Proceedings of the 2016 IEEE
International Conference on Robotics and Biomimetics
Qingdao, China, December 3-7, 2016

An Improved VR Training System for Vascular Interventional Surgery *

Shuxiang Guo, Xiaojuan Cai, Baofeng Gao, Yuhua, Jiang

Abstract—**Minimally invasive surgery is a specialized surgical technique that permits vascular interventions through very small incisions. It minimizes the patient's trauma and permits a faster recovery compared to traditional surgery. Although traditional invasive surgery training system can complete general training work, real-time performance and accuracy of most training system failed to meet the requirements of training work. Therefore, in this study, three parts, including 3D modeling, collision detection algorithm and application architecture were improved in the existing training system. Firstly, an improved Marching cubes algorithm was adopted to simplify the mathematical modeling of vessels by merging the related points of the mesh model. Secondly, a hybrid collision detection algorithm was proposed and implemented. Lastly, the CPU-GPU parallel computing architecture was adopted. Particularly, the design of the improved VR-based system and the experimental results were presented and analyzed. Moreover, experimental results showed that the proposed system was beneficial to improve the skill of surgeons in manipulating the catheter and guide wire. Thus, the simulators could be used for trial surgery training.**

I. INTRODUCTION

In recent years, as people's living standard unceasingly enhances, they are more easily affected by some certain diseases. Non-infectious diseases and chronic disease have become the first cause for death in the world's population, which accounts for about 60% of global deaths per year and also are the leading cause of death in China (about 82% of deaths in China) [1]. Chronic disease mainly includes cardiac-cerebral vascular disease, cancer, chronic respiratory diseases and diabetes, among which cardiac-cerebral vascular disease holds the first place. With the shortage of proficient vascular interventional surgeons, some vascular interventional surgeons have to work for long hours in one operation. Nevertheless, novices are not qualified for a real operation until they have participated in a lot of training.

VR-based surgery simulation has been a hot topic in the field of virtual reality in recent years. The use of VR aiming at improving surgeons' knowledge on anatomy and operating skills was first introduced by Satava et al. in 1994 [2]. Ever since then, VR-based surgery simulation has been developed rapidly. And several systems have been proposed to train novices all over the world. These VR-based systems include VR-VIS(BIT, Beijing, China), myringotomy VRT (Western University, London, UK), VR-based Simulator (Kagawa University, Japan) [3,5]. For example, Gao et al. [3] developed a Virtual Reality based Robotic Catheter System. The simulators could generate realistic virtual reality environment of blood vessels according to patient's special computed tomography (CT) or magnetic resonance imaging (MRI), and could carry out the intervention with haptic interfaces with force feedback, which provided the surgeon with a sense of touch. Huang et al. [4] developed a VR-based simulator to simulate all surgical aspects of myrintogomy and tube placement. Wang et al. [5] developed a Virtual reality-based interventional surgery training system to evaluate the published studies assessing VR as a training tool in interventional surgery and to compare virtual reality simulation to no training. However, there are some problems in most existing systems, such as the low real-time performance and long simulation time of the system, which are related to the rate of model rendering, collision detecting efficiency and communication of the training system. Generally, in our previous system, a function called 'StepSimulation' in the Bullet physics engine library was adopted to implement the rendering and redrawing process of vessels, catheter and wire guide models, which would be repeated in every step cycle [6]. Actually, a great part of the vascular model in the surgery training system maintained its anterior form without any change in mathematical and physical model. Accordingly, only a small part of the vessels model, which touched or almost touches the catheter or the guide wire would change in form. Consequently, this method worked with terribly low efficiency, which lead to poor fps (frame per second) in visual rendering and haptic rendering and low fidelity in the simulation.

Aiming at addressing these problems, a VR-based vascular interventional surgery training system was designed and implemented, which provided a cost-effective and efficient way for novices to become experienced vascular interventional surgeons. It surely can be a vascular interventional surgery preoperative simulator, which assured the safety and success rate of a vascular interventional surgery. The two processes, 'StepSimulation' and 'Uncounted', always took about 190 ms and 220 ms respectively, and the VR-based training system had a low fps (2.5 fps), which was difficult to reach expected target in real time. Thus, in the VR-based vascular interventional surgery training system, the following solution was proposed. Parallel multi-thread technology was adopted to accelerate the process of 'StepSimulation' and 'Uncounted'. Specifically, a new class in C++ project should be defined, who inherited all methods of its parent class, helping to extract and override some useful functions from

* This research is partly supported by the National Natural Science Foundation of China (61375094), National High Tech. Research and Development Program of China (No.2015AA043202).

Shuxiang Guo, Xiaojuan Cai, Baofeng Gao are with the Key Laboratory of Convergence Medical Engineering System and Healthcare Technology, the Ministry of Industry and Information Technology, School of Life Science, Beijing Institute of Technology, No.5, Zhongguancun South Street, Haidian District, 100081 Beijing, China (corresponding author to provide phone: 86-010-68915908; e-mail: gaobaofeng@bit.edu.cn).

Yuhua Jiang is with Department of Interventional Neurosurgery, Beijing Neurosurgical Institute and Beijing Tiantan Hospital, China (e-mail: jxy_200321@163.com).

Shuxiang Guo is also with Faculty of Engineering, Kagawa University, 2217-20 Hayashi-cho, Takamatsu, Japan (e-mail: guo@eng.kagawa-u.ac.jp).

rigorously packaged library. Thus, the initially expected goal for real-time simulation could be reached.

Aiming at addressing these problems, a VR-based vascular interventional surgery training system was designed and implemented, which provided a cost-effective and efficient way for novices to become experienced vascular interventional surgeons. It surely can be a vascular interventional surgery preoperative simulator, which assured the safety and success rate of a vascular interventional surgery. The two processes, 'StepSimulation' and 'Uncounted', always took about 190 ms and 220 ms respectively, and the VR-based training system had a low fps (2.5 fps), which was difficult to reach expected target in real time. Thus, in the VR-based vascular interventional surgery training system, the following solution was proposed. Parallel multi-thread technology was adopted to accelerate the process of 'StepSimulation' and 'Uncounted'. Specifically, a new class in C++ project should be defined, who inherited all methods of its parent class, helping to extract and override some useful functions from rigorously packaged library. Thus, the initially expected goal for real-time simulation could be reached.

The rest of this paper is organized as follows. The improvements and implementation to the modelling algorithm will be introduced in Section II. A hybrid collision detection algorithm will be elaborated in Section III. The GPU-CPU architecture will be described in Section IV. Section V and VI will present the experiments, conclusion and future work.

II. An Improved MC Algorithm

In VR-based vascular interventional surgery training system, three-dimensional model of the vessels is an important part which is directly related to the quality of the whole virtual training system. Contrary to the traditional training system, physical attributes should be endowed to the vessel models to simulate the real vessels as much as possible [7, 8]. Marching cubes (MC) algorithm currently is the most widely used method for surface rendering, which was firstly proposed by Lorensen in 1987 [9]. However, there are a huge number of triangles output by MC algorithm, which leads to low interactivity and real-time. Consequently, an improved MC algorithm was proposed to simplify the 3D model of the vessels, which, as a result, reduced the number of triangles of the model and promoted the speed of interaction.

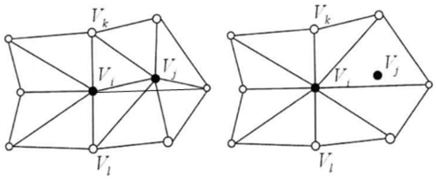


Fig. 1 Vertex merge schematic diagram of the MC algorithm

Fig. 1 shows the reconstructed 3D images by the MC algorithm was composed of triangles. The improved MC algorithm simplified the triangulation by merging two vertexes if the required constraint condition had been met. Accordingly, vertexes who could not meet the required constraint conditions maintained their state. Specifically, this method only changed the shape of certain triangles without any topological changes of the structure of mesh model of the vessels. Therefore, this method would not induce any difference to the rendering effect of the virtual models, which, however, reduced the amount of triangles of the model. There were two constraint conditions for reduction of the triangles in the MC algorithm: 1) the distance between two vertexes; 2) the angle between two vertexes.

The first constraint condition was constraint of distance between two vertexes, which could be expressed as:

$$\mathrm{d}(\mathrm{V}_i, \mathrm{V}_j) \leq d_{max} \tag{1}$$

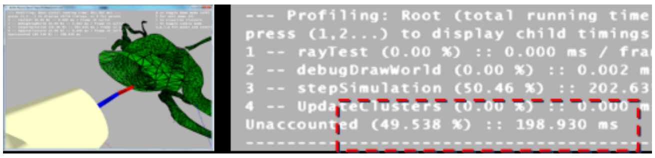
where $\mathrm{d}(\mathrm{V}_i, \mathrm{V}_j)$ was defined to be the distance between vertex V_i and vertex V_j, d_{max} was the predefined maximum distance. They needed to meet to the requirement restricted in the above equation (1).

The second constraint condition was the angle constraints between two vertexes, which could be expressed as:

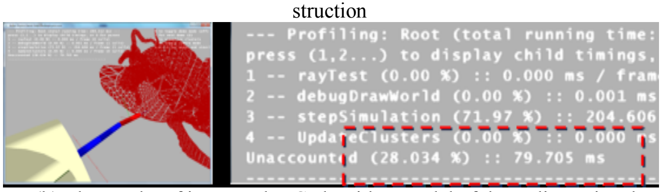
$$\alpha(\mathrm{V}_i, \mathrm{V}_j) \leq \alpha_{max} \tag{2}$$

$$\alpha(\mathrm{V}_i, \mathrm{V}_j) = \cos^{-1}(N_i \times N_j \,/\, |N_i||N_j|) \tag{3}$$

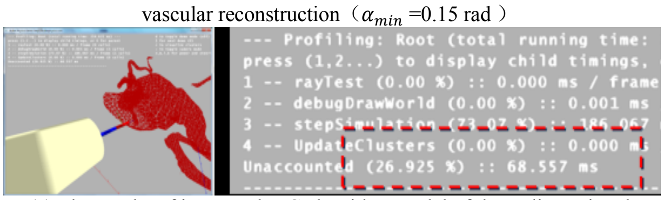
where $\alpha(\mathrm{V}_i, \mathrm{V}_j)$ was defined to be the angle value between the normal vector of the two vertexes V_i and V_j, which could be calculated by equation described in (3). α_{min} was the predefined minimum angle. And the value of $\alpha(\mathrm{V}_i, \mathrm{V}_j)$ needed to meet the relations described in equation (2).



(a) The results of early research model of three-dimensional vascular reconstruction



(b) The results of improved MC algorithm model of three-dimensional vascular reconstruction（α_{min} =0.15 rad ）



(c) The results of improved MC algorithm model of three-dimensional vascular reconstruction（α_{min} =0.30 rad）

Fig. 2 Comparison of the result of the model of the 3d-reconstruction

With these two constraint conditions, the number of triangles of the mesh model could be greatly reduced by merging two relative vertexes without any defect on the rendering or any structural properties of the mesh model of vessels, which would lead to the improvement of the real-time and realism of the system. Fig. 2 indicates the efficiency of the improved

MC algorithm by calculating the running time of the period of a frame flush in real-time simulation.

Also, it could be inferred from Table. I that the number of triangles of the mesh model was reduced from 12214 to 5009 and the time period of a frame flush was reduced from 198.930 ms to 68.557 ms with condition α_{min} =0.30 rad. This proved the efficiency of the improved MC algorithm by rigorous comparison of the results listed in Table. I and Table. II. We can see from Table. III that transmission frames per second (FPS) of the simulation system with improved MC algorithm is 14.70 fps, which was clearly increased by 3 times compared to the initial MC algorithm, and it was almost close to the resolution of human eyes. Because the resolution of human eyes is 30ms per second, and we aimed to keep up with this level of fps, results of our improvements shows that we got a tremendous promotion to the efficiency of our surgery system. This greatly contributed to the promotion of the VR-based vascular interventional surgery training system in real-time simulation. As a conclusion, the improved MC algorithm could be valuable and effective to the modeling of the vessels.

TABLE. I TRIANGLES OF IMPROVED MC ALGORITHM AND MC ALGORITHM

The early Stage of the Study	The Improved MC Algorithm	
	α_{min} =0.15rad	***α_{min} =0.30rad***
12214 (100%)	5873 （48.1%）	5009 （41.0%）

TABLE. II TIME-CONSUMING OF IMPROVED MC ALGORITHM AND MC ALGORITHM

The Early Stage of the Study	The Improved MC Algorithm	
	α_{min} =0.15rad	***α_{min} =0.30rad***
198.930ms	79.705ms (39.9%)	68.577ms (34.3%)

TABLE. III TRANSMISSION FREQUENCY OF IMPROVED MC ALGORITHM AND MC ALGORITHM

The Early Stage of the Study	The Improved MC Algorithm	
	α_{min} =0.15rad	***α_{min} =0.30rad***
5.05	12.65 (250.5%)	14.70 (291.1%)

III. A HYBRID COLLISION DETECTION ALGORITHM

The key idea of space decomposition method is to divide the whole space into small subspace according to various rules and repeat this process until a constraint factor is met, which allow intersectional testing only in a small range to detect collision [10]. Bounding box method is also a kind of widely used method for collision detection [11], which is totally different from space decomposition in main idea. In the Bounding box method, the main principle is to package the complex object with a bounding box, which is larger in volume and simpler in geometrical structure to approximate and represent the complex models [12]. The most widely used methods for collision detection are space decomposition method and the bounding box method respectively. Also, they are inter-related in main thoughts, for they all remove the region without collision and do precise collision detection only for the target part of the model, leading to promotion to efficiency of the collision detection method. Therefore, a new method combining the two methods could immensely improve the efficiency of collision detection while maintaining the advantages of the two methods respectively.

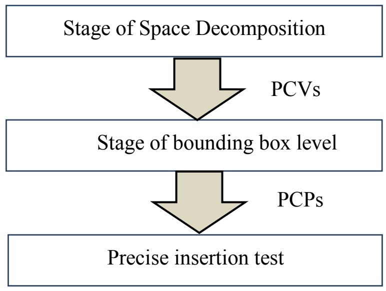


Fig. 3 Hybrid collision detection algorithm

Figure 3 shows a hybrid collision detection method was proposed and implemented in a logic order to combine the two. At the first step, the space decomposition method was adopted to get a spatial division of the 3D vascular model in the whole space, and primitives of all objects in the interventional surgery training system were allocated to the elements of the divisional frame, which helped to get the potential collision element. Then, all the potential collision elements were packaged as a set called potentially colliding volumes (PCVs).

At the second step, the hierarchical bounding box method was adopted on the PCVs (region near the tip of the catheter) for further study. Subsequently, a hierarchical tree of the bounding box was constructed for the PCVs, after which, a set of elements pairs called potentially colliding primitive pairs (PCPs) were obtained by traversing all bounding boxes of the hierarchical tree. The data set of PCPs was, then, imported into the intersectional tests to get precise testing result. As a result, the collision detection was perfectly and precisely realised, making the interaction real-time and fluently.

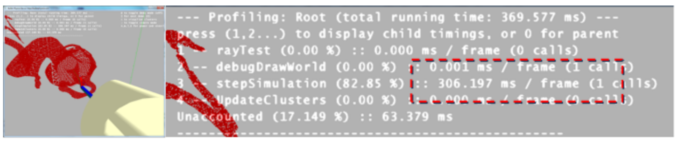
IV. CPU-GPU PARALLEL ARCHITECTURE

To ensure real-time performance during the simulation, the rendering for the simulation must be at least 30 fps [13, 14]. Thus, the software architecture should be implemented, for which, in this study, heterogeneous parallel computing was adopted to improve the rendering speed to make the simulation real-time.

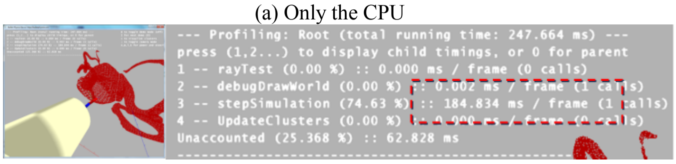
Heterogeneous parallel computing is the main trend of future development of high performance computing. With the development of GPU programming technology, a high-performance computing platform which is powerful and cost-effective for GPU-CPU mixtured programming could be essential to provide an efficient way for real-time simulation of the VR-based vascular interventional surgery training system of GPU collaborative CPU to build powerful computing performance and lower cost of high performance computing platform, is very beneficial to solve this problem ,the existing key problems in real-time character of the system.

Each stream processor of GPU can be considered as a processor that is separate, order execution supportive and single instruction flow allowed. GPU, generally, is always dependent on amounts of flow processors for parallel processing, which is more efficient in performance compared to the traditional serial computing.

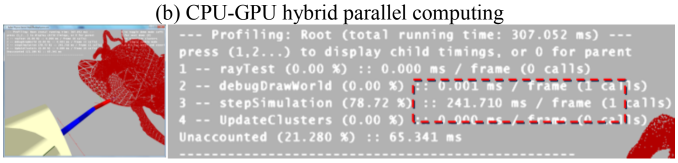
In this study, the CUDA library was selected as a parallel programming language, which is a widely used language for general computing. The core idea of CUDA programming is to make full use of all threads for parallel programming and exploit Thread Level Parallel, and these threads could be dynamically scheduled and executed in the hardware [15, 16]. Generally, GPU always shows its advantages while taking intensive tasks, in which GPU has to process huge amount of data sets, such as image processing, physical modelling, economical simulation, etc. [17] In this study, different modes of programming with CPU, CPU-GPU and CUDA were implemented respectively in the simulation process, and the time for 'StepSimulation' was recorded in real time, where 'StepSimulation' finishes the process of redrawing of the vascular model, which is shown in Fig. 4.



(a) Only the CPU



(b) CPU-GPU hybrid parallel computing



(c) CUDA library

Fig. 4 Comparison of the different application architecture's performance

TABLE. IV TIME-CONSUMING OF STEPSIMULATION IN THREE WAYS

Only the CPU	*CUDA Library*	*CPU-GPU Hybrid Parallel Computing*
306.197 ms (100%)	241.710 ms (78.9%)	184.834 ms (60.4%)

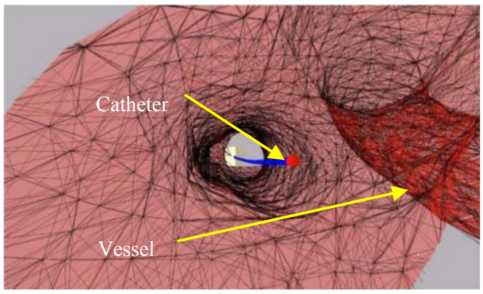
TABLE. V TIME-CONSUMING OF TOTAL IN THREE WAYS

Only the CPU	*CUDA library*	*CPU-GPU hybrid parallel computing*
369.577 ms (100%)	307.052 ms (83.1%)	247.644 ms (67.0%)

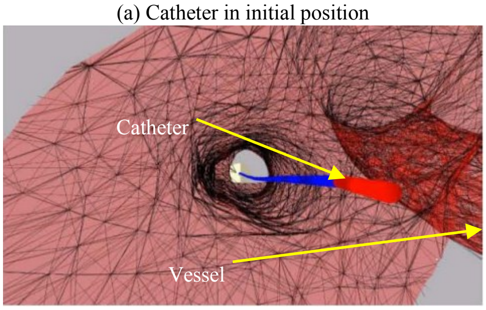
The result of the comparison of the three mode of programming was listed in Table. IV, which concluded the total time and the 'StepSimulation' time of the three modes. Table. IV shows the total time for the reconstruction of the model was reduced from 369.577 ms to 247.644 ms while adopting the mode of CPU-GPU mixtured programming, namely the reduction ratio was about 1/3. Because of the amounts of data sets causing computational burden to the hardware, leading to the incoherency to the simulation, the improvements on the framework of the program did contribute a lot to the refreshing rate and efficiency of our system. Meanwhile, it showed superiority to the mode of CUDA programming. Practically, there would be hundreds and thousands of data sets for processing in the system, which definitely caused a great burden to the hardware. However, CUDA programming did well in processing more data sets, which was not suitable in the study. As a result, the mode of CPU-GPU programming was adopted for parallel computing.

V. EXPERIMENT AND RESULTS

The platform of the virtual environment was established by the collaboration of Bullet, a physics engine, and OpenGL, a developing toolkit [18], and a virtual force needs to be sensed by the catheter in the virtual world. Therefore, Phantom, a haptic device with 6-DOF, should be tested in controlling the virtual catheter [19]. The input for haptic device could be radial and axial information of the handle's movement of haptic device. We used the OpenHaptic library to implement the haptic rendering and the interaction between the manipulator and the virtual world. Once the positional information of the pen-point of Phantom was changed, the encoder inside the pen-point would tell the virtual world to change the position of the catheter, just as shown in Fig.5. In order to evaluate the performance of the system, two tasks were designed for testing the VR-based vascular interventional surgery training system: linear motion and rotational movement.



(a) Catheter in initial position



(b) Catheter located after translation

Fig. 5 Process of the catheter inserting and rotating in the 3D vessel model

In detail, linear motion included a radial feed of 350mm, a radial retracement of 200 mm and a radial feed of 200mm combined motions; rotational movement included clockwise rotation of 360°, counter clockwise rotation of 360° and repetition of combined motions for five times (six of all). A test for the evaluation of VR-based vascular interventional surgery training system was carried out by a group of six people, three of whom are familiar with vascular interventional surgery and the other three were not. The first group of three people, who were experienced in manipulating the catheter and guide wire, were called experts group. On the contrary, the second group, with little experience on catheter manipulation, were called novices group.

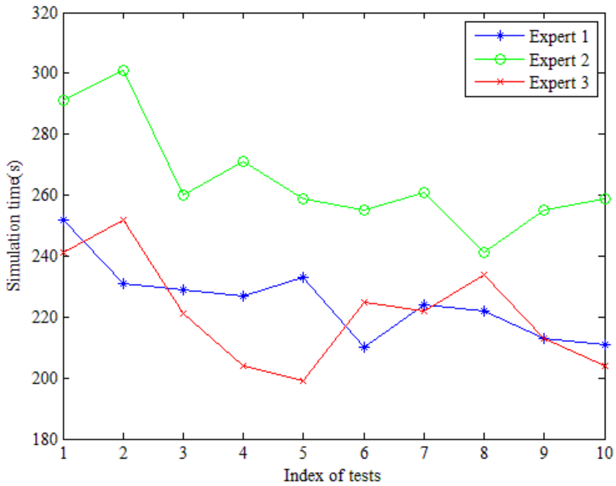


Fig. 6 Testing results of linear motion in the expert group

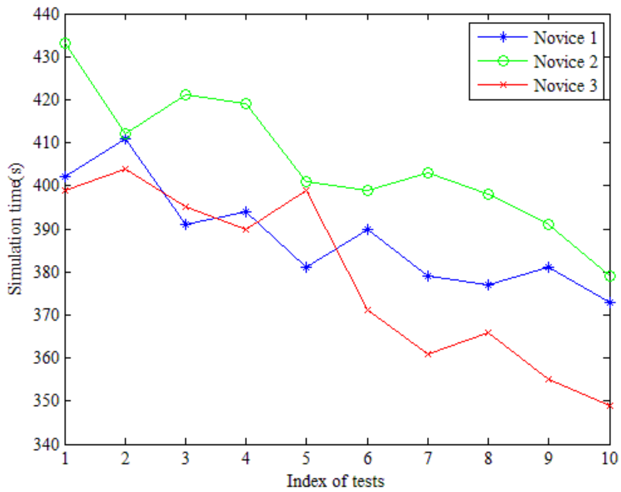


Fig. 7 Testing results of linear motion in the novice group

TABLE. VI CATHETER MODEL CONTROL (LINEAR)

Skilled Group			Unskilled Group		
No.1 (s)	***No. 2 (s)***	***No.3 (s)***	***No.4 (s)***	***No.5 (s)***	***No.6 (s)***
252	291	241	402	433	399
231	301	252	411	412	404
229	260	221	391	421	395
227	271	204	394	419	390
233	259	199	381	401	399
210	255	225	390	399	371
224	261	222	379	403	361
222	241	234	377	398	366
213	255	213	381	391	355
211	259	204	373	379	349

TABLE. VII CATHETER MODEL CONTROL (ROTATIONAL)

Skilled Group			Unskilled Group		
No.1 (s)	***No.2 (s)***	***No.3 (s)***	***No.4 (s)***	***No.5 (s)***	***No.6 (s)***
131	150	143	301	351	277
130	143	129	291	331	290
125	139	128	251	333	249
126	130	128	274	318	244
122	131	119	263	309	237
121	136	122	261	179	191
127	119	123	221	194	180
119	131	121	239	195	209
125	126	122	230	181	183
118	124	117	211	196	170

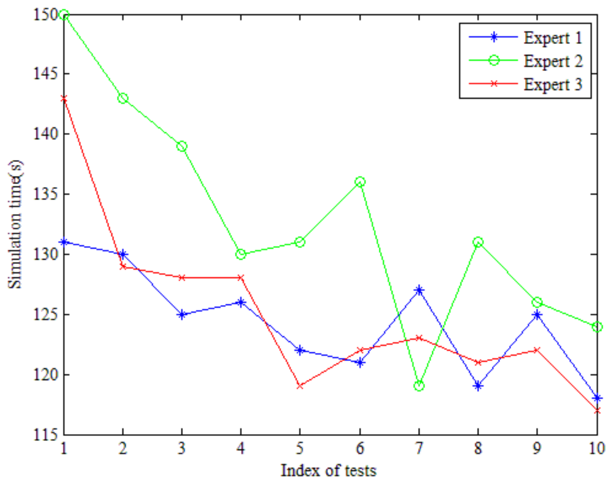


Fig. 8 Testing results of rotational motion in the expert group

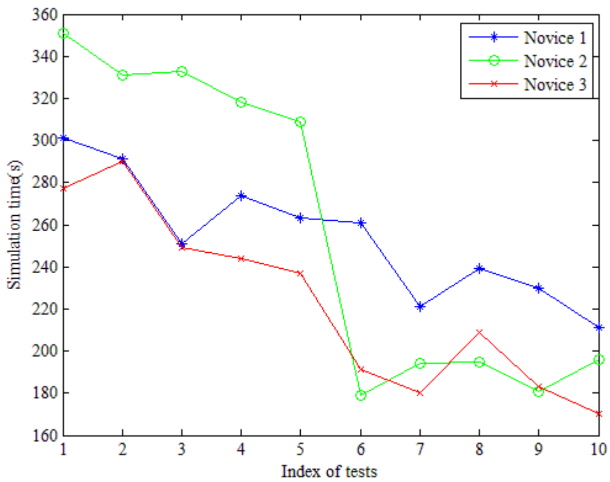


Fig. 9 Testing results of rotational motion in the novice group

Table. VI shows the testing results are listed. Accordingly, we can see from Fig.6 and Fig.7, during the process of testing, the experts group could finish the linear feed of the catheter and guide wire easily in a shorter time that of the novice group, who initially found it difficult to finished the same work. However, the novices group became more experienced and skilled and completed the task with shorter time after several cycles of training with the VR-based vascular interventional surgery training system. Meanwhile, it also indicates that the expert group also spent shorter time after rounds of training, which showed the benefits of the system.

During the tests of rotational motion, time for period of training became shorter and shorter, especially in the novices group, which indicated that training with the system were beneficial for novices in terms of interventional surgery skill learning, as shown in TABLE. VII, Fig.8 and Fig.9.

VI. Conclusions and Future Work

The top killer of cerebrovascular disease has always been a threat to human health. Radioactive interventional therapy has been proved to be the most effective treatments for cerebrovascular disease. Nevertheless, the traditional way to train novice surgeons is always limited by resources and time. In this research, a combination of VR technology and haptic force feedback technology was adopted and an improved VR-based vascular interventional surgery training system was developed, which provided a cost-effective and effective way for novices to become experienced vascular interventional surgeons.

In this study, method of 3D modelling of blood vessels, collision detection and rendering speed acceleration were improved and demonstrated.

The specific research results are as follows:

(1) number of triangles of mesh model of vessels was reduced from 12214 to 5009, frames transmitted per second of the system were increased from 5.05 to 14.70.

(2) a new method combined with the space decomposition and bounding box method was adopted, which improved the efficiency of collision detection.

(3) CPU-GPU mixed programming mode was adopted in computing which reduced the time for reconstruction and rendering from 369.58 ms to 247.64 ms.

Some future work can be done in three parts. Firstly, we plan to construct a mathematical model of the variables related to the feedback force by the virtual vessels and catheter and guide wire to accurately simulate an immersive feeling of manipulating the VR-based system. Once the collision or the deformation occurs, the manipulator will sense the feedback force in real-time. Next, we plan to endow the vessels elasticity to make the surgery more vivid. Moreover, we also plan to simulate an aural sensation of the surgery to make the virtual environment more real.

References

[1] EN NMH publications: Global status report on noncommunicable diseases 2014, *World Health Organization*, Switzerland, 2014.

[2] Vaughan N, Dubey V N, Wainwright T W, et al, "A review of virtual reality based training simulators for orthopaedic surgery," *Medical Engineering & Physics*, vol. 38, no. 2, pp. 59–71, 2015.

[3] J Guo, Y Gao, S Guo, Y Wang, "Kinematics analysis of the catheter for a novel VR robotic catheter system," in *Proceedings of 2014 IEEE International Conference on Mechatronics and Automation. (ICMA)*, pp. 1034–1039, 2014.

[4] C Huang, S K Agrawal, H M Ladak, "Virtual Reality Simulator for Training in Myringotomy with Tube Placement," *Journal of Medical & Biological Engineering*, vol. 36, no. 2, pp. 214–225, Apr. 2016.

[5] Y Wang, S Guo, T Tamiya, X Yin, "A virtual‐reality simulator and force sensation combined catheter operation training system and its preliminary evaluation," *International Journal of Medical Robotics + Computer Assisted Surgery Mrcas,* In press, 2016.

[6] Kangqi Hu, Baofeng Gao, Nan Xiao, et al, "Simulation of the virtual reality based robotic catheter system," *in Proceedings of 2013 International Conference on Complex Medical Engineering (ICME)*, pp. 59–63, 2013.

[7] Wang Yu, Shuxiang Guo, T Tamiya, Hidenori Ishihara, "A Blood Vessel Deformation Model Based Virtual-reality Simulator for the Robotic Catheter Operating System," *Neuroscience & Biomedical Engineering*, vol. 2, no. 3, pp. 1–1, 2014.

[8] Wang Yu, Shuxiang Guo, "Elasticity analysis of Mass-spring model-based virtual reality vascular simulator," *in Proceedings of 2013 International Conference on Mechatronics and Automation (ICMA)*, pp. 292–297, 2014.

[9] X. Zou, X. J. Li, Y. Y. Gao, K. F. He, J. J. Kun, "VSR Mechanism Based on MC Algorithm Simulation of Grain Growth Behavior of Sustainable Materials," *Applied Mechanics and Materials*, vol. 340, pp. 392–395, 2013.

[10] M Huang, P Pang L, Y Lu, Z Xia, "A Fast Space-decomposition Scheme for Nonconvex Eigenvalue Optimization," *Set-Valued and Variational Analysis*, vol. 24, no. 90, pp. 1–25, 2016.

[11] S F Johnsen, Z A Taylor, L Han, Sebastien Ourselin, "Detection and modelling of contacts in explicit finite-element simulation of soft tissue biomechanics," *International Journal of Computer Assisted Radiology & Surgery*, vol. 10, no. 11, pp. 1873–1891, 2015.

[12] Jiangchao Li, Baofeng Gao, Shuxiang Guo, "Design of collision detection algorithms and force feedback for a virtual reality training intervention operation system," *in Proceedings of 2013 IEEE International Conference on Robotics and Biomimetics (ICRB)*, pp. 1660–1665, 2015.

[13] T C Knott, T W Kuhlen, "Accurate and adaptive contact modeling for multi-rate multi-point haptic rendering of static and deformable environments," *Computers & Graphics*, vol. 57, pp. 68–80, 2016.

[14] S Müller, A Bihlmaier, S Irgenfried, Heinz Wörn, "Hybrid Rendering Architecture for Realtime and Photorealistic Simulation of Robot-Assisted Surgery," *Studies in Health Technology & Informatics*, vol. 220, no. 8, pp. 680–686, 2016.

[15] Zhu Q, Wu B, Shen X, Z Wang, "Understanding co-run performance on CPU-GPU integrated processors: observations, insights, directions," *in Proceedings of Frontiers of Computer Science*, pp. 1–17, 2016.

[16] Rek V, Němec I, "Parallel Computing Procedure for Dynamic Relaxation Method on GPU Using NVIDIA's CUDA," *Applied Mechanics & Materials*, vol. 821, pp. 331–337, 2016.

[17] De A, Zhang Y, Guo C, "A parallel adaptive segmentation method based on SOM and GPU with application to MRI image processing," *Neurocomputing*, vol. 198, no. C, pp. 180–189, 2016.

[18] Woods J, Christian J. Glidar, "An OpenGL-based, Real-Time, and Open Source 3D Sensor Simulator for Testing Computer Vision Algorithms," *Journal of Imaging,* vol. 2, no. 1, pp. 1–9, 2013.

[19] Kimmer S, Smisek J, Schiele A, "Effects of Haptic Guidance and Force Feedback on Mental Rotation Abilities in a 6-DOF Teleoperated Task," *in Proceedings of 2015 IEEE International Conference on Systems Man and Cybernetics*, pp. 3092–3097, 2015.