



A Ginzburg-Landau- H^{-1} Model and Its SAV Algorithm for Image Inpainting

Xiangyu Bai¹ · Jiebao Sun¹ · Jie Shen² · Wenjuan Yao¹ · Zhichang Guo¹

Received: 10 June 2022 / Revised: 23 March 2023 / Accepted: 5 May 2023 /
Published online: 15 June 2023

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract

Image inpainting models and the corresponding numerical algorithms play key roles in image processing. At present, the visual output of the oscillatory inpainting area is usually not natural. In this paper, we propose an image inpainting model based on the Ginzburg-Landau functional and H^{-1} -norm. In the model, the H^{-1} -fidelity term performs well in preserving the edges of the oscillatory inpainting areas, and the Ginzburg-Landau functional can provide additional geometric content. Theoretically, we prove the existence of the minimizer for the proposed energy functional. Based on the scalar auxiliary variable approach, we develop an efficient numerical scheme to solve the proposed model. Further, we use a time step adaptive strategy to accelerate the convergence. Experimental results validate the effectiveness of the proposed algorithm for image inpainting.

Keywords Image inpainting · Ginzburg-Landau functional · H^{-1} -norm · Oscillatory inpainting areas · Scalar auxiliary variable

Mathematics Subject Classification 65N20

1 Introduction

Image inpainting refers to the reconstruction of the contaminated and missing parts of an image using information from the surrounding areas, which is essentially an ill-posed inverse problem. In real life, the damage to binary images such as fax documents, digital signatures and contract invoices is common, and the goal of image inpainting is to restore these images as closely as possible to match the human visual perception. Image inpainting technology can be used in a wide range of fields, including the removal of redundant objects from precious photographs, the inpainting of collections of calligraphies and paintings, and extended to biomedical, aerospace, etc.

✉ Wenjuan Yao
mathywj@hit.edu.cn

¹ School of Mathematics, Harbin Institute of Technology, Harbin, China

² Department of Mathematics, Purdue University, West Lafayette, IN 47907, USA

Existing image inpainting methods can be roughly classified into four categories: the first is exemplar-based inpainting approaches [1, 2]; the second is sparse representation-based inpainting approaches [4, 5]; the third is deep learning based inpainting methods [6, 8, 9]; while the fourth is the PDE based approach [10–25].

The exemplar-based inpainting approach is based on the influential work of Efros and Leung [1]. Such techniques use statistical characteristics of the image and self-similarity as prior information. The texture inpainting is accomplished by learning the inpainting domain from the known parts of the texture. Learning includes the process of sampling, and copying or stitching together exemplars (patches extracted from known parts of an image). These methods are usually useful for texture images, but they do not perform very well in preserving the structures or edges of the images [3].

Sparse representation-based inpainting methods [4, 5] assume that unknown and known regions of the image have the same sparsity, so unknown regions can be sparsely represented by dictionaries. After calculating the sparse representation of the image in the overcomplete dictionary, the image is reconstructed by the corresponding dictionary and sparse coding. These methods are good for images with abundant known information and regular changes, but quite often, they are not very suitable for images with large damaged areas [3].

Pathak et al. [6] proposed an unsupervised Context Encoder model, which is an earlier image inpainting model based on deep learning. Since then, researchers had proposed improved inpainting models based on Convolutional Neural Networks and Generative Adversarial Networks [7–9]. This kind of methods can generate the matching content in the inpainting domain through network training. However, it requires a high computer hardware configuration, large training sets and long training time.

In the seminal work of Bertalmio et al. [10], they introduced the idea of the PDE into digital image inpainting technology and proposed the BSCB inpainting model. The essential idea is to spread information around the inpainting areas smoothly inward along the direction of the isophotes. The process of image inpainting is interleaved with the anisotropic diffusion process to ensure the proper evolution of the direction field. In subsequent work, variational models originally used for image denoising and segmentation tasks [11, 12] were also used for image inpainting [13, 14]. In the variational inpainting method, the classical total variation (TV) denoising model [11] was applied to the inpainting of nontextured images through simple adjustment [13], and the function of the fidelity term in the model was limited to the complement of the inpainting area. Similarly, Esedoglu and Shen extended the Mumford-Shah (MS) segmentation model [12] to the MS inpainting model [14]. The TV inpainting model and the MS inpainting model are second-order PDE inpainting models, which have effective inpainting for some small-scale damaged images but fail to meet the connectivity principle in the connection of some damaged edges (e.g. circle). The TV inpainting model also leads to the staircasing effect. To this end, a third-order PDE inpainting model based on curvature-driven diffusions (CDD) was proposed in [15], which can realize the inpainting of images with larger damaged areas than the two second-order PDE inpainting models. However, the inpainting effect of the CDD inpainting model is not ideal when the unknown region has corners. In order to achieve better results, more and more researchers are considering appropriate high-order PDE inpainting models for image inpainting.

Inspired by Mumford [16], a fourth-order variational inpainting approach was considered in [17] to overcome some shortcomings of low-order inpainting models. Chan et al. combined the CDD inpainting model [15] and the transport process [10] to smoothly connect isophotes across large inpainting areas, for which they proposed the Euler's elastica (EE) inpainting model. Due to the deficiency of the MS inpainting model, Esedoglu and Shen [14] proposed the Mumford-Shah-Euler (MSE) inpainting model with better performance and the ability to

restore missing edges more smoothly than the MS inpainting model. They approximated the proposed variational problem by applying the conjecture of De Giorgi [18] and solved it using the gradient descent method. Accordingly, a fourth-order nonlinear parabolic equation was obtained, which is similar to the Cahn–Hilliard equation [19]. But this fourth-order equation and corresponding numerical computation are relatively complex. To this end, a pioneering work was proposed by Bertozzi et al. [20, 21] with a fourth-order equation for binary image inpainting based on the Cahn–Hilliard equation. The modified Cahn–Hilliard (mCH) equation has many advantages of the MSE inpainting model, especially, it can match image intensity and direction of edges. Furthermore, it is simpler and requires less computation than the MSE inpainting model [21]. In various subsequent works [23–25], the image inpainting model based on the Cahn–Hilliard equation was further improved. In [23], Bertozzi et al. suggested a generalization for grayscale image inpainting, they split the grayscale image bitwise into some channels, and each binary channel was applied to the mCH equation inpainting model.

The main challenge of image inpainting using high-order equations is to develop suitable, simple models and efficient discrete schemes corresponding to the models. In particular, if the model can be cast into a gradient flow, then the many efficient techniques developed for gradient flows can be adopted. At present, numerical methods for solving gradient flow problems include the convex splitting method [26–28], stabilization method [29, 30], Lagrange multiplier method [31], invariant energy quadratization (IEQ) method [32, 33] and scalar auxiliary variable (SAV) method [34, 35], etc. Among them, the IEQ method is an efficient scheme to construct energy stability. Its disadvantages are that the elliptic equations with complex variable coefficients need to be solved at each time step, and a coupling system is caused in gradient flow with multiple components [34]. In order to overcome these shortcomings of the IEQ method, Shen et al. [34] proposed the SAV method. The SAV method essentially only needs to solve the decoupling equation with constant coefficients at each time step. It inherits all the advantages of the IEQ method, and is much more efficient and easy to implement. It has been successfully applied in many fields, for instance, materials science and fluid dynamics [36–39].

In order to better achieve the natural visual output of topologically complex (non-simply connected) inpainting areas and oscillatory areas, we propose a variational inpainting model based on the Ginzburg–Landau functional and the H^{-1} -norm. In particular, this model can be interpreted as a gradient flow. We then adopt the SAV approach to develop a numerical algorithm for the proposed variational inpainting model to meet the high efficiency requirements of image processing.

Specifically, the proposed variational inpainting approach has the following advantages:

- The Ginzburg–Landau functional can provide additional geometric information and the H^{-1} -fidelity term performs well in preserving the edges of the oscillatory areas. Combining the Ginzburg–Landau functional and the H^{-1} -norm, we propose a variational inpainting model that inherits the advantages of both.
- The existence of the minimizer for the proposed variational inpainting model is proved.
- An efficient and easy to implement numerical scheme based on the SAV algorithm is constructed for this proposed model, and a time step adaptive strategy is used to accelerate the convergence.
- The proposed model performs well in the oscillatory areas and topologically complex areas. In terms of output efficiency and applicability, our proposed method outperforms some classical PDE-based inpainting methods. The proposed method can also be applied to grayscale image inpainting.

The rest of the article is organized as follows. In Sect. 2, we introduce some fundamental notations and definitions. Related work and a variational inpainting model are dedicated in Sect. 3. We describe the specific discrete scheme in Sect. 4. Section 5 is presented the numerical results and experimental analysis. We conclude this paper in Sect. 6.

2 Preliminaries

First of all, we introduce some notations to be used in this paper. The L^2 inner product between the functions u and v is defined as: $(u, v) = \int_{\Omega} u(x)v(x)dx$ and $\|u\| = (u, u)^{\frac{1}{2}}$ represents the L^2 norm of u .

Lemma 1 [44] *Let $V_0 = \{u \in H^1(\Omega) : \int_{\Omega} u(x)dx = 0\}$. If $g \in L^2(\Omega)$, with $\int_{\Omega} g(x)dx = 0$, then the following problem:*

$$-\Delta u = g,$$

with homogeneous Neumann or periodic boundary condition admits a unique solution u in V_0 .

Lemma 1 states that if g satisfies the conditions $g \in L^2(\Omega)$ and $\int_{\Omega} g(x)dx = 0$, then there exists a unique solution u . This solution can be expressed as $u = \Delta^{-1}g$, where Δ^{-1} denotes the inverse of the negative Laplace operator with homogeneous Neumann or periodic boundary conditions. For later use, we are now to introduce the Sobolev space $H^{-1}(\Omega)$ here, which can be defined by $H^{-1}(\Omega) = \{F \in H^1(\Omega)^* \mid \langle F, 1 \rangle_{(H^1)^*, H^1} = 0\}$, endowed with the inner product $(v_1, v_2)_{-1} = (\nabla(\Delta^{-1})v_1, \nabla(\Delta^{-1})v_2)$ and together the corresponding norm $\|v_1\|_{-1} = (v_1, v_1)_{-1}^{\frac{1}{2}}$.

Now let $E(u)$ represent a free energy functional whose variational derivative is denoted by $\mu = \delta E / \delta u$. We consider the following general form of gradient flow:

$$\frac{\partial u}{\partial t} = \mathcal{G}\mu,$$

with suitable boundary conditions, and \mathcal{G} denotes a non-positive symmetric operator which provides the dissipation mechanism. In particular, $\mathcal{G} = -I$ for the L^2 gradient flow, $\mathcal{G} = \Delta$ for the H^{-1} gradient flow, and $\mathcal{G} = -(-\Delta)^{\alpha}$ ($0 < \alpha < 1$) for the non-local $H^{-\alpha}$ gradient flow. As long as \mathcal{G} is non-positive, the corresponding energy satisfies a dissipation law, i.e.,

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

3 The New Model and its Analysis

In this section, we first present some related work, then we propose a new variational model for image inpainting based on the Ginzburg-Landau functional and the H^{-1} -norm, and prove the existence of the solution for the proposed inpainting model.

3.1 Related Work

In terms of image inpainting for preserving edge and curvature problems, high-order PDE models usually perform better than second-order PDE models [40]. Specifically, for image

inpainting problems, high-order derivatives can provide supplementary information on the isoline directions and high-order PDE models can reproduce the “high-order” (curvature, corners, etc) features [41]. In this paper, we also consider using a high-order variational model for image inpainting.

The phase field approach was first proposed by Van der Waals [42]. Its key idea is to represent a sharp interface between two substances (one substance is 1 and the other is 0) by a smooth “phase” function $\phi(x)$, with a fast transition from 1 to 0 at the interface. The phase field function can be obtained by minimizing a corresponding energy functional. A typical example is the Ginzburg-Landau functional [19]:

$$E_1(u) = \int_{\Omega} \left(\frac{\epsilon}{2} |\nabla u|^2 + \frac{1}{\epsilon} F(u) \right) dx, \tag{1}$$

where F denotes the double-welled potential energy $F(u) = u^2(u - 1)^2$. The two wells correspond to u value taken by great majority grayscale values in the image inpainting model [23]. A potential with wells is selected as 0 (black) and 1 (white). The parameter ϵ plays a vital role in the image inpainting problem and it can be regarded as a measure of the transition domain between two grayscale states [20]. Moreover, a fourth-order PDE is obtained by gradient descent with respect to the H^{-1} inner product for the Ginzburg–Landau functional (1), leading to the Cahn–Hilliard equation. Hence, the Ginzburg–Landau functional is suitable for binary image inpainting.

We recall that the fourth-order mCH equation inpainting model in [20, 21] is as follows:

$$\begin{aligned} u_t &= -\Delta \left(\epsilon \Delta u - \frac{1}{\epsilon} F'(u) \right) + \lambda(x)(f - u), & \text{in } \Omega \times (0, T), \\ u(0, x) &= f(x), & \text{in } \Omega, \\ \frac{\partial u}{\partial n} &= \frac{\partial \Delta u}{\partial n} = 0, & \text{on } \partial\Omega \times (0, T), \end{aligned} \tag{2}$$

where f is the given binary image and $F'(u) = 4u^3 - 6u^2 + 2u$. The characteristic function $\lambda(x)$ is represented by $\lambda(x) = \begin{cases} 0, & x \in D \\ \lambda_0, & x \in \Omega \setminus D \end{cases}$, and $D \subset \Omega$ is the inpainting region. The original Cahn–Hilliard equation is the H^{-1} gradient flow for the energy (1), while the second term in the right-hand side of the Eq. in (2) is derived by the L^2 gradient flow for the energy:

$$E_2(u) = \int_{\Omega} \lambda(x)(f - u)^2 dx. \tag{3}$$

The mCH Eq. (2) was regarded as a superposition of different gradient descents for two energies (1) and (3) in [20, 21]. It is not a real gradient descent for the sum of the energies (1) and (3), no matter with respect to the inner product of L^2 or H^{-1} [21].

3.2 Two Classical Norms in the Fidelity Term

In the fidelity term of the image inpainting model, u can fit f under either the L^2 -norm or the H^{-1} -norm, where u is an inpainting version of f which is a given image in the area Ω . In this paper, we use the H^{-1} -norm in the construction of fidelity term. In fact, the H^{-1} -norm has been used as a suitable norm of functional space for image inpainting [41, 44, 45]. Compared with the L^2 -fidelity term adopted in most existing work [13, 14, 17], the H^{-1} -fidelity term owns the ability to preserve the edges of the oscillatory areas.

For image decomposition and image restoration problems, Osher et al. proposed a variational model by using a weak norm for the fidelity term [43], which is based on a norm for oscillatory functions introduced by Meyer [46] involving the H^{-1} -norm. In [43], the authors found that the H^{-1} -fidelity term was better than the L^2 -fidelity term for reproducing textured or oscillatory patterns due to the fact that the H^{-1} -norm allows the existence of oscillatory functions. Furthermore, they proposed a variational model based on the H^{-1} -fidelity term, which performed well in preserving the edges of oscillatory patterns.

To study the superiority of the H^{-1} -norm in the inpainting effect of oscillatory areas, we compare the inpainting effect subject to the two different fidelity terms, i.e., the L^2 - and H^{-1} -fidelity term. In order to present the character of the H^{-1} -fidelity term in the image inpainting task, we consider the inpainting of two oscillatory patterns. In the comparative experiment, we take the regularization term as $\frac{1}{2} \int_{\Omega} |\nabla u|^2 dx$. For comparison, we consider two different fidelity terms in the variational model, which are expressed as follows:

$$\|\lambda(x)(u - f)\|^2, \tag{4}$$

and

$$\|\lambda(x)(u - f)\|_{-1}^2, \tag{5}$$

i.e., the function u approximates f in the sense of L^2 and H^{-1} in the given region, respectively. A finite-difference scheme is used for the numerical scheme of this experiment. The results are shown in Fig. 1.

In Figs. 1a and d, the gray area is the inpainting area. As shown in Figs. 1b, c and 1e, f, the inpainting results using the H^{-1} -fidelity term (5) in the variational model are superior compared to the results using the L^2 -fidelity term (4). In particular, the H^{-1} -fidelity term in the image inpainting model performs better than the L^2 -fidelity term in preserving the edges of the oscillatory areas. Hence, it is advantageous to use the H^{-1} -fidelity term in the variational image inpainting model.

3.3 The New Model and its Properties

Noticing that the high-order PDE models facilitate the reproduction of the “high-order” features (corners and curvature), and high-order derivatives can provide additional information on the isoline direction, we shall use the high-order regularization term in the proposed image inpainting model.

Since the H^{-1} -fidelity term is beneficial in preserving the edges of the oscillatory inpainting areas, we adopt the H^{-1} -norm in the fidelity term, i.e.,

$$R = \frac{1}{2} \|\lambda(x)(u - f)\|_{-1}^2, \tag{6}$$

where

$$\lambda(x) = \begin{cases} \lambda_0 & \text{in } \Omega \setminus D \\ 0 & \text{in } D \end{cases},$$

and $D \subset \Omega$ denotes the inpainting area.

In order to better accomplish image inpainting for oscillatory regions and topologically complex regions, we propose the following variational model by combining the Ginzburg-Landau functional together with the H^{-1} -fidelity term:

$$\min_{u \in H^1(\Omega)} E(u) := \int_{\Omega} \left(\frac{\epsilon}{2} |\nabla u|^2 + \frac{1}{\epsilon} F(u) \right) dx + \frac{1}{2} \|\lambda(x)(u - f)\|_{-1}^2, \tag{7}$$

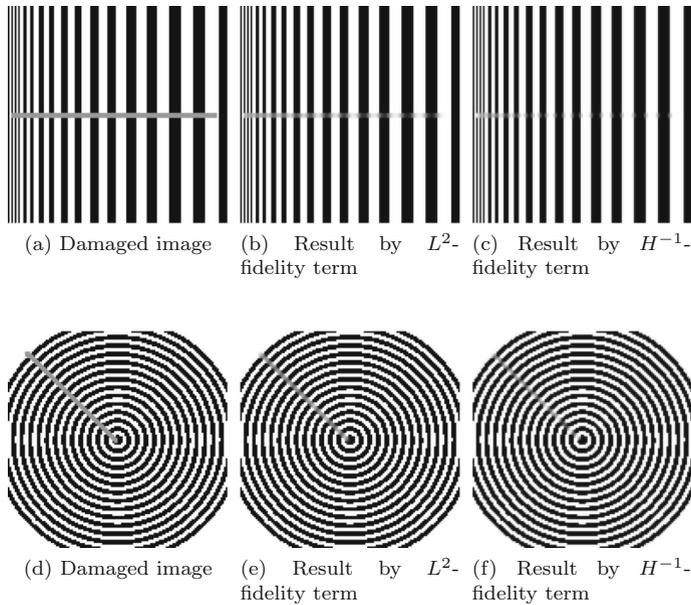


Fig. 1 Inpainting results by two different norms in the fidelity term ($\lambda_0 = 100$)

where $\epsilon > 0$ and $F(u) = u^2(u - 1)^2$.

It is not difficult to obtain the variation derivative μ of E , i.e.,

$$\mu := \frac{\delta E}{\delta u} = -\epsilon \Delta u + \frac{1}{\epsilon} F'(u) + \lambda(x)^2 \Delta^{-1}(u - f).$$

Hence the gradient descent with respect to the H^{-1} inner product for (7) is

$$u_t = \Delta \left(-\epsilon \Delta u + \frac{1}{\epsilon} F'(u) \right) + \lambda(x)^2 (f - u), \text{ in } \Omega \times (0, T), \tag{8}$$

subjected to the homogeneous Neumann or periodic boundary conditions, and the initial condition $u(0, x) = f(x)$ in Ω . Therefore, the fourth-order PDE Eq. (8) is the H^{-1} gradient flow with respect to (7).

Note that form of the gradient flow derived from the new variational problem (7) is the same as the Eq. (2). However, there are two essential differences between our work and the work of [20, 21]. Firstly, the proposed model (7) is essentially a variational model rather than the diffusion based model (2). We construct the model by designing a proper fidelity term to protect texture and topologically complex regions. Differently, the model (2) is derived from the perspective of constructing diffusion equations. They integrate H^{-1} and L^2 gradient flows, rather than considering them in a unified gradient flow. Secondly, the gradient flow structure of our approach enables us to use numerical methods developed for gradient flows, in particular, we adopt the efficient SAV approach, which is particularly developed for gradient flows, to solve (8), while the convex splitting method was used in [20, 21].

Remark 1 Integrating on both sides of the Eq. (8) over the domain Ω , thanks to the homogeneous Neumann or periodic boundary conditions, we derive that $\frac{d}{dt} \int_{\Omega} u dx = 0$. Therefore, in our model, u and f satisfy $\int_{\Omega \setminus D} (f - u) dx = 0$.

Theorem 1 For $f \in L^2(\Omega)$, there exists a solution $u \in H^1(\Omega)$ for the minimization problem

$$\inf_{u \in H^1(\Omega)} E(u) = \int_{\Omega} \left(\frac{\epsilon}{2} |\nabla u|^2 + \frac{1}{\epsilon} F(u) \right) dx + \frac{1}{2} \|\lambda(x)(u - f)\|_{-1}^2.$$

Proof Let u_k be a minimizing sequence of the non-negative functional $E(u)$, then there is a constant $M > 0$ such that $\int_{\Omega} |\nabla u_k|^2 \leq M^2$ for all $k \geq 0$. Moreover, $E(u_k)$ is bounded for $k \geq k_0$ for some k_0 . We can see that there exists a positive constant C such that

$$\begin{aligned} \int_{\Omega} u_k^2 dx &= \int_{\Omega \cap \{|u_k| > 2\}} u_k^2 dx + \int_{\Omega \cap \{|u_k| \leq 2\}} u_k^2 dx \\ &\leq \int_{\Omega \cap \{|u_k| > 2\}} (u_k)^2 (u_k - 1)^2 dx + 4|\Omega| \\ &\leq \int_{\Omega} F(u_k) dx + 4|\Omega| \leq C, \end{aligned}$$

we obtain that u_k is uniformly bounded in $L^2(\Omega)$, i.e., u_k is uniformly bounded in $H^1(\Omega)$: there exists a constant C' such that

$$\|u_k\|_{H^1(\Omega)} = \|u_k\|_{L^2(\Omega)} + \|\nabla u_k\|_{L^2(\Omega)} \leq C'.$$

Combined with the compact embedding $H^1(\Omega) \hookrightarrow L^2(\Omega)$, there exists a subsequence of u_k , still denoted u_k , and $u \in H^1(\Omega)$, such that

$$u_k \rightharpoonup u \text{ in } H^1(\Omega), u_k \rightarrow u \text{ a.e. } \Omega, u_k \rightarrow u \text{ in } L^2(\Omega), \text{ for } k \rightarrow \infty.$$

By the weak lower semicontinuity, it holds that

$$\int_{\Omega} |\nabla u|^2 dx \leq \liminf_{k \rightarrow \infty} \int_{\Omega} |\nabla u_k|^2 dx.$$

According to Fatou's lemma, we deduce

$$\int_{\Omega} u^2 (u - 1)^2 dx = \int_{\Omega} \liminf_{k \rightarrow \infty} (u_k)^2 (u_k - 1)^2 dx \leq \liminf_{k \rightarrow \infty} \int_{\Omega} (u_k)^2 (u_k - 1)^2 dx.$$

Since the embedding $L^2(\Omega) \hookrightarrow H^{-1}(\Omega)$ is continuous, we get

$$\|u\|_{-1} \leq C \|u\|_2.$$

We also have

$$u_k - f \rightarrow u - f \text{ in } L^2(\Omega) \text{ for } k \rightarrow \infty.$$

Further,

$$u_k - f \rightarrow u - f \text{ in } H^{-1}(\Omega) \text{ for } k \rightarrow \infty.$$

Combining the above analysis, it is not difficult to obtain the fact that

$$E(u) \leq \liminf_{k \rightarrow \infty} E(u_k),$$

therefore, u is a minimizer of E in $H^1(\Omega)$.

4 The Numerical Algorithm

In this section, we develop an efficient and easy-to-implement scheme based on the SAV method, and we use an adaptive time stepping strategy to improve the efficiency.

4.1 SAV Scheme for Solving the Proposed Model

The mCH equation inpainting model (2) is generally solved by the convex splitting approach, which was proposed by [47] and developed by [48]. The idea of the convex splitting approach is to split the energy functional as the sum of a convex functional and a concave functional, which is normally used to solve gradient flows. Specifically, the original Cahn–Hilliard equation is regarded as the H^{-1} gradient flow of the energy $E_1(u)$ in (1), while the fitting term in (2) is a gradient flow in L^2 for the energy $E_2(u)$ in (3). They use convex splitting for $E_1(u)$ and $E_2(u)$ by $E_1(u) = E_{11}(u) - E_{12}(u)$ and $E_2(u) = E_{21}(u) - E_{22}(u)$, respectively where

$$E_{11}(u) = \int_{\Omega} \frac{\epsilon}{2} |\nabla u|^2 + \frac{C_1}{2} |u|^2 dx, \quad E_{12}(u) = \int_{\Omega} -\frac{1}{\epsilon} F(u) + \frac{C_1}{2} |u|^2 dx,$$

and

$$E_{21}(u) = \int_{\Omega \setminus D} \frac{C_2}{2} |u|^2 dx, \quad E_{22}(u) = \int_{\Omega \setminus D} -\lambda_0 (f - u)^2 + \frac{C_2}{2} |u|^2 dx.$$

From the discussed above, the convex splitting scheme reads

$$\frac{u^{n+1} - u^n}{\Delta t} = -\nabla_{H^{-1}} \left(E_{11}^{n+1}(u) - E_{12}^n(u) \right) - \nabla_{L^2} \left(E_{21}^{n+1}(u) - E_{22}^n(u) \right),$$

where $\nabla_{H^{-1}}$ and ∇_{L^2} denote gradient descent in regard to the H^{-1} inner product, and the L^2 inner product, respectively, let $t^n = n\Delta t$ for $0 \leq n \leq N = \lceil T/\Delta t \rceil$ with $\Delta t > 0$ the time step size and u^n denotes the numerical approximation to $u(\cdot, t)|_{t=t^n}$. This produces the following numerical scheme:

$$\begin{aligned} \frac{u^{n+1} - u^n}{\Delta t} + \epsilon \Delta^2 u^{n+1} - C_1 \Delta u^{n+1} + C_2 u^{n+1} \\ = \Delta \left(\frac{1}{\epsilon} F'(u^n) \right) + \lambda(x) (f(x) - u^n) - C_1 \Delta u^n + C_2 u^n. \end{aligned} \tag{9}$$

The constants C_1 and C_2 are chosen to be large enough such that $E_{11}(u)$, $E_{12}(u)$, $E_{21}(u)$ and $E_{22}(u)$ are convex. The convex splitting method can solve the inpainting model (2), but as it stands, extra parameters such as C_1 and C_2 are introduced and the time is lost due to adjusting parameters. Furthermore, the convex splitting method involves solving a nonlinear system at each time step.

Taking the disadvantages of the convex splitting method into account, we adopt a recent proposed method, i.e., SAV algorithm to solve the proposed inpainting model (7). The SAV algorithm is a powerful technology that is suitable for a large class of gradient flow. According to Sect. 2, the H^{-1} gradient flow can be written as:

$$\frac{\partial u}{\partial t} = \Delta \mu. \tag{10}$$

Define $L = -\epsilon \Delta$ and consider the following free energy functional

$$E(u) = \frac{1}{2}(u, Lu) + \tilde{E}_1(u), \tag{11}$$

which consists of the linear part and the nonlinear part $\tilde{E}_1(u)$. It is not difficult to find that L is a non-negative symmetric linear operator independent of u . In addition, the free energy $\tilde{E}_1(u)$ in the SAV algorithm is required to be bounded from below, i.e., $\tilde{E}_1(u) \geq C_0 > 0$ holds for an existential constant C_0 . We can define this free energy as:

$$\tilde{E}_1(u) = \int_{\Omega} F_1(u) dx + C_0, \tag{12}$$

where

$$F_1(u) = \frac{u^2(u-1)^2}{\epsilon} + \frac{1}{2}(\nabla(\Delta^{-1})\lambda(x)(u-f))^2, \tag{13}$$

and C_0 can be selected to be a sufficiently small positive number. Next, we use the SAV algorithm to solve the proposed minimization energy functional problem. In view of (11), we consider the total energy functional

$$E(u) = \int_{\Omega} \left[\frac{\epsilon}{2} |\nabla u|^2 + F_1(u) \right] dx + C_0.$$

Hence, the corresponding H^{-1} gradient flow takes the following form:

$$\frac{\partial u}{\partial t} = \Delta \mu, \tag{14}$$

$$\mu = \delta E / \delta u = -\epsilon \Delta u + F'_1(u), \tag{15}$$

where

$$F'_1(u) = \frac{4u^3 - 6u^2 + 2u}{\epsilon} + \lambda^2(x) (\Delta^{-1})(u - f).$$

We introduce a scalar auxiliary variable:

$$r = \sqrt{\tilde{E}_1(u)}, \tag{16}$$

and expand the gradient flow (14) and (15) as follows:

$$\frac{\partial u}{\partial t} = \Delta \mu, \tag{17}$$

$$\mu = -\epsilon \Delta u + \frac{r}{\sqrt{\tilde{E}_1(u)}} F'_1(u), \tag{18}$$

$$\frac{dr}{dt} = \frac{1}{2\sqrt{\tilde{E}_1(u)}} \int_{\Omega} F'_1(u) u_t \, dx. \tag{19}$$

Since we are interested in the steady state solution, inspired by [49], we employ the following modified first-order SAV scheme:

$$\frac{u^{n+1} - u^n}{\Delta t} = \Delta \mu^{n+1}, \tag{20}$$

$$\mu^{n+1} = -\epsilon \Delta u^{n+1} + \frac{\bar{r}^{n+1}}{\sqrt{\tilde{E}_1(u^n)}} F'_1(u^n), \tag{21}$$

$$\frac{\bar{r}^{n+1} - r^n}{\Delta t} = \frac{1}{2\sqrt{\tilde{E}_1(u^n)}} \int_{\Omega} F'_1(u^n) \frac{u^{n+1} - u^n}{\Delta t} dx; \tag{22}$$

$$r^{n+1} = \sqrt{\tilde{E}_1(u^{n+1})}. \tag{23}$$

If we plug (21) and (22) in (20), we can eliminate \bar{r}^{n+1} to obtain

$$\frac{u^{n+1} - u^n}{\Delta t} = \Delta \left\{ -\epsilon \Delta u^{n+1} + \frac{F'_1(u^n)}{\sqrt{\tilde{E}_1(u^n)}} \left[r^n + \int_{\Omega} \frac{F'_1(u^n)}{2\sqrt{\tilde{E}_1(u^n)}} (u^{n+1} - u^n) dx \right] \right\}. \tag{24}$$

Rearranging the above equation yields

$$\begin{aligned} (I + \epsilon \Delta t \Delta^2) u^{n+1} - \frac{\Delta t}{2} \Delta b^n (b^n, u^{n+1}) \\ = u^n + \Delta t r^n \Delta b^n - \frac{\Delta t}{2} (b^n, u^n) \Delta b^n := g^n, \end{aligned} \tag{25}$$

where

$$b^n = \frac{F'_1(u^n)}{\sqrt{\tilde{E}_1(u^n)}}.$$

Multiplying (25) with A^{-1} , then taking the inner product with b^n , we find

$$(b^n, u^{n+1}) - \frac{\Delta t}{2} (A^{-1} \Delta b^n, b^n) (b^n, u^{n+1}) = (A^{-1} g^n, b^n), \tag{26}$$

where

$$A = I + \epsilon \Delta t \Delta^2.$$

Therefore, we can determine (b^n, u^{n+1}) from the above that

$$(b^n, u^{n+1}) = \frac{(A^{-1} g^n, b^n)}{1 - \frac{\Delta t}{2} (A^{-1} \Delta b^n, b^n)}. \tag{27}$$

Finally, combining the above with (25), we know that

$$u^{n+1} = A^{-1} \left(g^n + \frac{\Delta t}{2} (b^n, u^{n+1}) \Delta b^n \right). \tag{28}$$

Remark 2 In summary, each time step of the scheme (20)–(22) can be implemented as:

- (i) Compute b^n , and g^n defined by the righthand side of (25);
- (ii) With b^n and g^n known, we can compute (b^n, u^{n+1}) from (27);
- (iii) Finally, we solve u^{n+1} from (28) by determining (b^n, u^{n+1}) .

Therefore, only two linear equations with constant coefficients need to be solved at each time step according to (i) and (iii), the form of two linear systems as follows:

$$(I + \epsilon \Delta t \Delta^2) \bar{x} = \bar{b}.$$

Hence, the above SAV scheme is efficient and easy to implement.

Taking the L^2 inner product of (20), (21) and (22) with μ^{n+1} , $\frac{u^{n+1} - u^n}{\Delta t}$ and $2\bar{r}^{n+1}$ respectively, we obtain immediately the following:

Theorem 2 *The first-order scheme (20)–(22) satisfies a modified energy dissipation law as follows:*

$$\begin{aligned} & \frac{1}{\Delta t} [\bar{E}(u^{n+1}, \bar{r}^{n+1}) - \bar{E}(u^n, r^n)] \\ & + \frac{1}{\Delta t} \left[\frac{\epsilon}{2} (u^{n+1} - u^n, -\Delta(u^{n+1} - u^n)) + (\bar{r}^{n+1} - r^n)^2 \right] \\ & = -(\mu^{n+1}, \mu^{n+1}), \end{aligned}$$

where $\bar{E}(u, r) = -\frac{\epsilon}{2}(u, \Delta u) + r^2$ is the modified energy.

Remark 3 The above result does not mean that the first-order scheme (20)–(23) is unconditionally energy stable. However, if we set $r^{n+1} = \bar{r}^{n+1}$ in (23) as in the original SAV approach [34], the scheme becomes unconditionally energy stable in the sense that $\bar{E}(u^{n+1}, r^{n+1}) \leq \bar{E}(u^n, r^n)$.

4.2 Adaptive Time Stepping Strategy

In order to achieve numerical results efficiently and accurately, an adaptive time stepping strategy can be applied in the case where the scheme meets the unconditional energy stability [50, 51]. Correspondingly, we construct an adaptive time stepping algorithm for our SAV scheme. Let tol be a reference tolerance, e be the relative error of energy, and ρ be a default safety coefficient. We can apply the following formula to update the time step size,

$$A_{d\rho}(e, \tau) = \rho \left(\frac{tol}{e} \right)^{1/2} \tau.$$

In this paper, we set $\tau_{\min} = 10^{-4}$ and $\tau_{\max} = 10^{-1}$, where τ_{\min} and τ_{\max} denote the minimum and the maximum time step, respectively. We choose τ_{\min} as the initial time step. The procedure of the strategy is described as follows.

Algorithm 1 The procedure of time step adaptive strategy

- Input:** $\epsilon, \lambda, tol, \rho, \tau_{\min}, \tau_{\max}$;
 1: Calculate u^{n+1} by the SAV scheme with time step τ_n ;
 2: Update $E^{n+1} = E(u^{n+1})$;
 3: Compute $e_{n+1} = |E^n - E^{n+1}|/|E^{n+1}|$;
 4: **if** $e_{n+1} > tol$ **then**
 5: Recalculate $\tau_n = \max\{\tau_{\min}, \min\{A_{d\rho}(e_{n+1}, \tau_n), \tau_{\max}\}\}$ and go to step 1;
 6: **else**
 7: Update $\tau_{n+1} = \max\{\tau_{\min}, \min\{A_{d\rho}(e_{n+1}, \tau_n), \tau_{\max}\}\}$;
 8: **end if**
 9: **return** τ_{n+1} ;
-

4.3 Space Discretization of the Numerical Algorithm

In this subsection, we describe the space discretization of the SAV algorithm (20)–(22) at each iteration. We assume periodic boundary conditions, the Neumann boundary conditions can be treated similarly. The input discrete image u is assumed to be $M \times L$ pixels, and

such that $u(i, j) = u(i \Delta x_1, j \Delta x_2)$ for $i = 0, 1, \dots, M - 1, j = 0, 1, \dots, L - 1, \Delta x_1$ and Δx_2 denote the grid size, and we adopt $\Delta x_1 = \Delta x_2 = 1$ in the forthcoming tests. According to the scheme (20)–(22), we introduced the calculation of $\tilde{E}_1(u^n), F'_1(u^n)$ and u^{n+1} . Let $y = \Delta^{-1}U$, where $U = \lambda(x)(u - f)$, it is obvious that once we work out $\Delta^{-1}U$ i.e., y , we can calculate $\tilde{E}_1(u^n)$ and $F'_1(u^n)$. Similarly, once we figure out $A^{-1}W$, where $A = I + \epsilon \Delta t \Delta^2, W = g^n$ or $W = \Delta b^n$, we can figure out (b^n, u^{n+1}) and u^{n+1} .

We give the definition of the discrete forward and backward differential operators with the periodic condition as follows:

$$\begin{aligned} \partial_1^+ y(i, j) &= \begin{cases} (y(i + 1, j) - y(i, j))/\Delta x_1, & 0 \leq i < M - 1, \\ (y(0, j) - y(M - 1, j))/\Delta x_1, & i = M - 1, \end{cases} \\ \partial_1^- y(i, j) &= \begin{cases} (y(i, j) - y(i - 1, j))/\Delta x_1, & 0 < i \leq M - 1, \\ (y(0, j) - y(M - 1, j))/\Delta x_1, & i = 0, \end{cases} \\ \partial_2^+ y(i, j) &= \begin{cases} (y(i, j + 1) - y(i, j))/\Delta x_2, & 0 \leq j < L - 1, \\ (y(i, 0) - y(i, L - 1))/\Delta x_2, & j = L - 1, \end{cases} \\ \partial_2^- y(i, j) &= \begin{cases} (y(i, j) - y(i, j - 1))/\Delta x_2, & 0 < j \leq L - 1, \\ (y(i, 0) - y(i, L - 1))/\Delta x_2, & j = 0. \end{cases} \end{aligned}$$

Correspondingly, the central difference and the gradient operators can be defined as

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

We first solve the problem $y = \Delta^{-1}U$. From Remark 1, we know that $\int_{\Omega} U dx = 0$. Besides, y satisfies the condition $\int_{\Omega} y dx = 0$ by setting $y = y - \bar{y}$ with $\bar{y} = \int_{\Omega} y dx / |\Omega|$. According to Lemma 1, we can obtain the equation $-\Delta y = U$ has a unique solution. Next, we can use FFT to solve y . We apply the corresponding discretization:

$$-(\partial_1^- \partial_1^+ + \partial_2^- \partial_2^+) y = U.$$

We use the discrete Fourier transformation \mathcal{F} on both sides at the same time, and we get

$$\begin{aligned} \mathcal{F} \partial_1^\pm y(i, j) &= \pm \left(e^{\pm \sqrt{-1} z_i^1} - 1 \right) \mathcal{F} y(i, j), \\ \mathcal{F} \partial_2^\pm y(i, j) &= \pm \left(e^{\pm \sqrt{-1} z_j^2} - 1 \right) \mathcal{F} y(i, j). \end{aligned}$$

where $z_i^1 = 2\pi(i - 1)/(M - 1), i = 0, \dots, M - 1, z_j^2 = 2\pi(j - 1)/(L - 1), j = 0, \dots, L - 1$, we have the following equation:

$$-2 \left(\cos z_i^1 + \cos z_j^2 - 2 \right) \mathcal{F} y(i, j) = \mathcal{F} U(i, j).$$

Finally, we can solve y by

$$y(i, j) = \mathcal{F}^{-1} \left(\mathcal{F} U(i, j) / \left(-2 \left(\cos z_i^1 + \cos z_j^2 - 2 \right) \right) \right).$$

We can define the differential and the gradient operators that are similar to the previous. We also try to solve the equation $z = A^{-1}W$ through FFT. First of all, we need to obtain the

following discretization:

$$z + \epsilon \Delta t (\partial_1^- \partial_1^+ + \partial_2^- \partial_2^+) z = W.$$

We then employ the discrete Fourier transform on both sides of the equation, and get the equation as follows:

$$\left(1 + 4\epsilon \Delta t (\cos z_i^1 + \cos z_j^2 - 2)\right) \mathcal{F}z(i, j) = \mathcal{F}W(i, j).$$

Once we compute $\mathcal{F}z$, we can discover the value z by applying the discrete inverse Fourier transform.

5 Numerical Experiments

We present in this section binary image inpainting results of our proposed model (7), implemented by the SAV scheme, and discuss the efficiency of the adaptive time stepping strategy. In addition, the proposed method is generalized to grayscale image inpainting.

5.1 Inpainting of Binary Images

In the experiment, our inpainting model is solved with a large value of ϵ (e.g., $\epsilon = 1.4$) and this step is stopped when the steady state solution is obtained, which causes the topology of the inpainting area to be reconnected. Next, the pixel value of the binary image more than 0.5 is taken to 1, and the value less than 0.5 is set to 0. The simple threshold method can make the edges of images sharpened. The choice of ϵ is presented in Table 1. The geometric graphics in the table refer to Figs. 2, 3, 4 and 5 and the cartoon patterns refer to Figs. 10 and 11. Other parameters are chosen to be $\lambda_0 = 1$, $C_0 = e^{-5}$. The gray area in the input image represents the inpainting region and we choose two classes of initial values for it, we set random initial data in the gray area of Figs. 2, 3, 4, 5 and 6 and set 0.5 for the initial values of the remaining test images. The methods we compared are the BSCB inpainting model in [10], the TV inpainting model in [13], the CDD inpainting model in [15], the MSE inpainting model in [14], and the mCH equation inpainting model in [20].

The proposed inpainting model (7) is suitable for binary image inpainting. We first focus our attention on the inpainting area of the topology with corners. As shown in Fig. 2b, the inpainting domain is a rectangle that covers the upper part of the triangle. The pentagram is displayed in Fig. 3b and the inpainting area is irregularly covered on the image. Figure 2c–g show that triangles have been truncated and have become a shape similar to a trapezoid. A similar situation has occurred in Fig. 3c–g. In both examples, we see that using other inpainting methods is hard for the image information to continue to spread along the two sides of the occluded corners, which has destroyed the geometric structure of the original image and has lost most of the image information in the inpainting area. Examples indicate

Table 1 The values of ϵ for numerical experiments

Image	Geometric graphics	Text	Stripe	Sloping stripe	Ripple	Cartoon patterns
ϵ	1.4	1.4	0.7	0.5	0.4	1.2

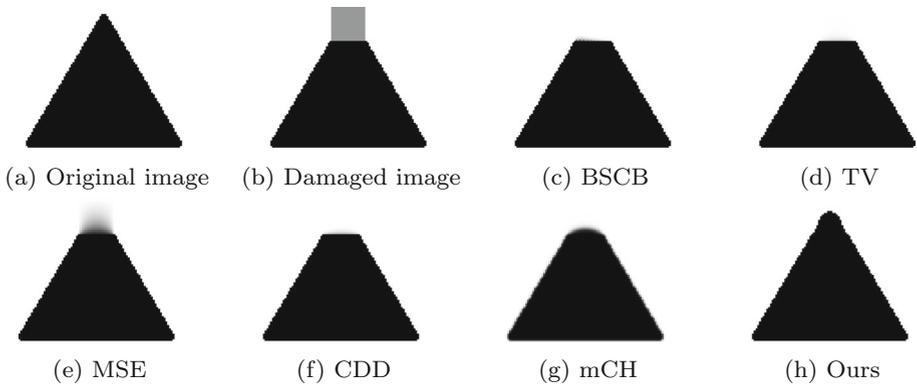


Fig. 2 Inpainting the triangle

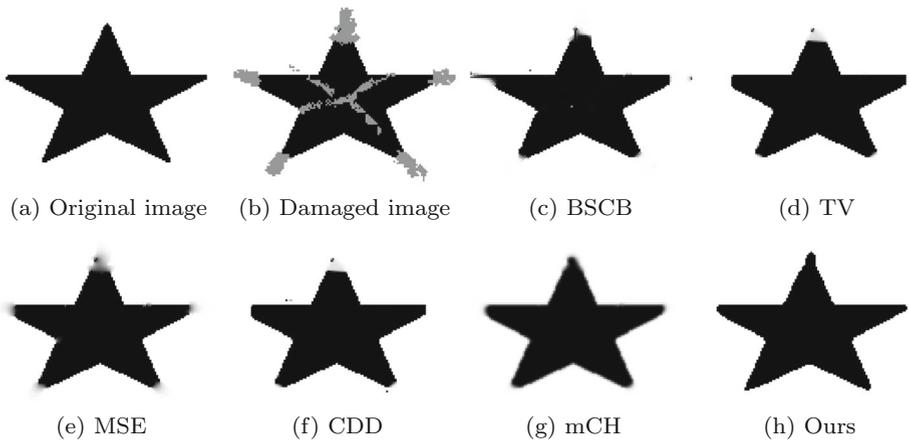


Fig. 3 Inpainting the pentagram

what our inpainting model is able to accomplish: continuing to spread geometric information along both sides of corners and recovering most of the image content, as shown in Figs. 2h and 3h.

In order to compare the results more clearly, we made statistics on the number of pixels with values less than 0.5 in Figs. 2 and 3, respectively. The value 0 represents black, and the value 1 stands for white. In the inpainting area, the approximate size of the black area obtained by each inpainting method is counted in Table 2. It shows that for the triangle image, our inpainting method was able to fill around 91% of the black area, while other inpainting methods can only reach about 0%–36%. Some of these inpainting methods do not recover any image content at all, and others only recover a small portion of the image. For the inpainting of the pentagram, it demonstrates that our inpainting method can reconstruct around 84% of the black area, while other inpainting methods only reconstruct about 30%–59%. One can see that our inpainting approach works well for topology with corners.

Then we turn our attention to the inpainting area that the topology with curvature. In Fig. 4 we display different inpainting results for the circle. Note that the MSE inpainting method output unconnected arcs on the edge and missing some black areas in the circle. The mCH

Table 2 Statistical results of the image pixel values

Category	Initial value	BSCB	TV	CDD	MSE	mCH	Ours
Triangle	174	4	0	0	63	45	159
Pentagram	249	135	75	75	127	147	208

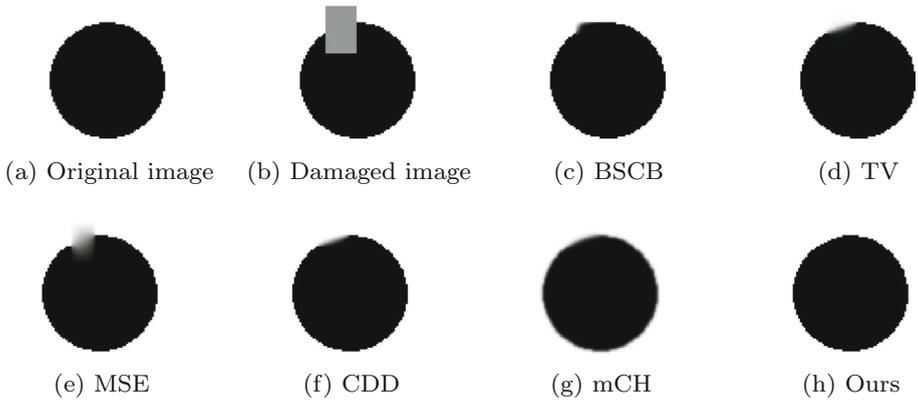


Fig. 4 Inpainting the circle

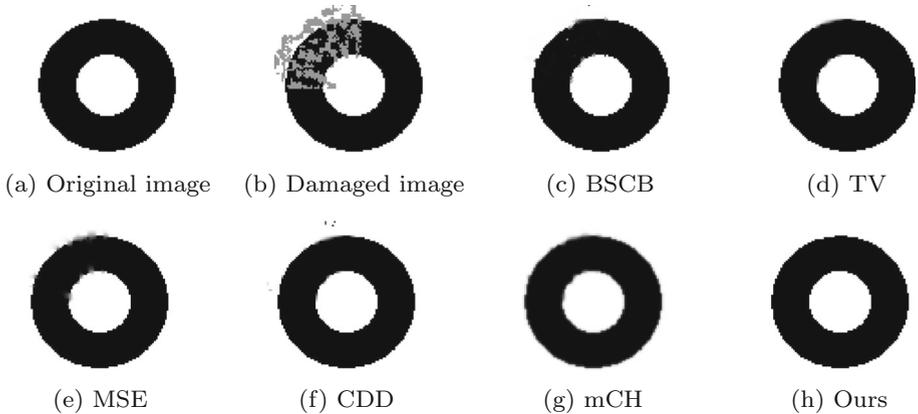


Fig. 5 Inpainting the ring

equation inpainting method is able to restore some missing regions, but it makes blurring edges. The others produce visible artificial corners, while our proposed inpainting method completes successfully the reconstruction of the circle. As for the inpainting results of the ring in Fig. 5, again the other inpainting approaches make the connection unsatisfactory. Our inpainting approach yields natural visual effects for both the large outer circle and the small inner circle.

As illustrated in Fig. 6, we show the inpainting performance of each inpainting approach on the text. For some oscillatory patterns, our inpainting approach also outperforms the other inpainting approaches. As can be seen in Fig. 7a, the width of the stripes gradually widens from left to right. Our method can inpainting the fine stripe in Fig. 7h, other methods can

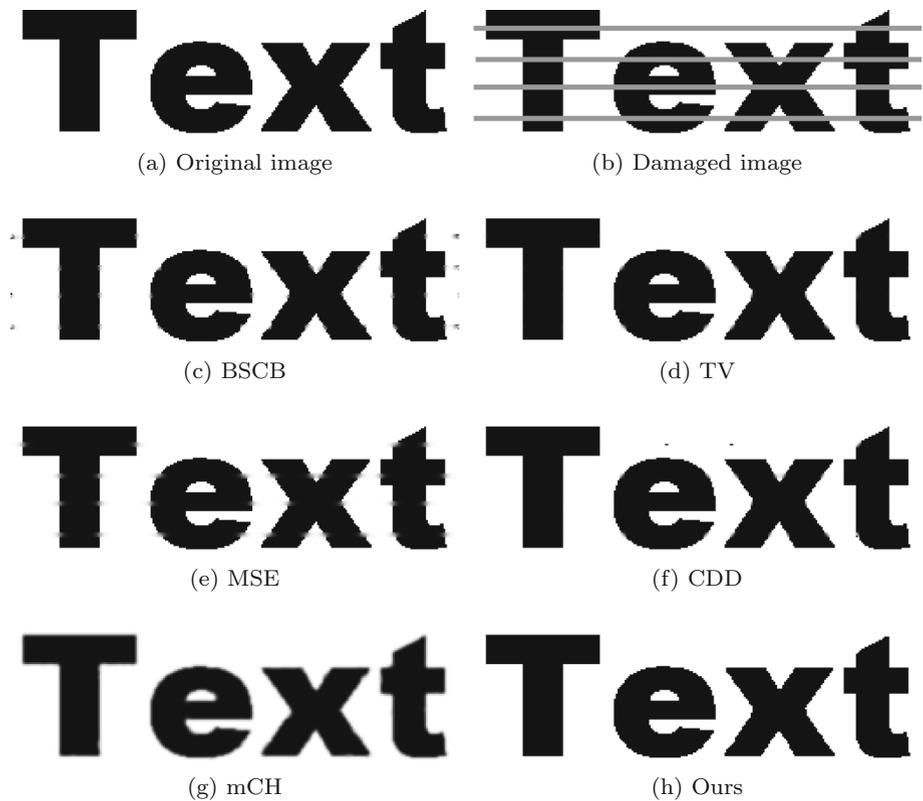


Fig. 6 Inpainting the text

only inpainting the wide stripe in Fig. 7c–g. It can be observed in Figs. 8 and 9 that our inpainting approach is suitable for preserving the edges of the oscillatory areas. Especially, the other inpainting methods fail to produce natural visual output, our inpainting method can connect the edges of images relatively smoothly and result in a reasonable reconstruction of the image.

The results of inpainting the cartoon patterns are displayed in Figs. 10 and 11. In both cases, the cartoon patterns were randomly corrupted. In Fig. 10, the BSCB, TV, MSE, and CDD inpainting methods all caused some contour edges of the cat to be disconnected, such as the bottom edge of the cat. In contrast, our inpainting method can smoothly connect these edges. In Fig. 11, the BSCB, TV, MSE, and CDD inpainting methods all have the problem of inaccurate shape restoration in some local areas of the cow, such as the lower left corner of the cow. However, our inpainting method can more accurately inpaint these areas. As shown in Figs. 10 and 11g, the mCH inpainting method will make the cartoon patterns blurry.

All the numerical experiments are performed under Windows 10, on a computer with an Intel Core i7-10750H CPU at 2.60 GHz with 16.0 GB of RAM, and run in MATLAB 9.8.0 (R2020a). In Table 3, we show the comparison of CPU time (in seconds) for results in four test images. One can easily observe that our method takes relatively less time to achieve good inpainting results, the overall performance is more efficient than the other five approaches.

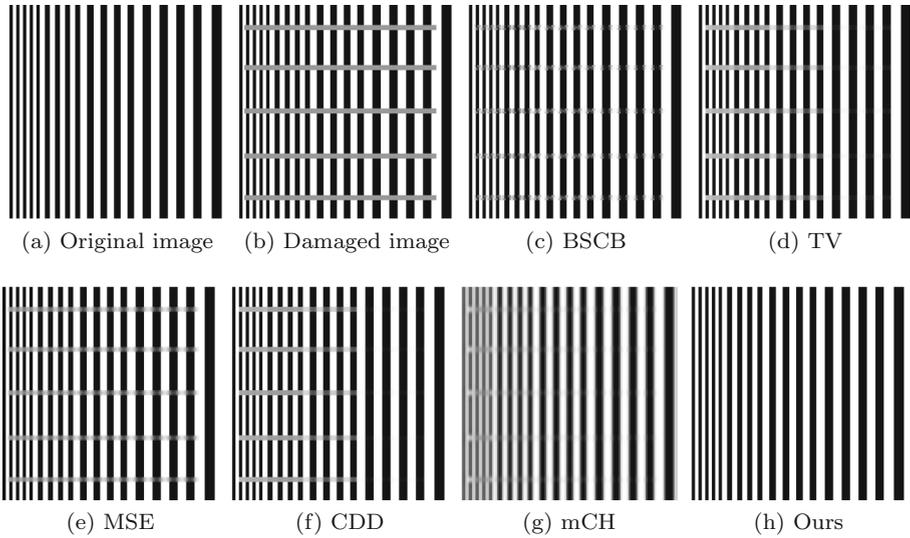


Fig. 7 Inpainting the stripe with unequal width

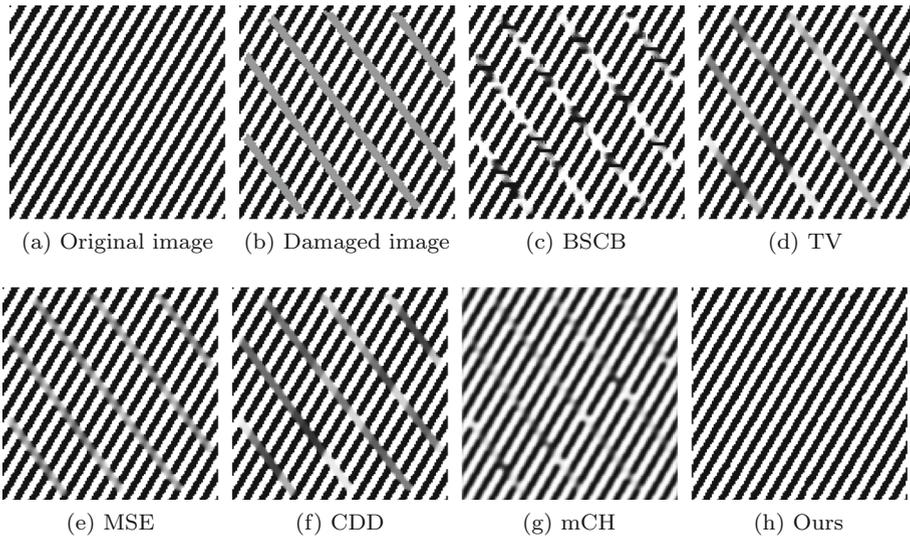


Fig. 8 Inpainting the sloping stripe

We discussed the impact of model parameters on experimental results. Two key parameters in the proposed model are ϵ and λ_0 . The role of ϵ is the interface thickness of a transition layer of the separated region which represents two different states. We conducted experiments with different values of ϵ while keeping the other parameters constant in the proposed model. A set of experimental results is shown in Fig. 12. If an excessively small value of ϵ is chosen, it will be unable to connect larger inpainting areas because the sharp transition region caused by the small value of ϵ makes it difficult to blend with surrounding pixels. Conversely, if an excessively large value of ϵ is chosen, it will result in the loss of the sharp features of the

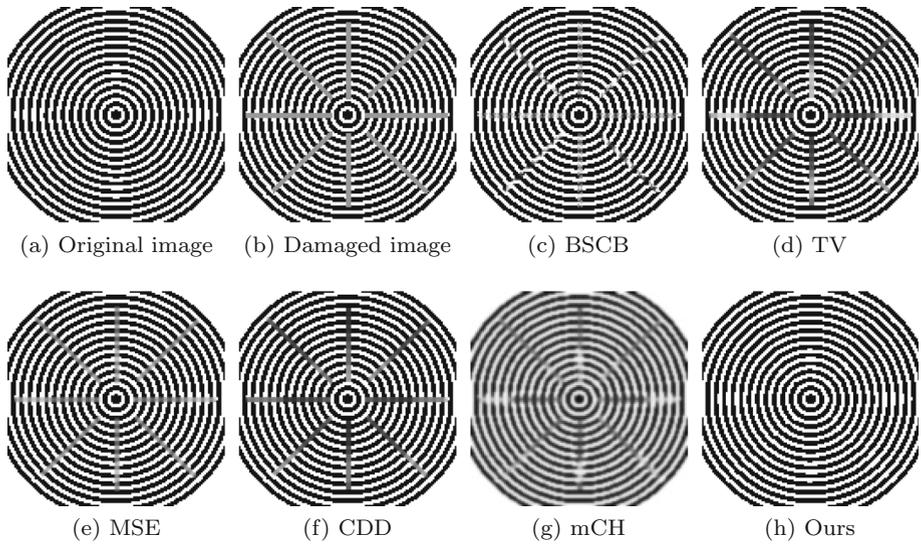


Fig. 9 Inpainting the ripple

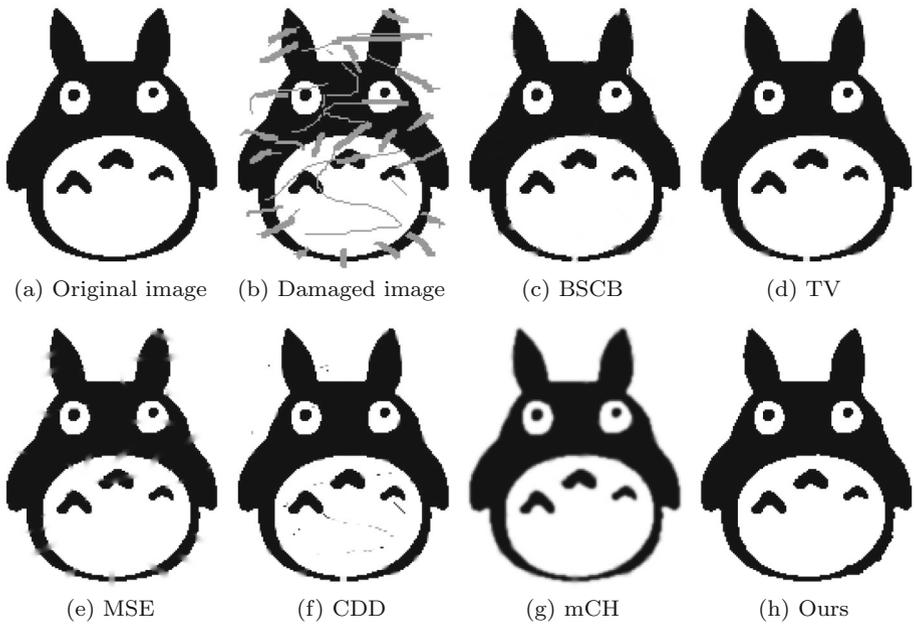


Fig. 10 Inpainting the cat

image because the large value of ϵ causes the transition region to become oversmoothed. Therefore, when choosing the value of ϵ , it is necessary to consider the size of the inpainting area and the characteristics of the image, while balancing the connection of the inpainting area and sharp features. We choose ϵ to be between 0.4 and 2.5 empirically.

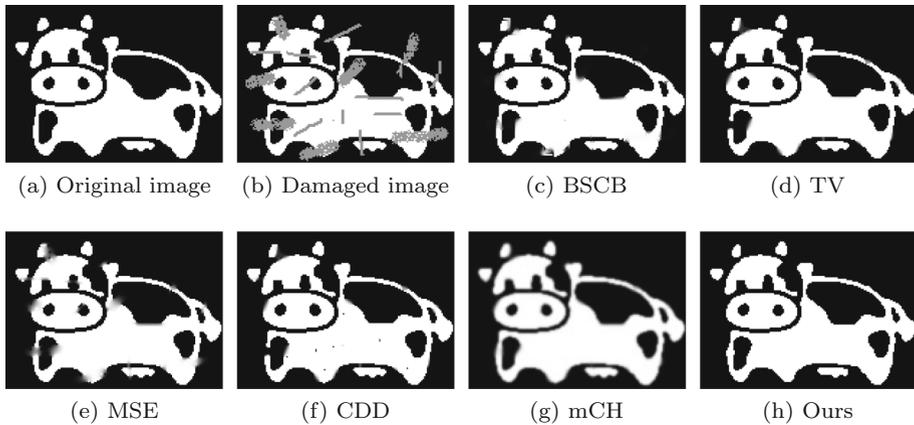


Fig. 11 Inpainting the cow

Table 3 Comparison of CPU time

Image	BSCB	TV	CDD	MSE	mCH	Ours
Pentagram	9.42	8.87	7.95	15.26	3.44	0.91
Text	48.13	7.52	109.82	16.10	4.56	4.12
Sloping stripe	9.14	9.16	14.43	14.91	3.54	0.73
Ripple	4.95	8.62	14.58	15.41	3.23	0.39

The parameter λ_0 in the proposed model is used to balance the regularization term and the fidelity term. Selecting an appropriate value of λ_0 is crucial for achieving high-quality inpainting results. We take the same initial condition except for different values of λ_0 . We show the results of inpainting triangle in Fig. 13. If the value of λ_0 that is too small is selected, the pixel values in the non-inpainting area will be affected, and the inpainting result may not appear natural. Conversely, if a value of λ_0 that is too large is selected, the connection between the inpainting area and the rest of the image may not be smooth, leading to a reduced visible effect. We suggest empirically choosing λ_0 between 0.8 and 1.2.

5.2 Application of the Adaptive Time Stepping Strategy

In this subsection, we present several test results to demonstrate the efficiency of the time step adaptive strategy. We choose three test images for the experiment by using the SAV algorithm. In all examples, we select $tol = 0.03$, $\rho = 0.05$. For comparison, we perform the experiment by the SAV algorithm using different fixed time steps i.e., a large fixed time step $\Delta t = 10^{-2}$ and a small time step $\Delta t = 10^{-4}$.

Table 4 shows the number of iterations and CPU time required by three test images in different time steps. We observe that the algorithm with time step adaptive strategy requires the least number of iterations and CPU time compared with the other two methods. We plot the evolution of energy curves for triangle image inpainting in Fig. 14. Similarly, we present three different time step methods. It can be seen that the energy of numerical solution decreases as the number of iterations increases, and ultimately converge to the steady state. Among them, the energy curve of using the adaptive time stepping strategy is first converged

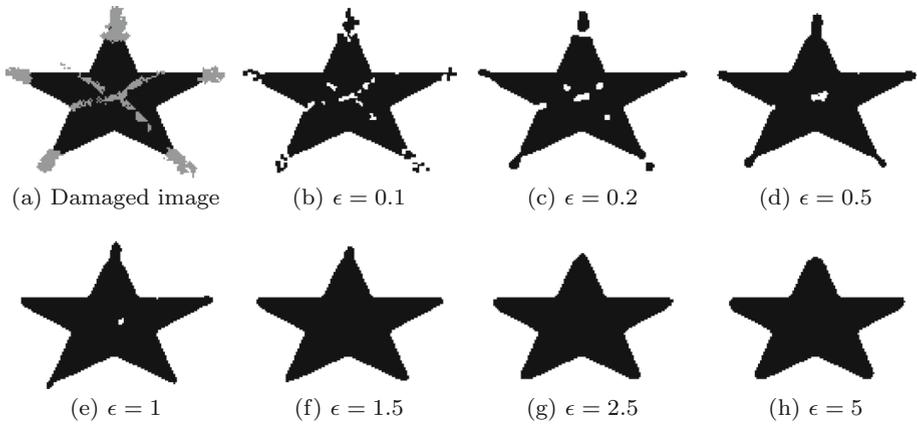


Fig. 12 Inpainting results with different ϵ parameter values

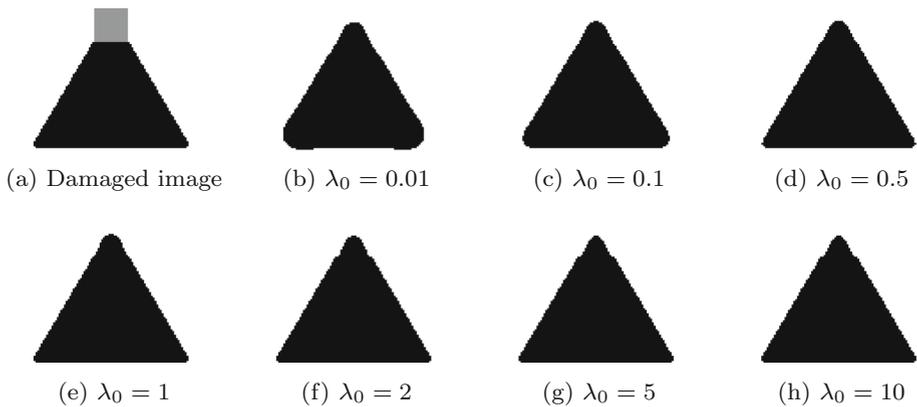


Fig. 13 Inpainting results with different λ_0 parameter values

to the steady state. In addition, we show in Fig. 15, the change of adaptive time steps during the triangle image inpainting. To sum up, these results verify that the time step adaptive strategy for the SAV algorithm is efficient.

5.3 Inpainting of Grayscale Images

In this subsection, we extend the proposed inpainting method to grayscale images by combining the reasonable strategy proposed in [23]. To be specific, we split the grayvalue image bitwise into K binary channels,

$$u(x) \approx \sum_{k=1}^K u_k(x) 2^{-(k-1)},$$

where $u(x)$ denotes the grayscale image, $u_k(x)$ is a binary channel. Our inpainting method is performed on each channel. Then the grayscale image inpainting can be implemented by assembling all channels. In our experiment, we set $K = 8$ and combine the two-step

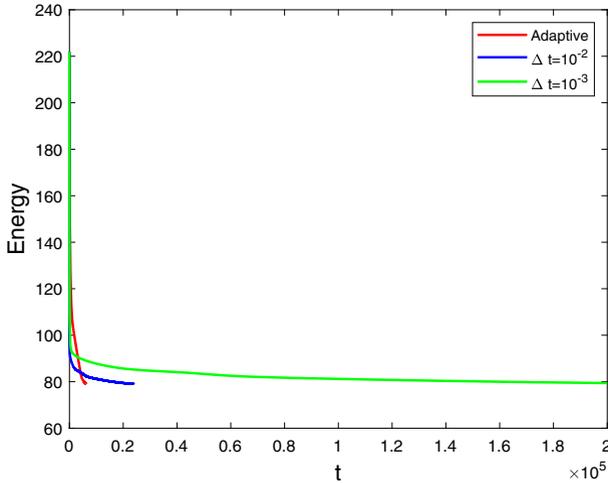


Fig. 14 Energy evolution of different time steps for inpainting the triangle

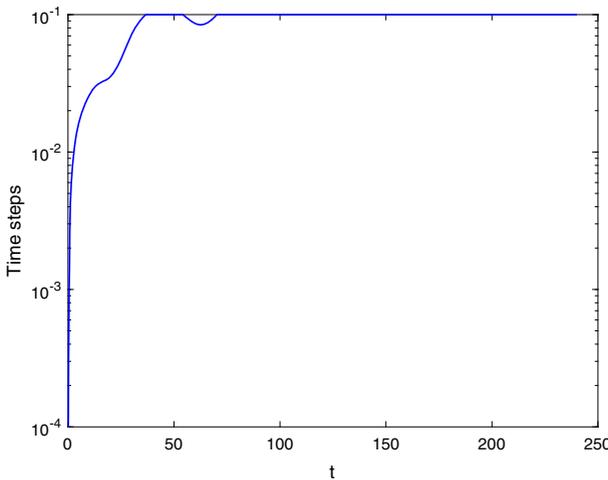


Fig. 15 Adaptive time steps of inpainting the triangle

approach in [20] to adjust experimental parameter. In the first step, we choose a large value of ϵ (e.g., $\epsilon = 1.4$), and the added step is we reduce the value of ϵ (e.g., $\epsilon = 0.5$) to get the result by taking the result of the first step as the initial data.

With the above technique, we are now to perform grayscale image inpainting. In Fig. 16 shows four original grayscale images and Fig. 17 shows corresponding damaged grayscale images. Our inpainting results are all displayed in Fig. 18. As shown in Fig. 17a, the inpainting areas are irregular, and the visible inpainting effects are produced in Fig. 18a. The grayscale images in Fig. 18b and c demonstrate the application of our image inpainting technique for erasing autokinetic text. The final example is the inpainting of a bar code on a commodity. It can be observed that our inpainting method is capable of achieving a satisfactory visual effect.

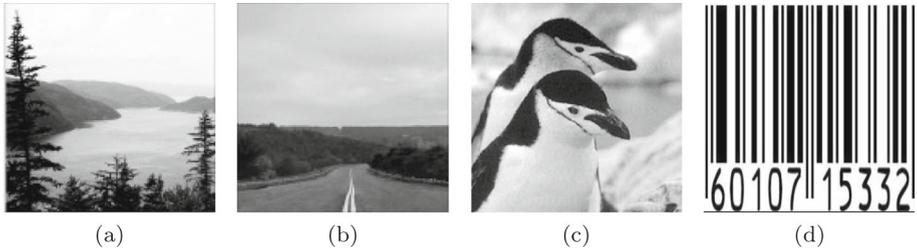


Fig. 16 The original grayscale images

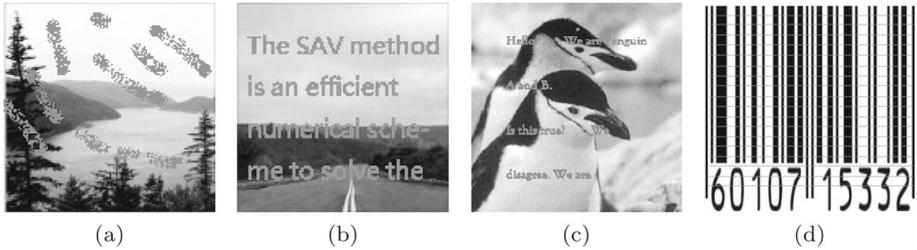


Fig. 17 The damaged grayscale images

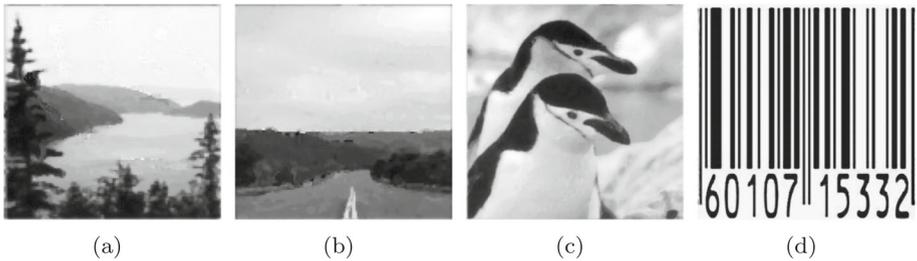


Fig. 18 Inpainting the grayscale images

6 Concluding Remarks

We developed a new image inpainting model by using the Ginzburg-Landau functional and a H^{-1} fidelity term. The proposed model can better capture some geometric contents of topologically complex inpainting areas. Meanwhile, the H^{-1} -fidelity term preserves well the edges of the oscillatory inpainting areas. We established the existence of the minimizer for the proposed variational inpainting model, and proposed an efficient numerical algorithm by adopting the SAV algorithm with the adaptive time stepping strategy. The scheme is very efficient and easy to implement.

Our numerical results indicate that the proposed inpainting approach works better compared with other tested approaches, even when the topology of the inpainting areas is complex or the areas are oscillatory. We also extended the image inpainting technique to grayscale images.

Acknowledgements This work is partially supported by the Fundamental Research Fund for the Central Universities (HIT. NSRIF202202), the National Natural Science Foundation of China (12171123, 11971131, 11871133, 11671111, U1637208, 61873071, 51476047). The Natural Science Foundation of Heilongjiang

Province of China (LH2021A011) and China Postdoctoral Science Foundation (2020M670893). China Society of Industrial and Applied Mathematics Young Women Applied Mathematics Support Research Project.

Funding The authors have not disclosed any funding.

Data Availability The data used to support the findings of this study are available from the corresponding author upon request.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Efros, A.A., Leung, T.K.: Texture synthesis by non-parametric sampling. *IEEE Int. Conf. Comput. Vis.* **2**, 1033–1038 (1999)
2. Criminisi, A., Pérez, P., Toyama, K.: Region filling and object removal by exemplar-based image inpainting. *IEEE Trans. Image Process.* **13**(9), 1200–1212 (2004)
3. Guillemot, C., Le Meur, O.: Image inpainting: overview and recent advances. *IEEE Signal Process. Mag.* **31**(1), 127–144 (2014)
4. Zhang, J., Zhao, D., Gao, W.: Group-based sparse representation for image restoration. *IEEE Trans. Image Process.* **23**(8), 3336–3351 (2014)
5. Elad, M., Starck, J.L., Querre, P., Donoho, D.L.: Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA). *Appl. Comput. Harmon. Anal.* **19**(3), 340–358 (2005)
6. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: feature learning by inpainting. In: *IEEE conference on computer vision and pattern recognition* pp 2536–2544 (2016)
7. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. *Adv. Neural. Inf. Process. Syst.* **27**, 2672–2680 (2014)
8. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Globally and locally consistent image completion. *ACM Trans. Graphics (ToG)* **36**(4), 1–14 (2017)
9. Yeh, R.A., Chen, C., Yian Lim, T., Schwing, A.G., Hasegawa-Johnson, M., Do, M.N.: Semantic image inpainting with deep generative models. In: *IEEE conference on computer vision and pattern recognition*, pp 5485–5493 (2017)
10. Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. In: *The 27th annual conference on computer graphics and interactive techniques*, pp 417–424 (2000)
11. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. *Physica D* **60**(1–4), 259–268 (1992)
12. Mumford, D.B., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math.* 577–685 (1989)
13. Shen, J., Chan, T.F.: Mathematical models for local nontexture inpaintings. *SIAM J. Appl. Math.* **62**(3), 1019–1043 (2002)
14. Esedoglu, S., Shen, J.: Digital inpainting based on the Mumford–Shah–Euler image model. *Eur. J. Appl. Math.* **13**(4), 353–370 (2002)
15. Chan, T.F., Shen, J.: Nontexture inpainting by curvature-driven diffusions. *J. Visual Comm. Image Rep.* **12**(4), 436–449 (2001)
16. Mumford, D.: *Algebraic Geometry and its Applications*, pp. 491–506. Springer, New York (1994)
17. Chan, T.F., Kang, S.H., Shen, J.: Euler’s elastica and curvature-based inpainting. *SIAM J. Appl. Math.* **63**(2), 564–592 (2002)
18. De Giorgi, E.: *Frontiere Orientate di Misura Minima. Lecture Notes, Sem. Mat. Scuola Norm. Sup. Pisa.* 61 (1960)
19. Cahn, J.W., Hilliard, J.E.: Free energy of a nonuniform system. I. Interfacial free energy. *J. Chem. Phys.* **28**(2), 258–267 (1958)
20. Bertozzi, A.L., Esedoglu, S., Gillette, A.: inpainting of binary images using the Cahn-Hilliard equation. *IEEE Trans. Image Process.* **16**(1), 285–291 (2006)
21. Bertozzi, A., Esedoglu, S., Gillette, A.: Analysis of a two-scale Cahn-Hilliard model for binary image inpainting. *Multiscale Model. Simul.* **6**(3), 913–936 (2007)
22. Bosch, J., Kay, D., Stoll, M., Wathen, A.J.: Fast solvers for Cahn–Hilliard inpainting. *SIAM J. Imag. Sci.* **7**(1), 67–97 (2014)

23. Bertozzi, A., Schönlieb, C.B.: Unconditionally stable schemes for higher order inpainting. *Commun. Math. Sci.* **9**(2), 413–457 (2011)
24. Burger, M., He, L., Schönlieb, C.B.: Cahn–Hilliard inpainting and a generalization for gray value images. *SIAM J. Imag. Sci.* **2**(4), 1129–1167 (2009)
25. Garcke, H., Lam, K.F., Styles, V.: Cahn–Hilliard inpainting with the double obstacle potential. *SIAM J. Imaging Sci.* **11**(3), 2064–2089 (2018)
26. Chen, W., Conde, S., Wang, C., Wang, X., Wise, S.: A linear energy stable scheme for a thin film model without slope selection. *J. Sci. Comput.* **52**, 546–562 (2012)
27. Chen, W., Wang, C., Wang, X., Wise, S.M.: A linear iteration algorithm for a second-order energy stable scheme for a thin film model without slope selection. *J. Sci. Comput.* **59**(3), 574–601 (2014)
28. Shen, J., Wang, C., Wang, X., Wise, S.M.: Second-order convex splitting schemes for gradient flows with Ehrlich–Schwoebel type energy: application to thin film epitaxy. *SIAM J. Numer. Anal.* **50**(1), 105–125 (2012)
29. Zhu, J., Chen, L.Q., Shen, J., Tikare, V.: Coarsening kinetics from a variable-mobility Cahn–Hilliard equation: application of a semi-implicit Fourier spectral method. *Phys. Rev. E* **60**(4), 3564–3572 (1999)
30. Shen, J., Yang, X.: Numerical approximations of Allen–Cahn and Cahn–Hilliard equations. *Discrete Contin. Dyn. Syst.* **28**(4), 1669–1691 (2010)
31. Guillén-González, F., Tierra, G.: On linear schemes for a Cahn–Hilliard diffuse interface model. *J. Comput. Phys.* **234**, 140–171 (2013)
32. Zhao, J., Wang, Q., Yang, X.: Numerical approximations for a phase field dendritic crystal growth model based on the invariant energy quadratization approach. *Int. J. Numer. Methods Eng.* **110**(3), 279–300 (2017)
33. Gong, Y., Zhao, J., Wang, Q.: Arbitrarily high-order unconditionally energy stable schemes for thermodynamically consistent gradient flow models. *SIAM J. Sci. Comput.* **42**(1), B135–B156 (2019)
34. Shen, J., Xu, J., Yang, J.: The scalar auxiliary variable (SAV) approach for gradient flows. *J. Comput. Phys.* **353**, 407–416 (2018)
35. Shen, J., Xu, J., Yang, J.: A new class of efficient and robust energy stable schemes for gradient flows. *SIAM Rev.* **61**(3), 474–506 (2019)
36. Cheng, Q., Shen, J., Yang, X.: Highly efficient and accurate numerical schemes for the epitaxial thin film growth models by using the SAV approach. *J. Sci. Comput.* **78**(3), 1467–1487 (2019)
37. Cao, D., Shen, J., Xu, J.: Computing interface with quasiperiodicity. *J. Comput. Phys.* **424**, 109863 (2021)
38. Chen, J., He, Z., Sun, S., Guo, S., Chen, Z.: Efficient linear schemes with unconditional energy stability for the phase field model of solid-state dewetting problems. *J. Comput. Math.* **3**, 452–468 (2020)
39. Antoine, X., Shen, J., Tang, Q.: Scalar auxiliary variable/Lagrange multiplier based pseudospectral schemes for the dynamics of nonlinear Schrödinger/Gross-Pitaevskii equations. *J. Comput. Phys.* **437**, 110328 (2021)
40. Belhachmi, Z., Kallel, M., Moakher, M., Theljani, A.: Weighted harmonic and complex Ginzburg–Landau equations for gray value image inpainting. *Partial Differ. Equ. Methods Regular. Tech. Image Inpaint.* **1**, 1–57 (2016)
41. Theljani, A., Belhachmi, Z., Moakher, M.: High-order anisotropic diffusion operators in spaces of variable exponents and application to image inpainting and restoration problems. *Nonlinear Anal. Real World Appl.* **47**, 251–271 (2019)
42. Van der Waals, J.D.: The thermodynamic theory of capillarity under the hypothesis of a continuous variation of density. *J. Stat. Phys.* **20**(2), 200–244 (1979)
43. Osher, S., Solé, A., Vese, L.: Image decomposition and restoration using total variation minimization and the H^{-1} norm. *Multiscale Model. Simul.* **1**(3), 349–370 (2003)
44. Schönlieb, C.B.: Total variation minimization with an H^{-1} constraint. *CRM Series* **9**, 201–232 (2009)
45. Theljani, A., Belhachmi, Z., Kallel, M., Moakher, M.: A multiscale fourth-order model for the image inpainting and low-dimensional sets recovery. *Math. Methods Appl. Sci.* **40**(10), 3637–3650 (2017)
46. Meyer, Y.: *Oscillating Patterns in Image Processing and Nonlinear Evolution Equations*, vol. 22. American Mathematical Soc, Washington (2001)
47. Elliott, C.M., Stuart, A.M.: The global dynamics of discrete semilinear parabolic equations. *SIAM J. Numer. Anal.* **30**(6), 1622–1663 (1993)
48. Eyre, D.J.: Unconditionally gradient stable time marching the Cahn–Hilliard equation. *MRS Online Proc. Library (OPL)* **529**, 39 (1998)
49. Zhuang, Q., Shen, J.: Efficient SAV approach for imaginary time gradient flows with applications to one- and multi-component Bose-Einstein condensates. *J. Comput. Phys.* **396**, 72–88 (2019)
50. Zhang, Z., Qiao, Z.: An adaptive time-stepping strategy for the Cahn–Hilliard equation. *Commun. Comput. Phys.* **11**, 1261–1278 (2012)

51. Lou, F., Tang, T., Xie, H.: Parameter-free time adaptivity based on energy evolution for the Cahn–Hilliard equation. *Commun. Comput. Phys.* **19**, 1542–1563 (2016)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.