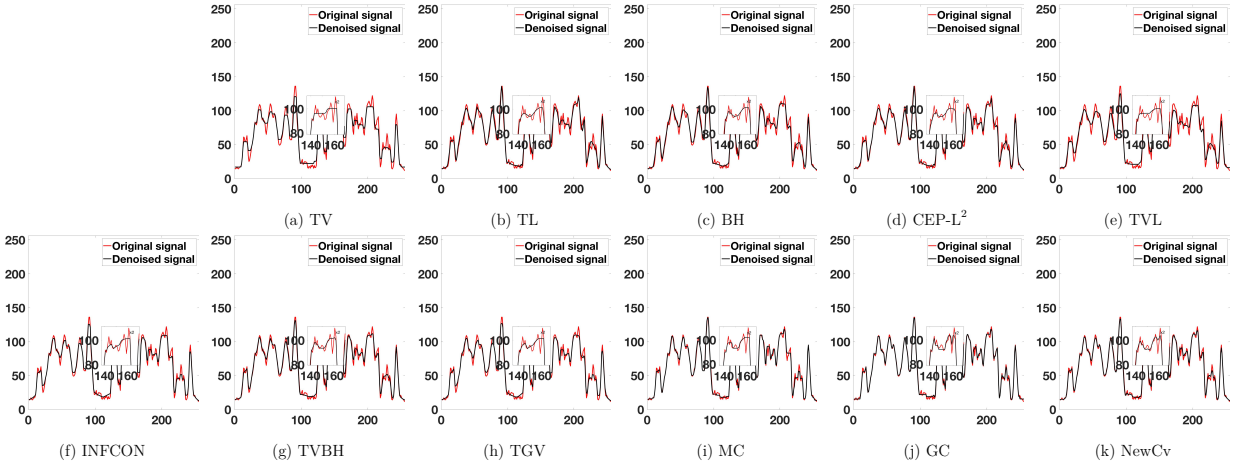
Figure 12: Plots of middle slices of the denoised images in Figure 11 "Abdomen" by eleven different model.



(a) Original (b) TV (c) TL (d) BH (e) CEP-L^2 (f) TVL

(g) INFCON (h) TVBH (i) TGV (j) MC (k) GC (l) NewCv

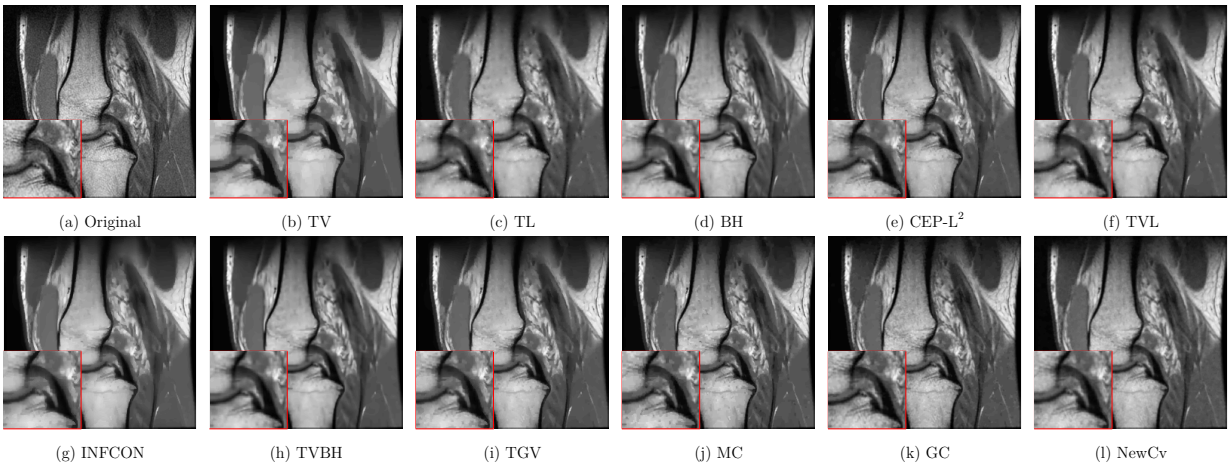
Figure 13: Comparison of the real "Brain" image by using 11 different models.



(a) TV (b) TL (c) BH (d) CEP-L^2 (e) TVL

(f) INFCON (g) TVBH (h) TGV (i) MC (j) GC (k) NewCv

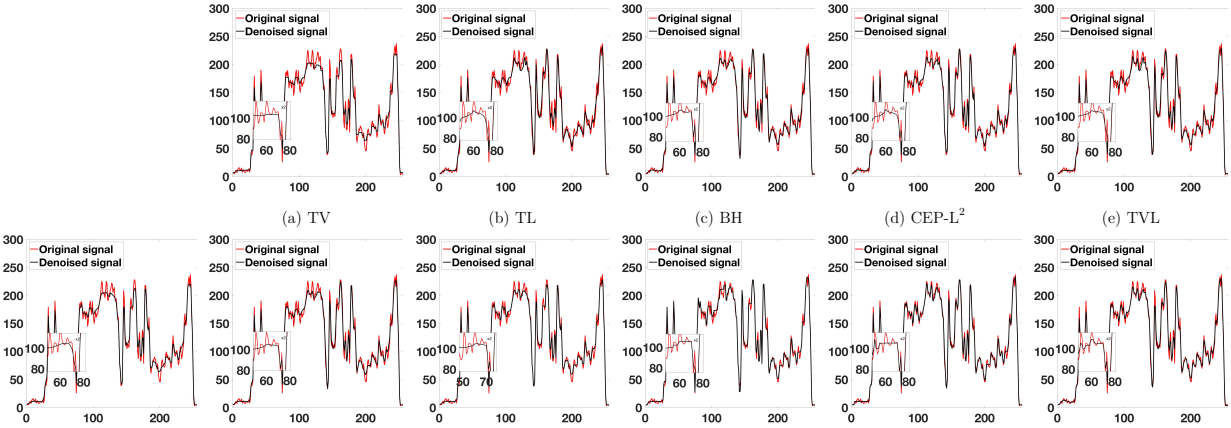
Figure 14: Plots of middle slices of the denoised images in Figure 13 "Brain" by eleven different models.



(a) Original (b) TV (c) TL (d) BH (e) CEP-L^2 (f) TVL

(g) INFCON (h) TVBH (i) TGV (j) MC (k) GC (l) NewCv

Figure 15: Comparison of the real "Knee" image by using eleven different models.



(a) TV (b) TL (c) BH (d) CEP-L^2 (e) TVL

(f) INFCON (g) TVBH (h) TGV (i) MC (j) GC (k) NewCv

Figure 16: Plots of middle slices of the denoised images in Figure 15 "Knee" by eleven different models.

results revealed that the new model effectively preserves edges and reduces the staircase effect.

Acknowledgments

The authors would like to thank the anonymous reviewers for their useful suggestions to improve the quality of the paper. The second author's work was partially supported by the Thailand Research Fund Grant ♯MRG6080169.

Appendix A – Derivation of the Euler-Lagrange Equation for (17)

Consider the proposed model in (17)

$$\min_u \mathcal{J}_{\mathrm{AN}}^{\mathrm{NewCv}}(u)$$

where

$$\mathcal{J}_{\mathrm{AN}}^{\mathrm{NewCv}}(u) = \mathcal{D}_{\mathrm{AN}}(u, z) + \gamma \mathcal{R}^{\mathrm{NewCv}}(u).$$

In order to compute the first variation of $\mathcal{J}_{\mathrm{AN}}^{\mathrm{NewCv}}$ for u in a direction of v in a test space $\mathcal{V}$, we will find the first variations of $\mathcal{D}_{\mathrm{AN}}$, $\mathcal{R}_1$ and $\mathcal{R}_2$ respect to u, and apply the sum rule.

The first variation of $\mathcal{D}_{\mathrm{AN}}$ for u in the direction of v is given by

$$\begin{aligned}
\frac{\delta \mathcal{D}_{\mathrm{AN}}(u, z; v)}{\delta u} &= \left. \frac{d\mathcal{D}_{\mathrm{AN}}((u + \varepsilon v), z)}{d\varepsilon} \right|_{\epsilon=0} \\
&= \left. \frac{d}{d\varepsilon} \left\{ \frac{1}{2} \int_\Omega \left((u + \varepsilon v) - z \right)^2 \, d\Omega \right\} \right|_{\varepsilon=0}, \\
&= \left. \frac{1}{2} \int_\Omega \frac{d}{d\varepsilon} \left((u + \varepsilon v) - z \right)^2 \right|_{\varepsilon=0} d\Omega, \\
&= \int_\Omega (u - z)\, v \, d\Omega. \qquad (64)
\end{aligned}$$

Let $\mathcal{R}_1(u) = \frac{1}{2} \int_\Omega \kappa_{\mathrm{M}}^2(u) d\Omega$ and $\mathcal{R}_2(u) = \frac{1}{2} \int_\Omega -2\kappa_{\mathrm{G}}(u) d\Omega$. Therefore

$$\mathcal{R}^{\mathrm{NewCv}}(u) = \mathcal{R}_1(u) + \mathcal{R}_2(u).$$

Thus, we have

$$\begin{aligned}
\frac{\delta \mathcal{R}_1(u; v)}{\delta u} &= \frac{d}{d\varepsilon} \left(\frac{1}{2} \int_\Omega \kappa_{\mathrm{M}}^2(u + \varepsilon v) \, d\Omega \right) \Big|_{\varepsilon=0} \\
&= \frac{1}{2} \int_\Omega \left. \frac{d}{d\varepsilon} \kappa_{\mathrm{M}}^2(u + \varepsilon v) \right|_{\varepsilon=0} d\Omega \\
&= \frac{1}{2} \int_\Omega 2\kappa_{\mathrm{M}}(u) \left. \frac{d}{d\varepsilon} (\kappa_{\mathrm{M}}(u + \varepsilon v)) \right|_{\varepsilon=0} d\Omega \\
&= \int_\Omega \kappa_{\mathrm{M}}(u) \left. \frac{d}{d\varepsilon} (\nabla \cdot \frac{\nabla(u + \varepsilon v)}{\sqrt{1 + |\nabla(u + \varepsilon v)|^2}}) \right|_{\varepsilon=0} d\Omega \\
&= \int_\Omega \kappa_{\mathrm{M}}(u) (\nabla \cdot \left. \frac{d}{d\varepsilon} \, \frac{\nabla(u + \varepsilon v)}{\sqrt{1 + |\nabla(u + \varepsilon v)|^2}}) \right|_{\varepsilon=0} d\Omega
\end{aligned}$$

or

$$\begin{aligned}
\frac{\delta \mathcal{R}_1(u;v)}{\delta u} &= \int_\Omega \kappa_{\mathrm{M}}(u) \left(\nabla \cdot \left(\frac{1}{\sqrt{1+|\nabla(u+\varepsilon v)|^2)}} \frac{d}{d\epsilon} \nabla(u+\epsilon v) \Big|_{\epsilon=0} \right.\right. \\
&\qquad \left.\left. + \nabla(u+\epsilon v) \frac{d}{d\epsilon} \frac{1}{\sqrt{1+|\nabla(u+\varepsilon v)|^2)}} \Big|_{\epsilon=0} \right) \right) d\Omega \\
&= \int_\Omega \kappa_{\mathrm{M}}(u) \left(\nabla \cdot \left(\frac{\nabla v}{\sqrt{1+|\nabla u|^2)}} \right.\right. \\
&\qquad + \nabla u \sum_{m=1}^{2} \frac{\partial [1+(u_{x_1}+\epsilon v_{x_1})^2 + (u_{x_2}+\epsilon v_{x_2})^2]^{-1/2}}{\partial (u_{x_m}+\epsilon v_{x_m})} \times \\
&\qquad \left.\left. \frac{\partial (u_{x_m}+\epsilon v_{x_m})^2}{\partial (u_{x_m}+\epsilon v_{x_m})} \frac{\partial (u_{x_m}+\epsilon v_{x_m})}{\partial \epsilon} \Big|_{\epsilon=0} \right) \right) d\Omega \\
&= \int_\Omega \kappa_{\mathrm{M}}(u) (\nabla \cdot (\frac{\nabla v}{\sqrt{1+|\nabla u|^2}} - \nabla u \sum_{m=1}^{2} \frac{u_{x_m} v_{x_m}}{\sqrt{1+|\nabla u|^2}^3})) d\Omega \\
&= \int_\Omega \kappa_{\mathrm{M}}(u) (\nabla \cdot (\frac{\nabla v}{\sqrt{1+|\nabla u|^2}} - (\nabla u \frac{\nabla u}{(\sqrt{1+|\nabla u|^2})^3}) \nabla v)) d\Omega.
\end{aligned}$$

After applying the divergence theorem, we get

$$\begin{aligned}
\frac{\delta \mathcal{R}_1(u;v)}{\delta u} &= \int_{\partial\Omega} \kappa_{\mathrm{M}}(u) \langle \frac{\nabla v}{\sqrt{1+|\nabla u|^2}} - (\nabla u \frac{\nabla u}{(\sqrt{1+|\nabla u|^2})^3}) \nabla v, \boldsymbol{n} \rangle d\boldsymbol{s} \\
&\quad - \int_\Omega \langle \kappa_{\mathrm{M}}(u), \frac{\nabla v}{\sqrt{1+|\nabla u|^2}} - (\nabla u \frac{\nabla u}{(\sqrt{1+|\nabla u|^2})^3}) \nabla v \rangle d\Omega \\
&= \int_{\partial\Omega} \kappa_{\mathrm{M}}(u) \langle \frac{1}{\sqrt{1+|\nabla u|^2}} (\mathbf{I} - (\nabla u \frac{\nabla u}{|\nabla u|^2})) \nabla v, \boldsymbol{n} \rangle d\boldsymbol{s} \\
&\quad - \int_\Omega \langle \nabla(\kappa_{\mathrm{M}}(u), \frac{1}{\sqrt{1+|\nabla u|^2}} (\mathbf{I} - (\nabla u \frac{\nabla u}{(\sqrt{1+|\nabla u|^2})^2})) \nabla v \rangle d\Omega \\
&= \int_{\partial\Omega} \kappa_{\mathrm{M}}(u) \langle \frac{1}{\sqrt{1+|\nabla u|^2}} (\mathbf{I} - \mathbf{P}) \nabla v, \boldsymbol{n} \rangle d\boldsymbol{s} \\
&\quad - \int_\Omega \langle \frac{1}{\sqrt{1+|\nabla u|^2}} (\mathbf{I} - \mathbf{P}) \nabla(\kappa_{\mathrm{M}}(u)), \nabla v \rangle d\mathbf{x}.
\end{aligned}$$

Here $\mathbf{I}(\mathbf{x}) = \mathbf{x}$ is the identity transform and $\mathbf{P}(\mathbf{x}) = (\mathbf{x} \cdot \frac{\nabla u}{\sqrt{1+|\nabla u|^2}}) \frac{\nabla u}{\sqrt{1+|\nabla u|^2}}$ is the orthogonal projection onto the normal direction. By using the divergence theorem with the second term, we have

$$\begin{aligned}
\frac{\delta \mathcal{R}_1(u;v)}{\delta u} &= \int_{\partial\Omega} \kappa_{\mathrm{M}}(u) \langle \frac{1}{\sqrt{1+|\nabla u|^2}} (\mathbf{I} - \mathbf{P}) \nabla v, \boldsymbol{n} \rangle \; d\boldsymbol{s} \\
&\quad + \int_\Omega (\nabla \cdot (\frac{1}{\sqrt{1+|\nabla u|^2}} (\mathbf{I} - \mathbf{P}) \nabla \kappa_{\mathrm{M}}(u)))) v \; d\Omega \\
&\quad - \int_{\partial\Omega} \langle \frac{1}{\sqrt{1+|\nabla u|^2}} (\mathbf{I} - \mathbf{P}) \nabla(\kappa_{\mathrm{M}}(u), \boldsymbol{n} \rangle v d\boldsymbol{s}
\end{aligned}$$

or

$$\frac{\delta \mathcal{R}_1(u;v)}{\delta u} = \int_\Omega (\nabla \cdot (\frac{1}{\sqrt{1+|\nabla u|^2}} \nabla \kappa_{\mathrm{M}}(u) - \frac{\nabla u \cdot \nabla \kappa_{\mathrm{M}}(u)}{(\sqrt{1+|\nabla u|^2})^3} \nabla u)) v d\Omega + \int_{\partial\Omega} \kappa_{\mathrm{M}}(u) \langle \frac{1}{\sqrt{1+|\nabla u|^2}} (\mathbf{I} - \mathbf{P}) \nabla v, \boldsymbol{n} \rangle d\boldsymbol{s} - \int_{\partial\Omega} \langle \frac{1}{\sqrt{1+|\nabla u|^2}} (\mathbf{I} - \mathbf{P}) \nabla (\kappa_{\mathrm{M}}(u), \boldsymbol{n} \rangle v d\boldsymbol{s}. \tag{65}$$

For $\mathcal{R}_2$, we have

$$\begin{aligned}
\frac{\delta \mathcal{R}_2(u;v)}{\delta u} &= -\frac{d}{d\varepsilon} \int_\Omega \kappa_{\mathrm{G}}(u+\varepsilon v)\, d\Omega \Big|_{\varepsilon=0} \\
&= -\int_\Omega \frac{d}{d\varepsilon} \kappa_{\mathrm{G}}(u+\varepsilon v) \Big|_{\varepsilon=0} d\Omega \\
&= -\int_\Omega \frac{d}{d\varepsilon} \frac{(u+\varepsilon v)_{x_1x_1}(u+\varepsilon v)_{x_2x_2} - (u+\varepsilon v)_{x_1x_2}(u+\varepsilon v)_{x_2x_1}}{(1+|\nabla(u+\varepsilon v)|^2)^2} \Big|_{\varepsilon=0} d\Omega \\
&= \int_\Omega \frac{d}{d\varepsilon} \frac{(u+\varepsilon v)_{x_1x_2}(u+\varepsilon v)_{x_2x_1} - (u+\varepsilon v)_{x_1x_1}(u+\varepsilon v)_{x_2x_2}}{(1+|\nabla(u+\varepsilon v)|^2)^2} \Big|_{\varepsilon=0} d\Omega \\
&= \Big[\int_\Omega \frac{u_{x_1x_2}v_{x_2x_1} - u_{x_1x_1}v_{x_2x_2} + u_{x_2x_1}v_{x_1x_2} - u_{x_2x_2}v_{x_1x_1}}{(1+|\nabla(u)|^2)^2}\, d\Omega \\
&\quad - \int_\Omega \frac{4(u_{x_1x_2}u_{x_2x_1} - u_{x_1x_1}u_{x_2x_2})(u_{x_1}v_{x_1} + u_{x_2}v_{x_2})}{(1+|\nabla(u)|^2)^3}\, d\Omega \Big] \\
&= \Big[\int_\Omega \frac{u_{x_1x_2}v_{x_2x_1} - u_{x_1x_1}v_{x_2x_2}}{(1+|\nabla(u)|^2)^2}\, d\Omega + \int_\Omega \frac{u_{x_2x_1}v_{x_1x_2} - u_{x_2x_2}v_{x_1x_1}}{(1+|\nabla(u)|^2)^2}\, d\Omega \\
&\quad - \int_\Omega \frac{4(u_{x_1x_2}u_{x_2x_1} - u_{x_1x_1}u_{x_2x_2})(u_{x_1}v_{x_1} + u_{x_2}v_{x_2})}{(1+|\nabla(u)|^2)^3}\, d\Omega \Big].
\end{aligned} \tag{66}$$

Applying the divergence theorem and integrating by parts yields

$$\begin{aligned}
&\int_\Omega \frac{u_{x_1x_2}v_{x_2x_1} - u_{x_1x_1}v_{x_2x_2}}{(1+|\nabla(u)|^2)^2}\, d\Omega + \int_\Omega \frac{u_{x_2x_1}v_{x_1x_2} - u_{x_2x_2}v_{x_1x_1}}{(1+|\nabla(u)|^2)^2}\, d\Omega \\
&= \Big[\int_{\partial\Omega} \langle (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2}, \frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2}) v_{x_2}, \boldsymbol{n} \rangle d\boldsymbol{s} \\
&\quad - \int_\Omega ((\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1} + (\frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2}) v_{x_2} d\Omega \\
&\quad + \int_{\partial\Omega} \langle (\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2}, \frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2}) v_{x_1}, \boldsymbol{n} \rangle d\boldsymbol{s} \\
&\quad - \int_\Omega ((\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1} + (\frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2}) v_{x_1} d\Omega \Big]
\end{aligned}$$

or

$$\begin{aligned}
&\int_\Omega \frac{u_{x_1x_2}v_{x_2x_1} - u_{x_1x_1}v_{x_2x_2}}{(1+|\nabla(u)|^2)^2}\, d\Omega + \int_\Omega \frac{u_{x_2x_1}v_{x_1x_2} - u_{x_2x_2}v_{x_1x_1}}{(1+|\nabla(u)|^2)^2}\, d\Omega \\
&= \Bigg[\int_{\partial\Omega} \langle (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2}, \frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})v_{x_2}, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad + \int_{\partial\Omega} \langle (\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2}, \frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})v_{x_1}, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad - \int_\Omega ((\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1})v_{x_1} + ((\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1})v_{x_2} d\Omega \\
&\quad + \int_\Omega ((\frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2})v_{x_1} + ((\frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2})v_{x_2} d\Omega\Bigg]
\end{aligned} \tag{67}$$

Applying the divergence theorem with the last two terms of (67) lead to

$$\begin{aligned}
&\int_\Omega \frac{u_{x_1x_2}v_{x_2x_1} - u_{x_1x_1}v_{x_2x_2}}{(1+|\nabla(u)|^2)^2}\, d\Omega + \int_\Omega \frac{u_{x_2x_1}v_{x_1x_2} - u_{x_2x_2}v_{x_1x_1}}{(1+|\nabla(u)|^2)^2}\, d\Omega \\
&= \Bigg[\int_{\partial\Omega} \langle (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2}, \frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})v_{x_2}, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad + \int_{\partial\Omega} \langle (\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2}, \frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})v_{x_1}, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad + \int_\Omega ((\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1x_1}) + (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1x_2})v d\Omega \\
&\quad - \int_{\partial\Omega} \langle ((\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1}, (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1})v, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad + \int_\Omega ((\frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2x_1}) + (\frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2x_2})v d\Omega \\
&\quad - \int_{\partial\Omega} \langle ((\frac{-u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2}, (\frac{u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2})v, \boldsymbol{n}\rangle d\boldsymbol{s}\Bigg]
\end{aligned}$$

or

$$\begin{aligned}
&\int_\Omega \frac{u_{x_1x_2}v_{x_2x_1} - u_{x_1x_1}v_{x_2x_2}}{(1+|\nabla(u)|^2)^2}\, d\Omega + \int_\Omega \frac{u_{x_2x_1}v_{x_1x_2} - u_{x_2x_2}v_{x_1x_1}}{(1+|\nabla(u)|^2)^2}\, d\Omega \\
&= \Bigg[\int_{\partial\Omega} \langle (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2}, \frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})v_{x_2}, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad + \int_{\partial\Omega} \langle (\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2}, \frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})v_{x_1}, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad - \int_{\partial\Omega} \langle ((\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1}, (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1})v, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad - \int_{\partial\Omega} \langle ((\frac{-u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2}, (\frac{u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2})v, \boldsymbol{n}\rangle d\boldsymbol{s} \\
&\quad + \int_\Omega \langle (\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1x_1} + (\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1x_2}, v\rangle d\Omega \\
&\quad + \int_\Omega \langle (\frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2x_1} + (\frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2x_2}, v\rangle d\Omega\Bigg].
\end{aligned} \tag{68}$$

Similarly, we have

$$
\begin{aligned}
&-\int_{\Omega}\frac{4(u_{x_1x_2}u_{x_2x_1}-u_{x_1x_1}u_{x_2x_2})(u_{x_1}v_{x_1}+u_{x_2}v_{x_2})}{(1+|\nabla(u)|^2)^3}\,d\Omega\\
&\quad=\Bigg[\int_{\Omega}\langle\nabla\cdot(\frac{4(u_{x_1x_2}u_{x_2x_1}-u_{x_1x_1}u_{x_2x_2})}{(1+|\nabla(u)|^2)^3})\nabla u,v\rangle d\Omega\\
&\quad-\int_{\partial\Omega}\langle(\frac{4(u_{x_1x_2}u_{x_2x_1}-u_{x_1x_1}u_{x_2x_2})}{(1+|\nabla(u)|^2)^3}\nabla u)v,\boldsymbol{n}\rangle d\boldsymbol{s}\Bigg].
\end{aligned}\tag{69}
$$

Substituting (68) and (69) into (66) and rearranging the terms yield

$$
\begin{aligned}
\frac{\delta}{\delta u}\mathcal{R}_2(u;v)=&\int_{\partial\Omega}\frac{1}{(1+|\nabla(u)|^2)^2}v_{x_2}\langle(u_{x_1x_2},-u_{x_1x_1}),\boldsymbol{n}\rangle d\boldsymbol{s}\\
&+\int_{\partial\Omega}\frac{1}{(1+|\nabla(u)|^2)^2}v_{x_1}\langle(-u_{x_2x_2},u_{x_2x_1}),\boldsymbol{n}\rangle d\boldsymbol{s}\\
&+\int_{\partial\Omega}\langle((\frac{u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1},(\frac{-u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1}),\boldsymbol{n}\rangle v d\boldsymbol{s}\\
&+\int_{\partial\Omega}\langle((\frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2},-(\frac{u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2}),\boldsymbol{n}\rangle v d\boldsymbol{s}\\
&+\int_{\Omega}\nabla\cdot((\frac{-u_{x_2x_2}}{(1+|\nabla(u)|^2)^2})_{x_1},(\frac{u_{x_1x_2}}{(1+|\nabla(u)|^2)^2})_{x_1})v d\Omega\\
&+\int_{\Omega}\nabla\cdot((\frac{u_{x_2x_1}}{(1+|\nabla(u)|^2)^2})_{x_2},(\frac{-u_{x_1x_1}}{(1+|\nabla(u)|^2)^2})_{x_2})v d\Omega\\
&+\int_{\Omega}(\nabla\cdot\left(\frac{4(u_{x_1x_2}u_{x_2x_1}-u_{x_1x_1}u_{x_2x_2})}{(1+|\nabla(u)|^2)^3}\nabla u\right)v d\Omega\\
&-\int_{\partial\Omega}\langle\frac{4(u_{x_1x_2}u_{x_2x_1}-u_{x_1x_1}u_{x_2x_2})}{(1+|\nabla(u)|^2)^3}\nabla u,\boldsymbol{n}\rangle v d\boldsymbol{s}.
\end{aligned}\tag{70}
$$

By dropping the boundary terms, a function u minimizing the joint functional $\mathcal{J}_{\mathrm{AN}}^{\mathrm{NewCv}}(u)$ in (17) has to satisfy the following partial differential equation

$$f(u,z)+\gamma\nabla\cdot\boldsymbol{V}=0,$$

subject to the natural boundary conditions

$$
\begin{aligned}
\kappa_{\mathrm{M}}(u)&=0,\\
-\frac{\partial\kappa_{\mathrm{M}}(u)}{\partial\boldsymbol{n}}+\frac{\nabla\kappa_{\mathrm{M}}(u)\cdot\nabla u}{(1+|\nabla u|^2)^2}\frac{\partial u}{\partial\boldsymbol{n}}&=0,\\
(u_{x_1x_2},-u_{x_1x_1})\cdot\boldsymbol{n}&=0,\\
(-u_{x_2x_2},u_{x_2x_1})\cdot\boldsymbol{n}&=0,\\
((\frac{u_{x_2x_2}}{(1+|\nabla u|^2)^2})_{x_1},-(\frac{u_{x_1x_2}}{(1+|\nabla u|^2)^2}))_{x_1}\cdot\boldsymbol{n}&=0,\\
((\frac{u_{x_2x_1}}{(1+|\nabla u|^2)^2})_{x_2},-(\frac{u_{x_1x_1}}{(1+|\nabla u|^2)^2}))_{x_2}\cdot\boldsymbol{n}&=0,\\
\frac{4(u_{x_1x_2}u_{x_2x_1}-u_{x_1x_1}u_{x_2x_2})}{(1+|\nabla u|^2)^3}\frac{\partial u}{\partial\boldsymbol{n}}&=0,
\end{aligned}
$$

where $\boldsymbol{V} = \boldsymbol{V}_1 + \boldsymbol{V}_2 + \boldsymbol{V}_3 + \boldsymbol{V}_4 + \boldsymbol{V}_5$,

$$\boldsymbol{V}_1 = \frac{1}{\sqrt{1+|\nabla u|^2}}\nabla\kappa_{\mathrm{M}}(u),\ \boldsymbol{V}_2 = -\frac{\nabla u\cdot\nabla\kappa_{\mathrm{M}}(u)}{(\sqrt{1+|\nabla u|^2})^3}\nabla u,$$
$$\boldsymbol{V}_3 = \frac{4(u_{x_1x_2}u_{x_2x_1} - u_{x_1x_1}u_{x_2x_2})}{(1+|\nabla u|^2)^3}\nabla u,\ \boldsymbol{V}_4 = ((-\frac{u_{x_2x_2}}{(1+|\nabla u|^2)^2})_{x_1}, (\frac{u_{x_1x_2}}{(1+|\nabla u|^2)^2})_{x_1}),$$
$$\boldsymbol{V}_5 = ((\frac{u_{x_2x_1}}{(1+|\nabla u|^2)^2})_{x_2}, (-\frac{u_{x_1x_1}}{(1+|\nabla u|^2)^2})_{x_2})$$

Appendix B – Derivation of the Euler-Lagrange Equation for $\boldsymbol{p}$-subproblem

Here our aim is to derive the Euler-Lagrange equation for $\boldsymbol{p}$-subproblem.

$$\min_{\boldsymbol{p}} \overline{\mathcal{E}}\,(\boldsymbol{p})\,,$$

where

$$\overline{\mathcal{E}}\,(\boldsymbol{p}) = \overline{\mathcal{E}}\,(p_1,p_2) = -\gamma\int_\Omega \frac{p_{1_x}p_{2_y} - p_{1_y}p_{2_x}}{(p_1^2+p_2^2+1)^2}d\Omega + \frac{\theta_1}{2}\int_\Omega (\boldsymbol{p} - \nabla u - \boldsymbol{b_1})^2 d\Omega$$
$$+ \frac{\theta_2}{2}\int_\Omega (\boldsymbol{m} - \frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2+1}} - \boldsymbol{b_2})^2 d\Omega,$$

$\boldsymbol{p} = (p_1, p_2)$, $\boldsymbol{m} = (m_1, m_2)$, $\boldsymbol{b}_1 = (b_{11}, b_{12})$ and $\boldsymbol{b}_2 = (b_{21}, b_{22})$.

Let $\mathcal{V}$ be a test space. For any perturbation $\varphi_1 \in \mathcal{V}$, the first variation of $\overline{\mathcal{E}}$ for p_1 in the direction of φ_1 is given by

$$\begin{aligned}
\frac{\delta\overline{\mathcal{E}}(p_1,p_2;\varphi_1)}{\delta p_1} &= \left.\frac{d\overline{\mathcal{E}}((p_1+\varepsilon_1\varphi_1),p_2)}{d\varepsilon_1}\right|_{\epsilon_1=0} \\
&= \frac{d}{d\varepsilon_1}\left\{-\gamma\int_\Omega \frac{(p_1+\varepsilon_1\varphi_1)_x p_{2_y} - (p_1+\varepsilon_1\varphi_1)_y p_{2_x}}{((p_1+\varepsilon_1\varphi_1)^2+p_2^2+1)^2}d\Omega\right. \\
&\quad + \frac{\theta_1}{2}\int_\Omega ((p_1+\varepsilon_1\varphi_1) - u_x - b_{11})^2 d\Omega \\
&\quad \left. + \frac{\theta_2}{2}\int_\Omega (m_1 - \frac{(p_1+\varepsilon_1\varphi_1)}{\sqrt{(p_1+\varepsilon_1\varphi_1)^2+p_2^2+1}} - b_{21})^2 d\Omega\right\}\Bigg|_{\varepsilon_1=0}.
\end{aligned}$$

Thus, we have

$$\begin{aligned}
\frac{\delta\overline{\mathcal{E}}(p_1,p_2;\varphi_1)}{\delta p_1} &= -\gamma\int_\Omega \frac{d}{d\varepsilon_1}\frac{(p_1+\varepsilon_1\varphi_1)_x p_{2_y}-(p_1+\varepsilon_1\varphi_1)_y p_{2_x}}{((p_1+\varepsilon_1\varphi_1)^2+p_2^2+1)^2}\ \Big|_{\varepsilon_1=0}\ d\Omega \\
&\quad+\frac{\theta_1}{2}\int_\Omega \frac{d}{d\varepsilon_1}((p_1+\varepsilon_1\varphi_1)-u_x-b_{11})^2\ \Big|_{\varepsilon_1=0}\ d\Omega \\
&\quad+\frac{\theta_2}{2}\int_\Omega \frac{d}{d\varepsilon_1}(m_1-\frac{(p_1+\varepsilon_1\varphi_1)}{\sqrt{(p_1+\varepsilon_1\varphi_1)^2+p_2^2+1}}-b_{21})^2\ \Big|_{\varepsilon_1=0}\ d\Omega, \\
&= -\gamma\int_\Omega \frac{(p_1^2+p_2^2+1)^2(\varphi_{1_x}p_{2_y}-\varphi_{1_y}p_{2_x})}{(p_1^2+p_2^2+1)^4}d\Omega \\
&\quad+\gamma\int_\Omega \frac{p_{1_x}p_{2_y}-p_{1_y}p_{2_x}}{(p_1^2+p_2^2+1)^4}\cdot 2(p_1^2+p_2^2+1)(2p_1\varphi_1)d\Omega \\
&\quad+\theta_1\int_\Omega (p_1-u_x-b_{11})\varphi_1\ d\Omega+\theta_2\int_\Omega (m_1-\frac{p_1}{\sqrt{p_1^2+p_2^2+1}}-b_{21})\times \\
&\quad(\frac{p_1^2\varphi_1}{(\sqrt{p_1^2+p_2^2+1})^3}-\frac{\varphi_1}{\sqrt{p_1^2+p_2^2+1}})\ d\Omega.
\end{aligned}$$

Therefore,

$$\begin{aligned}
\frac{\delta\overline{\mathcal{E}}(p_1,p_2;\varphi_1)}{\delta p_1} &= -\gamma\int_\Omega \frac{\varphi_{1_x}p_{2_y}-\varphi_{1_y}p_{2_x}}{\left(p_1^2+p_2^2+1\right)^2}\ d\Omega+4\gamma\int_\Omega \frac{p_{1_x}p_{2_y}-p_{1_y}p_{2_x}}{\left(p_1^2+p_2^2+1\right)^3}(p_1\varphi_1)\ d\Omega \\
&\quad+\theta_1\int_\Omega (p_1-u_x-b_{11})\varphi_1\ d\Omega+\theta_2\int_\Omega (m_1-\frac{p_1}{\sqrt{p_1^2+p_2^2+1}}-b_{21})\times \\
&\quad(\frac{p_1^2\varphi_1}{(\sqrt{p_1^2+p_2^2+1})^3}-\frac{\varphi_1}{\sqrt{p_1^2+p_2^2+1}})\ d\Omega, \\
&= \gamma\int_\Omega \frac{\varphi_{1_y}p_{2_x}-\varphi_{1_x}p_{2_y}}{\Gamma_2^2}\ d\Omega+\gamma\int_\Omega \Gamma p_1\varphi_1\ d\Omega+\theta_1\int_\Omega (p_1-u_x-b_{11})\varphi_1\ d\Omega \\
&\quad+\theta_2\int_\Omega (m_1-\psi p_1-b_{21})\psi^3 p_1^2\varphi_1\ d\Omega-\theta_2\int_\Omega (m_1-\psi p_1-b_{21})\psi\varphi_1\ d\Omega,
\end{aligned}$$

where $\Gamma=\frac{4\Gamma_1}{\Gamma_2^3}$, $\Gamma_1=p_{1_x}p_{2_y}-p_{1_y}p_{2_x}$, $\Gamma_2=p_1^2+p_2^2+1$ and $\psi=\dfrac{1}{\sqrt{\Gamma_2}}$.

Let $\boldsymbol{w}_1=(-\dfrac{p_{2_y}}{\Gamma_2^2},\dfrac{p_{2_x}}{\Gamma_2^2})$. Thus,

$$\int_\Omega \nabla\varphi_1\cdot\boldsymbol{w_1}\ d\Omega=\int_\Omega \frac{\varphi_{1_y}p_{2_x}-\varphi_{1_x}p_{2_y}}{\Gamma_2^2}\ d\Omega.$$

By Green's Theorem, we have

$$\int_\Omega \nabla\varphi_1\cdot\boldsymbol{w_1}\ d\Omega=\int_{\partial\Omega}\varphi_1(\boldsymbol{w_1}\cdot\boldsymbol{n}_1)\ d\boldsymbol{s}-\int_\Omega \varphi_1(\nabla\cdot\boldsymbol{w_1})\ d\Omega.$$

Therefore,

$$\int_\Omega \frac{\varphi_{1_y}p_{2_x}-\varphi_{1_x}p_{2_y}}{\Gamma_2^2}\ d\Omega=\int_{\partial\Omega}(\boldsymbol{w_1}\cdot\boldsymbol{n}_1)\varphi_1\ d\boldsymbol{s}-\int_\Omega (\nabla\cdot\boldsymbol{w_1})\varphi_1\ d\Omega.$$

It follows that

$$\begin{aligned}\frac{\delta\overline{\mathcal{E}}(p_1,p_2;\varphi_1)}{\delta p_1} &= -\gamma\int_\Omega(\nabla\cdot\boldsymbol{w_1})\varphi_1\ d\Omega+\gamma\int_{\partial\Omega}(\boldsymbol{w_1}\cdot\boldsymbol{n})\varphi_1\ d\boldsymbol{s}+\gamma\int_\Omega\Gamma p_1\varphi_1\ d\Omega\\ &\quad+\theta_1\int_\Omega(p_1-u_x-b_{11})\varphi_1\ d\Omega+\theta_2\int_\Omega(m_1-\psi p_1-b_{21})\psi^3p_1^2\varphi_1\ d\Omega\\ &\quad+\theta_2\int_\Omega\psi^2p_1\varphi_1\ d\Omega-\theta_2\int_\Omega(m_1-b_{21})\psi\varphi_1\ d\Omega\\ &=\gamma\int_\Omega(\Gamma p_1-\nabla\cdot\boldsymbol{w_1})\varphi_1\ d\Omega+\gamma\int_{\partial\Omega}(\boldsymbol{w_1}\cdot\boldsymbol{n})\varphi_1\ d\boldsymbol{s}+\theta_1\int_\Omega(p_1-u_x-b_{11})\varphi_1\ d\Omega\\ &\quad+\theta_2\int_\Omega((m_1-\psi p_1-b_{21})\psi p_1+1)\psi^2p_1\varphi_1\ d\Omega-\theta_2\int_\Omega(m_1-b_{21})\psi\varphi_1\ d\Omega.\end{aligned}$$

Similarly, the first variation of $\overline{\mathcal{E}}$ for p_2 in the direction of $\varphi_2\in\mathcal{V}$ is given by

$$\begin{aligned}\frac{\delta\overline{\mathcal{E}}(p_1,p_2;\varphi_2)}{\delta p_2} &=\gamma\int_\Omega(\Gamma p_2-\nabla\cdot\boldsymbol{w_2})\varphi_2\ d\Omega+\gamma\int_{\partial\Omega}(\boldsymbol{w_2}\cdot\boldsymbol{n})\varphi_2\ dS+\theta_1\int_\Omega(p_2-u_y-b_{12})\varphi_2\ d\Omega\\ &\quad+\theta_2\int_\Omega((m_2-\psi p_2-b_{22})\psi p_2+1)\psi^2p_2\varphi_2\ d\Omega-\theta_2\int_\Omega(m_2-b_{22})\psi\varphi_2\ d\Omega,\end{aligned}$$

where $\boldsymbol{w_2}=(\frac{p_{1_y}}{\Gamma_2^2},-\frac{p_{1_x}}{\Gamma_2^2})$. Therefore, a function $\boldsymbol{p}=(p_1,p_2)$ minimizing the energy functional $\overline{\mathcal{E}}$ for $\boldsymbol{p}$-subproblem has to satisfy the following Euler-Lagrange equation

$$\begin{cases}-\gamma\nabla\cdot\boldsymbol{w}_1+\gamma\Gamma p_1+\theta_1(p_1-u_x-b_{11})+\theta_2(m_1-\psi p_1-b_{21})(\psi^3p_1^2-\psi)=0\\ -\gamma\nabla\cdot\boldsymbol{w}_2+\gamma\Gamma p_2+\theta_1(p_2-u_y-b_{12})+\theta_2(m_2-\psi p_2-b_{22})(\psi^3p_2^2-\psi)=0\end{cases},\tag{71}$$

subject to the natural boundary condition $\boldsymbol{w_1}\cdot\boldsymbol{n}=0$ and $\boldsymbol{w_2}\cdot\boldsymbol{n}=0$.

References

[1] T. Loupas, W. McDicken, P. Allan, An adaptive weighted median filter for speckle suppression in medical ultrasound images, IEEE Transactions on Circuits and Systems 36 (1) (1989) 129–135.

[2] J. Koo, S. Oark, Speckle reduction with edge preservation in medical ultrasonic images using a homogenous region growing mean filter (HRGMF), Ultrasonic Imaging 13 (1990) 629–639.

[3] O. Michailovich, A. Tannenbaum, An accurate and efficient baysian model for automatic segmetation of brain MRI, IEEE Trans. Ultrason., Ferroelect., Freq. Contr. 53 (1) (2006) 64–78.

[4] P. Hellier, C. Kervrann, C. Barillot, Nonlocal means-based speckle filtering for ultrasound images, IEEE Trans. Image Process. 18 (10) (2009) 2221–2229.

[5] X. Hao, S. Gao, X. Gao, A novel multiscale nonlinear thresholding method for ultrasonic speckle suppressing, IEEE Trans. Med. Imaging. 18 (9) (1999) 787–794.

[6] J. Koo, S. Oark, Despeckling of medical ultrasound images using daubechies complex wavelet transform, Signal Processing 90 (2010) 428–439.

[7] F. Adamo, G. Andria, F. Attivissimo, A. Lanzolla, M. Spadavecchia, A comparative study on mother wavelet selection in ultrasound image denoising, Measurement 46 (2013) 2447–2456.

[8] A. Buades, B. Coll, J. M. Morel, A review of image denoising algorithms, with a new one, Multiscale Model. Simul. 4 (2) (2005) 490–530.

[9] P. Coupé, P. Hellier, C. Kervrann, C. Barillot, Nonlocal means-based speckle filtering for ultrasound images, IEEE Trans. Image Process. 18 (10) (2009) 2221–2229.

[10] P. Sudeep, P. Palanisamy, J. Rajan, H. Baradaran, L. Saba, A. Gupta, J. Suri, Speckle reduction in medical ultrasound images using an unbiased non-local means method, Biomedical Signal Processing and Control 28 (2016) 1–8.

[11] Q. Chen, P. Montesinos, Q. Sun, D. Xia, Ramp preserving Perona-Malik model, Signal Processing 90 (2010) 1963–1975.

[12] H. Yua, J.L.Tana, Y.Y.Wang, Ultrasound speckle reduction by a SUSAN-controlled anisotropic diffusion method, Pattern Recognition 43 (2010) 3083–3092.

[13] M. Islama, S. Mahbubur-Rahman, M. Omair-Ahmad, M. Swamy, Mixed Gaussian-impulse noise reduction from images using convolutional neural network, Signal Processing: Image Communication 68 (2018) 26–41.

[14] F. Zhang, N. Cai, J. Wu, G. Cen, H. Wang, X. Chen, Image denoising method based on a deep convolution neural network, IET Image Processing 12 (4) (2018) 485–493.

[15] W. Zuo, K. Zhang, L. Zhang, Convolutional Neural Networks for Image Denoising and Restoration, in Advances in Computer Vision and Pattern Recognition Series: Denoising of Photographic Images and Video, edited by M. Bertalmío , Springer, Cham, pp. 93-123, 2018.

[16] H. Shahdoost, Z. Rahemi, Edge-preserving image denoising using a deep convolutional neural network, Signal Processing 159 (2019) 20–32.

[17] Y. You, M. Kaveh, Fourth-order partial differential equations for noise removal, IEEE Transactions on Image Processing 9 (10) (2000) 1723–1730.

[18] O. Scherzer, Denoising with higher order derivatives of bounded variation and an application to parameter estimation, Computing 60 (1) (1998) 1–27.

[19] M. Lysaker, A. Lundervold, X.-C. Tai, Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time, IEEE Transactions on Image Processing 12 (12) (2003) 1579–1590.

[20] W. Hinterberger, O. Scherzer, Variational methods on the space of functions of bounded hessian for convexification and denoising, Computing 76 (1) (2006) 109–133.

[21] M. Bergounioux, L.Piffet, A second-order model for image denoising, Set-Valued and Variational Analysis 18 (3-4) (2010) 277–306.

[22] R. Lai, X.-C. Tai, T. Chan, A ridge and corner preserving model for surface restoration, SIAM Journal on Scientific Computing 35 (2) (2013) A675–A695.

[23] T. Chan, S. Esedoglu, F. Park, A fourth order dual method for staircase reduction in texture extraction and image restoration problems, 2010 17th IEEE International Conference on Image Processing (ICIP), Hongkong, China (2010) 4137–4140.

[24] S. Zheng, Z. Pan, C. Jiang, G. Wang, A new fast algorithm for image denoising, 3rd International Conference on Multimedia Technology (2013) 682–689.

[25] G. Wang, J. Xu, Q. Dong, Z. Pan, Active contour model coupling with higher order diffusion for medical image segmentation, International Journal of Biomedical Imaging 2014 (2014) 1–8.

[26] R. Chan, H. Liang, S. Wei, M. Nikolova, X.-C. Tai, High-order total variation regularization approach for axially symmetric object tomography from a single radiograph, Inverse Problems and Imaging 9 (1) (2015) 55–77.

[27] K. Papafitsoros, C. Schönlieb, A combined first and second order variational approach for image reconstruction, Journal of Mathematical Imaging and Vision 48 (2) (2014) 308–338.

[28] A. Chambolle, P. Lions, Image recovery via total variation minimization and related problems, Numerische Mathematik 76 (2) (1997) 167–188.

[29] K. Bredies, K. Kunisch, T. Pock., Color TV: Total variation methods for restoration of vector-valued images, SIAM J. Imaging Sci. 3 (3) (2010) 492–526.

[30] W. Zhu, T. Chan, Image denoising using mean curvature of image surface, SIAM J. Imaging Sci. 5 (1) (2012) 1–32.

[31] C. Brito-Loeza, K. Chen, V. Uc-Cetina, Image denoising using the gaussian curvature of the image surface, Numer. Meth. Part. Differ. Equ. 32 (3) (2016) 1066–1089.

[32] L. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Physica D 60 (1992) 259–268.

[33] J. Zhang, K. Chen, A total fractional-order variation model for image restoration with nonhomogeneous boundary conditions and its numerical solution, SIAM J. Imaging Sci. 8 (4) (2015) 2487–2518.

[34] W. Lu, J. Duan, Z. Qiu, Z. Pan, R. Lid, L. Bai, Implementation of high-order variational models made easy for image processing, Mathematical Methods in the Applied Sciences 39 (2016) 4208–4233.

[35] C. Brito-Loeza, K. Chen, On fast iterative algorithms for solving the minimisation of curvature-related functionals in surface fairing, International Journal of Computer Mathematics 90 (1) (2013) 92–110.

[36] T.F.Chan, S. Kang, J. Shen, Euler's elastica and curvature based inpaintings, SIAM J. Appl. Math. 63 (2) (2002) 564–592.

[37] C. Brito-Loeza, K. Chen, Multigrid algorithm for high order denoising, SIAM J.Imaging Sci. 3 (3) (2010) 363–389.

[38] T. Goldstein, S. Osher., The split bregman method for l1-regularized problems., SIAM Journal on Imaging Sciences 2 (2) (2009) 323–343.

[39] X.-C. Tai., C. Wu., Augmented Lagrangian method, dual methods and split Bregman iteration for ROF model, Scale Space and Variational Methods in Computer Vision, LNCS 5567 (556) (2009) 502–513.

[40] C. Wu., X.-C. Tai., Augmented lagrangian method, dual methods, and split bregman iteration for rof, vectorial tv, and high order models, SIAM J. Imaging Sci. 3 (3) (2010) 300–339.

[41] Y. Han, X.-C. Feng, G. Baciu, W.-W. Wang., Nonconvex spare regularizer based speckle noise removal, Pattern recognition 46 (2013) 989–1001.

[42] Y. Huang, M. Ng, Y. Wen, A new total variation method for multiplicative noise removal, SIAM J. Imaging Sci. 2 (1) (2009) 20–40.

[43] J.Huang, X.Yang, Fast reduction of speckle noise in real ultrasound images, Signal Processing 93 (2013) 684–694.

[44] L. Xiao, L.-L. Huang, Z.-H. Wei., A weberized total variation regularization-based image multiplicative noise removal algorithm, EURASIP Journal on Advances in Signal Processing 490384 (2010) :DOI:0.1155/2010/490384.

[45] Z.-F. P. B. Shi, L.Huang, Fast algorithm for multiplicative noise removal, J. Vis. Commun. Image R. 23 (2012) 126–133.

[46] P. Sroisangwan, N. Chumchob, A New Numerical Method for Gaussian Curvatuve Based Image Restoration, in Proceedings of Annual Pure and Applied Mathematics Conference 2018, Chulalongkorn University, Bangkok, Thailand, pp. 54-66, 2018.

[47] Z. Wang, A. Bovik, H. Sheikh, E. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612.

An Augmented Lagrangian Method for Solving Total Variation (TV)-based Image Registration Model

Noppadol Chumchob[a,*]

[a]*Department of Mathematics, Faculty of Science, Silpakorn University, Nakhon Pathom, Thailand*

Abstract

Variational methods for image registration basically involve a regularizer to ensure that the resulting well-posed problem admits a solution. Different choices of regularizers lead to different deformations. On one hand, the conventional regularizers, such as the elastic, diffusion and curvature regularizers, are able to generate globally smooth deformations and generally useful for many applications. On the other hand, these regularizers become poor in some applications where discontinuities or steep gradients in the deformations are required. As is well-known, the total (TV) variation regularizer is more appropriate to preserve discontinuities of the deformations. However, it is difficult in developing an efficient numerical method to ensure that numerical solutions satisfy this requirement because of the non-differentiability and non-linearity of the TV regularizer. In this work we focus on computational challenges arising in approximately solving TV-based image registration model. Motivated by many efficient numerical algorithms in image restoration, we propose to use augmented Lagrangian method (ALM). At each iteration, the computation of our ALM requires to solve two subproblems. On one hand for the first subproblem, it is impossible to obtain exact solution. On the other hand for the second subproblem, it has a closed-form solution. To this end, we propose an efficient nonlinear multigrid (NMG) method to obtain an approximate solution to the first subproblem. Numerical results on real medical images not only confirm that our proposed ALM is more computationally efficient than some existing methods, but also that the proposed ALM delivers the accurate registration results with the desired property of the constructed deformations in a reasonable number of iterations.

Keywords: Image registration, augmented Lagrangian method, nonlinear multigrid method, total variation regularization

2010 MSC: 93C10, 35Q94, 94A08, 68U10, 65K10, 65N55

*Corresponding author
Email address: chumchob_n@silpakorn.edu (Noppadol Chumchob)

1. Introduction

One of important tasks in image analysis is image registration. Generally speaking, it is the process of matching two or more images of the same or similar object obtained from different times, perspectives, and/or imaging sensors. This process can be done mathematically by computing a spatial geometric transformation of deformations (also known as displacement fields) that maps each point in one image onto a corresponding point in the other image with an *optimal* or *meaningful* manner. Image registration has played an important role in several areas of applications. Especially in medical applications, for example, it has been used routinely in medical diagnosis, treatment guidance and monitoring for providing complementary information. A good survey of the medical applications can be seen in [1, 2, 3, 4] and the references therein.

Variational methods for image registration have been actively and extensively studied and applied in the field of image analysis. The main idea is to find spatial correspondences between two given images, a so-called reference image $R : \Omega \subset \mathbb{R}^2 \to V \subset \mathbb{R}$ and a so-called template image $T : \Omega \subset \mathbb{R}^2 \to V \subset \mathbb{R}$, where the image domain Ω is assumed to be a rectangle. This is usually done by finding an optimal deformation $\boldsymbol{u} : \mathbb{R}^2 \to \mathbb{R}^2, \boldsymbol{u} : \mathbf{x} \mapsto \boldsymbol{u}(\mathbf{x}) = (u_1(\mathbf{x}), u_2(\mathbf{x}))^\top$, such that the transformed version of the template image $T_{\boldsymbol{u}}(\mathbf{x}) = T(\mathbf{x} + \boldsymbol{u}(\mathbf{x}))$ and the reference image R are spatially matched, according to an image-to-image dissimilarity measure, $\mathcal{D}(\boldsymbol{u})$. Without loss of generality, this work assumes that $\Omega = [0, 1]^2 \subset \mathbb{R}^2$ and $V = [0, 1]$ for 2D gray intensity images.

If the image intensities of the given images R and T are comparable, one may have various choices for $\mathcal{D}$. Probably the most popular choice for the dissimilarity measure is provided by the so-called *sum of squared differences* (SSD)

$$\mathcal{D}(\boldsymbol{u}) = \frac{1}{2} \int_\Omega \left(T_{\boldsymbol{u}}(\mathbf{x}) - R(\mathbf{x})\right)^2 d\mathbf{x}.$$

The registration task is then to solve the minimization problem

$$\min_{\boldsymbol{u}} \{\mathcal{D}(\boldsymbol{u})\}, \tag{1}$$

where $\boldsymbol{u}$ is searched over a set of admissible functions $\mathcal{U}$ minimizing $\mathcal{D}$. As is well-known, the minimization of $\mathcal{D}$ does not have a unique minimizer and it becomes necessary to impose a constraint on the solution $\boldsymbol{u}$ via a regularizer $\mathcal{R}$. By modifying (1), the desired deformation $\boldsymbol{u}$ is a minimizer of the variational problem

$$\min_{\boldsymbol{u}} \{\mathcal{J}_\alpha(\boldsymbol{u}) = \mathcal{D}(\boldsymbol{u}) + \alpha \mathcal{R}(\boldsymbol{u})\} \tag{2}$$

where $\alpha > 0$ is a positive constant that compromises the quality of the similarity between $T_{\boldsymbol{u}}$ and R, measured by $\mathcal{D}$, and the level of penalty for unwanted deformations, measured by $\mathcal{R}$.

In this work, we focus on the regularizer of the form

$$\mathcal{R}(\boldsymbol{u}) = \sum_{l=1}^{2} \mathcal{R}^{\mathrm{TV}}(u_l),\ \mathcal{R}^{\mathrm{TV}}(u_l) = \int_{\Omega} |\nabla u_l|\, d\mathbf{x} = \int_{\Omega} \sqrt{u_{l,x}^2 + u_{l,y}^2} d\mathbf{x} \tag{3}$$

with $u_{l,x} = \frac{\partial u_l}{\partial x}$ and $u_{l,y} = \frac{\partial u_l}{\partial y}$ to ensure that the constructed deformation $\boldsymbol{u}$ is unique and preserves discontinuities of the deformations. This term is known as the total variation (TV) regularizer [5, 6, 7]. In contrast to other traditional regularizers generating globally smooth deformations, e.g. elastic [8], diffusion [9], and curvature regularizers [10, 11], the TV regularizer is able to produces locally non-smooth deformations, which are required in matching several moved objects or partially occluded objects in medical applications, particularly at organ boundaries during the breathing induced organ motion.

In order to get a numerical solution of the variational problem (2), the standard gradient descent method can be applied. We first embed the associated Euler-Lagrange (EL) equation of (2) into a dynamic equation and drive it to a steady state. This yields the explicit scheme as given by

$$\begin{cases} \dfrac{u_1^{[k+1]} - u_1^{[k]}}{\tau} = \alpha \nabla \cdot \left(\dfrac{\nabla u_1^{[k]}}{\sqrt{(u_{1,x}^{[k]})^2 + (u_{1,y}^{[k]})^2 + \beta}} \right) - f_1(\boldsymbol{u}^{[k]}), \\ \dfrac{u_2^{[k+1]} - u_2^{[k]}}{\tau} = \alpha \nabla \cdot \left(\dfrac{\nabla u_2^{[k]}}{\sqrt{(u_{2,x}^{[k]})^2 + (u_{2,y}^{[k]})^2 + \beta}} \right) - f_2(\boldsymbol{u}^{[k]}), \end{cases} \tag{4}$$

where $f_l(\boldsymbol{u}^{[k]}) = (T_{\boldsymbol{u}^{[k]}} - R)\, \partial_{u_l} T_{\boldsymbol{u}^{[k]}},\ k = 0, 1, 2, \ldots$ is the time step, $\tau > 0$ is the step size, and $\beta > 0$ is a small real parameter to avoid zero division; see more details in [12, 13, 14]. This numerical scheme is easy to implement, but very slow to converge due to the constraint of stability conditions in the step size τ. In order to speed up the convergence of (4), we may linearize nonlinear 'coefficients' in the associated system and define the iteration step as follows:

$$\begin{cases} \dfrac{u_1^{[k+1]} - u_1^{[k]}}{\tau} - \alpha \nabla \cdot \left(\dfrac{\nabla u_1^{[k+1]}}{\sqrt{(u_{1,x}^{[k]})^2 + (u_{1,y}^{[k]})^2 + \beta}} \right) = -f_1(\boldsymbol{u}^{[k]}), \\ \dfrac{u_2^{[k+1]} - u_2^{[k]}}{\tau} - \alpha \nabla \cdot \left(\dfrac{\nabla u_2^{[k+1]}}{\sqrt{(u_{2,x}^{[k]})^2 + (u_{2,y}^{[k]})^2 + \beta}} \right) = -f_2(\boldsymbol{u}^{[k]}). \end{cases} \tag{5}$$

The idea of the linearized gradient descent (LGD) method in (5) is to linearize and solve the EL equation via a fixed-point (FP) iteration in a similar way to the so-called *Lagged-diffusivity* method [15] or Quasi-Newton scheme [16, 17]. For each iteration, a linear system needs to be solved. As can be seen, both gradient descent methods in (4) and (5) share two drawbacks. At first, these gradient descent methods provide only the the approximate solutions of the original problem (2), since the TV regularizer $\mathcal{R}(\boldsymbol{u})$ in (3) is replaced with

$$\mathcal{R}_\beta(\boldsymbol{u}) = \sum_{l=1}^{2} \mathcal{R}_\beta^{\mathrm{TV}}(u_l),\ \mathcal{R}_\beta^{\mathrm{TV}}(u_l) = \int_\Omega |\nabla u_l|_\beta \, d\mathbf{x} = \int_\Omega \sqrt{u_{l,x}^2 + u_{l,y}^2 + \beta} d\mathbf{x}, \tag{6}$$

for avoiding non-differentiability of $\mathcal{R}^{\mathrm{TV}}(u_l)$ and thus approximated to get (4) and (5). On the second, the choice of β will effect on the computational efficiency of the numerical methods and the smoothness of the constructed deformations. Larger the β, more efficient the methods are, whereas more smooth the constructed deformations will be. Therefore the registration performance in smoothing deformations by these gradient descent methods is very sensitive to the parameter β. Indeed, there are a few different methods existing in the literature for solving the associated EL equation of (2). For example, the use of the nonlinear multigrid (NMG) methods can be found in works of [5, 6, 18, 7]. However, these existing NMG methods also suffer from difficulties related to the non-differentiability and non-linearity of $\mathcal{R}^{\mathrm{TV}}(u_l)$. To develop efficient numerical methods for TV-based image registration is still a challenging task and has been an active research area so far.

Recently, the variable-splitting methods are the well-known techniques in the field of image restoration for solving variational models, which require the minimization of nonlinear and non-differentiable functionals. To the best of the author's knowledge, the variable-splitting methods for TV-based image registration model in (2) are still missing in the literature. In this paper, the proposed method is based on the so-called augmented Lagrangian method (ALM). We compare the performance of the proposed ALM with the LGD method in (5) and the NMG method developed by Chumchob [7]. Numerical results show that our proposed ALM is more computationally efficient than these existing methods

The rest of this paper is organized as follows. In Section 2, we derive our ALM algorithm to solve the variational problem (2), following its numerical implementation in Section 3, In Section 4, numerical comparisons are carried out to confirm the effectiveness of the proposed ALM. Finally, some concluding remarks are made in Section 5.

2. The proposed ALM

Before we derive the proposed ALM, we first rewrite (2) as

$$\min_{\boldsymbol{u}}\{\mathcal{J}(\boldsymbol{u}) = \mathcal{D}(\boldsymbol{u}) + \alpha\sum_{l=1}^{2}\int_{\Omega}|\nabla u_l|\,d\mathbf{x}\}. \tag{7}$$

Next, we introduce an auxiliary variable $\boldsymbol{w} = (\boldsymbol{w}_1, \boldsymbol{w}_2)$ such that $\boldsymbol{w}_l = \nabla u_l$ and reformulate the minimization problem (7) to the the following constrained minimization problem

$$\begin{aligned}\min_{\boldsymbol{u},\boldsymbol{w}} \mathcal{J}(\boldsymbol{u},\boldsymbol{w}), \qquad (8)\\ \mathcal{J}(\boldsymbol{u},\boldsymbol{w}) = \mathcal{D}(\boldsymbol{u}) + \alpha\sum_{l=1}^{2}\int_{\Omega}|\boldsymbol{w}_l|\,d\mathbf{x},\\ \text{s.t. } \boldsymbol{w}_l = \nabla u_l \text{ for } l = 1, 2.\end{aligned}$$

For the minimization of (8), this work proposes to use the ALM and rewrite the constrained minimization problem (8) into an unconstrained minimization problem. We define the augmented Lagrangian functional for the above constrained minimization as follows

$$\begin{aligned}\mathcal{L}(\boldsymbol{u},\boldsymbol{w};\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2) = \mathcal{J}(\boldsymbol{u},\boldsymbol{w}) + \sum_{l=1}^{2}\frac{\theta_l}{2}\int_{\Omega}|\boldsymbol{w}_l - \nabla u_l|^2\,d\mathbf{x}\\ + \sum_{l=1}^{2}\int_{\Omega}\boldsymbol{\lambda}_l\cdot(\boldsymbol{w}_l - \nabla u_l)\,d\mathbf{x}, \qquad (9)\end{aligned}$$

where θ_1, θ_2 are the positive penalty parameters and $\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2$ are the Lagrange multipliers.

We propose an iterative algorithm to solve the minimization of $\mathcal{L}(\boldsymbol{u},\boldsymbol{w};\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2)$; see Algorithm 1. Since $\boldsymbol{u}$ and $\boldsymbol{w}$ are coupled together in the minimization problem (9), it is very difficult to solve all variables simultaneously. We separate the minimization problem into two sub-problems and develop an alternating minimization procedure to approximate the solution. This process is repeated until one of the following stopping rules is satisfied:

$$\begin{aligned}\text{RelSSD}^{(m)} = \tfrac{\mathcal{D}(\boldsymbol{u}^{(m)})}{\mathcal{D}(\boldsymbol{u}^{(0)})} < \epsilon_1, \qquad (13)\\ \max\{\tfrac{\|u_1^{(m)}-u_1^{(m-1)}\|}{\|u_1^{(m)}\|}, \tfrac{\|u_2^{(m)}-u_2^{(m-1)}\|}{\|u_2^{(m)}\|}\} < \epsilon_2, \qquad (14)\\ \max\{\tfrac{\|\boldsymbol{w}_1^{(m)}-\nabla u_1^{(m)}\|}{|\Omega|}, \tfrac{\|\boldsymbol{w}_2^{(m)}-\nabla u_2^{(m)}\|}{|\Omega|}\} < \epsilon_3, \qquad (15)\\ \max\{\tfrac{\|\boldsymbol{\lambda}_1^{(m)}-\boldsymbol{\lambda}_1^{(m-1)}\|}{\|\boldsymbol{\lambda}_1^{(m)}\|}, \tfrac{\|\boldsymbol{\lambda}_2^{(m)}-\boldsymbol{\lambda}_2^{(m-1)}\|}{\|\boldsymbol{\lambda}_2^{(m)}\|}\} < \epsilon_5, \qquad (16)\\ m \geq \epsilon_4, \qquad (17)\end{aligned}$$

where ϵ_1, ϵ_2, ϵ_3 and ϵ_4 denote the predefined small positive numbers, and ϵ_5 is the maximum iteration. Here m denotes the index of the current iteration.

Algorithm 1 ALM for TV-based image registration

1) Initialization: set $m = 0$, choose $\theta_1, \theta_2 > 0$ and $\boldsymbol{\lambda}_1^{(0)}, \boldsymbol{\lambda}_2^{(0)}$.

2) **Repeat**

2.1) Compute $(\boldsymbol{u}^{(m)}, \boldsymbol{w}^{(m)})$ as an (approximate) minimizer of the augmented Lagrangian functional with the Lagrange multipliers $\boldsymbol{\lambda}_1^{(m-1)}$, $\boldsymbol{\lambda}_2^{(m-1)}$, i.e.,

$$(\boldsymbol{u}^{(m)}, \boldsymbol{w}^{(m)}) \approx \arg\min_{\boldsymbol{u},\boldsymbol{w}} \mathcal{L}(\boldsymbol{u}, \boldsymbol{w}; \boldsymbol{\lambda}_1^{(m-1)}, \boldsymbol{\lambda}_2^{(m-1)}). \tag{10}$$

2.2) Update Lagrange multipliers

$$\boldsymbol{\lambda}_1^{(m)} \leftarrow \boldsymbol{\lambda}_1^{(m-1)} + \theta_1(\boldsymbol{w}_1^{(m)} - \nabla u_1^{(m)}), \tag{11}$$

$$\boldsymbol{\lambda}_2^{(m)} \leftarrow \boldsymbol{\lambda}_2^{(m-1)} + \theta_2(\boldsymbol{w}_2^{(m)} - \nabla u_2^{(m)}). \tag{12}$$

until a stopping rule for ALM method is satisfied.

The two-subproblems are as follows:

***u*-subproblem.** Given $\boldsymbol{w} = (\boldsymbol{w}_1, \boldsymbol{w}_2)$, $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$, we search a minimizer $\boldsymbol{u} = (u_1, u_2)^\top$ of the minimization problem:

$$\min_{\boldsymbol{u}} \mathcal{J}(\boldsymbol{u}), \tag{18}$$

$$\mathcal{J}(\boldsymbol{u}) = \mathcal{D}(\boldsymbol{u}) + \sum_{l=1}^{2} \frac{\theta_l}{2} \int_\Omega |\boldsymbol{w}_l - \nabla u_l|^2 \, d\mathbf{x} - \sum_{l=1}^{2} \int_\Omega \nabla u_l \cdot \boldsymbol{\lambda}_l d\mathbf{x}. \tag{19}$$

According to the calculus of variation, the solution of $\boldsymbol{u}$-subproblem is determined by the associated EL equation

$$\begin{cases} -\theta_1 \triangle u_1 + f_1(\boldsymbol{u}) = g_1, \\ -\theta_2 \triangle u_2 + f_2(\boldsymbol{u}) = g_2, \end{cases} \tag{20}$$

subject to the homogeneous Neumann boundary conditions $\frac{\partial u_l}{\partial \boldsymbol{n}} = 0$, where $\boldsymbol{n}$ denotes the unit outward normal vector on the image boundary $\partial\Omega$, $f_l(\boldsymbol{u}) = (T_{\boldsymbol{u}} - R)\partial_{u_l} T_{\boldsymbol{u}}$, $g_l = -\theta_1 \nabla \cdot \boldsymbol{w}_l - \nabla \cdot \boldsymbol{\lambda}_l$, and $l = 1, 2$.

***w*-subproblem**. Given $\boldsymbol{u}$, $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$, the following minimization problem

$$\min_{\boldsymbol{w}} \mathcal{J}(\boldsymbol{w}), \tag{21}$$

$$\mathcal{J}(\boldsymbol{w}) = \alpha \sum_{l=1}^{2} \int_\Omega |\boldsymbol{w}_l| \, d\mathbf{x} + \sum_{l=1}^{2} \frac{\theta_l}{2} \int_\Omega |\boldsymbol{w}_l - \nabla u_l|^2 \, d\mathbf{x} + \sum_{l=1}^{2} \int_\Omega \boldsymbol{w}_l \cdot \boldsymbol{\lambda}_l d\mathbf{x}, \tag{22}$$

is solved for $\boldsymbol{w} = (\boldsymbol{w}_1, \boldsymbol{w}_2)$. Despite the fact that the variables $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ do not decouple we can still explicitly solve this minimization problem for $\boldsymbol{w}$ using the generalized shrinkage formula [19]:

$$\boldsymbol{w}_l = \max(|\boldsymbol{S}_l| - \frac{\alpha}{\theta_l}, 0)\frac{\boldsymbol{S}_l}{|\boldsymbol{S}_l|}, \tag{23}$$

with the convection ($\frac{0}{|0|} = 0$), where $\boldsymbol{S}_l = \nabla u_l - \frac{\boldsymbol{\lambda}_l}{\theta_l}$ and $l = 1, 2$.

3. Numerical implementation

In this section, we present the details of how to solve the equation (20) and update the variables $\boldsymbol{w}_1$, $\boldsymbol{w}_2$, $\boldsymbol{\lambda}_1$, and $\boldsymbol{\lambda}_2$ for each iteration.

3.1. Finite difference discretization

In order to discretize the EL equation (20), let

$$\begin{aligned}
\left(u_l^h\right)_{i,j} &= u_l^h\left(x_{1_i}, x_{2_j}\right), \ \left(\boldsymbol{u}^h\right)_{i,j} = ((u_1^h)_{i,j}, (u_2^h)_{i,j})^\top \\
\left(w_{l,1}^h\right)_{i,j} &= w_{l,1}^h\left(x_{1_i}, x_{2_j}\right), \ \left(w_{l,2}^h\right)_{i,j} = w_{l,2}^h\left(x_{1_i}, x_{2_j}\right), \\
\left(\boldsymbol{w}_l^h\right)_{i,j} &= ((w_{l,1}^h)_{i,j}, (w_{l,2}^h)_{i,j}), \ (g_l^h)_{i,j} = g_l^h\left(x_{1_i}, x_{2_j}\right),
\end{aligned}$$

denote the grid functions for $l = 1, 2$ with the grid spacing $h = (h_1, h_2) = (1/n_1, 1/n_2)$, where the integers $n_1 = 1/h_1$ and $n_2 = 1/h_2$ are the number of uniform intervals in the x_1 and x_2 coordinate directions. Each grid point $\mathbf{x}$ in the discretized domain $\Omega_h \subset \Omega$ is given by $\mathbf{x} = (x_{1_i}, x_{2_j})^\top = (\frac{(2i-1)h_1}{2}, \frac{(2j-1)h_2}{2})^\top$ for $1 \le i \le n_1$ and $1 \le j \le n_2$.

Applying the finite difference approximations with (20), the discrete EL equation at a grid point (x_{1_i}, x_{2_j}) over the discrete domain Ω_h is given by

$$\begin{cases}
\underbrace{-\theta_1 \triangle^h (u_1^h)_{i,j} + f_1^h\left(\boldsymbol{u}^h\right)_{i,j}}_{\mathcal{N}_1^h(\boldsymbol{u}^h)_{i,j}} = (g_1^h)_{i,j}, \\
\underbrace{-\theta_2 \triangle^h (u_2^h)_{i,j} + f_2^h\left(\boldsymbol{u}^h\right)_{i,j}}_{\mathcal{N}_2^h(\boldsymbol{u}^h)_{i,j}} = (g_2^h)_{i,j},
\end{cases} \tag{24}$$

with the following notation

$$\begin{aligned}
\delta_{x_1}^-(u_l^h)_{i,j} &= ((u_l^h)_{i,j} - (u_l^h)_{i-1,j})/h_1, \\
\delta_{x_1}^+(u_l^h)_{i,j} &= ((u_l^h)_{i+1,j} - (u_l^h)_{i,j})/h_1, \\
\delta_{x_2}^-(u_l^h)_{i,j} &= ((u_l^h)_{i,j} - (u_l^h)_{i,j-1})/h_2, \\
\delta_{x_2}^+(u_l^h)_{i,j} &= ((u_l^h)_{i,j+1} - (u_l^h)_{i,j})/h_2, \\
\nabla^h(u_l^h)_{i,j} &= (\delta_{x_1}^+(u_l^h)_{i,j}, \delta_{x_2}^+(u_l^h)_{i,j})^\top,
\end{aligned}$$

$$
\begin{aligned}
\nabla^h \cdot (\boldsymbol{w}_l^h)_{i,j} &= \delta_{x_1}^-(w_{l,1}^h)_{i,j} + \delta_{x_2}^-(w_{l,2}^h)_{i,j},\\
-\triangle^h(u_l^h)_{i,j} &= -\nabla^h \cdot (\nabla^h(u_l^h)_{i,j}),\\
&= \frac{1}{h_1^2}((\Sigma)_{i,j}^h\,(u_l^h)_{i,j} - (\overline{\Sigma}^h)_{i,j}(u_l^h)_{i,j}),\\
(\Sigma)_{i,j}^h &= 2(1+\gamma^2),\ \gamma = h_1/h_2,\\
\left(\overline{\Sigma}\right)_{i,j}^h (u_l^h)_{i,j} &= (u_l^h)_{i+1,j} + (u_l^h)_{i-1,j} + \gamma^2(u_l^h)_{i,j+1} + \gamma^2(u_l^h)_{i,j-1},\\
f_1^h\left(\boldsymbol{u}^h\right)_{i,j} &= f_1^h\left(u_1^h, u_2^h\right)_{i,j} = (T_{i,j}^{h^*} - R_{i,j}^h)((T_{i+1,j}^{h^*} - T_{i-1,j}^{h^*})/\,(2h_1)),\\
f_2^h\left(\boldsymbol{u}^h\right)_{i,j} &= f_2^h\left(u_1^h, u_2^h\right)_{i,j} = (T_{i,j}^{h^*} - R_{i,j}^h)((T_{i,j+1}^{h^*} - T_{i,j-1}^{h^*})/\,(2h_2)),\\
T_{i,j}^{h^*} &= T^h(i + (u_1^h)_{i,j}, j + (u_2^h)_{i,j}),\\
(g_l^h)_{i,j} &= -\delta_{x_1}^-(\theta_l(w_{l,1}^h)_{i,j} + (\lambda_{l,1}^h)_{i,j}) - \delta_{x_2}^-(\theta_l(w_{l,2}^h)_{i,j} + (\lambda_{l,2}^h)_{i,j}).
\end{aligned}
$$

We note that all finite difference approximations need to be adjusted at the image boundary $\partial\Omega_h$ using the approximations of the boundary conditions

$$
(u_l^h)_{i,1} = (u_l^h)_{i,2},\ (u_l^h)_{i,n_2} = (u_l^h)_{i,n_2-1},\ (u_l^h)_{1,j} = (u_l^h)_{2,j},\ (u_l^h)_{n_1,j} = (u_l^h)_{n_1-1,j}.
$$

In the following subsections the symbols 'h' and '$(\cdot)_{i,j}$' will sometimes drop for simplicity.

3.2. Nonlinear multigrid method for solving $\boldsymbol{u}$-subproblem

The most difficult part of the proposed ALM is the solution of the EL equation for $\boldsymbol{u}$-subproblem in (20). This subproblem is nonlinear and impossible to obtain a closed-form solution. It therefore requires one to approximately solve the corresponding nonlinear discrete system via some iterative method (e.g, a FP method and a Newton-type method). As is well-known, the computational costs of iterative methods are too expensive and we are motivated to develop an efficient numerical method for reducing the computational work.

In this section, we propose a nonlinear multigrid (NMG) method due to Brandt [20]. The basic idea is to accurate the convergence of some basic iterative method on the finest grid by relying on the complementary interplay of smoothing and coarse-grid correction principles. For a more comprehensive treatment of MG methods in the area of image registration, we refer [18, 21, 11, 22, 23, 7, 24, 25, 26] and references therein.

Let us denote the nonlinear discrete system in (24) using the following notation:

$$
\begin{cases}
\mathcal{N}_1^h(\boldsymbol{u}^h) = g_1^h,\\
\mathcal{N}_2^h(\boldsymbol{u}^h) = g_2^h.
\end{cases}
\tag{25}
$$

where $\boldsymbol{u}^h$ denotes the exact solution. Let $\overline{\boldsymbol{u}}^h = (\overline{u}_1^h, \overline{u}_2^h)^\top$ be an approximate solution of $\boldsymbol{u}^h$ resulting from applying a few iterations of the smoother on a fine-grid Ω_h (*pre-smoothing*

step). Define the algebraic error as $\boldsymbol{e}^h = \boldsymbol{u}^h - \overline{\boldsymbol{u}}^h$. Therefore, the residual equation is given by

$$\mathcal{N}_l^h(\overline{\boldsymbol{u}}^h + \boldsymbol{e}^h) - \mathcal{N}_l^h(\overline{\boldsymbol{u}}^h) = g_l^h - \mathcal{N}_l^h(\overline{\boldsymbol{u}}^h) = r_l^h \quad \text{for } l = 1, 2.$$

To correct $\overline{\boldsymbol{u}}^h$ numerically on Ω_h, $\boldsymbol{e}^h$ is required to compute, but this computation is considerably and prohibitively expensive on Ω_h. Since the smoother in the pre-smoothing step removes the high frequency component of the error, one can represent the nonlinear system to the coarse grid Ω_H as given by

$$\underbrace{\mathcal{N}_l^h(\overline{\boldsymbol{u}}^h + \boldsymbol{e}^h)}_{\mathcal{N}_l^h(\boldsymbol{u}^h)} = \underbrace{r_l^h + \mathcal{N}_l^h(\overline{\boldsymbol{u}}^h)}_{g_l^h} \longrightarrow \underbrace{\mathcal{N}_l^H(\overline{\boldsymbol{u}}^H + \boldsymbol{e}^H)}_{\mathcal{N}_l^H(\boldsymbol{u}^H)} = \underbrace{r_l^H + \mathcal{N}_l^H(\overline{\boldsymbol{u}}^H)}_{g_l^H}, \tag{26}$$

where H is the index for the new cell size $H_1 \times H_2$ with $H_1 \geq h_1$ and $H_2 \geq h_2$. Next, the nonlinear residual equation (26) on Ω_H has to be solved by an efficient method (*the coarsest grid solver*). The coarse-grid correction $\boldsymbol{e}^H = \boldsymbol{u}^H - \overline{\boldsymbol{u}}^H$ on Ω_H is then interpolated back to that on the fine grid $\boldsymbol{e}^h$, which can now be used to update $\overline{\boldsymbol{u}}^h$ on Ω_h by using $\overline{\boldsymbol{u}}^h_{new} = \overline{\boldsymbol{u}}^h + \boldsymbol{e}^h$ (*coarse-grid correction* step). Finally, the smoother is again performed with a few iterations on Ω_h to remove high frequency parts of the interpolated error (*post-smoothing* step). This procedure is known as a two-grid cycle, and it can be extended to a MG method with recursive application.

Below, we provide the technical details for our MG components.

1. The *standard coarsening method* is used in Ω_H by doubling the grid spacing in each space direction — i.e. $h_l \to 2h_l = H_l$ for $l = 1, 2$.
2. The *restriction operator* I_h^H and *interpolation operators* I_H^h are used as the intergrid transfer operators and determined by the averaging and bilinear interpolation techniques — cf. [27, 28, 29, 30, 31] for more details.
3. The *discretization coarse grid approximation method* is used to compute the coarse-grid operators.
4. The *coarsest grid solver* is the LGD method.
5. The *MG cycle* is $V(\nu_1;\nu_2)$-cycle, where ν_1 and ν_2 denote respectively the numbers of pre- and post-smoothing steps.
6. The *MG smoother* to be discussed shortly in Section 3.3 is obtained from a coupled outer-inner iteration method using a relaxation parameter ω.

The implementation of the proposed NMG method can be summarized as follows:

$$\overline{\boldsymbol{u}}^h \leftarrow \text{NMGCYC}(\overline{\boldsymbol{u}}^h, g_1^h, g_2^h, R^h, T^h, \nu_1, \nu_2, \theta_1, \theta_2, \omega, \text{Siter})$$

1) If $\Omega_h = \textit{coarsest grid}$ ($|\Omega_h| = 4 \times 4$), solve (25) using the LGD method and then stop. Else continue with following step.

2) Pre-smoothing:

For $k = 1$ to ν_1, $\left[\overline{\boldsymbol{u}}^h\right] \leftarrow Smoother(\overline{\boldsymbol{u}}^h, g_1^h, g_2^h, R^h, T^h, \theta_1, \theta_2, \omega, \text{Siter})$

(Siter represents the maximum number of the inner iterations)

3) Restriction to the coarse grid:

$\overline{u}_1^H \leftarrow I_h^H \overline{u}_1^h$, $\overline{u}_2^H \leftarrow I_h^H \overline{u}_2^h$, $R^H \leftarrow I_h^H R^h$, $T^H \leftarrow I_h^H T^h$

4) Set the initial solution for the coarse-grid problem:

$\left[\widetilde{u}_1^H, \widetilde{u}_2^H\right] \leftarrow \left[\overline{u}_1^H, \overline{u}_2^H\right]$

5) Compute the new right-hand side for the coarse-grid problem:

$g_1^H \leftarrow I_h^H(g_1^h - \mathcal{N}_1^h(\overline{\boldsymbol{u}}^h)) + \mathcal{N}_1^H\left(\overline{\boldsymbol{u}}^H\right)$, $g_2^H \leftarrow I_h^H(g_2^h - \mathcal{N}_2^h\left(\overline{\boldsymbol{u}}^h\right)) + \mathcal{N}_2^H(\overline{\boldsymbol{u}}^H)$

6) Implement the NMG method on the coarse-grid problem:

$\overline{\boldsymbol{u}}^H \leftarrow \text{NMGCYC}(\overline{\boldsymbol{u}}^H, g_1^H, g_2^H, R^H, T^H, \nu_1, \nu_2, \theta_1, \theta_2, \omega, \text{Siter})$

7) Add the coarse-grid corrections:

$\overline{u}_1^h \leftarrow \overline{u}_1^h + I_H^h\left(\overline{u}_1^H - \widetilde{u}_1^H\right)$, $\overline{u}_2^h \leftarrow \overline{u}_2^h + I_H^h\left(\overline{u}_2^H - \widetilde{u}_2^H\right)$

8) Post-smoothing:

For $k = 1$ to ν_2, $\left[\overline{\boldsymbol{u}}^h\right] \leftarrow Smoother(\overline{\boldsymbol{u}}^h, g_1^h, g_2^h, R^h, T^h, \theta_1, \theta_2, \omega, \text{Siter})$

To solve $\boldsymbol{u}$-subproblem using (25) numerically, the proposed NMG method is stopped if the maximum number of the MG cycles ε_1 is reached or the mean of the relative residuals obtained from the discrete EL equation (25) is smaller than a small prescribed number $\varepsilon_2 > 0$.

Finally, the pseudo-code implementation of the proposed NMG method can be given in Algorithm 2.

Algorithm 2 The NMG algorithm for $\boldsymbol{u}$-subproblem

$$\overline{\boldsymbol{u}}^h \leftarrow \text{NMG}\left(\overline{\boldsymbol{u}}^h, g_1^h, g_2^h, R^h, T^h, \nu_1, \nu_2, \theta_1, \theta_2, \omega, \text{Siter}, \overrightarrow{\varepsilon}\right)$$

1) Initialization: select $\overrightarrow{\varepsilon} = (\varepsilon_1, \varepsilon_2)^\top$, set $K = 0$, $[\overline{\boldsymbol{u}}^h]^{[K]} = \overline{\boldsymbol{u}}^h$ and RelRes $= \varepsilon_2 + 1$.

2) **Repeat**

2.1) $[\overline{\boldsymbol{u}}^h]^{[K+1]} \leftarrow \text{NMGCYC}([\overline{\boldsymbol{u}}^h]^{[K]}, g_1^h, g_2^h, R^h, T^h, \nu_1, \nu_2, \theta_1, \theta_2, \omega, \text{Siter})$.

2.2) Compute RelRes $= \text{mean}\{\frac{||g_l^h - \mathcal{N}_l^h([\overline{\boldsymbol{u}}^h]^{[K+1]})||}{||g_l^h - \mathcal{N}_l^h([\overline{\boldsymbol{u}}^h]^{[0]})||} \mid l = 1, ..., 2\}$.

2.3) Set $K = K + 1$.

until a stopping rule for NMG method is met.

3.3. The MG smoother

Following [18, 21, 11, 22, 23, 7, 24, 25, 26], the local Fourier analysis can be used to guarantee that there exists an efficient point-wise smoother within a MG method for solving the discrete nonlinear system (25). To obtain a high-potential point-wise smoother, this work proposes a coupled outer-inner iteration method in a FP framework.

Let ν denote the index for the outer step. We start the outer iteration by introducing the iterative scheme as follows

$$\begin{cases} -\theta_1 \triangle u_1^{[\nu+1]} + f_1(\boldsymbol{u}^{[\nu+1]}) = g_1, \\ -\theta_2 \triangle u_2^{[\nu+1]} + f_2(\boldsymbol{u}^{[\nu+1]}) = g_2. \end{cases}$$

Since this iterative scheme is fully implicit, a linearization procedure of the nonlinear term $f_l(\boldsymbol{u}^{[\nu+1]})$ is required. As pointed out in [21, 11, 7, 24, 25, 26], it is appropriate to apply a global linearization technique with the nonlinear terms $f_l(\boldsymbol{u}^{[\nu+1]})$. Thus, the resulting approximation is given by

$$\begin{aligned} f_l(\boldsymbol{u}^{[\nu+1]} = (u_1^{[\nu+1]}, u_2^{[\nu+1]})^\top) &\approx f_l(u_1^{[\nu]}, u_2^{[\nu]}) + \partial_{u_1} f_l(u_1^{[\nu]}, u_2^{[\nu]}) \delta u_1^{[\nu]} + \partial_{u_2} f_l(u_1^{[\nu]}, u_2^{[\nu]}) \delta u_2^{[\nu]}, \\ &= f_l(u_1^{[\nu]}, u_2^{[\nu]}) + \sigma_{l1}^{[\nu]} \delta u_1^{[\nu]} + \sigma_{l2}^{[\nu]} \delta u_2^{[\nu]}, \end{aligned} \tag{27}$$

where

$$\begin{aligned} \sigma_{l1}(\boldsymbol{u}^{[\nu]}) &= \partial_{u_1} f_l(u_1^{[\nu]}, u_2^{[\nu]}) = (\partial_{u_l} T(\boldsymbol{u}^{[\nu]}))(\partial_{u_1} T(\boldsymbol{u}^{[\nu]})) + (T(\boldsymbol{u}^{[\nu]}) - R)(\partial_{u_1 u_l} T(\boldsymbol{u}^{[\nu]})), \\ \sigma_{l2}(\boldsymbol{u}^{[\nu]}) &= \partial_{u_2} f_l(u_1^{[\nu]}, u_2^{[\nu]}) = (\partial_{u_l} T(\boldsymbol{u}^{[\nu]}))(\partial_{u_2} T(\boldsymbol{u}^{[\nu]})) + (T(\boldsymbol{u}^{[\nu]}) - R)(\partial_{u_2 u_l} T(\boldsymbol{u}^{[\nu]})). \end{aligned}$$

To obtain a simple and fast iterative scheme, we use the approximations for σ_{l1} and σ_{l2} as introduced by [32, 11, 33, 34]. Therefore, we have

$$\begin{aligned} \sigma_{11}(\boldsymbol{u}^{[\nu]}) &= (\partial_{u_1} T_{\boldsymbol{u}^{[\nu]}})\,(\partial_{u_1} T_{\boldsymbol{u}^{[\nu]}}) + (T_{\boldsymbol{u}^{[\nu]}} - R)\,(\partial_{u_1 u_1} T_{\boldsymbol{u}^{[\nu]}}) \approx (\partial_{u_1} T_{\boldsymbol{u}^{[\nu]}})\,(\partial_{u_1} T_{\boldsymbol{u}^{[\nu]}})\,, \\ \sigma_{22}(\boldsymbol{u}^{[\nu]}) &= (\partial_{u_2} T_{\boldsymbol{u}^{[\nu]}})\,(\partial_{u_2} T_{\boldsymbol{u}^{[\nu]}}) + (T_{\boldsymbol{u}^{[\nu]}} - R)\,(\partial_{u_2 u_l} T_{\boldsymbol{u}^{[\nu]}}) \approx (\partial_{u_2} T_{\boldsymbol{u}^{[\nu]}})\,(\partial_{u_2} T_{\boldsymbol{u}^{[\nu]}})\,, \\ \sigma_{12}(\boldsymbol{u}^{[\nu]}) &= \sigma_{21}(\boldsymbol{u}^{[\nu]}) = 0. \end{aligned}$$

These approximations leads to the following linearized system

$$\begin{cases} -\theta_1 \triangle u_1^{[\nu+1]} + \sigma_{11}(\boldsymbol{u}^{[\nu]}) u_1^{[\nu+1]} = g_1 - f_1(\boldsymbol{u}^{[\nu]}) + \sigma_{11}(\boldsymbol{u}^{[\nu]}) u_1^{[\nu]}, \\ -\theta_2 \triangle u_2^{[\nu+1]} + \sigma_{22}(\boldsymbol{u}^{[\nu]}) u_2^{[\nu+1]} = g_2 - f_2(\boldsymbol{u}^{[\nu]}) + \sigma_{22}(\boldsymbol{u}^{[\nu]}) u_2^{[\nu]}. \end{cases} \tag{28}$$

Next, we apply the finite difference discretization as discussed in Section 3.1 with (28) and solve the associated linear system for each outer step ν by the Gauss-Seidel (GS) method

as the inner iteration. The kth step of the GS method defined at a grid point (x_{1_i}, x_{2_j}) is then given by

$$(\boldsymbol{u}^{[\nu+1]})_{i,j}^{[k+1]} = (\boldsymbol{v})_{i,j}^{[k+1]}, \tag{29}$$

where

$$(\boldsymbol{v})_{i,j}^{[k+1]} = \begin{pmatrix} \dfrac{(g_1)_{i,j} - f_1(\boldsymbol{u}^{[\nu]})_{i,j} + (\sigma_{11}(\boldsymbol{u}^{[\nu]}))_{i,j}(u_1^{[\nu]})_{i,j} + (\theta_1/h_1^2)(\overline{\Sigma})_{i,j}(u_1^{[\nu+1]})_{i,j}^{[k+1/2]}}{(\theta_1/h_1^2)(\Sigma)_{i,j} + (\sigma_{11}(\boldsymbol{u}^{[\nu]}))_{i,j}} \\ \\ \dfrac{(g_2)_{i,j} - f_2(\boldsymbol{u}^{[\nu]})_{i,j} + (\sigma_{22}(\boldsymbol{u}^{[\nu]}))_{i,j}(u_2^{[\nu]})_{i,j} + (\theta_2/h_1^2)(\overline{\Sigma})_{i,j}(u_2^{[\nu+1]})_{i,j}^{[k+1/2]}}{(\theta_2/h_1^2)(\Sigma)_{i,j} + (\sigma_{22}(\boldsymbol{u}^{[\nu]}))_{i,j}} \end{pmatrix},$$

and

$$(\overline{\Sigma})_{i,j}(u_l^{[\nu+1]})_{i,j}^{[k+1/2]} = (u_l^{[\nu+1]})_{i+1,j}^{[k]} + (u_l^{[\nu+1]})_{i-1,j}^{[k+1]} + \gamma^2(u_l^{[\nu+1]})_{i,j+1}^{[k]} + \gamma^2(u_l^{[\nu+1]})_{i,j-1}^{[k+1]}.$$

Here the superscripts k, $k+1/2$, and $k+1$ denote the current, intermediate and new approximations computed by the GS method.

In order to obtain more efficiency, one introduces a relaxation parameter $\omega \in (0, 2)$ and iterates the successive over relaxation (SOR) steps by

$$(\boldsymbol{u}^{[\nu+1]})_{i,j}^{[k+1]} = (1-\omega)\,(\boldsymbol{u}^{[\nu+1]})_{i,j}^{[k]} + \omega(\boldsymbol{v})_{i,j}^{[k+1]}. \tag{30}$$

Finally, the implementation of the proposed MG smoother (30) on Ω_h is summarized in Algorithm 3.

Algorithm 3 The MG smoother for the proposed NMG algorithm (Algorithm 2)

$\overline{\boldsymbol{u}}^h \leftarrow Smoother(\overline{\boldsymbol{u}}^h, g_1^h, g_2^h, R^h, T^h, \theta_1, \theta_2, \omega, \text{Siter})$

1) Use input parameters to compute $(\boldsymbol{v})_{i,j}$ for all $1 \le i \le n_1$ and $1 \le j \le n_2$.

2) Perform SOR steps for $k = 0, 1, 2, ..., \text{Siter}$

   2.1) Update $(\overline{\boldsymbol{u}}^h)_{i,j}^{[k+1]}$ by (30) for all $1 \le i \le n_1$ and $1 \le j \le n_2$.

3.4. The closed form expressions for $\boldsymbol{w}_1$, $\boldsymbol{w}_2$, $\boldsymbol{\lambda}_1$, and $\boldsymbol{\lambda}_2$

Based on the formulation (23), we can get the closed form expressions for $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ as given by

$$(\boldsymbol{w}_l)_{i,j} = \max(|(\boldsymbol{S}_l)_{i,j}| - \frac{\alpha}{\theta_l}, 0)\frac{(\boldsymbol{S}_l)_{i,j}}{|(\boldsymbol{S}_l)_{i,j}|}, \tag{31}$$

where $\boldsymbol{S}_l = (\delta_{x_1}^+(u_l)_{i,j} - \frac{(\lambda_{l,1})_{i,j}}{\theta_l}, \delta_{x_2}^+(u_l)_{i,j} - \frac{(\lambda_{l,2})_{i,j}}{\theta_l})^\top$ for $l = 1, 2$.

Similarly, based on the formulations in (11) and (12), we may update all the Lagrangian multipliers by

$$(\boldsymbol{\lambda}_1^{(k+1)})_{i,j} = (\boldsymbol{\lambda}_1^{(k)})_{i,j} + \theta_1((\boldsymbol{w}_1)_{i,j}^{(k+1)} - \nabla^+(u_1)_{i,j}^{(k+1)}), \tag{32}$$

$$(\boldsymbol{\lambda}_2^{(k+1)})_{i,j} = (\boldsymbol{\lambda}_2^{(k)})_{i,j} + \theta_2((\boldsymbol{w}_2)_{i,j}^{(k+1)} - \nabla^+(u_2)_{i,j}^{(k+1)}). \tag{33}$$

4. Numerical experiments

In this section, we present a number of numerical experiments to

(i) compare the overall performance of the proposed ALM with two related numerical solutions, which are the LGD method (5) and the NMG method by Chumchob [7];

(ii) assess the accuracy and efficiency of the proposed ALM with regard to parameter changes.

We note first that four registration problems consisting of four real medical images to be denoted as Problems 1-4 were selected for the experiments, as shown respectively in Figure 1. Second, we used $n_1 = n_2 = n$ (i.e. we assume the grid spacing $h_1 = h_2 = h = 1/n$) in all numerical tests. Third, the bilinear interpolation was employed to compute the transformed template image $T_{\boldsymbol{u}}$ once the deformations are found in all cases. We also note that all numerical algorithms for the proposed ALM, LGD and NMG methods were started with $\boldsymbol{u}^{[0]} = \boldsymbol{0}$ and implemented under MATLAB R2018a and run on a machine configured with Intel(R) Core(TM) i7 Quad-core 4.2GHz and 32GB of RAM.

4.1. Performance comparison with the other two methods

In this test, the performance of three different methods for TV-based image registration model is compared. We apply the relative SSD (a qualitative measure in the accuracy), time per iteration (in seconds), total CPU time (in seconds), and total iterations to perform our evaluation on the four registration problems shown in Figure 1.

For the LGD method, we chose $\tau = 10^{-5}$ and $\beta = 10^{-6}$ and implemented a direct solver using MATLAB's \-operator (also known as `mldivide` operator) for solving each linear system of equations arising from applying the FP method with the nonlinear discrete EL system.

For the NMG method, we implemented with the FAS-NMG method of Chumchob [7]. The details of our numerical implementation can be summarized as follows. We solved the

Figure 1: Four registration problems of the real medical images. (*left column*) Reference images; (*right column*) Template images; (*top row*) Problem 1; (*2nd row*) Problem 2; (*3rd row*) Problem 3; (*bottom row*) Problem 4.

nonlinear discrete EL system with $\beta = 10^{-6}$ in a V-cycle framework, where the coarsest grid is 4×4. The LGD method was used to solve the coarsest problem with $\tau = 10^{-3}$. We used the standard coarsening method and standard interpolation technique for transferring data between grids. We applied 15 pre- and 15 post-smoothing steps with a so-called FP-SOR smoother.

For the proposed ALM, we chose the penalty parameters $\theta_1 = \theta_2 = 1/10$. We used the predefined numbers $\epsilon_1 = 10^{-2},\ \epsilon_2 = 10^{-4},\ \epsilon_3 = \epsilon_4 = 10^{-8}$ and $\epsilon_5 = 200$ with Algorithm 1. The predefined numbers $\varepsilon_1 = 20$ and $\varepsilon_2 = 10^{-6}$ were applied by the proposed NMG method

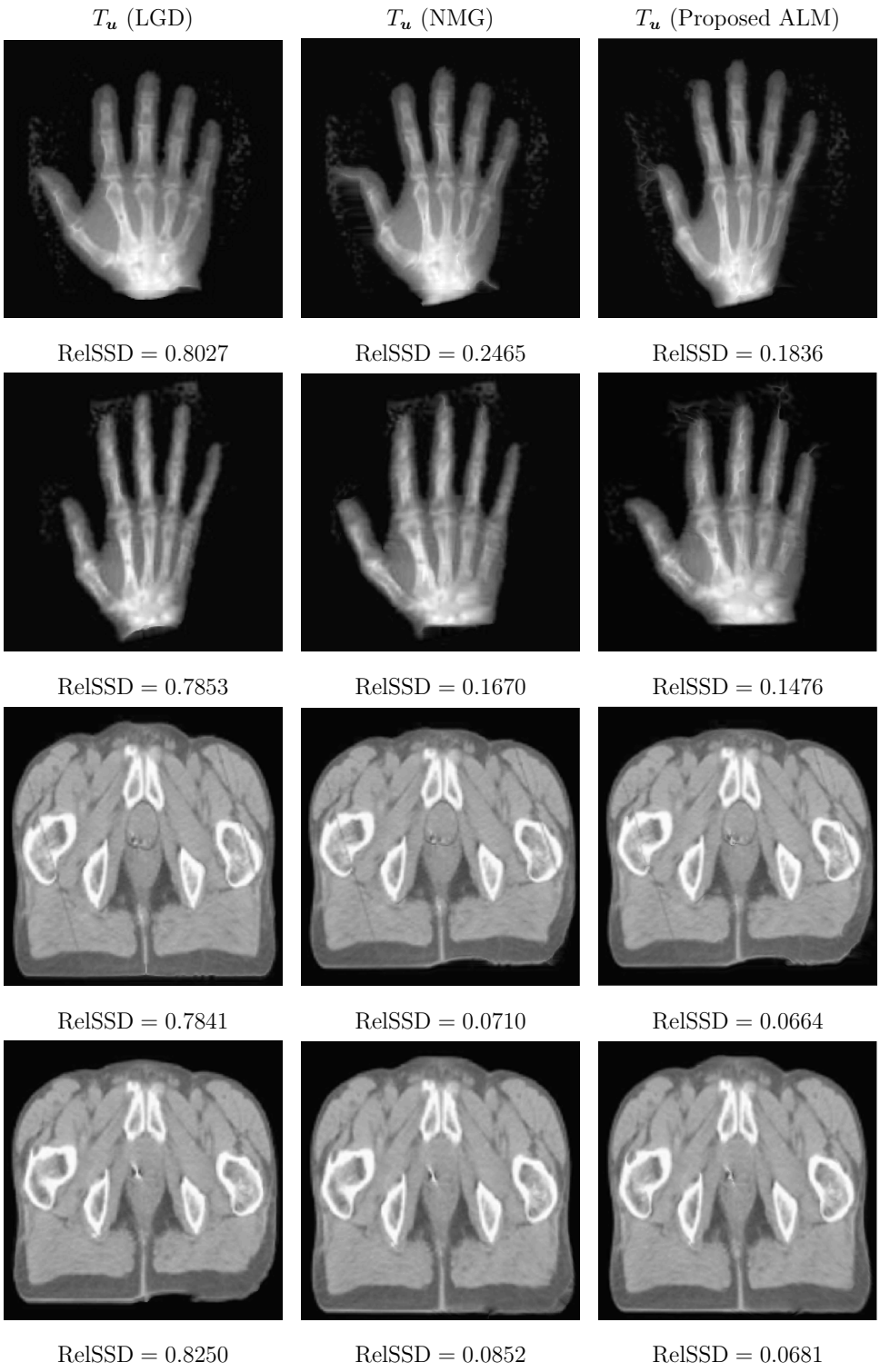


Figure 2: Qualitative comparison of registered images by three different methods on the four registration problems in Figure 1. (*left*) Registered images by LGD; (*middle*) registered images by NMG; (*right*) registered images by the proposed ALM. 'RelSSD' means the relative SSD defined in (13) to represent the dissimilarities between R and $T_{\boldsymbol{u}}$. Note that the small the value of RelSSD is, the accurate registered image is obtained.

in Algorithm 2 to solve $\boldsymbol{u}$-subproblem. In all registration problems, the proposed NMG method with the MG smoother in Algorithm 3 was performed with the MG parameters, $\nu_1 = \nu_2 = 5$ (5 pre- and 5 post-smoothing steps), $\omega = 1.85$ (the relaxation parameter), and Siter = 2 (2 iterations of SOR steps).

All numerical methods in this test used the same regularization parameter $\alpha = 1/50$ with the grid spacing $h = 1/256$. Each method was stopped when the relative solution, $\max\{\frac{\|u_1^{(m)} - u_1^{(m-1)}\|}{\|u_1^{(m)}\|}, \frac{\|u_2^{(m)} - u_2^{(m-1)}\|}{\|u_2^{(m)}\|}\}$, was brought below 10^{-4} or the number of iterations increases beyond a predefined number. Here we used the predefined numbers $m_{\max}^{\text{LGD}} = 200$ for the LGD method, $m_{\max}^{\text{NMG}} = 50$ for the NMG method, and $m_{\max}^{\text{ALM}} = 200$ for the proposed ALM.

Table 1: Comparison of the relative SSD, time per iteration, total CPU time (s), and total iterations by three different numerical methods on the four registration problems in Figure 1 with $\alpha = \frac{1}{50}$ and $h = \frac{1}{256}$. $*$ indicates that maximum number of iterations reached without convergence.

Method		**Registration problem**			
		1	**2**	**3**	**4**
LGD	Relative SSD	0.8027	0.7853	0.7841	0.8250
	Time per iteration (s)	0.2467	0.2458	0.2457	0.2463
	Total CPU time (s)	49.3423	49.1558	49.1301	49.2524
	Total iterations	200*	200*	200*	200*
NMG	Relative SSD	0.2465	0.1670	0.0710	0.0852
	Time per iteration (s)	0.4941	0.4930	0.4814	0.4914
	Total CPU time (s)	24.7045	24.6477	24.0701	24.5724
	Total iterations	50*	50*	50*	50*
Proposed ALM	Relative SSD	**0.1836**	**0.1476**	**0.0664**	**0.0681**
	Time per iteration (s)	0.5970	0.6893	0.7105	0.7138
	Total CPU time (s)	28.4776	40.6674	37.6567	38.5466
	Total iterations	41	59	53	54

From Tables 1, one can see that only the proposed ALM converges, whereas the LGD and NMG methods are unable to converge in a reasonable number of iterations. As expected, we have found in this case that the small value of β has a significant effect on the accuracy of the registered images and the convergence of the LGD and NMG method. Moreover, Figure 2 shows that for each registration problem the proposed ALM yields the best value of the relative SSD. It is important to note that the smaller the value of the relative SSD is, the accurate registered image is achieved. This evidence ensures that the registration results by the proposed ALM are more reliable than the LGD and NMG methods. Particularly, Figure 3 shows that the proposed ALM is computationally efficient than the other two methods

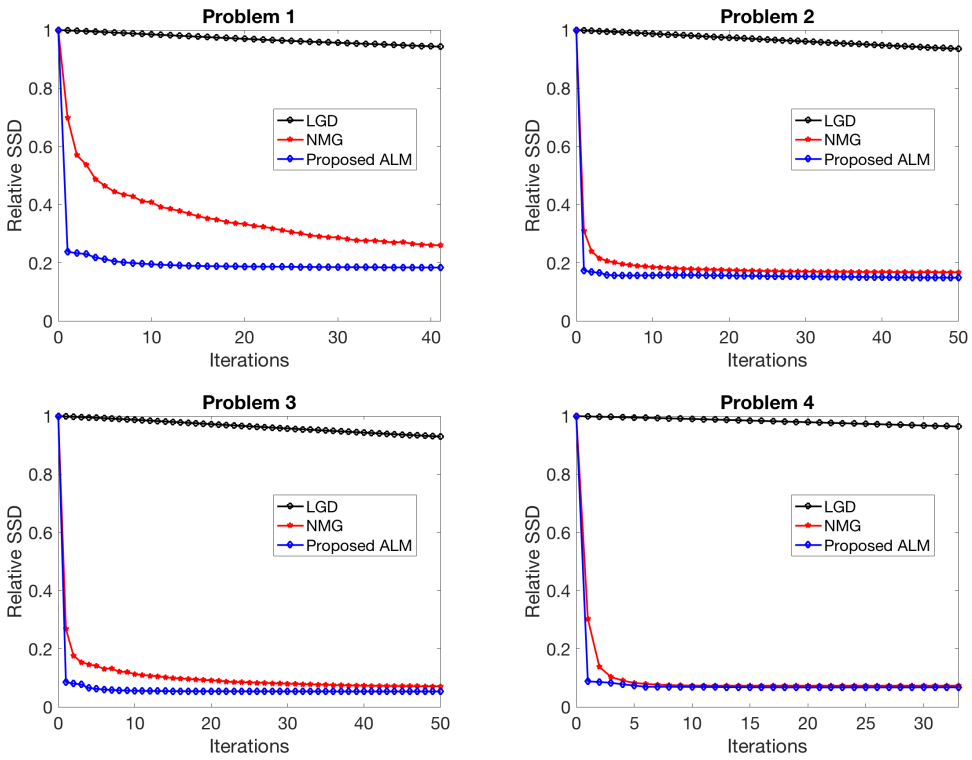


Figure 3: History of the relative SSD by three different methods in solving the four registration problems in Figure 1.

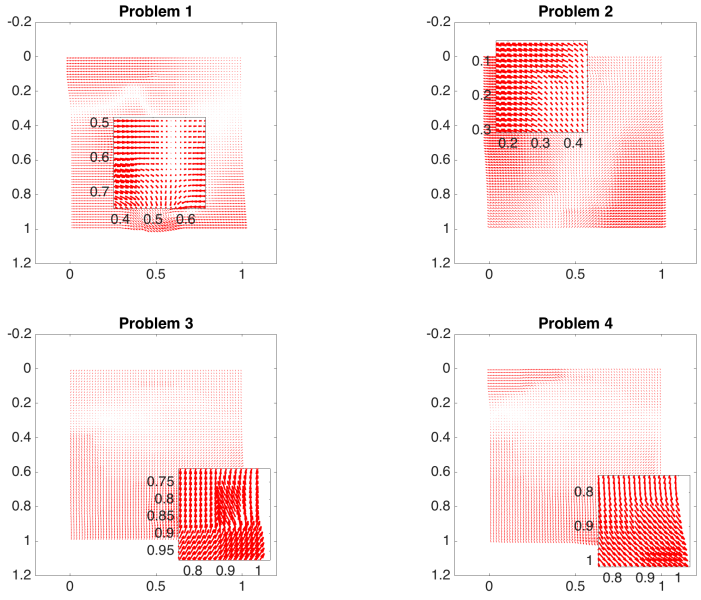


Figure 4: Constructed deformations by the proposed ALM for the four registration problems in Figure 1. Note that multiple motions in the corresponding image and motion discontinuities at the boundaries of the local regions determined by the constructed deformation can be observed in the corresponding close-up regions.

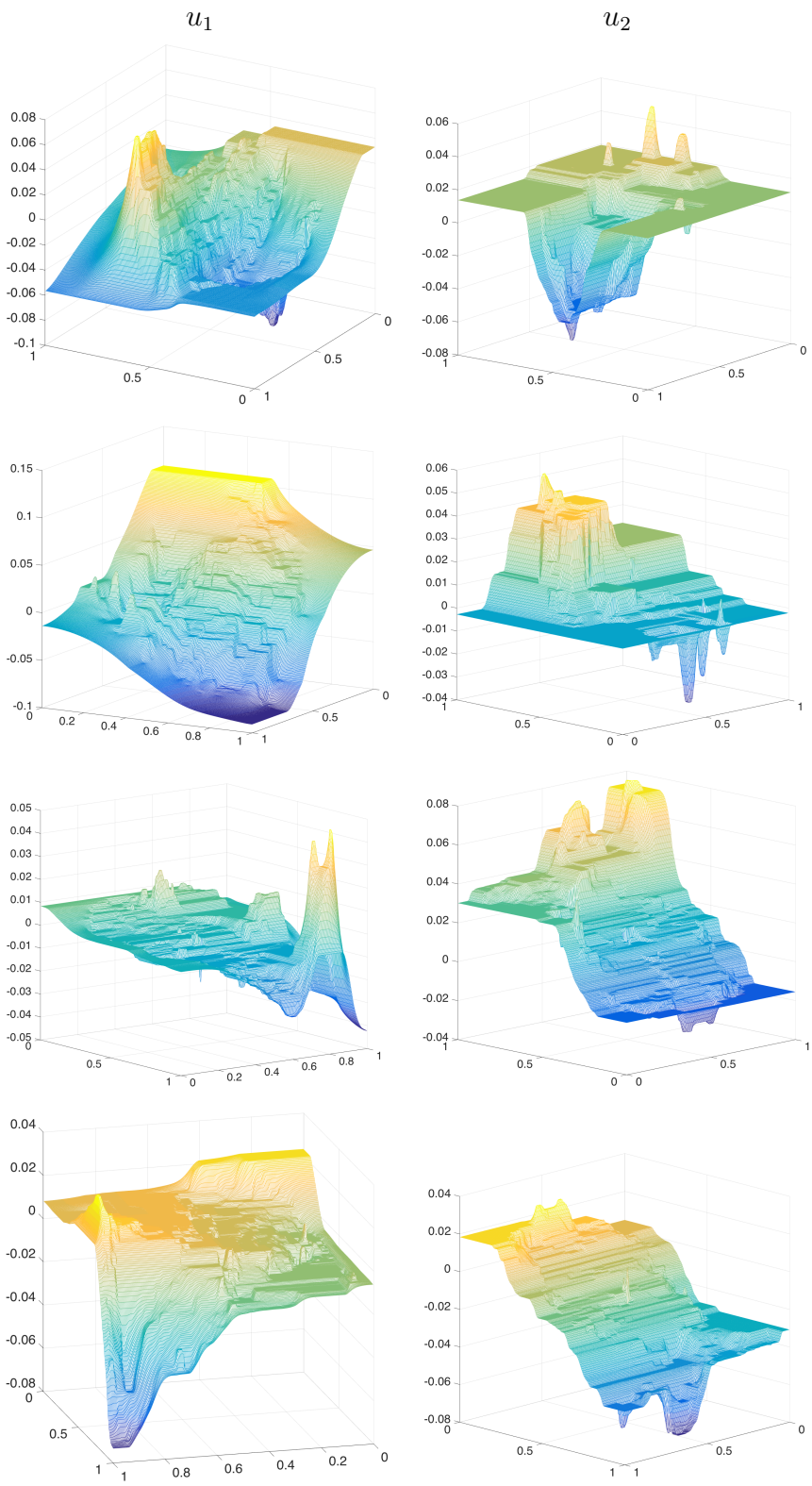


Figure 5: Constructed surfaces of both components of the constructed deformations by the proposed ALM for the four registration problems in Figure 1. (*left*) the first component u_1; (*right*) the second component u_2; (*top row*) Problem 1; (*2nd row*) Problem 2; (*3rd row*) Problem 3; (*bottom row*) Problem 4.

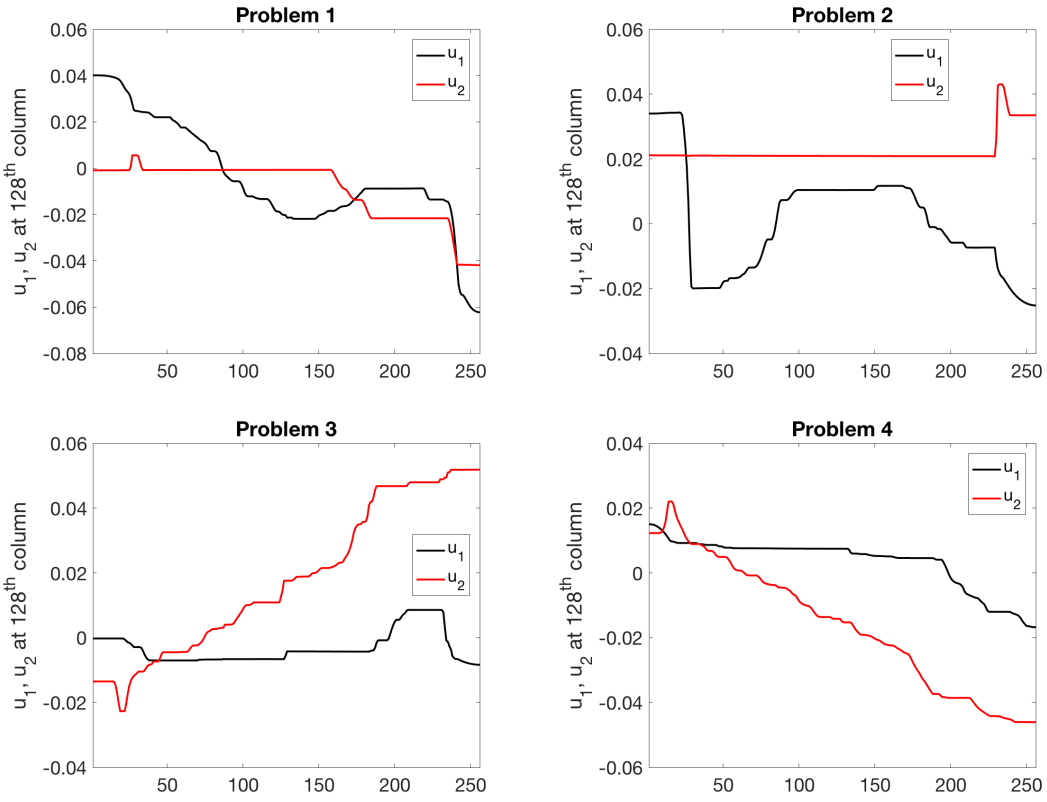


Figure 6: The middle slices of the corresponding surfaces of both components of the constructed deformations shown in Figure 5 by the proposed ALM.

in delivering the high quality of the registered images. The proposed ALM takes only a few iterations (say 5 iterations) to drop the relative SSD below 0.20, which means that the dissimilarities between the reference and registered images have been reduced more than 80% within the first 5 iterations. This is a remarkable result to conclude that the computational performance of the proposed ALM in solving TV-based image registration is much more efficient than those of the other two methods.

In terms of computation times per iteration, we can see that the proposed ALM requires to solve two sub-problems for $\boldsymbol{u}$ and $\boldsymbol{w}$ in each step. Thus, in accordance with our measurements, we expect that more computation time per iteration is required for the proposed ALM than those of the other two methods. As can be seen from Table 1, our numerical experiments reflect this expectation.

In terms of total CPU times, Table 1 illustrates the LGD method is the slowest method, while the NMG method is slightly faster than the proposed ALM. However, both LGD and NMG method are unable to converge. Thus we can conclude that the LGD and NMG methods are less computationally efficient than the proposed ALM.

In Figure 4 we present the constructed deformations by the proposed ALM for all reg-

istration problems. In Figure 5, we show the constructed surfaces of both components of the deformations in Figure 4. In Figure 6, we present the middle slices of the constructed surfaces of the components of the deformations in Figure 5. As expected from the use of TV regularizer, the visual inspection in Figure 4 shows that the proposed ALM delivers the visually appealing results in preserving the discontinuities of the constructed deformation as shown in the corresponding close-up regions. We can also see in Figure 4 that the constructed deformation yields multiple motions in the image to be registered and the motion discontinuities can be observed at the boundaries of the local regions. We see further that the corresponding deformation in each registration problem is neither smooth nor reflecting a homogeneous motion. For each local region, the motion tends to change smoothly, and the gradients of the constructed deformation in this area are small. At the boundaries of the local regions, the gradients of the constructed deformation are large, and the discontinuities of the constructed deformation can be observed. Moreover, we can see from Figures 5 and 6 that the constructed surfaces of both components of the deformations are non-smooth and their middle slices are almost piecewise constant for all registration problems.

To summarize we have successfully developed the efficient and effective numerical method for TV-based image registration model. Our registration results on the real medical applications shown in Figure 1 demonstrate that the proposed ALM is more computationally efficient and effective than the other two methods. The most attractive features of our proposed method is that it is able to produce the high quality of the registered images in a reasonable number of iterations while it satisfies the requirement in constructing the deformations by the TV regularizer.

4.2. Performance tests with the regard to parameter changes

We now present numerical results from several test cases, to assess the accuracy and efficiency of our proposed numerical techniques with the regard to parameter changes.

4.2.1. h-independence test

One of the key properties of MG techniques is that their convergence does not depend on the number of grid points. Thus, in this test we designed our numerical experiments to investigate this property with the proposed ALM in Algorithm 1, and to back up our proposed NMG method in Algorithms 2.

We implemented the proposed ALM with the same regularization parameter $\alpha = \frac{1}{250}$ and the penalty parameters θ_1 and θ_2 as shown in Table 2. In order to stop our proposed

Table 2: Registration results for Problems 1-4 in Figure 1 by the proposed ALM in Algorithm 1 with $\alpha = \frac{1}{250}$ and different values of grid spacing h. Note that 'Ratio' means the ratio of the total CPU times in increasing both image dimensions by a factor of 2.

Registration problem		**Grid spacing (h)**			
		$\frac{1}{256}$	$\frac{1}{512}$	$\frac{1}{1024}$	$\frac{1}{2048}$
1 ($\theta_1 = \theta_2 = 1$)	Relative SSD	0.0968	0.0958	0.0951	0.0951
	Total iterations	22	22	21	21
	Total CPU time (s)	22.4489	74.1501	308.4895	1,308.3476
	Ratio	–	3.3031	4.1603	4.2411
	Average MG cycles	6	6	6	6
2 ($\theta_1 = \theta_2 = 1$)	Relative SSD	0.1028	0.1027	0.1025	0.1002
	Total iterations	16	16	15	15
	Total CPU time (s)	16.1900	58.8500	257.0531	1,117.0938
	Ratio	–	3.6350	4.3679	4.4294
	Average MG cycles	7	7	7	7
3 ($\theta_1 = \theta_2 = \frac{1}{4}$)	Relative SSD	0.0359	0.0359	0.0358	0.0357
	Total iterations	36	34	33	33
	Total CPU time (s)	38.8623	137.9971	577.3338	2,578.2640
	Ratio	–	3.5509	4.1837	4.4658
	Average MG cycles	7	7	7	7
4 ($\theta_1 = \theta_2 = \frac{1}{4}$)	Relative SSD	0.0377	0.0375	0.0374	0.0373
	Total iterations	31	31	30	30
	Total CPU time (s)	30.9928	116.8559	500.9036	2,238.7849
	Ratio	–	3.7704	4.2865	4.4695
	Average MG cycles	6	6	6	6

ALM, we used the same predefined numbers used in Section 4.1. We also used the same MG parameters to solve $\boldsymbol{u}$-subproblem as given in the previous section.

In the numerical results shown in Table 2, one can see five quantities: the relative SSD, the total iterations, the total CPU times (in seconds), the ratio of the total CPU times in increasing both image dimensions by a factor of 2, and the number of average MG steps used to solve $\boldsymbol{u}$-subproblem by Algorithm 2 with different values of grid spacing h.

As expected from a numerical technique using a MG framework, Table 2 shows that Algorithm 1 not only converges within a few iterations, but it also provides the accurate registration results because the dissimilarities between the reference and registered images given have been reduced more than 90% for Problem 1, 89% for Problem 2, and 96% for Problems 3 and 4. Moreover, the NMG method in Algorithm 2 can reduce the mean of the relative residuals to $\varepsilon_2 = 10^{-6}$ within a few MG steps for different values of h. In addition, we see an increase in the total CPU time by a factor of approximately 4.5 when both image

Table 3: Registration results for Problems 1-4 in Figure 1 by the proposed ALM in Algorithm 1 with $h = \frac{1}{256}$ and different values of the regularization parameter α.

Registration problem		**Regularization parameter (α)**					
		$\frac{1}{250}$	$\frac{1}{200}$	$\frac{1}{150}$	$\frac{1}{100}$	$\frac{1}{50}$	$\frac{1}{25}$
1 ($\theta_1 = \theta_2 = 1$)	Relative SSD	0.0968	0.1015	0.1080	0.1313	0.2099	0.3466
	Total iterations	20	21	25	30	33	49
	Average MG cycles	6	6	6	6	6	6
2 ($\theta_1 = \theta_2 = 1$)	Relative SSD	0.1028	0.1028	0.1069	0.1174	0.1675	0.2104
	Total iterations	16	18	20	27	37	40
	Average MG cycles	7	6	6	6	6	6
3 ($\theta_1 = \theta_2 = \frac{1}{4}$)	Relative SSD	0.0359	0.0385	0.0436	0.0536	0.0958	0.2251
	Total iterations	36	44	49	65	90	125
	Average MG cycles	7	7	6	6	5	5
4 ($\theta_1 = \theta_2 = \frac{1}{4}$)	Relative SSD	0.0377	0.0406	0.0449	0.0548	0.0863	0.1909
	Total iterations	31	34	49	57	102	139
	Average MG cycles	6	6	5	5	5	5

dimensions is increased by a factor of 2. This shows that the proposed ALM has an optimal efficiency resulting from the NMG method proposed in Algorithm 2, which should be of order $\mathcal{O}(N \log(N))$ where $N = n^2$. Then we can conclude that the proposed ALM has no effect on the convergence for different values of h.

4.2.2. α-dependence test

Next we evaluate to show how our proposed ALM in Algorithm 1 is affected with varying the regularization parameter α.

To this end, we performed the proposed ALM on the four registration problems in Figure 1 with the same grid spacing $h = \frac{1}{256}$ and the penalty parameters θ_1 and θ_2 as given in Table 2. Our proposed ALM in Algorithm 1 was stopped with the same predefined numbers used in Section 4.1. We also applied the same predefined number and MG parameters as shown in Section 4.1 with Algorithms 2 and 3 to solve $\boldsymbol{u}$-subproblem.

Table 3 presents the numerical results by the proposed ALM in Algorithm 1 with different values of the regularization parameter α. It contains three quantities: the relative SSD, the total iterations, and the number of average MG steps used to solve $\boldsymbol{u}$-subproblem by Algorithm 2.

As presented in Table 3, decreasing the values of α has significant effects on the accuracy of the registered images and the convergence of the proposed ALM in Algorithm 1, whereas the value of α has only a small effect on the convergence of the proposed NMG method in

Algorithm 2. We can see that large α is not needed as small ones give better registration results, typically $\alpha = \frac{1}{250}, \frac{1}{200}, \frac{1}{150}$, and $\frac{1}{100}$. It is important to note that the process to select the optimal value of α is a separate but important issue because it is in general unknown a priori and it significantly affects on the accuracy of registered images and the ALM performance.

4.2.3. (θ_1, θ_2)-dependence test

We now present numerical results from several test cases to evaluate the registration performance of our proposed ALM with different values of θ_1 and θ_2.

We performed the proposed ALM in Algorithm 1 on the four registration problems shown in Figure 1 using $h = \frac{1}{256}$ and $\alpha = \frac{1}{250}$. The predefined numbers and the MG parameters for Algorithms 1-3 are the same values used in Section 4.1.

We note that we take $\theta_1 = \theta_2 = \theta$ in this numerical test to evaluate the performance of the proposed ALM. Table 4 presents the registration results for the four registration problems by the proposed ALM in Algorithm 1 with different values of θ. Three qualities for the performance tests are the relative SSD, the total iterations, and the number of average MG steps used to solve $\boldsymbol{u}$-subproblem by Algorithm 2.

One sees from Table 4 that as θ is decreased from 10 to $\frac{1}{10}$ the dissimilarities between the given images decrease (the accuracy of the registered image increase), whereas the cost of the proposed ALM increases. We observe that the smaller θ is the better the proposed NMG method performs and an increase in the number of smoothing steps is required to achieve fast MG convergence for large θ. Next, we see that decreasing the value of θ leads to the best registration result at $\theta = \frac{1}{10}$. We see further that large values of θ are not required as small ones deliver better registration performance. Typically, θ should be between $\frac{1}{10}$ and $\frac{1}{2}$ to give the accurate registration results in a reasonable number of iterations.

5. Concluding remarks

In this paper, we first explained how standard methods solve the TV-based image registration model. Next, we discussed how we are needed to develop a new method. In order to efficiently solve the model, we therefore proposed to use augmented Lagrangian method. We separated the associated minimization problem into two subproblems. As a result, the first subproblem is a nonlinear problem and impossible to obtain exact solution, whereas the other one has a closed-form solution. Next, we developed an efficient NMG method to

Table 4: Registration results for Problems 1-4 in Figure 1 by the proposed ALM in Algorithm 1 with $h = \frac{1}{256}$, $\alpha = \frac{1}{250}$ and different values of the penalty parameter θ. Recall that $*$ indicates that maximum number of MG cycles reached without convergence.

Penalty parameter		**Registration problem**			
		1	**2**	**3**	**4**
$\theta = 10$	Relative SSD	0.8128	0.8128	0.4923	0.5056
	Total iterations	3	3	3	3
	Average MG cycles	20*	15	18	18
$\theta = 8$	Relative SSD	0.7721	0.7774	0.4271	0.4401
	Total iterations	3	3	3	3
	Average MG cycles	20*	19	15	15
$\theta = 6$	Relative SSD	0.7098	0.7138	0.3476	0.3595
	Total iterations	3	3	3	3
	Average MG cycles	20*	20*	14	14
$\theta = 4$	Relative SSD	0.6033	0.6029	0.2615	0.2710
	Total iterations	3	3	3	3
	Average MG cycles	20*	20*	19	19
$\theta = 2$	Relative SSD	0.1935	0.2048	0.2187	0.2467
	Total iterations	8	7	10	9
	Average MG cycles	12	14	20*	20*
$\theta = 1$	Relative SSD	0.0968	0.1028	0.2057	0.2443
	Total iterations	20	16	17	19
	Average MG cycles	6	7	20*	20*
$\theta = \frac{1}{2}$	Relative SSD	0.0693	0.0696	0.0670	0.0441
	Total iterations	39	31	25	19
	Average MG cycles	5	5	20*	20*
$\theta = \frac{1}{4}$	Relative SSD	0.0617	0.0586	0.0359	0.0377
	Total iterations	60	44	36	31
	Average MG cycles	4	5	7	6
$\theta = \frac{1}{6}$	Relative SSD	0.0600	0.0549	0.0338	0.0348
	Total iterations	75	60	46	52
	Average MG cycles	4	4	5	5
$\theta = \frac{1}{8}$	Relative SSD	0.0584	0.0536	0.0328	0.0340
	Total iterations	89	100	63	59
	Average MG cycles	4	4	5	4
$\theta = \frac{1}{10}$	Relative SSD	0.0579	0.0535	0.0326	0.0336
	Total iterations	108	191	75	62
	Average MG cycles	4	4	4	4

solve the associated discrete nonlinear system. In order to assess the efficiency and effectiveness of our new method, we tested, using four registration problems of real medical images, how our new method performs. We found in our first numerical test that the registered images by different methods are not identical. Therefore, it can be concluded that differ-

ent numerical algorithms for TV-based image registration model have a significant effect on the accuracy of registered images. We also found that our new method outperforms the existing methods. Moreover we found by the performance comparison in this test that our new method is able to deliver the accurate registration results with the desired properties of the constructed deformations in a reasonable number of iterations. Next, we found in the second test that there are no effect on the convergence of the new method for different numbers of grid points. Moreover, we observed from the third and last tests that the choice of the regularization and penalty parameters is important for the quality of the registered images and the computational performance of the new method. The outlines in selecting these two parameters to obtain the accurate registration results were discussed. Future work will extend the proposed method to high-order variational models for image registration.

Acknowledgments

The author would like to thank the anonymous reviewers for a number of useful suggestions to improve the quality of the paper. The author's work was partially supported by the Thailand Research Fund Grant ♯MRG6080169.

References

[1] J. Maintz, M. Viergever, A survey of medical image registration, Med. Image. Anal. 2 (1) (1998) 1–36.

[2] J. Hajnal, D. Hill, D. Hawkes, Medical Image Registration, The Biomedical Engineering Series, CRC Press, 2001.

[3] V. Mani, S. Arivazhagan, Survey of medical image registration, J. Biomed. Eng. Technol. 1 (2013) 8–25.

[4] A. Sotiras, C. Davatzikos, N. Paragios, Deformable medical image registration: A survey, IEEE Trans. Med. Imaging 32 (2013) 1152–1190.

[5] C. F.-Schauf, S. Henn, L. Hömke, K. Witsch, Total variation based image registration, in Proceedings of the International Conference on PDE-Based Image Processing and Related Inverse Problems Series: Mathematics and Visualization, edited by X.-C. Tai, K.-A. Lie, T.F. Chan and S. Osher, Springer Verlag, 305-323, 2006.

[6] C. F.-Schauf, S. Henn, K. Witsch, Multigrid based total variation image registration, Comput. Visual. Sci. 11 (2008) 101–113.

[7] N. Chumchob, Vectorial total variation-based regularization for variational image registration, IEEE Trans. Image Process. 22 (11) (2013) 4551–4559.

[8] C. Broit, Optimal registration of deformed images, PhD thesis, University of Pennsylvania, USA, 1981.

[9] B. Fischer, J. Modersitzki, Fast diffusion registration, Contemporary Mathematics 313 (2002) 117–129.

[10] B. Fischer, J. Modersitzki, Curvature based image registration, Journal of Mathematical Imaging and Vision 18 (2003) 81–85.

[11] N. Chumchob, K. Chen, A robust multigrid approach for variational image registration models, Journal of Computational and Applied Mathematics 236 (2011) 653–674.

[12] R. Acar, C. Vogel, Analysis of bounded variation penalty method, Inverse Problems 10 (1994) 1217–1229.

[13] T. Chan, K. Chen, On a nonlinear multigrid algorithm with primal relaxation for the image total variation minimization, Journal of Numerical Algorithms 41 (2006) 387–411.

[14] L. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Physica D 60 (1992) 259–268.

[15] T. Chan, P. Mulet, On the convergence of the lagged diffusivity fixed point method in total variation image restoration, SIAM J. Numer. Analysis 36 (2) (2007) 354–367.

[16] C. Vogel, Computational Methods for Inverse Problems, SIAM Publications, Philadelphia, USA, 2002.

[17] C. Vogel, M. Oman, Iterative methods for total variation denoising, SIAM J. Sci. Comput. 17 (1) (1996) 227–238.

[18] N. Chumchob, K. Chen, A variational approach for discontinuity-preserving image registration, East-West Journal of Mathematics Special volume 2010 (2010) 266–282.

[19] Y. Wang, J. Yang, W. Yin, Y. Zhang, A new alternating minimization algorithm for total variation image reconstruction, SIAM J. Imaging Sci. 1 (2008) 248–272.

[20] A. Brandt, Multi-level adaptive solutions to BVPs, Math. Comp. 31 (1977) 333–390.

[21] N. Chumchob, K. Chen, C. Brito, A fourth order variational image registration model and its fast multigrid algorithm, SIAM J. Multiscale Modeling and Simulation 9 (2011) 89–128.

[22] N. Chumchob, K. Chen, An improved variational image registration model and a fast algorithm for its numerical approximation, Numer. Meth. Part. Differ. Equ. 28 (6) (2012) 1966–1995.

[23] N. Chumchob, K. Chen, C. Brito, A new variational model for removal of combined additive and multiplicative noise and a fast algorithm for its numerical approximation, International Journal of Computer Mathematics 90 (1) (2013) 140–161.

[24] S. Jewprasert, N. Chumchob, C. Chantrapornchai, A fourth-order compact finite difference scheme for higher-order PDE-based image registration, East Asian Journal on Applied Mathematics 5 (4) (2016) 361–386.

[25] N. Chumchob, S. Jewprasert, C. Chantrapornchai, Multigrid solution of the nonlinear PDEs arising in elastic image registration with application to a group of monomodal images, ScienceAsia 42 (2016) 415–422.

[26] T. Thompson, K. Chen, An effective diffeomorphic model and its fast multigrid algorithm for registration of lung CT images, Computational Methods in Applied Mathematics (2019) Retrieved 17 Feb. 2019, from doi:10.1515/cmam–2018–0126.

[27] W. Briggs, V. Henson, S. McCormick, A Multigrid Tutorial (2nd Edition), SIAM Publications, Philadelphia, USA, 2000.

[28] W. Hackbusch, Multi-grid Methods and Applications, Springe-Verlag, Berlin, Heidelberg, New York, 1985.

[29] U. Trottenberg, C. Oosterlee, A. Schuller, Multigrid, Academic Press, 2001.

[30] P. Wesseling, Multigrid Methods, Edwards: Philadelphis, PA, U.S.A., 2004.

[31] R. Wienands, W. Joppich, Practical Fourier Analysis for Multigrid Method, Chapman & Hall/CRC, U.S.A., 2005.

[32] J. Modersitzki, Numerical Methods for Image Registration, Oxford, 2004.

[33] E. Haber, J. Modersitzki, A multilevel method for image registration, SIAM J. Sci. Comput. 27 (5) (2006) 1594–1607.

[34] H. Köstler, K. Ruhnau, R. Wienands, Multigrid solution of the optical flow system using a combined diffusion- and curvature-based regularizer, Numer. Linear Algebra Appl. 15 (2008) 201–218.

A New Numerical Method for Gaussian Curvatuve Based Image Restoration

Pornpimon Sroisangwan[1] and Noppadol Chumchob[1,2]

[1]Department of Mathematics, Faculty of Science, Silpakorn University
[2]Centre of Excellence in Mathematics, CHE

[1]p.sroisangwan@gmail.com and [2]chumchob_n@silpakorn.edu

Abstract

Image denoising is one of the most important tasks of image restoration. The goal is to remove noises from a given corrupted digital image for improving the quality and ability of human or machine vision identifications. Total variation (TV) model, which is a classic model in image restoration, is well-known for reducing noise and recovering sharp edges of an image. However, it suffers form the undesired artifacts, such as the staircase effect. To overcome this drawback, we focus on the so-called Gaussian curvature (GC) model, which is a useful tool to remove the staircase effect as well as preserve the sharp edge. As a result, it leads to a complicated partial differential equation that is difficult to solve computationally. We therefore develop an alternating direction method of multiplier (ADMM) to overcome this situation. Numerical experiments on both synthetic and real images show that the proposed method outperforms the standard method in terms of restoration quality.

Mathematics Subject Classification: 62H35, 65N22, 65N55, 74G65, 74G75

Keywords: Gaussian curvature, regularization, Image denoising, Alternating direction method of multipliers

1 Introduction

Let $f : \Omega \subset \mathbb{R}^2 \to V \subset \mathbb{R}$ be an observed noisy image, where Ω is a rectangle of $\mathbb{R}^2$. The goal of image denoising problems is to restore (or recover) the original image $u : \Omega \subset \mathbb{R}^2 \to V \subset \mathbb{R}$ from the noisy image f. According to the maximum likelihood principle [2],

most image denoising problems involve solving the following variational problem:

$$\min_u \{\mathcal{J}_\alpha(u) = \mathcal{D}(u, f) + \alpha\mathcal{R}(u)\}, \tag{1.1}$$

where $\mathcal{D}(u, f)$ is the fidelity term, which is used to penalize the inconsistency between u and f, $\mathcal{R}(u)$ is the regularization term, which is used to filter out the noise from the observed noisy image as well as to preserve significant features such as edges and textures of the restored image, and $\alpha > 0$ is the regularization parameter, which compromises the fidelity term $\mathcal{D}(u, z)$ and the regularization term $\mathcal{R}(u)$.

In many situations, f can be represented by the so-called additive noise (AN) model

$$f = u + \eta, \tag{1.2}$$

where η is the zero-mean Gaussian white noise. Rudin, Osher, and Fatemi [4] proposed one of the most popular variational models in recovering u from the noisy image f. Their variational model, also known as the total variation (TV) model, is given by

$$\min_u \left\{\mathcal{J}^{\text{TV}}(u) = \mathcal{D}^{\text{ROF}}(u, f) + \alpha\mathcal{R}^{\text{TV}}(u)\right\} \tag{1.3}$$

involving the quadratic fidelity term resulting from the AN model (1.2)

$$\mathcal{D}^{\text{ROF}}(u, z) = \frac{1}{2}\int_\Omega (u - f)^2 \, d\Omega \tag{1.4}$$

and the TV regularization

$$\mathcal{R}^{\text{TV}}(u) = \int_\Omega |\nabla u| \, d\Omega. \tag{1.5}$$

Although $\mathcal{R}^{\text{TV}}$ is effective in preserving sharp edges while removing unwanted oscillations and noise, it has some undesired effects. In particular, it transforms smooth signals or signals that are not necessarily piecewise constant into piecewise constants. This phenomenon is known as staircase effect. From a practical point of view, staircase solutions of the TV model (1.3) fail to satisfy the evaluation of visual quality, and they can develop false edges that do not exist in the true image. The main challenge is thus to deal with the staircase effect while preserving image sharpness.

To overcome the drawback of the TV regularization, the variational model, which com-

bines the so-called Gaussian curvature (GC) regularization

$$\mathcal{R}^{\mathrm{GC}}(u) = \int_{\Omega} |\kappa_{\mathrm{G}}(u)| \, d\Omega \tag{1.6}$$

where

$$\kappa_{\mathrm{G}}(u) = \frac{u_{xx}u_{yy} - u_{xy}u_{yx}}{(1 + |\nabla u|^2)^2} \tag{1.7}$$

is the Gaussian curvature of image surface, named the GC model in this paper, was introduced by Brito-Loeza and Chen [1] as follows

$$\min_{u} \left\{ \mathcal{J}_{\alpha}^{\mathrm{GC}}(u) = \mathcal{D}^{\mathrm{ROF}}(u, f) + \alpha \mathcal{R}^{\mathrm{GC}}(u) \right\}. \tag{1.8}$$

By the calculus of variations, we obtain the associated Euler-Lagrange (EL) equation as follows

$$\alpha \nabla \cdot \left(\frac{4\,|u_{xx}u_{yy} - u_{xy}u_{yx}|}{\mathcal{N}^3} \nabla u \right) + \nabla \cdot \boldsymbol{B}_1 + \nabla \cdot \boldsymbol{B}_2 + u - f = 0 \tag{1.9}$$

with the boundary conditions

$$(-u_{xy}, u_{xx}) \cdot \boldsymbol{n} = 0, \;\; (u_{yy}, -u_{yx}) \cdot \boldsymbol{n} \quad = 0, \;\; -\boldsymbol{B}_1 \cdot \boldsymbol{n} = 0, \;\; -\boldsymbol{B}_2 \cdot \boldsymbol{n} \quad = 0, \tag{1.10}$$

where $\mathcal{N} = u_x^2 + u_y^2 + 1$, $\mathcal{S} = sign(u_{xx}u_{yy} - u_{xy}u_{yx})$, $\boldsymbol{B}_1 = \left(\left(\frac{\mathcal{S}u_{yy}}{\mathcal{N}^2} \right)_x, -\left(\frac{\mathcal{S}u_{xy}}{\mathcal{N}^2} \right)_x \right)$, $\boldsymbol{B}_2 = \left(\left(\frac{-\mathcal{S}u_{yx}}{\mathcal{N}^2} \right)_y, \left(\frac{\mathcal{S}u_{xx}}{\mathcal{N}^2} \right)_y \right)$, and $\boldsymbol{n}$ is the unit outward normal vector on the image boundary.

As can be seen, the EL equation (1.9) is highly nonlinear and of fourth-order with diffusion coefficients yielding anisotropic diffusion. Thus, the above equation is the apparent difficulty in developing an accurate numerical solver. In our initial tests, using the simple gradient descent method as numerical solver, this equation showed to be very stiff. To this end, we develop an efficient numerical approach via the idea of alternating direction method of multipliers (ADMM). Numerical tests using both synthetic and real images confirm that our proposed method is accurate and reliable.

The remainder of this paper is organized as follows. In Section 2, we propose our new numerical method for the GC model. The results of experiments from synthetic and real images are given in Section 3. In the final Section 4, the concluding remarks are drawn.

2 The proposed numerical method

In literature, ADMM has been successfully applied to minimize the energy functional involving TV regularization and quadratic or non-quadratic data fitting term. For example, the

authors in [3] used the split Bregman (SB) iteration to solve the TV model, while the augmented Lagrangian method (ALM) has been developed for the TV model in [5, 6]. Note that both SB and ALM employ ADMM to minimize their energy functionals and the convergence of ADMM is always guaranteed. In [5, 6], the authors have proven that the two methods are in fact equivalent. In this work, we consider the SB method for the GC model.

2.1 Discretization

Before starting our discussion on the proposed SB method, the finite difference discretizations of the first-oder and second-order derivative operators are required. They are respectively ∂_x^+, ∂_x^-, ∂_y^+, ∂_y^-, $\partial_x^-\partial_x^+$, $\partial_x^+\partial_x^-$, $\partial_y^-\partial_y^+$, and $\partial_y^+\partial_y^-$. Note that we apply periodic boundary conditions with these discretizations. It is often used for the image denoising problem since it is able to preserve jumps and image contrasts and allow fast numerical computations.

From now on, we assume $\Omega = [1, M] \times [1, N]$, where M and N are the positive integers representing the number of uniform intervals in the x and y coordinate directions. Let $\bar{\Omega} \subset \Omega$ be the discretized image domain as given by

$$\bar{\Omega} = \{(x, y) \in \Omega | (x, y) = (x_i, y_j),\ x_i = i,\ y_j = j, i = 1, 2, \ldots, M \text{ and } j = 1, 2, \ldots, N\}.$$

For simplicity, each grid point $(x_i, y_j) \in \bar{\Omega}$ is denoted by (i, j), where the coordinates x and y are oriented along columns and rows, respectively. All the variables are defined on these grid points. Let $\bar{\Omega} \to \mathcal{V} = \mathbb{R}^{M\times N}$ be the 2D grayscale image space with size MN. So, the first-order forward differences of the grayscale image u at grid point (i, j) along x and y directions are respectively

$$\partial_x^+(u)_{i,j} = \begin{cases} (u)_{i,j+1} - (u)_{i,j} & \text{if } 1 \le i \le M, 1 \le j < N \\ (u)_{i,1} - (u)_{i,j} & \text{if } 1 \le i \le M, j = N \end{cases} \tag{2.1}$$

and

$$\partial_y^+(u)_{i,j} = \begin{cases} (u)_{i+1,j} - (u)_{i,j} & \text{if } 1 \le i < M, 1 \le j \le N \\ (u)_{1,j} - (u)_{i,j} & \text{if } i = M, 1 \le j \le N \end{cases}. \tag{2.2}$$

The first-order backward differences are respectively

$$\partial_x^-(u)_{i,j} = \begin{cases} (u)_{i,j} - (u)_{i,j-1} & \text{if } 1 \le i \le M, 1 < j \le N \\ (u)_{i,j} - (u)_{i,N} & \text{if } 1 \le i \le M, j = 1 \end{cases} \tag{2.3}$$

and

$$\partial_y^-(u)_{i,j} = \begin{cases} (u)_{i,j} - (u)_{i-1,j} & \text{if } 1 < i \le M, 1 \le j \le N \\ (u)_{i,j} - (u)_{M,j} & \text{if } i = 1, 1 \le j \le N \end{cases}. \tag{2.4}$$

For $\boldsymbol{p} = (p_1, p_2) \in \mathcal{V}^2$ and $u \in \mathcal{V}$, the discrete version of the first-order adjoint divergence operator satisfies

$$\sum_{\substack{1\le i\le M \\ 1\le j\le M}} -(\nabla \cdot \boldsymbol{p}) \cdot u = \sum_{\substack{1\le i\le M \\ 1\le j\le M}} \boldsymbol{p} \cdot (\nabla u)$$

Therefore, according to the definition of discrete gradient operator

$$(\nabla u)_{i,j} = \begin{pmatrix} \partial_x^+(u)_{i,j} \\ \partial_y^+(u)_{i,j} \end{pmatrix},$$

the discrete divergence is given by

$$(\nabla \cdot \boldsymbol{p})_{i,j} = \partial_x^-(p_1)_{i,j} + \partial_y^-(p_2)_{i,j}. \tag{2.5}$$

The second-order derivative operators of u at grid point (i, j) are given by

$$\partial_x^+\partial_x^-(u)_{i,j} = \partial_x^-\partial_x^+(u)_{i,j} = \begin{cases} (u)_{i,N} - 2(u)_{i,j} + (u)_{i,j+1} & \text{if } 1 \le i \le M, j = 1 \\ (u)_{i,j-1} - 2(u)_{i,j} + (u)_{i,j+1} & \text{if } 1 \le i \le M, 1 < j < N \\ (u)_{i,j-1} - 2(u)_{i,j} + (u)_{i,1} & \text{if } 1 \le i \le M, j = N \end{cases} \tag{2.6}$$

$$\partial_y^+\partial_y^-(u)_{i,j} = \partial_y^-\partial_y^+(u)_{i,j} = \begin{cases} (u)_{M,j} - 2(u)_{i,j} + (u)_{i+1,j} & \text{if } i = 1, 1 \le j \le N \\ (u)_{i-1,j} - 2(u)_{i,j} + (u)_{i+1,j} & \text{if } 1 < i < M, 1 \le j \le N \\ (u)_{i-1,j} - 2(u)_{i,j} + (u)_{1,j} & \text{if } i = M, 1 \le j \le N \end{cases} \tag{2.7}$$

$$\partial_x^+\partial_y^-(u)_{i,j} = \begin{cases} (u)_{i,j+1} - (u)_{i,j} - (u)_{M,j+1} + (u)_{M,j} & \text{if } i = 1, 1 \le j < N \\ (u)_{i,1} - (u)_{i,j} - (u)_{M,1} + (u)_{M,j} & \text{if } i = 1, j = N \\ (u)_{i,j+1} - (u)_{i,j} - (u)_{i-1,j+1} + (u)_{i-1,j} & \text{if } 1 < i \le M, 1 \le j < N \\ (u)_{i,1} - (u)_{i,j} - (u)_{i-1,1} + (u)_{i-1,j} & \text{if } 1 < i \le M, j = N \end{cases} \tag{2.8}$$

and

$$\partial_y^+\partial_x^-(u)_{i,j} = \begin{cases} (u)_{i+1,j} - (u)_{i,j} - (u)_{i+1,N} + (u)_{i,N} & \text{if } 1 \le i < M, j = 1 \\ (u)_{1,j} - (u)_{i,j} - (u)_{1,N} + (u)_{i,N} & \text{if } i = M, j = 1 \\ (u)_{i+1,j} - (u)_{i,j} - (u)_{i+1,j-1} + (u)_{i,j-1} & \text{if } 1 \le i < M, 1 < j \le N \\ (u)_{1,j} - (u)_{i,j} - (u)_{1,j-1} + (u)_{i,j-1} & \text{if } i = M, 1 < j \le N \end{cases}. \tag{2.9}$$

Based on (2.6) and (2.7), the discrete Laplace operator is given by

$$(\triangle u)_{i,j} = \partial_x^-\partial_x^+(u)_{i,j} + \partial_y^-\partial_y^+(u)_{i,j}. \tag{2.10}$$

2.2 The proposed numerical algorithm

To solve the GC model (1.8) with the SB method, we introduce the new variable $\boldsymbol{p} = (p_1, p_2) \in \mathcal{V}^2$ and reformulate the variational problem in (1.8) as the following constrained optimization problem

$$\min_{u,\boldsymbol{p}} \left\{ \mathcal{D}^{\text{ROF}}(u, f) + \alpha \int_\Omega \left| \frac{p_{1_x} p_{2_y} - p_{1_y} p_{2_x}}{(|\boldsymbol{p}|^2 + 1)^2} \right| d\Omega \right\}, \quad \text{s.t. } \boldsymbol{p} = \nabla u. \tag{2.11}$$

The SB method for the above constrained optimization problem is obtained by minimizing the following functional

$$\mathcal{E}(u, \boldsymbol{p}; \boldsymbol{b}) = \mathcal{D}^{\text{ROF}}(u, f) + \alpha \int_\Omega \left| \frac{p_{1_x} p_{2_y} - p_{1_y} p_{2_x}}{(|\boldsymbol{p}|^2 + 1)^2} \right| d\Omega + \frac{\theta}{2} \int_\Omega (\boldsymbol{p} - \nabla u - \boldsymbol{b})^2 d\Omega, \tag{2.12}$$

where $\boldsymbol{b} = (b_1, b_2) \in \mathcal{V}^2$ is the Bregman parameter. The functional $\mathcal{E}$ can be minimized using alternative optimization technique. That is, we solve two subproblems for u and $\boldsymbol{p}$. This process is repeated until the following stopping criteria based on the relative error of the

solution is satisfied:

$$\frac{\left\|u^{[m+1]}-u^{[m]}\right\|_{l^2}^2}{\left\|u^{[m]}\right\|_{l^2}^2}<\epsilon \tag{2.13}$$

for predefined small $\epsilon>0$. The two subproblems of (2.12) are

$$\begin{aligned} &u\text{-subproblem}: \quad \min_u\left\{\mathcal{D}^{\text{ROF}}(u,f)+\frac{\theta}{2}\int_\Omega(\boldsymbol{p}-\nabla u-\boldsymbol{b})^2d\Omega\right\},\\ &\boldsymbol{p}\text{-subproblem}: \quad \min_{\boldsymbol{p}}\left\{\alpha\int_\Omega\left|\frac{p_{1_x}p_{2_y}-p_{1_y}p_{2_x}}{(|\boldsymbol{p}|^2+1)^2}\right|d\Omega+\frac{\theta}{2}\int_\Omega(\boldsymbol{p}-\nabla u-\boldsymbol{b})^2d\Omega\right\}. \end{aligned}$$

Note that the u-subproblem can be efficiently solved by its closed-form solution, while the solution of the $\boldsymbol{p}$-subproblem is determined by the associated Euler-Lagrange equation.

u-subproblem. Fixing variables ($\boldsymbol{p}$; $\boldsymbol{b}$) yields the Euler-Lagrange equation associated with u as follows

$$u-\theta\triangle u=G, \tag{2.14}$$

where $G=f-\theta\nabla\cdot(\boldsymbol{p}-\boldsymbol{b})$. To solve the linear partial differential equation in (2.14), we apply the discrete divergence operator in (2.5) and the discrete Laplace operator in (2.10) as follows

$$(u)_{i,j}-\theta\left(\partial_x^-\partial_x^+(u)_{i,j}+\partial_y^-\partial_y^+(u)_{i,j}\right)=(G)_{i,j}, \tag{2.15}$$

where $(G)_{i,j}=(f)_{i,j}-\theta\left(\partial_x^-((p_1)_{i,j}-(b_1)_{i,j})+\partial_y^-((p_2)_{i,j}-(b_2)_{i,j})\right)$. As periodic boundary conditions have been imposed on the discrete derivatives in Section 2.1, the discrete Fourier transform can be directly applied to the both sides of (2.15)

$$\mathcal{F}\left\{(u)_{i,j}-\theta\left(\partial_x^-\partial_x^+(u)_{i,j}+\partial_y^-\partial_y^+(u)_{i,j}\right)\right\}=\mathcal{F}\left\{(G)_{i,j}\right\}, \tag{2.16}$$

where $\mathcal{F}$ denotes the discrete Fourier transform. For discrete frequencies r and s, we have

$$\underbrace{\left(1-2\theta\left(\cos\frac{2\pi s}{N}+\cos\frac{2\pi r}{N}-2\right)\right)}_{\zeta}\mathcal{F}\left\{(u)_{i,j}\right\}=\mathcal{F}\left\{(G)_{i,j}\right\}, \tag{2.17}$$

where $i\in[1,M]$ and $j\in[1,N]$ are the indexes in discrete time domain, $r\in[0,M)$ and $s\in[0,N)$ are the frequencies in the discrete frequency domain. (2.17) provides us with a

closed-form solution of u as

$$(u)_{i,j} = \text{Real}\left(\mathcal{F}^{-1}\left\{\frac{\mathcal{F}\left\{(G)_{i,j}\right\}}{\zeta}\right\}\right), \tag{2.18}$$

where $\mathcal{F}^{-1}$ denotes the discrete inverse Fourier transform. 'Real' is the real part of a complex number. '—' stands for pointwise division of matrices.

***p*-subproblem.** Fixing variables (u; $\boldsymbol{b}$) yields the Euler-Lagrange equation associated with $\boldsymbol{p}$ as follows

$$\begin{cases} -\alpha\nabla\cdot\boldsymbol{w}_1 + \alpha\beta p_1 + \theta(p_1 - u_x - b_1) = 0 \\ -\alpha\nabla\cdot\boldsymbol{w}_2 + \alpha\beta p_2 + \theta(p_2 - u_y - b_2) = 0 \end{cases}, \tag{2.19}$$

where $\boldsymbol{w}_1 = \left(\frac{\bar{\mathcal{S}}(p_2)_y}{\beta_2^2}, -\frac{\bar{\mathcal{S}}(p_2)_x}{\beta_2^2}\right)$, $\boldsymbol{w}_2 = \left(-\frac{\bar{\mathcal{S}}(p_1)_y}{\beta_2^2}, \frac{\bar{\mathcal{S}}(p_1)_x}{\beta_2^2}\right)$, $\beta_1 = (p_1)_x(p_2)_y - (p_1)_y(p_2)_x$, $\beta_2 = p_1^2 + p_2^2 + 1$, $\beta = \frac{4|\beta_1|}{\beta_3^2}$, $\bar{\mathcal{S}} = \frac{\beta_1}{|\beta_1|}$. To solve the nonlinear partial differential equation in (2.19), we apply a coupled outer-inner iteration method as follows. In the outer iteration, we start from an initial solution $\boldsymbol{p}^{[0]}$ and compute a sequence of approximate solutions $\boldsymbol{p}^{[1]}, \boldsymbol{p}^{[2]}, \cdots, \boldsymbol{p}^{[\nu]}, \boldsymbol{p}^{[\nu+1]}, \cdots$ by solving the discrete system of linear equations

$$\begin{cases} (p_1^{[\nu+1]})_{i,j} = \dfrac{\alpha\nabla\cdot\boldsymbol{w}_1^{[\nu]} + \theta(u_x + b_1)}{\alpha\beta + \theta} \\ (p_2^{[\nu+1]})_{i,j} = \dfrac{\alpha\nabla\cdot\boldsymbol{w}_2^{[\nu]} + \theta(u_y + b_2)}{\alpha\beta + \theta} \end{cases}, \tag{2.20}$$

until the stopping rule $\left\|\boldsymbol{p}^{[\nu+1]} - \boldsymbol{p}^{[\nu]}\right\|^2 / \left\|\boldsymbol{p}^{[\nu]}\right\|^2 < \bar{\epsilon}_1$ or $\nu \geq \bar{\epsilon}_2$ is met for a given threshold $\bar{\epsilon}_1 > 0$ and the maximum iteration $\bar{\epsilon}_2 > 0$.

Finally, we update the Bregman iterative parameter by $\boldsymbol{b} \leftarrow \boldsymbol{b} + \nabla u - \boldsymbol{p}$, which can be found in Algorithm 2.1.

Algorithm 2.1. Our proposed numerical algorithm for GC model

u	the restored image
f	the observed noisy image
α	the regularization parameter
$\boldsymbol{p} = (p_1, p_2)$	the vector valued function related to the gradient of u
$\boldsymbol{b} = (b_1, b_2)$	Bregman iterative parameter
θ	penalty parameter
ϵ_1	the given threshold(typically, $\epsilon_1 = 10^{-4}$)
ϵ_2	the maximum number of iterations (typically, $\epsilon_2 = 1000$)

$$[u] \leftarrow \text{SBGC}\,(u, f, \alpha, \boldsymbol{p}, \boldsymbol{b}, \theta, \epsilon_1, \epsilon_2)$$

- Let $err = 1, k = 0, \boldsymbol{p} = 0, \boldsymbol{b} = 0, \alpha > 0, \theta > 0$ and $u = f$
- While $err > \varepsilon_1$ and $k < \varepsilon_2$
  - Set $u^{\text{old}} = u$
  - Compute u by (2.18)
  - Compute $\boldsymbol{p}$ by (2.20)
  - Updating the Bregman iterative parameter $\boldsymbol{b}$ by $\boldsymbol{b} \leftarrow \boldsymbol{b} + \nabla u - \boldsymbol{p}$
  - Compute $err = \frac{\|u - u^{\text{old}}\|_{l_2}}{\|u\|_{l_2}}$
  - Compute $k = k + 1$
- end

3 Numerical experiments

In this section, we present some numerical experiments to

(i) compare the restoration quality (the accuracy in removing noise) obtained from the proposed method (Algorithm 2.1) and the standard method by [4] in recovering the sharp edge and removing the staircase effect from a noisy image;

(ii) illustrate the performance of our proposed numerical method in removing noise from a real noisy image.

Figure 1 shows the original test images. The first two columns show the two synthetic images of 256×256 pixels, while the last column shows the real noisy image of 2048×2048 pixels. We note first that the noise formation model in (1.2) was used for creating the noisy versions of the synthetic images with the additive Gaussian noise of variance 0.03. We also

note that we used peak signal to noise ratio (PSNR) between the original and denoised images to measure the quality of image restoration; a higher PSNR value indicates the higher quality of the denoised image (better). PSNR is defined by

$$\text{PSNR} = 10\log\left(\frac{255^2}{\text{MSE}}\right), \tag{3.1}$$

where

$$\text{MSE} = \frac{1}{NM}\sum_{i=1}^{M}\sum_{j=1}^{N}\big((u^*)_{i,j} - (u)_{i,j}\big).$$

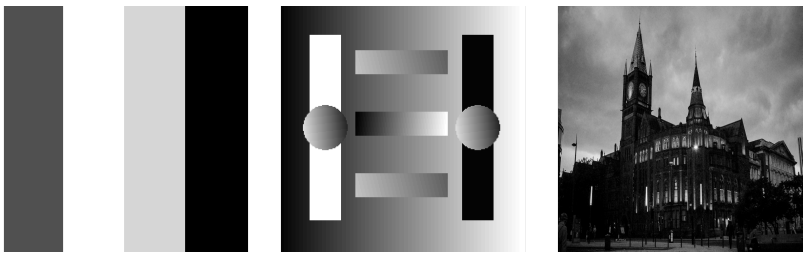
Here u^* and u are the original image and the denoised image, respectively. Next, we note that the stopping criteria in (2.13) with $\epsilon = 10^{-4}$ was used for all numerical tests. We also note that the model parameters in (2.12), α and θ, for each test were manually selected. Finally we note that all experiments were performed using Matlab R2016b on MAC OS X platform with an Intel Core i5 at 2.3GHz and 8GB of RAM.

In Figure 2, we test the edge- and smoothness preservation abilities of the proposed and standard methods on a piecewise constant image (top row) and a piecewise smooth image (bottom row), respectively. Note that the one-dimensional profiles (or signals) of the images in Figure 2 (selected from the 128th row with 256 columns) are shown in the top and bottom rows of Figures 3.

As expected, the standard method is less efficient than the proposed method in terms of PSNR and visual inspection for recovering the sharp edges of the denoised images. Moreover, it is clear from Figure 3 (top row) that the proposed method preserves the corners perfectly.

From Figure 3 (bottom row), one can easily find in the middle column that the denoised signal by the standard method is composed of jagged appearance which is staircase effect. In addition, the denoised results shown from the right column by the proposed method does not produce unfavorable staircase artifact. In terms of PSNR and visual inspection, the proposed method model is more efficient than the standard method for the smoothness-preservation ability.

Finally, we test the proposed and standard methods on a real noisy image. As can be seen from Figures 4 and 5, the proposed method delivers more visually pleasing restoration results aspect of visual perception and also provides the best tradeoff between noise suppression and image enhancement for detail preservation.



(a) (b) (c)

Figure 1: The original test images. Synthetic images of 256×256 pixels (the first two columns) and real noisy images of 2048×2048 pixels (the last column).

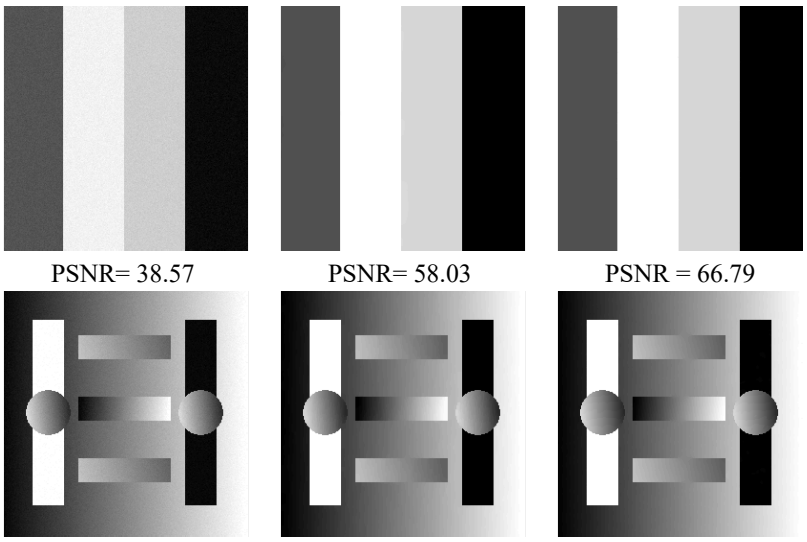
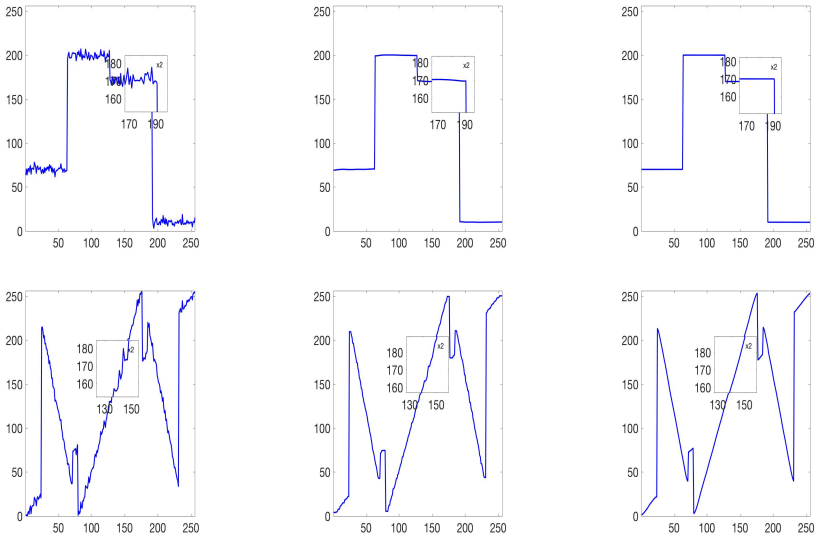


Figure 2: Restoration results on the synthetic images in Figure 1 (the first two columns). From top to bottom: edge-preservation ability test (top row) and smoothness-preservation ability test (bottom rom). From left to right: noisy images (left column); denoised images by standard method (middle column); and denoised images by the proposed method (right column).

4 Conclusion

The main focus of this work is to present a numerical method for GC model in removing noise from digital images. In order to solve the associated variational problem, we have proposed the SB method. As expected, numerical tests confirmed that the proposed method delivers more accurate and reliable restoration results than the standard method.



Noisy image Standard method Proposed method

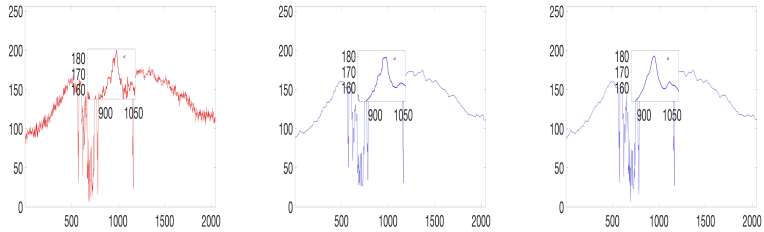
Figure 3: Corresponding signals of the 128th row from the images in Figure 2. Noisy signals (left column); denoised signals by the standard method (middle column); denoised signals by the proposed method (right column).

Noisy image Standard method Proposed method

Figure 4: Restoration results on the real noisy images in Figure 1 (the last column). Real noisy image (left column); denoised image by the standard method (middle column); and denoised image by the proposed method (right column).

Acknowledgements

The authors would like to thank all reviewers. The first author's work was partially supported by Centre of Excellence in Mathematics.



Noisy image Standard method Proposed method

Figure 5: Corresponding signals of the 512th row from the images in Figure 4. Real noisy signal (left column); denoised signal by the standard method (middle column); denoised signal by the proposed method (right column).

References

[1] C. Brito-Loeza, K. Chen, and V. Uc-Cetina, *Image denoising using the Gaussian curvature of the image surface*, Numer. Meth. Part. Differ. Equ. **32** (2016), no. 3, 1066–1089.

[2] S. Geman and D. Geman, *Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images*, IEEE Trans. Pattern Anal. Mach Intel. **6** (1984), no. 6, 721–741.

[3] T. Goldstein and S. Osher., *The split Bregman method for L1-regularized problems.*, SIAM Journal on Imaging Sciences **2** (2009), no. 2, 323–343.

[4] L. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Physica D **60** (1992), 259–268.

[5] X.-C. Tai. and C. Wu., *Augmented Lagrangian method, dual methods and split Bregman iteration for ROF model*, Scale Space and Variational Methods in Computer Vision, LNCS **5567** (2009), no. 556, 502–513.

[6] 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 Sci. **3** (2010), no. 3, 300–339.

An improved variational model for image restoration

Pornpimon Sroisangwan[1,*] and **Noppadol Chumchob**[1,2]

[1]Department of Mathematics, Faculty of Science, Silpakorn University, Nakorn Pathom 73000, Thailand;
p.sroisangwan@gmail.com;chumchob_n@silpakorn.edu
[2]Centre of Excellence in Mathematics, CHE, Si Ayutthaya Rd., Bangkok 10350, Thailand; chumchob_n@silpakorn.edu
[*]Correspondence: p.sroisangwan@gmail.com

Abstract: Image restoration is the most fundamental task in image processing. The goal is to remove or reduce the noise from a given corrupted digital image for improving the quality and ability of human or machine vision identifications. Total variation (TV) model, which is a classic model in image restoration, is well-known for reducing noise and recovering sharp edges from a observed noisy image. However, it suffers form the undesired artifacts, such as the staircase effect. To overcome the drawback of the TV model, this work proposes a hybrid model combining the TV regularization and the high-order regularization. As a result, the associated minimization problem is not appropriate to directly solve by some classical algorithms. We therefore develop a new alternating minimization algorithm. Numerical experiments on both synthetic and real medical images show that the quality of restored images by the proposed method is better than those by the TV model.

Keywords: Alternating minimization algorithm; total variation; regularization; image restoration

AMS Math Classification (2010) : 62H35, 65N22, 65N55, 74G65, 74G75

1 Introduction

Let $z : \Omega \subset \mathbb{R}^2 \to V \subset \mathbb{R}$ be an observed noisy image, where Ω is a rectangle of $\mathbb{R}^2$. The goal of image denoising problems is to restore (or recover) the original image $u : \Omega \subset \mathbb{R}^2 \to V \subset \mathbb{R}$ from the noisy image z. According to the maximum likelihood principle [1], most image denoising problems involve solving the following variational problem:

$$\min_u \{\mathcal{J}_\alpha(u) = \mathcal{D}(u, z) + \alpha\mathcal{R}(u)\}, \tag{1}$$

where $\mathcal{D}(u, z)$ is the fidelity term, which is used to penalize the inconsistency between u and z, $\mathcal{R}(u)$ is the regularization term, which is used to filter out the noise from the observed noisy image as well as to preserve significant features such as edges and textures of the restored image, and $\alpha > 0$ is the regularization parameter, which compromises the fidelity term $\mathcal{D}(u, z)$ and the regularization term $\mathcal{R}(u)$.

In many situations, z can be represented by the so-called additive noise (AN) model

$$z = u + \eta, \tag{2}$$

where η is the zero-mean Gaussian white noise. Rudin, Osher, and Fatemi [2] proposed one of the most popular variational models in recovering u from the noisy image z. Their variational model, also known as the total variation (TV) model, is given by

$$\min_u \left\{\mathcal{J}^{\mathrm{TV}}(u) = \mathcal{D}^{\mathrm{ROF}}(u, z) + \alpha\mathcal{R}^{\mathrm{TV}}(u)\right\} \tag{3}$$

involving the quadratic fidelity term resulting from the AN model (2)

$$\mathcal{D}^{\mathrm{ROF}}(u,z)=\frac{1}{2}\int_{\Omega}(u-z)^2\,d\Omega \tag{4}$$

and the TV regularization

$$\mathcal{R}^{\mathrm{TV}}(u)=\int_{\Omega}|\nabla u|\,d\Omega. \tag{5}$$

Although the TV regularization is effective in preserving sharp edges while removing unwanted oscillations and noise, it has some undesired effects. In particular, it transforms smooth signals or signals that are not necessarily piecewise constant into piecewise constants. This phenomenon is known as staircase effect. From a practical point of view, staircase solutions of the TV model (3) fail to satisfy the evaluation of visual quality, and they can develop false edges that do not exist in the true image. To overcome this kind of deficiency, a popular method is to replace the TV regularization by a high-order regularization consisting of high-order derivative terms. Particular, the curvature based regularization schemes are widely studied for overcoming the staircase effects while preserving the edges will in the restored image [3, 4].

In this paper, inspired by the successful applications of the TV regularization and the curvature based regularization, we propose a new hybrid model, which we call TVTC. This model combines the TV regularization and the so-called total curvature (TC) based regularization to be a new regularization term instead of the TV regularization term in the classic TV model (3). Under the framework of the popular split Bregman (SB) method [5], we propose an efficient iterative algorithm to solve this model. For reconstructing image from noisy image, our experiments will show that our new method can get restored images with edges and simultaneous smooth regions preserved. Moreover, compared with the classic TV model, our new method can overcome the staircase effect effectively, get better numerical result, and get higher visual quality.

The outline of the rest of the paper is as follows. In Section 2 we introduce the TVTC model and proposed an efficient algorithm based on SB method. Numerical results are given in Section 3. Finally, we conclude this paper in Section 4.

2 Proposed model and its numerical algorithm

2.1 TVTC model

Let $\kappa_1(u)=\kappa_1(u(x,y),x,y)$ and $\kappa_2(u)=\kappa_2(u(x,y),x,y)$ be the principal curvatures of the image surface induced by a given image u. According to theory of surfaces, the mean curvature

$$\kappa_{\mathrm{M}}(u)=\nabla\cdot\left(\frac{\nabla u}{\sqrt{1+|\nabla u|^2}}\right)=\frac{(1+u_x^2)u_{yy}-2u_xu_yu_{xy}+(1+u_y^2)u_{xx}}{(1+|\nabla u|^2)^{3/2}} \tag{6}$$

and the Gaussian curvature

$$\kappa_{\mathrm{G}}(u)=\frac{u_{xx}u_{yy}-u_{xy}u_{yx}}{(1+|\nabla u|^2)^2} \tag{7}$$

are related to the principal curvatures $\kappa_1(u)$ and $\kappa_1(u)$ by

$$\kappa_{\mathrm{M}}(u)=\kappa_1(u)+\kappa_2(u)\text{ and }\kappa_{\mathrm{G}}(u)=\kappa_1(u)\kappa_2(u).$$

Therefore, the sum of the squares of the principal curvatures of the image surface are related to the mean and Gaussian curvatures as follows:

$$\begin{aligned}\kappa_1^2(u)+\kappa_2^2(u)&=\left(\kappa_1(u)+\kappa_2(u)\right)^2-2\kappa_1(u)\kappa_2(u)\\&=\kappa_{\mathrm{M}}^2(u)-2\kappa_{\mathrm{G}}(u)\\&=\left(\frac{(1+u_x^2)u_{yy}-2u_xu_yu_{xy}+(1+u_y^2)u_{xx}}{(1+|\nabla u|^2)^{3/2}}\right)^2-2\frac{u_{xx}u_{yy}-u_{xy}u_{yx}}{(1+|\nabla u|^2)^2}.\end{aligned} \tag{8}$$

We note that the authors in [3] use

$$\kappa_{\mathrm{M}}(u) = \frac{\kappa_1(u) + \kappa_2(u)}{2} = \frac{(1+u_x^2)u_{yy} - 2u_x u_y u_{xy} + (1+u_y^2)u_{xx}}{2(1+|\nabla u|^2)^{3/2}},$$

justifying the name 'mean' curvature. We also note that $\kappa_{\mathrm{M}}(u_0) = 0$ and $\kappa_{\mathrm{G}}(u_0) = 0$ for piecewise constant or piecewise affine images u_0. This motivates us to propose the following functional

$$\mathcal{R}^{\mathrm{TC}}(u) = \frac{1}{2}\int_\Omega \left(\kappa_1^2(u) + \kappa_2^2(u)\right) d\Omega = \frac{1}{2}\int_\Omega \left(\kappa_{\mathrm{M}}^2(u) - 2\kappa_{\mathrm{G}}(u)\right) d\Omega \tag{9}$$

to be as our high-order regularization. As can be seen, the kernel of $\mathcal{R}^{\mathrm{TC}}$ denoted by

$$\ker\left(\mathcal{R}^{\mathrm{TC}}\right) = \left\{u : \Omega \to \mathbb{R} | \mathcal{R}^{\mathrm{TC}}(u) = 0\right\}$$

is given by

$$\ker\left(\mathcal{R}^{\mathrm{TC}}(u)\right) = \left\{u(\mathbf{x}) = \mathbf{A}\mathbf{x} + b \text{ a.e. in } \Omega \text{ for some } \mathbf{A} \in \mathbb{R}^{1\times 2},\ \mathbf{x} = (x,y)^\top \in \mathbb{R}^2,\ b \in \mathbb{R}\right\}.$$

It follows that $\mathcal{R}^{\mathrm{TC}}$ has the attractive property in dealing with different image characteristics such as piecewise constant and piecewise affine images. We note that $\mathcal{R}^{\mathrm{TC}}$ is also known as the total curvature regularization or the bending energy used to measure surface fairing and related to the strain energy of a thin elastic plate, which has lots of applications in computer graphics [6]. To the best of our knowledge, this is the first work that uses the total curvature regularization $\mathcal{R}^{\mathrm{TC}}$ for noise removal.

In order to take the advantage of both regularization terms, the new idea arises to incorporate these two regularization terms with the fidelity $\mathcal{D}^{\mathrm{ROF}}$. As a result, our proposed model comes out as given by

$$\min_u \left\{\mathcal{J}^{\mathrm{TVTC}}(u) = \mathcal{D}^{\mathrm{ROF}}(u,z) + \mathcal{R}^{\mathrm{TVTC}}(u)\right\} \tag{10}$$

with the TVTC regularization $\mathcal{R}^{\mathrm{TVTC}}(u) = \alpha_1\mathcal{R}^{\mathrm{TV}}(u) + \alpha_2\mathcal{R}^{\mathrm{TC}}(u)$ where $\alpha_1 > 0$ and $\alpha_2 > 0$ are the weighting parameters that balance the TV and TC regularization terms.

By the calculus of variations, we obtain the Euler-Lagrange (EL) equation associated with u as follows

$$\begin{aligned} &u - z - \alpha_1\nabla\cdot\left(\frac{\nabla u}{|\nabla u|}\right) + \alpha_2\nabla\cdot\left(\frac{1}{\sqrt{1+|\nabla u|^2}}\nabla\kappa_{\mathrm{M}}(u) - \frac{\nabla u\cdot\nabla\kappa_{\mathrm{M}}(u)}{\left(\sqrt{1+|\nabla u|^2}\right)^3}\nabla u\right) \\ &+ \alpha_2\nabla\cdot\left(\frac{4\left|u_{2xx}u_{2yy} - u_{2xy}u_{2yx}\right|}{\mathcal{N}^3}\nabla u\right) + \nabla\cdot\boldsymbol{B}_1 + \nabla\cdot\boldsymbol{B}_2 = 0 \end{aligned} \tag{11}$$

with the boundary conditions

$$\begin{aligned} &\nabla u\cdot\boldsymbol{n} = 0, \kappa_M(u)\cdot\boldsymbol{n} = 0,\ \frac{\nabla u\cdot\nabla\kappa_M(u)}{\left(\sqrt{1+|\nabla u|^2}\right)^3}\nabla u\cdot\boldsymbol{n} = 0,\ \frac{1}{\sqrt{1+|\nabla u|^2}}\nabla\kappa_M(u)\cdot\boldsymbol{n} = 0, \\ &(-u_{xy}, u_{xx})\cdot\boldsymbol{n} = 0,\ (u_{yy}, -u_{yx})\cdot\boldsymbol{n} = 0,\ -\boldsymbol{B}_1\cdot\boldsymbol{n} = 0,\ -\boldsymbol{B}_2\cdot\boldsymbol{n} = 0, \end{aligned}$$

where

$$\mathcal{N} = {u_x}^2 + {u_y}^2 + 1, \boldsymbol{B}_1 = \left(\left(\frac{u_{yy}}{\mathcal{N}^2}\right)_x, \left(\frac{-u_{xy}}{\mathcal{N}^2}\right)_x\right), \boldsymbol{B}_2 = \left(\left(\frac{-u_{yx}}{\mathcal{N}^2}\right)_y, \left(\frac{u_{xx}}{\mathcal{N}^2}\right)_y\right),$$

and $\boldsymbol{n}$ is the unit outward normal vector on the image boundary.

As can be seen, the Euler-Lagrange equation (11) is a complicated partial differential equation that is difficult to solve computationally. So, standard numerical algorithms such as time marching methods are not appropriate. To this end, we shall reformulate the proposed model (10) in SB framework in the next subsection.

2.2 Numerical implementation

Before starting our discussion on the reformulation of the proposed TVTC model in SB framework, the finite difference discretizations of the first-oder and second-order derivative operators are required. They are respectively ∂_x^+, ∂_x^-, ∂_y^+, ∂_y^-, $\partial_x^-\partial_x^+$, $\partial_x^+\partial_x^-$, $\partial_y^-\partial_y^+$, and $\partial_y^+\partial_y^-$. Note that we apply periodic boundary conditions with these discretizations. It is often used for the image denoising problem since it is able to preserve jumps and image contrasts and allow fast numerical computations.

From now on, we assume $\Omega = [1, M] \times [1, N]$, where M and N are the positive integers representing the number of uniform intervals in the x and y coordinate directions. Let $\bar{\Omega} \subset \Omega$ be the discretized image domain as given by

$$\bar{\Omega} = \{(x, y) \in \Omega | (x, y) = (x_i, y_j),\ x_i = i,\ y_j = j, i = 1, 2, \ldots, M \text{ and } j = 1, 2, \ldots, N\}.$$

For simplicity, each grid point $(x_i, y_j) \in \bar{\Omega}$ is denoted by (i, j), where the coordinates x and y are oriented along columns and rows, respectively. All the variables are defined on these grid points. Let $\bar{\Omega} \to \mathcal{V} = \mathbb{R}^{M\times N}$ be the 2D grayscale image space with size MN. So, the first-order forward differences of the grayscale image u at grid point (i, j) along x and y directions are respectively

$$\partial_x^+(u)_{i,j} = \begin{cases} (u)_{i,j+1} - (u)_{i,j} & \text{if } 1 \le i \le M, 1 \le j < N \\ (u)_{i,1} - (u)_{i,j} & \text{if } 1 \le i \le M, j = N \end{cases} \tag{12}$$

and

$$\partial_y^+(u)_{i,j} = \begin{cases} (u)_{i+1,j} - (u)_{i,j} & \text{if } 1 \le i < M, 1 \le j \le N \\ (u)_{1,j} - (u)_{i,j} & \text{if } i = M, 1 \le j \le N \end{cases}. \tag{13}$$

The first-order backward differences are respectively

$$\partial_x^-(u)_{i,j} = \begin{cases} (u)_{i,j} - (u)_{i,j-1} & \text{if } 1 \le i \le M, 1 < j \le N \\ (u)_{i,j} - (u)_{i,N} & \text{if } 1 \le i \le M, j = 1 \end{cases} \tag{14}$$

and

$$\partial_y^-(u)_{i,j} = \begin{cases} (u)_{i,j} - (u)_{i-1,j} & \text{if } 1 < i \le M, 1 \le j \le N \\ (u)_{i,j} - (u)_{M,j} & \text{if } i = 1, 1 \le j \le N \end{cases}. \tag{15}$$

For $\boldsymbol{p} = (p_1, p_2) \in \mathcal{V}^2$ and $u \in \mathcal{V}$, the discrete version of the first-order adjoint divergence operator satisfies

$$\sum_{\substack{1\le i\le M \\ 1\le j\le M}} -(\nabla \cdot \boldsymbol{p}) \cdot u = \sum_{\substack{1\le i\le M \\ 1\le j\le M}} \boldsymbol{p} \cdot (\nabla u)$$

Therefore, according to the definition of discrete gradient operator

$$(\nabla u)_{i,j} = \begin{pmatrix} \partial_x^+(u)_{i,j} \\ \partial_y^+(u)_{i,j} \end{pmatrix},$$

the discrete divergence is given by

$$(\nabla \cdot \boldsymbol{p})_{i,j} = \partial_x^-(p_1)_{i,j} + \partial_y^-(p_2)_{i,j}. \tag{16}$$

The second-order derivative operators of u at grid point (i, j) are given by

$$\partial_x^+\partial_x^-(u)_{i,j} = \partial_x^-\partial_x^+(u)_{i,j} = \begin{cases} (u)_{i,N} - 2(u)_{i,j} + (u)_{i,j+1} & \text{if } 1 \le i \le M, j = 1 \\ (u)_{i,j-1} - 2(u)_{i,j} + (u)_{i,j+1} & \text{if } 1 \le i \le M, 1 < j < N \\ (u)_{i,j-1} - 2(u)_{i,j} + (u)_{i,1} & \text{if } 1 \le i \le M, j = N \end{cases}, \tag{17}$$

$$\partial_y^+\partial_y^-(u)_{i,j} = \partial_y^-\partial_y^+(u)_{i,j} = \begin{cases} (u)_{M,j} - 2(u)_{i,j} + (u)_{i+1,j} & \text{if } i = 1, 1 \le j \le N \\ (u)_{i-1,j} - 2(u)_{i,j} + (u)_{i+1,j} & \text{if } 1 < i < M, 1 \le j \le N \\ (u)_{i-1,j} - 2(u)_{i,j} + (u)_{1,j} & \text{if } i = M, 1 \le j \le N \end{cases}, \tag{18}$$

$$\partial_x^+\partial_y^-(u)_{i,j} = \begin{cases} (u)_{i,j+1} - (u)_{i,j} - (u)_{M,j+1} + (u)_{M,j} & \text{if } i = 1, 1 \le j < N \\ (u)_{i,1} - (u)_{i,j} - (u)_{M,1} + (u)_{M,j} & \text{if } i = 1, j = N \\ (u)_{i,j+1} - (u)_{i,j} - (u)_{i-1,j+1} + (u)_{i-1,j} & \text{if } 1 < i \le M, 1 \le j < N \\ (u)_{i,1} - (u)_{i,j} - (u)_{i-1,1} + (u)_{i-1,j} & \text{if } 1 < i \le M, j = N \end{cases}, \tag{19}$$

$$\partial_y^+\partial_x^-(u)_{i,j} = \begin{cases} (u)_{i+1,j} - (u)_{i,j} - (u)_{i+1,N} + (u)_{i,N} & \text{if } 1 \le i < M, j = 1 \\ (u)_{1,j} - (u)_{i,j} - (u)_{1,N} + (u)_{i,N} & \text{if } i = M, j = 1 \\ (u)_{i+1,j} - (u)_{i,j} - (u)_{i+1,j-1} + (u)_{i,j-1} & \text{if } 1 \le i < M, 1 < j \le N \\ (u)_{1,j} - (u)_{i,j} - (u)_{1,j-1} + (u)_{i,j-1} & \text{if } i = M, 1 < j \le N \end{cases}. \tag{20}$$

Based on (17) and (18), the discrete Laplace operator is given by

$$(\triangle u)_{i,j} = \partial_x^-\partial_x^+(u)_{i,j} + \partial_y^-\partial_y^+(u)_{i,j}. \tag{21}$$

To reformulate the TVTC model (10) in SB framework, we rewrite the TVTC model in (10) by splitting variable method as follows:

$$\begin{aligned} \min_{u,\boldsymbol{p},\boldsymbol{m},\boldsymbol{n},q} \left\{ \mathcal{D}^{\text{ROF}}(u,z) + \alpha_1 \int_\Omega |\boldsymbol{p}|\, d\Omega + \frac{\alpha_2}{2}\int_\Omega q^2 d\Omega - \alpha_2 \int_\Omega \left| \frac{p_{1_x}p_{2_y} - p_{1_y}p_{2_x}}{(|\boldsymbol{p}|^2+1)^2} \right| d\Omega \right\}, \\ \text{s.t. } \boldsymbol{p} = \nabla u, \boldsymbol{m} = \frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2+1}}, \boldsymbol{n} = \boldsymbol{m} \quad \text{and} \quad q = \nabla\cdot\boldsymbol{n}. \end{aligned} \tag{22}$$

According to SB method, the solution of (22) can be obtained by minimizing the following functional

$$\begin{aligned} \mathcal{E}(u,\boldsymbol{p},\boldsymbol{m},\boldsymbol{n},q;\boldsymbol{b_1},\boldsymbol{b_2},\boldsymbol{b_3},b_4) =& \mathcal{D}^{\text{ROF}}(u,z) + \alpha_1\int_\Omega |\boldsymbol{p}|\, d\Omega + \frac{\alpha_2}{2}\int_\Omega q^2 d\Omega - \alpha_2 \int_\Omega \left| \frac{p_{1_x}p_{2_y} - p_{1_y}p_{2_x}}{(|\boldsymbol{p}|^2+1)^2} \right| d\Omega \\ &+ \frac{\theta_1}{2}\int_\Omega (\boldsymbol{p} - \nabla u - \boldsymbol{b_1})^2 d\Omega + \frac{\theta_2}{2}\int_\Omega (\boldsymbol{m} - \frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2+1}} - \boldsymbol{b_2})^2 d\Omega \\ &+ \frac{\theta_3}{2}\int_\Omega (\boldsymbol{n} - \boldsymbol{m} - \boldsymbol{b_3})^2 d\Omega + \frac{\theta_4}{2}\int_\Omega (q - \nabla\cdot\boldsymbol{n} - b_4)^2 d\Omega, \end{aligned} \tag{23}$$

where $\boldsymbol{b_1},\ \boldsymbol{b_2},\ \boldsymbol{b_3},\ b_4$ are the Bregman parameters and $\theta_1, \theta_2, \theta_3, \theta_4$ are the positive penalty parameters. The functional $\mathcal{E}$ can be minimized using an alternating minimization procedure. That is, we solve five subproblems for u, $\boldsymbol{p}$, $\boldsymbol{m}$,$\boldsymbol{n}$, q and update the Bregman parameters $\boldsymbol{b_1}$, $\boldsymbol{b_2}$, $\boldsymbol{b_3}$, b_4 as follows:

$$u^{[\text{new}]} = \arg\min_u \left\{ \bar{\mathcal{D}}^{\text{ROF}}(u,z) + \frac{\theta_1}{2}\int_\Omega (\boldsymbol{p}^{[\text{old}]} - \nabla u - \boldsymbol{b_1}^{[\text{old}]})^2 d\Omega \right\}, \tag{24}$$

$$\begin{aligned} \boldsymbol{p}^{[\text{new}]} = \arg\min_{\boldsymbol{p}} \Bigg\{ & \alpha_1\int_\Omega |\boldsymbol{p}|\, d\Omega - \alpha_2 \int_\Omega \left| \frac{p_{1_x}p_{2_y} - p_{1_y}p_{2_x}}{(|\boldsymbol{p}|^2+1)^2} \right| d\Omega \\ & + \frac{\theta_1}{2}\int_\Omega (\boldsymbol{p} - \nabla u^{[\text{new}]} - \boldsymbol{b_1}^{[\text{old}]})^2 d\Omega + \frac{\theta_2}{2}\int_\Omega (\boldsymbol{m}^{[\text{old}]} - \frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2+1}} - \boldsymbol{b_2}^{[\text{old}]})^2 d\Omega \Bigg\}, \end{aligned} \tag{25}$$

$$\boldsymbol{m}^{[\text{new}]} = \arg\min_{\boldsymbol{m}} \left\{ \frac{\theta_2}{2}\int_\Omega (\boldsymbol{m} - \frac{\boldsymbol{p}^{[\text{new}]}}{\sqrt{|\boldsymbol{p}^{[\text{new}]}|^2+1}} - \boldsymbol{b_2}^{[\text{old}]})^2 d\Omega + \frac{\theta_3}{2}\int_\Omega (\boldsymbol{n}^{[\text{old}]} - \boldsymbol{m} - \boldsymbol{b_3}^{[\text{old}]})^2 d\Omega \right\}, \tag{26}$$

$$\boldsymbol{n}^{[\text{new}]} = \arg\min_{\boldsymbol{n}} \left\{ \frac{\theta_3}{2}\int_\Omega (\boldsymbol{n} - \boldsymbol{m}^{[\text{new}]} - \boldsymbol{b_3}^{[\text{old}]})^2 d\Omega + \frac{\theta_4}{2}\int_\Omega (q^{[\text{old}]} - \nabla\cdot\boldsymbol{n} - b_4^{[\text{old}]})^2 d\Omega \right\}, \tag{27}$$

$$q^{[\text{new}]} = \min_q \left\{ \frac{\alpha_2}{2}\int_\Omega q^2 d\Omega + \frac{\theta_4}{2}\int_\Omega (q - \nabla\cdot\boldsymbol{n}^{[\text{new}]} - b_4^{[\text{old}]})^2 d\Omega \right\}, \tag{28}$$

$$\boldsymbol{b}_1^{[\text{new}]} = \boldsymbol{b}_1^{[\text{old}]} + \nabla u^{[\text{new}]} - \boldsymbol{p}^{[\text{new}]}; \tag{29}$$

$$\boldsymbol{b}_2^{[\text{new}]} = \boldsymbol{b}_2^{[\text{old}]} + \frac{\boldsymbol{p}^{[\text{new}]}}{\sqrt{|\boldsymbol{p}^{[\text{new}]}|^2+1}} - \boldsymbol{m}^{[\text{new}]}; \tag{30}$$

$$\boldsymbol{b}_3^{[\text{new}]} = \boldsymbol{b}_3^{[\text{old}]} + \boldsymbol{m}^{[\text{new}]} - \boldsymbol{n}^{[\text{new}]}; \tag{31}$$

$$b_4^{[\text{new}]} = b_4^{[\text{old}]} + \nabla\cdot\boldsymbol{n}^{[\text{new}]} - q^{[\text{new}]}. \tag{32}$$

Note that the $u, \boldsymbol{m}, \boldsymbol{n}, q$-subproblems can be efficiently solved by their closed-form solution, while the solution of the $\boldsymbol{p}$-subproblem is determined by the associated Euler-Lagrange equation.

u-subproblem. Fixing variables $(\boldsymbol{p}; \boldsymbol{b_1})$ yields the Euler-Lagrange equation associated with u as follows

$$u - \theta_1 \triangle u = G, \tag{33}$$

where $G = z + \theta_1 \nabla \cdot (\boldsymbol{b_1} - \boldsymbol{p})$. To solve the linear partial differential equation in (33), we apply the discrete divergence operator in (16) and the discrete Laplace operator in (21) as follows

$$(u)_{i,j} - \theta_1 \left(\partial_x^- \partial_x^+ (u)_{i,j} + \partial_y^- \partial_y^+ (u)_{i,j}\right) = (G)_{i,j}, \tag{34}$$

where $(G)_{i,j} = (z)_{i,j} + \theta_1 \left(\partial_x^- ((b_{11})_{i,j} - (p_1)_{i,j}) + \partial_y^- ((b_{12})_{i,j} - (p_2)_{i,j})\right)$. As periodic boundary conditions have been imposed on the discrete derivatives, the discrete Fourier transform (DFT) can be directly applied to the both sides of (34)

$$\mathcal{F}\left\{(u)_{i,j} - \theta_1 \left(\partial_x^- \partial_x^+ (u)_{i,j} + \partial_y^- \partial_y^+ (u)_{i,j}\right)\right\} = \mathcal{F}\left\{(G)_{i,j}\right\}, \tag{35}$$

where $\mathcal{F}$ denotes the discrete Fourier transform. For discrete frequencies r and s, we have

$$\underbrace{\left(1 - 2\theta_1 \left(\cos\frac{2\pi s}{N} + \cos\frac{2\pi r}{N} - 2\right)\right)}_{\zeta} \mathcal{F}\left\{(u)_{i,j}\right\} = \mathcal{F}\left\{(G)_{i,j}\right\}, \tag{36}$$

where $i \in [1, M]$ and $j \in [1, N]$ are the indexes in discrete time domain, $r \in [0, M)$ and $s \in [0, N)$ are the frequencies in the discrete frequency domain. (36) provides us with the closed-form solution of u as

$$(u)_{i,j} = \text{Real}\left(\mathcal{F}^{-1}\left\{\frac{\mathcal{F}\left\{(G)_{i,j}\right\}}{\zeta}\right\}\right), \tag{37}$$

where $\mathcal{F}^{-1}$ denotes the discrete inverse Fourier transform. 'Real' is the real part of a complex number. '—' stands for point-wise division of matrices.

q-subproblem. Fixing variables $(\boldsymbol{n}; b_4)$ yields the Euler-Lagrange equation associated with q as follows:

$$\alpha_2 q + \theta_4 (q - \nabla \cdot \boldsymbol{n} - b_4) = 0, \tag{38}$$

which can be determined using the closed-form formula

$$(q)_{i,j} = \frac{\theta_4((\partial_x^- (n_1)_{i,j} + \partial_y^- (n_2)_{i,j}) + (b_4)_{i,j})}{\alpha_2 + \theta_4}. \tag{39}$$

$\boldsymbol{m}$-subproblem. Fixing variables $(\boldsymbol{p}, \boldsymbol{n}; \boldsymbol{b_2}, \boldsymbol{b_3})$ yields the Euler-Lagrange equation associated with $\boldsymbol{m}$ as follows:

$$\theta_2(\boldsymbol{m} - \frac{\boldsymbol{p}}{\sqrt{|\boldsymbol{p}|^2 + 1}} - \boldsymbol{b_2}) - \theta_3(\boldsymbol{n} - \boldsymbol{m} - \boldsymbol{b_3}) = 0, \tag{40}$$

which can be determined using the closed-form formula

$$(\boldsymbol{m})_{i,j} = \frac{\theta_2((\psi \boldsymbol{m})_{i,j} + (\boldsymbol{b_2})_{i,j}) + \theta_3((\boldsymbol{n})_{i,j} - (\boldsymbol{b_3})_{i,j})}{\theta_2 + \theta_3} \tag{41}$$

where $\psi = \frac{1}{\sqrt{|\boldsymbol{p}|^2 + 1}}$.

n-subproblem. By fixing variables $(\boldsymbol{m}, q; \boldsymbol{b_3}, b_4)$, we have the Euler-Lagrange equation associated with $\boldsymbol{n} = (n_1\ n_2)$ as given by

$$\left(\theta_3 - \theta_4\partial_x^+\partial_x^-\right)(n_1)_{i,j} - \theta_4\partial_x^+\partial_y^-(n_2)_{i,j} = (h_1)_{i,j}, \tag{42}$$

$$\left(\theta_3 - \theta_4\partial_y^+\partial_y^-\right)(n_2)_{i,j} - \theta_4\partial_y^+\partial_x^-(n_1)_{i,j} = (h_2)_{i,j}, \tag{43}$$

where

$$(h_1)_{i,j} = \theta_3(m_1 + b_{31}) - \theta_4\partial_x^+(q - b_4), \tag{44}$$

$$(h_2)_{i,j} = \theta_3(m_2 + b_{32}) - \theta_4\partial_y^+(q - b_4). \tag{45}$$

Taking the DFT with (42) and (43) leads to the following system of linear equations

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

where

$$\begin{aligned}
a_{11} &= \theta_3 - 2\theta_4\left(\cos\frac{2\pi s}{N} - 1\right), \\
a_{12} &= -\theta_4\left(-1 - \cos\frac{2\pi r}{M} + \sqrt{-1}\sin\frac{2\pi r}{M}\right)\left(1 - \cos\frac{2\pi s}{N} + \sqrt{-1}\sin\frac{2\pi s}{N}\right), \\
a_{21} &= -\theta_4\left(-1 - \cos\frac{2\pi s}{N} + \sqrt{-1}\sin\frac{2\pi s}{N}\right)\left(1 - \cos\frac{2\pi r}{M} + \sqrt{-1}\sin\frac{2\pi r}{M}\right), \\
a_{22} &= \theta_3 - 2\theta_4\left(\cos\frac{2\pi r}{M} - 1\right).
\end{aligned}$$

For all discrete frequencies, the determinant of the coefficient matrix $\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$ is given by

$$D = \theta_3^2 - 2\theta_3\theta_4\left(\cos\frac{2\pi s}{N} + \cos\frac{2\pi r}{M} - 2\right), \tag{46}$$

which is always positive. Thus, the closed-form solution of n_1 and n_2 can be expressed as

$$\begin{cases} (n_1)_{i,j} = \text{Real}\left(\mathcal{F}^{-1}\left\{\frac{a_{22}\mathcal{F}\{(h_1)_{i,j}\} - a_{12}\mathcal{F}\{(h_2)_{i,j}\}}{D}\right\}\right), \\ (n_2)_{i,j} = \text{Real}\left(\mathcal{F}^{-1}\left\{\frac{a_{11}\mathcal{F}\{(h_2)_{i,j}\} - a_{21}\mathcal{F}\{(h_1)_{i,j}\}}{D}\right\}\right). \end{cases} \tag{47}$$

p-subproblem. Fixing variables (u,$\boldsymbol{m}$; $\boldsymbol{b_1}$,$\boldsymbol{b_2}$) yields the Euler-Lagrange equation associated with $\boldsymbol{p}$ as follows

$$\begin{cases} \alpha_1\psi p_1 - \alpha_2\nabla\cdot\boldsymbol{w}_1 + \alpha_2\beta p_1 + \theta_1(p_1 - u_x - b_{11}) + \theta_2(m_1 - \psi p_1 - b_{21})(\psi^3 p_1^2 - \psi) = 0 \\ \alpha_1\psi p_2 - \alpha_2\nabla\cdot\boldsymbol{w}_2 + \alpha_2\beta p_2 + \theta_1(p_2 - u_y - b_{12}) + \theta_2(m_2 - \psi p_2 - b_{22})(\psi^3 p_2^2 - \psi) = 0 \end{cases}, \tag{48}$$

where $\boldsymbol{w}_1 = \left(-\frac{(p_2)_y}{\beta_2^2}, \frac{(p_2)_x}{\beta_2^2}\right)$, $\boldsymbol{w}_2 = \left(\frac{(p_1)_y}{\beta_2^2}, -\frac{(p_1)_x}{\beta_2^2}\right)$, $\beta_1 = (p_1)_x(p_2)_y - (p_1)_y(p_2)_x$, $\beta_2 = p_1^2 + p_2^2 + 1$, $\beta = \frac{4\beta_1}{\beta_2^3}$, $\psi = \frac{1}{\sqrt{\beta_2}}$. To solve the nonlinear partial differential equation in (48), we apply a coupled outer-inner iteration method as follows. In the outer iteration, we start from an initial solution $\boldsymbol{p}^{[0]}$ and compute a sequence of approximate solutions $\boldsymbol{p}^{[1]}, \boldsymbol{p}^{[2]}, \cdots, \boldsymbol{p}^{[\nu]}, \boldsymbol{p}^{[\nu+1]}, \cdots$ by solving the discrete system of nonlinear equations using the following semi-implicit fixed point iteration scheme:

$$\begin{cases} (p_1^{[\nu+1]})_{i,j} = \dfrac{\alpha_2(\nabla\cdot\boldsymbol{w}_1^{[\nu]})_{i,j} + \theta_1(u_x + b_{11})_{i,j} + \theta_2\psi(m_1 - b_{21})_{i,j}}{\mathcal{N}_1^{[\nu]}} \\ (p_2^{[\nu+1]})_{i,j} = \dfrac{\alpha_2(\nabla\cdot\boldsymbol{w}_2^{[\nu]})_{i,j} + \theta_1(u_y + b_{12})_{i,j} + \theta_2\psi(m_2 - b_{22})_{i,j}}{\mathcal{N}_2^{[\nu]}} \end{cases}, \tag{49}$$

where

$$\mathcal{N}_1^{[\nu]} = \alpha_1\psi^{[\nu]} + \alpha_2\beta^{[\nu]} + \theta_1 + \theta_2\psi^{2[\nu]}((m_1 - \psi^{[\nu]}p_1^{[\nu]} - b_{21})\psi^{[\nu]}p_1^{[\nu]} + 1),$$
$$\mathcal{N}_2^{[\nu]} = \alpha_1\psi^{[\nu]} + \alpha_2\beta^{[\nu]} + \theta_1 + \theta_2\psi^{2[\nu]}((m_2 - \psi^{[\nu]}p_2^{[\nu]} - b_{22})\psi^{[\nu]}p_2^{[\nu]} + 1)$$

until the stopping rule $\max\{\frac{\|p_1^{[\nu]}-p_1^{[\nu-1]}\|}{\|p_1^{[\nu]}\|}, \frac{\|p_2^{[\nu]}-p_2^{[\nu-1]}\|}{\|p_2^{[\nu]}\|}\} < \bar{\epsilon}_1$ or $\nu \geq \bar{\epsilon}_2$ is met for a given threshold $\bar{\epsilon}_1 > 0$ and the maximum iteration $\bar{\epsilon}_2 > 0$.

Our alternating minimization process is repeated until the following stopping rule based on the relative error of the solution is satisfied:

$$\frac{\left\|u^{\text{new}} - u^{\text{old}}\right\|_{l^2}^2}{\left\|u^{\text{old}}\right\|_{l^2}^2} < \epsilon \tag{50}$$

for predefined small $\epsilon > 0$. Finally, proposed numerical algorithm is summarized in Algorithm 1.

Algorithm 1. *Our improved numerical algorithm for TVTC model*

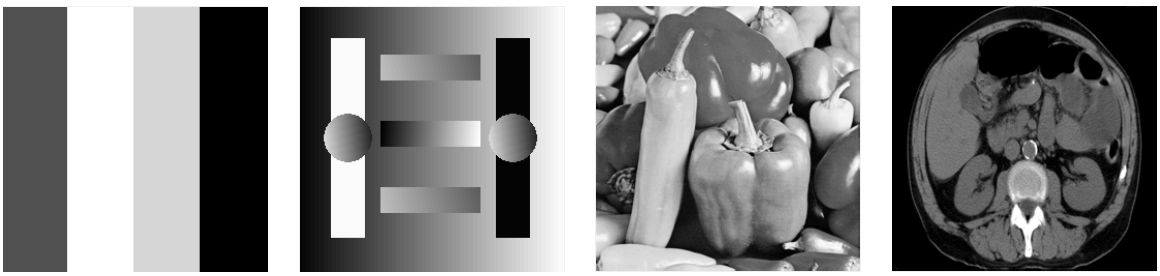
u	*the restored image*
z	*the observed noisy image*
α_1, α_2	*the regularization parameters*
$\boldsymbol{p}, \boldsymbol{m}, \boldsymbol{n}$	*the auxiliary spitting vector variables*
q	*the auxiliary spitting scalar variable*
$\boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}, b_4$	*Bregman iterative parameters*
$\theta_1, \theta_2, \theta_3, \theta_4$	*the penalty parameters*
ϵ_1	*the given threshold* (*typically,* $\epsilon_1 = 5.5 \times 10^{-5}$)
ϵ_2	*the maximum number of iterations* (*typically,* $\epsilon_2 = 1000$)

$$[u] \leftarrow \text{SBTVTC}\,(u, z, \alpha_1, \alpha_2, \boldsymbol{p}, \boldsymbol{m}, \boldsymbol{n}, q, \boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}, b_4, \theta_1, \theta_2, \theta_3, \theta_4, \epsilon_1, \epsilon_2)$$

- **Initialization**: Set $(\boldsymbol{p}, \boldsymbol{m}, \boldsymbol{n}, q; \boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}, b_4) = \boldsymbol{0}, \alpha_1, \alpha_2, \theta_1, \theta_2, \theta_3, \theta_4 > 0$ and $u = z$
- **repeat**
  - Compute u by (37)
  - Compute vector variable $\boldsymbol{p}$ by (49) , $\boldsymbol{m}$ by (41) and $\boldsymbol{n}$ by (47)
  - Compute scalar variable q by (39)
  - Update $\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3, b_4$ by (29) to (32)
- **until** convergence of u

3 Numerical experiments and discussion

In this section, we carry out numerical experiments from several test cases for both synthetic and real images to compare the restoration results of the proposed TVTC model with the classic TV model.



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

Figure 1: The original test images. Synthetic images of 256×256 pixels (the first three columns) and real noisy image of 256×256 pixels (the last column).

Figure 1 shows the original test images. The first three columns show the three synthetic images of 256×256 pixels, while the last column shows the real noisy image of 256×256 pixels. We note first that the noise formation model in (2) was used for creating the noisy versions of the synthetic images with the additive Gaussian noise of variance 0.07. We also note that we used the peak signal to noise ratio (PSNR) between the original and denoised images to measure the quality of image restoration; a higher PSNR value indicates the higher quality of the denoised image (better). PSNR is defined by

$$\text{PSNR} = 10 \log \left(\frac{255^2}{\text{MSE}} \right), \tag{51}$$

where $\text{MSE} = \frac{1}{NM} \sum_{i=1}^{M} \sum_{j=1}^{N} \left((u^*)_{i,j} - (u)_{i,j} \right)$. Here u^* and u are the original image and the denoised image, respectively. The model parameters in (23) were manually-selected. Finally, we note that all experiments were performed using Matlab R2016b on a machine with an Intel Core i5 at 2.3GHz with 8GB of RAM.

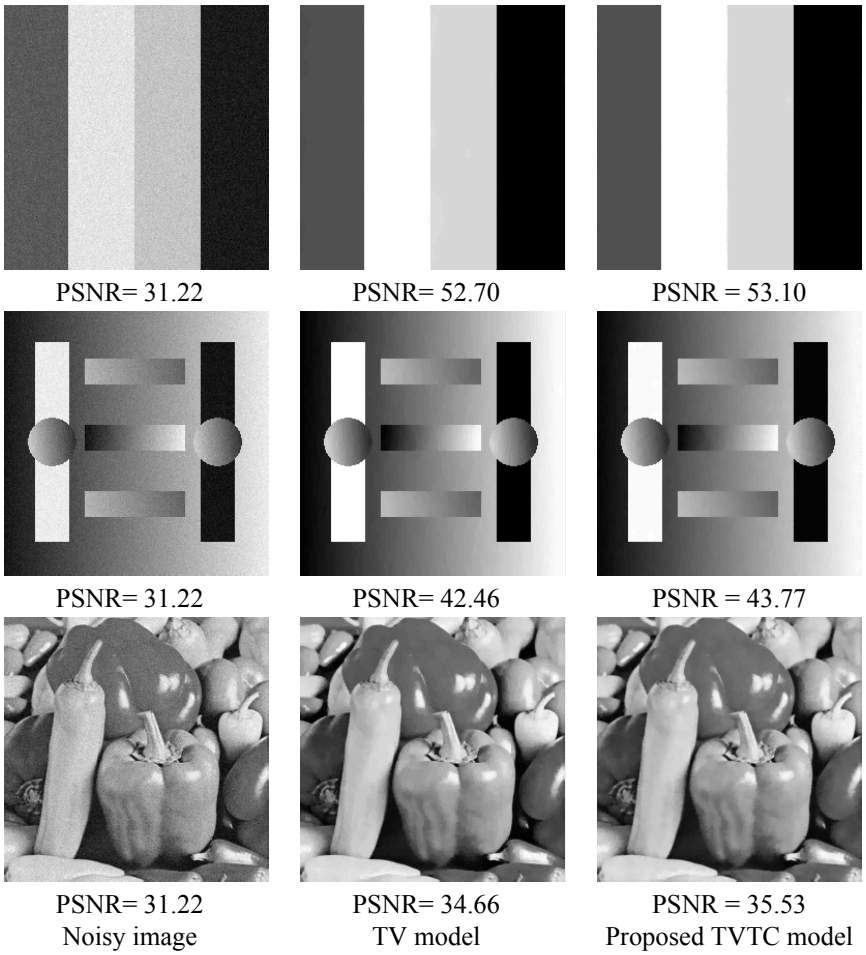


Figure 2: Restoration results on the synthetic images in Figure 1 (the first three columns). From top to bottom: edge-preservation ability test (top row); smoothness-preservation ability test (middle row); real image test (bottom row). From left to right: noisy images (left column); denoised images by the TV model (middle column); and denoised images by the proposed TVTC model (right column).

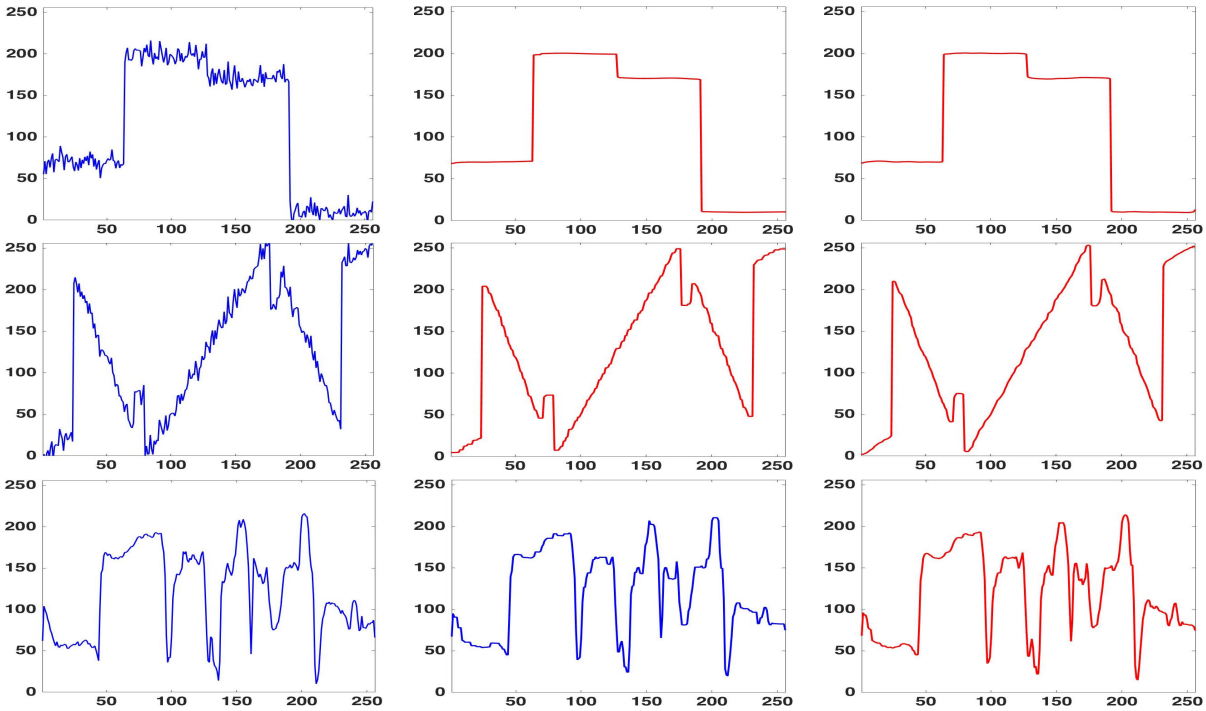


Figure 3: Corresponding signals of the 128th row from the images in Figure 2. Noisy signals (left column); denoised signals by the TV model (middle column); and denoised images by the proposed TVTC model (right column).

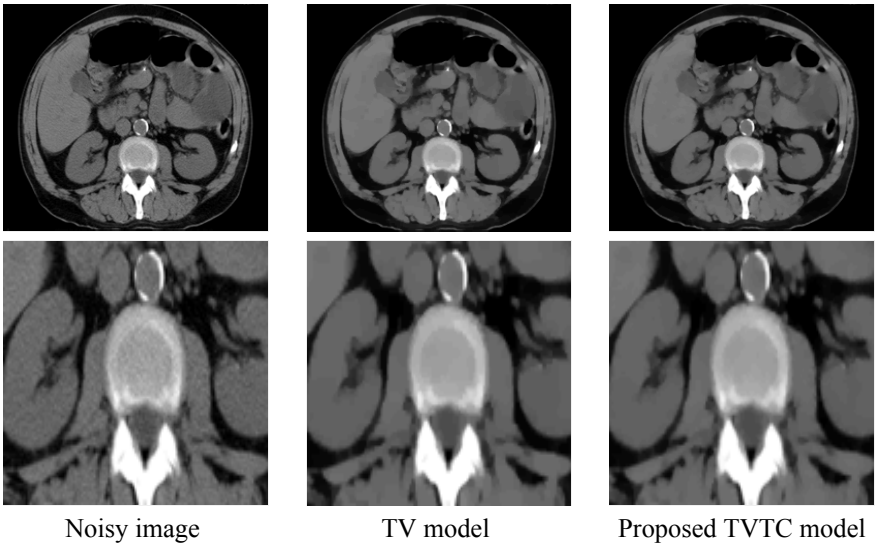


Figure 4: Restoration results on the real noisy images in Figure 1 (d). From top to bottom: full image (top row) and enlarged portions of the MRI "abdomen" image (bottom row); denoised image by the TV model (middle column); and denoised images by the proposed TVTC model (right column).

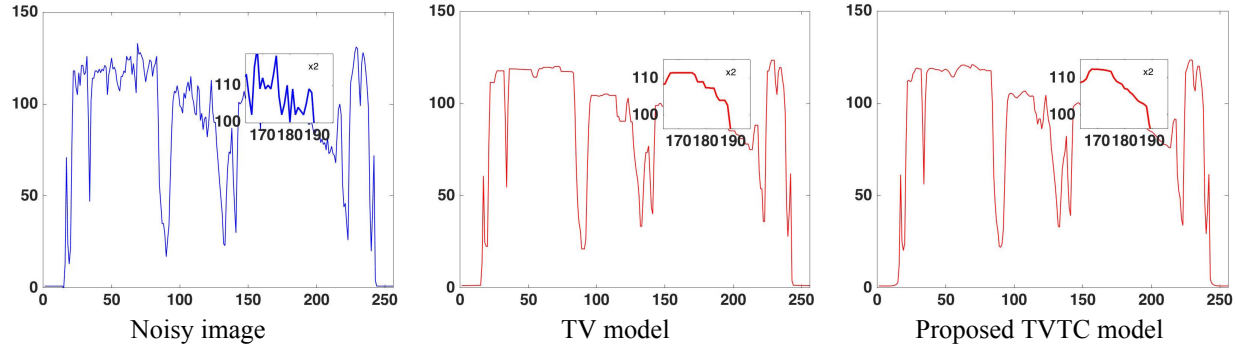


Figure 5: Corresponding signals of the 128th row from the images in Figure 4. Real noisy signal (left column); denoised signal by the TV model (middle column); denoised signal by the proposed TVTC model (right column).

In Figure 2, we test the edge- and smoothness-preservation abilities of the proposed TVTC model and the TV model on three noisy images (*left column*), which are the piecewise constant image (*top row*), the piecewise smooth image (*middle row*), and the real "pepper" image (*bottom row*). The second and last columns of Figure 2 show respectively the restored images by the TV model and the proposed TVTC model. Figure 3 show one-dimensional profiles (or signals) of the noisy images and restored images in Figure 2 (selected from the 128th row with 256 columns). Figures 4 and 5 show the restoration results in removing real noise from the medical image by the TV model and the proposed TVTC model.

As expected, the TV model is less efficient than the proposed TVTC model in terms of PSNR. Moreover, it is clear from Figure 3 (middle and right columns) that the proposed TVTC model is better in recovering the edges.

From Figure 3, one can easily find in the middle columns that the restored signals by the TV model are composed of jagged appearance which is staircase effect. In addition, the restoration results shown from the right column by the proposed TVTC model does not produce evident staircase artifact. In terms of visual inspection, the proposed TVTC model is more efficient than the classic TV model for the smoothness-preservation ability.

Finally, we test the performance of the TV model and the proposed TVTC model on the real MRI images, "abdomen" as shown in Figure 1 (d). As can be seen from Figures 4 and 5, the proposed TVTC model delivers more visually pleasing restoration results than the TV model.

4 Conclusions

This work proposes an improved variational model combining the TV and TC regularization term in removing additive noise from digital images. In order to solve the associated variational problem, we have proposed the efficient numerical algorithm in SB framework. As several, numerical tests confirmed that the proposed method delivers more accurate and reliable restoration results than the classic TV model.

Acknowledgments: The authors would like to thank all reviewers. The first author's work was partially supported by Centre of Excellence in Mathematics and Department of Mathematics, Faculty of Science, Silpakorn university.

References

[1] Geman S, Geman D. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE Trans Pattern Anal Mach Intel. 1984;6(6):721–741.

[2] Rudin L, Osher S, Fatemi E. Nonlinear total variation based noise removal algorithms. Physica D. 1992;60:259–268.

[3] Zhu W, Chan TF. Image Denoising Using Mean Curvature of Image Surface. SIAM J Imaging Sci. 2012;5(1):1–32.

[4] Brito-Loeza C, Chen K, Uc-Cetina V. Image denoising using the Gaussian curvature of the image surface. Numer Meth Part Differ Equ. 2016;32(3):1066–1089.

[5] Goldstein T, Osher S. The Split Bregman Method for L1-Regularized Problems. SIAM Journal on Imaging Sciences. 2009;2(2):323–343.

[6] Brito-Loeza C, Chen K. Fast iterative algorithms for solving the minimization of curvature-related functionals in surface fairing. International Journal of Computer Mathematics. 2013;90:92–108.