A parallel 3D image segmentation method for Coronary CT Angiography

Yu Chen*

School of Information Engineering, Yancheng Teachers University, 224002 Yan Cheng City, China

Abstract. Coronary artery (CA) disease is one of the major cardiovascular diseases that has been proved to be the leading cause of human death in world. In this paper, a 3D image segmentation algorithm based on Lattice Boltzmann (LB) is proposed for 3D CA segmentation. After investigating the behavior of Boundary Treatment schemes, a non-equilibrium extrapolation scheme is applied to keep the stability of the computation, and reduce the frequency of the re-initialization. The denoising and clipping method are also proposed for the segmentation refining. The segmentation result shows our LB model for 3D CAD segmentation is effective. More importantly, the LB model has natural parallelism. Our model can run on massively parallel architectures, ranging from inexpensive embedded FPGAs and DSPs up to GPUs.

1 Introduction

Coronary artery (CA) disease is one of the major cardiovascular diseases that has been proved to be the leading cause of human death in world. CA disease is caused by plaque buildup in the walls of the arteries that supply blood to the heart (called coronary arteries) and other parts of the body. Plaque buildup causes the inside of the arteries to narrow over time, which can partially or totally block the blood flow.Because of the fact that coronary artery disease often develops over decades, patients might not notice a problem until they have a significant blockage or a heart attack.

For those at risk of CA disease, Coronary CT angiography (CTA) is the major method to examine CA disease. CTA is convenient, noninvasive and effective to examine the stenosis or blockage of blood vessels. For this reason, CTA has become an important tool in screening, diagnosis and treatment of cardiovascular diseases. For many years, many researchers have carried out extensive research on quantitative description of cardiovascular disease and vascular dynamics based on CTA images. The key step of these studies is the two-dimensional (2D) or three-dimensional (3D) segmentation of CTA images.

However, there are challenges in CTA image segmentation, such as complex structure of coronary artery, variant vessel diameter, strong noise especially the motion artifacts caused by heart beating and low intensity contrast between the coronary artery and

* Corresponding author: TZCY@163.com

surrounding tissues. Therefore, CTA segmentation tends to segment CAD and the surrounding tissues as a whole which results in over segmentation.

In past few decades, the commonly used image segmentation methods are based on histogram threshold, active contour, region growing, artificial neural network, and etc. The methods on histogram threshold and region growing are prone to oversegment in the low intensity contrast region, while the artificial neural network needs huge workload on sample collection and labeling. Considering the image dimensions , compared to the traditional methods based on 2D image, 3D image segmentation is more accurate and reliable. This is because 3D image segmentation can directly use the depth information to separate the foreground and background, and can easily extract the target. 3D image segmentation has always been a classic challenge in the field of image processing[1][2][3]. The curve evolution methods based on active contour[4][5] and level sets[6][7] model are widely concerned, for example the CV model[8] proposed by Chan and Vese based on Munford-Shah segmentation model[9]. However, because of the huge amount of data in three-dimensional images, the computation of solving CV model is very large, which consumes a lot of time.

In this paper, a 3D image segmentation algorithm based on Lattice Boltzmann (LB) model[10] is proposed for 3D CAD segmentation. A a novel numerical tool, LB model is simple and efficient. More importantly, the LB model has natural parallelism. Our model can run on massively parallel architectures, ranging from inexpensive embedded FPGAs and DSPs up to GPUs.

2 CV Model

A CV model can be expressed in the summation of three energy terms.

$$E = E_1 + E_2 + E_3 \tag{1}$$

Where

$$E_1 = \mu \oint_C ds$$

$$E_2 = \lambda_1 \iint_{\Omega_1} (I - c_1)^2 dxdy$$

$$E_3 = \lambda_2 \iint_{\Omega_2} (I - c_2)^2 dxdy$$

In formula (1), Curve C is a curve that separates image domain Ω into two parts: Ω_1 (inside C) and Ω_2 (outside C). I is the image to be segmented. Variable c_1 and c_2 are the average pixel intensities in Ω_1 and Ω_2 respectively. μ , λ_1 , λ_2 are constant values that denotes the contributions of E_1 , E_2 , E_3 in curve evolution. Energy term E_1 represents the total length of curve C, while E_2 , E_3 are the intensity bias in Ω_1 and Ω_2 respectively.

Embedding curve C as the zero level set of distance function ϕ in domain Ω , Energy E can be rewritten as

$$E = \mu \iint_{\Omega} |\nabla H(\phi)| dxdy + \lambda_1 \iint_{\Omega} (I - c_1)^2 H(\phi) dxdy + \lambda_2 \iint_{\Omega} (I - c_2)^2 H(\phi) dxdy \quad , \tag{2}$$

Where H is Heaviside function:

$$H(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}.$$

The gradient of the Heaviside function H is:

$$\nabla H(\phi) = \nabla \phi \cdot \delta(\phi).$$

δ is so called Dirac function. Thus formula (1) is converted from line integral to surface integral in entire image domain.

Minimize formula (2) leads to the following partial differential equation.

$$\frac{\partial \phi}{\partial t} = \delta \mu \cdot div\left(\frac{\nabla \phi}{|\phi|}\right) - \delta\left[\lambda_1 (I - c_1)^2 - \lambda_2 (I - c_2)^2\right]. \tag{3}$$

The first term in the right side of equation (3) is a regular term that make equation (3) to be well-posed. The Dirac function is to limit the evolution in a narrow band nearby curve C. Practically, the regular term can be replaced by a simpler curvature term:

$$\frac{\partial \phi}{\partial t} = \mu \cdot div\left(\frac{\nabla \phi}{|\phi|}\right) - \delta_\varepsilon\left[\lambda_1 (I - c_1)^2 - \lambda_2 (I - c_2)^2\right]. \tag{4}$$

where,

$$\begin{cases} c_1 = \dfrac{\iint_{\Omega} H_\varepsilon(\phi) I dxdy}{\iint_{\Omega} H_\varepsilon(\phi) dxdy} \\ c_2 = \dfrac{\iint_{\Omega} [1 - H_\varepsilon(\phi)] I dxdy}{\iint_{\Omega} [1 - H_\varepsilon(\phi)] dxdy} \\ H_\varepsilon(x) = \dfrac{1}{2} + \dfrac{\arctan(x/\varepsilon)}{\pi} \\ \delta_\varepsilon(x) = \dfrac{1}{\pi} \dfrac{\varepsilon}{\varepsilon^2 + x^2} \end{cases} \quad , \tag{5}$$

H_ε , δ_ε are the regularized approximations of the Heaviside function H and δ respectively.

3 The basic lattice Boltzmann model

The basic idea of the LB method is the construction on simplified discrete dynamics for simulating the macroscopic model which are described by partial differential equations (PDE) using densities of particles moving on a regular lattice. The general Lattice Boltzmann modeling consists of two steps: a translation step in which particles move from node to node on a lattice and a collision step in which particles are redistributed at each node. The two steps can be represented by the discrete lattice Boltzmann equation (LBE). In this section, a D2Q9 model[11] is introduced, where "D2" stands for "2 dimensions", while "Q9" for "9 speeds". The discrete LBE is given by:

$$f_i\left(\vec{r}+\sigma_h\vec{e}_i,t+\sigma_t\right)-f_i\left(\vec{r},t\right)=-\frac{1}{\tau}\left(f_i\left(\vec{r},t\right)-f_i^{eq}\left(\vec{r},t\right)\right) \tag{6}$$

where τ is the relaxation factor, and $i=0,1,2,\ldots 8$ and $f_i\left(\vec{r}+\sigma_h\vec{e}_i,t+\sigma_t\right)$ is the particle density (also named density distribution function) on node $\left(\vec{r}+\sigma_h\vec{e}_i\right)$, at time $\left(t+\sigma_t\right)$ with time step σ_t and designating lattice spacing σ_h in direction $\vec{e}_i$:

$$\vec{e}_i=\begin{cases}(0,0), i=0\\ (1,0),(0,1),(-1,0),(0,-1), i=1,3,5,7\\ (1,1),(-1,1),(-1,-1),(1,-1), i=2,4,6,8\end{cases} \tag{7}$$

$$\sigma_h=v\sigma_t\,, \tag{8}$$

where v is the velocity of the particles.

The right hand side of the equation (6) represents the collision term, and the constant quantity τ is the single relaxation time, which controls the rate of approaching to equilibrium. The equilibrium distribution functions $f_i^{eq}\left(\vec{r},t\right)$ in direction $\vec{e}_i$ is only determined by the total density ρ which is defined in terms of the particle distribution functions:

$$f_i^{eq}=\begin{cases}\rho/(9c), i=1,2,3,\ldots,8\\ 1-8\rho/(9c), i=0\end{cases}, \tag{9}$$

where

$$\sum_{i=0}^{8}f_i=\sum_{i=0}^{8}f_i^{eq}=\rho, c\geq 1\ . \tag{10}$$

f_i^{eq} and f_i must be positive, and c is the regulator for the diffusion speed. Formula (6) -(10) described a D2Q9 (2 dimensions 9 directions) LB model. Using Chapman-Enskog expansion and Taylor developing both sided of equation (6), we can obtain the macroscopic equation of the LB model in the form of PDE as follows (The derivation can be referred to [10]):

$$\frac{\partial\rho}{\partial t}=\frac{\sigma_h^2}{18c\sigma_t}div\left[(\tau-0.5)\nabla\rho\right]. \tag{11}$$

4 LB based two-dimensional CV model

Equation (11) is an anisotropic diffusion model while the relaxation factor τ is variable value. In our model, τ is defined as: $\tau = 0.5 + \frac{1}{|\nabla\rho|}$. Therefore, the global description of the LB model introduced in section 3 is written as:

$$\frac{\partial\rho}{\partial t} = \frac{\sigma_h^2}{18c\sigma_t} div\left[(\tau - 0.5)\frac{\nabla\rho}{|\nabla\rho|}\right]. \tag{12}$$

Compared with equation (4), a new LB model is modified based on formula (6):

$$f_i(\vec{r} + \sigma_h\vec{e}_i, t + \sigma_t) - f_i(\vec{r}, t) = -\frac{1}{\tau}\left(f_i(\vec{r}, t) - f_i^{eq}(\vec{r}, t)\right) + \sigma_t F \ , \tag{13}$$

where F is diffusion source which is defined as:

$$F = \delta_\varepsilon\left[(I - c_1)^2 + (I - c_2)^2\right]. \tag{14}$$

Similar to section 2, we can obtain the macroscopic equation:

$$\frac{\partial\rho}{\partial t} = \frac{\sigma_h^2}{18c\sigma_t} div\left[(\tau - 0.5)\frac{\nabla\rho}{|\nabla\rho|}\right] + \sigma_t\delta_\varepsilon\left[(I - c_1)^2 + (I - c_2)^2\right]. \tag{15}$$

5 LB method based three-dimensional CV model

The two-Dimensional CV model can be directly modified to Three-Dimensional.

The equilibrium distribution function is as follows:

$$f_i^{eq} = \begin{cases} \rho/(15c), i = 1,2,3,...,14 \\ 1 - 14\rho/(15c), i = 0 \end{cases}. \tag{16}$$

Direction $\vec{e}_i$ (as shown in Fig. 1) is defined as:

$$\begin{aligned} \vec{e} &= \left[\vec{e}_0,\ \vec{e}_1,\ \vec{e}_2,\ \ldots\ ,\ \vec{e}_{12}\right] \\ &= \begin{bmatrix} 0 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}. \end{aligned} \tag{17}$$

The macroscopic equation is shown as following:

$$\frac{\partial\rho}{\partial t} = \frac{2\sigma_h^2}{3c\sigma_t} div\left[\frac{\nabla\rho}{|\nabla\rho|}\right] + \sigma_t\delta_\varepsilon\left[(I - c_1)^2 + (I - c_2)^2\right]. \tag{18}$$

For simplicity, σ_h and σ_t are set to 1:

$$\frac{\partial\rho}{\partial t} = \frac{2}{3c} div\left[\frac{\nabla\rho}{|\nabla\rho|}\right] + \sigma_\varepsilon\left[(I - c_1)^2 + (I - c_2)^2\right]. \tag{19}$$

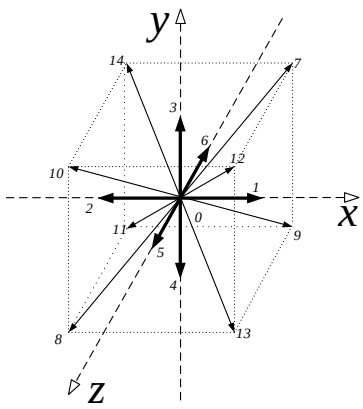


Fig. 1. Direction $\vec{e}_i$.

6 Experiment and discussion

6.1 Algorithm

The algorithm consists of following steps:

a. Initialize the distance function ρ ;

b. Initialize $f_i = \begin{cases} \rho/(15c), i = 1,2,3,...,14 \\ 1-14\rho/(15c), i = 0 \end{cases}$;

c. Initialize $f_i^{eq} = \begin{cases} \rho/(15c), i = 1,2,3,...,14 \\ 1-14\rho/(15c), i = 0 \end{cases}$;

d. Collision and streaming according to formula (13);

e. Update distance function $\rho(x,y,z) = \sum_{i=0} f_i$.

f. Reinitialize the distance function ρ ;

g. Go to step c.

h. Iteration stops if c_1 and c_2 do not changes.

6.2 Parameter settings

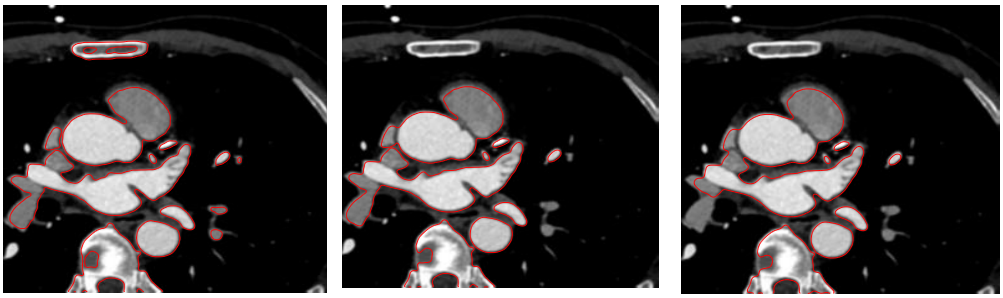
It should be noted that the value of the distance function ρ must be greater than 0 to ensure the stability of the LB model. That is because the LB will get unstable if the particle density is negative. For simplicity, we initialize the distance function as following formula:

$$\rho(x,y,z) = B + \sqrt{(x-x_c)^2 + (y-y_c)^2 + (z-z_c)^2} - radius, \tag{20}$$

where $\sqrt{(x-x_c)^2 + (y-y_c)^2 + (z-z_c)^2} - radius$ initialize the contour as a circle centered at point (x_c, y_c, z_c). The distance function is positive outside the contour and negative inside. Parameter B should satisfy $B >> radius$, so that distance function $\rho >> radius$. According

to equation (11), relaxation factor τ should be larger than 0.5, otherwise equation (11) will get ill-posed.

Diffusion coefficient $\frac{2}{3c}$ not only determines the contribution of the regular term to the curve evolution,but also affects the behavior of the segmentation. Obviously, A smaller c means increased rigidness of the curve, but more stability of the model, and vice versa. In Fig. 2(a)-(c), the CAD segmentation with c=50, 5, 1 is presented. With the decreasing of c, some small tissues are missed in the segmentation, and the segmentation is of lower accuracy. This is mainly because the curve are getting more rigid, in other words, the curvature of the curve is decreased. However, a larger c will reduce the contribution of the regular term, which will cause the instability of the model. In the experiments, we suggest set c to 10.



c=50 *c*=5 *c*=1

Fig. 2. CAD segmentation with c=50, 5, 1.

7 Boundary treatment

Periodic scheme and bounce-back scheme[12] are two common methods to deal with simple boundaries in lattice Boltzmann model. However, both the schemes have poor universality, for example, they can not deal with the boundary containing gradient information. In the experiments, these two boundary treatment schemes may degrade the distance function quickly, as a consequence, re-initialization of the distance function should be implemented frequently.

In our research, we use an non-equilibrium extrapolation scheme[13] which is more stable and easy to implement. As shown in Fig. 3, assume points E, O and A locate on the boundary of the image; B,C,D locate inside the image while F,G,H outside the image.

Take point O for instance, the particle density $f_i(O,t)$ is divided into two parts: equilibrium part $f_i^{eq}(O,t)$ and non-equilibrium part $f_i^{neq}(O,t)$.

$$f_i(O,t) = f_i^{eq}(O,t) + f_i^{neq}(O,t), \tag{21}$$

where

$$\begin{cases} f_i^{eq}(O,t) \approx f_i^{eq}(c,t) \\ f_i^{neq}(O,t) \approx f_i(c,t) - f_i^{eq}(c,t) \end{cases}. \tag{22}$$

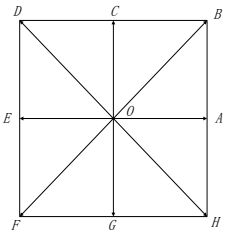


Fig. 3. Points E, O and A locate on the boundary of the image; B,C and D locate inside the image while F, G and H outside the image.

To investigate the performance of the non-equilibrium extrapolation scheme, two indexes are used, such as the mean value and variance of gradient $|\nabla\rho|$ denoted by MVG and VG respectively. It is worth noting that, we set F to 0 so that the distance function ρ is not effected by the diffusion source which can change the distance function dramatically in the image segmentation. The indexes MVG and VG over iterations based on non-equilibrium extrapolation and periodic scheme are plotted in Fig. 4 and Fig. 5 respectively. Fig. 4 illustrate that, compared to the non-equilibrium extrapolation scheme, the MVG based on periodic scheme decreased much rapidly, and almost equals to 0 after about 2700 iterations. This means the distance function ρ has degraded to a constant function. In Fig. 5, the VG based on periodic scheme increase to the maximum after 700 iterations, and then, decreased to 0 after 2700 iterations. Actually this is due to the fact that, in the first few iterations, the distance function degrades to 0 only nearby the image boundary. And then, because of the diffusion procedure, the degradation expands quickly to the whole image domain until the distance function equals to constant 0. This indicates that the periodic scheme is more unstable which need to re-initialize the distance function more frequently than non-equilibrium extrapolation scheme.

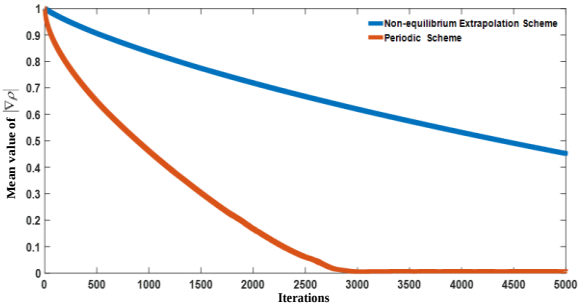


Fig. 4. Mean value of $|\nabla\rho|$ over iterations.

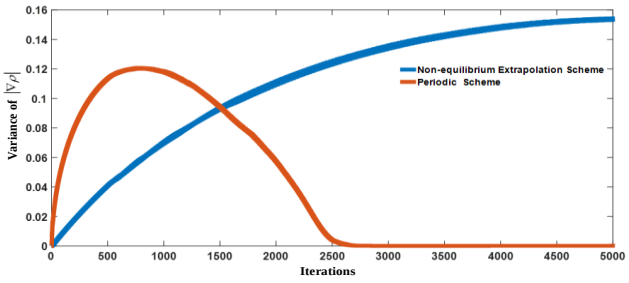


Fig. 5. Variance of $|\nabla\rho|$– over iterations.

7.1 Experiments

All the experiments are implemented on a PC with Intel(R) Core(TM) i7-4702MQ CPU, 8G RAM, with Matlab 2016. Considering the compromise between the calculation accuracy and time consuming, the D3Q15 LB model is used. Actually, there are many 3D LB models except for D3Q15, e.g. D3Q7, D3Q19 and D3Q27. A LB model with more velocities can achieve higher calculation accuracy but more time consuming.

Fig. 6 shows the 2D image (181 ×217×181) segmentation results. Figure 3 shows the CTA volume data segmentation in 3D view.

In Fig. 7, the segmentation are seriously corrupted by strong noise, such as the the artifacts of cardiac motion, tissues of similar contrast to the CAD, bones of high voxel intensity, and the noise introduced by the CTA image reconstruction. In this paper, a refining, including the data denoising and clipping, is implemented as the post-processing.

In practice, we use the connected components based algorithm to denoise the segmentation at first. For purposes of denoising we deal with the segmentation according to the following 3 major steps:

- Find the connected components in the segmentation;
- Search the largest connected component;
- Select the largest connected component as the denoised segmentation.

Fig. 8 shows the denoising result of the CAD volume segmentation. The result is displayed by open-source software ParaView. We can see that the left and right coronary artery have been clearly shown.

For purposes of clip the segmentation, we use a plane to cut the segmentation into 2 parts as shown in Fig. 9(a). The plane is defined in a point normal form. Considering the following function:

$$f(x,y,z)=n_x(x-x_0)+n_y(y-y_0)+n_z(z-z_0), \tag{23}$$

The point-normal form of the equation of a plane is $f(x,y,z)=0$, where $\langle n_x,n_y,n_z\rangle$ is the given normal vector and $(x_0,y_0,z_0)$is the given point. Apparently, a point to the same side of the plane as the direction of the normal vector satisfies $f(x,y,z)>0$, and the other side $f(x,y,z)<0$. A point (x_n,y_n,z_n) that will be ruled out should met the following conditions:

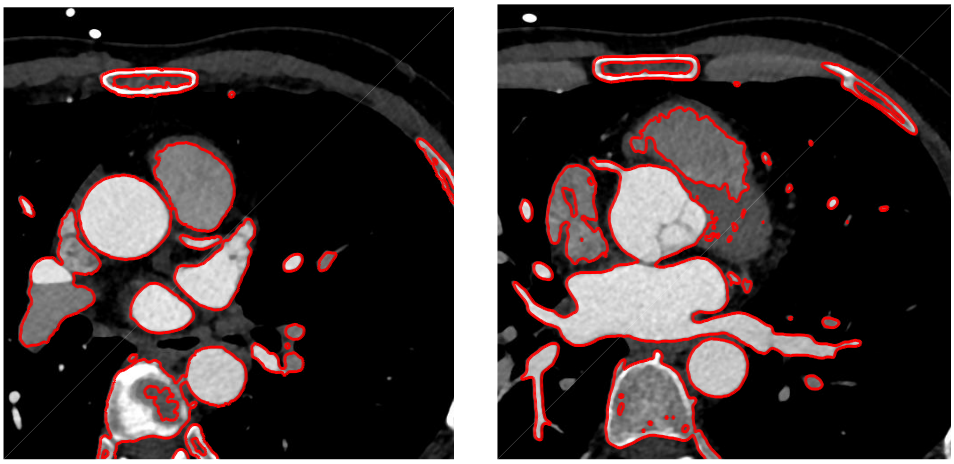


Fig. 6. CAD segmentation based on 2D CTA image.

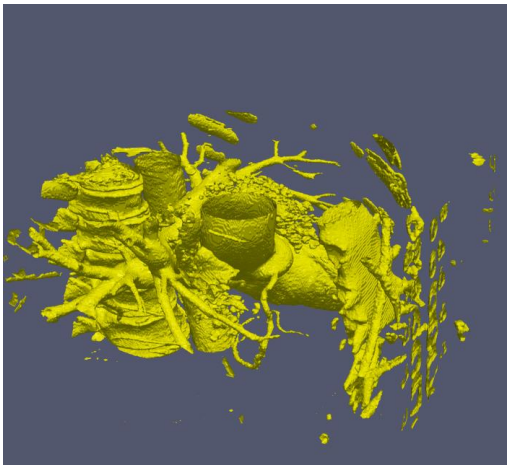


Fig. 7. CTA volume data segmentation.

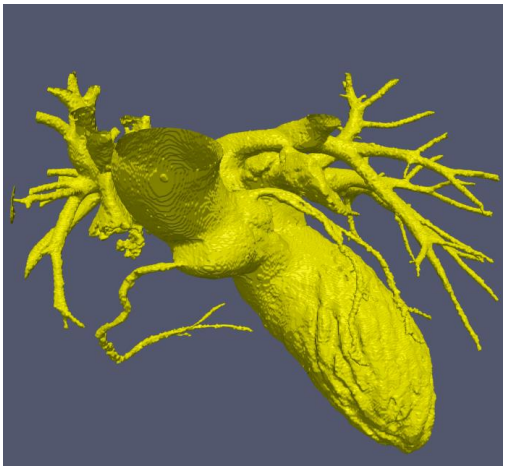


Fig. 8. CTA volume data segmentation after denoising.

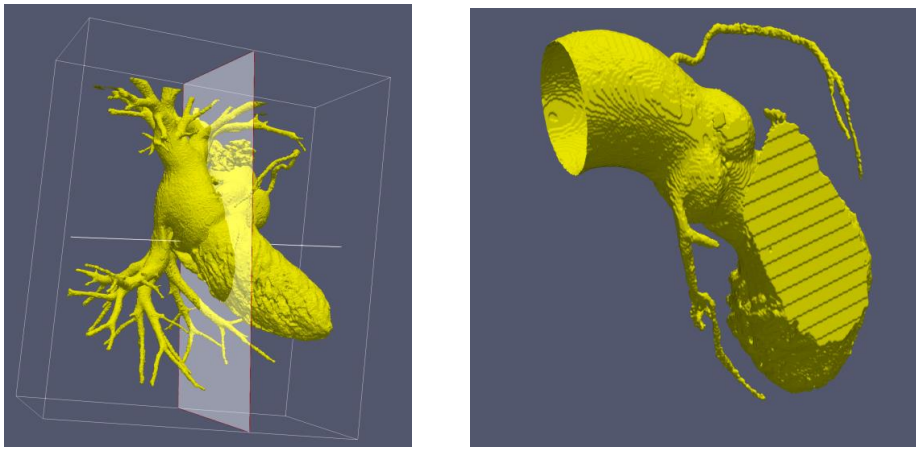
$$f(x_n, y_n, z_n) > 0;$$

The distance d between point (x_n, y_n, z_n) and line $\frac{x-x_0}{n_x} = \frac{y-y_0}{n_y} = \frac{z-z_0}{n_z}$ is less than R.

We suggest to refine the segmentation using data clipping at first, and then followed by the data denoising. Two refinings are performed in our experiment. Table 1 lists the $\langle n_x, n_y, n_z \rangle$, (x_0, y_0, z_0) and R of the refining. The refining results are shown in Fig. 9 - 10.

Table 1. Parameter setting in the refinings of our experiment.

Index of refining	$\langle n_x, n_y, n_z \rangle$	(x_0, y_0, z_0)	R	Results
1	$\langle 0.99, 0.093, 0.091 \rangle$	$(282.7, 237.8, 117.3)$	300	Fig. 9
2	$\langle 0.027, -0.646, -0.763 \rangle$	$(245.0, 208.6, 107.7)$	10	Fig. 10



a. Clipping plane b. Refining result

Fig. 9. The first refining. a) The plane used to clip the segmentation; b) the processing result after clipping and denoising.

8 Conclusion

In this paper, we proposed a 3D CA segmentation in coronary CT Angiography. The algorithm is simple and efficient. More importantly, the LB model has natural parallelism. We investigate the behavior of the affect of Boundary Treatment to the distance function. The value and variance of $|\nabla\rho|$ over iterations show different boundary treatments may cause different effects on the degrade speed of the distance function. In our research, the non-equilibrium extrapolation scheme is applied to keep the stability of the computation, and reduce the frequency of the re-initialization.

The segmentation result shows our LB model for 3D CAD segmentation is effective. The algorithm is simple and efficient. More importantly, the LB model has natural parallelism. Especially, our method can run on massively parallel architectures, ranging from inexpensive embedded FPGAs and DSPs up to GPUs.

This work was supported by Jiangsu industry university research cooperation project (BY2020650).

References

1. P. Badura. Virtual bacterium colony in 3D image segmentation. Computerized Medical Imaging and Graphics, vol.65, pp.152-166, 2017.
2. Z. Jianpeng, X. Yutong, W. Yan, X. Yong. Inter-Slice Context Residual Learning for 3D Medical Image Segmentation, IEEE Transactions on Medical Imaging, vol.40, pp.661-672, 2021.
3. W. Yan, L. Chen-Luh, M. Jan D. 3D image segmentation for analysis of multisize particles in a packed particle bed. Powder Technology, vol.301, pp.160-168, 2016.
4. S. Kichenassamy, A. Kumar, P. Olver, A Tannenbaum, A Yezzi. Gradient flows and geometric active contour models[C]. International conference on computer vision, pp.810-815, 1995.
5. Y. Rathi, N. Vaswani,A. Tannenbaum, A. Yezzi Particle filtering for geometric active contours with application to tracking moving and deforming objects[C]. 2005 IEEE

Computer Society Conference on Computer Vision and Pattern Recognition,vol.2, pp.2-9, 2005.

6. M. Subasic. Level Set Methods and Fast Marching Methods[J]. computer and information technology, vol.11, pp.79-79, 2003.
7. S. Osher. Level Set Methods and Dynamic Implicit Surfaces. Level sets methods and dynamic implicit surfaces. Springer. 2003.
8. T. F. Chan and L. A. Vese. Active contours without edges[J]. IEEE Transactions on Image Processing, vol.10 , pp: 266-277,2001.
9. A. Tsai, A. Yezzi, A. Willsky. Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification[J]. IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society, vol.10, pp:1169-86, 2001.
10. Y. H. Qian, D. D'Humières, P. Lallemand. Lattice BGK models for Navier-Stokes equation. Europhysics Letters, vol.17, pp: 479-484, 1992.
11. L. Li, R. Mei, and J. F. Klausner. Lattice Boltzmann models for the convection diffusion equation: D2Q5 vs, D2Q9. International Journal of Heat & Mass Transfer, vol. 108, pp:41-62, 2017.
12. A. Mohamad. Lattice Boltzmann method: fundamentals and engineering applications with computer codes. Springer Science & Business Media, 2011.
13. G. Zhaoli, Z. Chuangguang, S. Baochang. Non-equilibrium extrapolation method for velocity and pressure boundary conditions in the lattice Boltzmann method. Chinese physics, vol.11, pp:366-374, 2002.