Pattern Recognition 132 (2022) 108914

Contents lists available at ScienceDirect

Pattern Recognition

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

Weighted 3D volume reconstruction from series of slice data using a modified Allen–Cahn equation

Yibao Li[a], Xin Song[a], Soobin Kwak[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 21 April 2022
Revised 30 May 2022
Accepted 18 July 2022
Available online 21 July 2022

Keywords:
Shape transformation
3D volume reconstruction
Allen–Cahn equation

ABSTRACT

In this study, we develop a fast and accurate computational method for a weighted three-dimensional (3D) volume reconstruction from a series of slice data using a phase-field model. The proposed method is based on a modified Allen–Cahn (AC) equation with a fidelity term. The algorithm automatically generates the necessary slices between the given slices by solving the governing equation. To reconstruct a 3D volume, we first set a source slice and target slice. Next, we set the source slice as the initial condition and the target slice as the fidelity function. Finally, we retain the numerical solutions during an evolution as intermediate slices between the source and target slices. There are two criteria for choosing the intermediate slice: One is based on the area of the symmetric difference between the phase-field solution and the target and the other is based on the change of the phase-field solution relative to the area of the target. We use the weighted average of the two criteria. To validate the efficiency and accuracy of the proposed numerical algorithm, several computational experiments are conducted. Computational test results confirm the superior performance of the proposed algorithm.

1. Introduction

Reconstructing a three-dimensional (3D) volume from two-dimensional (2D) serial cross-sections has received significant attention because of its various fields of application, such as radiation treatment planning [1], medical image processing [2], surgery planning [3,4], and medical image inpainting [5]. Kim and Lee [6] developed a partial differential equation (PDE) for a 3D volume reconstruction using 2D slice data. The proposed model is based on the modified Cahn–Hilliard (CH) equation, allowing it to reconstruct a smooth surface and satisfy the slice constraints. Bretin et al. [7] presented a geometric variational approach for the reconstruction of a volume from slices that exactly or approximately fit the given slices.

Related approaches can be divided into two categories: topological- and variational-based methods. The first category is based on topological assumptions and uses either a parametric or implicit function to interpolate an accurate surface [8,9]. Ju et al. [10] reconstructed 3D surfaces with curve networks by considering the intersections of the plan projections. Liu et al. [11] extended the projection-based approach from parallel plans to an arbitrary convex space. Instead of individually reconstructing the surface within the space partitioned by the given slices, Zou et al. [12] proposed a surface reconstruction algorithm that allows global topology control. Huang et al. [13] defined interface sets and considered topology control in the context of multi-labeled domains.

The second category is from a variational viewpoint, in which the surface is obtained by solving an optimization problem [14,15]. In [16], Bertozzi et al. proposed a reconstruction algorithm for binary images based on the CH equation. Li et al. [17] utilized a fidelity term to reconstruct the surface and maintain the solution close to the given slices. Li et al. [18] proposed a new fidelity term to satisfy the constraints of a multi-component system. Using the CH equation, these methods can provide an accurate volume reconstruction. However, the CH equation incurs a large computational cost owing to the fourth-order operator term. In [19], the authors proposed a modified Allen–Cahn equation (AC) with mean curvature motion to recover a volume with small holes or unevenness owing to lost voxels.

In this study, we will utilize the second-order AC equation as the governing equation to obtain an efficient volume reconstruction. However, the original AC equation-based 3D volume reconstruction algorithm [20] can produce unacceptable results because it does not consider the fact that the slice area changes nonlinearly and quadratically in certain cases. To resolve this problem, a

* Corresponding author.
E-mail addresses: yibaoli@xjtu.edu.cn (Y. Li), holyxin@stu.xjtu.edu.cn (X. Song), oobin23@korea.ac.kr (S. Kwak), cfdkim@korea.ac.kr (J. Kim).
URL: https://mathematicians.korea.ac.kr/cfdkim/ (J. Kim)

https://doi.org/10.1016/j.patcog.2022.108914
0031-3203/© 2022 Elsevier Ltd. All rights reserved.

selection algorithm was proposed to select suitable data for intermediate volumes. The main contribution of this study is the utilization of the second-order equation and a weighted average with quadratic change of area for constructing a surface with low computational cost compared to the fourth-order methods proposed in Li et al. [17].

This paper is organized as follows. We present the governing equation and selection algorithm for reconstructing the 3D volume from two given slices in Section 2. Section 3 presents a numerical scheme for the governing equation. In Section 4, various computational experiments are presented to demonstrate the efficiency and robustness of the proposed algorithm. Finally, some concluding remarks are presented in Section 5.

2. 3D volume reconstruction process

The proposed reconstruction algorithm consists of two parts: the calculation of alternative solutions and the selection of intermediate data. When a series of slices is given, the proposed algorithm first calculates the alternative solutions for every two consecutive slices using the governing equation. These alternative solutions are then used to reconstruct the final volume. As the basic idea of the proposed algorithm, given two contiguous slices, the volume between them can be seen as a continuous transformation from one slice to the other. We first use the governing equation to calculate alternative solutions for the given slices. To ensure that the beginning and ending shapes are the same as the given slices, a fidelity term is added as follows [21]:

$$\frac{\partial \phi(\mathbf{x},t)}{\partial t} = -\frac{F'(\phi(\mathbf{x},t))}{\epsilon^2} + \Delta\phi(\mathbf{x},t) + \alpha\sqrt{F(\phi(\mathbf{x},t))}(\psi(\mathbf{x}) - \phi(\mathbf{x},t)), \tag{1}$$

where $t > 0$ and $\mathbf{x} \in \Omega$ is the point on the given slice. Here, $\phi(\mathbf{x},t)$ and $\psi(\mathbf{x})$ are phase-field functions representing the source and target shapes, respectively. In addition, $F(\phi) = 0.25(\phi^2 - 1)^2$, ϵ is an interfacial parameter, and α is a parameter controlling the fidelity. If $\alpha = 0$, the equation becomes the classical AC equation, which models motion by mean curvature:

$$\frac{\partial \phi(\mathbf{x},t)}{\partial t} = -\frac{F'(\phi(\mathbf{x},t))}{\epsilon^2} + \Delta\phi(\mathbf{x},t). \tag{2}$$

We use another property of the AC equation such as preservation of a smooth interface transition layer. While preserving the smooth interface transition layer, we evolve the phase-field function depending on the fidelity term $\alpha\sqrt{F(\phi(\mathbf{x},t))}\ (\psi(\mathbf{x}) - \phi(\mathbf{x},t))$ from $\phi(\mathbf{x},0)$ to $\psi(\mathbf{x})$.

The proposed selection algorithm for constructing a volume from two slice data and their alternative solutions is as follows: Let S_1 and S_2 be the given source and target slice data, respectively. We introduce a time-dependent domain $\Omega_1(t)$ with $\Omega_1(0) = S_1$. Using $\phi(\mathbf{x},t)$ and $\psi(\mathbf{x})$, we define $\Omega_1(t) = \{\mathbf{x}|\phi(\mathbf{x},t) \geq 0\}$ and $S_2 = \{\mathbf{x}|\psi(\mathbf{x}) \geq 0\}$. Let us introduce the following two time-dependent indicator functions:

$$\mathcal{A}(t) = Area((\Omega_1(t) \cup S_2) \setminus (\Omega_1(t) \cap S_2)) \tag{3}$$

$$= |(\Omega_1(t) \cup S_2) \setminus (\Omega_1(t) \cap S_2)| = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\left|\frac{\psi_{ij} - \phi_{ij}^n}{2}\right| h^2,$$

$$\mathcal{B}(t) = Area(\Omega_1(t)) = |\Omega_1(t)| = \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\left|\frac{1 + \phi_{ij}^n}{2}\right| h^2, \tag{4}$$

where $\mathcal{A}(t)$ and $\mathcal{B}(t)$ are the discrete areas of sets $(\Omega_1(t) \cup S_2) \setminus (\Omega_1(t) \cap S_2)$ and $\Omega_1(t)$ at time $t = n\Delta t$. Here, $\mathcal{A}(t)$ is the area of symmetric difference between the two sets, $\Omega_1(t)$ and S_2; and $\mathcal{A}(t)$ monotonically decreases over time. However, $\mathcal{B}(t)$ can increase or decrease over time. Fig. 1(a)–(d) show S_1, S_2, $\Omega_1(t)$, and $\mathcal{A}(t)$, respectively. Fig. 1(e) and (f) display the stacking of intermediate shapes between the source and target shapes; and the isosurface of the reconstructed volume at level zero, respectively.

During the temporal evolution of S_1 into S_2, we retain the intermediate slice data $\Omega_1(a_k)$ at time $t = a_k$ based on $\mathcal{A}(t)$ and intermediate data $\Omega_1(b_k)$ at time $t = b_k$ based on $\mathcal{B}(t)$ for $k = 1, \ldots, N_z$. Here, N_z denotes the number of slices between S_1 and S_2. $k = 1, \ldots, N_z$, let a_k be the minimum integer that satisfies Eq. (5):

$$\mathcal{A}(a_k\Delta t) \leq \frac{N_z + 1 - k}{N_z + 1}\mathcal{A}(0), \tag{5}$$

where $\mathcal{A}(0)$ is the initial area of the symmetric difference between the source S_1 and target S_2 data, that is, $\mathcal{A}(0) = (S_1 \cup S_2) \setminus (S_1 \cap S_2)$. Eq. (5) indicates that we save the indices that approximately satisfy

$$\mathcal{A}(a_1\Delta t) \approx \frac{N_z\mathcal{A}(0)}{N_z + 1},\ \mathcal{A}(a_2\Delta t) \approx \frac{(N_z - 1)\mathcal{A}(0)}{N_z + 1},\ \ldots,\ \mathcal{A}(a_{N_z}\Delta t) \approx \frac{\mathcal{A}(0)}{N_z + 1},$$

which also implies that we keep the indices whenever there is an approximately $\mathcal{A}(0)/(N_z + 1)$ decrement in $\mathcal{A}(t)$, and we assume that the symmetric difference $\mathcal{A}(t)$ changes linearly with respect to the height.

Next, we describe how to define the value of b_k based on $\mathcal{B}(t)$ and why it is needed. We consider arbitrary source and target shapes, as shown in Fig. 2(a). In general, the source and target areas differ. Fig. 2(b) shows the disks of the same areas corresponding to the arbitrary shapes in Fig. 2(a). If we continuously stack the slices, we then have a truncated cone (TC), as shown in Fig. 2(b). We define

$$\mathcal{B}_{\text{TC}}(z) = \pi\left(\frac{R_2 - R_1}{(N_z + 1)h}z + R_1\right)^2 \tag{6}$$

in terms of h, height coordinate z, radius of source R_1, and radius of target R_2. Here, h is the distance between two consecutive slices. As we can see from Eq. (6), the change in area with respect to height is nonlinear and is a quadratic function of z. In general, because $\mathcal{B}(t)$ increases or decreases depending on the sign of $R_1 - R_2$, we define b_k as the minimum integer that satisfies the following condition:

$$\text{sig}(R_1 - R_2)\mathcal{B}(b_k\Delta t) \leq \text{sig}(R_1 - R_2)\mathcal{B}_{\text{TC}}(kh),\ \text{ for } k = 1, \ldots, N_z, \tag{7}$$

where $\text{sig}(R)$ is the sign function of a real number R, which is -1 if $R < 0$, 1 else if $R > 0$, and 0 otherwise. Eq. (7) indicates that we save the indices which approximately satisfy

$$\mathcal{B}(b_1\Delta t) \approx \mathcal{B}_{\text{TC}}(h),\ \mathcal{B}(b_2\Delta t) \approx \mathcal{B}_{\text{TC}}(2h),\ \ldots,\ \mathcal{B}(b_{N_z}\Delta t) \approx \mathcal{B}_{\text{TC}}(N_z h).$$

We now describe the proposed algorithm. Using two conditions (5) and (7), we generate two index sets, $\mathbf{a} = \{a_1, a_2, \ldots, a_{N_z}\}$ and $\mathbf{b} = \{b_1, b_2, \ldots, b_{N_z}\}$. Let Ψ_{ijk} be a 3D reconstructed volume data on $\Omega_V = \{(x_i, y_j, z_k) :\ x_i = a + hi,\ y_j = c + hj,\ z_k = hk,\ 0 \leq i \leq N_x,\ 0 \leq j \leq N_y,\ 0 \leq k \leq N_z + 1\}$. For a given weighting parameter $0 \leq \theta \leq 1$, we define the 3D volume as

$$\Psi_{ijk} = \begin{cases} \phi_{ij}^0 & \text{if } k = 0, \\ \phi_{ij}^{n_k} & \text{if } k = 1, \ldots, N_z, \\ \psi_{ij} & \text{if } k = N_z + 1, \end{cases} \tag{8}$$

where $n_k = \text{round}((1 - \theta)a_k + \theta b_k)$. Here, $\text{round}(x)$ is the round function which rounds the input value x to the nearest integer. Fig. 3(a) and (b) show schematic illustrations of these processes; and the temporal evolution of $\mathcal{A}(t)$ and $\mathcal{B}(t)$ along with the corresponding morphologies, respectively.

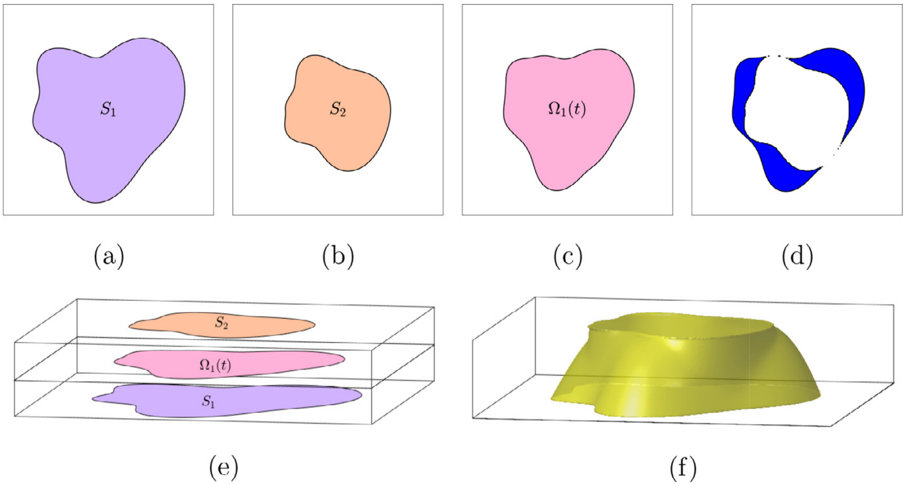

Fig. 1. Schematic representation of (a) source shape $\Omega_1(0) = S_1$, (b) target shape S_2, (c) intermediate shape $\Omega_1(t)$, (d) $(\Omega_1(t) \cup S_2) \setminus (\Omega_1(t) \cap S_2)$ at time t, (e) stacking intermediate shapes between source and target shapes, and (f) isosurface of the reconstructed volume at level zero.

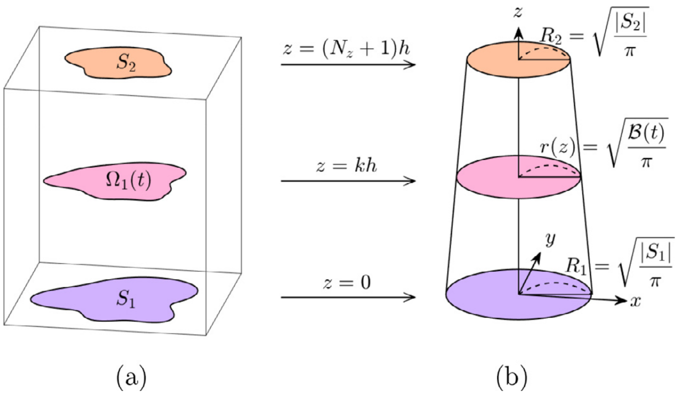
S_2
$\Omega_1(t)$
S_1
$z = (N_z + 1)h$
$z = kh$
$z = 0$
z
$R_2 = \sqrt{\frac{|S_2|}{\pi}}$
$r(z) = \sqrt{\frac{\mathcal{B}(t)}{\pi}}$
$R_1 = \sqrt{\frac{|S_1|}{\pi}}$
y
x
(a)
(b)

Fig. 2. Disk with an area equal to the slice area: (a) arbitrary shapes and (b) corresponding disk shapes with equivalent areas.

If $\theta = 0$, the volume is then reconstructed based only on $\mathcal{A}(t)$; if $\theta = 1$, then the volume is reconstructed based only on $\mathcal{B}(t)$. Note that if the areas of the source and target are the same, we cannot use $\theta = 1$ because $\mathcal{B}(t)$ is almost constant, and it is difficult to extract slices between the source and target shapes. Therefore, in this case, it is preferable to use $\theta = 0$.

3. Numerical solution

In this section, we introduce an efficient numerical scheme for the proposed algorithm. Let $\Omega = (L_x,\ R_x) \times (L_y,\ R_y)$ be a two-dimensional domain and $\Omega_h = \{(x_i, y_j) :\ x_i = L_x + hi,\ \ y_j = L_y + hj,\ 0 \le i \le N_x,\ 0 \le j \le N_y\}$ be the discrete computational domain, where h is the space step. Let $\phi_{ij}^n = \phi(x_i, y_j, n\Delta t)$, where Δt denotes the temporal step. Using the operator splitting method, we split Eq. (1) into the heat, nonlinear, and fidelity terms as follows:

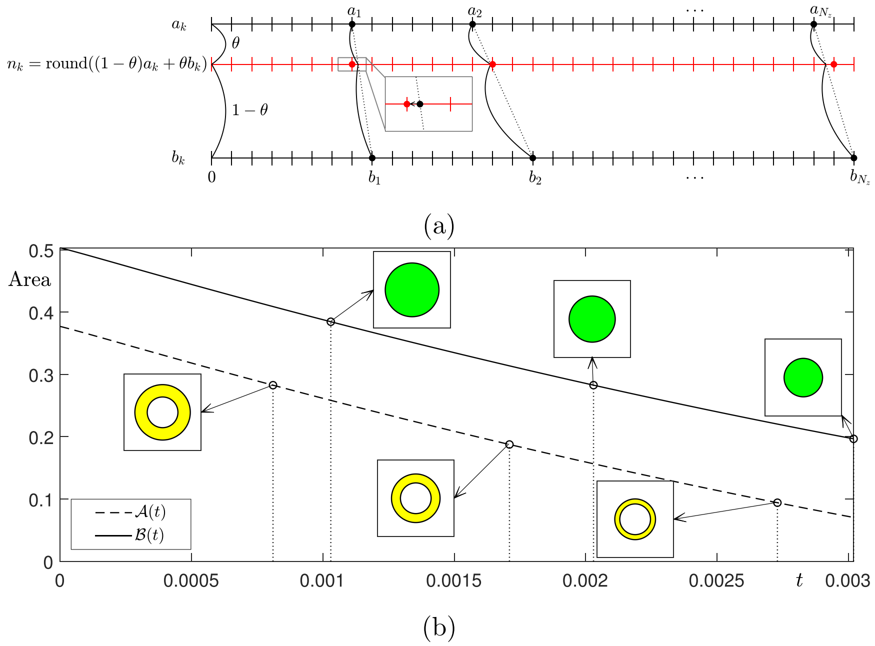

Fig. 3. (a) Schematic diagram of determining discrete time used to save intermediate data by applying θ. (b) Schematic illustration of the temporal evolution of $\mathcal{A}(t)$ and $\mathcal{B}(t)$ with corresponding morphologies.

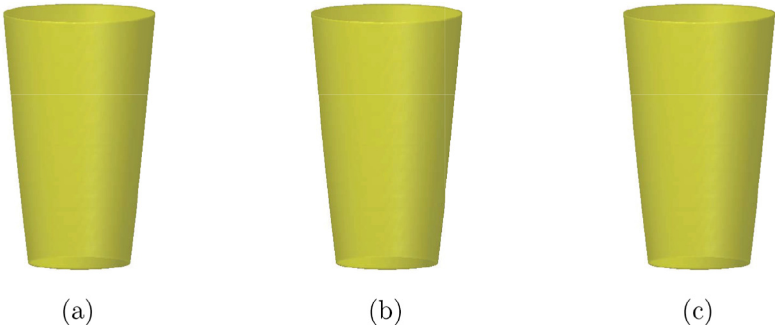

Fig. 4. Volume reconstruction from synthetic images: (a) $\theta = 0$, (b) $\theta = 0.5$, and (c) $\theta = 1$.

$$\frac{\partial \phi(\mathbf{x},t)}{\partial t} = \Delta \phi(\mathbf{x},t), \tag{9}$$

$$\frac{\partial \phi(\mathbf{x},t)}{\partial t} = -\frac{F'(\phi(\mathbf{x},t))}{\epsilon^2}, \tag{10}$$

$$\frac{\partial \phi(\mathbf{x},t)}{\partial t} = \alpha\sqrt{F(\phi(\mathbf{x},t))}(\psi(\mathbf{x}) - \phi(\mathbf{x},t)). \tag{11}$$

First, we compute the heat Eq. (9) using the finite difference method (FDM):

$$\frac{\phi_{ij}^* - \phi_{ij}^n}{\Delta t} = \Delta_d \phi_{ij}^n, \tag{12}$$

where $\Delta_d \phi_{ij}^n = (\phi_{i+1,j}^n + \phi_{i-1,j}^n + \phi_{i,j+1}^n + \phi_{i,j-1}^n - 4\phi_{ij}^n)/h^2$ under the Dirichlet boundary condition. Second, using the separation of variables [22], the analytical solution of nonlinear Eq. (10) is

$$\phi_{ij}^{**} = \phi_{ij}^* \Big/ \sqrt{\left(1 - \left(\phi_{ij}^*\right)^2\right)e^{-\frac{2\Delta t}{\epsilon^2}} + \left(\phi_{ij}^*\right)^2}. \tag{13}$$

Third, the numerical solution of the fidelity term (11) is an ordinary differential equation (ODE), which can be obtained as follows:

$$\phi_{ij}^{n+1} = \psi_{ij} + (\phi_{ij}^{**} - \psi_{ij})e^{-\alpha\Delta t\sqrt{F(\phi_{ij}^{**})}}. \tag{14}$$

The proposed method consists of two closed-form solutions for the ODEs and one numerical approximation for the explicit heat equation. The proposed computational method for Eqs. (12)–(14) is simple to implement and can achieve fast convergence. Let N be the mesh grid size. The computational complexity of the ODEs is $O(N)$. For the heat equation, we use an explicit FDM with a computational complexity of $O(N)$. In summary, the complexity of the entire processing is $O(N)$. Furthermore, we can apply a fast discrete cosine transform and implicit methods to Eq. (9) with a computational complexity of $O(N \log N)$. In addition, the proposed computational method can be straightforwardly applied to a GPU-accelerated discrete cosine transform implementation that performs up to several times faster than CPU-only alternatives. In this study, we propose a method (Eqs. (12)–(14)) which is simpler to implement.

4. Numerical experiments

In this study, we focus on 3D reconstruction using the given slice data. For preparation of the 2D slice images, we refer the reader to [23,25] for image segmentation techniques such as the level-set method [24].

4.1. Basic mechanism and accuracy of the algorithm

In this section, we demonstrate the results of the proposed algorithm with different θ values. The proposed algorithm is illustrated through the following synthetic function, the volume and surface of which are defined as positive-valued and zero-level sets, respectively:

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

where $\xi = 4h/[2\sqrt{2}\tanh^{-1}(0.9)]$. All tests are performed on $\Omega = (0,1)\times(0,1)\times(0,42/128)$ with a $128\times128\times42$ mesh grid. The other numerical parameters are $\Delta t = 0.1h^2$, $\epsilon = h$, $\alpha = 3000$, and $N_z = 40$. The results are presented in Fig. 4. We also measure our algorithm by using the error $e_h = \|(\phi_h - \psi_h)/\psi_h\|_2$, where ψ_h is the theoretical values obtained from the function. The errors are 4.7×10^{-4}, 1.6×10^{-3} and 1.8×10^{-3} for $\theta = 0,\ 0.5$, and 1, respectively. These numerical results are in good agreement with the theoretical values.

4.2. Volume reconstruction from multi-slice data

In this section, we describe the use of multi-slice data to examine whether the proposed algorithm reconstructs the volume well. We arbitrarily generate multi-slice data (S_1, S_2, S_3, and S_4) as shown in Fig. 5(a)–(d) in $\Omega = (0,1)\times(0,1)$. The parameters used are $\theta = 0.5$, $N_x = N_y = 100$, $N_z = 20$, $\Delta t = 0.1h^2$, $\epsilon = h$, and $\alpha = 3000$. Fig. 5(e) shows the reconstructed volume using S_1, S_2, S_3, and S_4, demonstrating that the proposed algorithm achieves a volume reconstruction well from multiple slices.

To further assess the efficiency of our method, we reconstruct a spherical shell surface with the same initial conditions as in Kim and Lee [6]. The experiments are performed on the domain $\Omega = (0,\ 65)\times(0,\ 65)\times(0,\ 65)$. The inner and outer radii of the spherical shell are 12 and 24, respectively. We choose 17 slices and use mesh grids consisting of $N = 33\times33\times33$, $65\times65\times65$, $129\times129\times129$, and $257\times257\times257$ with $N_z = 1,\ 3,\ 7$, and 15, respectively. In Fig. 6(a)–(c), we show the surface reconstructions with different mesh grid $65\times65\times65$, $129\times129\times129$, and $257\times257\times257$, respectively. The numerical parameters $\Delta t = 0.1h^2$, $\epsilon = h$, $\theta = 0$, and $\alpha = 3000$ are chosen. The computational results indicate that the proposed algorithm can successfully reconstruct smooth surfaces.

In Fig. 7, we plot the total CPU time versus mesh grid N with a linear fitting. The results indicate that the convergence rate of the computational cost is linear with respect to N as expected from the discretization. Furthermore, the proposed algorithm is faster than Kim and Lee's method [6].

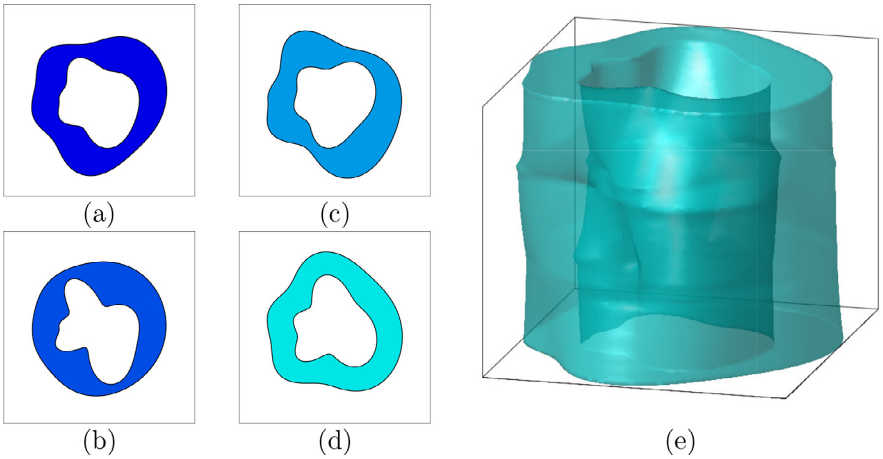

Fig. 5. (a)–(d) are multi slice data: S_1, S_2, S_3, and S_4, respectively. (e) is the reconstructed volume with S_1, S_2, S_3, and S_4.

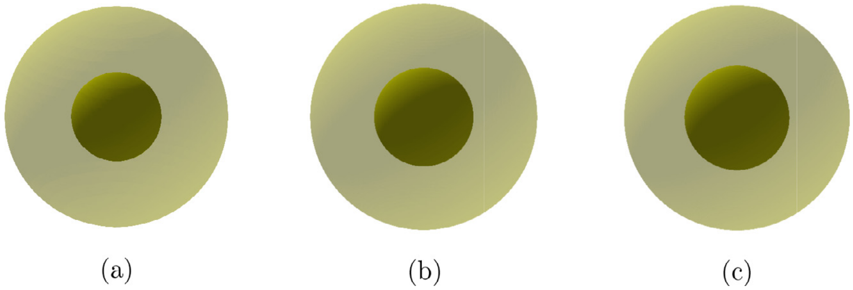

Fig. 6. Volume reconstruction with different sized mesh grids. (a) $N = 65 \times 65 \times 65$, (b) $N = 129 \times 129 \times 129$, and (c) $N = 257 \times 257 \times 257$.

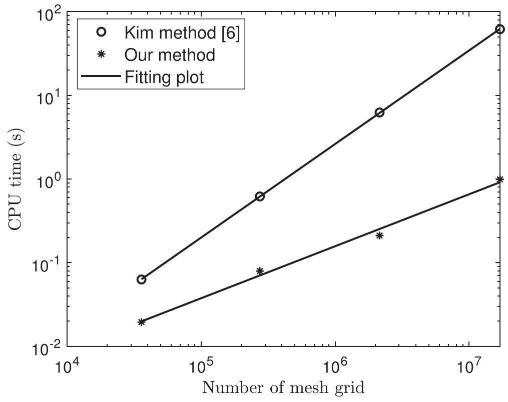

Fig. 7. Computational results with a linear fitting for the total CPU time versus the mesh grid.

4.3. Volume reconstructions using complex slice data

In this section, we describe the reconstruction of a 3D volume with several more complicated and complex slice data: the minimal surface, dragon, bunny, and armadillo. We consider the effect of a different N_z and the number of slice data between two given slices.

Fig. 8 (a)–(c) illustrate the volume reconstructions for the minimal surface with $N_z = 2$, 4, and 8, respectively. The tests are performed on $\Omega = (0, 1) \times (0, 1) \times (0, 1)$ with a mesh grid $128 \times 128 \times 128$. The other parameters are chosen as $\theta = 0$, $\Delta t = 0.1h^2$, $\epsilon = 3h$, and $\alpha = 10,000$. The results show that our algorithm works well for different numbers of inpainted slices.

Fig. 9 (a) and (b) show the reconstructed volume from the Chinese-dragon and XYZ-dragon models, respectively. From left to right, the original dragon model, parallel slices data used, and reconstructed volume are shown, respectively. Fig. 9(a) is performed on $\Omega = (0,\ 1) \times (0,\ 420/410) \times (0,\ 199/410)$ with a mesh grid size of $410 \times 420 \times 199$. Fig. 9(b) is performed on $\Omega = (0,\ 1) \times (0,\ 344/238) \times (0,\ 190/238)$ with a mesh grid $238 \times 344 \times 190$. In addition, 67 and 64 slices are chosen for testing the Chinese-dragon and XYZ-dragon models, respectively. The other parameters are chosen as $\theta = 0$, $\Delta t = 0.1\ \mathrm{h}^2$, $\epsilon = 3$ h, and $\alpha = 10,000$. By comparing the input slice data with their corresponding results, we can see that our algorithm can accurately reconstruct the volume and retain the details.

We also test our algorithm using the bunny and armadillo models, which have more details, such as the outline of the muscle and texture of the skin. These two reconstructions are performed on the domains $\Omega = (0,\ 1) \times (0,\ 213/169) \times (0,\ 193/169)$ and $\Omega = (0,\ 1) \times (0,\ 180/165) \times (0,\ 118/165)$ with $169 \times 213 \times 193$ and $165 \times 180 \times 118$ sized mesh grids, respectively. In addition, 65 and 40 slices are chosen for the bunny and armadillo models, respectively. The other parameters are the same as those used in the aforementioned tests. The computational results shown in Fig. 10 demonstrate that our algorithm can reconstruct clear volumes.

4.4. Quantitative comparisons

In this section, we compare our proposed method to related methods using the accuracy and efficiency as quantitative metrics. To show the difference between the reconstructed volume and ground truth, we use the Hausdorff distance between the vertices of two surfaces. The Hausdorff distance $d_H(X_1, X_2)$ between two vertex sets X_1 and X_2 is defined as follows:

$$d_H(X_1, X_2) = \max\{\sup_{x_1 \in X_1}\{\inf_{x_2 \in X_2}\{d(x_1, x_2)\}\}, \sup_{x_2 \in X_2}\{\inf_{x_1 \in X_1}\{d(x_1, x_2)\}\}\},$$

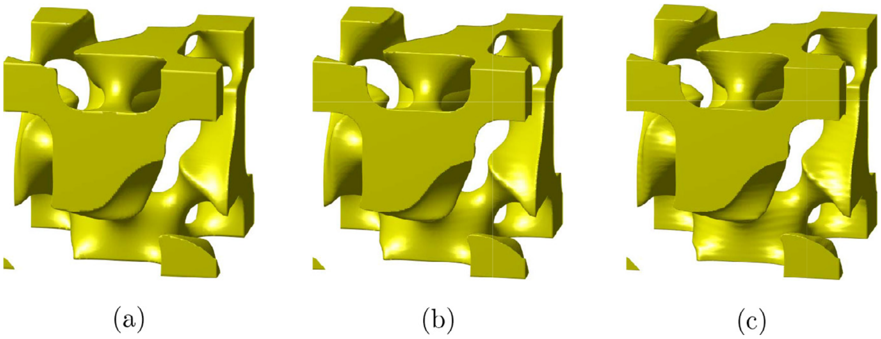

Fig. 8. Volume reconstruction of minimal surface with different numbers of inpainted slices: (a) $N_z = 2$, (b) $N_z = 4$, and (c) $N_z = 8$.

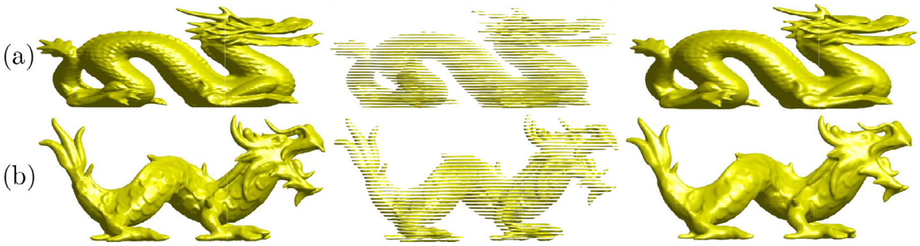

Fig. 9. Volume reconstructions from (a) Chinese-dragon and (b) XYZ-dragon models. From left to right, the original dragon model, parallel slice data used, and reconstructed volume, respectively.

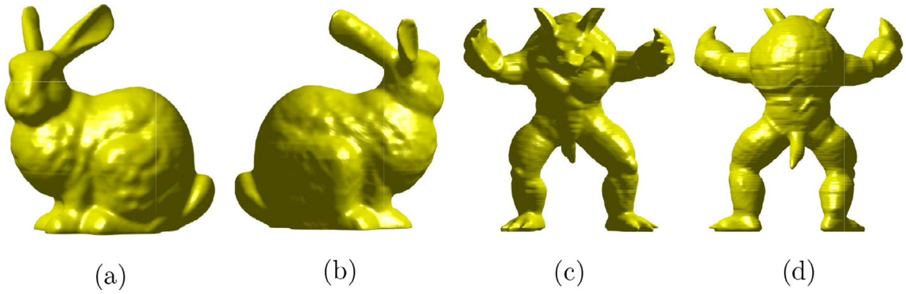

Fig. 10. Volume reconstructions for (a), (b) bunny and (c), (d) armadillo from different angles.

Table 1
Comparisons of reconstruction errors between our method and other methods. 'NSlice' is the number of parallel slices used.

NSlices	10	20	30	50	60
Li et al.'s method [17]	0.0471	0.0413	0.0367	0.0261	0.0173
Zou's method [15]	0.0468	0.0395	0.0312	0.0167	0.0104
Our method	0.0417	0.0312	0.0245	0.0113	0.0083

where $d(x_1, x_2)$ represents the Euclidian distance between vertices x_1 and x_2.

Table 1 lists the Hausdorff distance for the armadillo model with different numbers of parallel slices. We compare the current results with the those from Zou's method [15] and Li et al.'s method [17]. We observe that the computational results obtained from the proposed algorithm are qualitatively in good agreement with those in Zou [15].

For comparison with Li et al.'s method [17], we perform volume reconstructions using a human head and slice data of a human brain from [26]. The simulation of the human head is performed on $(0,1) \times (0,1) \times (0, 286/256)$ with a mesh grid size of $256 \times 256 \times 286$. Here, 96 parallel slice data are chosen. To test the human brain, we use 108 parallel slice data and set the simulation on the domain $(0,1) \times (0,1) \times (0, 322/256)$ with a $256 \times 256 \times 322$ mesh grid. The other parameters $\Delta t = 0.1\ \text{h}^2$, $\epsilon = 3$ h, and $\alpha = 10,000$ are used.

Fig. 11 shows the reconstructed volumes. From left to right, these are the original volumes, the results obtained using Li et al.'s method [17], and the results obtained using our proposed method, respectively. We can see that the proposed method can successfully manage complex topologies and maintain more topological details compared with Li et al.'s method [17]. The CPU times required for the human head and brain reconstruction are 12.72 s and 17.15 s, respectively. However, it takes 65.45 s and 91.26 s for Li et al.'s method [17]. The proposed algorithm achieves a reconstruction extremely quickly because the proposed method utilizes the second-order equation with lower computation cost compared to the fourth-order method in Li et al. [17].

4.5. Limitations of our method

Because our method first calculates alternative solutions for every two consecutive slices using the governing equation, and these alternative solutions are then selected to reconstruct the final volume, we can reconstruct the volume from a set of slice data. As a limitation of our proposed implementation, when we choose slice data with few slices, our method may fail to recover the volume with small details, and the volumes may be reconstructed with a staircase effect (see Fig. 12(a)). Here, 34 slice data are used. A possible way to overcome this disadvantage is to apply the modified

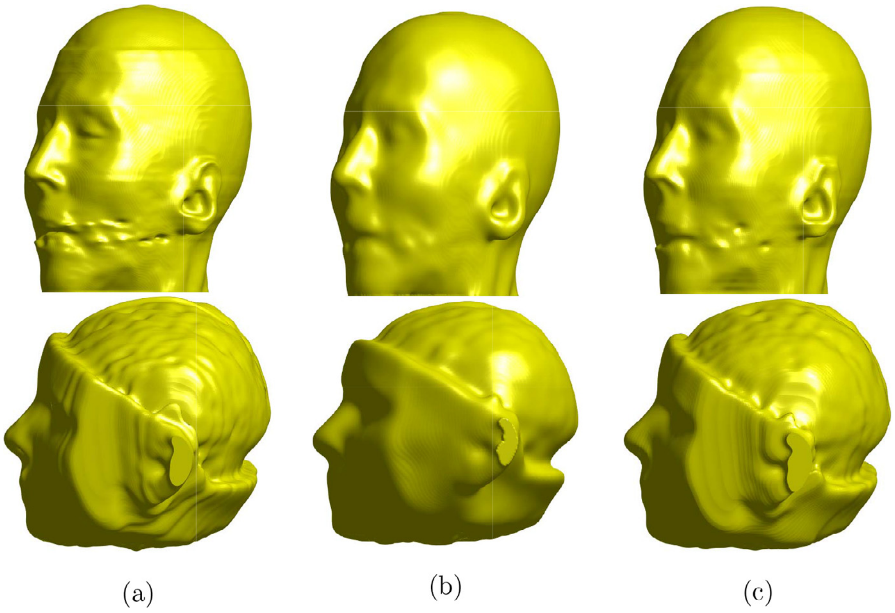

Fig. 11. Volume reconstructions for human head and brain models: (a) original surfaces, (b) results obtained by Li et al.'s method [17], and (c) results obtained using our approach.

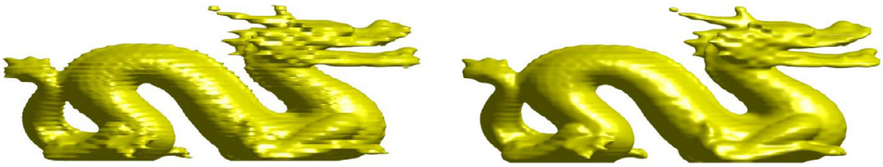

Fig. 12. Limitations of our method: (a) reconstructed volume with few slice data and (b) volume with post-processing using Eqs. (15) and (16).

AC equation to the entire domain with a few iterations:

$$\frac{\partial \varphi(\mathbf{x},t)}{\partial t} = -\frac{F'(\varphi(\mathbf{x},t))}{\epsilon^2} + \Delta\varphi(\mathbf{x},t) + \alpha\sqrt{F(\varphi(\mathbf{x},t))}(\phi(\mathbf{x}) - \varphi(\mathbf{x},t)), \quad (15)$$

$$\varphi(\mathbf{x},0) = \phi(\mathbf{x}). \quad (16)$$

Fig. 12(b) shows the results obtained using Eqs. (15) and (16) with the initial volume shown in Fig. 12(a). As can be seen the staircase effect is reduced and the volume becomes smooth. It is generally difficult to remove the staircase effect and retain small volumes. In the future, we will extend our method to recover small volumes with fewer slice data.

5. Conclusion

In this study, we presented an efficient and accurate computational algorithm for a weighted 3D volume reconstruction from multi-slice data using a shape transformation. The proposed algorithm is based on a modified AC equation with a fidelity term which includes the target shape. To reconstruct a 3D volume from the given source and target slices, we first set the source slice as the initial condition and the target slice as the fidelity function. After simulating the transformation process, the numerical solutions were maintained and used as intermediate slices between the source and target slices. Using a weighted average of two criteria, these intermediate slices were selected to reconstruct the final volume. To validate the efficiency and accuracy of the proposed computational method, several computational experiments were performed and the results of the numerical tests confirmed the superior performance of the proposed algorithm. Because the proposed method is intrinsically parallel, it would be interesting to implement the proposed algorithm on parallel computers in a future research topic.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The corresponding author (J.S. Kim) was supported by Korea University Grant. The authors appreciate the reviewers for their constructive comments, which have improved the quality of this paper.

References

[1] J.F. Guo, Y.L. Cai, Y.P. Wang, Morphology-based interpolation for 3D medical image reconstruction, Comput. Med. Imaging Graph. 9 (3) (1995) 267–279.

[2] J. Cheng, Y. Liu, 3-D reconstruction of medical image using wavelet transform and snake model, J. Multimed. 4 (6) (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, in: VCBM, 2008, pp. 77–82.

[4] C. Samir, I. Adouani, C^1 interpolating Bézier path on riemannian manifolds, with applications to 3D shape space, Appl. Math. Comput. 348 (2019) 371–384.

[5] G. Jo, Y.J. Lee, I. Ojeda-Ruiz, 2D and 3D image reconstruction from slice data based on a constrained bilateral smoothing and dynamic mode decomposition, Appl. Math. Comput. 420 (2022) 126877.

[6] J. Kim, C.O. Lee, Three-dimensional volume reconstruction using two-dimensional parallel slices, SIAM J. Imaging Sci. 12 (1) (2019) 1–27.

[7] E. Bretin, F. Dayrens, S. Masnou, Volume reconstruction from slices, SIAM J. Imaging. Sci. 10 (4) (2017) 2326–2358.

[8] D. Meyers, S. Skinner, K. Sloan, Surfaces from contours, ACM Trans. Graph. 11 (3) (1992) 228–258.

[9] J.D. Boissonnat, P. Memari, Shape reconstruction from unorganized cross-sections, in: Symposium on Geometry Processing, 2007, pp. 89–98.

[10] T. Ju, J. Warren, J. Carson, G. Eichele, C. Thaller, W. Chiu, M. Bello, I. Kakadiaris, Building 3D surface networks from 2D curve networks with application to anatomical modeling, Vis. Comput. 21 (8) (2005) 764–773.

[11] L. Liu, C. Bajaj, J.O. Deasy, D.A. Low, T. Ju, Surface reconstruction from non-parallel curve networks, Comput. Graph. Forum 27 (2) (2008) 155–163.

[12] M. Zou, M. Holloway, N. Carr, T. Ju, Topology-constrained surface reconstruction from cross-sections, ACM Trans. Graph. 34 (4) (2015) 1–10.

[13] Z. Huang, M. Zou, N. Carr, T. Ju, Topology-controlled reconstruction of multi--labelled domains from cross-sections, ACM Trans. Graph. 36 (4) (2017) 1–12.

[14] S.U. Kim, C.O. Lee, Accurate surface reconstruction in 3D using two-dimensional parallel cross sections, J. Math. Imaging Vis. 53 (2) (2015) 182–195.

[15] Q. Zou, A PDE model for smooth surface reconstruction from 2D parallel slices, IEEE Signal Process. Lett. 27 (2020) 1015–1019.

[16] A.L. Bertozzi, S. Esedoglu, A. Gillette, Inpainting of binary images using the Cahn–Hilliard equation, IEEE Trans. Image Process. 16 (1) (2006) 285–291.

[17] Y. Li, J. Shin, Y. Choi, J. Kim, Three-dimensional volume reconstruction from slice data using phase-field models, Comput. Vis. Image Underst. 137 (2015) 115–124.

[18] Y. Li, J. Wang, B. Lu, D. Jeong, J. Kim, Multicomponent volume reconstruction from slice data using a modified multicomponent Cahn–Hilliard system, Pattern Recognit. 9 (2019) 124–133.

[19] Y. Li, S. Lan, X. Liu, B. Lu, L. Wang, An efficient volume repairing method by using a modified Allen–Cahn equation, Pattern Recognit. 107 (2020) 107478.

[20] H. Kim, C. Lee, S. Kwak, Y. Hwang, S. Kim, Y. Choi, J. Kim, Three-dimensional volume reconstruction from multi-slice data using a shape transformation, Comput. Math. Appl. 113 (2022) 52–58.

[21] H. Kim, S. Yoon, J. Wang, C. Lee, S. Kim, J. Park, J. Kim, Shape transformation using the modified Allen–Cahn equation, Appl. Math. Lett. 107 (2020) 106487.

[22] D. Jeong, J. Kim, An explicit hybrid finite difference scheme for the Allen–Cahn equation, J. Comput. Appl. Math. 340 (2018) 247–255.

[23] Q. Yu, Y. Gao, Y. Zheng, J. Zhu, Y. Dai, Y. Shi, Crossover-net: leveraging vertical-horizontal crossover relation for robust medical image segmentation, Pattern Recognit. 113 (2021) 107756.

[24] Y. Yang, R. Wang, X. Shu, C. Feng, R. Xie, W. Jia, C. Li, Level set framework with transcendental constraint for robust and fast image segmentation, Pattern Recognit. 117 (2021) 107985.

[25] J.D. Yun, Y. Kim, Two-stage adaptive random fourier sampling method for image reconstruction, Pattern Recognit. 117 (2021) 107990.

[26] The Stanford volume data archive, Copyright © 2000, 2001 Marc Levoy, http://www-graphics.stanford.edu/data/voldata/.

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.

Xin Song is a Ph.D. student at the Department of Mathematics, Xin Jiaotong University, China. And he received B.S. degree in Mathematics from Xin Jiaotong University, China in 2019. His research interests are image processing, computational fluid dynamics, and scientific computing.

Soobin Kwak is a Ph.D. student at the Department of Mathematics, Korea University, Korea. And she received B.S. degree in Mathematics from Daegu University, Korea in 2020. Her research interests are mathematical modeling, computational fluid dynamics, and scientific computing.

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.