Three-dimensional volume reconstruction from slice data using phase-field models☆

Yibao Li[a], Jaemin Shin[b], Yongho Choi[c], Junseok Kim[c,*]

[a] *School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China*
[b] *Institute of Mathematical Sciences, Ewha W. University, Seoul 120-750, Republic of Korea*
[c] *Department of Mathematics, Korea University, Seoul 136-713, Republic of Korea*

A R T I C L E   I N F O

A B S T R A C T

We propose the application of a phase-field framework for three-dimensional volume reconstruction using slice data. The proposed method is based on the Allen–Cahn and Cahn–Hilliard equations, and the algorithm consists of two steps. First, we perform image segmentation on the given raw data using a modified Allen–Cahn equation. Second, we reconstruct a three-dimensional volume using a modified Cahn–Hilliard equation. In the modified Cahn–Hilliard equation, a fidelity term is introduced to keep the solution close to the slice data. The numerical methods use a hybrid method and an unconditionally stable nonlinear splitting scheme. The resulting discrete equations are solved using a multigrid method. The experiments on synthetic and real medical images are performed to demonstrate the accuracy and efficiency of the proposed method.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

Developing reconstruction algorithms attracts a significant amount of attention because the three-dimensional (3D) volume reconstruction from a sequence of medical images has numerous applications such as medical diagnostic, plastic and artificial limb surgery, virtual surgery system, anatomy teaching, and treatment planning [2,3]. Various algorithms have been proposed to reconstruct a surface or volume from a set of planar cross-sections. A method which combines the elastic interpolation algorithm, spline theory, and surface consistency theorem was proposed for reconstructing a smooth 3D object from serial cross sections [4]. Guo et al. presented a morphology-based mathematical method to implement the interpolation by means of a combined operation of weighted dilation and erosion [5]. Jones and Chen constructed surfaces from cross sections using a field function in each slice and the marching cubes algorithm to generate a surface consisting of polygonal facets [6]. In shape-based interpolation method, the signed distance value of a voxel to the edges of a cross section is calculated. After each slice has been assigned the distance values, distances for other slices are defined using

an interpolation. Then, the volume is obtained by the zero isosurface [7]. For other approaches to 3D reconstruction, refer to [8–11].

In this paper, we present a phase-field framework for 3D volume reconstruction using slice data. The proposed algorithm has two steps: image segmentation and 3D reconstruction using two partial differential equations, which are the modified Allen–Cahn and Cahn–Hilliard equations. This paper is organized as follows. In Section 2, we describe the governing equations for the image segmentation and volume reconstruction. Section 3 describes a practically stabilized nonlinear splitting scheme for the volume reconstruction and presents a multigrid method. In Section 4, we perform numerical experiments with synthetic and real medical images. Finally, our conclusion is given in Section 5.

2. Reconstruction process

In this section, we propose a numerical algorithm for 3D volume reconstruction from slice data. We start with an illustration of the process of the proposed algorithm when we have a set of cross-sectional slice data (Fig. 1(a)).

For the first stage, by using a modified Allen–Cahn equation, the image segmentation algorithm is applied for the given slice data f to obtain the phase-field function ψ (segmented image). Fig. 1(b) shows the filled contour plots of segmented slice data ψ. For the second stage, by using a modified Cahn–Hilliard equation and the segmented slice data, we reconstruct the volume (Fig. 1(c)).
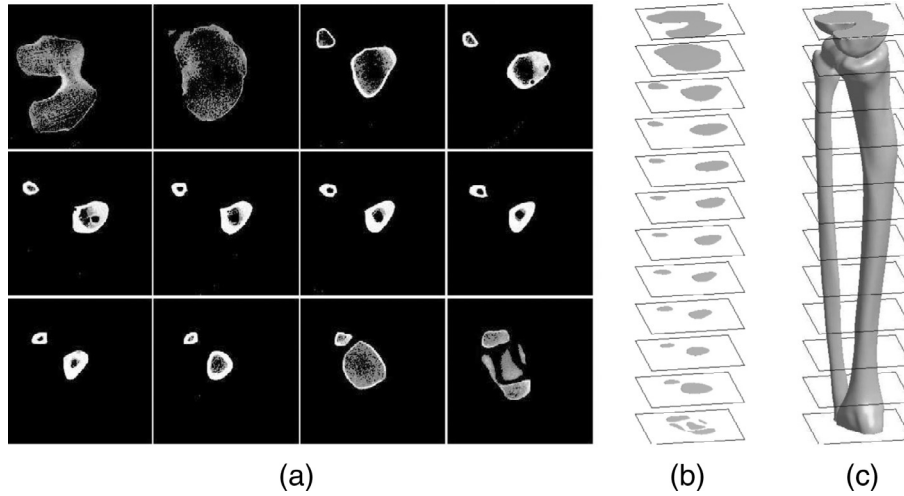

Fig. 1. Volume reconstruction from slice data: (a) given slice data, (b) filled contour plots of segmented slice data, and (c) iso-surface of the reconstructed volume.

2.1. Image segmentation: modified Allen–Cahn equation

Since the original image may have noises (Fig. 2(a) and (b)), to prepare the numerical slice data ψ (Fig. 2(c)) before applying the proposed method for volume reconstruction, we use an image segmentation algorithm [12–18]. The method we use for image segmentation is based on the Allen–Cahn equation and it enforces the diffuse interface to be the hyperbolic tangent profile. The geometric active contour model based on the mean curvature motion is given by the following evolution equation [14]:

$$\frac{\partial \psi(\mathbf{x}, t)}{\partial t} = g(f_0(\mathbf{x}))\left(-\frac{F'(\psi(\mathbf{x},t))}{\epsilon^2} + \Delta\psi(\mathbf{x},t)\right) + \beta g(f_0(\mathbf{x}))F(\psi(\mathbf{x},t)),$$

(1)

where $\mathbf{x} = (x, y)$ and $f_0(\mathbf{x}) = (f(\mathbf{x}) - f_{\min})/(f_{\max} - f_{\min})$. Here, $f_{\max}$ and $f_{\min}$ are the maximum and minimum values of the given slice image $f(\mathbf{x})$, respectively. Here, $\psi(\mathbf{x}, t)$ is a phase-field function which is close to 1 or -1. The function $g(f_0(\mathbf{x})) = 1/[1 + |\nabla(G_\sigma * f_0)(\mathbf{x})|^2]$ is the edge-stopping function, which stops the evolution when the contour reaches the edge. The function $(G_\sigma * f_0)(\mathbf{x}) = \int_\Omega G_\sigma(\mathbf{x} - \mathbf{y})f_0(\mathbf{y})d\mathbf{y}$ is the convolution of the given image f_0 with the Gaussian function $G_\sigma(\mathbf{x}) = \frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}$. Here, $F(\phi) = 0.25(\phi^2 - 1)^2$, ϵ is a constant that is related to the phase transition width, and β is a parameter. In this paper, we use $\sigma = 1.5$ and $\beta = 50,000$.

We apply a hybrid method [14] to solve Eq. (1) and we outline the numerical solution algorithm for the sake of completeness. We discretize Eq. (1) in $\Omega = (a, b) \times (c, d)$. Let N_x and N_y be positive even
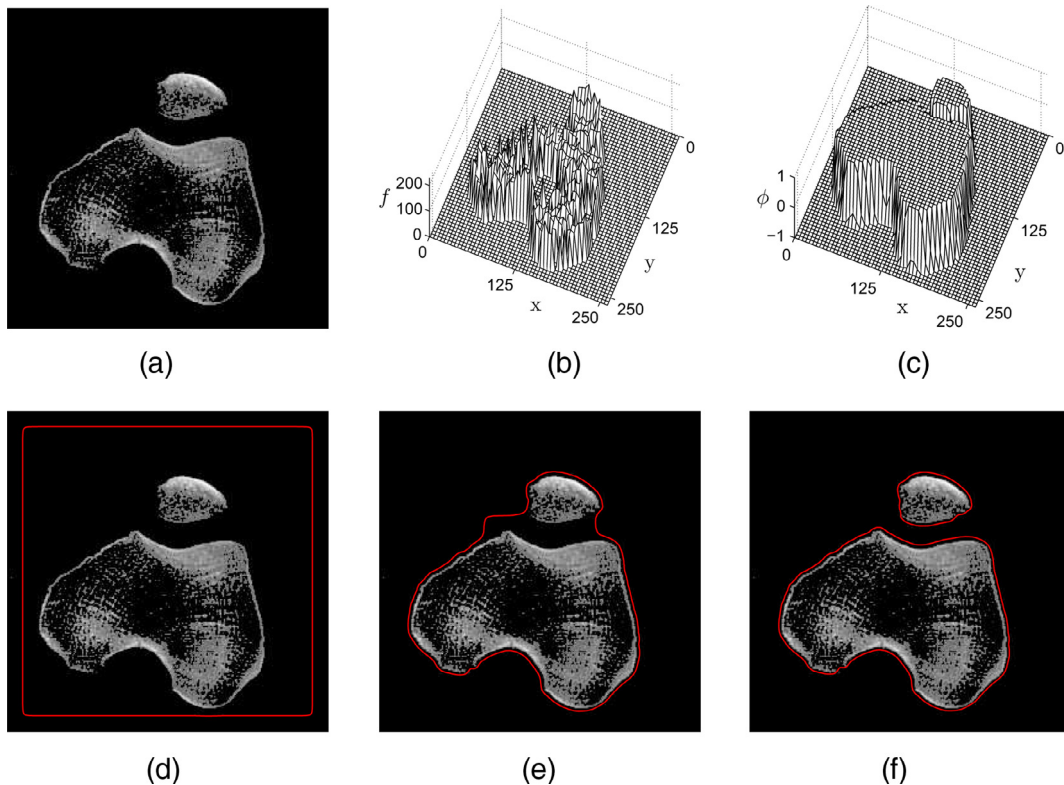

Fig. 2. Image segmentation process of the given slice data: (a) given medical image, (b) mesh plot of the given image, (c) mesh plot of the final result, and (d)–(f) evolutions of image segmentation, in which the curves are the zero contours of $\psi(x, y, t)$.
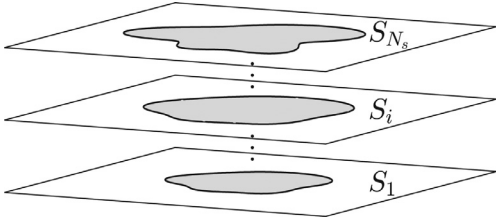
Fig. 3. Schematic of slice data.

integers, $h = (b - a)/N_x = (d - c)/N_y$ be the uniform mesh size. Let ψ_{ij}^n be approximations of $\psi(x_i, y_j, n\Delta t)$, where $x_i = a + (i - 0.5)h$, $y_j = (j - 0.5)h$, and Δt is the time step. The discrete edge function is defined by

$$g(f_0)_{ij} = \frac{1}{1 + (G_\sigma * f_0)_{x,ij}^2 + (G_\sigma * f_0)_{y,ij}^2},$$

where $(G_\sigma * f_0)_{x,ij} = [(G_\sigma * f_0)_{i+1,j} - (G_\sigma * f_0)_{i-1,j}]/(2h)$ and $(G_\sigma * f_0)_{y,ij} = [(G_\sigma * f_0)_{i,j+1} - (G_\sigma * f_0)_{i,j-1}]/(2h)$. The discrete convolution is defined as

$$(G_\sigma * f_0)_{ij} = \sum_{p=i-1}^{i+1} \sum_{q=j-1}^{j+1} \frac{f_{0\,pq}}{2\pi\sigma^2} e^{-\frac{[(i-p)^2 + (j-q)^2]h^2}{2\sigma^2}}.$$

The zero Neumann boundary condition is used. Then the following operator splitting numerical algorithm for Eq. (1) is as follows:

$$\frac{\psi_{ij}^* - \psi_{ij}^n}{\Delta t} = g_{ij}\Delta_d \psi_{ij}^* + \beta g_{ij} F(\psi_{ij}^n), \qquad (2)$$

which is solved by a multigrid method [1]. Here Δ_d is the standard five point discrete Laplacian operator. Next, using the method of separation of variables we analytically solve the equation

$$\psi_t = g\frac{\psi - \psi^3}{\epsilon^2}, \qquad (3)$$

with the condition $\psi^n = \psi^*$. Then the solution of Eq. (3) at $t = (n + 1)\Delta t$ is given as

$$\psi_{ij}^{n+1} = \psi_{ij}^* / \sqrt{e^{-\frac{2g_{ij}\Delta t}{\epsilon^2}} + (\psi_{ij}^*)^2 \left(1 - e^{-\frac{2g_{ij}\Delta t}{\epsilon^2}}\right)}.$$

Fig. 2 illustrates the process of image segmentation using Eq. (1). For the given medical image in Fig. 2(a), we define the scaled image $f_0(\mathbf{x})$ and initialize $\psi(\mathbf{x}, 0)$ as $\psi(\mathbf{x}, 0) = 1$ if $\mathbf{x}$ is inside the square contour, $\psi(\mathbf{x}, 0) = -1$ otherwise (see Fig. 2(d)). In Fig. 2(e) and (f), the initial contour evolves until it reaches the boundary of the image through the motion created by the mean curvature (the first term) and the second term on the right-hand side of Eq. (1). The term $\beta g(f_0(\mathbf{x}))F(\psi(\mathbf{x}, t))$ evolves the contour beyond the non-convex and disconnected regions.
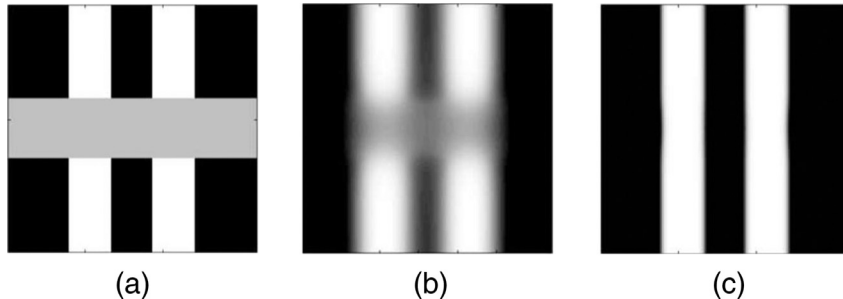
2.2. 3D reconstruction: Modified Cahn–Hilliard equation

In order to reconstruct a 3D volume from a set of segmented slice data, we consider the modified Cahn–Hilliard equation which contains a fidelity term:

$$\frac{\partial \phi(\mathbf{x}, t)}{\partial t} = \Delta \mu(\mathbf{x}, t) + \lambda(\mathbf{x})(\psi(\mathbf{x}) - \phi(\mathbf{x}, t)), \quad \mathbf{x} \in \Omega, \quad 0 < t \leq T,$$
$$(4)$$

$$\mu(\mathbf{x}, t) = F'(\phi(\mathbf{x}, t)) - \epsilon^2 \Delta \phi(\mathbf{x}, t), \qquad (5)$$

where $\mathbf{x} = (x, y, z)$, $\Omega = (0, L_x) \times (0, L_y) \times (0, L_z)$ is a domain, and

$$\lambda(\mathbf{x}) = \begin{cases} \lambda_0, & \text{if } \mathbf{x} \text{ is in the given slice data}, \\ 0, & \text{otherwise}. \end{cases}$$

Here, $\phi(\mathbf{x}, t)$ is a phase-field function which is close to 1 or -1 for the reconstructed volume's respective interior and exterior. The surface of the volume is represented by the zero-level set of ϕ. Let $\psi(x, y, z_i)$ for $i = 1, \ldots, N_s$ be the segmented slice data obtained by performing the image segmentation algorithm on $S_i := \Omega \cap \{z = z_i\}$, where $z_1 = 0$ and $z_{N_s} = L_z$. Here, N_s is the number of slice data. Fig. 3 shows the slice data's schematics.

To define the initial condition $\phi(\mathbf{x}, 0)$, we use a linear interpolation between two consecutive slices: We have

$$\phi(x, y, \theta z_{i+1} + (1 - \theta)z_i) = \theta \psi(x, y, z_{i+1}) + (1 - \theta)\psi(x, y, z_i), \qquad (6)$$

for $0 \leq \theta \leq 1$ and $i = 1, \ldots, N_s - 1$. Homogeneous Neumann boundary conditions for both ϕ and μ are applied: $\mathbf{n} \cdot \nabla \phi = \mathbf{n} \cdot \nabla \mu = 0$ on $\partial \Omega$ except $z = z_1$ and $z = z_{N_s}$, where Dirichlet boundary condition for ϕ is applied. Let us consider the following total energy functional:
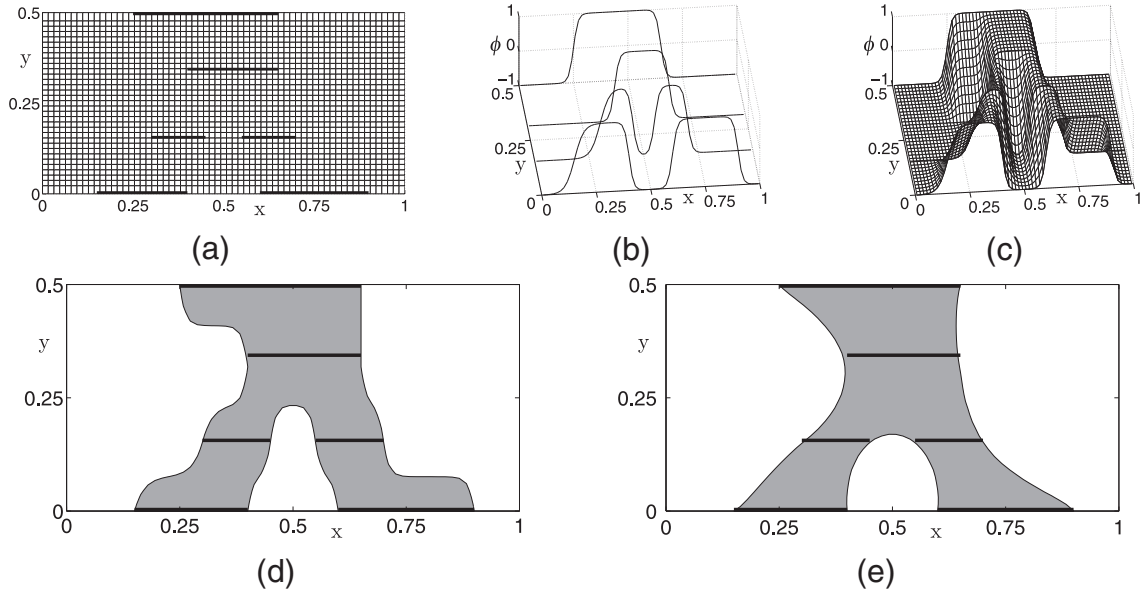
$$\mathcal{E}(\phi) = \mathcal{E}_{CH}(\phi) + \mathcal{E}_F(\phi), \qquad (7)$$

where

$$\mathcal{E}_{CH}(\phi) = \int_\Omega \left[F(\phi(\mathbf{x}, t)) + \frac{\epsilon^2}{2}|\nabla \phi(\mathbf{x}, t)|^2\right]d\mathbf{x}, \qquad (8)$$

$$\mathcal{E}_F(\phi) = \sum_{i=1}^{N_s} \int_{S_i} \frac{\lambda_0}{2}(\psi(\mathbf{x}) - \phi(\mathbf{x}, t))^2 dx dy. \qquad (9)$$

Then, the first term and second term in Eq. (4) are negative gradient descents with respect to the H^{-1} and L^2 inner products for the energies $\mathcal{E}_{CH}$ and $\mathcal{E}_F$, respectively [19]. It should be pointed that Eq. (4) is neither a gradient flow in H^{-1} nor L^2. If $\lambda_0 = 0$, then Eqs. (4) and (5) become the Cahn–Hilliard equation [20], which has been widely employed as a useful mathematical model in order to understand the phase separation phenomena observed when binary alloys are quenched at critical temperatures. For physical, mathematical, and numerical derivations of the binary Cahn–Hilliard equation, see [21] and the references therein.

(a)   (b)   (c)

Fig. 4. Example of the image inpainting: (a) initial data (inpainting region in gray), (b) intermediate state, and (c) steady state. Image domain is 128×128 and gap distance of the inpainting region is 30 grid points. Reprinted from Bertozzi et al. [22] with permission from Society for Industrial and Applied Mathematics.

Fig. 5. Basic mechanism of the proposed algorithm: (a) mesh grid with initial data, (b) processed slice data, (c) initial guess obtained using a linear interpolation, (d) zero-filled contour of ϕ at $t = 0$, and (e) zero-filled contour at a steady state after 36 iterations.
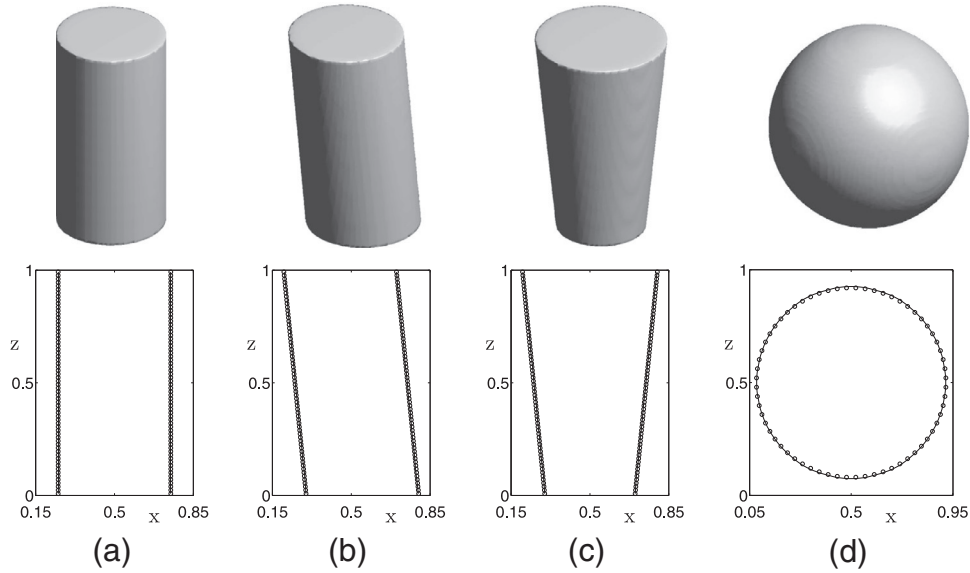
Fig. 6. Accuracy test for our proposed method: top row shows zero-isosurfaces of numerical solution and bottom row displays comparison between the numerical (solid) and reference (circle) solutions at plane $y = 0.5$.

Eqs. (4) and (5) have also been used in image inpainting problem, which is the process of filling in missing parts of damaged images based on information from the surrounding areas [19]. The basic mechanism of the inpainting model is illustrated in Fig. 4. The gray region in Fig. 4(a) denotes the inpainting region. Let $\psi(\mathbf{x})$ be the image data from Fig. 4(a). In the figure, black, gray, and white colors represent $\psi(\mathbf{x}) \approx -1, 0$, and 1, respectively. Take the initial condition as $\phi(\mathbf{x}, 0) = \psi(\mathbf{x})$ and solve Eqs. (4) and (5). Then outside the inpainting region, due to the fidelity term $\lambda(\mathbf{x})(\psi(\mathbf{x}) - \phi(\mathbf{x}, t))$, the temporal evolution of $\phi(\mathbf{x}, t)$ does not deviate much from the original image data $\psi(\mathbf{x})$. On the other hand, inside the inpainting region where $\lambda(\mathbf{x}) = 0$, the Cahn–Hilliard dynamics takes place. The gray value will evolve to either -1 or 1 depending on the boundary values of the inpainting region. Fig. 4(b) and (c) show the intermediate and steady states, respectively. More details can be found in [19].

We use a nearly identical governing equation to those used for image inpainting, which is two-dimensional. However, we use the equation in a different context, meaning we reconstruct a 3D volume from a set of slice data. Our approach involves the salient application of the partial differential equation used in image inpainting.

3. Numerical solution

For the stable temporal discretization, we use the efficient numerical scheme based on the unconditionally gradient stable scheme [23,24], which allows large time steps. We discretize the governing equations, Eqs. (4) and (5), in our 3D space, $\Omega = (0, L_x) \times (0, L_y) \times (0, L_z)$. Let $x_i = (i - 0.5)h$, $y_j = (j - 0.5)h$, $z_k = (k - 0.5)h$, $1 \leq i \leq N_x$, $1 \leq j \leq N_y$, and $1 \leq k \leq N_z$, where N_x, N_y, and N_z are positive

Table 1
Discrete l_2-norms of errors, $\|e_h\|_2$.

Case	Fig. 6(a)	Fig. 6(b)	Fig. 6(c)	Fig. 6(d)
$h = 1/64$	1.410E−2	1.460E−2	1.501E−2	2.756E−2
$h = 1/128$	5.024E−3	5.041E−3	5.341E−3	8.976E−3

even integers and $h = L_x/N_x = L_y/N_y = L_z/N_z$ is the uniform mesh size. Let $\mathbf{x}_{ijk} = (x_i, y_j, z_k)$ and let ϕ_{ijk}^n be an approximation of $\phi(\mathbf{x}_{ijk}, n\Delta t)$, where $\Delta t = T/N_t$ is the time-step, T is the final time, and N_t is the total number of time-steps. Then the discrete domain is defined by $\Omega_h = \{\mathbf{x}_{ijk} | 1 \le i \le N_x, 1 \le j \le N_y, 1 \le k \le N_z\}$. In addition, we denote the grid function as $\phi_h = \{\phi_{ijk} | \mathbf{x}_{ijk} \in \Omega_h\}$ and discrete l_2-norm of ϕ_h as

$$\|\phi_h\|_2 = \sqrt{\sum_{\mathbf{x}_{ijk} \in \Omega_h} h^3 \left(\phi_{ijk}^n\right)^2}. \tag{10}$$

We then have the following discretization.

$$\frac{\phi_{ijk}^{n+1} - \phi_{ijk}^n}{\Delta t} = \lambda_{ijk}(\psi_{ijk} - \phi_{ijk}^n) + \frac{\mu_{i+1,jk}^{n+1} + \mu_{i-1,jk}^{n+1}}{h^2}$$
$$+ \frac{\mu_{i,j+1,k}^{n+1} + \mu_{i,j-1,k}^{n+1} + \mu_{ij,k+1}^{n+1} + \mu_{ij,k-1}^{n+1} - 6\mu_{ijk}^{n+1}}{h^2}, \tag{11}$$
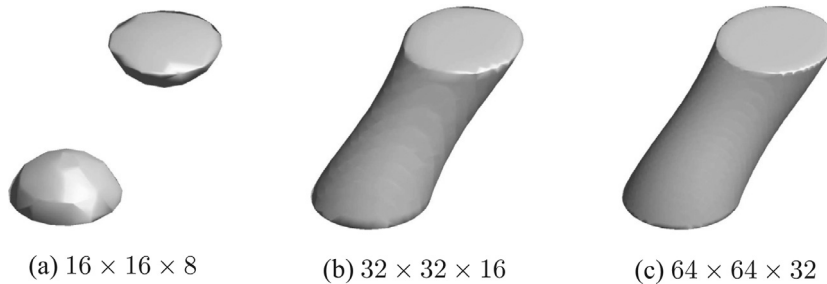
$$\mu_{ijk}^{n+1} = \left(\phi_{ijk}^{n+1}\right)^3 - \phi_{ijk}^n - \epsilon^2 \frac{\phi_{i+1,jk}^{n+1} + \phi_{i-1,jk}^{n+1}}{h^2} - \epsilon^2$$
$$\times \frac{\phi_{i,j+1,k}^{n+1} + \phi_{i,j-1,k}^{n+1} + \phi_{ij,k+1}^{n+1} + \phi_{ij,k-1}^{n+1} - 6\phi_{ijk}^{n+1}}{h^2}. \tag{12}$$

The resulting discrete equations, Eqs. (11) and (12), are solved using a multigrid method. In order to condense the discussion, we describe only the relaxation step of this method. Let $\phi_{ijk}^{n+1,m}$ and $\phi_{ijk}^{n+1,m+1}$ be the respective current and updated approximations of ϕ_{ijk}^{n+1}. We set the initial guess to be the previous time-step solution as $\phi_{ijk}^{n+1,0} = \phi_{ijk}^n$. We linearize $\left(\phi_{ijk}^{n+1}\right)^3$ as
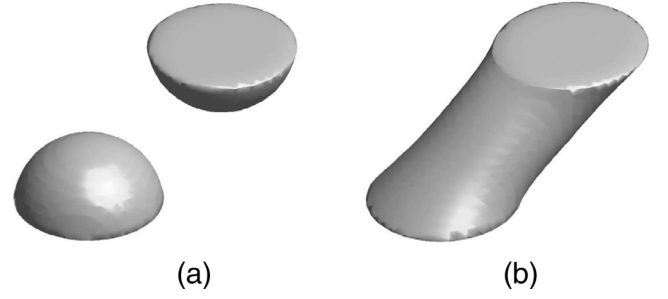
$$\left(\phi_{ijk}^{n+1}\right)^3 \approx \left(\phi_{ijk}^{n+1,m}\right)^3 + 3\left(\phi_{ijk}^{n+1,m}\right)^2 \left(\phi_{ijk}^{n+1} - \phi_{ijk}^{n+1,m}\right).$$

Then we apply the Gauss–Seidel relaxation to the multigrid method:

$$\frac{\phi_{ijk}^{n+1,m+1}}{\Delta t} + \frac{6\mu_{ijk}^{n+1,m+1}}{h^2} = \frac{\phi_{ijk}^n}{\Delta t} + \lambda_{ijk}(\psi_{ijk} - \phi_{ijk}^n)$$
$$+ \frac{\mu_{i+1,jk}^{n+1,m} + \mu_{i-1,jk}^{n+1,m+1} + \mu_{i,j+1,k}^{n+1,m} + \mu_{i,j-1,k}^{n+1,m+1} + \mu_{ij,k+1}^{n+1,m} + \mu_{ij,k-1}^{n+1,m+1}}{h^2},$$
$$-\left(\frac{6\epsilon^2}{h^2} + 3(\phi_{ijk}^{n+1,m})^2\right)\phi_{ijk}^{n+1,m+1} + \mu_{ijk}^{n+1,m+1} = -2(\phi_{ijk}^{n+1,m})^3$$
$$-\phi_{ijk}^n - \epsilon^2 \frac{\phi_{i+1,jk}^{n+1,m} + \phi_{i-1,jk}^{n+1,m+1} + \phi_{i,j+1,k}^{n+1,m} + \phi_{i,j-1,k}^{n+1,m+1} + \phi_{ij,k+1}^{n+1,m} + \phi_{ij,k-1}^{n+1,m+1}}{h^2}.$$

(a)   (b)

Fig. 8. Steady solution for different interface dimension.

For more details about the multigrid algorithm of a 3D Cahn–Hilliard equation, please refer to [23]. It should be noted that without the fitting term (Eq. (4) with $\lambda_0 = 0$), the proposed scheme is an unconditionally gradient stable scheme [25].

4. Numerical tests

In this section, we perform several numerical experiments in order to demonstrate the performance of our proposed scheme. Across the interfacial transition region, the phase-field function ϕ varies from -0.9 to 0.9 over a distance of approximately $2\sqrt{2} \in \tanh^{-1}(0.9)$ if $\lambda_0 = 0$. Therefore, if we want this distance to be approximately m grid points, $\in$ is defined as $\in_m = hm/[2\sqrt{2}\tanh^{-1}(0.9)]$ [21]. We regard the numerical result as the steady state solution if the relative error $\|\phi^{n+1} - \phi^n\|_2^2 / \|\phi^n\|_2^2$ is less than a tolerance tol. Unless otherwise specified, we use $\in = \in_4$, time step $\Delta t = 0.5h$, $\lambda_0 = 1000$, and $tol = 0.002$. Throughout the paper, we use isotropic grids and we apply the image segmentation step for the slice data unless we use the analytic hyperbolic tangent function.
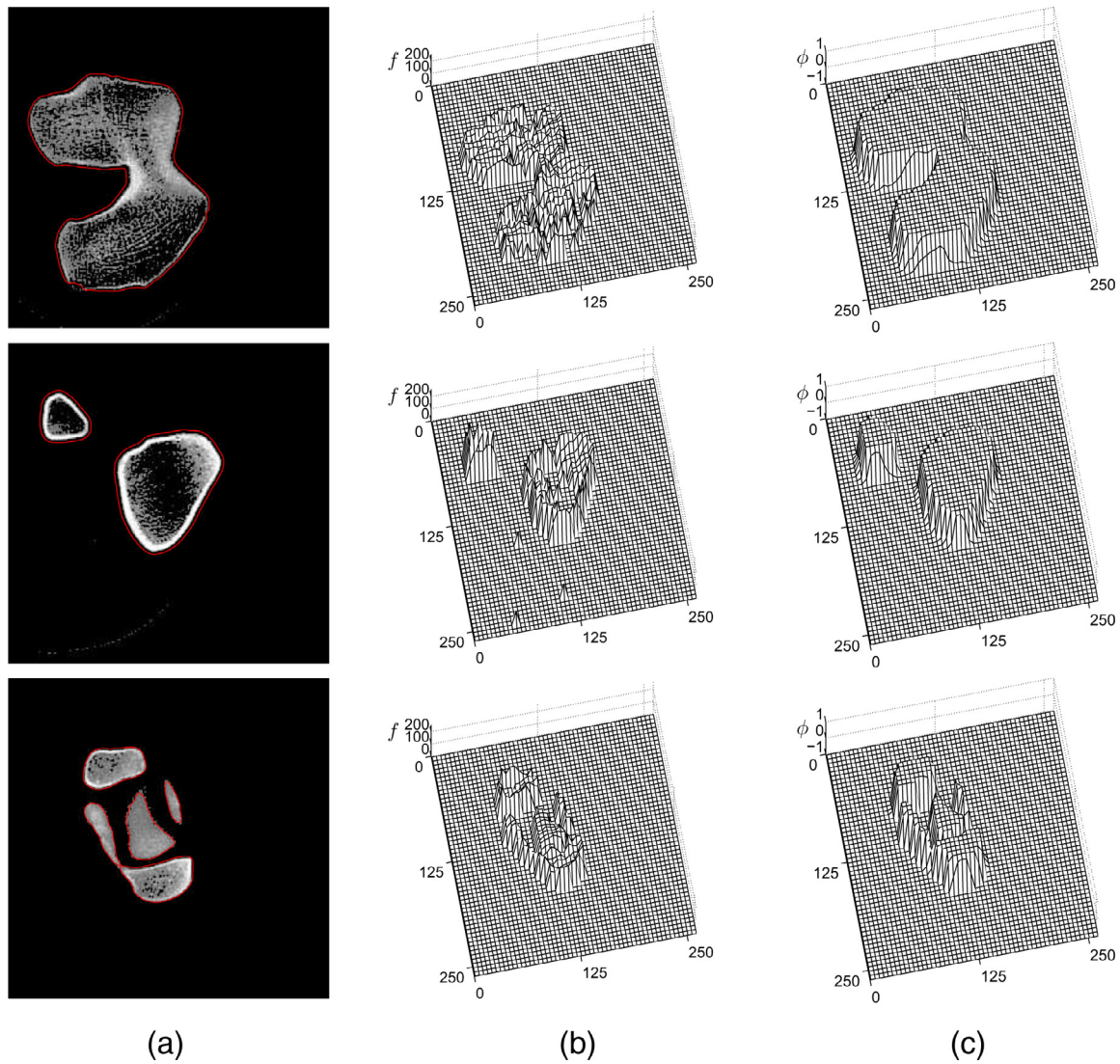
4.1. Basic mechanism of the algorithm

We start with an example in the two-dimensional (2D) space $\Omega = (0, 1) \times (0, 0.5)$ with a 64×32 mesh grid to show the algorithm's basic mechanism in Eqs. (4) and (5).

In Fig. 5(a), thick line segments are the given slices and has following hyperbolic tangent profiles, which are shown in Fig. 5(b):

$$\psi(x, y_0) = -1 + \tanh[(x - 0.15)/(\sqrt{2}\in_4)] - \tanh[(x - 0.4)/(\sqrt{2}\in_4)]$$
$$+ \tanh[(x - 0.6)/(\sqrt{2}\in_4)] - \tanh[(x - 0.9)/(\sqrt{2}\in_4)],$$
$$\psi(x, y_{11}) = -1 + \tanh[(x - 0.3)/(\sqrt{2}\in_4)] - \tanh[(x - 0.45)/(\sqrt{2}\in_4)]$$
$$+ \tanh[(x - 0.55)/(\sqrt{2}\in_4)] - \tanh[(x - 0.7)/(\sqrt{2}\in_4)],$$
$$\psi(x, y_{21}) = -1 + \tanh[(x - 0.4)/(\sqrt{2}\in_4)] - \tanh[(x - 0.65)/(\sqrt{2}\in_4)],$$
$$\psi(x, y_{33}) = -1 + \tanh[(x - 0.25)/(\sqrt{2}\in_4)] - \tanh[(x - 0.65)/(\sqrt{2}\in_4)].$$

(a) $16 \times 16 \times 8$ (b) $32 \times 32 \times 16$ (c) $64 \times 64 \times 32$

Fig. 7. Steady solutions for different pixel numbers. Here, only the bottom and top slices are used.

Fig. 9. Image segmentation process of the given slice data: (a) given medical images with contours obtained by image segmentation overlayed, (b) mesh plots of the given image, (c) mesh plots of the final result.
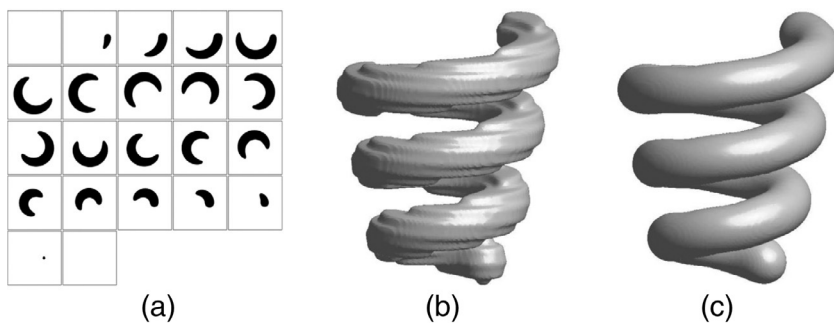
Fig. 10. (a) Synthetic slice data of a spiral (ordered left to right and top to bottom), (b) initial shape, and (c) reconstructed volume.

Because this is an example in 2D, we assume that the given slice data has the hyperbolic tangent profile. However, in the 3D case, we need an image segmentation algorithm as a preprocess before the volume reconstruction. We guess the initial state using the liner interpolation (see Fig. 5(c)), and the filled contour at level zero is shown in Fig. 5(d). Fig. 5(e) displays the reconstructed image obtained by our proposed method.

4.2. Accuracy of the proposed method

In order to test the accuracy of our proposed method, we examine four synthetic functions whose surface and volume are implicitly defined as zero-level and positive valued sets, respectively:
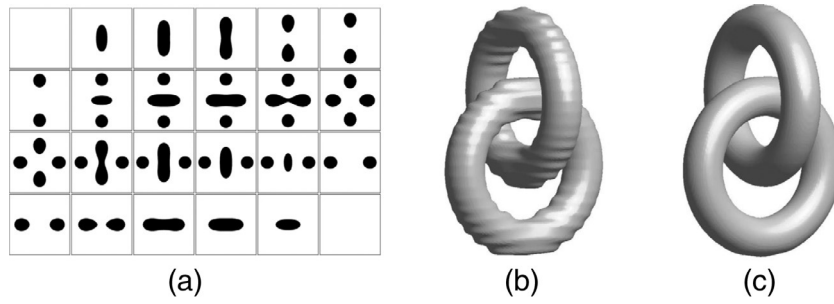
Fig. 11. (a) Synthetic slice data of two linked tori (ordered left to right and top to bottom), (b) initial shape, and (c) reconstructed volume.
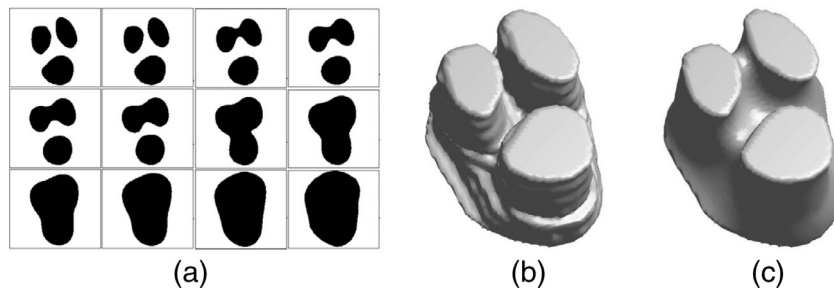

Fig. 12. (a) Synthetic slice data of three branching (ordered left to right and top to bottom), (b) initial shape, and (c) reconstructed volume.
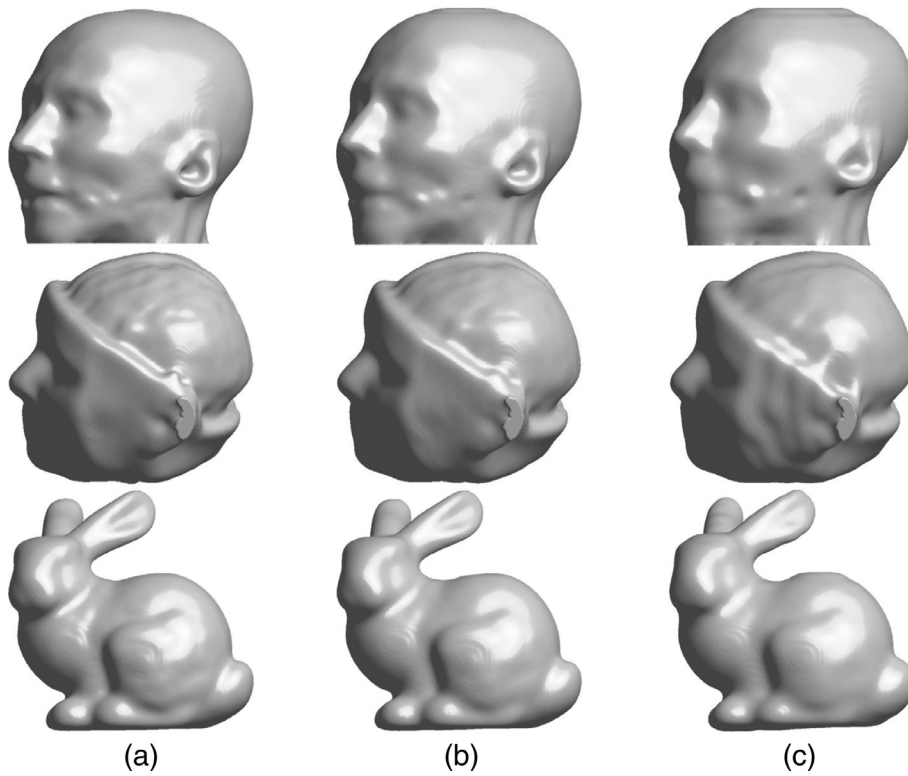

Fig. 13. Volume reconstructions for human head, brain, and terra-cotta bunny. (b) and (c) are results using half and quarter numbers of slice data from (a), respectively.

Table 2
List of original slice data information [26] and our slice data with NS and ITER refereing to the respective numbers of slices and iterations.

Case		Image pixels	NS	NS between two data	ITER
Human head	Original data,	256 × 256	96	2	
	Fig. 13(a) top,	256 × 256	96	2	12
	Fig. 13(b) top,	256 × 256	48	4	14
	Fig. 13(c) top,	256 × 256	24	8	18
Human brain	Original data,	256 × 256	108	2	
	Fig. 13(a) middle,	256 × 256	108	2	17
	Fig. 13(b) middle,	256 × 256	54	4	22
	Fig. 13(c) middle,	256 × 256	27	8	28
Bunny	Original data,	512 × 512	360	2	
	Fig. 13(a) bottom,	256 × 256	180	2	8
	Fig. 13(b) bottom,	256 × 256	90	4	11
	Fig. 13(c) bottom,	256 × 256	45	8	18

$$\psi(\mathbf{x}) = \tanh[(0.25 - \sqrt{(x - 0.5)^2 + (y - 0.5)^2})/(\sqrt{2}\epsilon_4)],$$

$$\psi(\mathbf{x}) = \tanh[(0.25 - \sqrt{(x - 0.1z - 1)^2 + (y - 0.5)^2})/(\sqrt{2}\epsilon_4)],$$

$$\psi(\mathbf{x}) = \tanh[(0.2 - \sqrt{(x - 0.5)^2 + (y - 0.5)^2} + 0.1z)/(\sqrt{2}\epsilon_4)],$$

$$\psi(\mathbf{x}) = \tanh[(0.42 - \sqrt{(x - 0.5)^2 + (y - 0.5)^2 + (z - 0.5)^2})/(\sqrt{2}\epsilon_4)].$$

Using different mesh grids $64 \times 64 \times 64$ and $128 \times 128 \times 128$, we run the evolution up to $T = 0.024$ with a time-step $\Delta t = 2h^2$. All tests are performed in the unit cube domain $\Omega = (0, 1) \times (0, 1) \times (0, 1)$. We obtain the slice data at $z = z_0, z_5, \ldots, z_{N_z+1}$ using the given synthetic function $\psi(\mathbf{x})$. We define the error of a grid as $e_h = \phi_h - \psi_h$. In Fig. 6, the top row shows the reconstructed volumes. These are compared with the respective numerical and reference solutions at $y = 0.5$ in the bottom row. The discrete l_2-norms of errors are given in Table 1. The numerical results agree with the theoretical values.

4.3. Effect of image pixel and interslice dimension

We perform simulations to show the effect of pixel dimension on 3D volume reconstruction with increasing pixel number, $16 \times 16 \times 8$, $32 \times 32 \times 16$, and $64 \times 64 \times 32$. We set numerical parameters as $\epsilon = \epsilon_4$, $\Delta t = 0.1h$, and $tol = 0.0001$ on a domain $\Omega = (0, 1) \times (0, 1) \times (0, 0.5)$. The given slice data is a set of two circles whose radii are same as $r = 0.2$ and centers are apart by $d = 0.4$. Fig. 7 shows the steady solutions with different pixel numbers. We can see that, with a few grid points, the reconstructed surface could not be connected,

however, with increasing pixel number, it is getting smooth and connected.

We consider the effect of interslice dimension between two consecutive slices. A $64 \times 64 \times 16$ mesh grid is used and numerical parameters are set as $\epsilon = \epsilon_4$, $\Delta t = 0.5h$, and $tol = 0.0001$ on a domain $\Omega = (0, 1) \times (0, 1) \times (0, 0.25)$. The given slice data is a set of two circles which radii are same as $r = 0.15$ and centers are apart by $d = 0.32$, see Fig. 8(a). The reconstructed surface is not connected because the circles are not close enough comparing with the given interslice. However, if we add a slice at $z = z_8$, which has same radius and is located in middle in two circles, then we have smoothly connected surface as shown in Fig. 8(b).

4.4. Image segmentation step

In this section, we show several numerical results of image segmentation step on real medical images of a bone in Fig. 9. From left to right columns, they are given medical images with contours obtained by the image segmentation, mesh plots of the given images, and mesh plots of the final results. As can be seen, even though the original images are with noises, the image segmentation step produces visually clear results. We will use the image segmentation afterward, unless otherwise mentioned.

4.5. Volume reconstructions using synthetic slice data

We reconstruct the 3D volume using the three set of more complicated synthetic slice data, consisting a spiral, two linked tori, and three branching. In this section, we use $tol = 0.001$. First, we

(a)   (b)
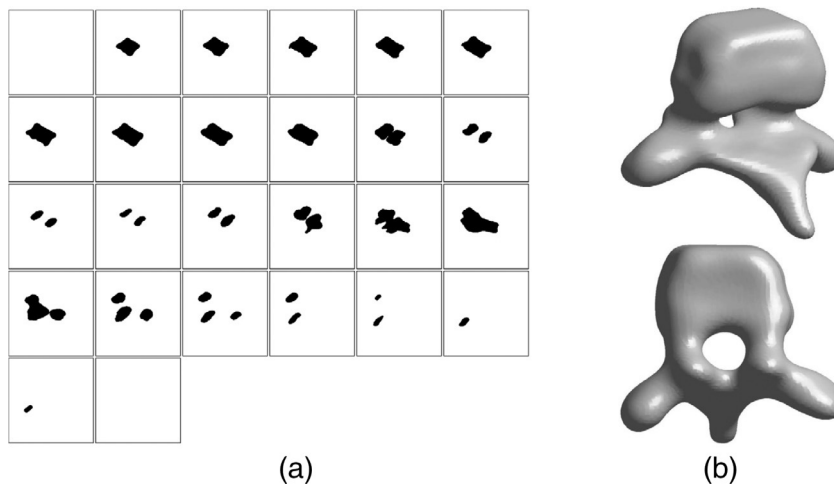
Fig. 14. Volume reconstruction of human vertebra: (a) slice data (ordered left to right and top to bottom) and (b) reconstructed volume of the human vertebra from different angles.
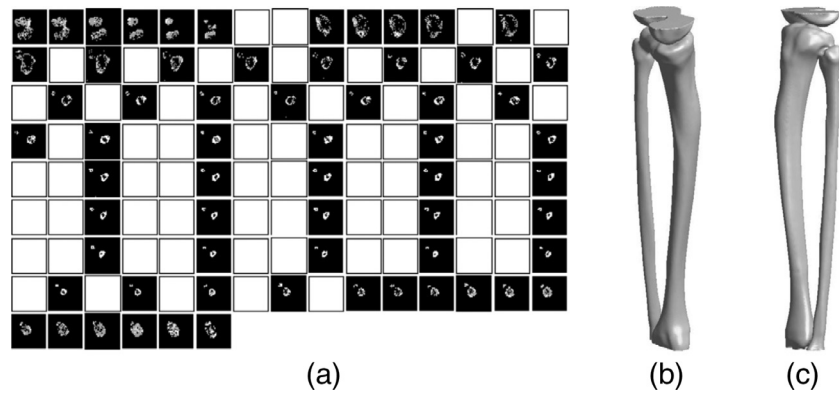
Fig. 15. Volume reconstruction from medical images of a human bone (tibia and fibula): (a) slice data (ordered left to right and top to bottom), in which empty boxes represent skipped data, and (b) and (c) reconstructed volumes obtained from two different angles.

consider 3D spiral with twenty-seven slice data (Fig. 10(a)). We have five slices between any two consecutive slice data sets except for one slice between the last two slice data sets. Fig. 10(b) and (c) exhibit the initial shape and final result obtained after 18 iterations. A $128 \times 128 \times 128$ mesh grid is used for the unit cube domain $\Omega = (0,1) \times (0,1) \times (0,1)$. Second, we consider two 3D linked tori with twenty-four slice data (Fig. 11(a)). We have seven slices between any two consecutive slice data sets except for four slices between the last two given slices. A $128 \times 128 \times 160$ mesh grid is used for the domain $\Omega = (0,1) \times (0,1) \times (0,1.25)$. Fig. 11(b) and (c) display the initial shape and final result obtained after 17 iterations.

Third, we employ a similar test to that used in [4], where a more complex case of branching from one to three contours was tested using synthetic contours. Twelve slice data are used (Fig. 12(a)) and there are two slices between any two consecutive slice data sets. This simulation is performed on the domain $(0,1) \times (0,1.125) \times (0,0.531)$ with a $64 \times 72 \times 34$ mesh grid. Fig. 12(b) and (c) are the initial shape by linear interpolation and final result obtained after 44 iterations.

4.6. Volume reconstructions using real slice data

In this section, we simulate several volume reconstructions by using real slice data. Fig. 13 shows the reconstructed volumes such as a human head, brain, and terra-cotta bunny, whose slice data are obtained from [26]. Each column in Fig. 13 shows the result of the volume reconstruction using different interslice dimensions. The detail information of the data we used is displayed in Table 2. This simulation is performed on the domain $(0,1) \times (0,1) \times (0,1)$ with a $256 \times 256 \times 256$ mesh grid. We can see that our proposed method successfully manages this complex topology and the reconstructed surface is very smooth. Furthermore, our method is working well with small set of slice data. And, Table 2 shows that the iteration number for convergence slightly increase when the number of slice data decrease.

Next, we consider a human vertebra with a twenty-six slice data from [27] (Fig. 14(a)). There are two slices between any two given successive slice data sets, except for the first and last sets, between which we use only one slice. This simulation is performed on the domain $(0,1) \times (0,1) \times (0,0.45)$ with a $160 \times 160 \times 72$ mesh grid. Fig. 14(b) and (c) represent the initial state by the linear interpolation and final result obtained after 14 iterations. As can be seen, our proposed method produces visually clear results, even though the contours are not convex and multiple links between contours are exist.

Our last test examines the volume reconstruction from medical images of a human bone (tibia and fibula). We have a set of bone slice images with an image size of 216×216 (Fig. 15(a)). The number of slices is 128 and there are seven slices between any two consecutive slices. Note that, because our method can reconstruct the volume with fewer slice data, we remove some similar slices (the empty boxes in Fig. 15(a)) and use 63 slices. We perform the resolution on the domain $(0,1) \times (0,1) \times (0,4.120)$ with a $216 \times 216 \times 890$ mesh grid. Fig. 15(b) and (c) show results of volume reconstruction with different angles after only 23 iterations. As can be seen, our algorithm represents the bone image well and produces good visual quality.

5. Conclusion

We presented a fast, robust, and accurate numerical method for creating a mathematical model that produces 3D volume reconstruction using slice data. The governing equations are based on the Allen–Cahn and Cahn–Hilliard equations with a fidelity term. This term keeps the solutions close to the given slice data. The proposed algorithm has two steps: image segmentation for the raw given slice data and 3D volume reconstruction using the segmented images. We applied a hybrid method and an unconditionally stable nonlinear splitting scheme, and then we solved the resulting system of discrete equations using a multigrid method. We demonstrated the accuracy, efficiency, and robustness of the method on both synthetic and real medical images, such as a spiral, two linked tori, three branching, head, brain, bunny, vertebra, tibia, and fibula. The reconstructed volumes had smooth surfaces, fitting the given original slice data. As a future work, it would be of interest to investigate automatic criterions about how many interslices are needed and what is an optimal tolerance for the iterations.

References

[1] U. Trottenberg, C. Oosterlee, A. Schüller, Multigrid, Academic Press, USA, 2001.

[2] J. Cheng, Y. Liu, 3-D reconstruction of medical image using wavelet transform and snake model, J. Multimed. 4 (2009) 427–434.

[3] J. Dornheim, D.J. Lehmann, L. Dornheim, B. Preim, G. Strauß, Reconstruction of blood vessels from neck CT datasets using stable 3D mass-spring models, VCBM (2008) 77–82.

[4] W.C. Lin, S.Y. Chen, C.T. Chen, A new surface interpolation technique for reconstructing 3D objects from serial cross-sections, Comput. Vis. Graph. Image Process. 48 (1989) 124–143.

[5] J.F. Guo, Y.L. Cai, Y.P. Wang, Morphology-based interpolation for 3D medical image reconstruction, Comput. Med. Imaging Graph. 19 (1995) 267–279.

[6] M.W. Jones, M. Chen, A new approach to the construction of surfaces from contour data, Comput. Graph. Forum 13 (1994) 75–84 (Blackwell Science Ltd.).

[7] S.P. Raya, J.K. Udupa, Shape-based interpolation of multidimensional objects, IEEE Trans. Med. Imaging. 9 (1990) 32–42.

[8] D.-J. Yoo, Three-dimensional surface reconstruction of human bone using a B-spline based interpolation approach, Comput.-Aided Des. 43 (8) (2011) 934–947.

[9] A. Gutierrez, M.J. Jimenez, D. Monaghan, N.E. ÓConnor, Topological evaluation of volume reconstructions by voxel carving, Comput. Vis. Image Underst. 121 (2014) 27–35.

[10] M. Lhuillier, S. Yu, Manifold surface reconstruction of an environment from sparse structure-from-motion data, Comput. Vis. Image Underst. 117 (2013) 1628–1644.

[11] S.F. Gibson, Constrained elastic surface nets: generating smooth surfaces from binary segmented data, Med. Image Comput. Comput. Assist. Interv., Springer, Berlin, Heidelberg, 1998, pp. 888–898.

[12] V. Caselles, F. Catte, T. Coll, F. Dibos, A geometric model for active contours in image processing, Numer. Math. 66 (1993) 1–31.

[13] T.F. Chan, L.A. Vese, Active contours without edges, IEEE Trans. Image Process. 10 (2001) 266–277.

[14] Y. Li, J. Kim, A fast and accurate numerical method for medical image segmentation, J. KSIAM 14 (2010) 201–210.

[15] Y. Li, J. Kim, Multiphase image segmentation using a phase-field model, Comput. Math. Appl. 62 (2011) 737–745.

[16] Y. Li, J. Kim, An unconditionally stable numerical method for bimodal image segmentation, Appl. Math. Comput. 219 (2012) 3083–3090.

[17] Y. Li, J. Kim, An unconditionally stable hybrid method for images segmentation, Appl. Numer. Math. 82 (2014) 32–43.

[18] G. Paul, J. Cardinale, I.F. Sbalzarini, Coupling image restoration and segmentation: a generalized linear model/Bregman perspective, Int. J. Comput. Vis. 104 (2013) 69–93.

[19] A.L. Bertozzi, S. Esedoglu, A. Gillette, Inpainting of binary images using the Cahn–Hilliard equation, IEEE Trans. Image Process. 16 (2007) 285–291.

[20] J.W. Cahn, J.E. Hilliard, Free energy of a nonuniform system. I. Interfacial free energy, J. Chem. Phys. 28 (1958) 258–267.

[21] D. Lee, J.Y. Huh, D. Jeong, J. Shin, A. Yun, J. Kim, Physical, mathematical, and numerical derivations of the Cahn–Hilliard equation, Comput. Mater. Sci. 81 (2014) 216–225.

[22] A.L. Bertozzi, S. Esedoglu, A. Gillette, Analysis of a two-scale Cahn–Hilliard model for image inpainting, Multiscale Model. Simul. 6 (2007) 913–936.

[23] S.D. Yang, H.G. Lee, J. Kim, A phase-field approach for minimizing the area of triply periodic surfaces with volume constraint, Comput. Phys. Commun. 181 (2010) 1037–1046.

[24] Y. Li, D. Jeong, J. Shin, J. Kim, A conservative numerical method for the Cahn–Hilliard equation with Dirichlet boundary conditions in complex domains, Comput. Math. Appl. 65 (2013) 102–115.

[25] D.J. Eyre, Computational and mathematical models of microstructural evolution, Mater. Res. Soc. PA (1998) 39–46.

[26] The Stanford Volume Data Archive, Copyright ©2000Marc Levoy, 2001. <http://www-graphics.stanford.edu/data/voldata/>.

[27] A.B. Ekoule, F.C. Peyrin, C.L. Odet, A triangulation algorithm from arbitrary shaped multiple planar contours, ACM Trans. Graph. 10 (1991) 182–199.