A fast operator-splitting method for Beltrami color image denoising

In memory of Roland Glowinski–a dear friend, mentor, colleague and great leader

Yuping Duan[1], Qiuxiang Zhong[1], Xue-Cheng Tai[2] and Roland Glowinski[2,3]

[1]Center for Applied Mathematics, Tianjin University, Tianjin, 300072, China.
[2]Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong.
[3]Department of Mathematics, University of Houston, Honston, TX 77204, USA.

Contributing authors: yuping.duan@tju.edu.cn; zhongqiuxiang@tju.edu.cn; tai@hkbu.edu.hk;

Abstract

The Beltrami framework is a successful technique for color image denosing by regarding color images as manifolds embedded in a five dimensional spatial-chromatic space. It can ideally model the coupling between the color channels rather than treating them as if they were independent. However, the resulting model with high nonlinearity makes the related optimization problems difficult to solve numerically. In this paper, we propose an operator-splitting method for a variant of the Beltrami regularization model. From the optimality conditions associated with the minimization of the Beltrami regularized functional, we derive an initial value problem (gradient flow). We solve the gradient flow problem by an operator-splitting scheme involving three fractional steps. All three subproblem solutions can be obtained in closed form or computed by one-step Newton's method. We demonstrate the efficiency and robustness of the proposed algorithm by conducting a series of experiments on real image denoising problems, where more than half of the computational time is saved compared to the existing augmented Lagrangian method (ALM) based algorithm for solving the Beltrami minimization model.

Keywords: Color image denosing, Beltrami minimization, diffusion, operator-splitting method

1 Introduction

Along with the development of image acquisition technology, color images become increasingly popular and the demand for color image processing is fast growing [1–5]. One of the fundamental and challenging tasks for color image processing is the coupling between the channels. Let Ω be an open bounded domain in $\mathbb{R}^2$, and $\boldsymbol{f} = \boldsymbol{v} + \boldsymbol{n} : \Omega \to \mathbb{R}^3$ be the degraded color image with $\boldsymbol{v} : \Omega \to \mathbb{R}^3$ being the original color image and $\boldsymbol{n}$ being the interference noises. The goal of color image denoising is to recover the latent image $\boldsymbol{v}$ from the corrupted image $\boldsymbol{f}$ by various methods such as filter methods,

partial differential equation (PDE) based methods, variational regularization methods and machine learning based methods.

In the past several decades, filter methods have been common strategies for effectively suppressing the disturbing noises. Deng and Cahill [6] proposed an adaptive Gaussian filter to reduce the Gaussian noises and preserve the broad outlines. However, it may result in signal distortion. Chen *et al.* [7] presented a nonlinear filter, named the tri-state median filter, which can simultaneously suppress impulse noises and preserve fine image details. The bilateral filter proposed by Tomasi and Manduchi [8] is a non-iterative, local and nonlinear filter, which can smooth out noises as well as preserve edges based on both geometric closeness and photometric similarity. The Wiener filter was used in the wavelet domain, which minimizes the mean squared error between the estimated random process and the desired process by the statistical approach [9]. Nevertheless, it is difficult to provide perfect restoration owing to the uncertain statistical procedure during the acquisition. More recently, the non-local means [10] and sparse 3D transform-domain collaborative filtering (BM3D) [1] were developed by taking advantage of the similarities among patches for noise removal. Although this kind of methods can achieve state-of-the-art denoising performance, it requires high computational costs in practice. By taking the geometric properties of image into account, Gong *et al.* [11] proposed the curvature filter to reduce the variational energies for image restoration. On the other hand, PDE based diffusion methods have also been studied for image denoising problems [12–15]. For example, Tschumperlé [16] introduced a tensor-driven PDE based on the curvatures of specific integral curves to well preserve thin structures in image restoration. Shan, Sun and Guo [17] proposed a nonlinear diffusion equation with smooth solution to remove multiplicative noises.

Although the filtering and PDE based methods can effectively reduce noises and preserve the main image structures, these methods ignore prior information of images and lack theoretical guarantees. Rudin, Osher and Fetami [18] proposed the well-known total variation (TV) regularizer for gray-scale image denoising, which was shown effective for removing noises and preserving sharp jumps. The TV model has been intensively studied and extended to the color image denoising problems by Blomgren and Chan [19], leading to

$$\min_{\boldsymbol{v}} \int_{\Omega} \sqrt{\sum_{i=1}^{3} |\nabla v_i|^2} d\mathbf{x} + \frac{\eta}{2} \int_{\Omega} |\boldsymbol{f} - \boldsymbol{v}|^2 d\mathbf{x}, \tag{1}$$

where the multi-dimensional TV is defined as the Euclidean norm of the vector of channel-wise scalar TV. Although the color TV couples the channels together and is global, it is very weak, all channels being treated the same. Goldluecke *et al.* [20] derived a convex vectorial TV from the generalized Jacobians in geometric measure theory. However, it may lead to undesirable uneven color effects and result in a computational burden. To overcome these limitations, Ono and Yamada [21] presented a decorrelated vectorial TV prior by measuring the discrete gradients of luminance component and chrominance component. Later, Lefkimmiatis *et al.* [22] constructed a novel generic energy functional by penalizing the eigenvalues of the structure tensor, named as the structure tensor total variation, which can provide more robust measurement of the image variation by making use of the structure tensor in capturing the first-order information. Jia, Ng and Wang [23] developed a saturation-value total variation in the hue, saturation, and value color space for color image restoration, which achieves better performance than the existing color image total variation both quantitatively and visually.

Different from the methods mentioned above, Batard and Bertalmio [24] presented a generalization of the Euclidean and Riemannian total variation to a vector bundle based covariant differentiation for color images, where the standard differentiation of a function is replaced by the covariant differentiation of a section. Particularly, the generalized Laplacians acting on sections of vector bundles over Riemannian manifolds were studied for multi-channel image processing in [25]. As a representative of generalized Laplacians, the Laplace-Beltrami operator has been studied for color image

problems [26], which generalizes the Laplacian from Euclidean space (flat domains) to Riemannian space (curved manifolds) by regarding a color image as a two dimensional manifold embedded into a five-dimensional spatial-chromatic space. More specifically, the Beltrami regularized model for color image denoising can be formulated as follows

$$\min_{\boldsymbol{v}} \int_{\Omega} \sqrt{1+\beta^2 \sum_{i=1}^{3} |\nabla v_i|^2 + \frac{\beta^4}{2} \sum_{i,j=1}^{3} |\nabla v_i \times \nabla v_j|^2} d\mathbf{x} + \frac{\eta}{2} \int_{\Omega} |\boldsymbol{f} - \boldsymbol{v}|^2 d\mathbf{x}, \tag{2}$$

where β is a positive constant used to determine the ratio between the spatial and color distances. Note that the cross product term in (2) can penalize the misalignment of the gradient directions between the color channels.

The essential idea of the Beltrami filter is to minimize the area of the image manifold. Due to its success in color image denoising, the Beltrami filter has also been applied to other color image processing tasks such as image enhancement [27], blind deblurring [28], image segmentation [29], and image dehazing [30]. However, the minimization of the Beltrami functional is very challenging. Kimmel, Malladi and Sochen [31] originally solved the Euler-Lagrange equation of the Beltrami minimization model by gradient descent, which is stable only for very small time steps and therefore require many iterations. Spira [32] constructed a 2D short time kernel for the Beltrami flow, which enables the implementation of the Beltrami flow by convolving the image with the kernel similar to the solution of the heat equation by a convolution with a Gaussian kernel. Rosman *et al.* [33] used vector extrapolation techniques to accelerate the convergence of explicit schemes for the Beltrami flow. Rosman *et al.* [34] introduced a semi-implicit Crank-Nicolson scheme based on locally one-dimensional/additive operator splitting for implementing the anisotropic Beltrami operator. Later, the authors proposed an augmented Lagrangian method (ALM) for color image processing with Beltrami regularization [35]. Very recently, Wang and Ng [36] investigated vector bundles of color images in saturation-value color space.

In this paper, we propose a novel operator-splitting method based on Lie scheme [37, 38] to efficiently and effectively solve the Beltrami-related minimization problem for color image restoration. We introduce the vector-valued and matrix-valued variables to replace the gradient norm and the cross product term in (2), by which we decouple the nonlinearity from the differential operators and make the energy easier to deal with. Then, we reformulate the energy functional by drawing into two indicator functions and obtain an unconstrained minimization problem. Moreover, we derive the optimality PDE system and transform it into an initial value problem by introducing an artificial time variable. Eventually, we obtain the solutions to all subproblems associated with the Lie scheme fractional steps by either closed-form solution/fast Fourier transform (FFT) or one-step iterative scheme. Different from the augmented Lagrangian method, our proposed approach has much fewer parameters to adjust and converges much faster under the same stopping criteria. Numerical experiments demonstrate the efficiency and robustness of the proposed method for color image denoising both quantitatively and qualitatively.

The remainder of this paper is organized as follows: The reformulation of the Beltrami minimization model is described in Sect. 2. We describe the operator-splitting method based on the Lie scheme in Sect. 3. In Sect. 4 we discuss the full discretization of the sub-problems and the solution of their discrete analogues. In Sect. 5 we apply our methodology to the solution of various color imaging problems. Finally, we summarize and conclude our paper in Sect. 6.

2 Reformulating the Beltrami minimization

Considering the advantages of Beltrami flow in color image problems and the lack of tractable solution methodologies, we aim to establish more efficient methods to solve the Beltrami minimization problem.

Supposing that $\beta \gg 1$, we can simplify the Beltrami minimization problem (2) by removing the constants as follows

$$\inf_{\boldsymbol{v}\in\boldsymbol{V}} \int_\Omega \sqrt{|\nabla \boldsymbol{v}|^2 + \beta^2 \sum_{i,j} |\nabla \boldsymbol{v}_i \times \nabla \boldsymbol{v}_j|^2} d\mathbf{x} + \frac{\eta}{2} \int_\Omega |\boldsymbol{f} - \boldsymbol{v}|^2 d\mathbf{x}, \tag{3}$$

where $\boldsymbol{V}$ denotes the Sobolev space $(\mathcal{H}^1(\Omega))^3$. The proposed model can be regarded as a modified color TV minimization problem, where we introduce a coupling cross term to model the alignments between channels of color images. The minimization problem (3) is highly nonlinear, mostly due to the coupled cross product term in the regularization. Thus, we introduce vector-valued and matrix-valued variables to replace the gradient and cross product terms, respectively. In particular, we define the matrix-valued function $\boldsymbol{q}$ as

$$\boldsymbol{q} = \begin{pmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \\ q_{31} & q_{32} \end{pmatrix} = \nabla \boldsymbol{v},$$

where q_{kl} denotes the partial derivative $\frac{\partial \boldsymbol{v}_k}{\partial x_l}$ for $k = 1, 2, 3$ and $l = 1, 2$ making $\boldsymbol{q}$ the Jacobi matrix of $\boldsymbol{v}$. We then use the vector-valued function $\boldsymbol{F}(\boldsymbol{q})$ to denote the cross product term as follows

$$\boldsymbol{F}(\boldsymbol{q}) = \begin{pmatrix} q_{21}q_{32} - q_{22}q_{31} \\ q_{31}q_{12} - q_{11}q_{32} \\ q_{11}q_{22} - q_{12}q_{21} \end{pmatrix} = \boldsymbol{q}_k \times \boldsymbol{q}_l,$$

where $\boldsymbol{q}_k = (q_{k1}\ q_{k2})$ for $k = 1, 2, 3$. We further define the sets Σ_f and Ξ by

$$\Sigma_f = \Big\{ \boldsymbol{q} \in \boldsymbol{Q} \ \Big|\ \exists\ \boldsymbol{v} \in \boldsymbol{V},\ \text{s.t. } \boldsymbol{q} = \nabla \boldsymbol{v},\quad \int_\Omega \boldsymbol{v} d\mathbf{x} = \int_\Omega \boldsymbol{f} d\mathbf{x},\quad \boldsymbol{Q} = (L^2(\Omega))^{3\times 2} \Big\},$$

and

$$\Xi = \Big\{ (\boldsymbol{q}, \boldsymbol{z}) \in \boldsymbol{Q} \times \boldsymbol{Z} \ \Big|\ \boldsymbol{z} = \boldsymbol{F}(\boldsymbol{q}),\ \boldsymbol{Q} = (L^2(\Omega))^{3\times 2},\ \boldsymbol{Z} = (L^2(\Omega))^3 \Big\}.$$

Based on the above notation, the original Beltrami minimization problem (3) can be reformulated in the following equivalent form

$$\min_{(\boldsymbol{q},\boldsymbol{z})\in\boldsymbol{Q}\times\boldsymbol{Z}} \int_\Omega \sqrt{|\boldsymbol{q}|^2 + \beta^2 |\boldsymbol{z}|^2} d\mathbf{x} + I_{\Sigma_f}(\boldsymbol{q}) + I_\Xi(\boldsymbol{q}, \boldsymbol{z}) + \frac{\eta}{2} \int_\Omega |\boldsymbol{f} - \boldsymbol{v}(\boldsymbol{q})|^2 d\mathbf{x}, \tag{4}$$

where I_{Σ_f} and I_Ξ are the indicator functionals defined by

$$I_{\Sigma_f}(\boldsymbol{q}) = \begin{cases} 0, & \text{if } \boldsymbol{q} \in \Sigma_f, \\ +\infty, & \text{otherwise,} \end{cases}$$

and

$$I_\Xi(\boldsymbol{q}, \boldsymbol{z}) = \begin{cases} 0, & \text{if } (\boldsymbol{q}, \boldsymbol{z}) \in \Xi, \\ +\infty, & \text{otherwise.} \end{cases}$$

Then, as introduced in [39, 40], $\boldsymbol{v}(\boldsymbol{q})$ becomes the solution of the following system

$$\begin{cases} \nabla^2 \boldsymbol{v}(\boldsymbol{q}) = \nabla \cdot \boldsymbol{q} & \text{in } \Omega, \\ \boldsymbol{v}(\boldsymbol{q}) \text{ verifies periodic boundary condition} & \text{on } \partial\Omega, \\ \int_\Omega \boldsymbol{v}(\boldsymbol{q}) d\mathbf{x} = \int_\Omega \boldsymbol{f} d\mathbf{x}. \end{cases} \tag{5}$$

Above, $\boldsymbol{q}$ and $\boldsymbol{v}$ are vector and matrix-valued functions. The operators ∇^2 and $\nabla\cdot$ are row-wise operators. If $\boldsymbol{q} \in (L^2(\Omega))^{3\times 2}$, problem (5) has a optimal solution, which minimizes the original problem (3).

Remark 2.1. *Note that if $\beta = 0$ in the Beltrami regularization, the proposed model reduces to the classical color TV model, which can remove noises and preserve sharp jumps, but suffers from staircasing and uneven color effects. For $\beta > 0$ in the Beltrami regularization, the cross product term can penalize the misalignment of the gradient directions between the color channels, which is helpful to handle the drawbacks of color TV regularization.*

Remark 2.2. *Although the periodic boundary condition is used, there is no problem to switch to Neumann boundary condition. Actually, the model and algorithm proposed in this paper can be adapted to zero Neumann boundary condition with minor modification.*

Remark 2.3. *The constraint $\int_\Omega v dx = \int_\Omega f dx$ in Σ_f assumes the observed images are corrupted by the additive Gaussian noises with zero mean and the standard deviation σ. When the noises are non-additive or with non-zero mean, the constraint has to be changed accordingly, but it will not affect the general framework of the operator-splitting method. Suppose the images are degraded by the additive Gaussian noises with mean μ ($\mu > 0$) and standard deviation σ. The constraint becomes $\frac{1}{|\Omega|}\int_\Omega(\boldsymbol{f}-\boldsymbol{v})d\mathbf{x} = \mu$ and $\frac{1}{|\Omega|}\int_\Omega(\boldsymbol{f}-\boldsymbol{v}-\mu)^2 d\mathbf{x} = \sigma^2$ with $|\Omega|$ representing the area of image domain [41]. Similarly, when the images are degraded by multiplicative noises with zero mean and the standard deviation σ, the data-fidelity term in (3) should be modified, subject to $\int_\Omega(\boldsymbol{v} - \boldsymbol{f}\log\boldsymbol{v})d\mathbf{x} = 0$, and the constraint becomes $\int_\Omega \boldsymbol{v} d\mathbf{x} = \int_\Omega \boldsymbol{f} d\mathbf{x}$ and $\frac{1}{|\Omega|}\int_\Omega(\frac{\boldsymbol{f}}{\boldsymbol{v}} - 1)^2 d\mathbf{x} = \sigma^2$ accordingly; see the reference [42] for more details.*

3 An operator-splitting method for the solution of problem (4)

In this section, we develop an operator-splitting approach to solve the unconstrained optimization problem (4). Operator-splitting is a methodology which has been widely used for the solution of a large variety of problems from Mechanics and Physics (see [38] for details and further references). Recently it founds applications to the solution of various problems from image processing [37, 43, 44].

3.1 The optimality condition

Let us define the functionals J_1 and J_2 by

$$\begin{cases} J_1(\boldsymbol{q},\boldsymbol{z}) = \int_\Omega \sqrt{|\boldsymbol{q}|^2 + \beta^2|\boldsymbol{z}|^2}d\mathbf{x}, \\ J_2(\boldsymbol{q}) = \frac{\eta}{2}\int_\Omega |\boldsymbol{f} - \boldsymbol{v}(\boldsymbol{q})|^2 d\mathbf{x}. \end{cases} \tag{6}$$

Suppose that $(\boldsymbol{p},\boldsymbol{y})$ is a minimizer of the functional in (4). Then $(\boldsymbol{p},\boldsymbol{y})$ should (at least formally) satisfy the following first-order optimality condition

$$\begin{cases} \partial_{\boldsymbol{q}} J_1(\boldsymbol{p},\boldsymbol{y}) + \partial_{\boldsymbol{q}} I_\Xi(\boldsymbol{p},\boldsymbol{y}) + \partial I_{\Sigma_f}(\boldsymbol{p}) + DJ_2(\boldsymbol{p}) \ni \mathbf{0}, \\ \partial_{\boldsymbol{z}} J_1(\boldsymbol{p},\boldsymbol{y}) + \partial_{\boldsymbol{z}} I_\Xi(\boldsymbol{p},\boldsymbol{y}) \ni \mathbf{0}, \end{cases} \tag{7}$$

where $\{\partial_{\boldsymbol{q}}, \partial_{\boldsymbol{z}}\}$ represent the partial derivatives with respect to (w.r.t.) $\{\boldsymbol{q},\boldsymbol{z}\}$ for smooth functions and the sub-derivative for the indicator functions, and D denotes classical differentials.

In order to solve (7), we introduce an artificial time variable and solve the following initial value problem to steady state

$$\begin{cases} \frac{\partial \boldsymbol{p}}{\partial t} + \partial_{\boldsymbol{q}} J_1(\boldsymbol{p},\boldsymbol{y}) + \partial_{\boldsymbol{q}} I_\Xi(\boldsymbol{p},\boldsymbol{y}) + \partial I_{\Sigma_f}(\boldsymbol{p}) + DJ_2(\boldsymbol{p}) \ni \boldsymbol{0}, \\ \gamma \frac{\partial \boldsymbol{y}}{\partial t} + \partial_{\boldsymbol{z}} J_1(\boldsymbol{p},\boldsymbol{y}) + \partial_{\boldsymbol{z}} I_\Xi(\boldsymbol{p},\boldsymbol{y}) \ni \boldsymbol{0}, \\ (\boldsymbol{p}(0),\boldsymbol{y}(0)) = (\boldsymbol{p}_0,\boldsymbol{y}_0), \end{cases} \tag{8}$$

where $(\boldsymbol{p}_0,\boldsymbol{y}_0)$ denotes a given initial condition, and γ is a positive parameter used to control the evolution speed of $\boldsymbol{y}$. The initial value problem (8) is a natural candidate to a solution method of the operator-splitting type, where Lie and Marchuk-Yanenko schemes are simplest and popular choices and have been used for Euler's elastica model [39, 40] and Gaussian curvature model [45].

3.2 An operator-splitting method

We employ an operator-splitting method based on the Lie scheme [37] to time-discretize problem (8). Let n be the iteration number and $\tau > 0$ be a time discretization step. We set $t^n = n\tau$ and compute $(\boldsymbol{p},\boldsymbol{y})$ by the following procedure

Initialization. Set $(\boldsymbol{p}^0,\boldsymbol{y}^0) = (\boldsymbol{p}_0,\boldsymbol{y}_0)$. Then, do the following for $n = 0, 1, , 2\cdots$.
The 1st fractional step. Solve

$$\begin{cases} \begin{cases} \frac{\partial \boldsymbol{p}}{\partial t} + \partial_{\boldsymbol{q}} J_1(\boldsymbol{p},\boldsymbol{y}) \ni \boldsymbol{0}, \\ \gamma \frac{\partial \boldsymbol{y}}{\partial t} + \partial_{\boldsymbol{z}} J_1(\boldsymbol{p},\boldsymbol{y}) \ni \boldsymbol{0}, \end{cases} & \text{in } \Omega \times (t^n, t^{n+1}), \\ (\boldsymbol{p}(t^n),\boldsymbol{y}(t^n)) = (\boldsymbol{p}^n,\boldsymbol{y}^n), \end{cases} \tag{9}$$

and set $(\boldsymbol{p}^{n+1/3},\boldsymbol{y}^{n+1/3}) = (\boldsymbol{p}(t^{n+1}),\boldsymbol{y}(t^{n+1}))$.
The 2nd fractional step. Solve

$$\begin{cases} \begin{cases} \frac{\partial \boldsymbol{p}}{\partial t} + \partial_{\boldsymbol{q}} I_\Xi(\boldsymbol{p},\boldsymbol{y}) \ni \boldsymbol{0}, \\ \gamma \frac{\partial \boldsymbol{y}}{\partial t} + \partial_{\boldsymbol{z}} I_\Xi(\boldsymbol{p},\boldsymbol{y}) \ni \boldsymbol{0}, \end{cases} & \text{in } \Omega \times (t^n, t^{n+1}), \\ (\boldsymbol{p}(t^n),\boldsymbol{y}(t^n)) = (\boldsymbol{p}^{n+1/3},\boldsymbol{y}^{n+1/3}), \end{cases} \tag{10}$$

and set $(\boldsymbol{p}^{n+2/3},\boldsymbol{y}^{n+2/3}) = (\boldsymbol{p}(t^{n+1}),\boldsymbol{y}(t^{n+1}))$.
The 3rd fractional step. Solve

$$\begin{cases} \begin{cases} \frac{\partial \boldsymbol{p}}{\partial t} + \partial I_{\Sigma_f}(\boldsymbol{p}) + DJ_2(\boldsymbol{p}) \ni \boldsymbol{0}, \\ \gamma \frac{\partial \boldsymbol{y}}{\partial t} = \boldsymbol{0}, \end{cases} & \text{in } \Omega \times (t^n, t^{n+1}), \\ (\boldsymbol{p}(t^n),\boldsymbol{y}(t^n)) = (\boldsymbol{p}^{n+2/3},\boldsymbol{y}^{n+2/3}), \end{cases} \tag{11}$$

and set $(\boldsymbol{p}^{n+1},\boldsymbol{y}^{n+1}) = (\boldsymbol{p}(t^{n+1}),\boldsymbol{y}(t^{n+1}))$.

As can be observed, the Lie scheme (9)-(11) is only semi-discrete with three initial value problems. Suppose that all subproblems are discretized using *one step of the backward Euler scheme*. We have the following time discretization scheme, namely Marchuk-Yanenko scheme

$$(\boldsymbol{p}^0,\boldsymbol{y}^0) = (\boldsymbol{p}_0,\boldsymbol{y}_0). \tag{12}$$

For the iteration $n \geq 0$, we update $(\boldsymbol{p}^n, \boldsymbol{y}^n) \to (\boldsymbol{p}^{n+1/3}, \boldsymbol{y}^{n+1/3}) \to (\boldsymbol{p}^{n+2/3}, \boldsymbol{y}^{n+2/3}) \to (\boldsymbol{p}^{n+1}, \boldsymbol{y}^{n+1})$ step by step as follows

$$\begin{cases} \frac{\boldsymbol{p}^{n+1/3}-\boldsymbol{p}^n}{\tau} + \partial_{\boldsymbol{q}} J_1(\boldsymbol{p}^{n+1/3}, \boldsymbol{y}^{n+1/3}) \ni \boldsymbol{0}, \\ \gamma\frac{\boldsymbol{y}^{n+1/3}-\boldsymbol{y}^n}{\tau} + \partial_{\boldsymbol{z}} J_1(\boldsymbol{p}^{n+1/3}, \boldsymbol{y}^{n+1/3}) \ni \boldsymbol{0}, \end{cases} \quad \text{in } \Omega \Rightarrow (\boldsymbol{p}^{n+1/3}, \boldsymbol{y}^{n+1/3}), \tag{13}$$

then

$$\begin{cases} \frac{\boldsymbol{p}^{n+2/3}-\boldsymbol{p}^{n+1/3}}{\tau} + \partial_{\boldsymbol{q}} I_\Xi(\boldsymbol{p}^{n+2/3}, \boldsymbol{y}^{n+2/3}) \ni \boldsymbol{0}, \\ \gamma\frac{\boldsymbol{y}^{n+2/3}-\boldsymbol{y}^{n+1/3}}{\tau} + \partial_{\boldsymbol{z}} I_\Xi(\boldsymbol{p}^{n+2/3}, \boldsymbol{y}^{n+2/3}) \ni \boldsymbol{0}, \end{cases} \quad \text{in } \Omega \Rightarrow (\boldsymbol{p}^{n+2/3}, \boldsymbol{y}^{n+2/3}), \tag{14}$$

and finally

$$\begin{cases} \frac{\boldsymbol{p}^{n+1}-\boldsymbol{p}^{n+2/3}}{\tau} + \partial I_{\Sigma_f}(\boldsymbol{p}^{n+1}) + DJ_2(\boldsymbol{p}^{n+1}) \ni \boldsymbol{0}, \\ \gamma\frac{\boldsymbol{y}^{n+1}-\boldsymbol{y}^{n+2/3}}{\tau} = \boldsymbol{0}, \end{cases} \quad \text{in } \Omega \Rightarrow (\boldsymbol{p}^{n+1}, \boldsymbol{y}^{n+1}). \tag{15}$$

Scheme (13)-(15) is an approximation to the gradient flow of (4), the convergence of which relies on the choice of the step size τ. Because of the complex structure of the functions J_1, J_2, I_{Σ_f} and I_Ξ in our model, it is difficult to prove the convergence of the scheme theoretically, an issue need to be studied in the future. More discussion on the convergence of Lie and Marchuk-Yanenko schemes are provided in Appendix.

Remark 3.1. *In fact, the above operator-splitting method can be implemented even when the constant 1 is kept in the Beltrami regularized model (3). In that condition, the Euler-Lagrange equation of system (13) becomes a highly nonlinear minimization problem, which is difficult to solve numerically, resulting in slow convergence as well as high computational costs. More importantly, the reduced model (3) can produece comparable restoration results shown by the numerical experiments in Sect. 5. Thus, for the sake of simplicity and efficiency, we concern with the reduced Beltrami minimization model and its operator-splitting scheme in this work.*

3.3 On the solution of (13)

System (13) is nothing but the Euler-Lagrange equation of the following minimization problem

$$\begin{aligned} (\boldsymbol{p}^{n+1/3}, \boldsymbol{y}^{n+1/3}) = \arg \inf_{(\boldsymbol{q},\boldsymbol{z})\in \boldsymbol{Q}\times \boldsymbol{Z}} & \frac{1}{2}\int_\Omega \Big(|\boldsymbol{q}|^2 + \gamma|\boldsymbol{z}|^2\Big)d\mathbf{x} \\ & - \int_\Omega \Big(\boldsymbol{p}^n \cdot \boldsymbol{q} + \gamma \boldsymbol{y}^n \cdot \boldsymbol{z}\Big)d\mathbf{x} + \tau \int_\Omega \sqrt{|\boldsymbol{q}|^2 + \beta^2|\boldsymbol{z}|^2}d\mathbf{x}. \end{aligned} \tag{16}$$

Pay attention that the choice of γ is used to balance the two variables $\boldsymbol{q}$ and $\boldsymbol{z}$ by $|\boldsymbol{q}|^2 + \gamma|\boldsymbol{z}|^2$. Since the combination is also determined by the Beltrami regularization term (i.e., $|\boldsymbol{q}|^2 + \beta^2|\boldsymbol{z}|^2$), we simply let $\gamma = \beta^2$ to maintain the consistency and reduce the burden of parameter selection. Hereafter, γ is replaced by β^2 in all subproblems without further specification. Consequently, we can reformulate the minimization problem (16) into an equivalent L^2 and L^1 minimization problem as follows

$$(\boldsymbol{p}^{n+1/3}, \boldsymbol{y}^{n+1/3}) = \arg \min_{(\boldsymbol{q},\boldsymbol{z})\in \boldsymbol{Q}\times \boldsymbol{Z}} \frac{1}{2}\big\|(\boldsymbol{q}, \beta\boldsymbol{z})\big\|_2^2 - \big\langle(\boldsymbol{p}^n, \beta\boldsymbol{y}^n), (\boldsymbol{q}, \beta\boldsymbol{z})\big\rangle + \tau\|(\boldsymbol{q}, \beta\boldsymbol{z})\|_2, \tag{17}$$

where $\big\langle(\boldsymbol{p}, \boldsymbol{y}), (\boldsymbol{q}, \boldsymbol{z})\big\rangle = \int_\Omega \boldsymbol{p}\cdot\boldsymbol{q} + \boldsymbol{y}\cdot\boldsymbol{z}d\mathbf{x}$, and $\big\|(\boldsymbol{q}, \boldsymbol{z})\big\|_2 = \sqrt{\big\langle(\boldsymbol{q}, \boldsymbol{z}), (\boldsymbol{q}, \boldsymbol{z})\big\rangle}$ for $\boldsymbol{p}, \boldsymbol{q} \in \boldsymbol{Q}$ and $\boldsymbol{y}, \boldsymbol{z} \in \boldsymbol{Z}$. It is well-known that the above minimization problem has a closed-from solution. More specifically,

a linear relation between the solutions of $(n+1/3)$-th iteration and n-th iteration holds such that

$$\boldsymbol{p}^{n+1/3} = \lambda \boldsymbol{p}^n, \quad \boldsymbol{y}^{n+1/3} = \lambda \boldsymbol{y}^n,$$

with λ a positive scalar to be determined. Thus, we can rewrite (16) in terms of λ as

$$\lambda^* = \arg\min_{\lambda \in \mathbb{R}} \ (\frac{1}{2}\lambda^2 - \lambda)\big(|\boldsymbol{p}^n|^2 + \beta^2|\boldsymbol{y}^n|^2\big) + \lambda\tau\sqrt{|\boldsymbol{p}^n|^2 + \beta^2|\boldsymbol{y}^n|^2}, \tag{18}$$

the optimality condition of which satisfies

$$(\lambda - 1)\big(|\boldsymbol{p}^n|^2 + \beta^2|\boldsymbol{y}^n|^2\big) + \tau\sqrt{|\boldsymbol{p}^n|^2 + \beta^2|\boldsymbol{y}^n|^2} = 0. \tag{19}$$

As can be seen, there exists a closed-form solution to the above one-dimensional minimization problem, which is

$$\lambda^* = \left(1 - \frac{\tau}{\sqrt{|\boldsymbol{p}^n|^2 + \beta^2|\boldsymbol{y}^n|^2}}\right)^+, \tag{20}$$

where the operator $(\cdot)^+$ is defined as $(a)^+ := \max(a, 0)$ for $a \in \mathbb{R}$. Therefore, the solution to $(\boldsymbol{p}^{n+1/3}, \boldsymbol{y}^{n+1/3})$ is given by

$$\begin{pmatrix} \boldsymbol{p}^{n+1/3} \\ \boldsymbol{y}^{n+1/3} \end{pmatrix} = \left(1 - \frac{\tau}{\sqrt{|\boldsymbol{p}^n|^2 + \beta^2|\boldsymbol{y}^n|^2}}\right)^+ \begin{pmatrix} \boldsymbol{p}^n \\ \boldsymbol{y}^n \end{pmatrix}. \tag{21}$$

3.4 On the solution of (14)

Similarly, the multi-variable system (14) with respect to $(\boldsymbol{q}, \boldsymbol{z})$ is the Euler-Lagrange equation of the following minimization problem

$$(\boldsymbol{p}^{n+2/3}, \boldsymbol{y}^{n+2/3}) = \arg\inf_{(\boldsymbol{q},\boldsymbol{z}) \in \Xi} \frac{1}{2}\int_\Omega \Big(|\boldsymbol{q}|^2 + \beta^2|\boldsymbol{z}|^2\Big)d\mathbf{x} - \int_\Omega \big(\boldsymbol{p}^{n+1/3} \cdot \boldsymbol{q} + \beta^2 \boldsymbol{y}^{n+1/3} \cdot \boldsymbol{z}\big)d\mathbf{x}. \tag{22}$$

By replacing $\boldsymbol{z}$ by $\boldsymbol{F}(\boldsymbol{q})$, we obtain

$$\begin{cases} \boldsymbol{y}^{n+2/3} = \boldsymbol{F}(\boldsymbol{p}^{n+2/3}) \ \text{ with} \\ \boldsymbol{p}^{n+2/3} = \arg\min\limits_{\boldsymbol{q} \in \boldsymbol{Q}} \ \frac{1}{2}\int_\Omega (|\boldsymbol{q}|^2 + \beta^2|\boldsymbol{F}(\boldsymbol{q})|^2)d\mathbf{x} - \int_\Omega \Big(\boldsymbol{p}^{n+1/3} \cdot \boldsymbol{q} + \beta^2 \boldsymbol{y}^{n+1/3} \cdot \boldsymbol{F}(\boldsymbol{q})\Big)d\mathbf{x}, \end{cases} \tag{23}$$

in which we have

$$|\boldsymbol{q}|^2 = \sum_{k=1}^{3}\sum_{l=1}^{2} q_{kl}^2,$$

and

$$|\boldsymbol{F}(\boldsymbol{q})|^2 = (q_{21}q_{32} - q_{22}q_{31})^2 + (q_{31}q_{12} - q_{11}q_{32})^2 + (q_{11}q_{22} - q_{12}q_{21})^2.$$

By regarding $\boldsymbol{q}$ as a vector in $\mathbb{R}^6$, the first-order optimality condition of the minimization problem (23) gives us a cubic system $G\Big((q_{kl})\Big) = \mathbf{0}$, which is given as follows

$$\begin{cases} \beta^2\big[(q_{22}^2+q_{32}^2)q_{11}-(q_{21}q_{22}+q_{31}q_{32})q_{12}\big]+q_{11}+\beta^2(y_2^{n+1/3}q_{32}-y_3^{n+1/3}q_{22})-p_{11}^{n+1/3}=0;\\ \beta^2\big[(q_{21}^2+q_{31}^2)q_{12}-(q_{21}q_{22}+q_{31}q_{32})q_{11}\big]+q_{12}+\beta^2(y_3^{n+1/3}q_{21}-y_2^{n+1/3}q_{31})-p_{12}^{n+1/3}=0;\\ \beta^2\big[(q_{12}^2+q_{32}^2)q_{21}-(q_{11}q_{12}+q_{31}q_{32})q_{22}\big]+q_{21}+\beta^2(y_3^{n+1/3}q_{12}-y_1^{n+1/3}q_{32})-p_{21}^{n+1/3}=0;\\ \beta^2\big[(q_{11}^2+q_{31}^2)q_{22}-(q_{11}q_{12}+q_{31}q_{32})q_{21}\big]+q_{22}+\beta^2(y_1^{n+1/3}q_{31}-y_3^{n+1/3}q_{11})-p_{22}^{n+1/3}=0;\\ \beta^2\big[(q_{12}^2+q_{22}^2)q_{31}-(q_{11}q_{12}+q_{21}q_{22})q_{32}\big]+q_{31}+\beta^2(y_1^{n+1/3}q_{22}-y_2^{n+1/3}q_{12})-p_{31}^{n+1/3}=0;\\ \beta^2\big[(q_{11}^2+q_{21}^2)q_{32}-(q_{11}q_{12}+q_{21}q_{22})q_{31}\big]+q_{32}+\beta^2(y_2^{n+1/3}q_{11}-y_1^{n+1/3}q_{21})-p_{32}^{n+1/3}=0. \end{cases} \tag{24}$$

The multivariate nonlinear equation system (24) can be solved by various iterative methods, such as fixed-point method [46], Jacobi iterative method [47], Gauss-Seidel iterative method [48] and Newton's method [49, 50]. In this work, we use the one-step Newton's method to obtain the solution $\boldsymbol{p}^{n+2/3}$ and update $\boldsymbol{y}^{n+2/3}$ by $\boldsymbol{y}^{n+2/3}=\boldsymbol{F}(\boldsymbol{p}^{n+2/3})$.

3.5 On the solution of (15)

We can directly obtain the solution $\boldsymbol{y}^{n+1}$ from (15) as

$$\boldsymbol{y}^{n+1}=\boldsymbol{y}^{n+2/3}. \tag{25}$$

Thus, we only need to solve $\boldsymbol{p}^{n+1}$ from the following minimization problem

$$\boldsymbol{p}^{n+1}=\arg\inf_{\boldsymbol{q}\in\Sigma_f}\ \frac{1}{2}\int_\Omega|\boldsymbol{q}|^2d\mathbf{x}-\int_\Omega\boldsymbol{p}^{n+2/3}\cdot\boldsymbol{q}d\mathbf{x}+\frac{\tau\eta}{2}\int_\Omega|\boldsymbol{f}-\boldsymbol{v}(\boldsymbol{q})|^2d\mathbf{x}. \tag{26}$$

Since $\boldsymbol{v}(\boldsymbol{q})$ is the solution of system (5), we have $\boldsymbol{p}^{n+1}=\nabla\boldsymbol{u}^{n+1}$ with $\boldsymbol{u}^{n+1}=(u_1^{n+1},u_2^{n+1},u_3^{n+1})^T$, which means that problem (26) is equivalent to

$$\begin{cases} \boldsymbol{p}^{n+1}=\nabla\boldsymbol{u}^{n+1}\ \ \text{with}\\ \boldsymbol{u}^{n+1}=\arg\min\limits_{\boldsymbol{v}\in\boldsymbol{V}}\ \frac{1}{2}\int_\Omega|\nabla\boldsymbol{v}|^2d\mathbf{x}-\int_\Omega\boldsymbol{p}^{n+2/3}\cdot\nabla\boldsymbol{v}d\mathbf{x}+\frac{\tau\eta}{2}\int_\Omega|\boldsymbol{f}-\boldsymbol{v}|^2d\mathbf{x}. \end{cases} \tag{27}$$

As can be seen, $\boldsymbol{u}^{n+1}$ is the solution of the following well-posed linear variational problem

$$\begin{cases} \boldsymbol{u}^{n+1}\in\boldsymbol{V}\ \ \text{s.t.}\\ \int_\Omega\nabla\boldsymbol{u}^{n+1}\cdot\nabla\boldsymbol{v}d\mathbf{x}+\tau\eta\int_\Omega\boldsymbol{u}^{n+1}\cdot\boldsymbol{v}d\mathbf{x}=\int_\Omega\boldsymbol{p}^{n+2/3}\cdot\nabla\boldsymbol{v}d\mathbf{x}+\tau\eta\int_\Omega\boldsymbol{f}\cdot\boldsymbol{v}d\mathbf{x},\quad\forall\boldsymbol{v}\in\boldsymbol{V}. \end{cases} \tag{28}$$

Equivalently, the solution to (28) is the weak solution of the following linear elliptic problem

$$\begin{cases} -\nabla^2\boldsymbol{u}^{n+1}+\tau\eta\boldsymbol{u}^{n+1}=-\nabla\cdot\boldsymbol{p}^{n+2/3}+\tau\eta\boldsymbol{f} & \text{in }\Omega,\\ \boldsymbol{u}^{n+1}\text{ verifies periodic boundary condition} & \text{on }\partial\Omega, \end{cases} \tag{29}$$

which can be solved by fast dedicated algorithms such as FFT [51]. Once we obtain the solution $\boldsymbol{u}^{n+1}$, we update $\boldsymbol{p}$ by $\boldsymbol{p}^{n+1}=\nabla\boldsymbol{u}^{n+1}$.

3.6 Our algorithm

In summary, the proposed operator-splitting method based on the Lie-scheme to solve the Beltrami minimization problem (4) for color image denoising is summarized in Algorithm 1 below.

Algorithm 1 The operator-splitting method for the Beltrami regularized model (3) and (4)

1: **Input:** The degraded image $\boldsymbol{f}$, the weight parameters $\eta > 0$, $\beta > 0$, time step size τ, and stopping threshold ϵ.
2: **Initialization:** $\boldsymbol{u}^0 = \boldsymbol{f}$ and $\boldsymbol{p}^0 = \nabla \boldsymbol{f}$, $\boldsymbol{y}^0 = \boldsymbol{F}(\boldsymbol{p}^0)$, and set $n = 0$.
3: **while** $\|\boldsymbol{u}^{n+1} - \boldsymbol{u}^n\|_2 / \|\boldsymbol{u}^{n+1}\|_2 > \epsilon$ **do**
4: Compute $\{\boldsymbol{p}^{n+1/3}, \boldsymbol{y}^{n+1/3}\}$ based on (16) and (21) to solve system (13);
5: Compute $\{\boldsymbol{p}^{n+2/3}, \boldsymbol{y}^{n+2/3}\}$ based on (23) and (24) to solve system (14);
6: Compute $\{\boldsymbol{u}^{n+1}, \boldsymbol{p}^{n+1}, \boldsymbol{y}^{n+1}\}$ based on (25), (27) and (29) to solve system (15);
7: Update $n = n + 1$;
8: **end while**
9: **Output:** The denoised image $\boldsymbol{u}^* = \boldsymbol{u}^{n+1}$.

4 Space discretization

4.1 Discrete operators

Let $\Omega \in \mathbb{R}^2$ be the discretized image domain with $I \times J$ pixels, and let x, y denote the horizontal and vertical directional coordinates, respectively. For a vector-valued function $\boldsymbol{u} = (u_1, u_2, u_3)^T$ (resp. scalar-valued function u) defined on Ω, its (i,j)-th pixel is expressed by $\boldsymbol{u}(i,j) = (u_1(i,j), u_2(i,j), u_3(i,j))^T$ (resp. $u(i,j)$). Based on the periodic boundary conditions, we define the discrete forward $(+)$ and backward $(-)$ differential operators as

$$\partial_x^+ u(i,j) = \begin{cases} \frac{u(i+1,j)-u(i,j)}{\Delta x}, & 1 \le i < I, \\ \frac{u(1,j)-u(I,j)}{\Delta x}, & i = I, \end{cases}$$

$$\partial_y^+ u(i,j) = \begin{cases} \frac{u(i,j+1)-u(i,j)}{\Delta y}, & 1 \le j < J, \\ \frac{u(i,1)-u(i,J)}{\Delta y}, & j = J, \end{cases}$$

$$\partial_x^- u(i,j) = \begin{cases} \frac{u(i,j)-u(i-1,j)}{\Delta x}, & 1 < i \le I, \\ \frac{u(1,j)-u(I,j)}{\Delta x}, & i = 1, \end{cases}$$

$$\partial_y^- u(i,j) = \begin{cases} \frac{u(i,j)-u(i,j-1)}{\Delta y}, & 1 < j \le J, \\ \frac{u(i,1)-u(i,J)}{\Delta y}, & j = 1, \end{cases}$$

where Δx and Δy denote the spatial mesh sizes. With the above notation, the discrete gradient operator is defined using the forward differential operators as follows

$$\nabla u(i,j) = \big(\partial_x^+ u(i,j), \partial_y^+ u(i,j)\big).$$

Besides, we also define the discrete divergence operators using the backward differential operators for $\boldsymbol{p} = (p_1\ p_2)$ as

$$\mathrm{div}\boldsymbol{p}(i,j) = \partial_x^- p_1(i,j) + \partial_y^- p_2(i,j).$$

4.2 Approximating $(\mathbf{p}^{n+1/3}, \mathbf{y}^{n+1/3})$

From (21), the discrete analogue of $(\boldsymbol{p}^{n+1/3}, \boldsymbol{y}^{n+1/3})$ in problem (16) is given by

$$p_{kl}^{n+1/3}(i,j) = \max\left(1 - \frac{\tau}{\sqrt{\sum_{k=1}^{3}\sum_{l=1}^{2}|p_{kl}^n(i,j)|^2 + \beta^2\sum_{k=1}^{3}|y_k^n(i,j)|^2}}, 0\right)p_{kl}^n(i,j), \quad \text{for } k=1,2,3;\ \ l=1,2 \tag{30}$$

and

$$y_k^{n+1/3}(i,j) = \max\left(1 - \frac{\tau}{\sqrt{\sum_{k=1}^{3}\sum_{l=1}^{2}|p_{kl}^n(i,j)|^2 + \beta^2\sum_{k=1}^{3}|y_k^n(i,j)|^2}}, 0\right)y_k^n(i,j), \quad \text{for } k=1,2,3. \tag{31}$$

4.3 Approximating $(\mathbf{p}^{n+2/3}, \mathbf{y}^{n+2/3})$

The minimization problem in (22) is a highly nonlinear functional. Its Euler Lagrange equation gives the multivariate nonhomogeneous and nonlinear system (24). Using the one-step Newton's method to solve it, we obtain

$$\boldsymbol{p}^{n+2/3} = \boldsymbol{p}^{n+1/3} - G'\Big((p_{kl}^{n+1/3})\Big)^{-1} G\Big((p_{kl}^{n+1/3})\Big), \tag{32}$$

where G' denotes the Jacobian matrix of G. Correspondingly, we use

$$\boldsymbol{y}^{n+2/3}(i,j) = \boldsymbol{F}(\boldsymbol{p}^{n+2/3}(i,j)) = \begin{pmatrix} p_{21}^{n+2/3}(i,j)p_{32}^{n+2/3}(i,j) - p_{22}^{n+2/3}(i,j)p_{31}^{n+2/3}(i,j) \\ p_{31}^{n+2/3}(i,j)p_{12}^{n+2/3}(i,j) - p_{11}^{n+2/3}(i,j)p_{32}^{n+2/3}(i,j) \\ p_{11}^{n+2/3}(i,j)p_{22}^{n+2/3}(i,j) - p_{12}^{n+2/3}(i,j)p_{21}^{n+2/3}(i,j) \end{pmatrix} \tag{33}$$

to obtain $\boldsymbol{y}^{n+2/3}$.

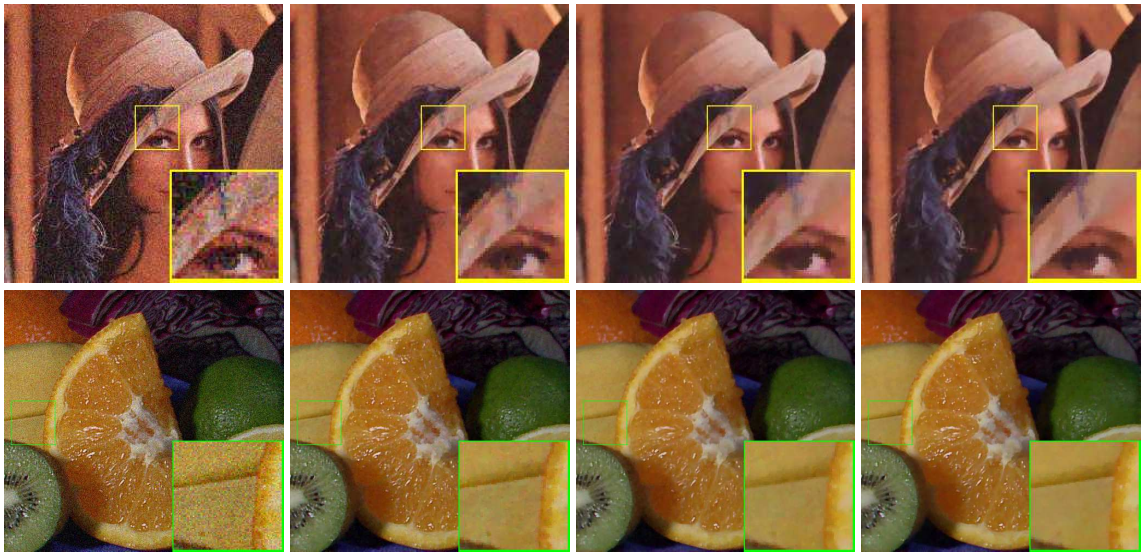
4.4 Approximating $(\mathbf{p}^{n+1}, \mathbf{y}^{n+1})$

The tensor-valued function $\boldsymbol{p}^{n+1}$ is obtained by $\boldsymbol{p}^{n+1} = \nabla \boldsymbol{u}^{n+1}$, where $\boldsymbol{u}^{n+1} = (u_1^{n+1}, u_2^{n+1}, u_3^{n+1})^T$ is the solution of the following linear elliptic system

$$\begin{cases} -\nabla^2 u_k^{n+1} + \tau\eta u_k^{n+1} = -\nabla\cdot\boldsymbol{p}_k^{n+2/3} + \tau\eta f_k & \text{in } \Omega, \\ u_k^{n+1} \text{ verifies periodic boundary condition} & \text{on } \partial\Omega, \end{cases} \quad \text{for } k=1,2,3. \tag{34}$$

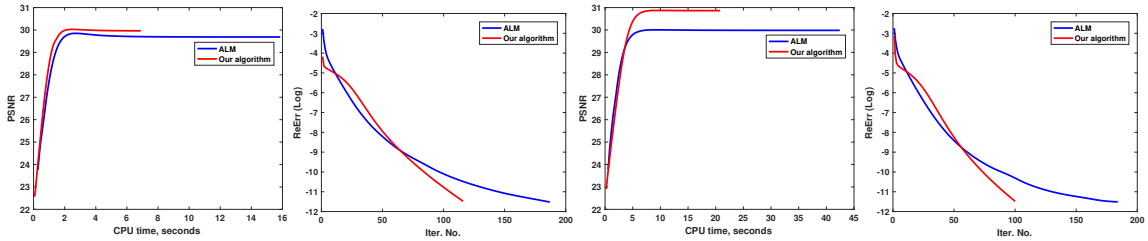
Thus, we discretize equation (34) as follows

$$(-\partial_x^-\partial_x^+ - \partial_y^-\partial_y^+ + \tau\eta\mathcal{I})u_k^{n+1} = -\partial_x^- p_{k1}^{n+2/3} - \partial_y^- p_{k2}^{n+2/3} + \tau\eta f_k,$$



(a) Noisy image ($\sigma = 20$) (b) Color TV (c) ALM (d) Our algorithm

Fig. 1 The Gaussian denoising results ($\sigma = 20$) for 'Lena' (top) and 'Fruits' (bottom) by color TV, augmented Lagrangian method and the proposed operator splitting method for Beltrami regularization model. The parameters are set as: (b) Color TV: $\lambda = 15$, $r = 40$, (c) ALM: $\lambda = 0.4$, $\beta = \sqrt{4000}$, (d) Our algorithm: $\eta = 10$, $\tau = 0.02$, $\beta = 10$.



(a) PSNR('Lena') (b) ReErr('Lena') (c) PSNR('Fruits') (d) ReErr('Fruits')

Fig. 2 Evolution of PSNR and relative error for 'Lena' and 'Fruits' corrupted by a white Gaussian noises with mean zero and standard deviation $\sigma = 20$.

where $\mathcal{I}$ denotes the identity operator. Then one can solve the above linear equation by FFT as

$$u_k^{n+1} = \text{Real}\left[\mathcal{F}^{-1}\left(\frac{\mathcal{F}(-\partial_x^- p_{k1}^{n+2/3} - \partial_y^- p_{k2}^{n+2/3} + \tau\eta f_k)}{\mathcal{F}(-\partial_x^-\partial_x^+ - \partial_y^-\partial_y^+ + \tau\eta\mathcal{I})}\right)\right], \quad \text{for } k = 1, 2, 3, \tag{35}$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote the discrete fast Fourier transform and its inverse, respectively, and Real$(\cdot)$ denotes the real part of its argument. When we obtain $\boldsymbol{u}^{n+1}$, we compute $\boldsymbol{p}^{n+1}$ as

$$\boldsymbol{p}_k^{n+1} = \nabla u_k^{n+1}, \quad \text{for } k = 1, 2, 3. \tag{36}$$

According to (25), one has

$$y_k^{n+1} = y_k^{n+2/3}, \quad \text{for } k = 1, 2, 3. \tag{37}$$

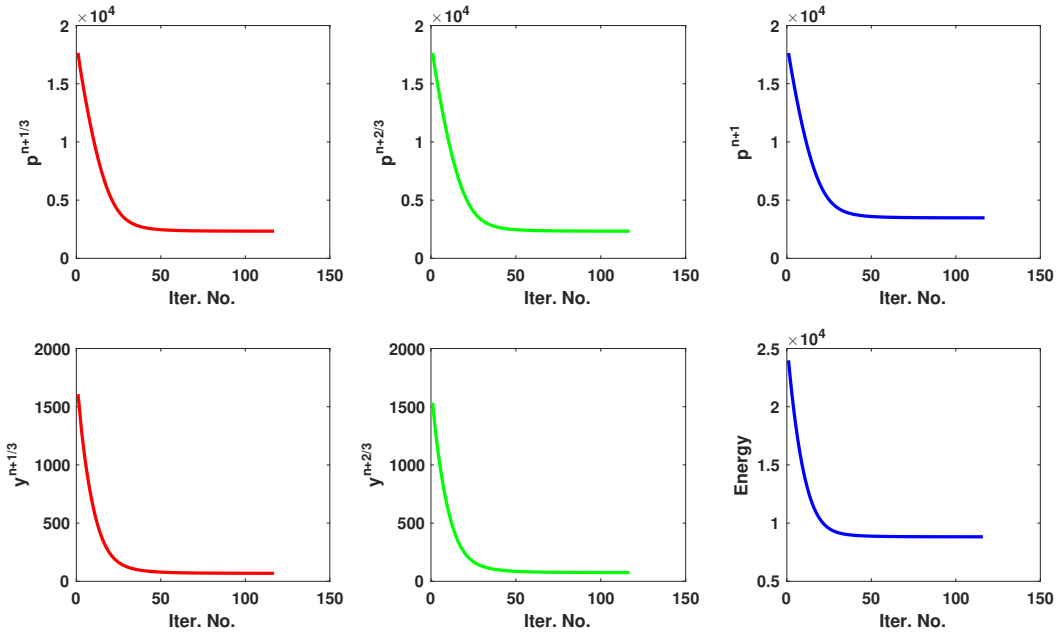


Fig. 3 Energy changes for each subproblem for Gaussian denoising of 'Lena' ($\sigma = 20$) by the proposed method. From left to right: the $\boldsymbol{p}^{n+1/3}$-subproblem, the $\boldsymbol{p}^{n+2/3}$-subproblem, the $\boldsymbol{p}^{n+1}$-subproblem, the $\boldsymbol{y}^{n+1/3}$-subproblem, the $\boldsymbol{y}^{n+2/3}$-subproblem and the total energy, respectively.

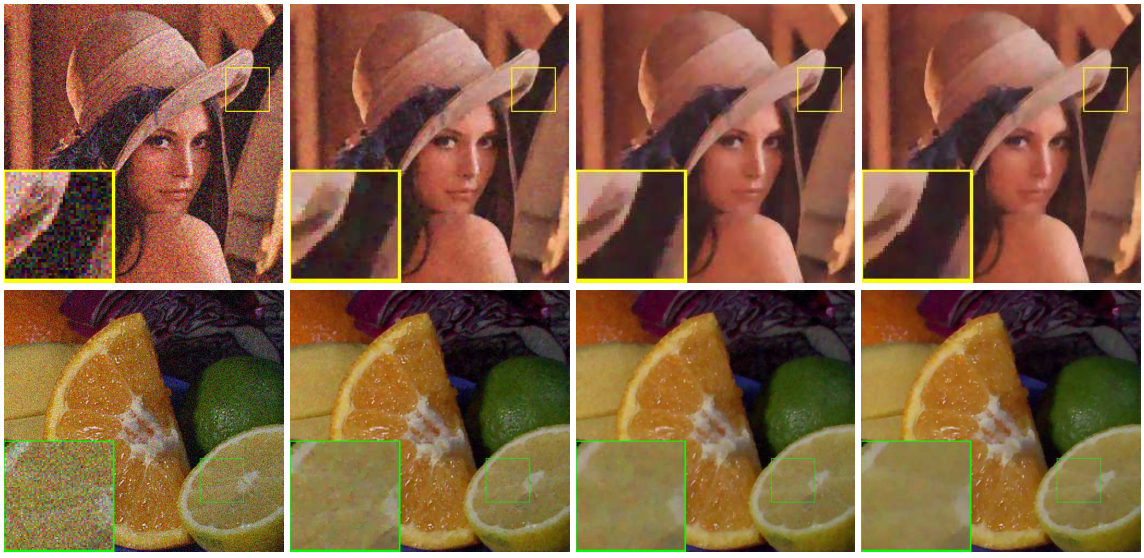
Table 1 Comparison of color TV, augmented Lagrangian method and the proposed algorithm on color image Gaussian noise removal.

Noise level	Images	Methods	PSNR	SSIM	Iterations	Time(s)	Time/Iteration
$\sigma = 20$	Lena (256 × 256)	Color TV	29.47	0.9420	**69**	**2.20**	**0.0319**
		ALM	29.68	0.9425	188	15.83	0.0842
		Our algorithm	**29.94**	**0.9468**	115	6.95	0.0604
	Fruits (512 × 480)	Color TV	30.24	0.9355	**68**	**9.71**	**0.1428**
		ALM	30.01	0.9425	185	42.61	0.2303
		Our algorithm	**30.86**	**0.9480**	100	20.99	0.2099
$\sigma = 30$	Lena (256 × 256)	Color TV	27.76	0.9189	**102**	**3.21**	**0.0315**
		ALM	27.94	0.9210	227	17.03	0.0750
		Our algorithm	**28.21**	**0.9261**	177	10.95	0.0619
	Fruits (512 × 480)	Color TV	28.56	0.9130	**101**	**14.26**	**0.1412**
		ALM	28.24	0.9205	209	55.18	0.2640
		Our algorithm	**28.98**	**0.9265**	162	36.25	0.2238

5 Numerical experiments

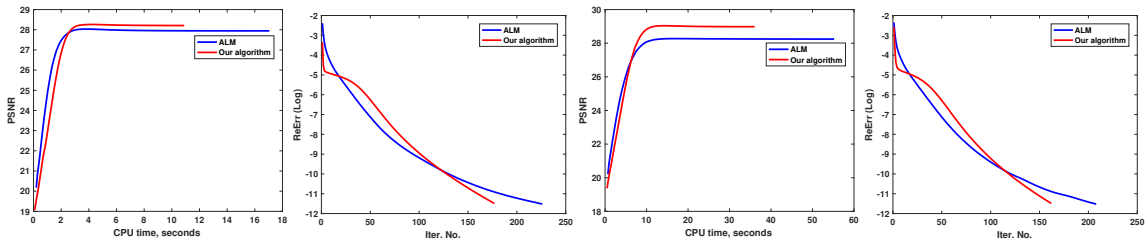
In this section, the results of comprehensive experiments on color image restoration with different noise distributions are reported. All numerical experiments are performed in a Matlab R2017b environment on a machine with 3.40GHz Intel(R) Core(TM) i7-6700 CPU and 32GB RAM. Note that the intensities of all images are normalized to the range of $[0, 1]$. For simplicity, we set the mesh size as $\Delta x = \Delta y = 1$.

In our experiments, we adopt the popular Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measurement (SSIM) [52] to quantitatively evaluate the restoration performance under different image degradation conditions. Generally speaking, higher PSNR and SSIM values mean better denoising results. In addition, both relative error and numerical energy are provided to illustrate the numerical convergence of the proposed operator-splitting algorithm. These quantities



(a) Noisy($\sigma = 30$) (b) Color TV (c) ALM (d) Our algorithm

Fig. 4 The Gaussian denoising results ($\sigma = 30$) for 'Lena' (top) and 'Fruits' (bottom) by color TV, augmented Lagrangian method and the proposed operator splitting method for Beltrami regularization model. The parameters are set as: (b) Color TV: $\lambda = 9$, $r = 40$, (c) ALM: $\lambda = 0.25$, $\beta = \sqrt{4000}$, (d) Our algorithm: $\eta = 6$, $\tau = 0.02$, $\beta = 10$.



(a) PSNR('Lena') (b) ReErr('Lena') (c) PSNR('Fruits') (d) ReErr('Fruits')

Fig. 5 The comparison of PSNR and relative error for 'Lena' and 'Fruits' corrupted by Gaussian noises with mean zero and standard deviation $\sigma = 30$ between ALM and our operator-splitting method.

are defined by

$$\mathrm{ReErr}(\boldsymbol{u}^k) = \frac{\|\boldsymbol{u}^k - \boldsymbol{u}^{k-1}\|_2}{\|\boldsymbol{u}^k\|_2}, \tag{38}$$

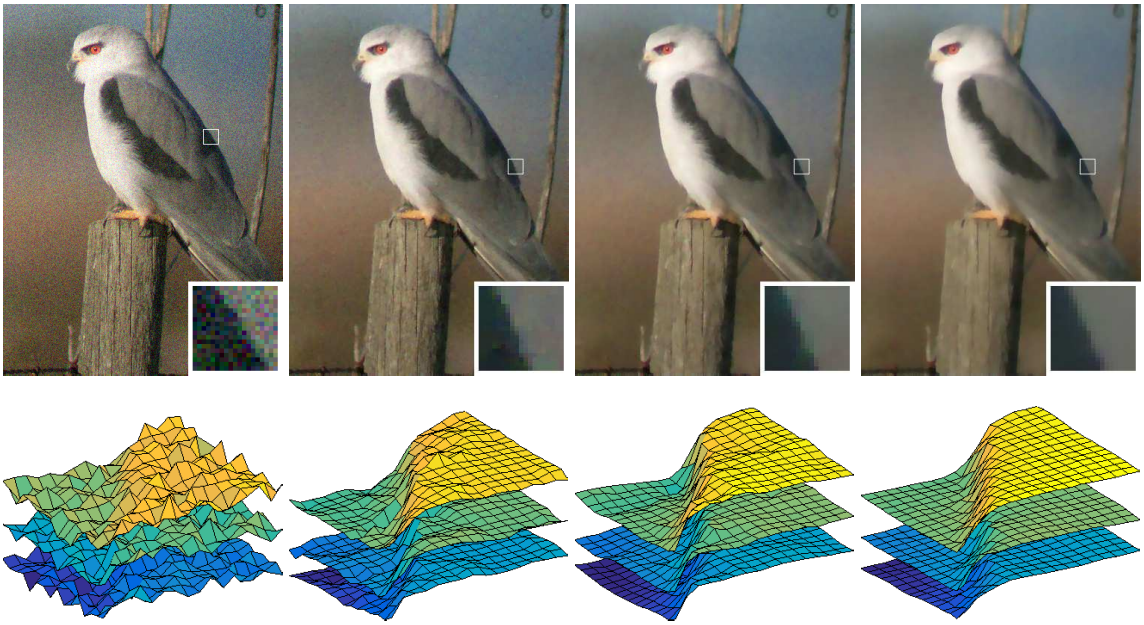
and

$$E(\boldsymbol{u}^k) = \int_\Omega \sqrt{|\nabla \boldsymbol{u}^k|^2 + \beta^2 \sum_{i,j} |\nabla \boldsymbol{u}_i^k \times \nabla \boldsymbol{u}_j^k|^2} d\mathbf{x} + \frac{\eta}{2} \int_\Omega |\boldsymbol{f} - \boldsymbol{u}^k|^2 d\mathbf{x}, \tag{39}$$

respectively.

5.1 Comparison methods and parameters selection

We verify the efficiency and robustness of the proposed operator-splitting method for (3) by comparing with both the color TV method [53] and the augmented Lagrangian method for Beltrami minimization model [35]. For a fair comparison, the parameters of each method are tuned in order to provide the best possible results, the settings of which are given as follows

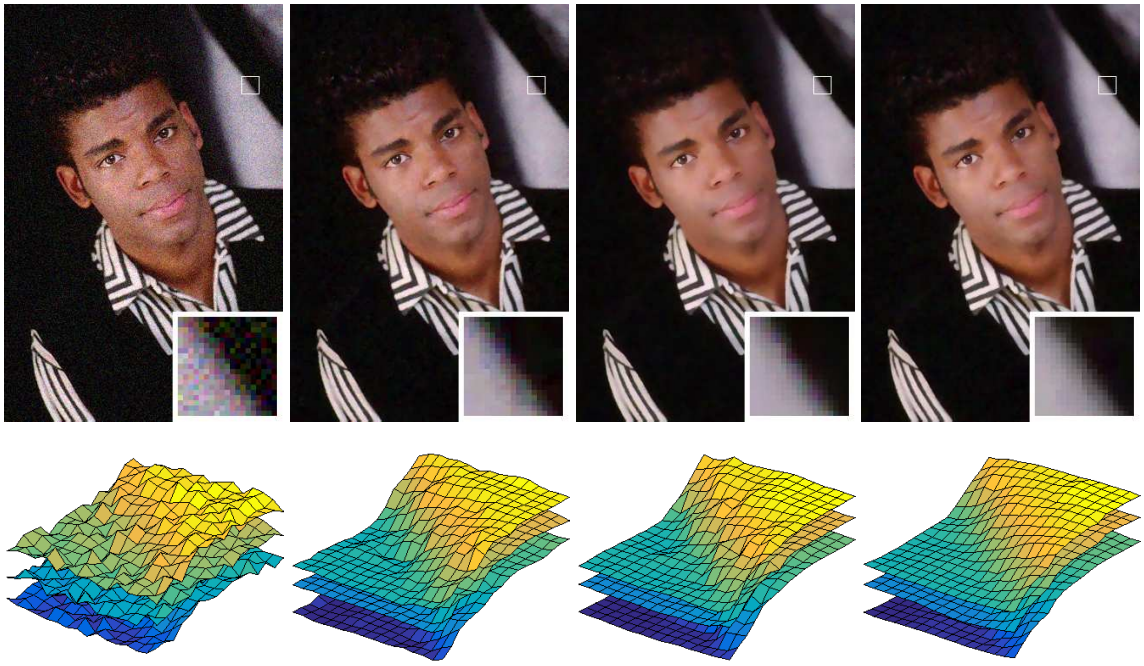


(a) Noisy image (b) Color TV (c) ALM (d) Our algorithm

Fig. 6 Gaussian denoising results and local surfaces for the 'Bird' image. (a) Noisy image corrupted by mean 0 and standard deviation 20; (b) denoising result by the color TV method (PSNR/SSIM: 31.73/0.9005); (c) denoising result by ALM (PSNR/SSIM: 32.01/0.9263); (d) denoising result by the proposed algorithm (PSNR/SSIM: 32.22/0.9298).

- Color TV method: the regularization parameter is selected from $\lambda \in [0.5, 15]$, and the penalty factor is set as $r \in [10, 50]$.
- Augmented Lagrangian method (ALM): the regularization parameter is set as $\lambda \in [0.01, 0.5]$, the parameter β is chosen from $[10, 100]$, and the penalty factor is selected as $r \in [0.1, 10]$.
- Operator-splitting method: the regularization parameter is adjusted in the range $\eta \in [0.5, 15]$, the parameter β is selected from $[1, 20]$, and the time step is set as $\tau \in [0.01, 0.05]$.

The parameters of all methods are appropriately chosen for the best restoration results as well as a fair comparison, the specific values of which are provided in each experiment. In particular, we discuss the effects of the three parameters, namlely η, β, τ, to the performance of our method as follows. The most important parameter in our algorithm is η, which is used to balance the contribution between the data fidelity and Beltrami regularization term. The smaller η is, the smoother the restoration is. If η is too large, the model fails to remove the noises and results in low-quality images. On the other hand, if η is too small, the restoration becomes over-smoothed and some features may be lost. The parameter β is used to balance the computational efficiency and reconstruction quality. Although we assume $\beta \gg 1$ should be held to obtain the simplified Beltrami minimization (3), β is usually chosen in interval $[1, 20]$. There are two reasons for that: a larger β will result in slow convergence and small β can produce satisfactory results. One of the main advantages of the proposed algorithm is that it only involves the time step τ as the free algorithmic parameter to be chosen. The time step τ affects the convergence speed and robustness of the proposed algorithm. A larger τ may lead to fast convergence and inferior restoration of the proposed method. Specifically, we use the parameter sweep method to find the optimal values for the parameters by fixing one parameter and exhaustive searching the other two parameters through a manually specified subset of the parameter space. Last but not least, the stopping criterion for the iterative procedure is given as $\epsilon = 1 \times 10^{-5}$ throughout all the experiments.



(a) Noisy image (b) Color TV (c) ALM (d) Our algorithm

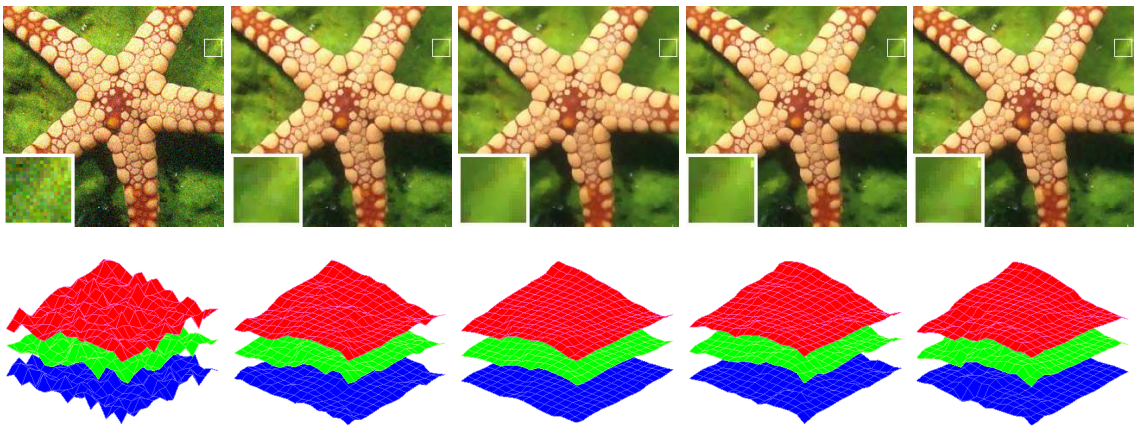
Fig. 7 Gaussian denoising results and local surfaces for the 'Man' image. (a) Noisy image corrupted by mean 0 and standard deviation 20; (b) denoising result by the color TV method (PSNR/SSIM: 31.34/0.8981); (c) denoising result by ALM (PSNR/SSIM: 31.80/0.9269); (d) denoising result by the proposed algorithm (PSNR/SSIM: 32.17/0.9327).

5.2 Gaussian noise removal

To demonstrate the effectiveness and efficiency of the proposed operator-splitting method, we first evaluate the performance on Gaussian noise removal. As shown in Figs. 1 and 4, the color images 'Lena' and 'Fruits' are contaminated by additive Gaussian noises with mean zero and standard deviation $\sigma = 20$ and $\sigma = 30$. The experiment-dependent parameters for our operator-splitting algorithm are set as $\eta = \{10, 6\}$ for $\sigma = \{20, 30\}$, $\tau = 0.02$ and $\beta = 10$ for the two noise levels, respectively.

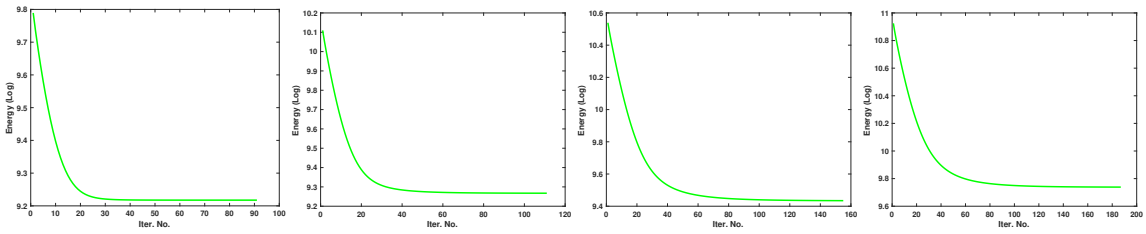
Both the denoising results and corresponding local magnification views are displayed in Figs. 1 and 4, where all the methods we compare can effectively remove noises. However, the color TV method suffers from unnatural staircase-like artifacts on slanted regions and loses some important image details (e.g., the brim and peel). On the contrary, both Beltrami-based methods can effectively remove the artifacts that do not fit the appearance of natural images and preserve slight image details. Table 1 reports the PSNR, SSIM, computational time and the number of iterations for each algorithm, where the highest PSNR and SSIM are always obtained by the proposed method for both test images. More importantly, our proposed algorithm converges much faster than the ALM, saving more than half of the computational costs, which verifies the advantage of the operator-splitting method.

We also tracked the evolution of PSNR and the relative error on $\boldsymbol{u}^k$ for both ALM and our proposed algorithm. As displayed in Fig. 2 and Fig. 5, our method can obtain higher and more stable PSNR with less CPU time and reach the predefined relative error with fewer iterations compared to the ALM based Beltrami minimization. Moreover, as reported in Fig. 3, we monitored the numerical energy changes of each subproblem and the original problem (39) on image 'Lena' corrupted by white Gaussian noise with mean zero and standard deviation $\sigma = 20$. As can be seen, the energies of the $\boldsymbol{p}$-subproblems $\{\boldsymbol{p}^{n+1/3}, \boldsymbol{p}^{n+2/3}, \boldsymbol{p}^{n+1}\}$ and $\boldsymbol{y}$-subproblems $\{\boldsymbol{y}^{n+1/3}, \boldsymbol{y}^{n+2/3}\}$ decrease to a stable value as the iteration number n increases. Pay attention that the total numerical energy decreases to a



(a) Noisy image (b) $\beta = 0$ (c) $\beta = 10.0$ (d) $\beta = 20.0$ (e) ALM

Fig. 8 Gaussian denoising results and local surfaces for the 'Starfish' image by the proposed algorithm for different values of β and ALM.



(a) $\beta = 0$ (b) $\beta = 10.0$ (c) $\beta = 20.0$ (d) ALM

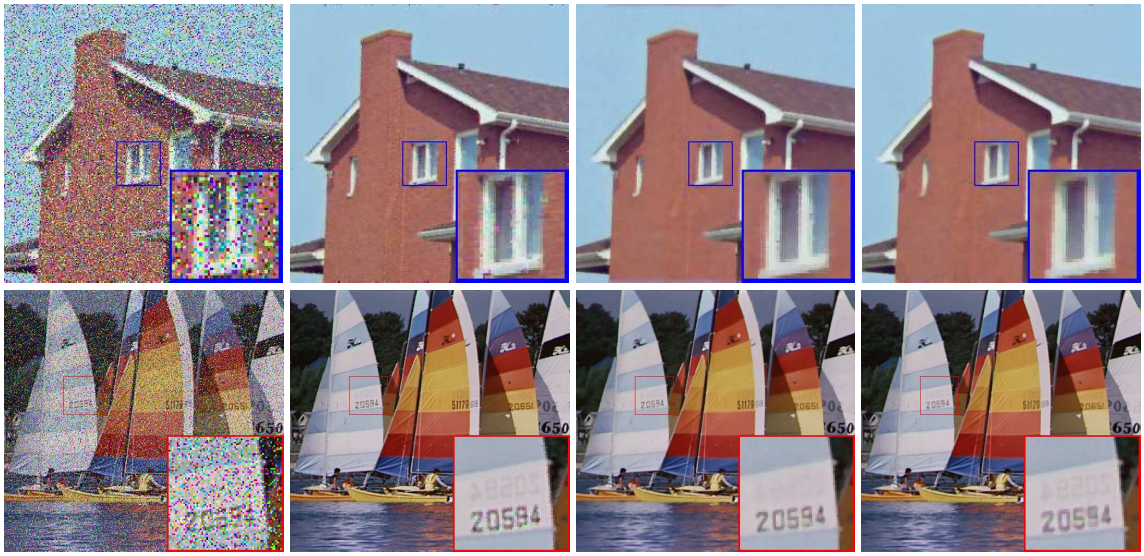
Fig. 9 Energy decays for the 'Starfish' image corrupted by Gaussian noises, where different values of β are used in the proposed algorithm.

higher value since it takes the fidelity term into account. These plots can demonstrate the convergence of the iterative process and the stability of the proposed algorithm.

To further verify the effectiveness of our proposed algorithm, we implemented the color surface denoising by comparing with the classical color TV and ALM of Beltrami regularization. The test images 'Bird' and 'Man' are degraded by Gaussian noises with mean zero and standard deviation $\sigma = 20$. The parameters in our method are set as $\eta = 8$, $\beta = 10$ and $\tau = 0.02$. We present the denoising results and local surfaces generated by the color TV, ALM and our operator-splitting method in Fig. 6 and Fig. 7. We see that, the color TV method suffers from obvious staircase artifacts resulting in the rough and uneven color surfaces, while the ALM presents certain oscillations near the edges and inhomogeneity in the slanted regions. On the other hand, our algorithm produces better restoration results with sharp edges and homogeneous color effects, which are also verified by the PSNR and SSIM comparison.

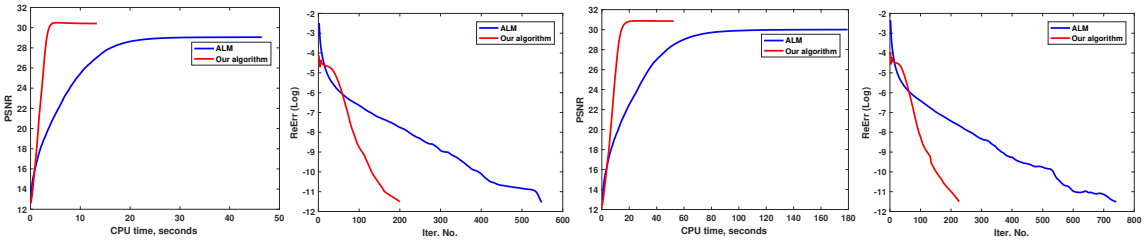
5.3 The effect of parameter β

In this subsection, we explore the effect of β with fixed parameters $\eta = 10$, $\tau = 0.02$ for Gaussian noise removal. The image 'Starfish' is corrupted by additive Gaussian noise with mean zero and standard deviation $\sigma = 20$, where β in our operator-splitting method is chosen as $\beta = \{0, 10.0, 20.0\}$ to evaluate the denoising performances. The noisy image and restoration results are displayed in the first row of Fig. 8. It is clear that our Beltrami minimization model reduces to the color TV minimization



(a) Noisy image ($\theta = 20\%$) (b) Color TV (c) ALM (d) Our algorithm

Fig. 10 Comparison results for 'House' (top) and 'Yacht' (bottom) corrupted by salt and pepper noises $\theta = 20\%$ between the color TV, augmented Lagrangian method and our algorithm for Bletrami minimization model. The parameters are set as: Color TV: $\lambda = 1.2$, $r_1 = 10$, $r_2 = 20$, ALM: $\lambda = 0.05$, $r = 0.5$, $\beta = \sqrt{4000}$, and our algorithm: $\eta = 0.8$, $r = 0.2$, $\tau = 0.04$, $\beta = 2.0$.



(a) PSNR('House') (b) ReErr('House') (c) PSNR('Yacht') (d) ReErr('Yacht')

Fig. 11 Comparison for PSNR and relative error on 'House' and 'Yacht' corrupted by the salt and pepper noises $\theta = 20\%$ between ALM and our operator-splitting method.

model for $\beta = 0$. As can be observed, the restoration tends to be increasingly smoother as β becomes larger in a certain degree, while the results become inhomogeneous and noise-retaining for a too large β. On the second row of Fig. 8, we display the surface plots of the zoomed region. The surfaces obtained by our model look much more homogeneous than the color TV model, which convinces that the alignment between channels is important for color image processing. Indeed, our results are even better than the ALM for minimizing the Beltrami regularization model because it is much easier to choose parameters in our approach. Besides, the energy decays obtained by different values of β are exhibited in Fig. 9, where all energies go down as the iteration number increases. Note that more iterations are consumed to reach the convergence for large values of β. Therefore, we usually choose β less than 20 in practice.

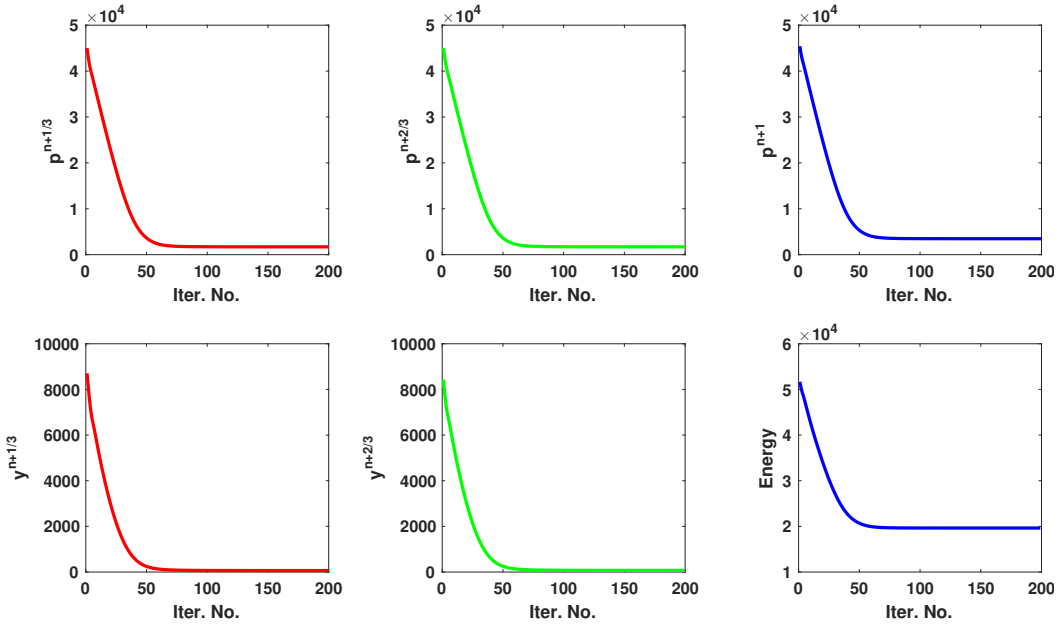


Fig. 12 Energy changes for each subproblem for salt and pepper denoising of 'House' ($\theta = 20\%$) by the proposed method. From left to right: the $\boldsymbol{p}^{n+1/3}$-subproblem, the $\boldsymbol{p}^{n+2/3}$-subproblem, the $\boldsymbol{p}^{n+1}$-subproblem, the $\boldsymbol{y}^{n+1/3}$-subproblem, the $\boldsymbol{y}^{n+2/3}$-subproblem and the total energy, respectively.

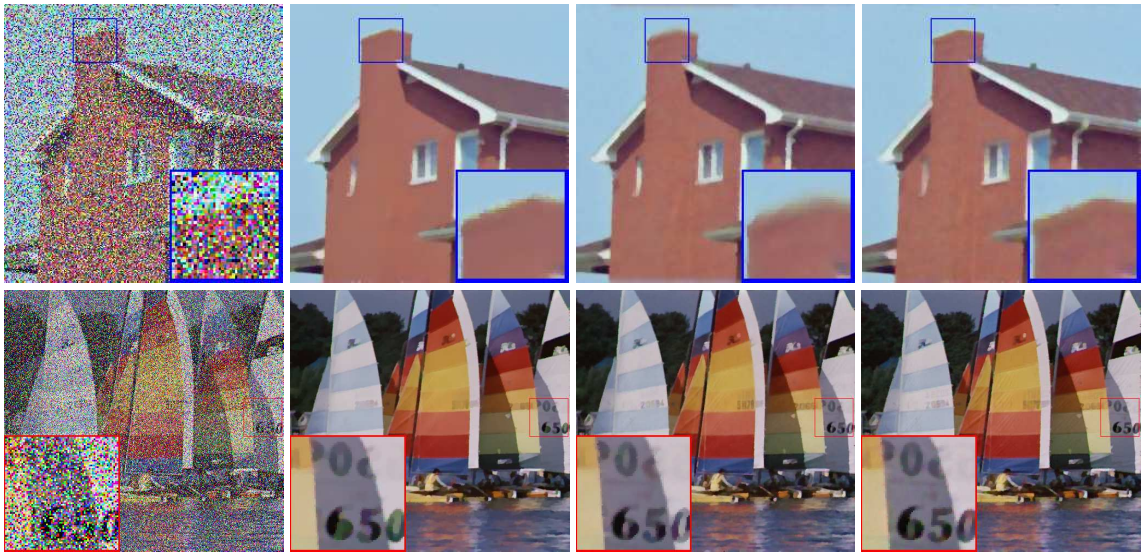
Table 2 The comparison of PSNR, SSIM, number of iterations, computational time and efficiency on color images corrupted by salt and pepper noises.

Noise level	Images	Methods	PSNR	SSIM	Iterations	Time(s)	Time/Iteration
$\theta = 20\%$	House (256 × 256)	Color TV	29.90	0.9654	262	**7.66**	**0.0292**
		ALM	29.42	0.9631	549	46.44	0.0846
		Our algorithm	**30.40**	**0.9688**	**200**	13.38	0.0669
	Yacht (512 × 480)	Color TV	30.34	0.9617	240	**30.29**	**0.1262**
		ALM	30.07	0.9416	742	178.67	0.2408
		Our algorithm	**30.85**	**0.9636**	**227**	52.56	0.2315
$\theta = 40\%$	House (256 × 256)	Color TV	27.56	0.9365	**260**	**7.40**	**0.0285**
		ALM	27.02	0.9302	615	51.46	0.0837
		Our algorithm	**27.88**	**0.9418**	288	19.19	0.0666
	Yacht (512 × 480)	Color TV	25.22	0.8779	**234**	**29.10**	**0.1244**
		ALM	25.44	0.8823	722	174.03	0.2410
		Our algorithm	**25.66**	**0.8955**	296	68.72	0.2322

5.4 Salt and pepper noise removal

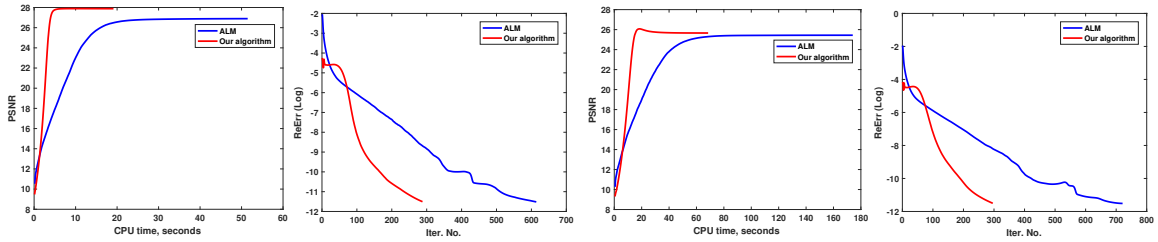
We implemented the color TV minimization model [54], Beltrami regularization model [35], and our operator splitting method on another two color images 'House' and 'Yacht', which are degraded by the salt and pepper noises with $\theta = \{20\%, 40\%\}$. The parameters in our method are set as $\eta = \{0.8, 0.6\}$ for $\theta = \{20\%, 40\%\}$, $\tau = 0.04$ and $\beta = 2.0$ for the two noise levels, respectively.

The noisy images and restoration results are displayed in Fig. 10 and Fig. 13. Similarly, we obverse that the images obtained by color TV model exhibit obvious staircase effect and some remaining noises, especially on the edges. The ALM for Beltrami minimization tends to provide over-smoothed recovery results with fine details missing, such as the digital watermarking and edges of the chimney. By contrast, the proposed operator splitting method produces better restoration results with clear watermarking and sharp edges. The associated PSNR, SSIM and computational time are reported in Table 2, which agree with the visual results. Our proposed algorithm gives the highest PSNR and



(a) Noisy($\theta = 40\%$) (b) Color TV (c) ALM (d) Proposed

Fig. 13 Comparison results for 'House' (top) and 'Yacht' (bottom) corrupted by salt and pepper noises $\theta = 40\%$ among the color TV, augmented Lagrangian method and our algorithm for Bletrami minimization model. The parameters are set as: Color TV: $\lambda = 0.6$, $r_1 = 10$, $r_2 = 20$, ALM: $\lambda = 0.03$, $r = 0.5$, $\beta = \sqrt{4000}$, and our algorithm: $\eta = 0.6$, $r = 0.2$, $\tau = 0.04$, $\beta = 2.0$.



(a) PSNR('House') (b) ReErr('House') (a) PSNR('Yacht') (b) ReErr('Yacht')

Fig. 14 Comparison for PSNR and relative error on 'House' and 'Yacht' corrupted by the salt and pepper noises $\theta = 40\%$ between ALM and our operator-splitting method.

SSIM values, which illustrate the importance of modeling the couplings between the channels and the ease of parameter selection.

Furthermore, we monitored the PSNR value against running time and relative error decay against the number of iterations in Fig. 11 and Fig. 14. Compared to the augmented Lagrangian method, our proposed algorithm always reaches the stopping criteria with a significantly fewer number of iterations. Fig. 12 shows the energy decays, where all energies decrease to a stable value as the iteration number n increases. Thus, we conclude that the proposed method is more effective and efficient than the ALM based Beltrami minimization model in removing salt and pepper noises.

5.5 Comparison on dataset

In this subsection, we further compare the denoising performance of the color TV, ALM and our algorithm on a dataset consisting of 20 color images. As shown in Fig. 15, the sizes of images from the first row to fourth row images are: 256×256, 321×481, 480×512 and 512×512, respectively.

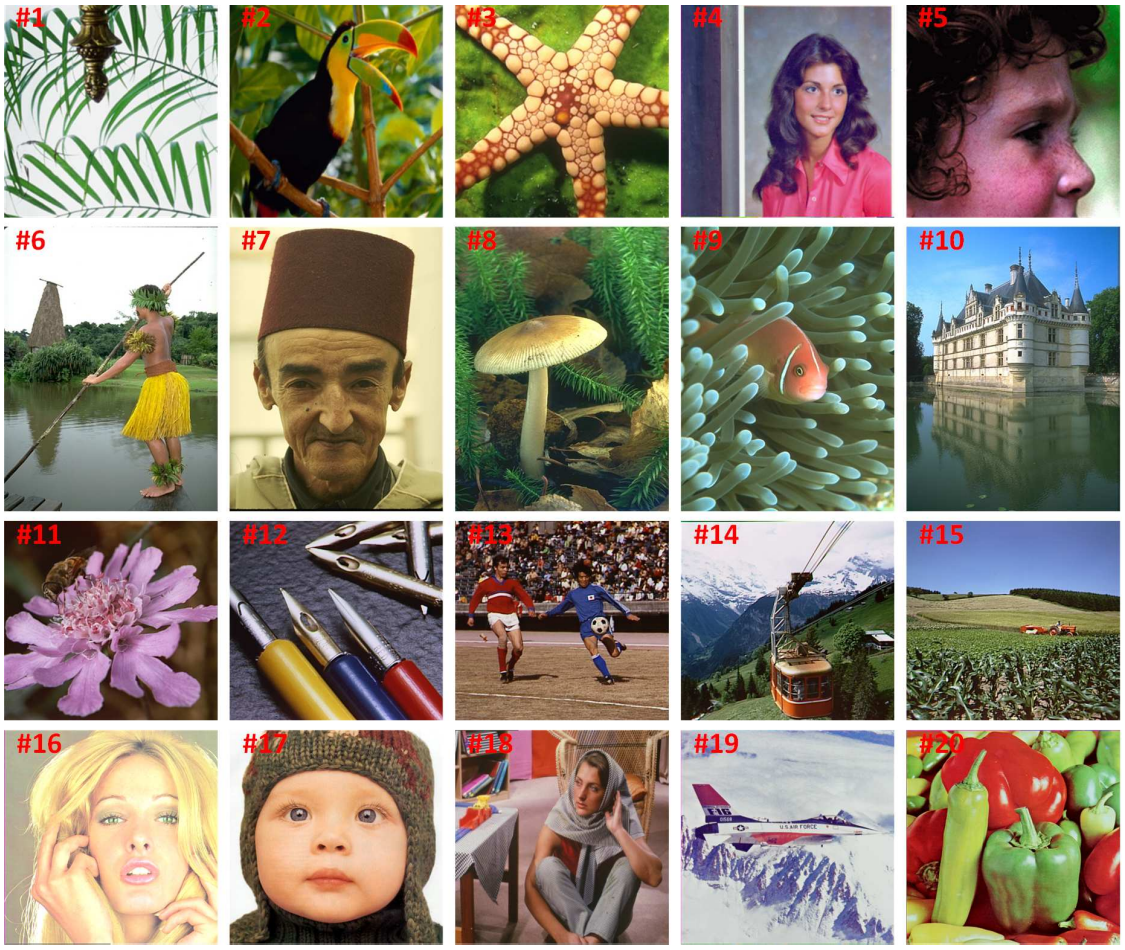
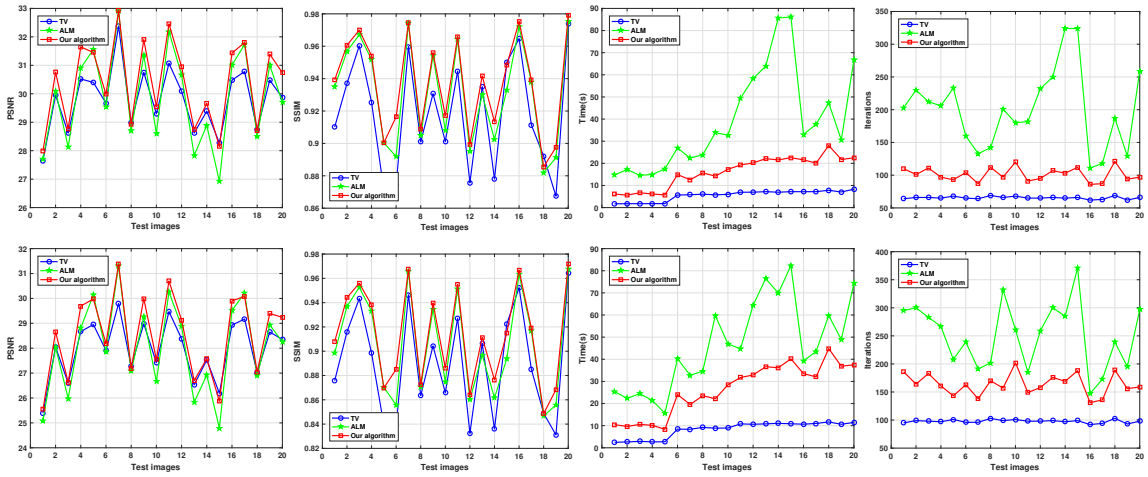


Fig. 15 Display of the 20 color test images for comparing the color TV, augmented Lagrangian method and our algorithm for Bletrami minimization model. The size of images are set as: the 1st row of 256×256, the 2nd row of 321×481, the 3rd row of 480×512 and the 4th row of 512×512, respectively.

The Gaussian noises of zero mean and variance $\sigma = \{20, 30\}$ are added into clean images. We track the PSNR, SSIM, CPU time and iterations on 20 test images as displayed in Fig. 16, where the averaged PSNR, SSIM, number of iterations, total computational time and time per iteration are recorded in Table 3. As can be observed, our operator-splitting method achieves the highest values of both PSNR and SSIM. Meanwhile, our method consumes less computational time and iterations than ALM, which verify the superiority and effectiveness of the operator-splitting strategy. Moreover, we introduce the salt and peppers noises with noise level $\theta = \{20\%, 40\%\}$ to the test images. The comparison results are exhibited in Fig. 17 and Table 4, which are completely in accord with the case of Gaussian noises. Thus, we arrive at the conclusion that our operator-splitting algorithm is more effective for color image denoising problems, providing better restoration quality in reasonable time.

6 Conclusion

We proposed an effective and efficient operator-splitting method to deal with the Beltrami minimization model for color image denoising problems. We introduced two auxiliary variables and



(a) PSNR (b) SSIM (c) Time(s) (d) Iterations

Fig. 16 Comparison results on 20 test images in Fig. 15 corrupted by Gaussian noises $\sigma = 20$ (top) and $\sigma = 30$ (bottom) among the color TV (blue curves), augmented Lagrangian method (green curves) and our algorithm (red curves). The parameters for $\sigma = \{20, 30\}$ are set as: color TV: $\lambda = \{15, 9\}$ and $r = 40$, ALM: $\lambda = \{0.4, 0.25\}$ and $\beta = \sqrt{4000}$, and our algorithm: $\eta = \{10, 6\}$, $\tau = 0.02$ and $\beta = 10$.

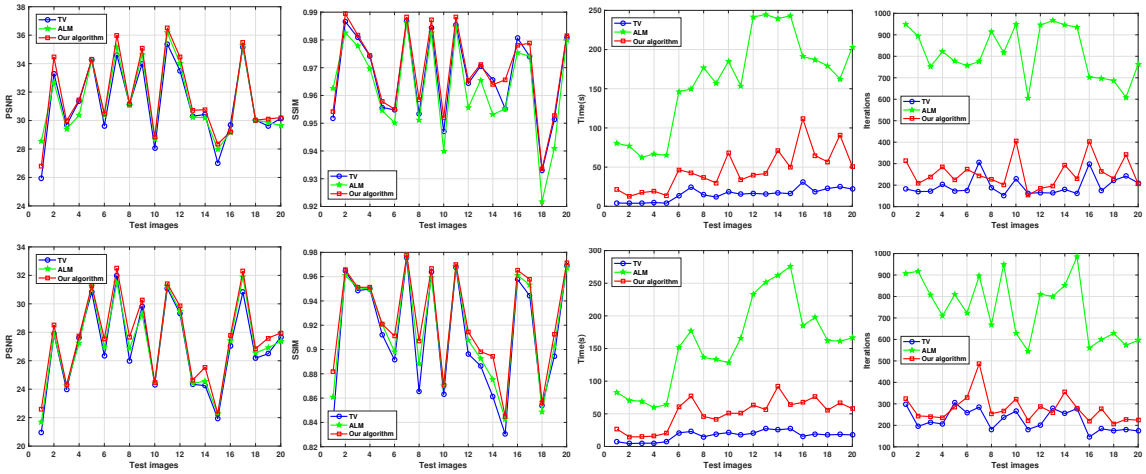
Table 3 The comparison among color TV, ALM and our method on the averaged PSNR, SSIM, number of iterations, computational time and time per iteration of the 20 test images in Fig. 15 corrupted by Gaussian noises.

Noise level	Methods	PSNR	SSIM	Iterations	Time(s)	Time/Iteration
$\sigma = 20$	Color TV	29.80	0.9182	**66**	**5.55**	**0.0849**
	ALM	29.90	0.9314	201	38.88	0.1946
	Our algorithm	**30.39**	**0.9371**	101	16.23	0.1617
$\sigma = 30$	Color TV	27.96	0.8899	**98**	**8.34**	**0.0854**
	ALM	28.04	0.9053	251	46.36	0.1878
	Our algorithm	**28.52**	**0.9132**	164	26.47	0.1631

Table 4 The comparison among color TV, ALM and our method on the averages PSNR, SSIM, number of iterations, computational time and time per iteration of 20 test images in Fig. 15 corrupted by salt and pepper noises.

Noise level	Methods	PSNR	SSIM	Iterations	Time(s)	Time/Iteration
$\theta = 20\%$	Color TV	31.17	0.9668	**196**	**15.18**	**0.0758**
	ALM	31.39	0.9632	813	160.60	0.1998
	Our algorithm	**31.72**	**0.9689**	256	45.98	0.1772
$\theta = 40\%$	Color TV	26.95	0.9118	**225**	**16.73**	**0.0767**
	ALM	27.17	0.9178	748	156.79	0.2206
	Our algorithm	**27.65**	**0.9245**	278	50.95	0.1868

reformulated the original model into an equivalent form, which was solved by a time discretization problem using an operator-splitting method. Compared to the existing augmented Lagrangian method, the proposed algorithm contained only one time step parameter, which was much easier to be implemented in practice. Plentiful numerical experiments were provided to demonstrate the superiority of the proposed algorithm, which was shown more efficient and effective than the ALM based algorithm, and outperformed the color TV model in restoration quality. The proposal has a wide potential in other color image processing problems, such as image deblurring, image inpainting, image super-resolution, etc.



(a) PSNR (b) SSIM (c) Time(s) (d) Iterations

Fig. 17 Comparison results on 20 test images in Fig. 15 corrupted by salt and pepper noises $\theta = 20\%$ (top) and $\theta = 40\%$ (bottom) among the color TV (blue curves), augmented Lagrangian method (green curves) and our method (red curves). The parameters for $\theta = \{20\%, 40\%\}$ are set as: color TV: $\lambda = \{1.2, 0.6\}$, $r_1 = 10$ and $r_2 = 20$, ALM: $\lambda = \{0.05, 0.03\}$, $r = 0.5$ and $\beta = \sqrt{4000}$, and our method: $\eta = \{1.0, 0.6\}$, $r = 0.2$, $\tau = 0.04$ and $\beta = 2$.

Appendix

Now we provide a brief introduction on the Lie and Marchuk-Yanenko schemes for the time-discretization of initial value problems. Consider the following *steady-state problem*

$$A(\boldsymbol{X}) = 0, \tag{40}$$

where the operator A maps the vector space $\boldsymbol{V}$ into itself. The classical method to solve (40) is to formulated it as the following initial value problem

$$\begin{cases} \frac{\partial \boldsymbol{X}}{\partial t} + A(\boldsymbol{X}) \ni 0 \quad \text{on} \quad (0, T) \text{ (with } 0 < T \leq +\infty), \\ \boldsymbol{X}(0) = \boldsymbol{X}_0. \end{cases} \tag{41}$$

Suppose that the problem (41) has steady-state solutions. These solutions are necessarily solutions of problem (40). We further assume the operator A has a nontrivial decomposition, namely

$$A = \sum_{j=1}^{J} A_j, \tag{42}$$

where $J \geq 2$ and all $A_j, 0 \leq j \leq J$, are individually simpler than A. Let $\tau > 0$ be a time-discretization step, and denote $n\tau$ by t^n. Assuming that $\boldsymbol{X}^n$ is the approximation of $\boldsymbol{X}(t^n)$, the Lie scheme for solving (41) is given by (see Chapter 6 of [55] for its derivation)

$$\boldsymbol{X}^0 = \boldsymbol{X}_0, \tag{43}$$

then, for $n \geq 0$, $\boldsymbol{X}^n \to \boldsymbol{X}^{n+1}$ are updated as follows

$$\begin{cases} \frac{\partial \boldsymbol{X}_j}{\partial t} + A_j(\boldsymbol{X}_j) = 0 \ \text{ on } \ (t^n, t^{n+1}), \\ \boldsymbol{X}_j(t^n) = \boldsymbol{X}^{n+(j-1)/J}, \ \boldsymbol{X}^{n+j/J} = \boldsymbol{X}_j(t^{n+1}), \end{cases} \quad \text{for } \ j = 1, \cdots, J. \tag{44}$$

The Lie scheme (43) and (44) is only semiconstructive since it requires to solve a series of initial value problems. Therefore, by discretizing the subproblems (44) using one step of backward Euler scheme, we obtain the following Marchuk-Yanenko scheme

$$\frac{\boldsymbol{X}^{n+j/J} - \boldsymbol{X}^{n+(j-1)/J}}{\tau} + A_j(\boldsymbol{X}^{n+j/J}) = 0, \quad \text{for } \ j = 1, \cdots, J. \tag{45}$$

Due to asymptotic properties of the Lie and Marchuk-Yanenko schemes, the following phenomena generically holds [39]

- If converging for $j = 1, \ldots, J$, the sequences $(\boldsymbol{X}^{n+j/J})_{n\geq 0}$ converge to different limits than the solution of (40) with the distances between them being $\mathcal{O}(\tau)$ at best.
- None of the above limits is a steady-state solution, but their distance to a steady-state solution converges to 0 as $\tau \to 0$ (if a steady-state solution does exist).

The above convergence theory has been proved in Chapter 6 of [55] supposing the space $\boldsymbol{V}$ being finite dimensional and the operators A_j being affine. Since the properties for A_j do not hold in our Beltrami minimization model, the existing convergence theory cannot be applied to our algorithm, which need to be studied separately as our future works. Moreover, when the Lie or Marchuk-Yanenko scheme applies to multivalued operators such as the subdifferential of proper, lower semicontinuous, convex functionals, the first order accuracy is not guaranteed as well (our case in Sect. 3).

Acknowledgments

This paper is dedicated to the memory of our dear co-worker Prof. Roland Glowinski, who passed away while this paper was being peer-reviewed. The authors would like to thank Dr. Liangjian Deng for sharing the MATLAB code of Lie scheme based operator splitting method [39]. The work was supported by National Natural Science Foundation of China (NSFC 12071345, 11701418), Major Science and Technology Project of Tianjin 18ZXRHSY00160 and Recruitment Program of Global Young Expert. The work was also supported by projects HKBU 12300819, NSF/RGC Grant N-HKBU214-19, ANR/RGC Joint Research Scheme (A-HKBU203-19) and RC-FNRA-IG/19-20/SCI/01.

References

[1] Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3-D transform-domain collaborative filtering. IEEE Transactions on image processing **16**(8), 2080–2095 (2007)

[2] Mairal, J., Elad, M., Sapiro, G.: Sparse representation for color image restoration. IEEE Transactions on image processing **17**(1), 53–69 (2007)

[3] Xu, J., Zhang, L., Zhang, D., Feng, X.: Multi-channel weighted nuclear norm minimization for real color image denoising. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1096–1104 (2017)

[4] Kim, S.: PDE-based image restoration: A hybrid model and color image denoising. IEEE Transactions on Image Processing **15**(5), 1163–1170 (2006)

[5] Lefkimmiatis, S.: Non-local color image denoising with convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3587–3596 (2017)

[6] Deng, G., Cahill, L.: An adaptive gaussian filter for noise reduction and edge detection. In: IEEE Conference Record Nuclear Science Symposium and Medical Imaging Conference, pp. 1615–1619 (1993)

[7] Chen, T., Ma, K.-K., Chen, L.-H.: Tri-state median filter for image denoising. IEEE Transactions on Image processing **8**(12), 1834–1838 (1999)

[8] Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: Sixth IEEE International Conference on Computer Vision, pp. 839–846 (1998)

[9] Kazubek, M.: Wavelet domain image denoising by thresholding and wiener filtering. IEEE Signal Processing Letters **10**(11), 324–326 (2003)

[10] Buades, A., Coll, B., Morel, J.-M.: A non-local algorithm for image denoising. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 60–65 (2005)

[11] Gong, Y., Sbalzarini, I.F.: Curvature filters efficiently reduce certain variational energies. IEEE Transactions on Image Processing **26**(4), 1786–1798 (2017)

[12] Tschumperle, D., Deriche, R.: Diffusion PDEs on vector-valued images. IEEE Signal Processing Magazine **19**(5), 16–25 (2002)

[13] Tschumperlé, D., Deriche, R.: Vector-valued image regularization with PDEs: A common framework for different applications. IEEE transactions on pattern analysis and machine intelligence **27**(4), 506–517 (2005)

[14] Barbu, T., Barbu, V., Biga, V., Coca, D.: A PDE variational approach to image denoising and restoration. Nonlinear Analysis: Real World Applications **10**(3), 1351–1361 (2009)

[15] Bettahar, S., Stambouli, A.B., Lambert, P., Benoit, A.: PDE-based enhancement of color images in RGB space. IEEE transactions on image processing **21**(5), 2500–2512 (2011)

[16] Tschumperlé, D.: Fast anisotropic smoothing of multi-valued images using curvature-preserving PDE's. International Journal of Computer Vision **68**(1), 65–82 (2006)

[17] Shan, X., Sun, J., Guo, Z.: Multiplicative noise removal based on the smooth diffusion equation. Journal of Mathematical Imaging and Vision **61**(6), 763–779 (2019)

[18] Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D: Nonlinear Phenomena **60**(1-4), 259–268 (1992)

[19] Blomgren, P., Chan, T.F.: Color TV: Total variation methods for restoration of vector-valued images. IEEE transactions on image processing **7**(3), 304–309 (1998)

[20] Goldluecke, B., Strekalovskiy, E., Cremers, D.: The natural vectorial total variation which arises from geometric measure theory. SIAM Journal on Imaging Sciences **5**(2), 537–563 (2012)

[21] Ono, S., Yamada, I.: Decorrelated vectorial total variation. In: Proceedings of the IEEE

Conference on Computer Vision and Pattern Recognition, pp. 4090–4097 (2014)

[22] Lefkimmiatis, S., Roussos, A., Maragos, P., Unser, M.: Structure tensor total variation. SIAM Journal on Imaging Sciences **8**(2), 1090–1122 (2015)

[23] Jia, Z., Ng, M.K., Wang, W.: Color image restoration by saturation-value total variation. SIAM Journal on Imaging Sciences **12**(2), 972–1000 (2019)

[24] Batard, T., Bertalmío, M.: On covariant derivatives and their applications to image regularization. SIAM Journal on Imaging Sciences **7**(4), 2393–2422 (2014)

[25] Batard, T., Sochen, N.: A class of generalized laplacians on vector bundles devoted to multichannel image processing. Journal of mathematical imaging and vision **48**(3), 517–543 (2014)

[26] Sochen, N., Kimmel, R., Malladi, R.: A general framework for low level vision. IEEE transactions on image processing **7**(3), 310–318 (1998)

[27] Sochen, N.A., Gilboa, G., Zeevi, Y.Y.: Color image enhancement by a forward-and-backward adaptive Beltrami flow. In: International Workshop on Algebraic Frames for the Perception-Action Cycle, pp. 319–328 (2000)

[28] Kaftory, R., Sochen, N., Zeevi, Y.: Color image denoising and blind deconvolution using the Beltrami operator. In: International Symposium on Image and Signal Processing and Analysis, vol. 1, pp. 1–4 (2003)

[29] Bresson, X., Vandergheynst, P., Thiran, J.-P.: Multiscale active contours. International Journal of Computer Vision **70**(3), 197–211 (2006)

[30] Dou, Z., Han, Y., Sheng, W., Ma, X.: Image dehaze using alternating Laplacian and Beltrami regularizations. Journal of Electronic Imaging **24**(2), 023004 (2015)

[31] Kimmel, R., Sochen, N., Malladi, R.: From high energy physics to low level vision. In: International Conference on Scale-Space Theories in Computer Vision, pp. 236–247 (1997)

[32] Spira, A., Kimmel, R., Sochen, N.: A short-time Beltrami kernel for smoothing images and manifolds. IEEE Transactions on Image Processing **16**(6), 1628–1636 (2007)

[33] Rosman, G., Dascal, L., Sidi, A., Kimmel, R.: Efficient Beltrami image filtering via vector extrapolation methods. SIAM Journal on Imaging Sciences **2**(3), 858–878 (2009)

[34] Rosman, G., Dascal, L., Tai, X.-C., Kimmel, R.: On semi-implicit splitting schemes for the Beltrami color image filtering. Journal of Mathematical Imaging and Vision **40**(2), 199–213 (2011)

[35] Rosman, G., Tai, X.-C., Dascal, L., Kimmel, R.: Polyakov action minimization for efficient color image processing. In: Proceedings of the Springer European Conference on Computer Vision, pp. 50–61 (2010)

[36] Wang, W., Ng, M.K.: Color image restoration by saturation-value total variation regularization on vector bundles. SIAM Journal on Imaging Sciences **14**(1), 178–197 (2021)

[37] Glowinski, R., Pan, T.-W., Tai, X.-C.: Some facts about operator-splitting and alternating direction methods. In: Splitting Methods in Communication, Imaging, Science, and Engineering, pp.

19–94 (2016)

[38] Glowinski, R., Osher, S.J., Yin, W.: Splitting Methods in Communication, Imaging, Science, and Engineering. Springer, Berlin (2017)

[39] Deng, L.-J., Glowinski, R., Tai, X.-C.: A new operator splitting method for the Euler elastica model for image smoothing. SIAM Journal on Imaging Sciences **12**(2), 1190–1230 (2019)

[40] Liu, H., Tai, X.-C., Kimmel, R., Glowinski, R.: A color elastica model for vector-valued image regularization. SIAM Journal on Imaging Sciences **14**(2), 717–748 (2021)

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

[42] Rudin, L., Lions, P.-L., Osher, S.: Multiplicative denoising and deblurring: theory and algorithms. In: Geometric Level Set Methods in Imaging, Vision, and Graphics, pp. 103–119. Springer, ??? (2003)

[43] Glowinski, R., Luo, S., Tai, X.-C.: Fast operator-splitting algorithms for variational imaging models: Some recent developments. In: Handbook of Numerical Analysis vol. 20, pp. 191–232 (2019)

[44] MacNamara, S., Strang, G.: Operator splitting. In: Splitting Methods in Communication, Imaging, Science, and Engineering, pp. 95–114 (2016)

[45] Liu, H., Tai, X.-C., Glowinski, R.: An operator-splitting method for the gaussian curvature regularization model with applications in surface smoothing and imaging. arXiv preprint, arXiv:2108.01914 (2021)

[46] Koczka, G., Auberhofer, S., Biro, O., Preis, K.: Optimal convergence of the fixed-point method for nonlinear eddy current problems. IEEE Transactions on Magnetics **45**(3), 948–951 (2009)

[47] Amiri, A., Cordero, A., Darvishi, M.T., Torregrosa, J.R.: Preserving the order of convergence: Low-complexity jacobian-free iterative schemes for solving nonlinear systems. Journal of Computational and Applied Mathematics **337**, 87–97 (2018)

[48] Biazar, J., Ghanbary, B.: A new technique for solving systems of nonlinear equations. Applied Mathematical Sciences **2**(55), 2699–2703 (2008)

[49] Kelley, C.T.: Solving Nonlinear Equations with Newton's Method. SIAM, Philadelphia (2003)

[50] Abbasbandy, S., Ezzati, R.: Newtons method for solving a system of fuzzy nonlinear equations. Applied Mathematics and Computation **175**(2), 1189–1199 (2006)

[51] Brigham, E.O.: The Fast Fourier Transform and Its Applications. Prentice-Hall, New Jersey (1988)

[52] Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004)

[53] Yang, J., Yin, W., Zhang, Y., Wang, Y.: A fast algorithm for edge-preserving variational multichannel image restoration. SIAM Journal on Imaging Sciences **2**(2), 569–592 (2009)

[54] Yang, J., Zhang, Y., Yin, W.: An efficient TVL1 algorithm for deblurring multichannel images corrupted by impulsive noise. SIAM Journal on Scientific Computing **31**(4), 2842–2865 (2009)

[55] Glowinski, R.: Finite element methods for incompressible viscous flow. Handbook of numerical analysis **9**, 3–1176 (2003)