Efficient SAV Algorithms for Curvature Minimization Problems

Chenxin Wang[1], Zhenwei Zhang[1], Zhichang Guo, Tieyong Zeng, and Yuping Duan*

***Abstract*—The curvature regularization method is well-known for its good geometric interpretability and strong priors in the continuity of edges, which has been applied to various image processing tasks. However, due to the non-convex, non-smooth, and highly non-linear intrinsic limitations, most existing algorithms lack a convergence guarantee. This paper proposes an efficient yet accurate scalar auxiliary variable (SAV) scheme for solving both mean curvature and Gaussian curvature minimization problems. The SAV-based algorithms are shown unconditionally energy diminishing, fast convergent, and very easy to be implemented for different image applications. Numerical experiments on noise removal, image deblurring, and single image super-resolution are presented on both gray and color image datasets to demonstrate the robustness and efficiency of our method. Source codes are made publicly available at *https://github.com/Duanlab123/SAV-curvature*.**

***Index Terms*—Mean curvature, Gaussian curvature, scalar auxiliary variable, energy convergent, image denoising, image super-resolution, image deblurring**

I. Introduction

MODEL based methods are important tools for various image processing tasks, which are well-known for the good interpretability and mathematical properties. The Total Variation (TV) regularization proposed by Rudin, Osher and Fatemi [1] has been proven its success in different image application by removing noises as well as preserving sharp edges [2]–[5]. However, TV has the stair-casing effect by losing the repeated tinny image details and textures. To capture fine image structures, curvature regularization has been investigated and intensively studied for image processing problems.

Curvature as an important geometric concept can depict the amount of curves from being straight as in the case of a line or a surface deviating from being a flat plane, which has been successfully used in various computer vision tasks, such as structure estimation [6], graph embedding [7], classification [8], and deep learning methods [9], [10] etc. The curvature regularization was successfully used as the regularization term for various image processing tasks to overcome the drawbacks of TV, such as Euler's elastica [11], mean curvature [12], Gaussian curvature [13], and total curvature [14] etc. Specifically, Zhu and Chan [12] employed the L^1-norm of mean curvature of the image surface for image denoising, which is shown to be able to keep image contrast, edges and corners of objects. To explore the performance of Gaussian curvature, Brito-Loeza, Chen and Uc-Cetina [15] minimized the L^1-norm of Gaussian curvature for image denoising and theoretically verified its ability in preserving image contrast and sharp edges. Zhong, Yin and Duan [13] proposed a fast method to estimate the discrete curvatures in the local neighborhood according to differential geometry theory, and minimized certain functions of curvatures for image restoration task.

Although the intuition to employ curvature as regularization is straightforward, it is difficult to optimize the curvature energies since they are non-convex, non-smooth, and highly non-linear optimization problems. The existing numerical optimization methods for solving the curvature related energy minimization problems can be roughly divided into two categories: gradient descend and operator splitting method. The gradient descent approach can provide an accurate solution by solving the time dependent PDE system. Originally, the gradient descent method [12] was presented to solve mean curvature minimization model, which had to solve fourth-order nonlinear evolution equations. Gong [16] developed the gradient descent method to solve the mean curvature minimization model, where mean curvature was estimated locally in a discrete manner. However, the gradient descent suffers the numerical instability and low computational efficiency. Recently, the operator splitting and alternating direction method of multipliers become popular to solve the high-order models. Zhu, Tai and Chan utilized augmented Lagrangian method to develop efficient algorithms for solving the mean curvature model [17], [18]. Zhong, Yin and Duan [13] proposed a fast method to estimate discrete curvature and used the alternating direction method of multipliers to solve the mean curvature and Gaussian curvature minimization problems. Liu, Tai and Glowinski [24] solved the Gaussian curvature based model by using the operator-splitting method and Lie scheme. However, the aforementioned methods all lack the convergence guarantee due to the non-convexity and non-smoothness of the curvature minimization problems. Although some convergence results for the alternating direction method of multipliers

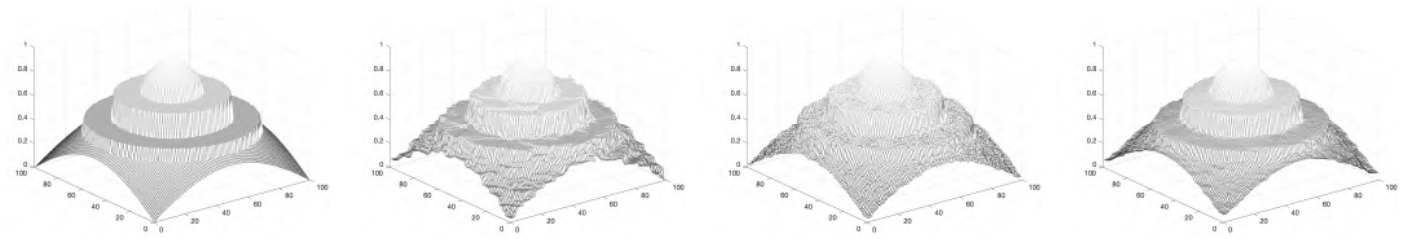
The work was supported by the National Natural Science Foundation of China (NSFC 12071345, 11701418), the Major Science and Technology Project of Tianjin (18ZXRHSY00160), and the Recruitment Program of Global Young Expert. The first two authors are equally contributed. The third author Z Guo was supported by the National Natural Science Foundation of China (12171123, 11971131, U21B2075), the Fudament Research Funds for the Gentral Universities (Grant NSRIF. 2020081), the Natural Sciences Foundation of Heilongjiang Province (ZD2022A001), China Society of Industrial and Applied Mathematics Young Women Applied Mathematics Support Research Project. *Asterisk indicates the corresponding author.*

C. Wang is with Department of Mathematics, Hong Kong Baptist University, Hong Kong, China.

Z. Zhang and Y. Duan are with Center for Applied Mathematics, Tianjin University, Tianjin, China. E-mail: yuping.duan@tju.edu.cn.

Z. Guo is with School of Mathematics, Harbin Institute of Technology, Harbin, China.

T. Zeng is with the Center for Mathematical Artificial INtelligence, Department of Mathematics, The Chinese University of Hong Kong, Hong Kong, China.



(a) Clean Image (b) Total Variation (c) Total Fractional-order Variation (d) Our model

Fig. 1. The image surfaces of the clean image and reconstructed images by the TV regularization, total fractional-order variation [22] and our model for the image 'Circle4'.

based algorithms were obtained in [20], [21] by assuming the subjectiveness of the linear mapping in the composite non-convex term, the curvature minimization model obviously does not meet these assumptions.

In this work, we reformulate the mean curvature and Gaussian curvature models by replacing the nonlinear area term by the length term, which can reduce the computational complexity of the curvature term without sacrificing the image qualities. We then develop a novel numerical scheme for minimizing the mean curvature and Gaussian curvature energies based on the Scalar Auxiliary Variable (SAV) method, which is shown much more efficient than the Alternating Direction Method of Multipliers (ADMM) used in [13]. To the best of our knowledge, it is the first attempt to build up the unconditionally energy diminishing algorithm for the mean and Gaussian curvature minimization problems. As shown by the surface plots in Fig.1, our method has obvious advantages in preserving edges and eliminating the staircase effect, yielding much satisfactory results than total variation model and the recent total fractional-order variation model [22]. To sum up, we conclude the main contributions as follows:

1) We develop an efficient SAV algorithm for solving the general curvature minimization problems, where both mean curvature and Gaussian curvature are regarded as a re-weighted total variation (TV) minimization problem to avoid the calculation of higher derivatives.
2) We theoretically prove the energy convergence of the SAV method for curvature minimization problems. Unlike existing methods such as gradient method [12], ALM [17], [18], [23], ADMM [13] and operator splitting method [19], our SAV scheme enjoys the advantages of unconditionally energy diminishing and the first-order convergence.
3) We extend the applications of curvature regularization method for various image processing tasks. The performance of our model are evaluated by four representative image restoration tasks on Gaussian denoising, mixed noise removal, image deblurring and single image super-resolution on both gray and color image datasets, which demonstrate the advantages of the proposed model in terms of accuracy and efficiency.

The rest of this paper is organized as follows. Sect. II dedicates to review the estimation of curvatures and the SAV approach. Sect. III introduces our proposed model and the associated SAV algorithm in detail. We present the numerical experiments to verify the efficiency and superiority of the proposed method upon image denoising, single image super-resolution, image deblurring and mixed noise removal in Sect. IV. The concluding remarks and possible future works are summarized in Sect. V.

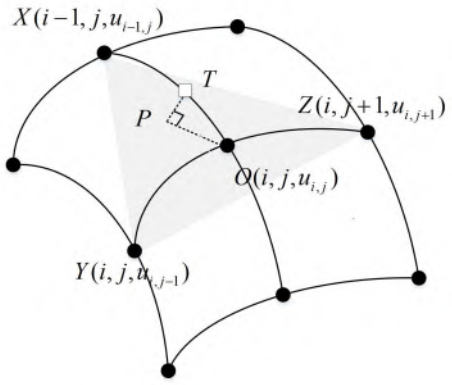


Fig. 2. Illustration of the distance to the tangent plane.

II. Preliminaries

A. Curvature estimation

Given a domain $\Omega \subset \mathbb{R}^2$, $u : \Omega \to \mathbb{R}$ is an image defined on Ω. Suppose the associated image surface for a 2D image is characterized by $\mathcal{S} = (x, u(x))$ for $x \in \Omega$. It is well-known that the *normal curvature* at a point of surface $\mathcal{S}$ can be defined by the quotient of *the second fundamental form* and *the first fundamental form*, i.e.,

$$\kappa_\ell = \frac{\mathrm{II}}{\mathrm{I}} \approx \frac{2d_\ell}{ds^2} \tag{1}$$

where II can be estimated by the distance of the neighboring point to the tangent plane and I is the square of arc-length between the neighboring point and the central point.

The normal curvature can be estimated in a 3×3 local window in the discrete setting. As shown in Fig. 2, we consider the normal curvature on the point O along the direction $\overrightarrow{OX}$, where $x = (i, j)$ indicates the coordinates and $u_{i,j}$ denotes the image intensity. Then the directed distance from

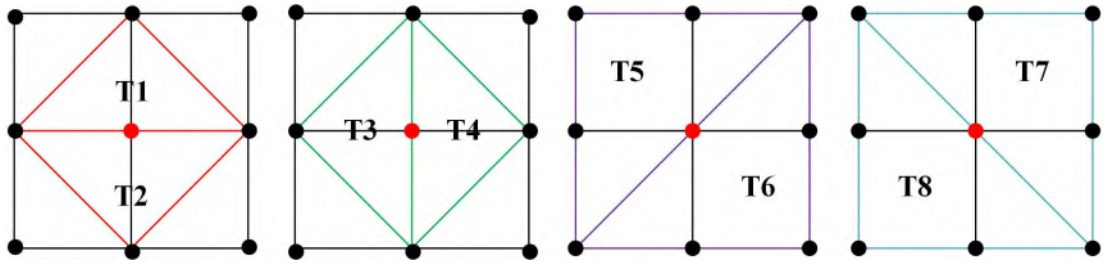


Fig. 3. Illustration of the eight tangent planes located in a 3×3 local patch on the finest layer, which locate pairwise centrosymmetric with regard to the center point nodal (marked by red color).

O to the tangent plane can be calculated by the half point $T(i-\frac{1}{2}, j, u_{i-\frac{1}{2},j})$ as follow

$$d_\ell = \overrightarrow{PT} \cdot \boldsymbol{n} = \frac{u_{i,j-1} + u_{i,j+1} - 2u_{i,j}}{2\sqrt{(u_{i,j-1} + u_{i,j+1} - 2u_{i-1,j})^2 + (u_{i,j+1} - u_{i,j-1})^2 + 4}}, \tag{2}$$

where $\boldsymbol{n}$ is defined by the cross product of the vector $\overrightarrow{XZ}$ and $\overrightarrow{XY}$. On the other hand, the arc-length $\widehat{OT}$ can be approximated in the following way

$$ds = \widehat{OT} \approx \sqrt{(u_{i-\frac{1}{2},j} - u_{i,j})^2 + h^2}, \tag{3}$$

where $u_{i-\frac{1}{2},j}$ can be estimated as the mean of points $u_{i-1,j}$ and $u_{i,j}$, and h is the grid step size. Then the normal curvature along the direction $\overrightarrow{OX}$ can be computed.

Fig. 3 displays the eight triangle planes located physically nearest to the center point O, which are used to approximate the tangent planes in different directions. Therefore, we can enumerate eight normal curvatures on a 3×3 local window. The principal curvatures are obtained by

$$\kappa_{\min}(u) = \min\{\kappa_\ell(u)\}, \ \kappa_{\max}(u) = \max\{\kappa_\ell(u)\}, \tag{4}$$

where $\ell = 1, 2, \cdots, 8$. Then the mean curvature and Gaussian curvature are defined as follows

$$H = \frac{1}{2}(\kappa_{\min} + \kappa_{\max}), \ K = \kappa_{\min}\kappa_{\max}. \tag{5}$$

B. The SAV scheme

The SAV method was introduced [25], [26] for a class of general gradient flows such as phase field models [27], [28], nonlinear Schrödinger equations [29], [30], Bose-Einstein Condensates [31], viscous fluid equations [32]. Recently, the SAV algorithm has been used for to solve the hybrid fractional total variation model [22], which is effective for image super-resolution problems. The SAV scheme was introduced to minimize the following free energy problem

$$\min_u \quad E(u) = \frac{1}{2}(u, \mathcal{L}u) + E_1(u), \tag{6}$$

where $\mathcal{L}$ is a symmetric non-negative linear operator, and $E_1(u)$ is a nonlinear functional bounded from below. The gradient flow of the above free energy (6) gives

$$\begin{cases} \dfrac{\partial u}{\partial t} = \mathcal{G}\mu, \\ \mu = \dfrac{\delta E}{\delta u}, \end{cases} \tag{7}$$

subject to either periodic boundary condition or Neumann boundary condition, where $\mathcal{G} = -I$ for the L^2 gradient flow and $\mathcal{G} = -\Delta$ for the H^{-1} gradient flow. A time discretization scheme for (7) is said to be energy stable if it satisfies a discrete energy dissipation law

$$\frac{d}{dt}E(u) = \frac{\delta E}{\delta u} \cdot \frac{\partial u}{\partial t} = (\mu, \mathcal{G}\mu) \leq 0.$$

It is shown that the SAV approach is an efficient and unconditionally energy stable scheme for gradient flows [25], [26].

We can simply classify the existing approaches to construct energy stable schemes for gradient flows into three categories as follows. The first category is the convex splitting methods by expressing the free energies as the difference of two convex functionals, which gives the first-order accurate schemes. The second kind of approaches is the so-called stabilization method introduced in [33], where the main idea is to introduce an artificial stabilization term to balance the explicit treatment of the nonlinear term. The stabilization method can be extended to second-order accurate schemes, but they are not unconditionally energy stable in general [26]. The third category is the method of invariant energy quadratization (IEQ) [34], which is a generalization of the method of Lagrange multipliers of auxiliary variables, which allows to construct linear, unconditionally stable, and second-order unconditionally energy stable schemes for a large class of gradient flows. However, it requires to solve a linear system involving VARIABLE coefficients at each time step. The SAV approach modifies the IEQ scheme by introducing a scalar auxiliary variable, which makes the linear equations be of CONSTANT coefficients. Thus, the SAV inherits all advantages of IEQ approach, while overcome its shortcomings.

The SAV scheme was firstly used for image super-resolution problem was [22], which combined the TV regularization and the fractional differential regularization as follows

$$\min_u \frac{\epsilon}{2}\int_\Omega |\nabla u|^2 dx + \beta \int_\Omega |\nabla^\alpha u| dx + \frac{\lambda}{2}\int_\Omega (f - Hu)^2 dx,$$

for $u \in BV^\alpha(\Omega) \cap H^1(\Omega)$. The above energy functional has the same formulation as (6), which is a direct application of the original SAV scheme. In this work, we extend the SAV scheme in both theory and application, solving the more sophisticated curvature minimization models for general image restoration problem.

III. Our Approach

Given an image domain Ω and an observed image $f : \Omega \to \mathbb{R}$, the mean curvature and Gaussian curvature regularization model aims to recover an clean image surface $(x, u(x))$ by minimizing the following the curvature energy [13]

$$\min_u \quad \int_\Omega \phi(\kappa)\sqrt{1+|\nabla u|^2}dx + \lambda\int_\Omega (Hu-f)^2dx, \quad (8)$$

where $\phi(\kappa) = 1 + \alpha|\kappa|$ is a function of curvature, λ is a positive parameter, and H is a linear operator varying with different image processing tasks. Since the surface area term introduces strong nonlinearity, we simplify it by replacing the area term with total variation term as follows

$$\min_u \quad E(u) := \int_\Omega \phi(\kappa)|\nabla u|dx + \lambda\int_\Omega (Hu-f)^2dx, \quad (9)$$

which means the curvature is defined over the level curves of the given images. For the ease of illustration, we define $F(u)$ as the energy density function

$$F(u(x)) = \phi\big(\kappa(u(x))\big)|\nabla u(x)| + \lambda(Hu(x)-f(x))^2. \quad (10)$$

Since $F(u) \geq 0$, it is easy to check that the energy $E(u)$ has the lower bound 0. Thus, we can assume that there exists a constant C_0 to make $E(u) + C_0 > 0$.

Define a scalar auxiliary variable

$$r = \sqrt{E(u)+C_0}.$$

Then we can rewrite the variational derivative $F'(u)$ as follows

$$F'(u) = \frac{r}{r}F'(u) = \frac{r}{\sqrt{E(u)+C_0}}F'(u).$$

By taking the above form into (7), we can obtain an equivalent system of gradient flow with a scalar auxiliary variable r as follows

$$\begin{cases} \dfrac{\partial u}{\partial t} = -\mu, \\ \mu = \dfrac{r}{\sqrt{E(u)+C_0}}F'(u), \\ \dfrac{\partial r}{\partial t} = \dfrac{1}{2\sqrt{E(u)+C_0}}\displaystyle\int_\Omega F'(u)u_t dx. \end{cases} \quad (11)$$

To illustrate the energy decay of scheme (11) , we take inner product of the first two equations in (11) with μ and $\frac{\partial u}{\partial t}$, which gives

$$\int_\Omega \frac{\partial u}{\partial t}\mu dx = -\|\mu\|^2 = \frac{r}{\sqrt{E(u)+C_0}}\int_\Omega F'(u)\frac{\partial u}{\partial t}dx. \quad (12)$$

Then, by multiplying the third equation in (11) with $2r$, we get

$$\frac{\partial r}{\partial t}2r = \frac{2r}{2\sqrt{E(u)+C_0}}\int_\Omega F'(u)u_t dx \quad (13)$$

where $\frac{\partial r}{\partial t}2r = \frac{\partial r^2}{\partial t}$. Let $\widetilde{E}(u) = r^2 = E(u) + C_0$. By the above two equations (12) and (13), it can be easily checked that the scheme (11) satisfies energy dissipation law, such as

$$\frac{d}{dt}\widetilde{E}(u) = -\|\mu\|^2. \quad (14)$$

In what follows, taking the time step Δt, the scheme (11) is amenable to simple and efficient numerical scheme as follows

$$\begin{cases} \dfrac{u^{n+1}-u^n}{\Delta t} = -\mu^{n+1}, \\ \mu^{n+1} = \dfrac{r^{n+1}}{\sqrt{E(u^n)+C_0}}F'(u^n), \\ \dfrac{r^{n+1}-r^n}{\Delta t} = \dfrac{1}{2\sqrt{E(u^n)+C_0}}\displaystyle\int_\Omega F'(u^n)\frac{u^{n+1}-u^n}{\Delta t}dx. \end{cases} \quad (15)$$

Then we have following theorem to guarantee the above scheme (15) is energy stable for the gradient flow (7).

Theorem 1. *The scheme* (15) *is unconditionally energy stable in the sense that*

$$\widetilde{E}(u^{n+1}) - \widetilde{E}(u^n) = -\Delta t(\mu^{n+1})^2 - (r^{n+1}-r^n)^2 \leq 0 \quad (16)$$

where $\widetilde{E}(u^{n+1}) = (r^{n+1})^2 = E(u^{n+1}) + C_0$ *is the modified energy.*

Proof. On the first place, we multiply the three equations in (15) with μ^{n+1}, $\frac{u^{n+1}-u^n}{\Delta t}$, and $2r^{n+1}$, respectively. Then by integrating the first two equations and adding them together, we can obtain

$$2r^{n+1}(r^{n+1}-r^n) = -\Delta t(\mu^{n+1})^2.$$

Using the identity

$$2r^{n+1}(r^{n+1}-r^n) = (r^{n+1})^2 - (r^n)^2 + (r^{n+1}-r^n)^2,$$

one can obtain

$$\begin{aligned} \widetilde{E}(u^{n+1}) - \widetilde{E}(u^n) &= (r^{n+1})^2 - (r^n)^2 \\ &= -\Delta t(\mu^{n+1})^2 - (r^{n+1}-r^n)^2 \\ &\leq 0, \end{aligned} \quad (17)$$

which shows the diminishing of energy. □

A. Solution to discrete scheme (15)

Although it looks a little bit complicated, we can obtain a linear equation from the scheme (15) to solve u^{n+1}. Indeed, taking μ^{n+1} and r^{n+1} to the first equation of (15), we have

$$\frac{u^{n+1}-u^n}{\Delta t} = -\frac{F'(u^n)}{\sqrt{E(u^n)+C_0}}\Big(r^n + \frac{\int_\Omega F'(u^n)(u^{n+1}-u^n)}{2\sqrt{E(u^n)+C_0}}\Big).$$

For convenience, we denote $b^n = \frac{F'(u^n)}{\sqrt{E(u^n)+C_0}}$. Then the equation (15) can be simplified as

$$u^{n+1} + \frac{\Delta t}{2}b^n\left(b^n, u^{n+1}\right) = c^n, \quad (18)$$

where

$$c^n = u^n - \Delta t r^n b^n + \frac{\Delta t}{2}(b^n, u^n)b^n. \quad (19)$$

Thus, we need to compute $\left(b^n, u^{n+1}\right)$. By taking the inner product with b^n, we then get

$$\left(b^n, u^{n+1}\right) + \frac{\Delta t}{2}\|b^n\|^2\left(b^n, u^{n+1}\right) = \left(b^n, c^n\right), \quad (20)$$

Fig. 4. Test images and their sizes used in parameter discussing.

which gives

$$\left(b^n, u^{n+1}\right) = \frac{(b^n, c^n)}{1 + \frac{\Delta t}{2}\|b^n\|^2}. \tag{21}$$

In what follows, we can obtain u^{n+1} by taking (21) back into (18).

B. The variational derivative of $F(u)$

The only left issue is how to compute the variational derivative of the functional $F(u)$. To avoid estimate the gradient of the curvature related term, we regard $\phi(\kappa)$ as a weighted function of the total variation term, and then obtain $F'(u)$ as follows

$$F'(u^n) = -\phi(\kappa(u^n))\nabla \cdot \frac{\nabla u^n}{\sqrt{|\nabla u^n|^2 + \beta}} + H^T(Hu^n - f), \tag{22}$$

where the curvature $\kappa(u^n)$ is either mean curvature or Gaussian curvature being explicitly evaluated in the local 3×3 window by the normal curvatures as discussed previously, H^T is the transpose operator to H, and β being a small positive parameter to avoid the singularity.

C. Our SAV algorithm

The implementation of the algorithm to solve the curvature-based model (9) is summarized in Algorithm 1.

Remark 1. *In order to accelerate the SAV scheme speed, we follow the adaptive time stepping strategy in [26]. The time step size τ is updated adaptive using the formula*

$$\tau_{n+1} = \max\left\{\tau_{\min}, \min\left\{A_{d\rho}\left(e_{n+1}, \tau_n\right), \tau_{\max}\right\}\right\}$$

with $A_{d,\rho}(e, \tau) = \rho(\frac{tol}{e})^{\frac{1}{2}}\tau$, where ρ is a default safety coefficient, tol is a reference tolerance, and e is the relative error at each time level.

IV. Numerical Experiments

This section presents the numerical results to evaluate the performance of our SAV method on image denoising, single image super-resolution, and image deblurring problems. All experiments are implemented in a MATLAB R2016a environment on a desktop with an Intel Core i5 CPU at 3.3 GHz and 8 GB memory.

The qualities of the recovered images sized at $m_1 \times m_2$ are measured using both peak signal-to-noise ratio (PSNR)

$$\text{PSNR}(u, \underline{u}) = 10\log_{10}\frac{\underline{u}_{\max}^2 \cdot m_1 m_2}{\|u - \underline{u}\|^2},$$

Algorithm 1: The SAV algorithm for solving the curvature minimization problem (9)

Input: given image f, linear operator H, parameters λ, α, tol, ρ, τ_{min}, τ_{max};
Output: u^{n+1}

for $n = 1, 2, \ldots$ **do**
- /* Compute $F'(u^n)$ from (22) */
- $F'(u^n) = -\phi(\kappa(u^n))\nabla \cdot \frac{\nabla u^n}{\sqrt{|\nabla u^n|^2+\beta}} + H^T(Hu^n - f)$;
- $b^n = \frac{F'(u^n)}{\sqrt{E(u^n)+C_0}}$;
- /* Compute c^n from (19) */
- $c^n = u^n - \tau_n r^n b^n + \frac{\tau_n}{2}(b^n, u^n)b^n$;
- /* Compute (b^n, u^{n+1}) from (21) */
- $(b^n, u^{n+1}) = \frac{(b^n, c^n)}{1+\frac{\tau_n}{2}\|b^n\|^2}$;
- $u^{n+1} = c^n - \frac{\tau_n}{2} b^n \left(b^n, u^{n+1}\right)$;
- /* Adaptive time stepping */
- $e_{n+1} = \frac{\left\|E(u^n) - E(u^{n+1})\right\|}{\|E(u^{n+1})\|}$;
- $\tau_{n+1} = \max\left\{\tau_{\min}, \min\left\{A_{d\rho}\left(e_{n+1}, \tau_n\right), \tau_{\max}\right\}\right\}$;
- /* Stopping condition */
- End till some stopping criterion meets;

end

and structural similarity index measure (SSIM)

$$\text{SSIM}(u, \underline{u}) = \frac{\left(2M_u M_{\underline{u}} + C_1\right) + \left(2\sigma_{u\underline{u}} + C_2\right)}{\left(M_u^2 + M_{\underline{u}}^2 + C_1\right)\left(\sigma_u^2 + \sigma_{\underline{u}}^2 + C_2\right)},$$

where u and $\underline{u}$ denote the restored and original images, respectively, $\underline{u}_{\max}$ represents the possible maximum pixel value of $\underline{u}$, M_u and $M_{\underline{u}}$ are the mean values of u and $\underline{u}$, σ_u and $\sigma_{\underline{u}}$ are the variances, $\sigma_{u\underline{u}}$ is the covariance of u and $\underline{u}$, and C_1 and C_2 are two constants. In addition, we use the following relative error of the numerical energy as the stopping criterion

$$RelErr\left(E(u^{n+1})\right) = \frac{E(u^{n+1}) - E(u^n)}{E(u^{n+1})} \leq \epsilon, \tag{23}$$

such that the iterations stop as long as the relative error in numerical energy smaller than the predefined tolerance ϵ. We simply fix $\epsilon = 10^{-4}$ in the following experiments.

A. Datasets

Referring to five widely used datasets, we evaluate the performance of our method and comparison methods, the

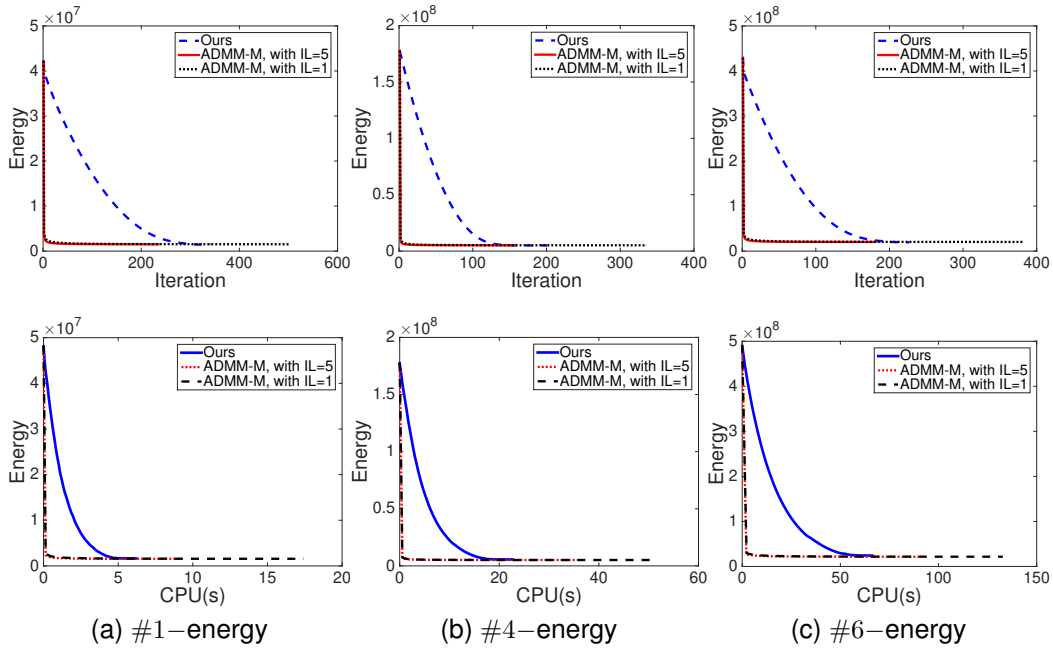


Fig. 5. Numerical energy evolution with respect to iteration and CPU time of the SAV and ADMM approaches for $\lambda = 0.08$, the ADMM-M method owns different inner loop (IL) 1 and 5.

TABLE I
IMAGE RESTORATION RESULTS ON TEST IMAGES WITH RESPECT TO DIFFERENT MINIMUM TIME STEPS, WHERE THE PARAMETER ARE GIVEN AS $\lambda = 0.08$ AND $\tau_{\max} = 10^{-1}$.

$\tau_{\min}$	Image	10^{-2}	10^{-4}	10^{-6}
PSNR	#1	29.34	29.34	29.34
	#2	28.79	28.92	30.72
	#3	30.92	30.92	30.92
	#4	30.95	30.95	30.95
	#5	28.91	28.91	28.91
	#6	29.82	29.82	29.87
Energy	#1	1.53	1.53	1.53
	#2	1.51	1.51	1.51
	#3	5.07	5.07	5.07
	#4	5.03	5.03	5.03
	#5	2.19	2.19	2.19
	#6	2.09	2.09	2.09
CPU	#1	2.74	2.85	2.91
	#2	2.69	2.63	2.56
	#3	13.25	13.34	13.00
	#4	11.82	11.74	11.43
	#5	56.01	56.32	56.10
	#6	48.80	48.83	48.20

TABLE II
RESULTS OF $\lambda = 0.08$ FOR THE FIXED MINIMUM TIME STEP $\tau_{\min} = 10^{-4}$ WITH DIFFERENT MAXIMUM TIME STEPS.

$\tau_{\max}$	Image	10^{-2}	0.1	0.3
PSNR	#1	29.13	**29.34**	29.18
	#2	28.61	28.92	**28.96**
	#3	30.61	**30.92**	30.67
	#4	30.86	30.95	**31.12**
	#5	28.62	**28.91**	28.75
	#6	29.55	**29.82**	29.67
Energy	#1	**1.52**	**1.52**	1.61
	#2	1.54	**1.51**	1.63
	#3	**5.07**	**5.07**	5.45
	#4	5.20	**5.03**	5.34
	#5	2.20	**2.19**	2.29
	#6	2.14	**2.09**	2.25
CPU	#1	2.37	2.85	**2.35**
	#2	2.94	**2.63**	2.91
	#3	15.19	13.34	**11.18**
	#4	15.04	**11.74**	13.80
	#5	60.59	56.32	**51.65**
	#6	62.22	48.83	**42.34**

details of which are described as follows

1) BSD68 dataset: It contains 68 gray images obtained from Berkeley segmentation dataset [35], which are downloaded from https://github.com/cszn/FFDNet/tree/master/testsets. The dataset is composed of a large variety of images ranging from natural images to object-specific such as plants, people, food etc. The sizes of the images in BSD68 vary by 241×161, 481×321 and 962×462.

2) Kodak dataset: The dataset contains total 24 color images, which are downloaded from http://r0k.us/graphics/kodak/. Kodak is a dataset used frequently for image denoising and super-resolution, which is released by Kodak in 1991 derived from a variety of film source materials. Note that each image in Kodak dataset is either 768x512 or 512x768 in size.

3) Set14 dataset: The Set14 dataset is a dataset consisting of 14 images, which is commonly used as a benchmark for image restoration tasks. It was introduced by Roman Zeyde *et al.* [36], which is composed of 13 color images and 1 gray image of different sizes.

4) McMaster dataset: The McMaster dataset [37] contains 18 color sub-images cropped from eight high-resolution natural images, the size of each image is 500×500 pixels. Compared to Kodak dataset, the images in McMaster dataset exhibit more saturated colors.

TABLE III
COMPARISON OF IMAGE DENOISING BETWEEN THE SAV AND ADMM METHODS WITH DIFFERENT VALUES OF λ, WHERE THE NOISE LEVEL IS GIVEN AS $\sigma = 20$.

Model		Mean Curvature						Gaussian Curvature					
Index	Image	$\lambda = 0.06$		$\lambda = 0.08$		$\lambda = 0.12$		$\lambda = 0.05$		$\lambda = 0.07$		$\lambda = 0.09$	
		ADMM	SAV	ADMM	SAV	ADMM	SAV	ADMM	SAV	ADMM	SAV	ADMM	SAV
PSNR	#1	28.58	28.82	29.05	**29.08**	28.98	28.86	28.31	28.88	29.12	**29.33**	29.03	28.38
	#2	27.87	28.26	28.34	**28.70**	28.49	28.14	27.82	28.11	28.42	**28.79**	28.11	27.92
	#3	30.36	30.61	30.68	**30.69**	30.10	30.02	30.17	30.17	30.97	**30.99**	29.11	28.75
	#4	30.39	29.42	31.16	**31.21**	31.09	30.00	30.22	30.71	31.63	**31.66**	28.51	28.35
	#5	27.66	28.47	28.14	**28.63**	28.45	28.56	27.68	28.17	28.49	**28.91**	27.38	27.55
	#6	28.99	29.68	29.37	**29.71**	29.68	29.35	29.14	29.61	29.42	**29.82**	28.21	28.25
Energy	#1	1.40	**1.39**	1.51	**1.49**	1.73	**1.60**	1.38	**1.33**	1.57	**1.52**	1.91	**1.89**
	#2	1.46	**1.41**	1.55	**1.48**	1.80	**1.72**	1.41	**1.36**	1.58	**1.51**	1.92	**1.88**
	#3	4.60	**4.58**	5.08	**4.94**	5.98	**5.40**	4.32	**4.26**	5.19	**5.07**	6.61	**6.48**
	#4	4.05	**4.01**	5.01	**4.72**	5.97	**5.39**	4.29	**4.24**	5.11	**5.03**	6.43	**6.25**
	#5	2.00	**1.98**	2.20	**2.17**	2.61	**2.42**	1.85	**1.82**	2.23	**2.19**	2.79	**2.73**
	#6	**1.91**	1.92	2.13	**2.06**	2.50	**2.29**	1.82	**1.81**	2.16	**2.09**	**2.69**	2.71
CPU	#1	21.43	**11.86**	21.61	**10.63**	21.55	**7.01**	2.79	**1.52**	4.95	**2.73**	4.13	**2.83**
	#2	21.09	**13.84**	21.43	**10.04**	21.48	**4.70**	2.75	**1.55**	4.82	**2.03**	4.47	**2.61**
	#3	98.04	**37.70**	99.74	**34.21**	98.00	**25.43**	15.42	**10.99**	18.84	**13.35**	24.09	**16.24**
	#4	98.83	**30.53**	99.57	**33.53**	98.05	**21.84**	15.27	**10.11**	37.77	**11.06**	24.19	**17.33**
	#5	426.19	**61.01**	428.96	**66.91**	431.96	**65.78**	65.78	**49.23**	80.94	**56.05**	107.84	**61.95**
	#6	426.08	**55.88**	430.07	**71.86**	434.40	**111.71**	65.87	**45.21**	80.71	**48.82**	105.67	**54.41**

5) Pan et al.*'s Text and Low-Illumination Dataset:* The dataset constructed by Pan *et al.* [38] consists of two parts, the text part containing 15 color document images and the low-illumination part containing 6 color low-illumination images.

B. Parameters discussing

To get the appropriate adaptive time step, we set different time steps as a comparison. In the experiment, we test six images of different scales, as shown in Fig. 4, which are degraded by Gaussian noise with zero mean and standard deviation of $\sigma = 20$. We set the parameters as $tol = 0.07$, $\rho = 0.4$, and $\lambda = 0.08$ for all images. In Table I, we fix $\tau_{\max} = 10^{-1}$ with different minimum time steps of $\{10^{-2}, 10^{-4}, 10^{-6}\}$. As can be observed, the minimum time step $\tau_{\min}$ has almost no effect on the denoising results.

Next, we discuss how to select the optimal maximum time step for the SAV algorithm. To be specific, we fix $\tau_{\min} = 10^{-4}$ with respect to different maximum time steps given as $\{10^{-2}, 0.1, 0.3\}$. The PSNR, numerical energy and CPU time are recorded in Table II, where higher PSNR and lower CPU time are obtained with $\tau_{\max} = 0.1$. Therefore, we fix the minimum and maximum time steps as $\tau_{\min} = 10^{-4}$ and $\tau_{\max} = 10^{-1}$ in the following experiments, and the initial time step is taken as $\tau_{\min}$.

There is another important parameter in our curvature minimization model, i.e., λ, which is used to balance the contribution between the data fidelity and regularization term. We compare our SAV approach with the ADMM algorithm in [13] to illustrate the effect of λ. The parameters of the ADMM method are selected as suggested in [13], where the penalty parameter is fixed as $r = 2$. We select the parameter in the curvature function as $a = 1$ and $b = 0.1$ for the mean curvature (MC) minimization models (both SAV and ADMM algorithms), while we let $a = 1$ and $b = 5$ for the Gaussian minimization (GC) models (both SAV and ADMM algorithms). For convenience, we use the ADMM-M and SAV-M to denote the MC methods, while the ADMM-G and SAV-G to denote the GC methods. Tables III lists the comparison results of SAV and ADMM method with respect to different values of λ. As can be observed, the best recovery results are obtained with $\lambda = 0.08$ for MC model and $\lambda = 0.07$ for GC model, which are obtained by our SAV method. Thanks to the unconditional energy stability of the SAV scheme, our algorithm always gives smaller numerical energies with less CPU time. It is worthy to mention that the advantage of SAV method is dominant especially for larger regularization parameters.

To further demonstrate the efficiency of the SAV approach, we track the decays of the numerical energies for different sizes of test images. For ADMM method, the nonlinear sub-minimization problem needs to be solved by Newton method, which results in the rising of computational costs. Since the Newton method is terminated by the number of its iteration, we set the maximum number of iterations to be 1 and 5 (i.e., inner loop, denoted by IL = 1, 5) in the comparison. Fig. 5 displays the plots of energy decays, where the numerical energies of both methods decrease and finally approach to the similar steady states. The plots demonstrate that the advantage of the unconditionally energy stability for the proposed methods. Specifically, we have the following two conclusions

- It is shown by the energy decay curves that the decay speed of the ADMM algorithm relies on the inner loop. Both the numbers of the outer iteration and CPU time decrease by increasing the number of inner loops from IL = 1 (the black line) to IL = 5 (the red line). However, it does not mean that the more inner loops, the more efficient. Too many inner loops will also result in the increase of computational costs.
- Our SAV algorithm is more efficient than the ADMM-based algorithm. Although the CPU time of the ADMM algorithm decreases as the number of the inner iteration increases to IL = 5, it is still higher than our SAV

TABLE IV
AVERAGE PSNR(DB)/CPU(S) OBTAINED BY DIFFERENT DENOISING METHODS ON BSD68 DATASET CORRUPTED BY NOISE LEVEL $\sigma = 20$. THE TOP BEST VALUE AMONG ALL THE METHODS AND VARIATIONAL METHODS ARE GIVEN IN ITALIC AND BOLD, RESPECTIVELY.

	size	ROF	Euler	CFMC	MC	TC	TFOV	ADMM-G	ADMM-M	DnCNN	CBDNet	NN	SAV-G	SAV-M
PSNR	241x161	27.62	28.15	26.43	28.38	28.41	28.17	28.44	28.54	30.22	*30.88*	29.50	28.77	**28.88**
	481x321	27.60	28.07	26.78	28.32	28.46	28.13	28.34	28.45	30.11	*30.89*	29.39	28.72	**28.85**
	962x462	27.93	28.37	27.87	28.40	28.52	28.32	28.52	28.53	30.29	*30.53*	29.44	**28.93**	28.89
CPU(s)	241x161	5.14	13.21	**0.25**	6.21	10.41	16.05	15.67	9.05	2.59	*1.84*	162.61	3.59	4.86
	481x321	23.84	53.95	**1.16**	37.21	39.61	98.08	40.48	39.33	*3.21*	4.09	375.98	14.66	24.23
	962x462	53.05	133.81	**9.52**	89.66	110.48	186.93	116.58	98.82	23.35	*8.02*	869.53	82.02	54.92

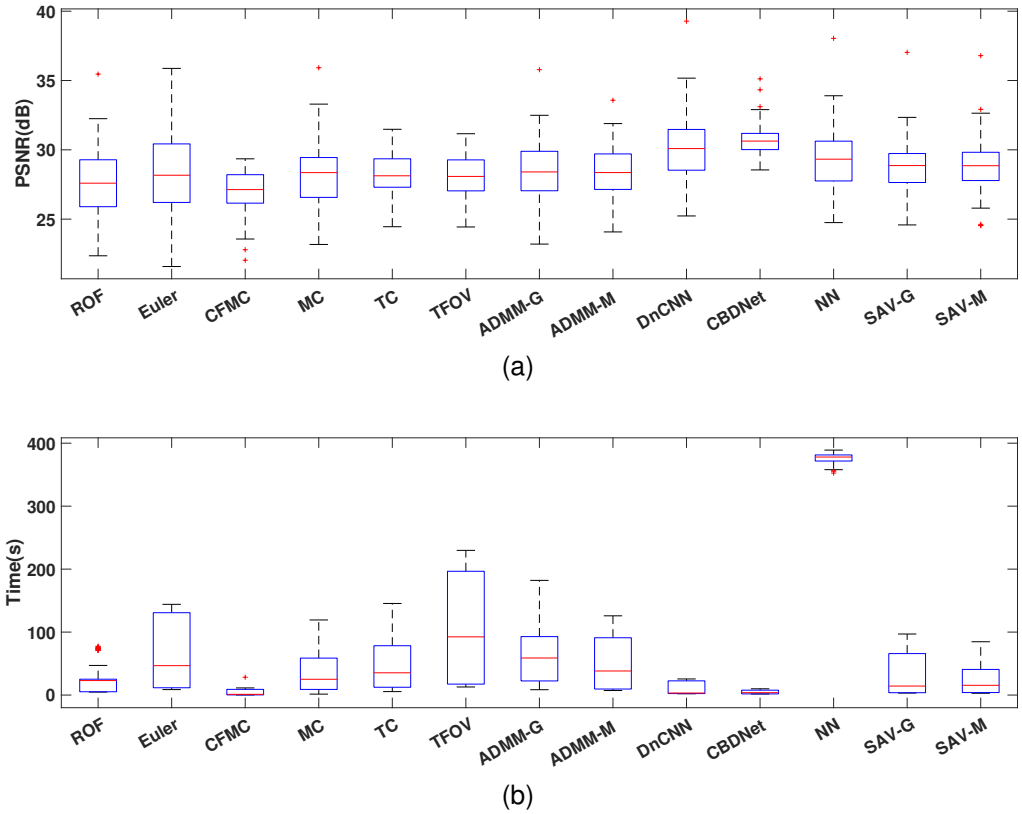


Fig. 6. Comparison of the PSNR and CPU time (s) among different methods for image denoising on the BSD68 dataset.

algorithm with respect to different scales of the test images. The reason behind is the SAV scheme consumes less time per iteration than ADMM algorithm with inner loops being 5.

In the following comparison studies, we fix the inner loop of ADMM algorithm to be IL = 5 to achieve the best computational efficiency.

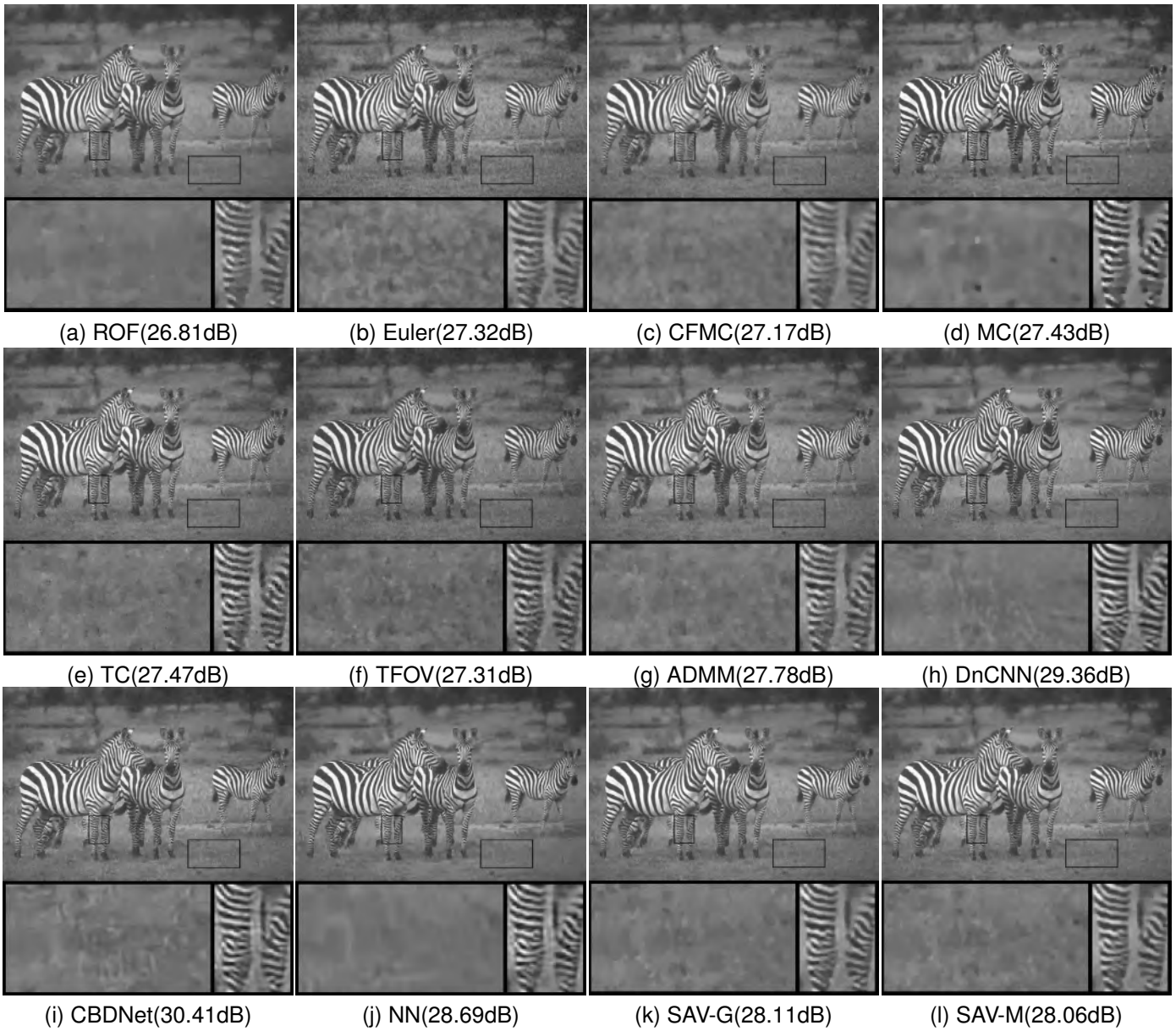
C. Image denoising

In this subsection, we assess the performance of our method on image denoising problems

$$f = Hu + n,$$

where $H = I$ with I being the identity operator and n being the additive white Gaussian noise (AWGN). We compare our method with the state-of-the-art variational methods including the Rudin-Osher-Fatemi (ROF) model [1], Euler's elastica (Euler) [11], curvature filter (CFMC) [16], mean curvature model (MC) [18], ADMM-based curvature minimization model (ADMM-M and ADMM-G) [13], total curvature model (TC) [14], total fractional-order variation (TFOV) [22], and deep learning methods including DnCNN [39], CBDNet [40] and NN [41]. The results of the compared methods are obtained by running the codes released by the authors. For fair and reasonable comparative experiments, the parameters for these methods are provided as suggested in their original papers

- ROF [1]: The total variation model is solved by primal-dual method, where the step size is given as $\tau = \frac{1}{4}$ and $\sigma = \frac{1}{8\tau}$ for primal and dual variable, respectively, and $r_1 = 10$, $\lambda = 15$.
- Euler [11]: Euler's elastica model is solved using the ADMM with the parameters of $a = 1$, $b = 10$, $\mu = 10$, $\eta = 200$, $r_1 = 2$, $r_2 = 200$, and $r_4 = 250$.
- CFMC [16]: The regularization parameter is set as $\alpha =$



(a) ROF(26.81dB) (b) Euler(27.32dB) (c) CFMC(27.17dB) (d) MC(27.43dB)

(e) TC(27.47dB) (f) TFOV(27.31dB) (g) ADMM(27.78dB) (h) DnCNN(29.36dB)

(i) CBDNet(30.41dB) (j) NN(28.69dB) (k) SAV-G(28.11dB) (l) SAV-M(28.06dB)

Fig. 7. Denoising results and local magnification for the #51 image on BSD68 dataset with noise level 20 using the comparative methods.

5, and the stopping criterion is given as

$$RelErr\big(E(u^{n+1})\big) = \frac{E(u^{n+1}) - E(u^n)}{E(u^{n+1})} \leq 10^{-4}.$$

- MC [18]: The MC model is solved using the ADMM with parameters given as $\varepsilon = 0.4$, $\eta = 1600$, $r_1 = 200$, $r_2 = 200$, $r_3 = 1 \times 10^4$, and $r_4 = 1 \times 10^4$.
- ADMM-M and ADMM-G [13]: The total absolute curvature model is solved using the ADMM algorithm with the parameters given by $r = 2$, $a = 1$, $b = 0.1$, and $b = 5$ for the model relying on the mean and Gaussian curvatures.
- TC [14]: The discrete total curvature regularity, minimizing the ℓ^1 norm of the normal curvatures in different directions, is solved by ADMM algorithm with $\mu = 20$, $\alpha = 5$, $T_{max} = 300$ and $\epsilon = 2 \times 10^{-5}$.
- TFOV [22]: The total fractional-order variation model integrated with total variation and fractional-order TV is solved by SAV method with the parameters $\alpha = 1.1$, $\beta = 1$, $\epsilon = 10^{-18}$, $\epsilon_1 = 10^{-9}$.
- DnCNN [39]: The supervised DnCNN model was trained by 400 images of size 180×180 for the range of the Gaussian noise levels as $[0, 55]$.
- CBDNet [40]: We re-trained the CBDNet on DSB400 dataset the same as DnCNN.
- NN [41]: It is an unsupervised deep learning method, which combined an existing Gaussian denoising approach with the proposed neural network. It does not require any training sample except the input noisy image itself. In our comparison, we use the model with Block Matching and 3D Filtering (BM3D) [42] as Gaussian denoiser.

Except for the parameters of time step, the others of the

TABLE V
AVERAGE PSNR(DB)/SSIM OBTAINED BY DIFFERENT COMPARISON METHODS ON SINGLE IMAGE SUPER-RESOLUTION PROBLEM WITH THE UPSCALING FACTOR OF 2, 3 AND 4 ON BSD68, KODAK AND MCMASTER DATASET.

Dataset	Upscaling Factor	WBM	TVRBM	TFOV	SAV-M	SAV-G
		PSNR / SSIM	PSNR / SSIM	PSNR / SSIM	PSNR / SSIM	PSNR / SSIM
BSD68	2	29.55 / 0.8577	29.65 / 0.8632	29.92 / 0.8396	31.18 / **0.8705**	**31.22** / 0.8585
	3	26.92 / 0.7888	26.69 / 0.7948	27.92 / 0.7510	**28.75** / **0.7971**	28.70 / 0.7934
	4	25.16 / 0.7183	25.29 / 0.7157	26.27 / 0.6857	26.86 / 0.7278	**26.89** / **0.7332**
Kodak	2	26.79 / 0.8765	26.93 / 0.8784	27.07 / 0.8607	**27.97** / 0.8798	**27.97** / **0.8801**
	3	25.03 / 0.8159	25.14 / 0.8265	25.74 / 0.8101	26.65 / 0.8398	**26.66** / **0.8409**
	4	23.78 / 0.7744	23.70 / 0.7851	24.81 / 0.7902	**25.25** / **0.8931**	25.23 / 0.8940
McMaster	2	28.34 / 0.9225	28.37 / 0.9235	29.09 / 0.9118	**30.13** / **0.9310**	30.10 / 0.9303
	3	25.93 / 0.8843	25.71 / 0.8849	27.18 / 0.8757	27.82 / 0.8931	**27.85** / **0.8940**
	4	24.93 / 0.8547	24.50 / 0.8503	25.83 / 0.8427	26.23 /0.8593	**26.26** / **0.8600**

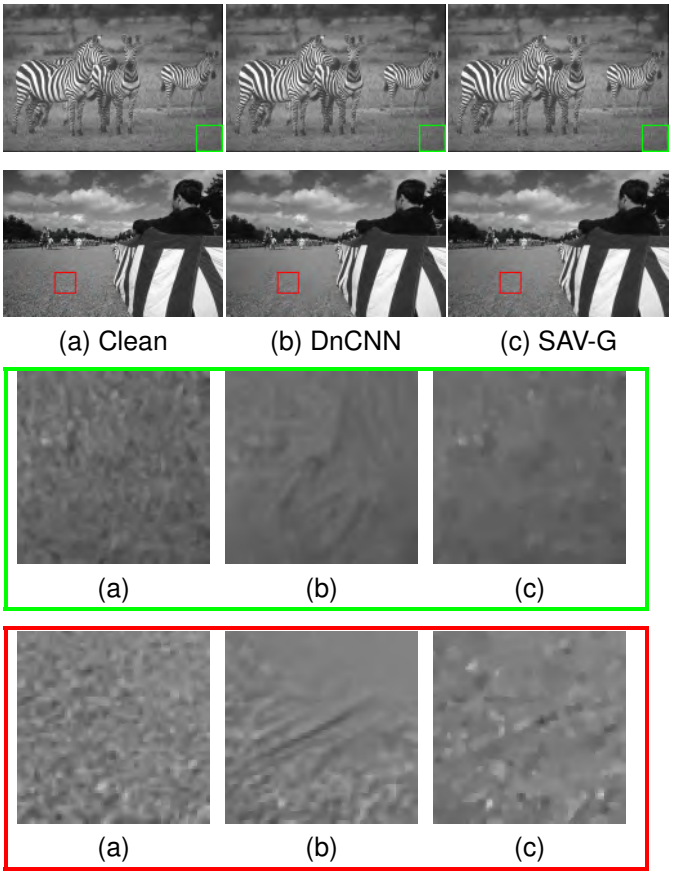


Fig. 8. Denoising results and local magnification for the #19 and #51 images on BSD68 dataset with noise level 20 by our method and DnCNN.

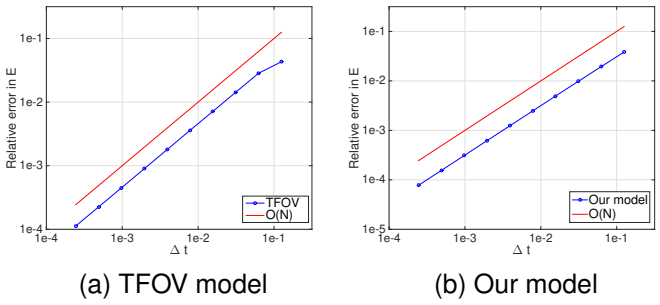


Fig. 9. Energy decay rates of the SAV-based schemes.

SAV-M and SAV-G algorithms are chosen the same with the ADMM-M and ADMM-G, respectively.

We evaluate the denoising performance on the BSD68 dataset with the noise level $\sigma = 20$. Fig. 6 presents the PSNRs and CPU time for different denoising methods by using boxplot. It can be seen the maximum, minimum and the median PSNR of our SAV method is the highest among all the variational methods shown by Fig. 6 (a). Particularly, the width of the boxes indicates the interquartile range of our results is smaller than other variational methods, which means our method can achieve more consistent restoration results with high qualities. On the other hand, the median CPU time of our SAV method is the smallest compared with other variational methods without any outliers except for CFMC as shown by Fig. 6 (b). Therefore, our SAV scheme performs faster than all the other high-order methods.

The average PSNR and CPU running time obtained by different methods on the BSD68 dataset are displayed in Table IV. As one can see, our proposed SAV-G and SAV-M achieve the best PSNR results among all competing variational methods. More importantly, the curvature regularization methods, i.e., MC, TC, ADMM-M/G, SAV-M/G, perform better than total fractional-order variation model, which convince the advantage of curvature regularization methods. Note that CFMC performs poorly compared to other curvature methods because the curvature estimation is not accurate enough. In addition to denoising quality, another superiority of our SAV method is its high computational efficiency. It can be seen that the SAV method can save much CPU compared to other high-order models. Although the CFMC [16] performs faster than our SAV methods, by taking the restoration quality into consideration, our method is still shown to be the most competitive image denoising method. And especially, the SAV method saves much time compared to unsupervised deep learning method NN [41].

We present the restoration results in Fig. 7 to better visualize the denoising performance of these methods, where the selected local magnification views are also used for better comparison. Obviously, our SAV method gives better visual results than other variational methods. Note that ROF, MC and TFOV tend to produce over-smooth results in the texture areas. In particular, our method has better capability in recovering edges and small scale details. For example, the stripes of the zebra and and the grassland in our restored image are much

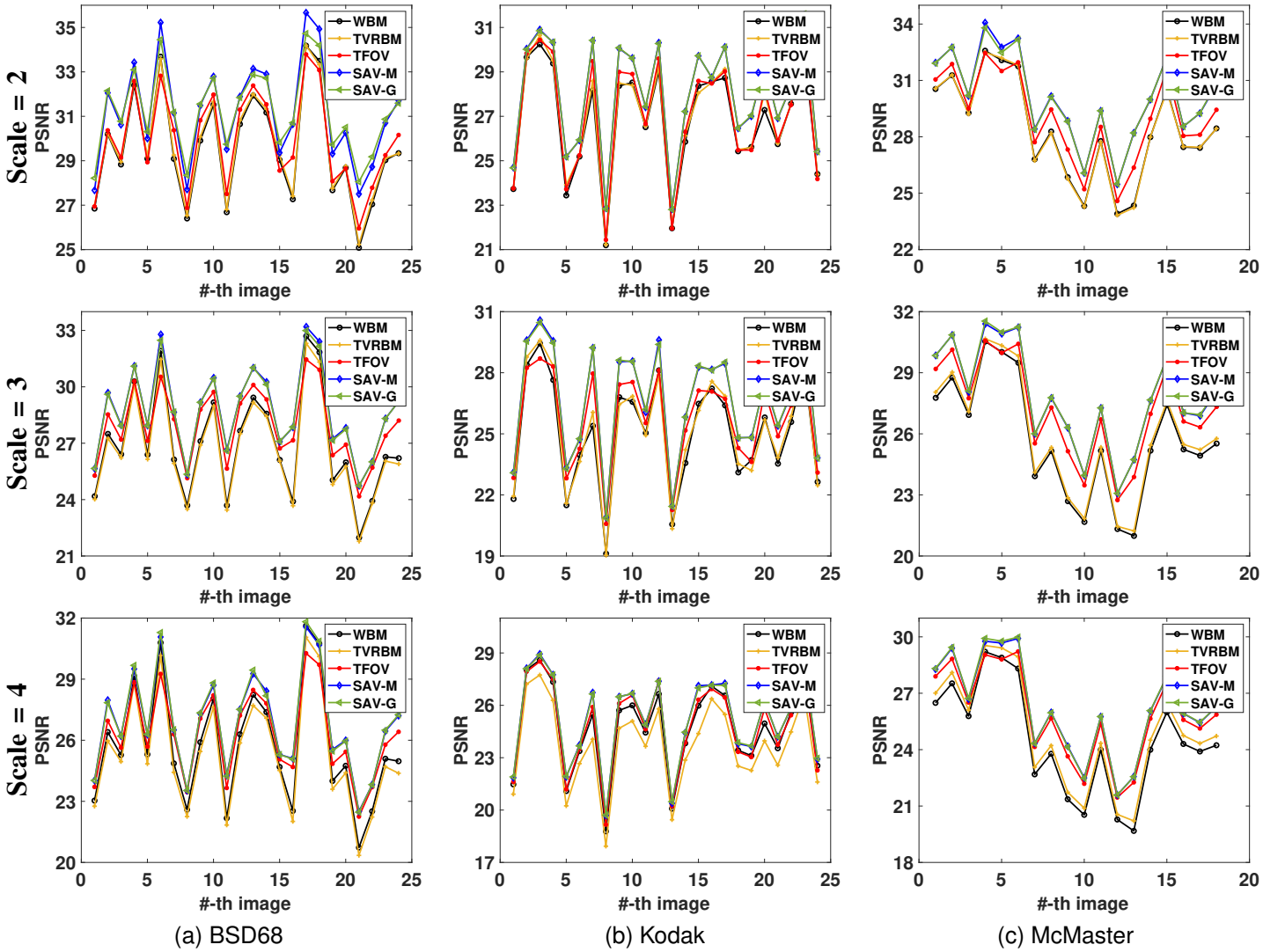


Fig. 10. PSNR comparison of our SAV for image super-resolution with 3 other methods on BSD68, Kodak and McMaster datasets.

better than that obtained by other variational methods. In what follows, we compare the energy decay rates of the two SAV-based methods, TFOV and our model. Since the minimum time step τ_{min} almost has no effect to the restoration, we fix $\tau_{min} = t/2^{18}$ and $\tau_{max} = t/2^i$ with $t = 8$ for $i = 6, 7, ..., 15$. As shown by the plots of the numerical energies in Fig. 9, both SAV-based algorithms are of the first-order energy convergence.

Though it can be seen from Fig. 6 and Table IV that DnCNN outperforms our SAV method in terms of PSNR and CPU time, deep learning methods are not perfect. Fig. 8 illustrates two representative denoised images selected from BSD68 dataset, which are obtained by our SAV method and DnCNN model, respectively. As can be observed, the DnCNN model tends to generate some unnatural artifacts, while our SAV method can keep natural structures of the images. As the representative deep learning method, the DnCNN model contains more six hundred thousand parameters, which are required to be learned on the training dataset. And it is a typical non-convex minimization problem with no guarantee of convergence. More importantly, the deep learning methods lack interpretability making the artifacts unpredictable. In contrast, the variational models have good mathematical properties and theoretical guarantee.

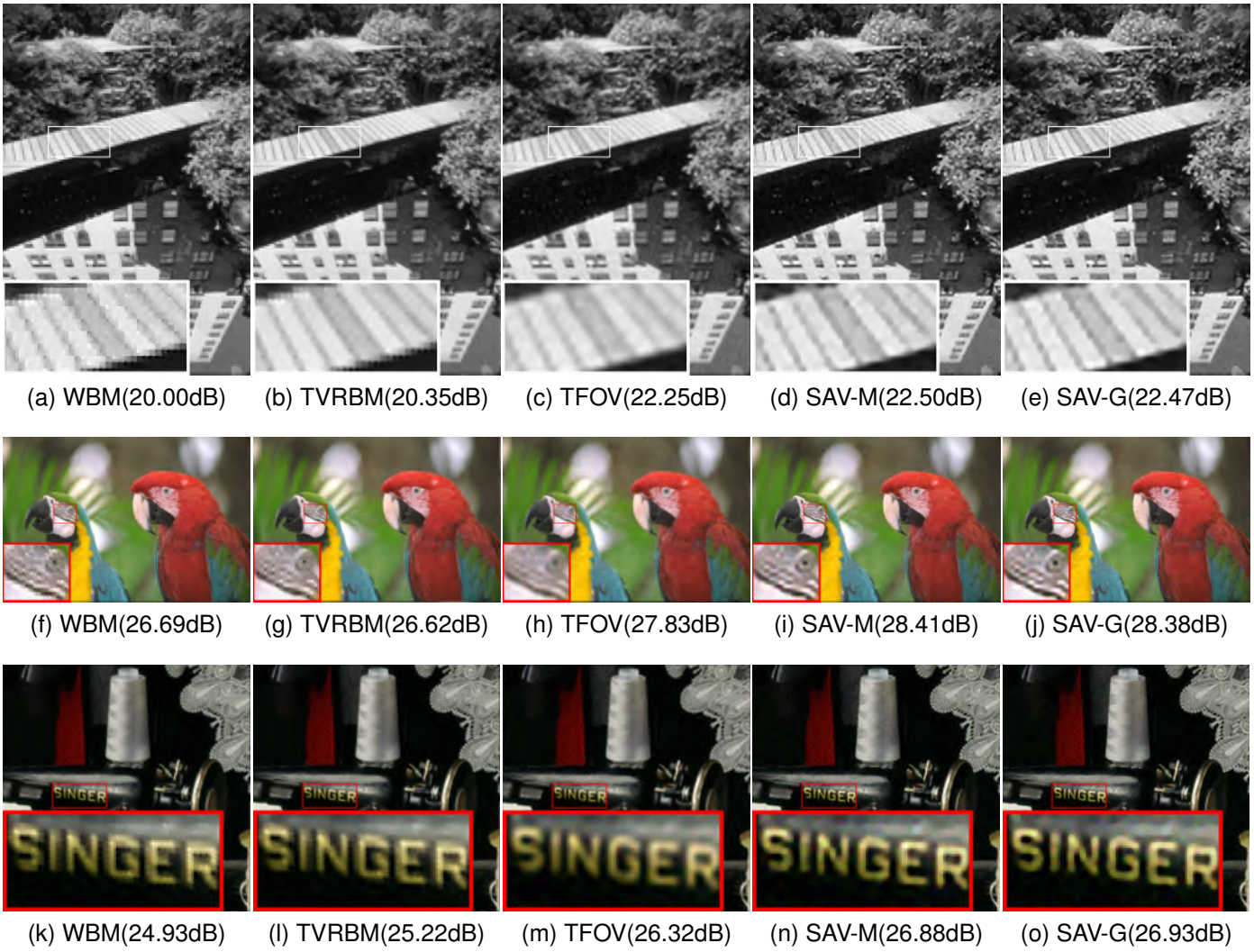
D. Image super-resolution

Next, we consider the performance of our model on single-image super-resolution problem and the degradation image model becomes

$$f = Hu + n,$$

where $Hu = D(h * u)$ with D being a down-sampling linear operator and h being a translation invariant convolution kernel for single image super-resolution. Accordingly, there is $H^T(Hu - f) = h^T * \big(D^T(D(h * u) - f)\big)$ with D^T being the up-sampling operator and h^T being the adjoint operator of h.

We evaluate and compare the performance of our method with several state-of-the-art SR methods including the wavelet-based model [43], TV regularization model [43], and total



(a) WBM(20.00dB) (b) TVRBM(20.35dB) (c) TFOV(22.25dB) (d) SAV-M(22.50dB) (e) SAV-G(22.47dB)

(f) WBM(26.69dB) (g) TVRBM(26.62dB) (h) TFOV(27.83dB) (i) SAV-M(28.41dB) (j) SAV-G(28.38dB)

(k) WBM(24.93dB) (l) TVRBM(25.22dB) (m) TFOV(26.32dB) (n) SAV-M(26.88dB) (o) SAV-G(26.93dB)

Fig. 11. The visual comparison of image super-resolution problems, where the images are down-sampled by the factor 4 and corrupted by Gaussian noises of $\sigma = 5$.

fractional-order variation model [22]. The degraded low resolution (LR) images are produced by applying a truncated 15×15 Gaussian kernel with a standard deviation of 1 to the original image followed by down-sampling with factors s=2, 3 and 4, respectively. The additive Gaussian noises of zero mean and variation $\sigma = 5$ are also introduced to the LR images. The implementation details of the SR methods are described as follows

- Wavelet-based model (WBM) [43]: The model is solved by the ADMM method with parameters $\tau = 2 \times 10^{-4}$ and $\mu = 0.05$.
- Total variation regularization-based model (TVRBM) [43]: The model is solved using the ADMM with parameters $\tau = 2 \times 10^{-3}$ and $\mu = 0.05$.
- Total fractional-order variation model (TFOV) [22]: The model is solved using the SAV algorithm with the parameters $\alpha = 1.1$, $\beta = 1$, $\epsilon = 10^{-4}$, and $\lambda = 30$. The minimum and maximum time steps are taken as $\tau_{\min} = 10^{-2}$ and $\tau_{\max} = 10^{-1}$, respectively.
- SAV-M and SAV-G: We fix $a = 1$ and $b = 0.1$ for the SAV-M model and $a = 1$ and $b = 5$ for the SAV-G model while the other parameters are the same as the TFOV model.

Table V presents the average PSNR and SSIM for different methods on BSD68, Kodak and McMaster dataset with the upscaling factor of 2, 3 and 4, respectively. We can observe that the high-order regularization methods significantly improve the super-resolution performance compared to the TV regularization model. And our SAV scheme yields the highest PSNR and SSIM with 0.4 dB higher PSNR than TFOV model and 1.2 dB higher PSNR than TV model. Fig. 10 shows the PSNR comparison over each image on BSD68, Kodak and McMaster dataset, respectively. As can be seen, our SAV-M (denoted by the blue line) and SAV-G (denoted by the green line) always exhibit higher PSNR than the other three methods. We also display the CPU time for different methods on the three datasets in Fig. 12. Although the high-order models consume more computational time than TVRBM, our curvature regularization models are much faster than TFOV and WBM. We present selective results from BSD68, Kodak and McMaster dataset for visual comparison as displayed in Fig. 11. It can be seen that both SAV-M and SAV-G can produce

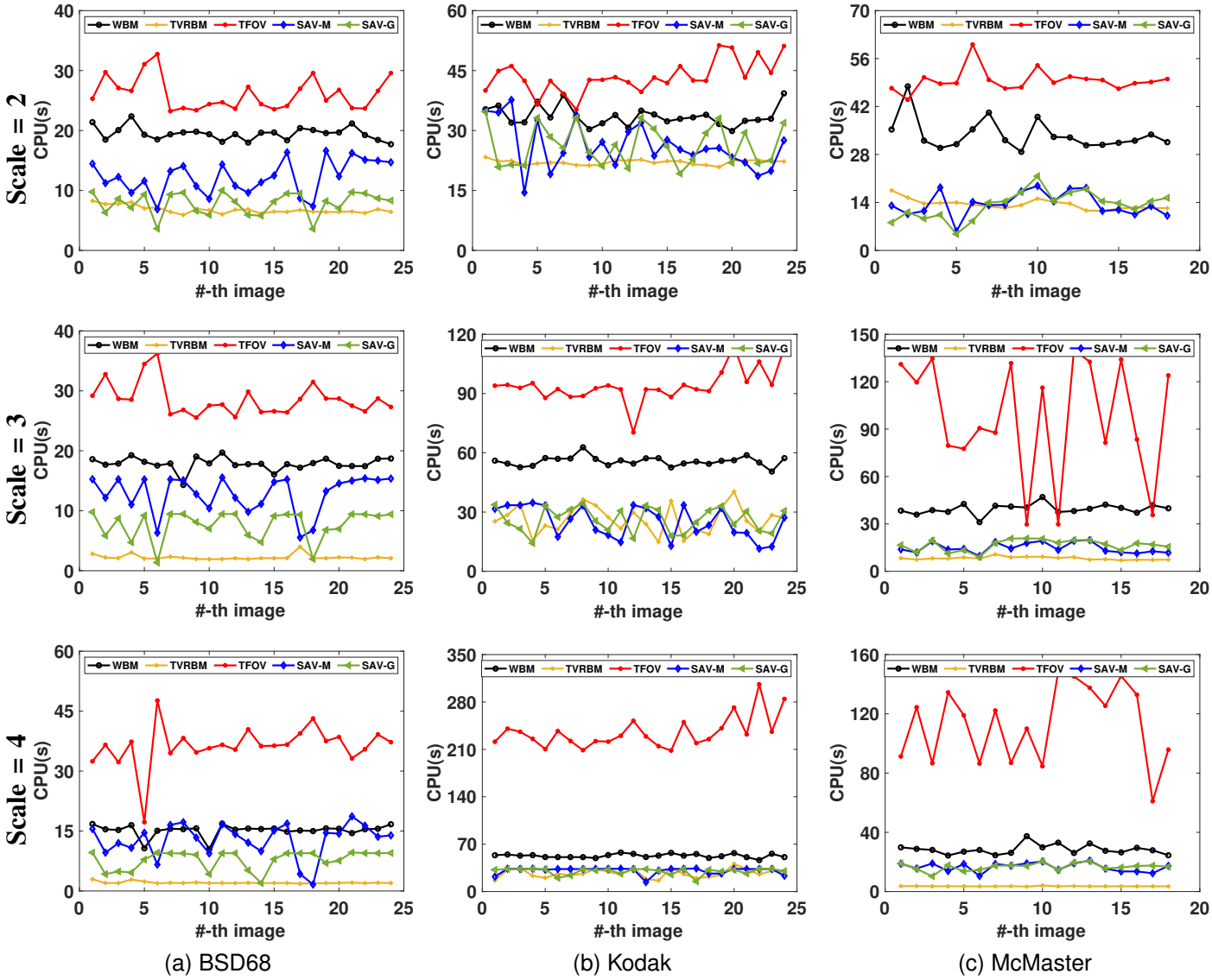


Fig. 12. CPU run time (in seconds) comparison on image super-resolution problem.

fine details and sharp edges whereas the total fractional order variational [22] tends to generate blurred edges, and the WBM and TVRBM methods are likely to generate staircase effect.

E. Image deblurring

We further extend the proposed model to image deblurring problem, and we compare it with the combined first- and second-order variational model [44]. The Gaussian blur of size 5×5 with a standard deviation 5 and additive Gaussian noises of variation $\sigma = 5$ are used to degrade the images. In implementation, the parameters of the comparative methods are detailed as follows

- Combined first- and second-order variational model [44] (TV-TV^2): The model is solved using the ADMM algorithm where we set $\alpha = 0.003$, $\beta = 0.001$, $p_1 = 0.08$, and $p_2 = 0.08$.
- SAV-M and SAV-G: We set $a = 1$, and $b = 0.1$ for the SAV-M model and $a = 1$ and $b = 5$ for the SAV-G model with regularization parameter $\lambda = 60$.

TABLE VI
AVERAGE PSNR(DB)/SSIM COMPARISON AMONG DIFFERENT METHODS FOR IMAGE DEBLURRING ON BSD68, KODAK AND PAN *et al.* DATASETS.

Dataset	TV-TV^2	SAV-M	SAV-G
	PSNR / SSIM	PSNR / SSIM	PSNR / SSIM
BSD68	29.59 / 0.8361	30.09 / 0.8371	**30.26 / 0.8382**
Kodak	27.19 / 0.8379	27.83 / 0.8706	**27.86 / 0.8748**
Text	23.69 / 0.9105	24.79 / 0.9198	**24.87 / 0.9200**
Low-illumination	27.20 / 0.8346	27.82 / 0.8568	**27.83 / 0.8589**

Similarly, we evaluate the deblurring methods on the BSD68, Kodak24 and Pan *et al.*'s text and low-illumination dataset. Table VI shows the average PSNR and SSIM obtained by different methods, where our methods are shown superior in PSNR and SSIM. As shown by Fig. 13, our method performs favorably against the TV-TV^2 method. For visual comparison, we display the selective restored images from the Kodak dataset and a low-illumination image from Pan *et al.*'s dataset in Fig. 14. Although the original image contains

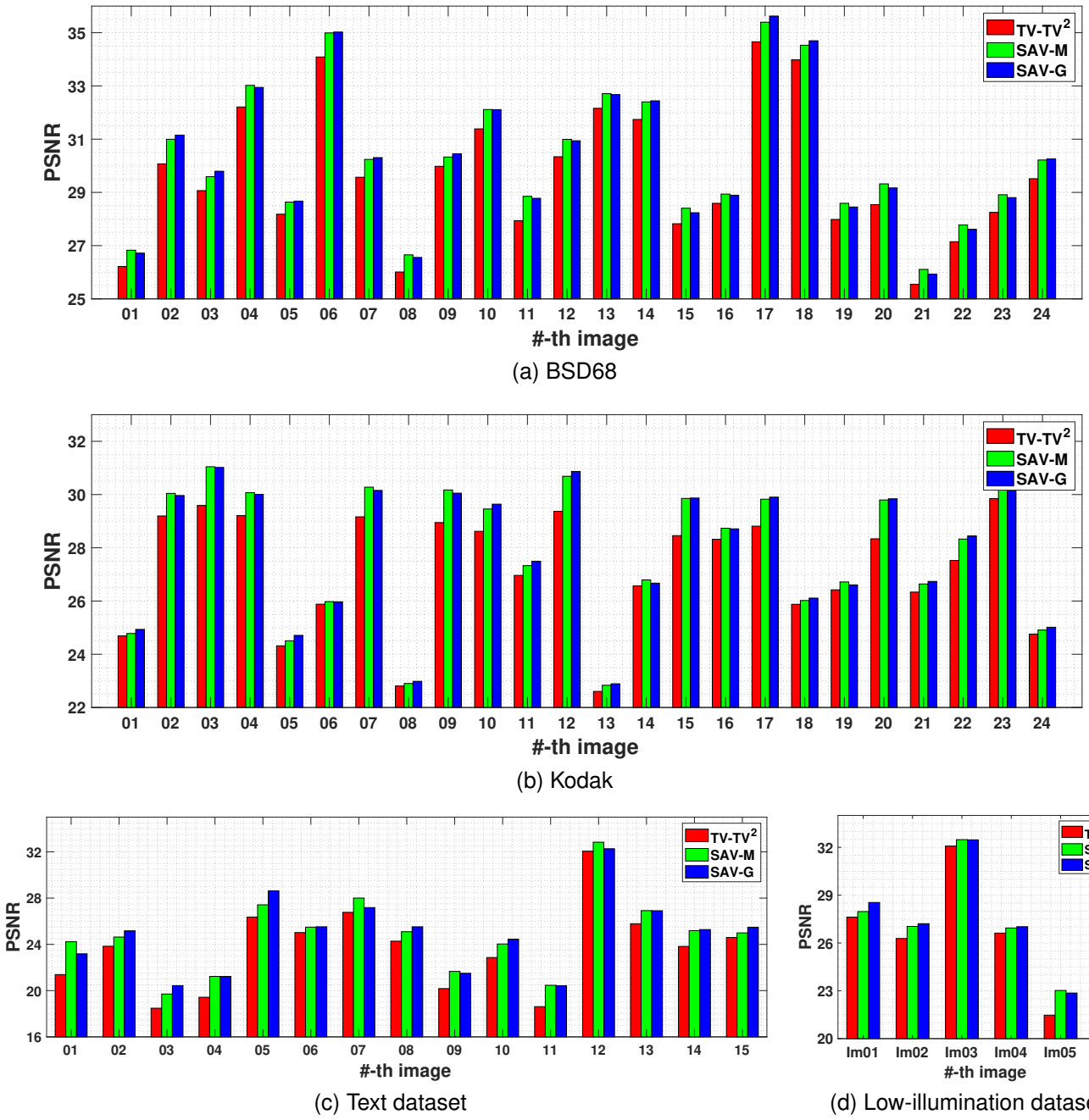


Fig. 13. PSNR comparison on image deblurring problem, where the images are corrupted by 5×5 Gaussian blur kernel and Gaussian noises of $\sigma = 5$.

abundant textures such as the folds on mountains, the blur makes the textures being smoothed out. As can be seen, our methods outperform TV-TV^2 in recovering edges shown by the boxes and zoomed in regions. In particular, the letters and outlines of the building recovered by our method is much clearer than TV-TV^2. Furthermore, Fig. 15 exhibits the running time, which illustrates that our SAV scheme requires much less computational costs than the TV-TV^2 model.

F. Mixed noises denoising

We have discussed our method on additive white Gaussian noises (AWGN) removal. However, Poisson noises that characterizes the fluctuation in counting number of photons are very common in optical images. Thus, we explore the extended application of our method on the mixed Gaussian-Poisson noises removal problems. Let u, $f \in \mathbb{R}^n$ be the ground truth and observed image corrupted by the mixed Poisson-Gaussian noises, respectively, satisfying

$$f = v + \mathbf{n}, \quad \text{with} \quad v \sim Poisson(u) \quad \text{and} \quad \mathbf{n} \sim \mathcal{N}\left(0, \sigma^2\right).$$

We aim to solve the mixed noise model proposed by Calatroni *et al.* [45], which is defined as follows

$$\min_{u,v} \int_\Omega \phi(\kappa)|\nabla u| \mathrm{d}x + \frac{\lambda_1}{2} \int_\Omega (f-v)^2 \mathrm{d}x + \lambda_2 \int_\Omega (u - v \ln \frac{u}{v} - v) \mathrm{d}x.$$

Although the mixed noise model contains two variables, SAV scheme can still be used to solve such problem and

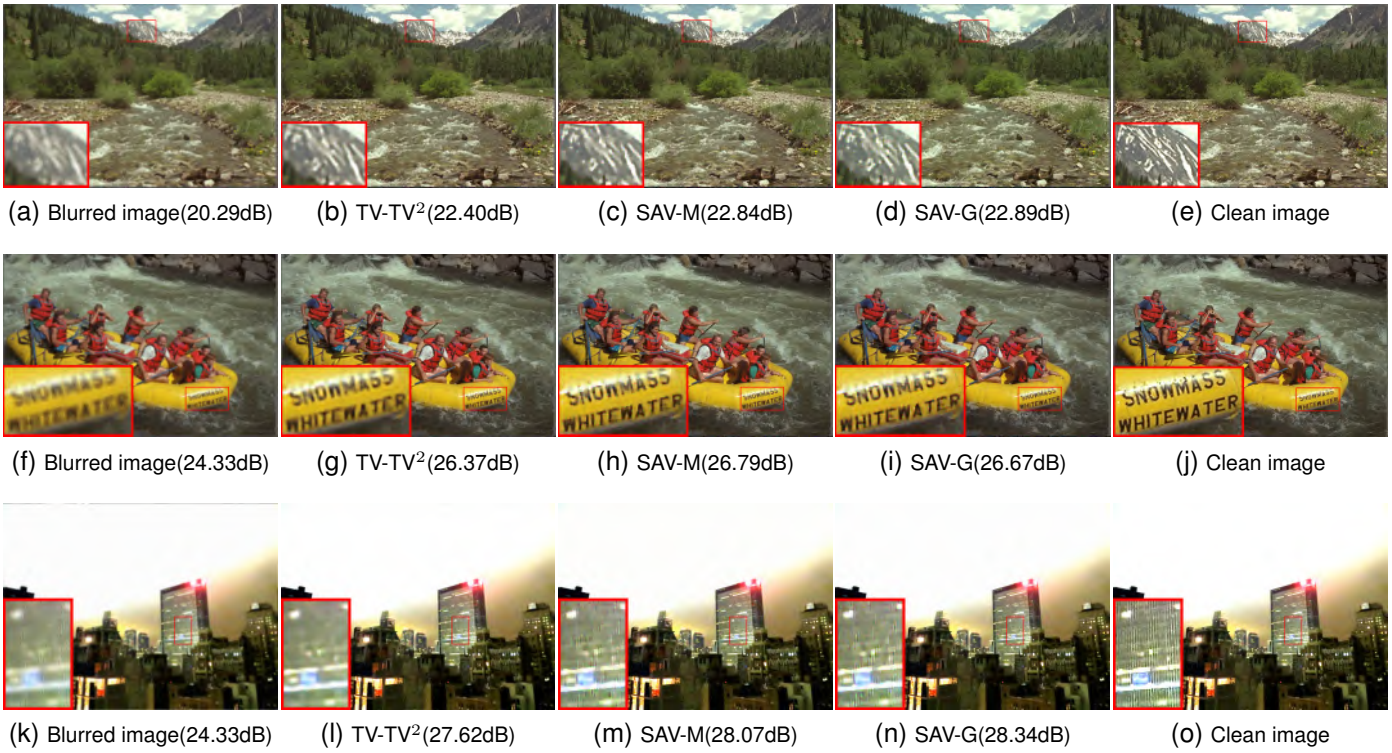


Fig. 14. Visual comparison for image deblurring problems, where images are corrupted by 5×5 Gaussian blur kernel and noise level of $\sigma = 5$.

TABLE VII
AVERAGE PSNR(DB)/SSIM/CPU(S) RESULTS OF DIFFERENT METHODS FOR REMOVING MIXED GAUSSIAN AND POISSON NOISES ON SET14.

	Noises	DnCNN	CBDNet	NN	BCA	SAV
PSNR	$\sigma = 5$	28.63	31.00	27.45	30.55	**31.01**
	$\sigma = 10$	28.55	29.83	27.26	29.64	**30.53**
	$\sigma = 20$	27.73	27.09	27.06	27.86	**28.45**
SSIM	$\sigma = 5$	0.8001	0.8303	0.7815	0.8492	**0.8591**
	$\sigma = 10$	0.7911	0.7869	0.7619	0.8071	**0.8239**
	$\sigma = 20$	0.7582	0.6825	0.7354	0.7648	**0.7843**
CPU(s)	$\sigma = 5$	**2.53**	2.85	310.14	28.24	22.14
	$\sigma = 10$	**2.64**	2.76	300.48	30.25	25.45
	$\sigma = 20$	**2.63**	2.93	310.74	35.24	27.15

ensure unconditionally energy diminishing. What is more, our SAV scheme does not need to introduce additional auxiliary variables, while ADMM-based algorithm in [46] introduced two auxiliary variables.

We choose both deep learning based method DnCNN [39], CBDNet [40], NN [41] with DnCNN as the Gaussian denoiser, and the total variation model solved by bilinear constraint based ADMM (BCA) [46] as the comparison methods. We introduce the mixed Gaussian and Poisson noises into the images in Set14 dataset, where different Gaussian noise level are used, i.e., $\sigma = 5, 10$ and 20. The average PSNR and SSIM of different methods are reported in Table VII. As can be observed, SAV performs the best among all comparison methods. Despite the fact that DnCNN has a very competitive performance for AWGN (as shown in section (IV-C)), it loose lose its effectiveness for mixed Gaussian and Poisson noises. Although we use NN combined the DnCNN as denoiser, it still performs poorly on mixed noises removal problems. Benefited from the noise estimation subnetwork and asymmetric loss, the CBDNet presents better generalization ability to mixed noises, achieving better restoration results than the other two deep learning methods. However, it still loses its ability compared to our method, because the deep learning methods are very sensitive to test images even though only little amount of Poisson noises being introduced. Thus, the model-based methods work more reliable than deep learning methods in real scenarios. For the running speed, it can be seen that our SAV scheme converges much faster than the other variational method BCA, and also much faster than the unsupervised learning model NN. Fig. 16 provides two representative denoising results from the Set14 dataset. It is clear shown that our SAV method performs favorably in balancing noise removal and structure preservation.

V. CONCLUSION AND FUTURE WORKS

In this paper, we proposed an efficient and accurate SAV algorithm to minimize the mean curvature and Gaussian curvature energies for various image processing tasks. To the best of our knowledge, it is the first attempt to obtain the unconditionally energy diminishing algorithm for the curvature minimization problems. Compared to the existing ADMM method, our algorithm has the advantages of less parameters, faster convergence and more accurate results. Extensive numerical experiments on additive Gaussian denoising, image deblurring, single image super-resolution and mixed Gaussian and Poisson denoising are conducted on multiple image datasets, where our method has better performance in quality and speed compared to other variational methods.

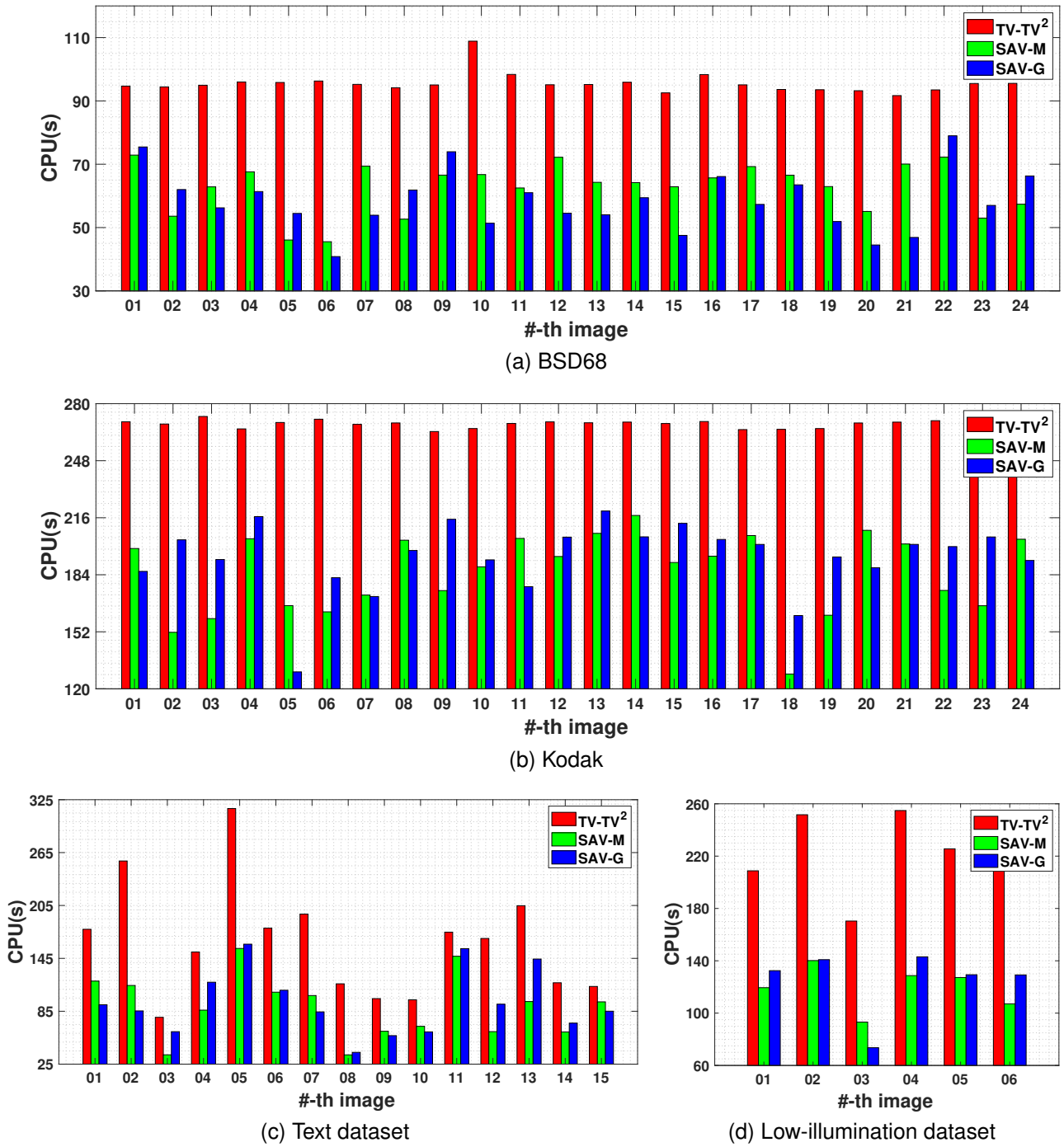


Fig. 15. CPU run time (in seconds) comparison on image deblurring problem, where the images are corrupted by 5×5 Gaussian blur kernel and noise level $\sigma = 5$.

Although it is unfair to compare the model-based methods with the learning-based methods, the experiments on Gaussian and mixed noise removal problems reveal that both model-based optimization methods and learning-based methods have their respective merits and drawbacks. Our method consumes more inference time compared to the supervised learning based methods such as DnCNN and CBDNet, but it does not require cumbersome time and resources for training millions of parameters as deep learning methods. Besides, the model-based methods also perform more reliable than both the supervised learning methods, which lose their effect when only little amount of Poisson noises are introduced into the noisy images. We believe that model driven and data driven methods can be integrated to obtain efficient and reliable algorithms, which are our direction for the future work.

REFERENCES

[1] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms" *Physica D: Nonlinear Phenomena*, vol. 60, no. 4, pp. 259–268, 1992.

[2] T. Wu, X. Gu, Z. Li, Z. Li, J. Niu, and T. Zeng, "Efficient boosted dc algorithm for nonconvex image restoration with rician noise," *SIAM Journal on Imaging Sciences*, vol. 15, no. 2, pp. 424–454, 2022.

Fig. 16. The visual comparison on mixed Gaussian and Poisson noise removal problems, where the images are degraded by both Poisson noise and Gaussian noise of $\sigma = 20$.

[3] T. Zhang, J. Chen, C. Wu, Z. He, T. Zeng, and Q. Jin, *Edge adaptive hybrid regularization model for image deblurring,* Inverse Problems, 2022. [Online]. Available: http://iopscience.iop.org/article/10. 1088/1361-6420/ac60bf

[4] H. Sun, X.-C. Tai, and J. Yuan, "Efficient and convergent preconditioned admm for the potts models," *SIAM Journal on Scientific Computing*, vol. 43, no. 2, pp. B455–B478, 2021.

[5] J. Xie, G. Hou, G. Wang, and Z. Pan, "A variational framework for underwater image dehazing and deblurring," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 6, pp. 3514– 3526, 2022.

[6] D. Marin, Y. Zhong, M. Drangova, and Y. Boykov, "Thin structure estimation with curvature regularization," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[7] H. Pei, B. Wei, K. Chang, C. Zhang, and B. Yang, "Curvature regularization to prevent distortion in graph embedding," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 20 779–20 790.

[8] S.-M. Moosavi-Dezfooli, A. Fawzi, J. Uesato, and P. Frossard, "Robustness via curvature regularization, and vice versa," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019, pp. 9070–9078.

[9] M. Yang, D. Xu, H. Chen, Z. Wen, and M. Chen, "Enhance curvature information by structured stochastic quasi-newton methods," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 10 654–10 663.

[10] B. Song, J. Zhou, and H. Wu, "Multi-stage curvature-guided network for progressive single image reflection removal," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2022.

[11] X. C. Tai, J. Hahn, and G. J. Chung, "A fast algorithm for euler's elastica model using augmented lagrangian method," *Siam Journal on Imaging Sciences*, vol. 4, no. 1, pp. 313–344, 2011.

[12] W. Zhu and T. Chan, "Image denoising using mean curvature of image surface," *SIAM Journal on Imaging Sciences*, vol. 5, no. 1, pp. 1–32,

2012.
[13] Q. Zhong, K. Yin, and Y. Duan, "Image reconstruction by minimizing curvatures on image surface," *Journal of Mathematical Imaging and Vision*, vol. 63, no. 1, pp. 30–55, 2021.
[14] Q. Zhong, Y. Li, Y. Yang, and Y. Duan, "Minimizing discrete total curvature for image processing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.
[15] C. Brito-Loeza, K. Chen, and V. Uc-Cetina, "Image denoising using the gaussian curvature of the image surface," *Numerical Methods for Partial Differential Equations*, vol. 32, no. 3, pp. 1066–1089, 2016.
[16] Y. Gong, "Mean curvature is a good regularization for image processing," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 8, pp. 2205–2214, 2018.
[17] W. Zhu, X.-C. Tai, and T. Chan, "A fast algorithm for a mean curvature based image denoising model using augmented lagrangian method," in *Efficient Algorithms for Global Optimization Methods in Computer Vision*, 2014, pp. 104–118.
[18] W. Zhu, X. C. Tai, and T. Chan, "Augmented lagrangian method for a mean curvature based image denoising model," *Inverse Problems & Imaging*, vol. 7, no. 4, pp. 1409–1432, 2017.
[19] H. Liu, X.-C. Tai, and R. Glowinski, "An operator-splitting method for the gaussian curvature regularization model with applications to surface smoothing and imaging," *SIAM Journal on Scientific Computing*, vol. 44, no. 2, pp. A935–A963, 2022.
[20] Y. Wang, W. Yin, and J. Zeng, "Global convergence of admm in nonconvex nonsmooth optimization," *Journal of Scientific Computing*, vol. 78, no. 1, pp. 29–63, 2019.
[21] M. Hong, Z.-Q. Luo, and M. Razaviyayn, "Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 337–364, 2016.
[22] W. Yao, J. Shen, Z. Guo, J. Sun, and B. Wu, "A total fractional- order variation model for image super-resolution and its sav algorithm," *Journal of Scientific Computing*, vol. 82, no. 81, 2020.
[23] C. Brito-Loeza and K. Chen, "Multigrid algorithm for high order denoising," *Siam Journal on Imaging Sciences*, vol. 3, no. 3, pp. 363–389, 2010.
[24] H. Liu, X.-C. Tai, and R. Glowinski, "An operator-splitting method for the gaussian curvature regularization model with applications to surface smoothing and imaging," *SIAM Journal on Scientific Computing*, vol. 44, no. 2, pp. A935–A963, Apr. 2022.
[25] J. Shen, J. Xu, and J. Yang, "The scalar auxiliary variable approach for gradient flows," *Journal of Computational Physics*, vol. 353, pp. 407–416, 2018.
[26] J. Shen, J. Xu, and J. Yang, "A new class of efficient and robust energy stable schemes for gradient flows," *SIAM Review*, vol. 61, no. 3, pp. 474–506, 2019.
[27] Z. Liu and X. Li, "The exponential scalar auxiliary variable (e-sav) approach for phase field models and its explicit computing," *SIAM Journal on Scientific Computing*, vol. 42, no. 3, pp. B630–B655, 2020.
[28] J. Yang and J. Kim, "An improved scalar auxiliary variable (sav) approach for the phase-field surfactant model," *Applied Mathematical Modelling*, vol. 90, pp. 11–29, 2021.
[29] X. Antoine, J. Shen, and Q. Tang, "Scalar auxiliary variable/lagrange multiplier based pseudospectral schemes for the dynamics of nonlinear Schrödinger/gross-pitaevskii equations," *Journal of Computational Physics*, pp. 110–128, 2021.
[30] Y. Fu, D. Hu, and Y. Wang, "High-order structure-preserving algorithms for the multi-dimensional fractional nonlinear Schrödinger equation based on the sav approach," *Mathematics and Computers in Simulation*, vol. 185, pp. 238–255, 2021.
[31] Q. Zhuang and J. Shen, "Efficient sav approach for imaginary time gradient flows with applications to one-and multi-component bose-einstein condensates," *Journal of Computational Physics*, vol. 396, pp. 72–88, 2019.
[32] L. Lin, Z. Yang, and S. Dong, "Numerical approximation of incompressible navier-stokes equations based on an auxiliary energy variable," *Journal of Computational Physics*, vol. 388, pp. 1–22, Jul. 2019.
[33] J. Shen and X. Yang, "Numerical approximations of allen-cahn and cahn-hilliard equations," *Discrete and Continuous Dynamical Systems*, vol. 28, no. 4, pp. 1669–1691, 2010.
[34] X. Yang, "Linear, first and second-order, unconditionally energy stable numerical schemes for the phase field model of homopolymer blends," *Journal of Computational Physics*, vol. 327, pp. 294–316, 2016.
[35] S. Roth and M. J. Black, "Fields of experts," *International Journal of Computer Vision*, vol. 82, no. 2, pp. 205–229, 2009.
[36] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *Curves and Surfaces. Springer Berlin Heidelberg*, 2012, pp. 711–730.
[37] L. Zhang, X. Wu, A. Buades, and X. Li, "Color demosaicking by local directional interpolation and nonlocal adaptive thresholding," *Journal of Electronic Imaging*, vol. 20, no. 2, p. 023016, 2011.
[38] J. Pan, Z. Hu, Z. Su, and M.-H. Yang, "l0 -regularized intensity and gradient prior for deblurring text images and beyond," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 2, pp. 342–355, 2017.
[39] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.
[40] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, "Toward convolutional blind denoising of real photographs," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
[41] D. Zheng, S. H. Tan, X. Zhang, Z. Shi, K. Ma, and C. Bao, "An unsupervised deep learning approach for real-world image denoising," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=tIjRAiFmU3y
[42] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, 2007.
[43] "Fast Single Image Super-Resolution Using a New Analytical Solution for $\ell_2 - \ell_2$ Problems," *IEEE Transactions on Image Processing*, vol. 25, no. 8, pp. 3683–3697, 2016.
[44] K. Papafitsoros and C. B. Schönlieb, "A combined first and second order variational approach for image reconstruction," *Journal of Mathematical Imaging and Vision*, vol. 48, no. 2, pp. 308–338, 2014.
[45] L. Calatroni, J. C. De Los Reyes, and C.-B. Schönlieb, "Infimal convolution of data discrepancies for mixed noise removal," *SIAM Journal on Imaging Sciences*, vol. 10, no. 3, pp. 1196–1233, 2017.
[46] J. Zhang, Y. Duan, Y. Lu, M. K. Ng, and H. Chang, "Bilinear constraint based admm for mixed poisson-gaussian noise removal," *Inverse Problems and Imaging*, no. 2, pp. 339–366, 2021.

Chenxin Wang is currently pursuing the Ph.D. degree with the Department of Mathematics in Hong Kong Baptist University. She received the B.S. and M.S. degrees from Tianjin University and National University of Singapore, respectively. Her research interests include image processing and machine learning.

Zhenwei Zhang is a PhD student in the Department of Mathematics of Tianjin University, Tianjin, China. He received the B.S. degree from Shanxi University, and the M.S. degree from Tianjin University. His research interests include computer vision, image processing, and machine learning.

Zhichang Guo received the Ph.D. degree from Jilin University, Changchun, China, in 2010. He is currently an associate professor with the School of Mathematics, Harbin Institute of Technology, Harbin, China. His research interests include partial differential equations, nonlinear analysis, and mathematical methods in image analysis.

Tieyong Zeng is a Professor at the Department of Mathematics, The Chinese University of Hong Kong (CUHK). Together with colleagues, he has founded the Center for Mathematical Artificial Intelligence (CMAI) since 2020 and served as the director of CMAI. He received the B.S. degree from Peking University, Beijing, China, the M.S. degree from Ecole Polytechnique, Palaiseau, France, and the Ph.D. degree from the University of Paris XIII, Paris, France, in 2000, 2004, and 2007, respectively. His research interests include image processing, optimization, artificial intelligence, scientific computing, computer vision, machine learning, and inverse problems. He is laureate of the 2021 Hong Kong Mathematical Society (HKMS) Young Scholars Award, due to the significant contributions in mathematical imaging and data science.

Yuping Duan received the Ph.D. degree from Nanyang Technological University, Singapore, in 2012. She is currently a professor with the Center for Applied Mathematics, Tianjin University, Tianjin, China. Her research interests are image processing, numerical optimization, and surgical simulation.