Pattern Recognition 123 (2022) 108405

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

# A robust and efficient fingerprint image restoration method based on a phase-field model

Yibao Li[a], Qing Xia[a], Chaeyoung Lee[b], Sangkwon Kim[b], Junseok Kim[b,*]

[a] School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China
[b] Department of Mathematics, Korea University, Seoul 02841, Republic of Korea

ARTICLE INFO

*Article history:*
Received 14 July 2021
Revised 30 September 2021
Accepted 24 October 2021
Available online 26 October 2021

*Keywords:*
Fingerprint restoration
Diblock copolymer
Nonlocal Cahn–Hilliard equation

ABSTRACT

In this study, we present a robust and efficient fingerprint image restoration algorithm using the nonlocal Cahn–Hilliard (CH) equation, which was proposed for modeling the microphase separation of diblock copolymers. We take a small local region embedding the damaged domain and solve the nonlocal CH equation to restore the fingerprint image. A Gauss–Seidel type iterative method, which is efficient and simple to implement, is used. The proposed method has the advantage in that the pixel values in the damaged fingerprint domain can be obtained using the image information from the outside of the damaged fingerprint region. Fingerprint restoration based on adjacent pixel information can ensure the accuracy of the fingerprint information with a low computational cost. Computational experiments demonstrated the superior performance of the proposed fingerprint restoration algorithm.



## 1. Introduction

Fingerprints have been extensively used for personal recognition in civil, forensic, and commercial applications owing to their uniqueness, invariability, and low cost [1]. The accuracy of the fingerprint recognition algorithm depends principally on the quality of the fingerprint image. However, there are generally certain degrees of fracture and scratches in the collected fingerprint images and to ensure the continuity of the enhanced ridges and reduce the number of redundant minutia points, it is necessary to restore damaged fingerprint images with a low contrast that lack clear boundaries between the ridges [2,3].

There has been an increasing interest during the past decades in restoring damaged fingerprint images and improving the matching performance. Existing methods for fingerprint restoration can be divided into two types: filtering approach and neural network-based fingerprint image enhancement methods. Hong et al. [4] proposed a Gabor filtering method to correct and amplify local ridge patterns. In addition, circular Gabor [6], log-Gabor filters [7], and curved Gabor [5] have been applied to restore damaged fingerprints with a unique curved structure. Feng and Jain [8] developed a technique for fingerprint reconstructing using a minutiae template. Their method is evaluated by comparing the reconstructed fingerprints with the original fingerprints. Li and Kot [9] proposed an algorithm for reconstructing an entire fingerprint image from the minutiae points. They estimated the continuous phase from a ridge pattern. Hsieh et al. [10] proposed a wavelet-based technique for fingerprint image enhancement, which combines the global texture and local orientation characteristics with a lower computational cost. Sutthiwichaiporn et al. [11] proposed an adaptive boosted spectral filtering algorithm that can enhance the singular point zones and iteratively improve the quality of damaged fingerprints. Furthermore, various neural network-based methods have been proposed for fingerprint restoration. Inspired by a convolutional neural network (CNN), Li et al. [12] proposed the FingerNet method, which is based on a multi-task learning strategy. Their method contains an enhancement branch and an orientation branch, which are used to remove the structured noise and guide the enhancement, respectively. Prabhu et al. [13] developed a multi-scale convolutional network to tackle the challenge of suppressing complicated artifacts while preserving fine textures. Their model uses dilated convolutions and removes the padding of the damaged fingerprint. In addition, Yadav and Tiwari [14] discussed different CNN-dependent methods and found that increasing the depth and width of the network can increase the receptive fields of an image with a high computational cost. Meanwhile, existing fingerprint imprinting algorithms lack strong theoretical support and a mathematical foundation.

The reaction-diffusion technique was first described in [15] for fingerprint synthesis. Ito et al. [16] presented an algorithm for fingerprint restoration using a digital reaction-diffusion system,

* Corresponding author.
*E-mail address:* cfdkim@korea.ac.kr (J. Kim).
*URL:* https://mathematicians.korea.ac.kr/cfdkim/ (J. Kim)

https://doi.org/10.1016/j.patcog.2021.108405
0031-3203/

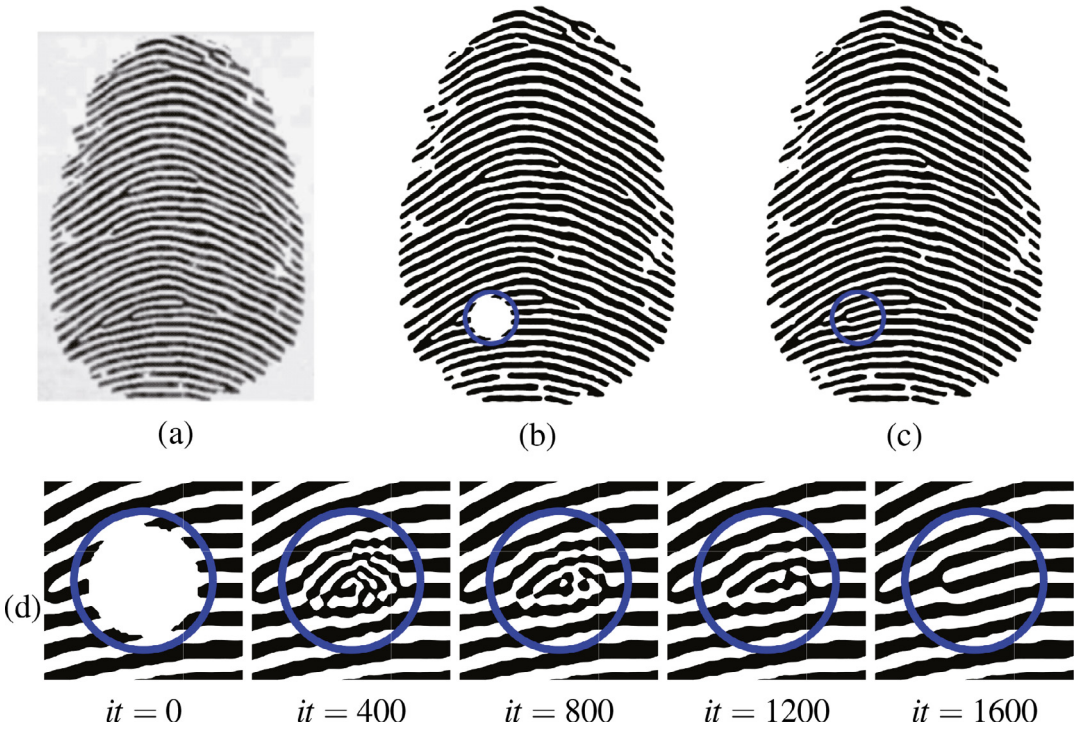


**Fig. 1.** Temporal evolution of impaired fingerprint restoration. (a) Original complete image [30]. Schematic representation of (b) incomplete fingerprint image and (c) restored image after applying the fingerprint restoration algorithm. (d) Temporal evolution of the restoration process. The computational iterations are shown in each figure.

which is used to generate biological textures, patterns, and structures. As a simple and efficient fingerprint restoration method, we used a nonlocal Cahn–Hilliard (CH) equation to model the phase separation of diblock copolymers, which is similar to the fingerprint image approach [17,18]. A diblock copolymer is a linear chain molecule consisting of two covalently bonded sub-chains. When the temperature is below the critical point, the copolymer melt undergoes phase separation, resulting in periodic mesoscopic structures such as lamellae, spheres, cylinders, hexagons, and gyroids [19]. A nonlocal term is supplemented to the standard CH free energy, which is generally used for modeling the phase separation [20,21]. Inspired by this phase separation phenomenon of a diblock copolymer, we propose a simple and efficient fingerprint restoration algorithm (see Fig. 1).

In this paper, we present a simple and efficient fingerprint image restoration algorithm using a phase-field mathematical model. We utilize the property of the self-assembly structure formation of the Ohta–Kawasaki functional [17,22]. We minimize the nonlocal CH free energy based on the monomer density functional theory [23] and use the equilibrium state of the phase field to restore an incomplete fingerprint image. To the best of our knowledge, the present approach is the first attempt at restoring a fingerprint image using the nonlocal phase-field model. Compared to the existing fingerprint restoration method, the advantages of the proposed algorithm are as follows: (i) The method transforms the problem of image processing into an optimization problem in which the existence and uniqueness of the solution can be theoretically proven. (ii) The pixel values within the damaged fingerprint domain are obtained through the proposed method, which uses the image information from the outside of the damaged fingerprint region. (iii) The pixel values outside of the damaged fingerprint region are the same as those in the original image because we do not change the outside region of the damaged fingerprint. Therefore, the proposed algorithm is computationally efficient. Finally, (iv) the numerical scheme is simple and easy to implement. Several computational tests are conducted to confirm the efficiency and robustness of the proposed method.

The outline of this paper is as follows. In Section 2, we describe the proposed mathematical model. In Section 3, we present a computational solution for the proposed fingerprint restoration method. In Section 4, various numerical experiments are performed to demonstrate the performance of the proposed method. Finally, in Section 5, the concluding remarks are presented.

## 2. Methodology

For a 2D image representation, let $\phi(\mathbf{x},t)$ be the relative local monomer density difference at point $\mathbf{x}=(x,y)$ and time $t$. Letting $f(\mathbf{x})$ be a given image, the nonlocal CH equation [17] is as follows:

$$\frac{\partial\phi(\mathbf{x},t)}{\partial t} = \begin{cases} \Delta\mu - \alpha(\phi(\mathbf{x},t)-\bar{\phi}) & \text{if } \mathbf{x}\in\Omega_{in}, \\ 0 & \text{otherwise}, \end{cases} \tag{1}$$

$$\mu = F'(\phi(\mathbf{x},t)) - \varepsilon^2\Delta\phi(\mathbf{x},t), \tag{2}$$

$$\phi(\mathbf{x},0) = \begin{cases} 0 & \text{if } \mathbf{x}\in\Omega_{in}, \\ \frac{2f(\mathbf{x})-f_{\min}-f_{\max}}{f_{\max}-f_{\min}} & \text{otherwise}, \end{cases} \tag{3}$$

$$\mathbf{n}\cdot\nabla\phi(\mathbf{x},\mathbf{t}) = 0, \quad \mathbf{x}\in\partial\Omega, \tag{4}$$

where $f_{\max}$ and $f_{\min}$ are the maximum and minimum values of $f(\mathbf{x})$, respectively. Here, $\mathbf{n}$ is a normal vector for the domain boundaries. Furthermore, let $\Omega_{in}\subset\Omega$ be the damaged fingerprint domain, $\partial\Omega_{in}$ be the boundary, and $\Omega\backslash\Omega_{in}$ be the complement of $\Omega_{in}$ in $\Omega$, (see Fig. 2(a)).

Here, we use $F(\phi) := 0.25(\phi^2-1)^2$ to define the Helmholtz free energy. The phase field $\phi\in[-1,\ 1]$ denotes the order parameter of a two-phase system. The phase field $\phi=\pm1$ corresponds to the pure state. In addition, $\epsilon$ and $\alpha$ are positive constants, and

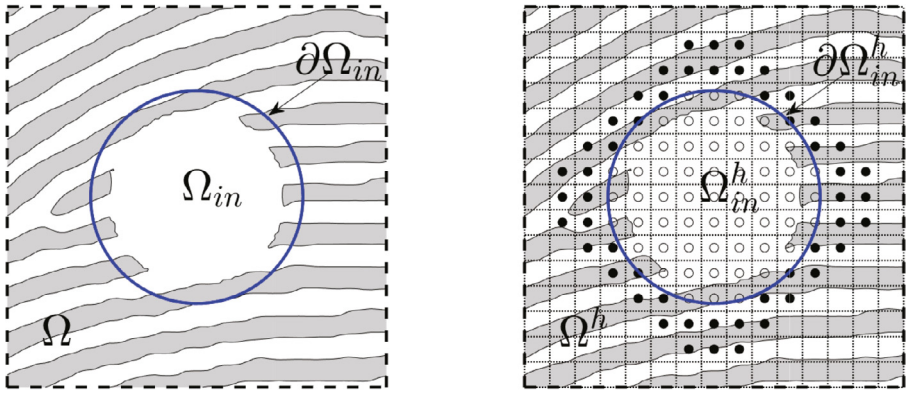


**Fig. 2.** (a) Complex domain $\Omega_{in}$ and (b) discrete complex domain $\Omega_{in}^h$.

$\bar{\phi} := \int_\Omega \phi(\mathbf{x},0)d\mathbf{x}/\int_\Omega d\mathbf{x}$ [24]. The total free energy functional can be defined as

$$E_{total} := E_{short} + E_{long} = \int_\Omega \left(\frac{\epsilon^2}{2}|\nabla\phi|^2 + F(\phi)\right)d\mathbf{x} + \frac{\alpha}{2}\int\int_\Omega G(\mathbf{x},\mathbf{y})(\phi(\mathbf{x})-\bar{\phi})(\phi(\mathbf{y})-\bar{\phi})d\mathbf{x}d\mathbf{y}, \tag{5}$$

where $G(\mathbf{x},\mathbf{y})$ denotes the Green function [25], i.e. $-\Delta G(\mathbf{x}) = \delta(\mathbf{x})$. We use $E_{short}$ and $E_{long}$ to denote the short-range (local) and long-range (nonlocal) parts of the free energy functional, respectively. If we choose $\alpha = 0$, then Eq. (1) becomes the CH equation, which describes the phase separation process [19]. If $\alpha \neq 0$, then the long-range effect is considered. The minimization of $E_{short}$ yields the domains of the pure phases [26] and $E_{long}$ induces oscillations between the phases [22]. Equation (1) can be obtained using the $H^{-1}$ gradient flow in Eq. (2):

$$\begin{aligned} E_{total} :&= E_{short} + \frac{\alpha}{2}\int\int_\Omega G(\mathbf{x},\mathbf{y})(\phi(\mathbf{x})-\bar{\phi})(\phi(\mathbf{y})-\bar{\phi})d\mathbf{x}d\mathbf{y} \\ &= E_{short} + \frac{\alpha}{2}\int_\Omega \Delta_{\mathbf{x}}\psi(\mathbf{x})\left[\int_\Omega \Delta_{\mathbf{y}}G(\mathbf{x}-\mathbf{y})\psi(\mathbf{y})d\mathbf{y}\right]d\mathbf{x} \\ &= E_{short} + \frac{\alpha}{2}\int_\Omega |\nabla\psi(\mathbf{x})|^2 d\mathbf{x}. \end{aligned} \tag{6}$$

Here, $\psi$ satisfies $-\Delta\psi = \phi - \bar{\phi}$. Applying the solution to Eq. (1), the energy dissipation law can be obtained as follows:

$$\begin{aligned} \frac{d}{dt}E_{total}(t) &= \frac{d}{dt}E_{short}(t) + \frac{d}{dt}E_{long}(t) \\ &= \int_\Omega (F'(\phi) - \epsilon^2\Delta\phi)\phi_t d\mathbf{x} - \int_\Omega (\alpha\Delta\psi)\psi_t d\mathbf{x} \\ &= \int_\Omega \mu\phi_t d\mathbf{x} - \int_\Omega \alpha\psi\Delta\psi_t d\mathbf{x} = \int_\Omega \mu\phi_t d\mathbf{x} + \int_\Omega \alpha\psi\phi_t d\mathbf{x} \\ &= \int_\Omega (\mu+\alpha\psi)\phi_t d\mathbf{x} = \int_\Omega (\mu+\alpha\psi)\cdot\Delta(\mu+\alpha\psi)d\mathbf{x} \\ &= -\int_\Omega |\nabla(\mu+\alpha\psi)|^2 d\mathbf{x} \le 0. \end{aligned} \tag{7}$$

Thus, the total energy decreases with respect to time.

## 3. Fingerprint restoration algorithm

We present an algorithm for numerically solving the nonlocal CH equation under the Dirichlet boundary condition. For the numerical domain $\Omega = (L_x, R_x)\times(L_y, R_y)$ of a given damaged fingerprint image, we consider a subdomain $\Omega_{in}$ containing the damaged area (see Fig. 2(a)). Let us define the discrete domain $\Omega^h = \{(x_i, y_j)|x_i = L_x + (i-0.5)h,\ y_j = L_y + (j-0.5)h,\ 1 \le i \le N_x, 1 \le j \le N_y\}$, where $h = (R_x - Lx)/N_x = (R_y - Ly)/N_y$ is the grid size, and $\Omega_{in}^h$ and $\partial\Omega_{in}^h$ are the discrete computational domain and the discrete boundary, respectively (see Fig. 2(b)). The blue solid circle represents the discrete boundary of the computational domain $\Omega_{in}^h$.

Let $\phi_{ij}^n$ be the numerical approximation of $\phi(x_i, y_j, n\Delta t)$, where $\Delta t = T/N_t$ is the time step, $T$ is the final time, and $N_t$ is a positive integer. Let us start with the following linear convex splitting scheme [27–29]:

$$\frac{\phi_{ij}^{n+1}-\phi_{ij}^n}{\Delta t} = \Delta_d((\phi_{ij}^n)^3 - 3\phi_{ij}^n) + 2\Delta_d\phi_{ij}^{n+1} - \epsilon^2\Delta_d^2\phi_{ij}^{n+1} - \alpha(\phi_{ij}^{n+1}-\bar{\phi}), \tag{8}$$

where $\Delta_d\phi_{ij}$ is the standard 5-point discrete Laplace operator and $\Delta_d^2\phi_{ij}$ is the standard 13-point discrete biharmonic operator. If we expand these discrete operators and use a Gauss–Seidel (GS) iterative marching scheme, we have

$$\begin{aligned} \frac{\phi_{ij}^{n+1}-\phi_{ij}^n}{\Delta t} &= \Delta_d((\phi_{ij}^n)^3 - 3\phi_{ij}^n) + \frac{2}{h^2}\left(\phi_{i-1,j}^{n+1} + \phi_{i+1,j}^n - 4\phi_{ij}^{n+1} + \phi_{i,j-1}^{n+1} + \phi_{i,j+1}^n\right) \\ &\quad - \frac{\epsilon^2}{h^4}\left[\phi_{i-2,j}^{n+1} + \phi_{i+2,j}^n + \phi_{i,j-2}^{n+1} + \phi_{i,j+2}^n\right. \\ &\quad + 2(\phi_{i-1,j-1}^{n+1} + \phi_{i-1,j+1}^n + \phi_{i+1,j-1}^{n+1} + \phi_{i+1,j+1}^n) \\ &\quad \left. - 8(\phi_{i-1,j}^{n+1} + \phi_{i+1,j}^n + \phi_{i,j-1}^{n+1} + \phi_{i,j+1}^n) + 20\phi_{ij}^{n+1}\right] - \alpha(\phi_{ij}^{n+1}-\bar{\phi}). \end{aligned} \tag{9}$$

Although several iterative methods such as Jacobi, successive over-relaxation, multigrid, conjugate gradient, and generalized minimal residual methods are available for a numerical approximation of the equations, nonlinear GS iterations are used here for the semi-implicit finite difference approximation and run until the norm of the residual is reached within a given tolerance. The GS iterative method can work well to solve the nonlocal CH equation because the coefficient matrix of Eq. (3) is strictly diagonally dominant or symmetric positive. Therefore the GS iterative approach is one of the simplest iterative methods available. By simplifying Eq. (3), we can obtain the following equation:

$$\begin{aligned} \phi_{ij}^{n+1} = \frac{1}{r}\Big[&\frac{\phi_{ij}^n}{\Delta t} + \Delta_d((\phi_{ij}^n)^3 - 3\phi_{ij}^n) + \frac{2}{h^2}\left(\phi_{i-1,j}^{n+1} + \phi_{i+1,j}^n + \phi_{i,j-1}^{n+1} + \phi_{i,j+1}^n\right) \\ &- \frac{\epsilon^2}{h^4}\left[\phi_{i-2,j}^{n+1} + \phi_{i+2,j}^n + \phi_{i,j-2}^{n+1} + \phi_{i,j+2}^n + 2(\phi_{i-1,j-1}^{n+1} + \phi_{i-1,j+1}^n\right. \\ &\left. + \phi_{i+1,j-1}^{n+1} + \phi_{i+1,j+1}^n) - 8(\phi_{i-1,j}^{n+1} + \phi_{i+1,j}^n + \phi_{i,j-1}^{n+1} + \phi_{i,j+1}^n)\right] + \alpha\bar{\phi}\Big], \end{aligned}$$

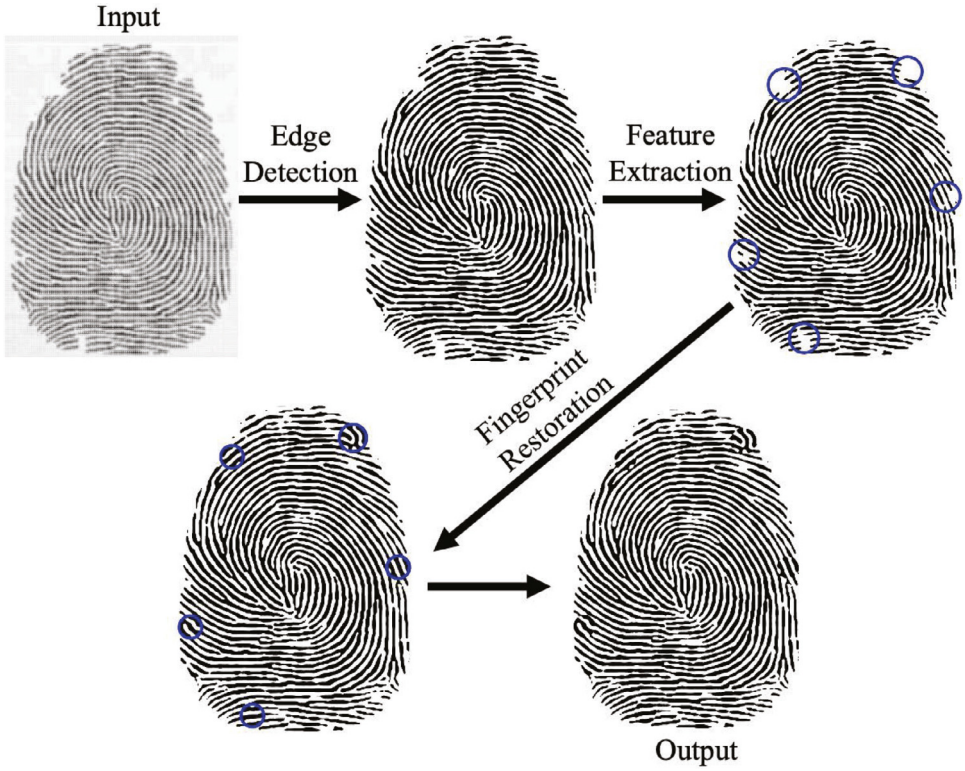


**Fig. 3.** Outline of the entire fingerprint restoration process.

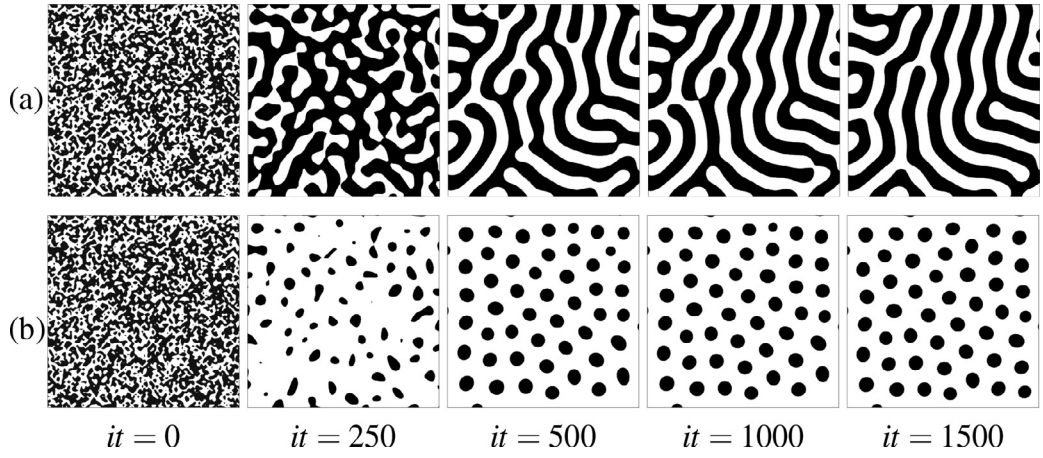


**Fig. 4.** Temporal evolution of the computational solutions at different iterations ($it$): (a) $\bar{\phi} = 0$ and (b) $\bar{\phi} = 0.3$. From left to right, the number of iterations is $it = 0$, 250, 500, 1000, and 1500, respectively.

where $r = 1/\Delta t + 8/h^2 + 20\epsilon^2/h^4 + \alpha$. The discrete energy can be defined as

$$E^h(\phi^n) = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\left(h^2 F(\phi_{ij}^n) + \frac{\epsilon^2}{2}\left[\left(\phi_{i+1,j}^n - \phi_{ij}^n\right)^2 + \left(\phi_{i,j+1}^n - \phi_{ij}^n\right)^2\right] + \frac{\alpha}{2}\left[\left(\psi_{i+1,j}^n - \psi_{ij}^n\right)^2 + \left(\psi_{i,j+1}^n - \psi_{ij}^n\right)^2\right]\right). \quad (10)$$

## 4. Numerical experiments

In this section, we focus on a numerical analysis of the fingerprint restoration algorithm. First, we compare the textures with different average concentrations. A non-increasing discrete energy test will be performed to confirm that the restoration has the unique solution. Then, we will conduct four qualitative analysis

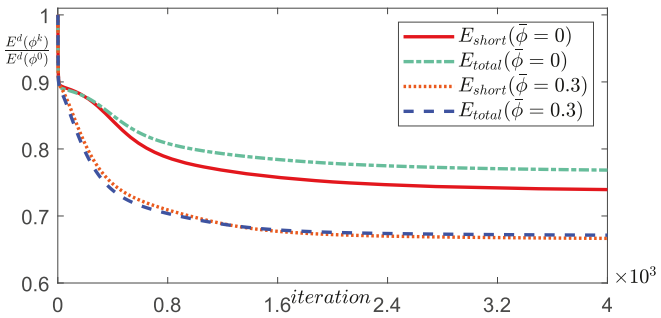


**Fig. 5.** Time evolution of the scaled discrete energy at different iterations($it$). The four curves correspond to $E_{short}(\bar{\phi}=0)$, $E_{total}(\bar{\phi}=0)$, $E_{short}(\bar{\phi}=0.3)$, and $E_{total}(\bar{\phi}=0.3)$.

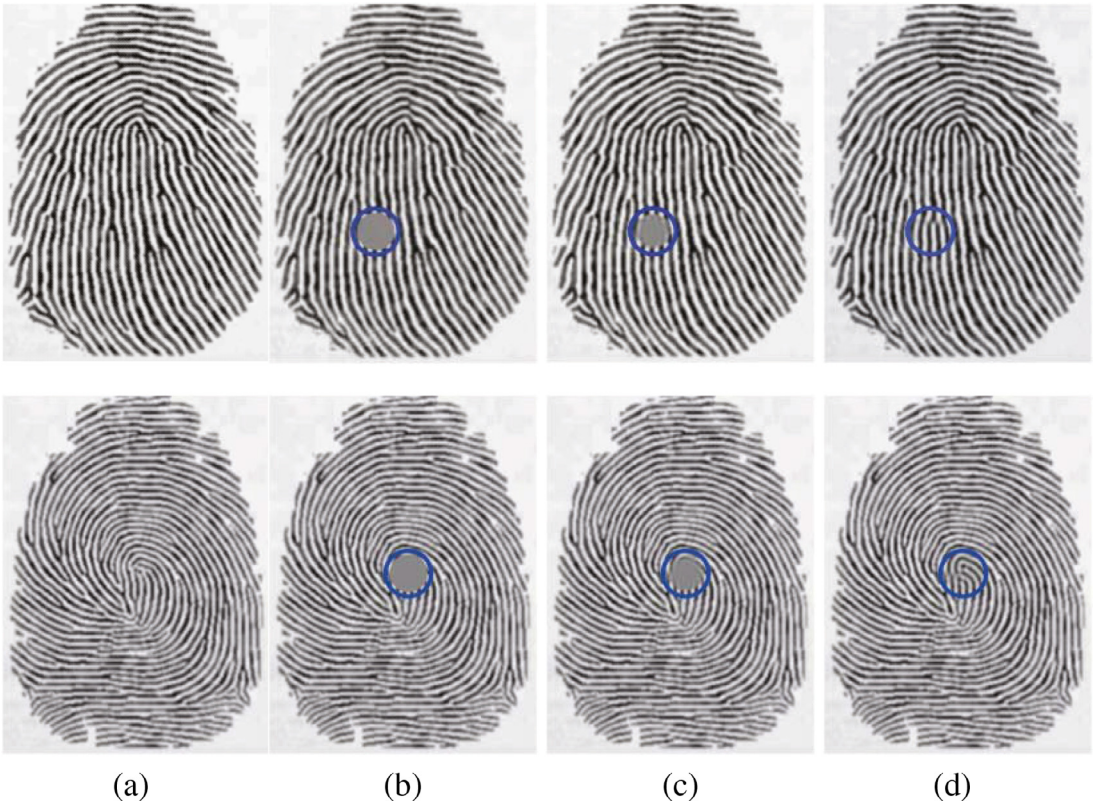


**Fig. 6.** Temporal evolution of different types of fingerprint restoration at different iterations($it$): (a) original complete image [30], (b) schematic representation of incomplete fingerprint image, (c) image in the middle of the restoration process, (d) the final result after applying the fingerprint restoration algorithm. The top row is the processing of the damaged image of the arch pattern and the bottom row is the processing of the damaged image of the loop pattern.

tests to verify the efficiency of the proposed method with different characteristics, such as the damage type, damage domain size, fingerprint texture, and parameter sensitivity analysis. Subsequently, we will perform a quantitative test to compare the peak signal-to-noise ratio, structural similarity index map and equal error rates using the relative methods. Finally, a computational test is performed to verify the efficiency of the proposed method. It should be noted that we stop the evolution and regard the numerical results as equilibrium solutions when the relative error $||\phi^{n+1}-\phi^n||_\infty/||\phi^n||_\infty$ is smaller than the tolerance $tol$. Unless otherwise stated, we will use the Dirichlet boundary condition for the boundaries.

For a given damaged fingerprint image, the outline of the entire process for fingerprint restoration is shown in Fig. 3. We first apply an edge detection algorithm [43,44] for the input images to extract the texture part of the fingerprint. The feature extraction algorithm [45,46] was then applied to locate the area that needs to be restored. Using our proposed method, we repair the damaged area in the entered fingerprint images. It should be noted that the feature extraction algorithm can be manually replaced. The following numerical tests are conducted to verify the robustness of the restoration method.

### 4.1. Phase separation of random mixture with different average concentration

In this section, we describe the evolution of a random perturbation with different average concentrations $\bar{\phi}$. The initial condition is chosen as $\phi(x,y,0)=\text{rand}(x,y)$ in $\Omega=(0,128)\times(0,128)$ with a $128\times128$ mesh grid, where $\text{rand}(x,y)$ is a random number between $-1$ and 1. The spatial and temporal steps are considered as $h_x=h_y=1$, and $\Delta t=0.1$, respectively. The other parameters are chosen as $\alpha=0.1$, $\epsilon=1.2$, and $tol=10^{-4}$. Fig. 4 displays the time evolution of the numerical solutions with $\bar{\phi}=0$ (Fig. 4(a)) and $\bar{\phi}=0.3$ (Fig. 4(b)), respectively. Fig. 4(a) shows the formation of lamellar patterns and Fig. 4(b) shows the formation of hex-cylinder patterns [32,33]. It is obvious that the lamellar patterns are similar to fingerprints; thus, we use them to restore the damaged fingerprint images.

To show the decrease in energy, we study the temporal evolution of the scaled discrete energy $E^h(\phi^n)/E^h(\phi^0)$. The initial conditions and parameters are the same as in Section 4.1. The four discrete energy curves are $E_{short}(\bar{\phi}=0)$, $E_{total}(\bar{\phi}=0)$, $E_{short}(\bar{\phi}=0.3)$, and $E_{total}(\bar{\phi}=0.3)$. The numerical results shown in Fig. 5 suggest that the four normalized discrete energy curves are non-increasing and tend to be constant values with respect to time.

### 4.2. Restoration of different fingerprint shapes

Fingerprint patterns have three basic shapes: whorl, arch, and loop. These shapes are caused by the different directions of the pressure of the subcutaneous tissue on the cuticle of the finger belly [31]. In this subsection, we use the proposed method to restore damaged fingerprint images with two different types of patterns, as shown in Fig. 6. The chosen parameters are $h=1$, $\Delta t=0.1$, $\epsilon=1.2$, $\alpha=0.14$, and $tol=2\times10^{-4}$. Fig. 6(a) is the original fingerprint image. Fig. 6(b) is a schematic representation of an incomplete fingerprint image. Fig. 6(c) is the restored image in the middle of the process. Fig. 6(d) is the final results after applying the fingerprint restoration algorithm. The top row is the processing of the damaged image of the arch pattern, and the bottom row is the processing of the damaged image of the loop pattern. Comparing the restoration results (Fig. 6(d)) with the original fingerprint images (Fig. 6(a)), we can observe that our algorithm can

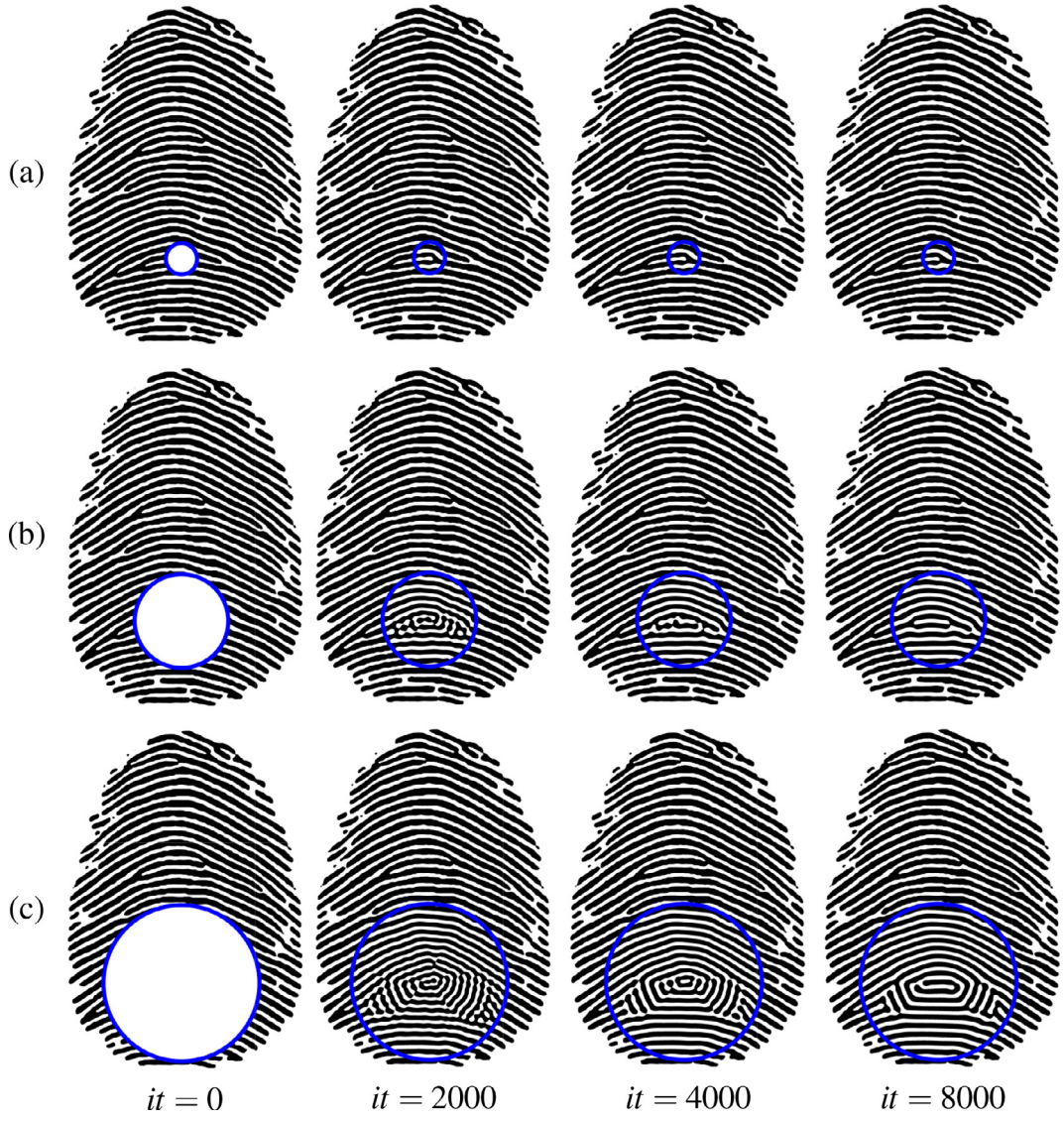


**Fig. 7.** Temporal evolution of the fingerprint restoration with damaged areas of different sizes at different iterations($it$). From Figs. 7(a) through 7(c), the radius of the computational domain is $R = 20,\ 60$, and 100, respectively. From left to right, the number of the iterations is $it = 0$, 2000, 4000, and 8000, respectively.

clearly restore the damaged fingerprint images with different patterns. The proposed method describes the phase separation process and provides a theoretical basis for the physical context.

### 4.3. Effect of damaged domain size

In this subsection, we describe the simulation up to the final time $T = 1600$ with different damage sizes. From Figs. 7(a) through 7(c), the computational domains are $\Omega_{in} = \{(x, y) | \sqrt{(x-163)^2 + (y-118)^2} < R\}$ with radius $R = 20,\ 60$, and 100, respectively. The chosen parameters are $\Delta t = 0.2$, $\epsilon = 1.2$, and $\alpha = 0.1$, respectively. Comparing the final results in Fig. 7(a), (b), and (c), the following remarks can be made: (i) The proposed algorithm works well for relatively large damaged regions, (ii) when the damaged area is small, the separation of the pattern is influenced more by the information of surrounding ridges, (iii) when the damaged area is large, the nonlocal CH equation plays a major role, which is the reason the whorl pattern is formed, and (iv) based on the results, we suggest that the size of the damaged domain may be not larger than $120 \times 120$ pixels ($R = 60$).

### 4.4. Fingerprint restoration with different damaged images

In this subsection, we introduce a simple algorithm for the restoration of fingerprints on an arbitrary domain. Let $\Omega_{in}$ be a user-selected domain that contains the damaged area of the fingerprint image. Fig. 8 (a) illustrates the damaged area of the fingerprint image and its user-selected arbitrary domain $\Omega_{in}$. Let $\Omega_{in}^h$ be a discrete domain and $\partial\Omega_{in}^h$ be its discrete boundary as shown in Fig. 8 (b) with spatial step size $h$.

Now, we only run the simulation on the user-selected domain. The parameters are chosen as $h = 1$, $\Delta t = 0.2$, $\epsilon = 1.1$, $\alpha = 0.1$, and $tol = 8 \times 10^{-5}$. Fig. 9 shows the temporal evolution of the damaged fingerprint image. From left to right, the indicated iterations are $it = 0$, 70, 140, and 210, respectively.

Subsequently, we consider the restoration of fingerprints with different damaged properties based on the latent fingerprint dataset FVC2002DB3_B [34] and FVC2004DB3_B [35] to demonstrate the robustness and effectiveness of our proposed method. Fig. 10 shows visual examples of the proposed method with fingerprint images of different properties. Fig. 10(a) is a real fingerprint of FVC2004DB3_B obtained by the thermal sweeping sensor and

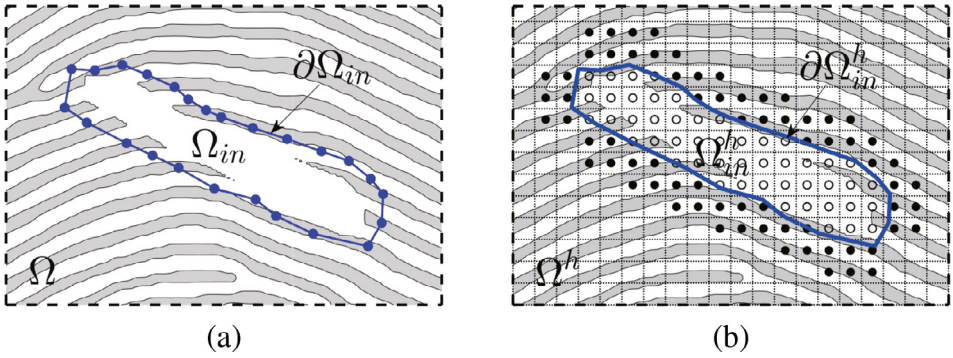


**Fig. 8.** (a) User-selected domain $\Omega_{in}$ and (b) discrete user-selected domain $\Omega_{in}^h$.

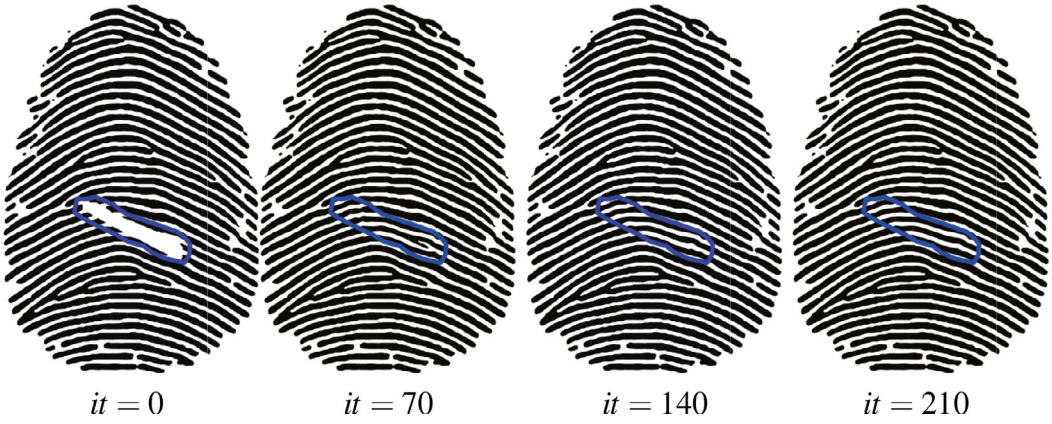


**Fig. 9.** Temporal evolution of the damaged fingerprint image in the user-selected domain at different iterations ($it$). From left to right, the indicated iterations are $it$ = 0, 70, 140, and 210.

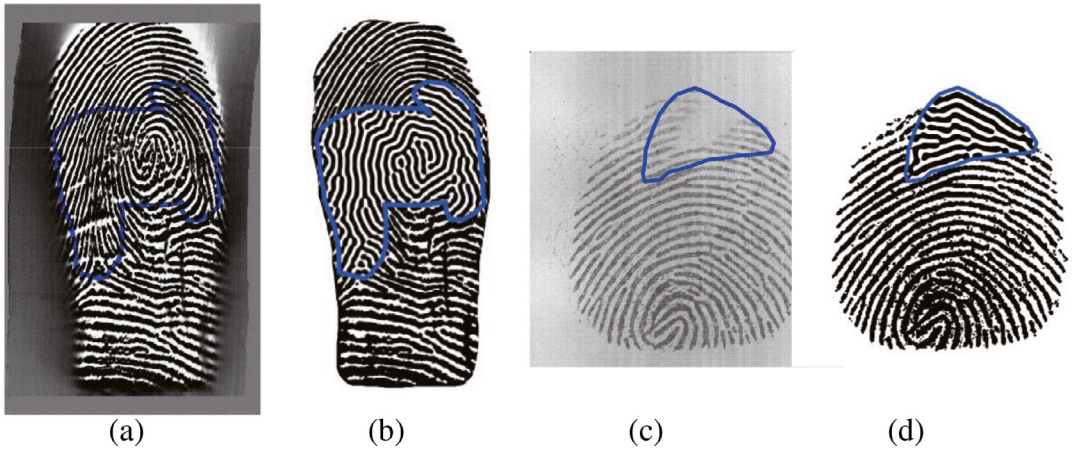


**Fig. 10.** Visual examples demonstrating the performance of the proposed method with fingerprint images of different properties: (a) real fingerprint of FVC2004 DB3_B obtained using a thermal sweeping sensor, (b) restoration of the frayed fingerprint image (a), (c) real fingerprint image of FVC2002 DB3_B obtained using a capacitive sensor, and (d) restoration of (c).

Fig. 10(c) is the real fingerprint image of FVC2002DB3_B obtained by the capacitive sensor with a significant loss. As can be seen from the results, our method can preserve the original information of the fingerprint and repair worn areas (Fig. 10(b)) and generate ridges according to the reaction-diffusion mechanism (Fig. 10(d)).

### 4.5. Parameter sensitivity analysis

This subsection describes a parameter sensitivity analysis conducted for the nonlocal parameter $\alpha$, which is used to ensure that $\phi$ remains close to $\bar{\phi}$. As shown in Fig. 11, we obtain different results with different $\alpha$ values. Figs. 11(a) and (e) are the original fingerprint images with a damaged area. From left to right of Fig. 11, the choosen $\alpha$ is 0.01, 0.1, and 1, respectively. The computational domains of two fingerprints Fig. 11(a) and (e) are $\Omega_{in} = \{(x,y)|\sqrt{(x-105.5)^2+(y-193.5)^2} < 40\}$ and $\Omega_{in} = \{(x,y)|\sqrt{(x-165.5)^2+(y-228.5)^2} < 40\}$, respectively. As shown in the top and bottom rows of Fig. 11, we choose $\epsilon = 0.95$ and $\epsilon = 1.2$, respectively. The other chosen parameters are $h = 1$, $\Delta t = 0.1$, and $tol = 6 \times 10^{-3}$. As can be observed in Figs. 11(c) and (g), the results of a proper $\alpha$ demonstrate that the simulation results are in good agreement with the information retained in the original images. Trough a Comparison of the results in Fig. 11, we can see


**Fig. 11.** Sensitivity analysis for $\alpha$ in a nonlocal term. (a) and (e) original fingerprint images of a damaged area. From left to right, $\alpha = 0.01$, 0.1, and 1.0, respectively. The stop condition $tol$ is the same for all results at $tol = 6 \times 10^{-3}$.

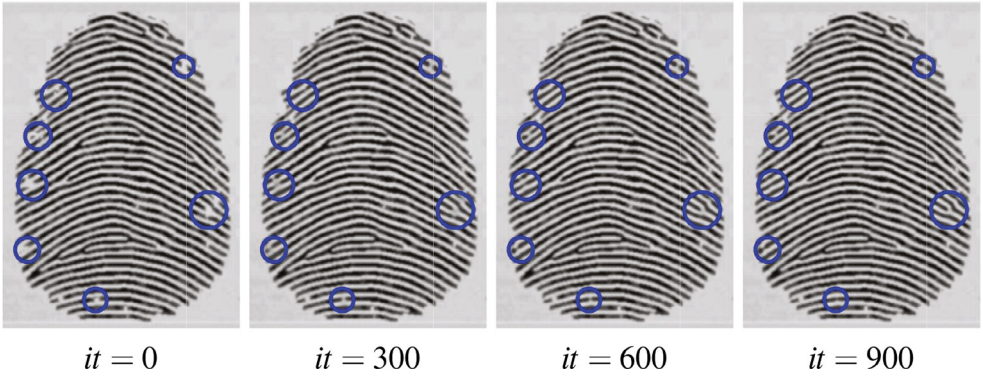


**Fig. 12.** Temporal evolution of an incomplete fingerprint image with multiple minutiae or missing ridges at different iterations ($it$). From left to right, the indicated iterations are $it = 0,\ 300,\ 600,$ and 900, respectively.

that $\alpha$ is inversely proportional to the wave-length of the fingerprint pattern. Furthermore, when $\alpha$ exceeds a certain range, the pattern cannot evolve as shown in Fig. 11(h).

### 4.6. Simultaneous restoration of the fingerprint with multiple damaged regions

In this subsection, to confirm the robustness of the proposed method, we demonstrate the simultaneous restoration of a fingerprint image with multiple damaged regions. Fig. 12 shows the temporal evolution of an incomplete fingerprint image with multiple minutiae or missing ridges. From left to right, the indicated number of iterations is $t = 0$, 300, 600, and 900, respectively. The parameters used here are $h = 1$, $\Delta t = 0.2$, $\epsilon = 1.2$, $\alpha = 0.1$, and $tol = 10^{-3}$. As can be seen from the results, different areas can be restored simultaneously and our method can work well for various fingerprint patterns.

### 4.7. Measurement of quality of damaged fingerprint restoration

As a quantitative measure of the ability of our algorithm to restore damaged fingerprints, we use the peak signal-to-noise ratio (PSNR) and structural similarity index map (SSIM), which are defined as follows:

$$\text{PSNR} = 10\log_{10}\frac{I_{max}^2}{\frac{1}{n_x \cdot n_y}\sum_{i=1}^{n_x}\sum_{j=1}^{n_y}\left(I_{i,j}-K_{i,j}\right)^2}, \tag{11}$$

$$\text{SSIM} = \frac{(2\mu_I\mu_K+c_1)(2\sigma_{IK}+c_2)}{(\mu_I^2+\mu_K^2+c_1)(\sigma_I^2+\sigma_K^2+c_2)}, \tag{12}$$

where $I_{i,j} = f_{i,j}/(f_{\max} - f_{\min})$, $K_{i,j} = 0.5(\phi_{i,j}+1)$; $\mu_I$, $\mu_K$ are the averages of $I$ and $K$, respectively, and $\sigma_I^2$, $\sigma_K^2$ are the variances of $I$ and $K$, respectively. Here, $\sigma_{IK}$ is the covariance of $I$ and $K$, $c_1 = (0.01L)^2, and c_2 = (0.03L)^2$ with $L = 1$. High PSNR and SSIM values indicate a good restoration of the fingerprint image. We run the simulation up to the terminal time $T = 1200$ with $\Delta t = 0.2$,

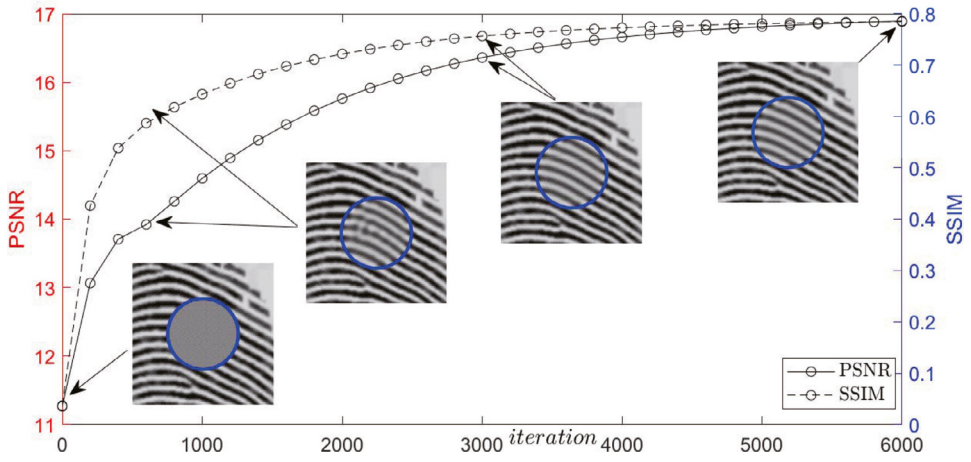


**Fig. 13.** Time evolution of a quantitative measure of the ability to restore a damaged image.

**Table 1**
Comparison of the average PSNR between the proposed method and other existing approaches. The values in the first four columns were taken directly from [36]. The results were obtained based on the FVC2004 dataset.

| Method | Corruption | | | | |
|---|---|---|---|---|---|
| | SN+Sc | SN+W | SN+D | SN+OP | SN+VN |
| No enhancement [36] | 8.07 | 9.66 | 7.38 | 8.08 | 8.59 |
| Gabor filtering [36] | 7.12 | 6.99 | 7.14 | 6.24 | 6.87 |
| BaseNet [36] | 18.93 | 17.95 | 16.74 | 10.55 | 16.17 |
| BaseNet-bin [36] | 19.61 | 18.21 | 15.74 | 9.87 | 17.25 |
| OFFIENet-tri [36] | **21.67** | **20.87** | **17.25** | 10.96 | 19.01 |
| OFFIENet-tri-shared [36] | 21.28 | 20.45 | 17.24 | **11.63** | 18.91 |
| Our method | 19.89 | 20.12 | 16.71 | 11.54 | **18.92** |

$\epsilon = 1.2$, and $\alpha = 0.1$. The computational domain shown in Fig. 13 is $\Omega_{in} = \{(x, y) | \sqrt{(x-205.5)^2 + (y-328.5)^2} < 40\}$. Fig. 13 shows the PSNR and SSIM over the iterations. The small-inscribed figures are the temporal evolution of the restoration process in the computational domain at $it = 0$, 600, 3000, and 6000. At $it = 6000$, the PSNR and SSIM scores are 16.8933 and 0.7848, respectively. To verify the efficiency of the proposed method, we compute the PSNR and compare the results with the value of other state-of-art methods in [36], as shown in Table 1. From these computational results, we can confirm that the proposed algorithm performs better than the Gabor filtering and BaseNet-bin methods for different types of corrupted fingerprint images. Compared with the new OFFIENet-tri-shared method in [36], the performance of our method is only slightly worse on SN+Sc and SN+OP corrupted fingerprint images. Note that this comparison is unfair in some respects because the proposed method works well without a training cost, whereas network-based methods require expensive training costs to improve the corresponding accuracy.

We then conduct a numerical comparison, as shown in Table 2, to evaluate the equal-error rates (EERs) of different methods based on different datasets. We define the false rejection rate (FRR) and false acceptance rate (FAR) [37] as follows:

$$\text{FRR} = \frac{\text{Number of rejected genuine claims}}{\text{Total number of genuine accesses}} \times 100\%, \tag{13}$$

$$\text{FAR} = \frac{\text{Number of accepted imposter claims}}{\text{Total number of imposter accesses}} \times 100\%, \tag{14}$$

where FAR is the ratio of a fake fingerprint being mistaken as a genuine fingerprint, and FRR is the probability that a genuine fingerprint is improperly rejected as a fake fingerprint [40]. Therefore, an EER can be calculated as follows:

$$\text{EER} = \frac{\text{FAR} + \text{FRR}}{2}, \quad \text{when} \quad \text{FAR} = \text{FRR}. \tag{15}$$

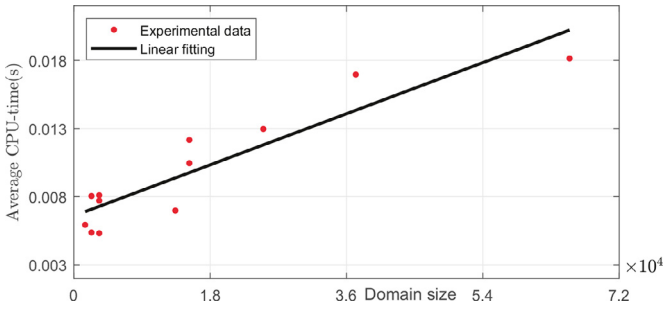


**Fig. 14.** Experimental data and linear fitting of average CPU time per iteration versus the domain size. The average CPU time refers to the quotient of the total CPU time and the iterations. The domain size is the number of pixels within the computational domain.

A smaller EER corresponds to a better reconstruction by the algorithm. As can be seen from Table 2, the proposed method demonstrated a good performance and resulted in a smaller EER on datasets FVC2002DB4, FVC2004DB1, and FVC2004DB4 as compared with other methods in previous studies. Although the results obtained by our method do not provide the highest accuracy for the numerical test based on the corresponding datasets, there are various advantages compared with the existing algorithm: (i) Our algorithm does not require much training time and reduces the time cost by sacrificing an acceptable level of accuracy, (ii) our method does not depend on a specific dataset, and (iii) the proposed scheme is simple to apply.

### 4.8. Computational cost

Finally, we present the performance of all tests in Table 3 and Fig. 14. First, we record the CPU time for each test. We then calculate the average CPU time per iteration and fit the curve with the computational domain size using the MATLAB routine polyfit. The CPU times (in seconds) of the computations, which are performed in MATLAB, are measured on a computer with a 3-GHz CPU and 8GB of RAM. From the results shown in Table 3, we can see that the computational cost is low. Based on the results in Fig. 14, we can see that the computational cost is linear with respect to the domain size. Several advantages of the proposed method have been summarized as follows: (i) The Gauss–Seidel-type iterative method, which is one of the simplest iterative methods, can work well to solve the proposed method (Eqs. (1)–(4)). (ii) The convergence rate of the computational cost of Gauss–Seidel-type algorithm is linear with respect to the image size. Finally, (iii) the

**Table 2**
Comparison of the EER(%) of the proposed method with existing approaches. The values in the first four rows were taken directly from the respective papers.

| Method | FVC2002 | | | FVC2004 | | |
|---|---|---|---|---|---|---|
| | DB1 | DB2 | DB4 | DB1 | DB2 | DB4 |
| Yang et al. [37] | – | – | – | 3.12 | 2.50 | 4.19 |
| Sutthiwichaiporn et al. [38] | 2.07 | 0.88 | 1.53 | 5.65 | 5.46 | 2.59 |
| Verifinger [39] | 0.25 | 0.31 | 0.35 | 1.80 | **0.82** | 1.06 |
| Wong et al. [36] | **0.18** | **0.14** | 0.25 | 1.83 | 3.66 | 0.32 |
| MCC [41] | 1.00 | 0.49 | – | – | – | – |
| OFFIENet-tri + MCC [36] | 0.43 | **0.14** | 0.43 | 2.93 | 3.43 | 1.14 |
| Gabor filtering + MCC [36] | 0.71 | 0.47 | 2.86 | 3.97 | 5.54 | 5.71 |
| BaseNet + MCC [36] | 0.57 | **0.14** | 1.29 | 3.43 | 4.43 | 1.57 |
| Our method | 0.25 | 0.62 | **0.23** | **1.24** | 0.93 | **0.30** |

**Table 3**
Total CPU time(s) of every numerical test applied in this study.

| Figure | Fig. 1 | Fig. 4 | Fig. 6(top) | Fig. 6(bottom) | |
|---|---|---|---|---|---|
| CPU time(s) | 8.39 | 7.26 | 6.72 | 7.48 | |
| Figure | Fig. 7(a) | Fig. 7(b) | Fig. 7(c) | Fig. 10(b) | Fig. 10(d) |
| CPU time(s) | 6.87 | 28.43 | 63.04 | 59.13 | 53.21 |
| Figure | Fig. 13 | Fig. 12 | Fig. 11(top) | Fig. 11(bottom) | Fig. 9 |
| CPU time(s) | 6.99 | 48.23 | 6.14 | 6.34 | 5.23 |

**Table 4**
Average CPU time(s) for the restoration on datasets FVC2000, FVC2002, and FVC2004 with our proposed method.

| Datasets / CPU time(s) / Subsets | DB1_B | DB2_B | DB3_B | DB4_B |
|---|---|---|---|---|
| FVC2000 [42] | 53.96 | 40.47 | 54.90 | 43.74 |
| FVC2002 [34] | 55.41 | 37.77 | 58.72 | 42.42 |
| FVC2004 [35] | 53.26 | 43.37 | 44.23 | 45.33 |

computational cost is not $O(N)$ but $O(N_{is})$, where $N$ and $N_{is}$ denote the total and damaged domain sizes, respectively, because the proposed method is only performed on the damaged fingerprint region. Therefore, our method is simple and fast.

However, we cannot implement an existing method to test the corresponding CPU time. Thus, we only demonstrate the average CPU time for the proposed method with different datasets in Table 4. Because the purpose of this study is to restore the corrupted region in the fingerprint images, detection of the damaged area should be applied before the restoration. The CPU time recorded in Table 4 only includes the computation cost of the restoration without detection. From the results, we can see that the average CPU time on the respective datasets FVC2000, FVC2002, and FVC2004 is not high, which implies that our algorithm is relatively efficient.

## 5. Conclusions

In this study, we developed a simple and efficient fingerprint image restoration algorithm using a phase-field model based on the nonlocal CH equation. The mathematical equation consists of local and nonlocal terms associated with short- and long-range interactions respectively by taking a variational derivative of the total phenomenological free energy. We use an extremely simple numerical solution algorithm that is practically stable and explicit Gauss–Seidel-type iterative. Compared to existing fingerprint restoration algorithms, the proposed framework provides a mathematical foundation for the evolution of the fingerprint ridges based on a reaction-diffusion system. To the best of our knowledge, the present approach is the first attempt to restore a fingerprint image using the nonlocal phase-field model. Several computational experiments demonstrated the superior performance of the proposed fingerprint restoration algorithm.

## Declaration of Competing Interest

The authors declare that there is no conflict of interests regarding the publication of this article.

## Acknowledgment

Y.B. Li is supported by the Fundamental Research Funds for the Central Universities (No.XTR042019005). The corresponding author (J.S. Kim) was supported by the National Research Foundation(NRF), Korea, under project BK21 FOUR. The authors appreciate the reviewers for their constructive comments, which have improved the quality of this paper.

## References

[1] A.A.A. Hamid, M.S.M. Rahim, A.S. Al-Mazyad, T. Saba, Analysis of proposed noise detection & removal technique in degraded fingerprint images, 3D Res. 38 (2015) 1–8.
[2] D.K. Misra, S.P. Tripathi, A study report on fingerprint image enhancement methods, IJCSC 3 (2012) 163–170.
[3] X. Yang, D. Wang, Z. Yang, A fingerprint inpainting technique using improved partial differential equation methods, In International Conference on Graphic and Image Processing (ICGIP 2011) (2011) 828571.
[4] L. Hong, Y. Wan, A. Jain, Fingerprint image enhancement: algorithm and performance evaluation, IEEE Trans. Pattern Anal. Mach. Intell. 20 (8) (1998) 777–789.
[5] C. Gottschlich, Curved-region-based ridge frequency estimation and curved gabor filters for fingerprint image enhancement, IEEE Trans. Image Process. 21 (4) (2012) 2220–2227.
[6] E. Zhu, J. Yin, G. Zhang, Fingerprint enhancement using circular gabor filter, in: International Conference on Image Analysis and Recognition, Springer, 2004, pp. 750–758.
[7] W. Wang, J. Li, F. Huang, Design and implementation of log-gabor filter in fingerprint image enhancement, Pattern Recognit. Lett. 29 (2008) 301–308.
[8] J. Feng, A.K. Jain, Fingerprint reconstruction: from minutiae to phase, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2011) 209–223.
[9] S. Li, A.C. Kot, An improved scheme for full fingerprint reconstruction, IEEE Trans. Inf. Forensics Security 7 (2012) 1906–1912.
[10] C.-T. Hsieh, E. Lar, Y.C. Wang, An effective algorithm for fingerprint image enhancement based on wavelet transform, Pattern Recognit. 36 (2003) 303–312.
[11] P. Sutthiwichaiporn, V. Areekul, Adaptive boosted spectral filtering for progressive fingerprint enhancement, Pattern Recognit 46 (2013) 2465–2486.
[12] J. Li, J. Feng, C.C.J. Kuo, Deep convolutional neural network for latent fingerprint enhancement, Signal Process-Image 60 (2018) 52–63.
[13] R. Prabhu, X. Yu, Z. Wang, D. Liu, A.A. Jiang, U-finger: multiscale dilated convolutional network for fingerprint image denoising and inpainting, Inpainting and Denoising Challenges (2019) 45–50.
[14] M. Yadav, N. Tiwari, Performance comparison of image restoration techniques using CNN and their applications, 2021 5th International Conference on Computing Methodologies and Communication (ICCMC) (2021) 1146–1151.

[15] A. Witkin, M. Kass, Reaction-diffusion textures, ACM SIGGRAPH Computer Graphics 25 (1991) 299–308.
[16] K. Ito, T. Aoki, T. Higuchi, Fingerprint restoration using digital reaction-diffusion system and its evaluation, IEICE Trans. Fundamentals E84–A (2003) 1916–1924.
[17] T. Ohta, K. Kawasaki, Equilibrium morphology of block copolymer melts, Macromolecules 19 (1986) 2621–2632.
[18] D. Jeong, Y. Choi, J. Kim, Numerical investigation of local defectiveness control of diblock copolymer patterns, Condens. Matter. Phys. 19 (2016) 1–10.
[19] D. Jeong, S. Lee, Y. Choi, J. Kim, Energy-minimizing wavelengths of equilibrium states for diblock copolymers in the hex-cylinder phase, Curr. Appl. Phys. 15 (2015) 799–804.
[20] I. Ohnishi, Y. Nishiura, M. Imai, Y. Matsushita, Analytical solutions describing the phase separation driven by a free energy functional containing a long-range interaction term, Chaos 9 (1999) 329–341.
[21] A. Segatti, On the hyperbolic relaxation of the cahn–hilliard equation in 3d: approximation and long time behaviour, Math. Mod. Meth. Appl. Sci. 17 (2007) 411–437.
[22] R. Choksi, M. Maras, J.F. Williams, 2D phase diagram for minimizers of a cahn–hilliard functional with long-range interactions, SIAM J. Appl. Dyn. Syst. 10 (2011) 1344–1362.
[23] D. Jeong, J. Shin, Y. Li, Y. Choi, J.H. Jung, S. Lee, J. Kim, Numerical analysis of energy-minimizing wavelengths of equilibrium states for diblock copolymers, Curr. Appl. Phys. 14 (2014) 1263–1272.
[24] Y. Nishiura, I. Ohnishi, Some mathematical aspects of the micro-phase separation in diblock copolymers, Phys. D 85 (1995) 31–39.
[25] R. Choksi, M.A. Peletier, J.F. Williams, On the phase diagram for microphase separation of diblock copolymers: an approach via a nonlocal cahn–hilliard functional, SIAM J. Appl. Math. 69 (2009) 1712–1738.
[26] D. Jeong, J. Kim, Microphase separation patterns in diblock copolymers on curved surfaces using a nonlocal Cahn–Hilliard equation, Eur. Phys. J. E. 38 (2015) 117.
[27] Q. Du, L. Ju, X. Li, Z. Qiao, Stabilized linear semi-implicit schemes for the nonlocal cahn–hilliard equation, J. Comput. Phys. 363 (2018) 39–54.
[28] Y. Li, J. Wang, B. Lu, D. Jeong, J. Kim, Multicomponent volume reconstruction from slice data using a modified multicomponent cahn–hilliard system, Pattern Recogn. 93 (2019) 124–133.
[29] Y. Li, S. Lan, X. Liu, B. Lu, L. Wang, An efficient volume repairing method by using a modified Allen–Cahn equation, Pattern Recogn. 107 (2020) 107478.
[30] Y. Tu, Z. Yao, J. Xu, Y. Liu, Z. Zhang, Fingerprint restoration using cubic bezier curve, BMC Bioinformatics. 21 (2020) 1–19.
[31] K. Karu, A.k. Jain, Fingerprint classification, Pattern Recogn. 29 (1996) 389–404.
[32] Y. Li, H.G. Lee, B. Xia, J. Kim, A compact fourth-order finite difference scheme for the three-dimensional Cahn–Hilliard equation, Comput. Phys. Commun. 200 (2016) 108–116.
[33] Y. Li, J. Kim, N. Wang, An unconditionally energy-stable second-order time-accurate scheme for the Cahn–Hilliard equation on surfaces, Commun. Nonlinear. Sci. 53 (2017) 213–227.
[34] D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, A.K. Jain, FVC2002: fingerprint verification competition, in: Proc. International Conference on Pattern Recognition (ICPR), 2002, pp. 744–747. Quebec City, Canada, August
[35] D. Maio, D. Maltoni, R. Cappelli, J. Wayman, A.K. Jain, FVC2004: fingerprint verification competition 2004, in: Springer LNCS–3072, International Conference on Biometric Authentication (ICBA), 2004, pp. 1–7. Hong Kong, China
[36] W.J. Wong, S.H. Lai, Multi-task CNN for restoring corrupted fingerprint images, Pattern Recognit 101 (2020) 107203.
[37] J. Yang, N. Xiong, A.V. Vasilakos, Two-stage enhancement scheme for low-quality fingerprint images by learning from the images, IEEE Trans. Hum. Mach. Syst. 43 (2013) 235–248.
[38] P. Sutthiwichaiporn, V. Areekul, Adaptive boosted spectral filtering for progressive fingerprint enhancement, Pattern Recognit 46 (2013) 2465–2486.
[39] ????, (????). Neurotechnology, Verifinger SDK, (http://www.neurotechnology.com/verifinger.html).
[40] X. Li, W. Cheng, C. Yuan, W. Gu, B. Yang, Q. Cui, Fingerprint liveness detection based on fine-grained feature fusion for intelligent devices, Mathematics 8 (2020) 517.
[41] M. Ferrara, D. Maltoni, R. Cappelli, Noninvertible minutia cylinder-code representation, IEEE Trans. Inf. ForensicsSecur 7 (6) (2012) 1727–1737.
[42] D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, A.K. Jain, FVC2000: Fingerprint verification competition, IEEE Trans. Pattern Analysis and Machine Intelligence 24 (2002) 402–412.
[43] L.S. Davis, A survey of edge detection techniques, Comput. Graph. Image Process 4 (1975) 248–270.
[44] Z.J. Hou, G.W. Wei, A new approach to edge detection, Pattern Recogn 35 (2002) 1559–1570.
[45] R. Nevatia, K.R. Babu, Linear feature extraction and description, Comput. Graph. Image Process. 13 (1980) 257–269.
[46] X. Wang, K.K. Paliwal, Feature extraction and dimensionality reduction algorithms and their applications in vowel recognition, Pattern Recogn 36 (2003) 2429–2439.

**Yibao Li** received the B.S. degree from the Department of Mathematics, Xin University of Technology, China in 2007. He received the M.S. and Ph.D. degrees in Applied Mathematics from Korea University, Korea, in 2011 and 2013, respectively. Before he joined the School of Mathematics and Statistics, Xin Jiaotong University, China in 2014, he held a research position in Department of Computational Science and Engineering, Yonsei University, Korea. He is currently a full professor at the Department of Applied Mathematics. His research interests include image processing, computational fluid dynamics, and scientific computing.

**Qing Xia** received the B.S. degree from the School of mathematics and Statistics, Xin Jiaotong University, China in 2019. Now, he is under a Ph.D. candidate in School of mathematics and Statistics, Xin Jiaotong University, China. His research interests include Additive Manufacturing (3D Printing), computational fluid dynamics, and scientific computing.

**Chaeyoung Lee** is a postdoctoral researcher at Korea University. She received the M.S. and Ph.D. degrees in Applied Mathematics from Korea University, Korea, in 2014 and 2021, respectively. And she received B.S. degree in Mathematics from Korea University in 2012. Her research interests include numerical analysis, Mathematical modeling, computational finance, and scientific computing.

**Sangkwon Kim** is a Ph.D. candidate at the Department of Mathematics, Korea University, Korea. And he received his M.S. degree in Applied Mathematics and B.S. degree in Mathematics from the Korea University in 2019 and Hanshin University in 2012, respectively. His research interests are in computational finance and computational fluid dynamics and deep learning in physics.

**Junseok Kim** received his Ph.D. in Applied Mathematics from the University of Minnesota, U.S.A. in 2002. He also received his B.S. degree from the Department of Mathematics Education, Korea University, Korea in 1995. He joined the faculty of Korea University, Korea in 2008 where he is currently a full professor at the Department of Mathematics. His research interests are in computational finance and computational fluid dynamics.