ORIGINAL

Accelerated organ region segmentation by the revised radial basis function network using a graphics processing unit

Takeshi Konishi[1], Tadashi Kondo[2], Hiroki Moriguchi[3], Masato Tagi[1], and Jun Hirose[1]

[1]*Department of Medical Informatics, Institute of Biomedical Sciences, Tokushima University Graduate School, Tokushima, Japan,* [2]*Department of Medical Image Informatics, Graduate School of Health Sciences, Tokushima University, Tokushima, Japan,* [3]*Uchida Neurosurgery Clinic, Kochi, Japan*

Abstract : This study aimed to accelerate the segmentation of organs in medical imaging with the revised radial basis function (RBF) network, using a graphics processing unit (GPU). We segmented the lung and liver regions from 250 chest x-ray computed tomography (CT) images and 160 abdominal CT images, respectively, using the revised RBF network. We compared the time taken to segment images and their accuracy between serial processing by a single-core central processing unit (CPU), parallel processing using four CPU cores, and GPU processing. Segmentation times for lung and liver organ regions shortened to 57.80 and 35.35 seconds for CPU parallel processing and 20.16 and 11.02 seconds for GPU processing, compared to 211.03 and 124.21 seconds for CPU serial processing, respectively. The concordance rate of the segmented region to the normal region in slices excluding the upper and lower ends (173 lung and 111 liver slices) was 98% for lung and 96% for liver. The use of CPU parallel processing and GPU shortened the organ segmentation time in the revised RBF network without compromising segmentation accuracy. In particular, segmentation time was shortened to less than 10% with GPU. This processing method will contribute to workload reduction in imaging analysis. J. Med. Invest. 66 : 86-92, February, 2019

Keywords : *RBF networks, GPGPU, organ segmentation*

INTRODUCTION

With the advancement of medical imaging devices in recent years, several hundred images are obtained for each subject in computed tomography (CT) examinations. Radiologists must review all images taken during an examination in detail for the organs and lesions of interest, but in order to prevent misdiagnoses or missed diagnoses, they must also review the images that capture non-target organs and lesions, thereby increasing the workload. Therefore, workload reduction in image interpretation is an important issue.

Computer aided diagnosis (CAD), in which results of quantitative analysis by computers are used to aid in diagnosis, is one method of reducing the workload in radiologic interpretation (1) and has already been in the stages of practical use in imaging diagnosis for lung cancer (2). CAD is easily applicable in the lung, since the density difference in CT value compared to adjacent organs is relatively large, and a lung segmentation method using threshold values (3), as well as open source software (4), have been developed. In contrast, liver segmentation using thresholds alone is difficult, as the density difference in CT value between the liver and its adjacent organs and blood vessels is small. Thus, methods using level sets (5) and artificial neural networks (6) are being tested.

Artificial neural networks have attracted attention in recent years. In these systems, the process is divided into two stages, i.e., learning and prediction, but both stages have the drawback of being time-consuming. The learning stage has many hyperparameters, which require manual adjustment, necessitating repeated actions until an optimal value is found ; thus, it takes a long time before the learning is complete. To overcome this disadvantage, the utility of the radial basis function (RBF) network, a type of artificial neural network, has been reported (7). In this method, since the weight can be obtained in a single action by solving simultaneous linear equations, the learning time is short, and there are only three hyperparameters. In addition, by performing predictions in advance using the least squares method, the revised RBF network, which only requires one hyperparameter for the repeated computations for finding the optimal value, was developed (8). This method successfully shortened the adjustment time required to find the optimal value for liver region segmentation, while automatically determining the hyperparameter and simultaneously improving accuracy. The revised RBF network is advantageous in that it can be applied to not only the liver but also to other organs, by changing the learning data. However, the prediction time, which is the other disadvantage of artificial neural networks, could not be shortened even with the RBF network, and the revised RBF network has an even longer prediction time, since parameters are predicted in advance using the least squares method.

The purpose of the present study was to accelerate the organ region segmentation process in the revised RBF network, with the aim of shortening the prediction time of the artificial neural network. To that end, we compared organ region segmentation by prediction processing using the revised RBF network accelerated by parallel processing using a central processing unit (CPU) with multiple cores and a graphics processing unit (GPU), which have been successfully applied in artificial neural networks (9, 10).

Received for publication June 26, 2018 ; accepted September 9, 2018.

Address correspondence and reprint requests to Takeshi Konishi, Japan and Fax : +81-88-625-8104.

MATERIALS AND METHODS

System structure

We used a personal computer (PC) with the specifications indicated in Table 1 and created an application for image segmentation and evaluation using the C++ programming language. To increase speed, we used four parallel CPU cores and GPU. For parallel processing using CPU cores, four CPU cores were used in this study, thus the thread number was four, and each core was assigned a thread to execute, so that all four cores would be used. The number of cores in GPU was 2,884, and its parallel capability was higher than the four cores in CPU. The number of floating-point operations per second (FLOPS) was 5,345 giga FLOPS (GFLOPS), which was faster than the 177 GFLOPS for CPU (Table 2). For the GPU program, we used the Compute Unified Device Architecture (CUDA) developed by NVIDIA (Santa Clara, USA) (11). In CUDA, threads are organized into a structure consisting of three layers, which are referred to as the thread (bottom layer), block (middle layer), and grid (top layer) (Figure 1). Therefore, in CUDA programming, we set the number of threads within CUDA blocks as 256, and the number of blocks within grids as 1,024, so that the total number of threads becomes 262,144, which enables 512 x 512 pixels, or a total of 262,144 pixels, to be processed at once. In addition, we attempted to accelerate calculation using shared memory accessible at high speed with CUDA architecture.

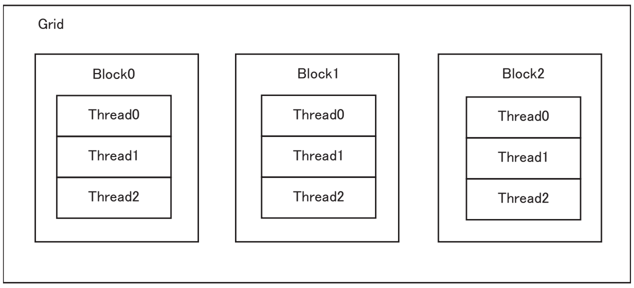

Figure 1. CUDA thread organization
In CUDA, threads are organized into a structure consisting of three layers, which are referred to as the thread (bottom layer), block (middle layer), and grid (top layer).

Organ region segmentation

The RBF network is an artificial neural network composed of three layers (8). The first layer consists of input variables (A), the second layer is the middle layer which inputs the output from the first layer into a radial basis function (B), and the third layer, the last layer, is a linear combination of the middle layer output and weights (C). The square difference between the output from the final layer and the training signal is defined as evaluation function E (D), and RBF network learning is performed by solving for the weight that minimizes E. In the revised RBF network, equation (B) was set as equation (E), and by substituting the linear function (F) determined from the regression coefficients a_0 and a_1 estimated from the least squares method, the center c, which was the adjusting item, and standard deviation r can automatically be estimated. With these steps, the standard deviation r, center c, and the regularization parameter λ, which were manually adjusted items in the RBF network, become only the regularization parameter λ in the revised RBF network. We performed segmentation of organ regions using this revised RBF network.

Table 1. Personal computer specification

CPU	i7-4770k
Memory	16G
OS	Windows8.1 x64
GPU	GTX 780Ti

CPU, central processing unit ; OS, operating system ; GPU, graphics processing unit

Table 2. Specifications of the central processing unit (CPU) and the graphics processing unit (GPU)

Model number/Specification	CPU	GPU
Model number	i7-4770K	GTX 780Ti
Processor core clock (MHz)	3,500	875
ALU clock (MHz)	3,900	928
Memory size (MB)	16,384	3,072
Bandwidth core memory (GB/s)	25.6	336
Number of processor cores	4	2,880
Total SP FLOPS peak performance (GFLOPS)	177	5,345
Thermal Design Power (watt)	84	250

ALU, arithmetic logic unit ; SP, streaming processor ; FLOPS, floating-point operations per second.

$$x = (x_1 \;, ..., x_n) \tag{A}$$

$$h_j\,(x) = exp\left(-\frac{\| x - c_j \|^2}{r_j^2}\right) \tag{B}$$

where c is the center position, and r is the standard deviation,

$$O_j\,(x) = \textstyle\sum_{j=1}^{m} w_j\, h_j\,(x) \tag{C}$$

where w is the weight, and m is the middle layer element number,

$$E = \frac{1}{2}\textstyle\sum_{i=1}^{p}(y_i - O\,(x_i))^2 + \frac{1}{2}\lambda\, \sum_{j=1}^{m} w_j^2 \tag{D}$$

where y is the training data, $\frac{1}{2}\lambda\, \sum_{j=1}^{m} w_j^2$ is the regularization item, λ is the regularization parameter, and p is the number of learning data,

$$h_j\,(x) = exp\,(-z_j^2) \tag{E}$$

$$z_j \;= a_0 + a_1\, d_j \tag{F}$$

where a is the regression parameter, and d is the distance between the training data and the center,

$$d_j = \| \, u - c \, \| \tag{G}$$

where u is the training data, and c is the center.
w that minimizes E is determined using the following equation (H) (8).

$$w = (H^T + A\,)^{-1} H^T \tag{H}$$

where

$$\mathrm{H} = \begin{bmatrix} h_1\,(x_1) & h_2\,(x_1) & \cdots & h_m\,(x_j) \\ h_1\,(x_2) & h_2\,(x_2) & \cdots & h_m\,(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ h_1\,(x_p) & h_2\,(x_p) & \cdots & h_m\,(x_p) \end{bmatrix} \tag{I}$$

and where

$$A = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_m \end{bmatrix} \quad \text{(J)}$$

Targeted images were 250 chest CT images of a lung cancer patient and 160 abdominal CT images in the early-contrast phase of a liver cancer patient. Each CT image had a slice thickness of 1 mm. First, for each organ, we chose the CT image slice in which the organ occupied the largest area, and used this slice as the training data to perform organ region learning for the revised RBF network. In each organ of interest, we marked learning points both inside and outside the organ, and marked the cancerous region and blood vessels within the organ as being outside the organ (Figures 2, 3). We set the input parameters for the revised RBF network as the density distribution, mean density, standard deviation, x-axis, y-axis, and density range in the 5 x 5 pixels centralized around the learning points, and set the number of learning points as the number of middle layers. Next, we performed organ region segmentation, including normal and cancer regions, in all CT images, and in order to improve accuracy, we performed perimeter processing to remove isolated points, and closing processing to expand and fill in cancer regions and blood vessel regions. In order to evaluate the segmentation results, we used the aforementioned program, visually confirmed the concordance of overlapping pixels, and adjusted learning points to eliminate non-concordance. Moreover, in order to avoid overlearning, in which only the training data match, we adjusted the regularization parameter λ. Once adjustments of learning points and the regularization parameter λ no longer led to improvement, we evaluated the concordance rate of all CT images.

Evaluation

We investigated the organ segmentation time and accuracy of CPU serial processing (CPU serial), CPU four-core parallel processing (CPU parallel), and GPU processing (GPU) in using the training data on each organ image. In CPU serial, we performed organ segmentation by serially processing images one at a time using a single thread (Figure 4a). In CPU parallel, four images were processed at a time in parallel to segment organs (Figure 4b). In GPU, images were processed one at a time serially in a single thread, but the organ segmentation method differed from that of CPU processing in that organ data were transferred to the GPU side to perform organ segmentation (Figure 4c). In all three methods, measurement codes were embedded into the program code, and organ segmentation times were measured 10 times, and the mean value was computed (12). Organ segmentation accuracy was evaluated using the dice similarity coefficient (DSC), which is computed as $2(|A \cap B|)/(|A| + |B|)$, where the correct region manually marked using ImageJ (NIH, Maryland, USA) was designated as A, and the predicted segmentation region was designated as B, in all CT images (13, 14). In both lung and liver, since the DSC decreases dramatically at the upper and lower ends, we separated the upper and lower portions with DSC < 0.9 from the central part. We selected the slices in which each organ occupied the largest area, and by using these two slices along with the central slice as the training data in the interactive method (15), we evaluated the segmentation accuracy of organ regions and compared the results to that in which only the central slice with the largest organ area was used as the training data.

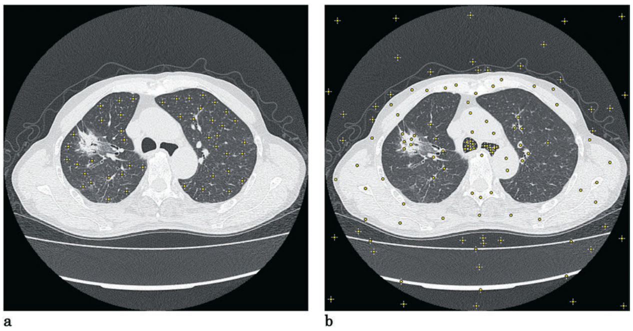

Figure 2. Learning points in lung
a. We marked learning points with + (darker area) and ○ (lighter area) inside the lung region. b. We marked learning points with + (darker area) and ○ (lighter area) outside the lung region.

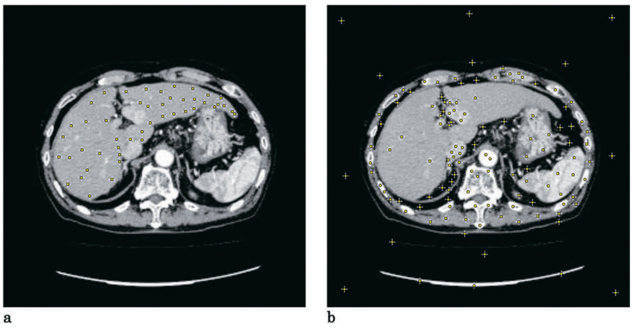

Figure 3. Learning points in liver
a. We marked learning points with + (darker area) and ○ (lighter area) inside the liver region. b. We marked learning points with + (darker area) and ○ (lighter area) outside the liver region.

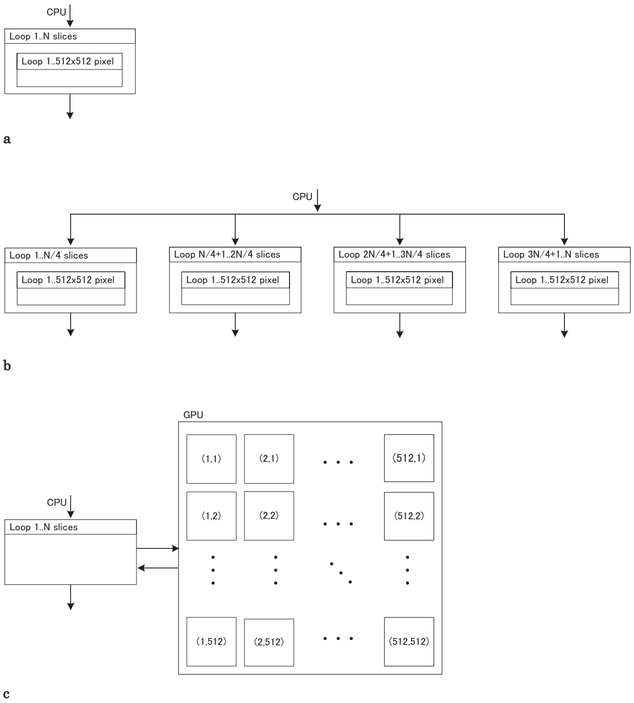

Figure 4. Flow of CPU serial, CPU parallel, and GPU processing
a. In CPU serial, CT slices and slice pixels are processed sequentially. b. In CPU parallel, CT slices are processed in parallel, and slice pixels are processed sequentially. c. In GPU, CT slices are processed sequentially by CPU, and slice pixels are processed in parallel by GPU.

RESULTS

The organ regions segmented by lung CT images are shown in Figure 5, and segmented organ regions of the liver are shown in Figure 6. When the segmented region and correct region were adjusted to match as much as possible by gross visualization, the regularization parameter became 0.003 for both lung and liver. The learning time was 0.27 seconds for lung regions and 0.25 seconds for liver regions. Mean segmentation times for lung regions were 211.03 seconds for CPU serial, 57.80 seconds for CPU parallel, and 20.16 seconds for GPU (Table 3). Mean segmentation times for liver regions were 124.21 seconds for CPU serial, 35.35 seconds for CPU parallel, and 11.02 seconds for GPU. Lung segmentation regions were the same in all three methods, at 6,514,752 pixels. Similarly, liver segmentation regions were the same in all three methods at 1,932,532 pixels, and the segmentation accuracy was consistent for both organs among the three processing methods. Therefore, evaluation of the segmentation accuracy with DSC was performed only with GPU processing.

In terms of the DSC value of each slice, slices with low DSC values were the upper slices #1-23 and lower slices #197-250 (Figure 7) in the lung, and the upper slices #1-11 and lower slices # 120-160 in the liver (Figure 8). The segmentation accuracy of organ regions in all 250 lung slices was DSC 0.80 when a single slice was used as training data, and DSC 0.87 when three slices were used as training data (Table 4). For central slices #24-196, which excluded the upper slices #1-23 and lower slices #197-250, the DSC was 0.98 regardless of whether the training data consisted of

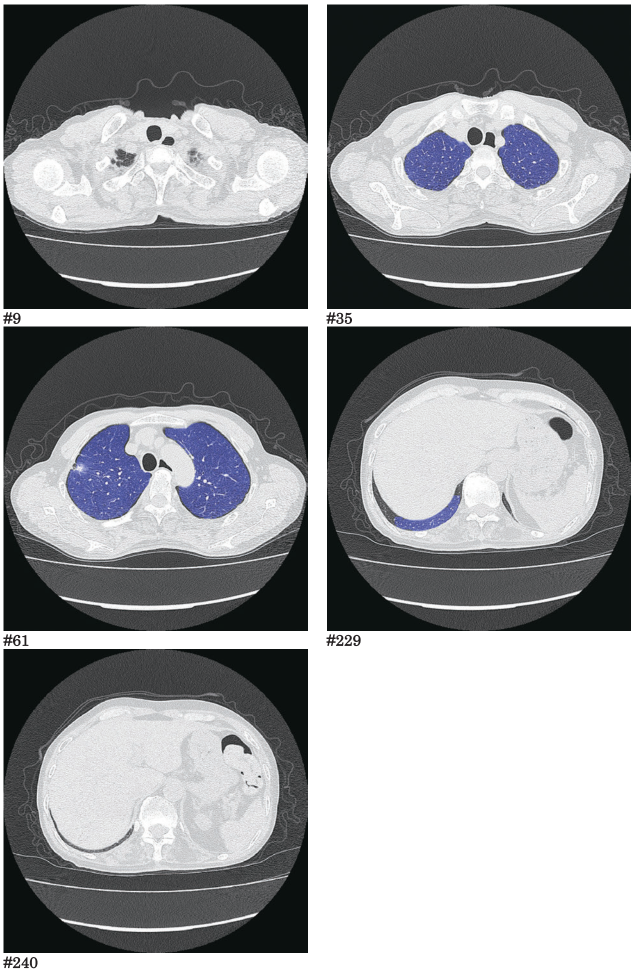

Figure 5. Segmentation image of the lung at each level
The segmented lung region (blue) is superimposed on the original image. # indicates the slice number.

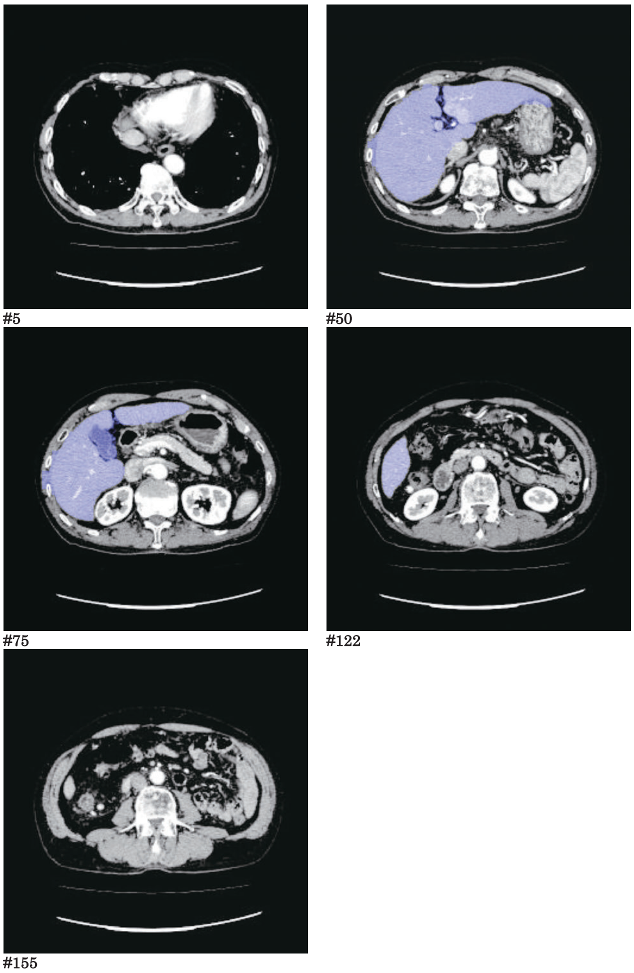

Figure 6. Segmentation image of the liver at each level
The segmented liver region (blue) is superimposed on the original image. # indicates the slice number.

Table 3. Segmentation times for organ regions

Processing method	Lung	Liver
CPU serial (seconds)	211.03 ± 0.95	124.21 ± 0.32
CPU parallel (seconds)	57.80 ± 2.33	35.35 ± 0.24
GPU (seconds)	20.16 ± 0.04	11.02 ± 0.28

CPU, central processing unit ; GPU, graphics processing unit.

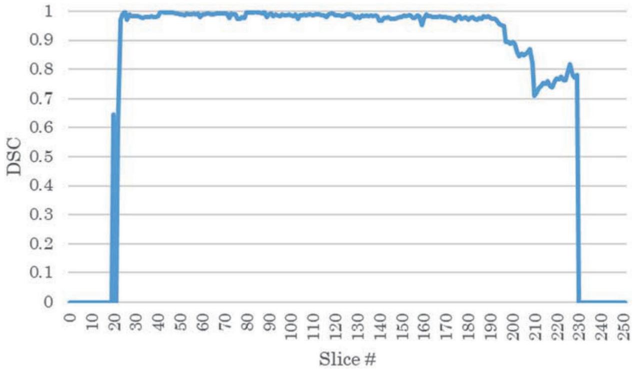

Figure 7. Dice similarity coefficient (DSC) of each lung slice
The DSC of central slices #24–196 was high, with an average of 0.98.

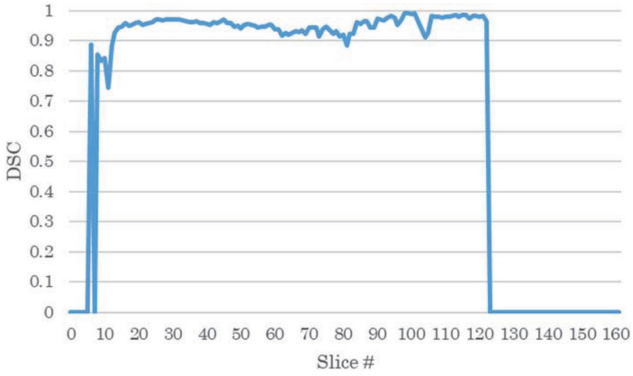

Figure 8. Dice similarity coefficient (DSC) of each liver slice
The DSC of central slices #12–122 was high, with an average of 0.96.

one or three slices. The segmentation accuracy of organ regions in all 160 liver slices was DSC 0.69 when one slice was used as training data, and DSC 0.84 when three slices were used as training data (Table 5). Similar to lung, for central slices #12-122 excluding the upper slices #1-11 and lower slices #123-160, the DSC was 0.96 regardless of whether the training data consisted of one or three slices.

DISCUSSION

To accelerate the revised RBF network, we used either four-core parallel processing with CPU or GPU processing to perform organ region segmentation of the lung and liver. We found that, for both lung and liver, organ region segmentation time shortened to about one-quarter in CPU parallel processing, and to about one-tenth in GPU processing, compared to CPU serial processing. Image segmentation accuracy was the same as that in CPU serial processing for both accelerated methods, regardless of organ, and using three slices for the training data improved the concordance rate between predicted region and correct region.

With regard to artificial neural network prediction, by using GPU processing, we were able to achieve over a 10-fold increase in the speed of the revised RBF network. Regarding learning in a convolutional neural network, which is one of the artificial neural networks, a 6.1-fold increase in speed has been achieved using 8 GPUs in a single machine (9). GPU has a larger number of cores compared to CPU, and since it is accelerated due to its ability to process parallel computations without branches at a high speed, we believe that computations using GPU are very appropriate for the revised RBF network. Moreover, while the shared memory used in GPU programming (11) is shared in the thread within blocks and can be accessed at a fast speed, the available capacity is only 48 KB, and thus, the 512 x 512 pixels cannot all be stored at once. Therefore, in the present study, we copied into the shared memory the 5 x 5 pixel data required in computations for image segmentation, rather than directly referencing in the global memory within GPU, which takes longer to access, thus enabling acceleration.

With regard to image segmentation, CPU serial, CPU parallel, and GPU processing all had the same accuracy. Since the image segmentation processing method is the same in CPU serial and CPU parallel, these two methods have the same organ segmentation accuracy. In contrast, programming affects segmentation accuracy in GPU processing. In the present study, CUDA was programmed using C. In CUDA architecture, since the programming is performed using a C language that is expanded for GPU, similar to the C language called CUDA C, the portability from the program written in the C language is high. Therefore, as a result of programming parameters as single precision floating point num-

Table 4. Dice similarity coefficient (DSC) in lung

Slice #	Sum of pixels of organ region in each slice (Percentage of total slices)	DSC	
		Training data 1 slice	Training data 3 slices
1–250	6,886,460 (100.0)	0.80 ± 0.36	0.87 ± 0.28
1–23	85,115 (1.2)	0.10 ± 0.26	0.59 ± 0.48
24–196	6,302,081 (91.5)	0.98 ± 0.01	0.98 ± 0.01
197–250	499,264 (7.3)	0.49 ± 0.39	0.65 ± 0.37

Table 5. Dice similarity coefficient (DSC) in liver

Slice #	Sum of pixels of organ region in each slice (Percentage of total slices)	DSC	
		Training data 1 slice	Training data 3 slices
1–160	2,058,533 (100.0)	0.69 ± 0.43	0.84 ± 0.26
1–11	27,207 (1.3)	0.38 ± 0.42	0.43 ± 0.39
12–122	1,965,693 (95.5)	0.96 ± 0.02	0.96 ± 0.02
123–160	65,633 (3.2)	0.00 ± 0.00	0.61 ± 0.32

bers, similar to CPU, the image segmentation processing part of the program became the same process as CPU, and the image segmentation accuracy matched that of CPU serial. In other words, we obtained evidence that by using CPU parallel or GPU, segmentation time could be shortened without compromising image segmentation accuracy.

The DSC used for evaluating organ segmentation accuracy is an evaluation scale that should be measured in medical imaging segmentation, and its value ranges from 0-1.00, where the concordance rate is higher the closer the DSC is to 1.00 (14). In the central slices in the present study, we obtained good results, with a lung DSC of 0.98 and liver DSC of 0.96. In contrast, since the occupying areas became extremely small in the upper and lower parts in both organs, when an organ region was segmented using one slice as the training data, the accuracy was low and very inconsistent. Therefore, in order to improve segmentation accuracy, we performed interactive segmentation based on the predicted segmentation regions of each slice. Interactive segmentation is a method that requires some manual operation during the segmentation process, and can use information based on rough sketches of manually outlined organs in each slice, or manually correct missed segmentations or over segmentations of target regions (15). While the workload is greater compared to a fully automated organ segmentation method, the segmentation accuracy is higher. In the present study, when training data were added to slices with a very low DSC value (lung : #23, #197 ; liver : #11, #123), segmentation accuracy increased, and variability decreased.

This study has two limitations. First, we only used one case each of lung and liver for the analysis of training data and test data. Compared with fully automated methods that assess training data and test data from different patients, the interactive method used in this study can more accurately assess training data and test data from the same patient (15). Moreover, our methodology was based on that of Karimi *et al*. (12) (in which measurements of a single subject were obtained 10 times each, and its mean was computed), because the purpose of this study was to examine the acceleration of the revised RBF network. Yet, in terms of image segmentation accuracy, analysis of CT images from multiple cases or healthy individuals may improve the reliability of our results. Second, since the target regions were small in the upper and lower ends of the organs, segmentation accuracy was low in these areas. Despite performing interactive corrections, segmentations could not be performed when the organ region was small, in both lung and liver. Future studies should examine processing methods that can achieve a high accuracy even in these regions.

Since the PC clock number reached its limit in the first half of the year 2000 (16), parallel processing using multiple cores or GPU has already been in use for PC acceleration (9). In the present study, we achieved acceleration of the prediction process of the revised RBF network by GPU, and organ region segmentation was performed without lowering the accuracy compared to CPU serial processing. Organ segmentation accuracy, when slices with DSC $\leq$ 0.9 were removed, was high at 98% for lung and 96% for liver, thus clinical application is realistically possible. With further advancement of imaging studies in clinical practice and development of even more precise image acquisition, the number of images to process will likely continue to increase. We believe our study results will contribute to reducing the workload of medical staff involved in image analysis.

CONFLICT OF INTEREST

The authors have no conflicts of interest.

ACKNOWLEDGEMENTS

The authors thank Professor Junji Ueno (Department of Diagnostic Radiology, Graduate School of Health Sciences, Tokushima University) for providing CT images.

REFERENCES

1. Toriwaki J : Forty Years Progress in Computer Aided Diagnosis of X-ray Images. The journal of the Society of Scientific Photography of Japan 5 : 28-39, 2001
2. Fujita H : Current Status and Future on Developments of Computer-aided Diagnosis Systems for Medical Images. Computer Aided Diagnosis of Medical Images 66 : 484-490, 2003
3. Leader JK, Zheng B, Rogers RM, Sciurba FC, Perez A, Chapman BE, Patel S, Fuhrman CR, Gur D : Automated lung segmentation in X-ray computed tomography : development and evaluation of a heuristic threshold-based scheme. Acad Radiol 10(11) : 1224-1236, 2003
4. Alnaser A, Gong B, Moeller K : Evaluation of open-source software for the lung segmentation. Current Directions in Biomedical Engineering 2 : 515-518, 2016
5. Al-Shaikhli SDS, Yang MY, Rosenhahn B : Automatic 3D Liver Segmentation Using Sparse Representation of Global and Local Image Information via Level Set Formulation 2015. Available at : https : //arxiv.org/abs/1508.01521 Accessed 2017/12/18.
6. Lu F, Wu F, Hu P, Peng Z, Kong D : Automatic 3D liver location and segmentation via convolutional neural networks and graph cut 2016. Avqalilable at : https : //arxiv.org/abs/1605.03012 Accessed 2017/12/18.
7. Orr MJL : Introduction to radial basis function networks. Institute for Adaptive and Neural Computation, Division of Informatics, Edinburgh University 1996. Available at : https : //www.cc.gatech.edu/~isbell/tutorials/rbf-intro.pdf Accessed 2018/4/18.
8. Nakagawa M, Kondo T, Kudo T, Takao T, Ueno J : Three-dimensional medical image recognition of cancer of the liver by a revised radial basis function (RBF) neural network algorithm. Artif Life Robot 14 : 118-122, 2009
9. Krizhevsky A : One weird trick for parallelizing convolutional neural networks 2014. Available at : https : //arxiv.org/abs/1404.5997 Accessed 2018/1/18.
10. NVIDIA Propels Deep Learning with TITAN X, New DIGITS Training System and DevBox : Available at : https : //blogs.nvidia.com/blog/2015/03/17/digits-devbox/ Accessed 2017/11/10
11. NVIDIA, CUDA Programming Guide Version 4.2, NVIDIA Corporation, 2012. Available at : https : //developer.download.nvidia.com/compute/DevZone/docs/html/C/doc/CUDA_C_Programming_Guide.pdf Accessed 2017/12/10.
12. Karimi K, Dickson NG, Hamze F : A Performance Comparison of CUDA and OpenCL. Available at : https : //arxiv.org/abs/1005.2581 Accessed 2018/4/10
13. Farag A, Lu L, Roth HR, Liu J, Turkbey E, Summers RM : Automatic Pancreas Segmentation Using Coarse-to-Fine Superpixel Labeling. In : Lu L, Zheng Y, Carneiro G, Yang L, eds. Deep Learning and Convolutional Neural Networks for Medical Image Computing. Advances in Computer Vision and Pattern Recognition. Springer, Cham, 2017, pp.279-302
14. Yeghiazaryan V, Voiculescu I : An Overview of Current Evaluation Methods Used in Medical Image Segmentation. Technical Report CS-RR-15-08, Department of Computer Science, University of Oxford, Oxford, UK, 2015

15. Heimann T, van Ginneken B, Styner M, Arzhaeva Y, Aurich V, Bauer C, Beck A, Becker C, Beichel R, Bekes G, Bello F, Binnig G, Bischof H, Bornik A, Cashman P, Chi Y, Cordova A, Dawant B, Fidrich M, Furst J, Furukawa D, Grenacher L, Hornegger J, Kainmuller D, Kitney R, Kobatake H, Lamecker H, Lange T, Lee J, Lennon B, Li R, Li S, Meinzer HP, Nemeth G, Raicu D, Rau AM, van Rikxoort E, Rousson M, Rusko L, Saddi K, Schmidt G, Seghers D, Shimizu A, Slagmolen P, Sorantin E, Soza G, Susomboon R, Waite J, Wimmer A, Wolf I : Comparison and evaluation of methods for liver segmentation from ct datasets. IEEE Trans Med Imaging 28(8) : 1251-1265, 2009

16. Danowitz A, Kelley K, Mao J, Stevenson JP, Horowitz M : CPU DB : Recording Microprocessor History. Association for Computing Machinery 2012 Available at : https : //queue.acm.org/detail.cfm?id=2181798 Accessed 2018/01/18.