

Medical Image Segmentation Using a Combination of Lattice Boltzmann Method and Fuzzy Clustering Based on GPU CUDA Parallel Processing

<https://doi.org/10.3991/ijoe.v17i11.24459>

Ignasius Boli Suban, Suyoto^(✉), Pranowo
Universitas Atma Jaya Yogyakarta, Yogyakarta, Indonesia
suyoto@staff.uajy.ac.id

Abstract—The rapid development of computer technology has had a significant influence on advances in medical science. This development concerns segmenting medical images that can be used to help doctors diagnose patient diseases. The boundary between objects contained in an image is captured using the level set function. The equation of the level set function is solved numerically by combining the Lattice Boltzmann (LBM) method and fuzzy clustering. Parallel processing using a graphical processing unit (GPU) accelerates the execution of the segmentation process. The results showed that image segmentation with a relatively large size could be done quickly. The use of parallel programming with the GPU can accelerate up to 39.22 times compared to the speed of serial programming with the CPU. In addition, the comparisons with other research and benchmark data show consistent results.

Keywords—Lattice Boltzmann, fuzzy clustering, GPU

1 Introduction

Image segmentation has a vital role in medical imaging. Image segmentation is the process of separating the objects contained in an image. Thus, this segmentation can be used to study the body's anatomy [1][2], volume estimation [3][4], tumors identify [5][6], cancer [7], or the eyes and surrounding organs [8]. The obstacles faced in image segmentation are artifacts such as noise, intensity inhomogeneity, low contrast, or non-sharp edges in the image. The main difficulties in the study of images are mainly image segmentation [9].

A vast number of methods have been proposed in the field of medical image segmentation. A well-established class of ways is the level set method (LSM) proposed by Osher and Sethian (1988) and Choudhry and R. Kapoor (2018) to solve surface evolution problems [10][11]. Medical image processing algorithms are generally sensitive to noise and consist of low informative regions [12]. The Level set is robust and accurate, and the segmentation process does not need prior initialization knowledge. This method involves a spatial function that can be expressed as a contour where the absolute value of the normal vector perpendicular to the contour is always one. These contours can

dynamically change shape following a pixel gradient, so they can be used to find boundaries and enclose areas of an object in the image. The model equation of the level set function is a partial differential equation (PDE). This equation is solved using a numerical method called the Finite Difference method. This method's shortcoming is the high computational load, and the contour is deteriorated due to the numerical dissipation effect. Hence, the level set is no longer a signed distance function. For long time calculations, the calculation accuracy will be significantly reduced,

Reinitialization is used in the LSM to remedy the interface's numerical deterioration. This costly reinitialization allows us to keep the norm of the gradient close to unity. Several researchers have attempted to improve the performance of the set-level method without reinitialization. Li et al. at 2005 proposed a variational formulation to force the set level function to be close to a signed distance function. Then Li, in 2010, continues by developing the distance regularized level set evolution (DRLSE) method. Li et al. also showed that their proposed plans have outstanding performances [13]. Zhang et al. (2013) added a diffusion equation to the Level Set equation to avoid the reinitialization process [14]. The calculation is divided into two steps; the first step calculates the set level method and is continued by solving the diffusion term. Diffusion terms can increase the stability of numerical calculations. The calculation results show that the proposed way is more straightforward, has better performance, and is more effective than the conventional LSM.

Generally, fuzzy clustering algorithms merge different clustering parameters for a cluster head (CH) election [15]. Li et al. (2016) discussed fuzzy multilevel image thresholding based on modified discrete grey wolf optimizer and local information aggregation [16][4]. Fuzzy clustering is used to initialize the LSM so that the LSM requires fewer iteration. Khosravanian, Asieh et al. (2020) also developed similar medical image segmentation methods [6].

The use of the finite difference method for the LSM's discretization increases the computational load. The Lattice Boltzmann method (LBM) is proposed to reduce the LSM's numerical solution's computational load. The LBM is a mesoscopic method developed based on statistical mechanics theory and gas-particle kinetic theory. The LBM simulates the motion and collision of gas particles. Using the Chapman Enskog transformation, a partial differential equation can be obtained, including the level set function. This method's advantage is that implementing a computer program is relatively simple, and the execution time is shorter [17]. Balla-Arabé and Gao (2012) proposed image multi-thresholding by combining the LBM and a localized level set algorithm [18]. Then in 2013, they developed an image multi-thresholding method for the localized LSM, which is solved using the LBM. The results show that this approach is faster and accurate [19]. Wen et al. (2014) presented the LBM based on image area integration for the medical image segmentation process. This method can be used to handle images with the inhomogeneous property [20]. Chen et al. (2014) developed the Lattice Boltzmann Geodesic Active Contour Method (LBGM) for the segmentation of medical images of giant intracranial aneurysms (GIA) [21]. Wang (2018) proposed a hybrid LSM using the LBM and Sparse Field Constraint. The accuracy, efficiency, and robustness of the segmentation process can be improved using the proposed method [13] [22].

On the other hand, parallel processing of image segmentation will perform multiple image segmentation in parallel, saving a lot of time for the user [23]. Recently parallel processing based on graphics processing units (GPUs) is used today in various applications, including medical image processing. Because they are cheap, efficient, and they can dramatically accelerate the numerical computing process. Roels et al. show achieving speedup factors of more than 50x GPU compared to single-threaded CPU execution are not uncommon due to parallel processing [24]. Chen et al. used a GPU-based parallel LBM for raw image contour detection screening. Execution time can be shortened by 25x, and contours can still be detected even if the image is contaminated with noise [21]. Kumar et al. proposed a joint LSM and LBM-based approach for histogram thresholding in image segmentation [25]. Kalaiselvi et al. presented medical image processing algorithms implemented in CUDA running on GPU-based Machines [26], and Huang and Li use CUDA to accelerate in parallel computation [27].

Balla-Arabé and Gao (2013) developed the LBM combined fuzzy clustering as the external force function to solve the LSM. Using this method, segmentation of image contained intensity inhomogeneity, and blurred object boundaries can be handled easily. It is also shown that the technique needs fewer iteration and is faster compared with a well-known method, such as the CV method and Gibou–Fedkiw method [19]. Based on the paper, we propose combining the Lattice Boltzmann method and fuzzy clustering based on GPU Compute Unified Device Architecture (CUDA) Parallel Processing for medical image segmentation.

2 Materials and methods

2.1 Hardware and software

The hardware used in this research is 1 unit PC. The specifications: CPU consists of 6 Processor Intel I7 @ 3.70 GHz, GPU Nvidia GeForce GTX 1080 Ti, and RAM 32 GB DDR4, and the specifications of the software are: Windows 10 operating system, C++ Ms. Visual Studio 2012, and CUDA Toolkit Version 9.

2.2 Dataset

The dataset in this study consisted of two types of images, namely real-world images and biomedical images. The real-world image and biomedical image file format are .jpg. Real-world images are used to compare the author's segmented image with the segmented image in previous studies. Then the author continues to do the segmentation process using biomedical images.

The images and the humans' segmentations are taken from the Berkeley segmentation dataset BSDS300 (www2.eecs.berkeley.edu). The author uses six real-world images with a size of 256x256. Then, the Biomedical Image used by the author is six images where the images have sizes that vary from 512x512 to 4096x4096.

2.3 GPU CUDA

GPU CUDA uses the kernel to execute a command n-times parallel on the GPU device from a programming perspective. An n-thread number handles this execution. The kernel is a subprogram in C language implemented on the GPU device and called by the host (CPU). The sequence interaction between the host (CPU) and device (GPU) is illustrated in Figure 1. The host launches the kernel and the thread asynchronously to the GPU and waits for parallel execution in the GPU to finish before tackling the next kernel.

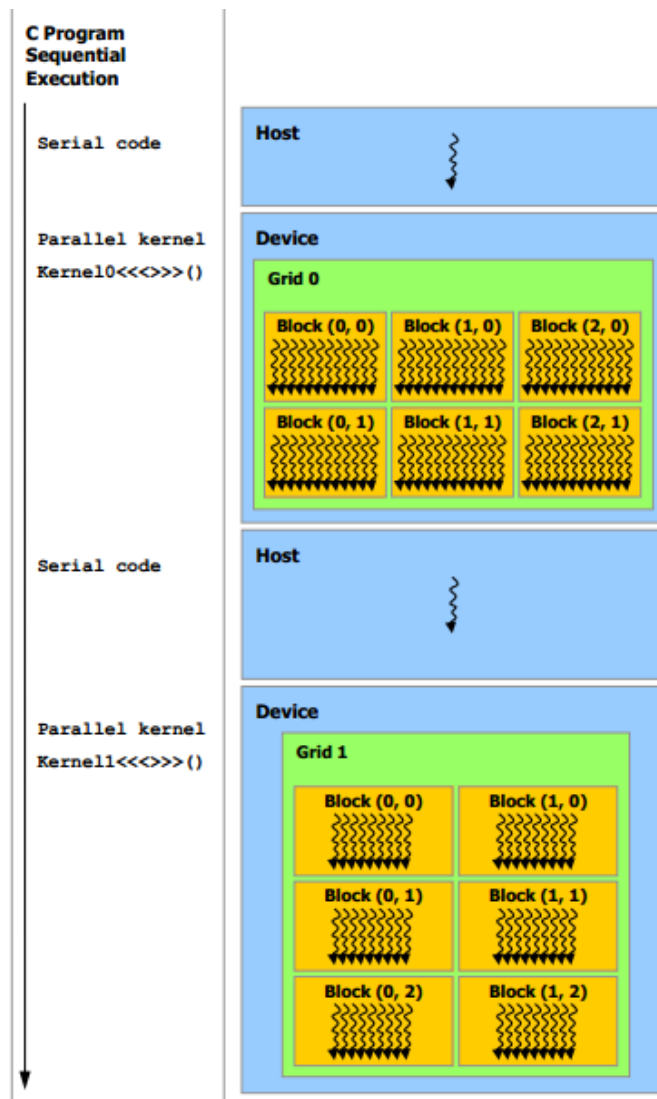


Fig. 1. CPU-GPU execution interactions

GPU CUDA executes a kernel using many threads, and each thread runs a specific statement. The threads are organized into blocks and grids. The maximum number of threads per block is 1024. Threads in a block can be arranged according to a 1,2, or 3-dimensional hierarchy. Each grid consists of some blocks, arranged according to a hierarchy of 1, 2, or 3-dimensions. The number of blocks should be a multiple of 32. Figure 2 shows the organization of threads, blocks, and grids, which are indexed using 2-dimensions.

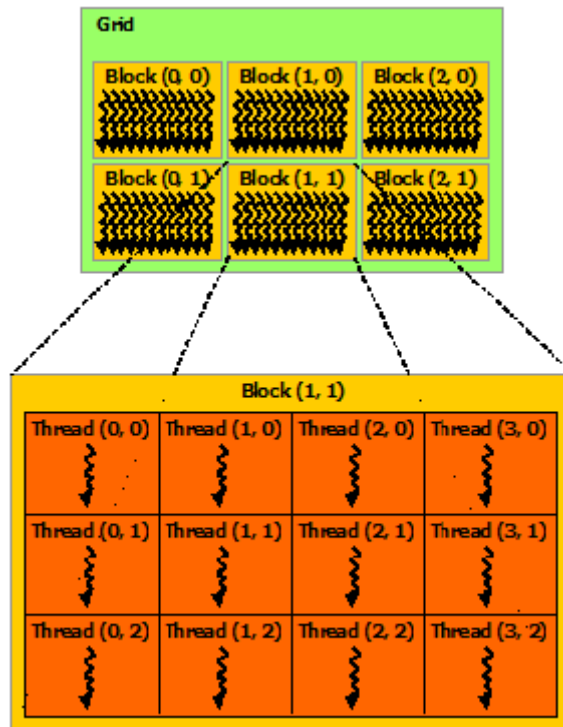


Fig. 2. Organization of threads, blocks, and grids

2.4 Numerical method

a) Mathematical model

The mathematical model for the evolution of level set change is written as:

$$\frac{\partial \phi}{\partial t} = \gamma \left(\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} \right) + F \quad (1)$$

where:

ϕ is the level set function

F is an external force function in the form of fuzzy clustering

(x,y) are the coordinates of the pixel position
 t is time
 γ is the diffusion coefficient

Hagan and Zou's (2009) external force function accommodates the speed function and the curve curvature change process [28]. In Balla-Arabé and Gao (2013), the part accommodates pixel grouping using the fuzzy clustering method [19].

b) Lattice Boltzmann method

The Lattice Boltzmann (LBM) method was developed for solving fluid flow problems but later developed for image processing, which uses partial differential equations (PDE) as the mathematical model [13][17][22]. For image processing, the pixel position is analogous to the lattice, where several variables are called the particle distribution function. This study uses the D2Q5 scheme [29], which has five kinds of particle distribution in each lattice ($f_i, i = 0, \dots, 4$). The five particle distribution functions move at the following speed and direction \mathbf{e}_i as described below:

$$\mathbf{e}_i = \begin{cases} (0,0), & i = 0 \\ (1,0), (0,1), (-1,0), (0,-1), & i = 1,2,3,4 \end{cases} \quad (2)$$

The particle distribution function's speed and direction for each lattice for the D2Q5 scheme are described in Figure 3.

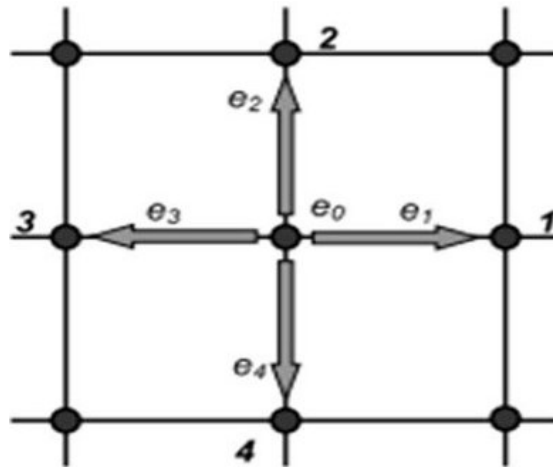


Fig. 3. The D2Q5 scheme, along with the speed direction of the particle distribution function

The process of changing the particle distribution function at position \mathbf{x} (x, y) and time t with the Bhatnagar-Gross-Krook (BGK) approach [30][31] can be written as below

$$f_i(\mathbf{x} + \mathbf{e}_i \Delta t, t + \Delta t) = f_i(\mathbf{x}, t) - \frac{1}{\tau} (f_i(\mathbf{x}, t) - f_i^{eq}(\mathbf{x}, t)) + \frac{2\tau-1}{2\tau} F_i \quad (3)$$

where Δx , Δt dan τ are the lattice spacing, the time step, and the relaxation time, while f_i^{eq} the particle distribution function is in a state of balance (equilibrium) and is calculated by:

$$\begin{aligned} f_0^{eq}(x, y, t) &= \frac{\phi(x, y, t)}{3} \\ f_i^{eq}(x, y, t) &= \frac{\phi(x, y, t)}{6}, \quad i = 1, \dots, 4 \end{aligned} \quad (4)$$

The relaxation time (τ) is related to the diffusion coefficient by:

$$\tau = \frac{9\gamma + 2}{4} \quad (5)$$

The value of the level set in each lattice is the sum of the five particle distribution functions:

$$\phi(t, x, y) = \sum_{i=0}^4 f_i(t, x, y) = \sum_{i=0}^4 f_i^{eq}(t, x, y) \quad (6)$$

The calculation of equation (3) consists of 2 steps, the first step is the collision, and the second step is streaming:

$$f_i^*(\mathbf{x}, t) = f_i(\mathbf{x}, t) - \frac{1}{\tau} (f_i(\mathbf{x}, t) - f_i^{eq}(\mathbf{x}, t)) + \frac{2\tau - 1}{2\tau} F_i \quad (7a)$$

$$f_i(\mathbf{x} + \mathbf{e}_i \Delta t, t + \Delta t) = f_i^*(\mathbf{x}, t) \quad (7b)$$

f_i^* is the post-collision particle distribution function. The streaming step is illustrated in Figure 4.

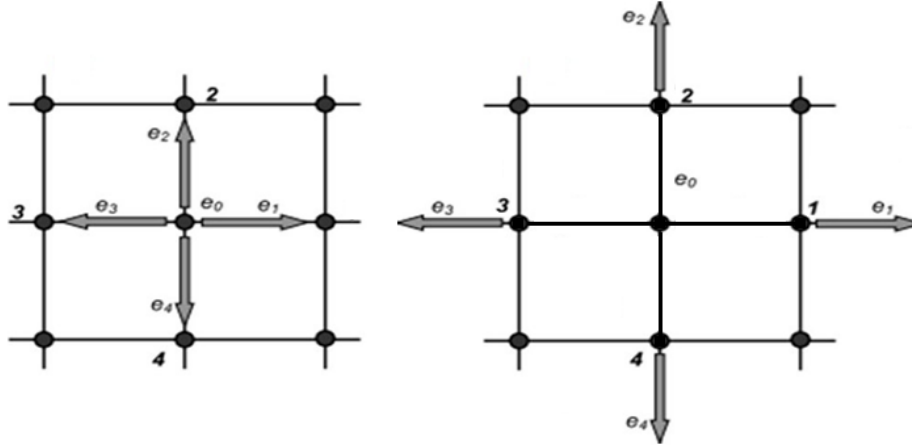


Fig. 4. Streaming step illustration.

c) Fuzzy clustering

In this study, the cluster of the segmentation consists of 2 classes only. The calculation of the external force function that accommodates Fuzzy clustering is written as follows:

$$F(x, y) = \lambda \left(U_1^p(x, y) \|Y(x, y) - B(x, y) - v_1\|^2 - U_2^p(x, y) \|Y(x, y) - B(x, y) - v_2\|^2 \right) \quad (8)$$

where

- λ : positive parameter
- $Y(x, y)$: observed image intensity
- $B(x, y)$: bias of field image intensity
- $U(x, y)$: Partition matrix containing fuzzy membership values at every pixel
- (v_1, v_2) : centroids of class

Partition matrix calculation using the formula:

$$U_k(x, y) = \frac{1}{\sum_{l=1}^2 \left(\frac{\|Y(x, y) - B(x, y) - v_l\|}{\|Y(x, y) - B(x, y) - v_k\|} \right)^{\frac{2}{p-1}}} \quad (9)$$

The formula calculates the class centroid:

$$v_k = \frac{\int_{\Omega} U_k^p(x, y) (Y(x, y) - B(x, y)) dx dy}{\int_{\Omega} U_k^p(x, y) dx dy} \quad (10)$$

Image intensity bias calculations are as follows:

$$B(x, y) = Y(x, y) - \frac{\sum_{k=1}^2 U_k^p(x, y) v_k}{\sum_{k=1}^2 U_k^p(x, y)} \quad (11)$$

2.5 Proposed method

The proposed method is a combination of LBM and fuzzy clustering based on GPU CUDA parallel processing with an algorithm is as follows:

- a) The programming is based on the Matlab program by Balla-Arabé and Gao [19].
- b) Read the image and convert it into RGB, initialize the level set using the circle signed distance function.
- c) Initialize the equilibrium particle distribution functions using equation (4).
- d) Start the iterative calculation
- e) The iteration calculation process starts:
 - i. Compute Partition Matrix (Fuzzy membership) using equation (9).
 - ii. Compute class centroids using equation (10).
- f) Compute Bias image intensity using equation (11).
- g) Compute external force function equation (8).
- h) Compute the equilibrium condition (equation 4) and collision step (equation 7a).
- i) Compute streaming step using equation (7b).
- j) Compute the level set using equation (6).
- k) If the segmentation is not done, then go back to step 4.

In this study, the authors use medical images because the authors want to help the medical world accelerate segmentation results and the previous method where previous researchers used non-medical images to assess segmentation results.

In this study, the authors use GPU and CPU. The most significant difference between GPU and CPU is in the architecture and how they work. The GPU architecture consists of many threads so that they can work in parallel. In contrast, the architecture on the CPU has many cores or fewer cores than the GPU how the CPU works in serial. For this reason, even though GPU and CPU programming are used simultaneously, programming with GPU is in parallel. When programming CPU serially. So there is a difference in coding at the time of the computing process.

3 Results and discussion

Performance study was conducted using various resolutions of the medical images, i.e., 512×512, 1024×1024, 1536×1536, 2048×2048, 2560×2560 dan 4096×4096 pixels. All calculations using a two-dimensional block, in which the number of threads per block is 16×16. The GPU used the global memory for all calculations, except for the class centroids calculation, which used shared memory. The performance study consists of 3 parts. First, it compares the contour of the segmentation results between the CPU serial programming and the GPU parallel programming segmentation contour. The sec-

ond part compares the elapsed time of segmentation processes between CPU serial programming and GPU parallel programming. While the third part compares the results of the segmentation process using parallel GPUs with the results of the segmentation process by Arbeláez (2006) [32], Balla-Arabé and Gao (2013) [19], and segmentation by a human. Arbeláez and human segmentation are taken from the BSDS300 dataset. The segmentation process is carried out using the Lattice Boltzmann and Fuzzy Clustering methods on the CPU and GPU. The segmentation results produce the same segmentation results, but the execution time on the CPU and GPU is different. To assess whether the researcher's segmentation process is correct, the researcher compares the segmentation results with the previous methods.

Figures 5 - 10 show that the image segmentation contours with the serial CPU and parallel GPU programs for various resolutions produced in this study resemble the image segmentation contours generated by the comparison data mentioned above. From these figures, it is clear that the segmentation results between the two programming methods show no difference. The result fits to compare the contours of the segmentation results of this study with the benchmark data is to evaluate the level of similarity of the segmentation results obtained in this study with previous studies.

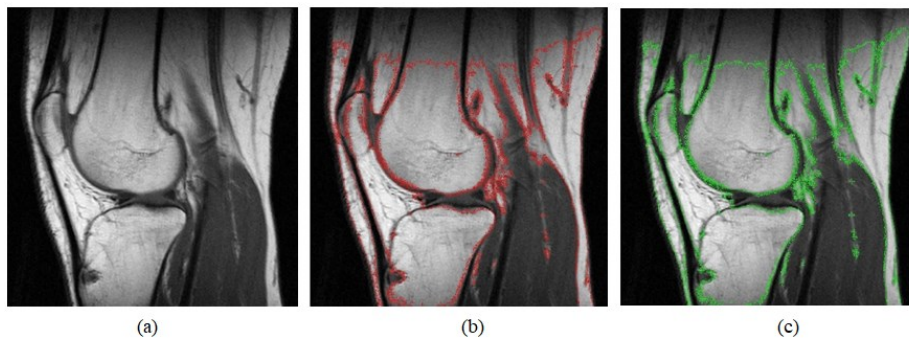


Fig. 5. The segmentation results for medical images with a size of 512×512 : (a) the original image, (b) segmentation result contour with CPU, and (c) segmentation result contour with GPU.

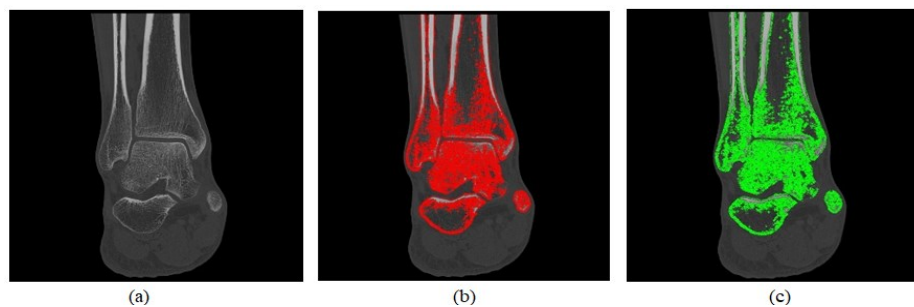


Fig. 6. The segmentation results for medical images with a size of 1024×1024 : (a) the original image, (b) segmentation result contour with CPU, and (c) segmentation result contour with GPU.

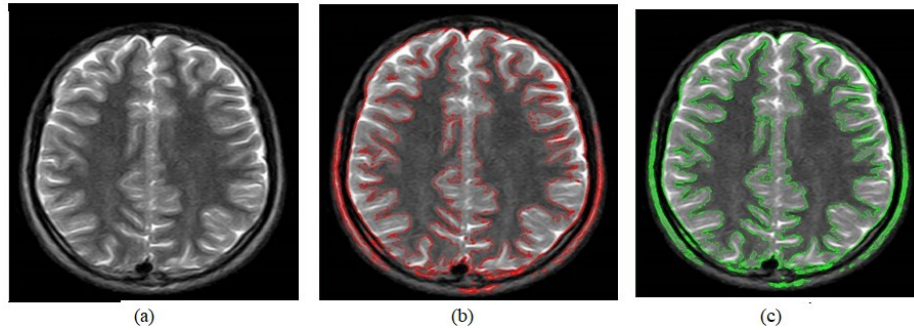


Fig. 7. The segmentation results for medical images with a size of 1536×1536 : (a) the original image, (b) segmentation result contour with CPU, and (c) segmentation result contour with GPU.

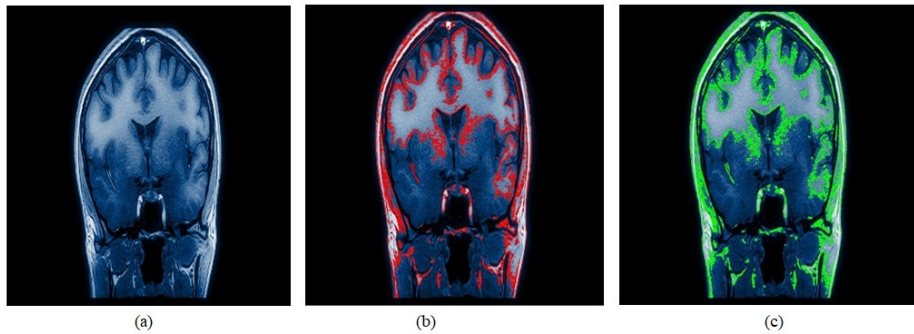


Fig. 8. The segmentation results for medical images with a size of 2048×2048 : (a) the original image, (b) segmentation result contour with CPU, and (c) segmentation result contour with GPU.

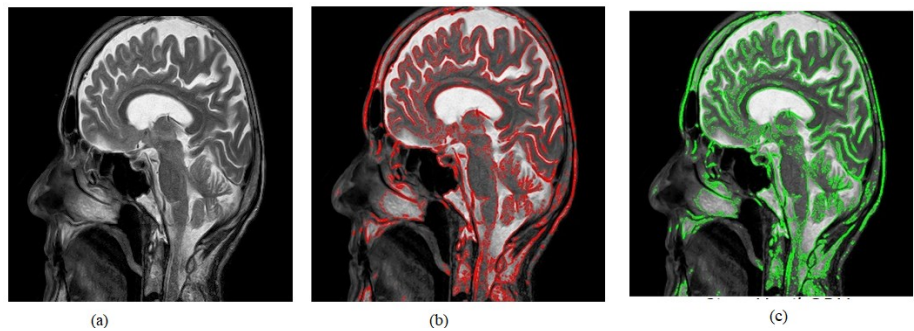


Fig. 9. The segmentation results for medical images with a size of 2560×2560 : (a) the original image, (b) segmentation result contour with CPU, and (c) segmentation result contour with GPU.

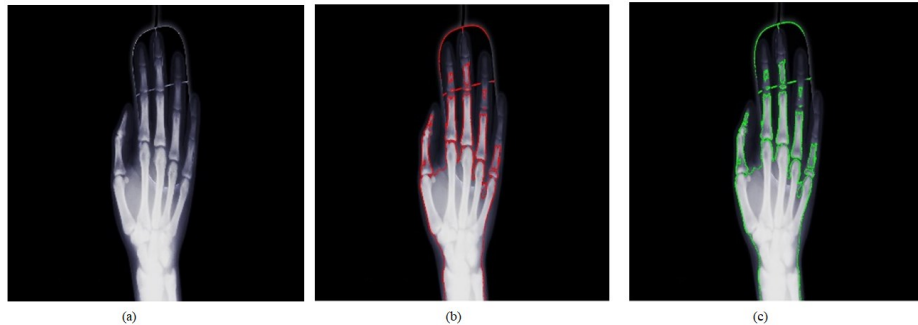


Fig. 10. The segmentation results for medical images with a size of 4096×4096 : (a) the original image, (b) segmentation result contour with CPU, and (c) segmentation result contour with GPU.

Table 1 shows the elapsed time for the serial and parallel segmentation processes. Based on this table, it can be seen that the use of parallel programming with GPU CUDA can increase the speed significantly, being able to be 39.22 times faster than serial programming with a CPU. The results of this study are the results of Kalaiselvi et al. [26] and Chang et al. [33]. Improving image resolution also increases the execution time, but the speedup also increases. The increase in speedup experienced a slight slowdown for the image resolution of 2560×2560 , but it went up again after that. The growth is because the GPU works optimally if the image resolution is a power of 2.

Table 1. Comparison results of elapsed time

Image Resolution	Elapsed time (s)		Speedup (×)
	Serial CPU	Parallel GPU	
512×512	0.174833	0.005787	30.21
1024×1024	0.685656	0.020438	33.54
1536×1536	1.530913	0.043604	35.10
2048×2048	2.787990	0.076732	36.33
2560×2560	4.281307	0.118423	36.15
4096×4096	11.345838	0.289226	39.22

This section compares the contour of the segmentation results obtained in this study with the results of previous methods, namely: the method proposed by Arbelaez (2006) [32], Balla-Arabé, and Gao (2013) [19], and the contour segmentation performed by humans. The image used in this study follows the results of segmentation by humans taken from the Berkeley segmentation data set BSDS300. The original images (481×321 and 321×481) were then converted to 256×256 . The comparison results obtained are shown in Table 2. The comparison indicates that the image segmentation contour produced in this study resembles the image segmentation contour produced by the comparative data mentioned above.

These results agree with Bobbia, S. et al. (2021) with their paper entitled Iterative Boundaries Implicit Identification for Superpixel Segmentation: Real-Time Approach.

The author proposes an optimal iteration method to reduce the computation time further. This optimization allowed the authors to accelerate segmentation by a factor of 4 on the Jetson TX1 embedded GPU platform [34].

Table 2. The comparison results

	Test image	Arbelaez (2006) [32]	Balla-Arabé and Gao (2013) [19]	Proposed method	Humans (BSDS300 data set)
1					
2					
3					
4					
5					
6					

The contour of segmentation results: (a) the original image, (b) segmentation results from the Arbelaez POCV2006 method (2006), (c) segmentation results from the FRLBM method (2013), (d) segmentation of the results of this study, and (e) segmentation by humans.

4 Conclusion

The conclusions that can be drawn from this research are:

1. Combination with the Fuzzy clustering method can reduce the number of iterations in the calculation of the segmentation process
2. GPU CUDA-based parallel programming can significantly speed up the calculation time compared to serial programming with the CPU. In this research, parallel programming with GPU CUDA can speed up 39.22 times.
3. The segmentation process for images with a size of more than 2000×2000 pixels can be handled easily.
4. The comparison of the results of the segmentation process with benchmark data has good similarities.

5 Acknowledgment

This research was funded by grants from the Research and Community Engagement Directorate (DRPM) Ministry of Research, Technology and Higher Education of the Republic of Indonesia under the Research of Master Thesis Schemes (No. 806.5/LL5/PG/2020) and Universitas Atma Jaya Yogyakarta.

6 References

- [1] K. Siemionow, C. Luciano, C. Forsthoefel, and S. Aydogmus, "Autonomous image segmentation and identification of anatomical landmarks from lumbar spine intraoperative computed tomography scans using machine learning: A validation study," *J. Craniovertebr. Junction Spine*, vol. 11, no. 2, pp. 99–103, 2020, <https://doi.org/10.4103/jcvjs.jcvjs.37.20>
- [2] X. Chen *et al.*, "Anatomy-Regularized Representation Learning for Cross-Modality Medical Image Segmentation," *IEEE Trans. Med. Imaging*, vol. 40, no. 1, pp. 274–285, 2021, <https://doi.org/10.1109/tmi.2020.3025133>
- [3] C. Y. Chang, P. C. Chung, Y. C. Hong, and C. H. Tseng, "A neural network for thyroid segmentation and volume estimation in CT images," *IEEE Comput. Intell. Mag.*, vol. 6, no. 4, pp. 43–55, 2011, <https://doi.org/10.1109/mci.2011.942756>
- [4] S. S. Chouhan, A. Kaul, and U. P. Singh, *Soft computing approaches for image segmentation: a survey*, vol. 77, no. 21. Multimedia Tools and Applications, 2018. <https://doi.org/10.1007/s11042-018-6005-6>
- [5] H. S. Goushchi and Y. Pourasad, "Clustering of Brain Tumors in Brain MRI Images Based on Extraction of Textural and Statistical Features," *Int. J. online Biomed. Eng.*, vol. 16, no. 12, pp. 116–132, 2020, <https://doi.org/10.3991/ijoe.v16i12.16929>
- [6] A. Khosravanian, M. Rahmanimanesh, P. Keshavarzi, and S. Mozaffari, "Fast level set method for glioma brain tumor segmentation based on Superpixel fuzzy clustering and lattice Boltzmann method," *Comput. Methods Programs Biomed.*, vol. 198, p. 105809, 2021, <https://doi.org/10.1016/j.cmpb.2020.105809>
- [7] Y. Zhong, J. He, C. Zhang, and B. Ardlee, "Treatment of Differentiated Thyroid Cancer and Recurrent Laryngeal Nerve Function with 131 Iodine Based on Positron Emission

- Tomography/Computed Tomography Image Segmentation Algorithm," *World Neurosurg.*, vol. 149, pp. 428–435, 2021, <https://doi.org/10.1016/j.wneu.2020.10.160>
- [8] Y. Sun *et al.*, "Accurate and rapid CT image segmentation of the eyes and surrounding organs for precise radiotherapy," *Med. Phys.*, vol. 46, no. 5, pp. 2214–2222, 2019, <https://doi.org/10.1002/mp.13463>
- [9] C. Kalyani, K. Ramudu, and G. R. Reddy, "Enhancement and Segmentation of Medical Images Using AGCWD and ORACM," *Int. J. online Biomed. Eng.*, vol. 16, no. 13, pp. 45–56, 2020, <https://doi.org/10.3991/ijoe.v16i13.18501>
- [10] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations," *J. Comput. Phys.*, vol. 79, no. 1, pp. 12–49, 1988, [https://doi.org/10.1016/0021-9991\(88\)90002-2](https://doi.org/10.1016/0021-9991(88)90002-2)
- [11] M. S. Choudhry and R. Kapoor, "A novel fuzzy energy based level set method for medical image segmentation," *Cogent Eng.*, vol. 5, no. 1, pp. 1–18, 2018, <https://doi.org/10.1080/23311916.2018.1475032>
- [12] A. K. Jilani Saudagar, "Biomedical Image Compression Techniques for Clinical Image Processing," *Int. J. online Biomed. Eng.*, vol. 16, no. 12, pp. 133–154, 2020, <https://doi.org/10.3991/ijoe.v16i12.17019>
- [13] Z. Wang, B. Ma, and Y. Zhu, "Review of Level Set in Image Segmentation," *Arch. Comput. Methods Eng.*, no. 0123456789, 2020, <https://doi.org/10.1007/s11831-020-09463-9>
- [14] D. Cheng, F. Tian, L. Liu, X. Liu, and Y. Jin, "Image segmentation based on multi-region multi-scale local binary fitting and Kullback–Leibler divergence," *Signal, Image Video Process.*, vol. 12, no. 5, pp. 895–903, 2018, <https://doi.org/10.1007/s11760-017-1234-0>
- [15] C. Wang and H. Hu, "Fuzzy controller based unequal clustering algorithm with fault tolerance," *Int. J. Online Eng.*, vol. 13, no. 12, pp. 174–191, 2017, <https://doi.org/10.3991/ijoe.v13i12.7263>
- [16] L. Li, L. Sun, W. Kang, J. Guo, C. Han, and S. Li, "Fuzzy Multilevel Image Thresholding Based on Modified Discrete Grey Wolf Optimizer and Local Information Aggregation," *IEEE Access*, vol. 4, no. c, pp. 6438–6450, 2016, <https://doi.org/10.1109/access.2016.2613940>
- [17] M. Alahyane, A. Hakim, A. Laghrib, and S. Raghay, "A lattice Boltzmann method applied to the fluid image registration," *Appl. Math. Comput.*, vol. 349, pp. 421–438, 2019, <https://doi.org/10.1016/j.amc.2018.12.051>
- [18] S. Balla-Arabé and X. Gao, "Image multi-thresholding by combining the lattice Boltzmann model and a localized level set algorithm," *Neurocomputing*, vol. 93, pp. 106–114, 2012, <https://doi.org/10.1016/j.neucom.2012.04.019>
- [19] S. Balla-Arabé, X. Gao, and B. Wang, "A fast and robust level set method for image segmentation using fuzzy clustering and lattice boltzmann method," *IEEE Trans. Cybern.*, vol. 43, no. 3, pp. 910–920, 2013, <https://doi.org/10.1109/tsmcb.2012.2218233>
- [20] J. Wen, Z. Yan, and J. Jiang, "Novel lattice Boltzmann method based on integrated edge and region information for medical image segmentation," *Biomed. Mater. Eng.*, vol. 24, no. 1, pp. 1247–1252, 2014, <https://doi.org/10.3233/bme-130926>
- [21] J. Chen, Z. Chai, B. Shi, and W. Zhang, "Lattice Boltzmann method for filtering and contour detection of the natural images," *Comput. Math. with Appl.*, vol. 68, no. 3, pp. 257–268, 2014, <https://doi.org/10.1016/j.camwa.2014.05.023>
- [22] D. Wang, "A Fast Hybrid Level Set Model for Image Segmentation Using Lattice Boltzmann Method and Sparse Field Constraint," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 32, no. 6, pp. 1–24, 2018, <https://doi.org/10.1142/s0218001418540150>

- [23] M. N. Akhtar, J. Mohamad Saleh, and C. Grelck, "Parallel Processing of Image Segmentation Data Using Hadoop," *Int. J. Integr. Eng.*, vol. 10, no. 1, pp. 74–84, 2018, <https://doi.org/10.30880/ijie.2018.10.01.012>
- [24] J. Roels, J. De Vylder, Y. Saeys, B. Goossens, and W. Philips, "Decreasing time consumption of microscopy image segmentation through parallel processing on the GPU," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10016 LNCS, pp. 147–159, 2016, https://doi.org/10.1007/978-3-319-48680-2_14
- [25] R. Kumar *et al.*, "Histogram Thresholding in Image Segmentation: A Joint Level Set Method and Lattice Boltzmann Method Based Approach," in *Advances in Intelligent Systems and Computing* 455, 2017, pp. 529–539, https://doi.org/10.1007/978-3-319-38771-0_52
- [26] T. Kalaiselvi, P. Sriramakrishnan, and K. Somasundaram, "Performance of medical image processing algorithms implemented in CUDA running on GPU based machine," *Int. J. Intell. Syst. Appl.*, vol. 10, no. 1, pp. 58–68, 2018, <https://doi.org/10.5815/ijisa.2018.01.07>
- [27] H. Huang and Z. Li, "Real-time image transmission algorithm in WSN with limited bandwidth," *Int. J. Online Eng.*, vol. 14, no. 12, pp. 57–71, 2018, <https://doi.org/10.3991/ijoe.v14i12.9494>
- [28] A. Hagan and Y. Zhao, "Parallel 3D image segmentation of large data sets on a GPU cluster," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5876 LNCS, no. PART 2, pp. 960–969, 2009, https://doi.org/10.1007/978-3-642-10520-3_92
- [29] M. Espinoza-Andaluz, A. Moyón, and M. Andersson, "A comparative study between D2Q9 and D2Q5 lattice Boltzmann scheme for mass transport phenomena in porous media," *Comput. Math. with Appl.*, vol. 78, no. 9, pp. 2886–2896, 2019, <https://doi.org/10.1016/j.camwa.2019.02.012>
- [30] P. L. Bhatnagar, E. P. Gross, and M. Krook, "A Model for Collision Processes in Gases. I. Small Amplitude Processes in Charged and Neutral One-Component Systems," *Phys. Rev.*, vol. 94, no. 3, pp. 511–525, 1954, <https://doi.org/10.1103/physrev.94.511>
- [31] M. De Florio, E. Schiassi, B. D. Ganapol, and R. Furfaro, "Physics-informed neural networks for rarefied-gas dynamics: Thermal creep flow in the Bhatnagar-Gross-Krook approximation," *Phys. Fluids*, vol. 33, no. 4, 2021, <https://doi.org/10.1063/5.0046181>
- [32] P. Arbeláez, "Boundary extraction in natural images using ultrametric contour maps," *2006 Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2006, 2006, <https://doi.org/10.1109/cvprw.2006.48>
- [33] H. H. Chang, Y. J. Lin, and A. H. Zhuang, "An Automatic Parameter Decision System of Bilateral Filtering with GPU-Based Acceleration for Brain MR Images," *J. Digit. Imaging*, vol. 32, no. 1, pp. 148–161, 2019, <https://doi.org/10.1007/s10278-018-0110-y>
- [34] S. Bobbia, R. Macwan, Y. Benezeth, K. Nakamura, and J. Dubois, "Iterative Boundaries Implicit Identification for Superpixels Segmentation: A Real-Time Approach," *IEEE Access*, vol. 9, pp. 77250–77263, 2021, <https://doi.org/10.1109/access.2021.3081919>

7 Authors

Ignasius Boli Suban received his Master degree in the Informatics Department, Universitas Atma Jaya Yogyakarta. His interests are Mobile Applications, Gamification, Computer Networking, Artificial Intelligence, and the Internet of Things (Email: ignasiusbolisuban@gmail.com).

Suyoto is Professor in Informatics Department at Universitas Atma Jaya Yogyakarta, Yogyakarta, Indonesia. He got a Ph.D. in 2000 from the National University of Malaysia, Malaysia. His research interests are multimedia, computer graphics, visualization, mobile application, and artificial intelligence.

Pranowo is currently a senior lecturer in Informatics Department at Universitas Atma Jaya Yogyakarta, Yogyakarta, Indonesia. He got a Ph.D. degree from the Department of Electrical Engineering and Information Technology, Gadjah Mada University, in 2010. His research interests include numerical simulation and modeling, computer vision, and parallel programming based on GPU (Email: pranowo@uajy.ac.id).

Article submitted 2021-06-01. Resubmitted 2021-07-21. Final acceptance 2021-08-04. Final version published as submitted by the authors.