



Title	Development of three-dimensional patient face model that enables real-time collision detection and cutting operation for a dental simulator
Author(s)	Yamaguchi, Satoshi; Yamada, Yuya; Yoshida, Yoshinori et al.
Citation	Dental Materials Journal. 2012, 31(6), p. 1047-1053
Version Type	VoR
URL	https://hdl.handle.net/11094/93088
rights	Copyright 2012 The Japanese Society for Dental Materials and Devices
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

<https://ir.library.osaka-u.ac.jp/>

The University of Osaka

Development of three-dimensional patient face model that enables real-time collision detection and cutting operation for a dental simulator

Satoshi YAMAGUCHI¹, Yuya YAMADA¹, Yoshinori YOSHIDA¹, Hiroshi NOBORIO² and Satoshi IMAZATO¹

¹Department of Biomaterials Science, Osaka University Graduate School of Dentistry, 1-8 Yamadaoka, Suita, Osaka 565-0871, Japan

²Department of Computer Science, Osaka Electro-Communication University, 1130-70 Kiyotaki, Shijonawate, Osaka 575-0063, Japan

Corresponding author, Satoshi YAMAGUCHI; E-mail: yamagu@dent.osaka-u.ac.jp

The virtual reality (VR) simulator is a useful tool to develop dental hand skill. However, VR simulations with reactions of patients have limited computational time to reproduce a face model. Our aim was to develop a patient face model that enables real-time collision detection and cutting operation by using stereolithography (STL) and deterministic finite automaton (DFA) data files. We evaluated dependence of computational cost and constructed the patient face model using the optimum condition for combining STL and DFA data files, and assessed the computational costs for operation in do-nothing, collision, cutting, and combination of collision and cutting. The face model was successfully constructed with low computational costs of 11.3, 18.3, 30.3, and 33.5 ms for do-nothing, collision, cutting, and collision and cutting, respectively. The patient face model could be useful for developing dental hand skill with VR.

Keywords: Virtual reality, Hand skill training, Real-time simulation, Stereolithography, Deterministic finite automaton

INTRODUCTION

Although it is essential for dental students to acquire advanced hand skill for daily treatment, conventional mannequin-based trainings have limitations for case setting¹. To overcome such problems, much attention has been paid to the virtual reality (VR) simulator, which has been recognized as a strong tool in the car industry²⁻⁴, airplane industry⁵, and medicine⁶⁻¹¹. Haptic devices that make users feel force feedbacks are useful for acquiring hand skill efficiently in VR environments^{12,13}. Therefore, attempts have been made to develop VR simulators with haptic devices for caries detection¹⁴, pocket probing, scaling^{15,16}, and amalgam fillings^{16,17}. VR training simulators also have the advantage of enabling repeated training using standardized scoring criteria. The Iowa Dental Surgical Simulator¹⁸ has been developed to evaluate learning performance of caries detection by dental students and dentists. Suebnukarn *et al.* reported a useful system to measure the process and outcome of expert/novice performance in preparing crowns¹⁹ and accessing opening tasks²⁰.

However, conventional VR simulators are not yet enough to develop dental skills with realistic clinical settings because they do not reproduce patients' reactions. It is important to improve simulation technology by achieving a three-dimensional patient face model that reproduces patients' reactions to dental procedures. Although Marras *et al.* have reported a three-dimensional face and oral cavity model with face animation play in MPEG-4 data format¹⁷, the animation was not interactive for dental operations. The reason for the difficulty in developing interactive models is that the computational costs depend on collision detection

techniques. To implement an interactive animation in dental simulators, collision detections between dental instruments and oral cavity during cutting operation are indispensable, which normally results in a great increase in the computational cost.

Previously, we have reported a new VR dental simulator²¹ in which the volumetric tooth models for collision and cutting operations are represented by deterministic finite automaton (DFA) data files²². In addition, to enhance the visual reality, surface models in stereolithography (STL)²³ data format were covered with their corresponding volumetric tooth models in this simulator²¹. Such a combination of DFA and STL data files is considered effective to reduce the computational costs of collision and cutting operations, and may be useful for achieving interactive animation. The aims of this study were to determine optimum conditions for combining DFA and STL data files to reduce the computational costs, and to develop a three-dimensional patient face model that enables real-time (update time for drawing should be less than 33 ms)²⁴ collision detection and cutting operation.

MATERIALS AND METHODS

System configuration of simulator

The VR simulator consisted of a workstation (xw4600, Hewlett Packard Japan, Tokyo, Japan), haptic device (PHANTOM Omni, SensAble Technologies, Wilmington, MA, USA), foot pedal (CH Pro Pedal, CH Products, Vista, CA, USA), and finger rest (Fig. 1). Table 1 shows the workstation specifications. Virtual models for training tasks can be divided into a surface part (do-nothing) and volumetric part (collision, cutting, and collision-cutting). Here, do-nothing means that an operator did nothing while the VR simulator was running. On the other

Color figures can be viewed in the online issue, which is available at J-STAGE.

Received Jun 20, 2012; Accepted Sep 3, 2012

doi:10.4012/dmj.2012-164 JOI JST.JSTAGE/dmj/2012-164

hand, a tip of dental turbine collided (collision) and cut the tooth (cutting). In addition, a tip of dental turbine cut the tooth and a body of the dental turbine collided with a lip or a tongue (collision and cutting). The surface part such as the gingiva is represented by a STL data format that can be covered with the volumetric part. The cutting part such as enamel, dentin, pulp, and caries are represented by octree-algorithm-based DFA data files corresponding to collision detection and cutting. In DFA data files, the voxel resolution can be calculated with

$$R = S/2^{OL} \quad (1)$$

Table 1 Workstation specifications

Model	Hewlett-Packard HP xw4600 Workstation
Operating System	Windows XP Professional Service Pack 3
Processor	Intel(R) Core(TM)2 Duo CPU E8400@3.00GHz(2 CPUs)
Memory	DDR-2 SDRAM/800MHz 3072MB RAM
Graphics	NVIDIA Quadro FX 3700

where R is the voxel resolution, S is the scale of the octree space, and OL is the octree level.

Determination of optimum conditions to develop patient face model

To determine an optimum condition to develop the patient face model, we evaluated the dependence of computational costs in terms of step time on 1) the STL data file size, 2) the number of layers in DFA data files, 3) the voxel resolution of DFA data files, and 4) non-existent, visible or invisible states of DFA data files on a human interface. Here, the optimum condition, within 33 ms, signified that the best combinations of STL and DFA files that are used to construct patient face model. The following conditions were prepared for each parameter. For each evaluation, averages of step times were calculated for five times by using an internal clock of the workstation. Here, each step time meant that a required time to finish both Thread 1 and Thread 2 as shown in Fig. 2.

1. Size of STL data files

We prepared six models: 0 (control), 10, 20, 30, 40, and 50 MB STL files (100×100×9 mm) combining a DFA data file (100×100×9 mm). As shown in Fig. 3, sizes of STL files were adjusted by varying an area of STL surface.

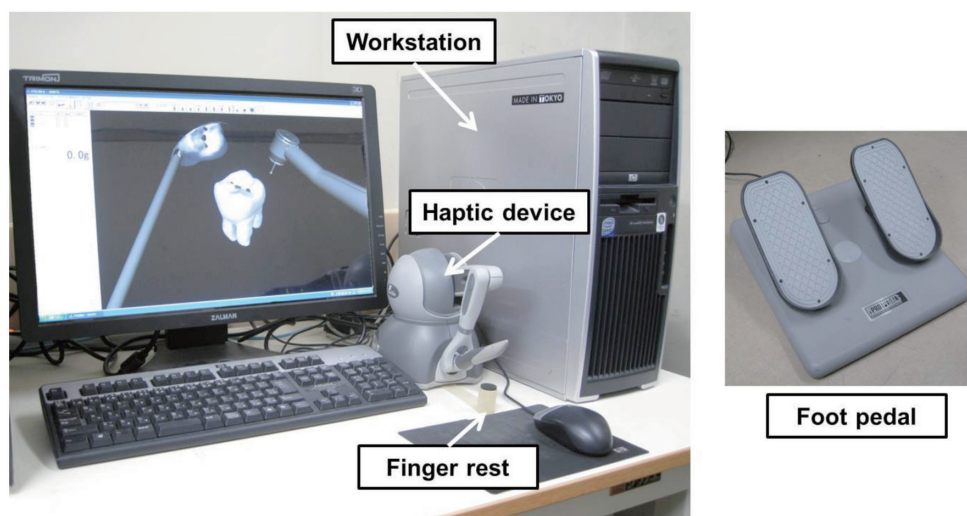


Fig. 1 System configuration of VR simulator.

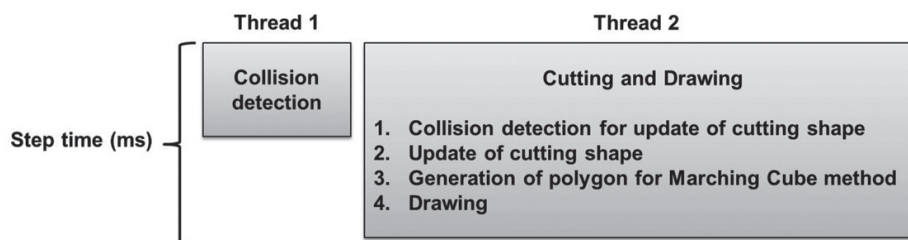


Fig. 2 Diagram of step time.

2. Number of layers in DFA data files

Three types of data files were constructed (one layer: 100×100×9 mm and 554 KB (Fig. 4a); one of two layers: 100×100×4.5 mm and 277 KB (Fig. 4b); one of three layers: 100×100×3 mm and 275 KB (Fig. 4c) assuming a combined DFA model with enamel, dentin, and pulp. We constructed the double or triple layers by dividing the single layer model in two or three, and measured the computational costs of the do-nothing and cutting operations (with multiple layers cut at the same time).

3. Voxel resolutions of DFA data files

We prepared a DFA model named 80-size (layer A: 100×100×4.5 mm and 277 KB, layer B: 100×100×80 mm (Fig. 5) that combined small and large models such as

a tooth and face. The octree level of layer A was set to 9. The octree level of layer B was set to 7, 8, or 9 (17, 25, and 277 KB). We measured the computational costs of the do-nothing operation, the collision operation for layer B, and the cutting operation for layer A.

Then, we prepared four types of single-layer DFA models with a size of 100×100×9 mm (same dimension of one layer model as shown in Fig. 4a) and octree level of 7, 8, 9, or 10 (34, 49, 554, and 2190 KB), and measured the computational costs of the do-nothing and cutting operations.

4. Non-existent, visible or invisible states of DFA data files

We prepared the 80-size combined DFA models (807

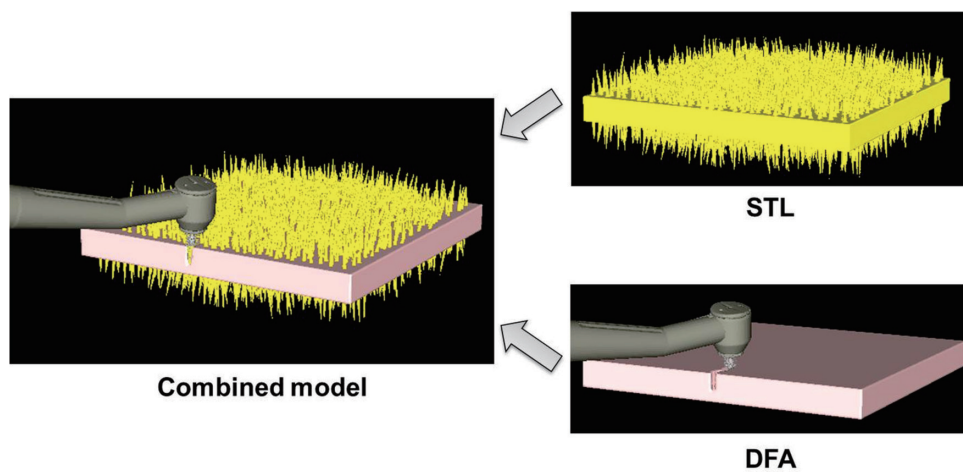


Fig. 3 Experimental model for size of STL data files.

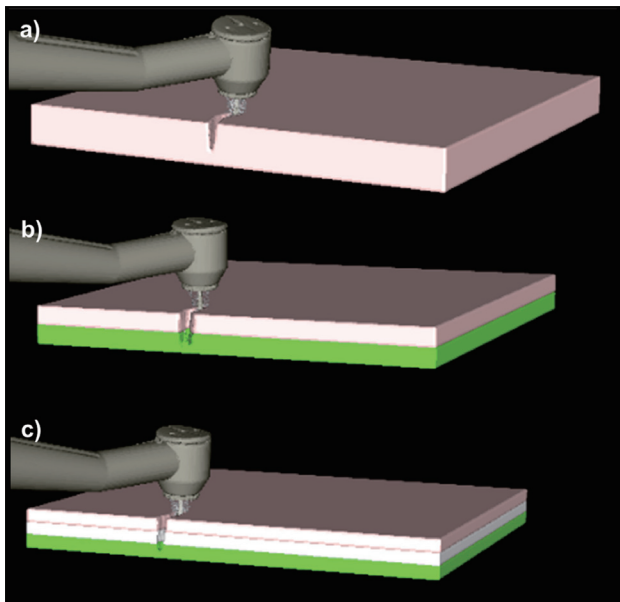


Fig. 4 Experimental models for number of layers in DFA data files.
a) one-layer, b) two-layers, c) three-layers.

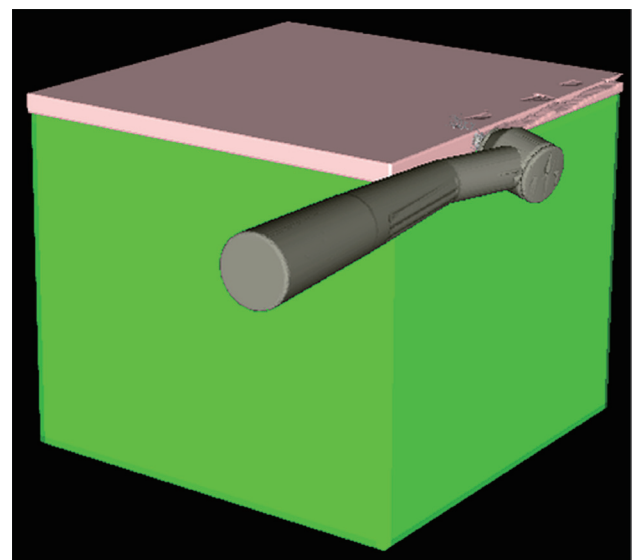


Fig. 5 Experimental model for voxel resolutions of DFA data files (top: layer A, bottom: layer B).

KB) with three different states (non-existent as control (Fig. 6a), visible (Fig. 6b), and invisible (Fig. 6c), and measured the computational costs of cutting and do-nothing operation. The non-existent state did not include layer B. The invisible state was combinations of a visible layer A and an invisible layer B that was available for collision detection. For a visible state, the visible layer A and visible layer B were combined.

We also compared the computational costs of the cutting operation for three different types of combined DFA models named 40-size (layer B: 100×100×40 mm and 516 KB), 80-size (layer B: 100×100×80 mm and 807 KB), and 160-size (layer B: 100×100×160 mm and 1740 KB) by varying a depth of layer B as shown in Fig. 5. The size of layer A was 100×100×4.5 mm and 277 KB in all models.

For all models, the step time of collision and cutting

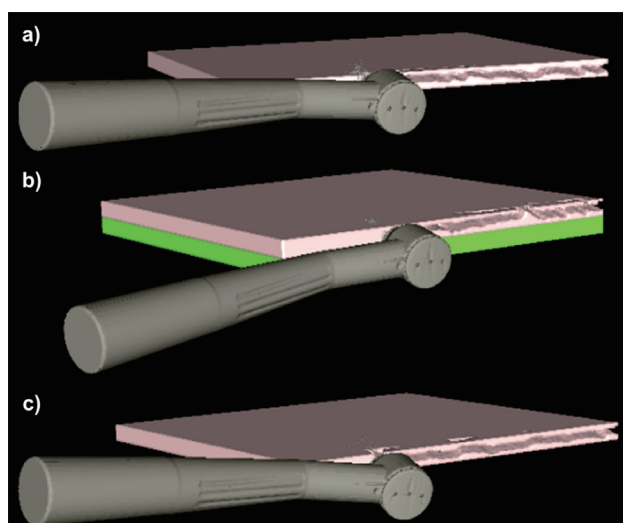


Fig. 6 Experimental models for non-existent, visible or invisible states of DFA data files.

a) non-existent model, b) visible model (top: layer A, bottom: layer B), c) invisible model (top: layer A, bottom: invisible layer B)

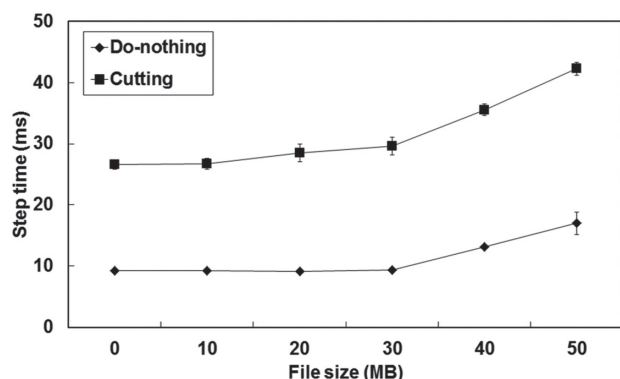


Fig. 7 Step time vs. computational cost in file size for do-nothing and cutting operations.

was not measured because the step time of Thread 1 related with collision is few in comparison with that of Thread 2 as shown in Fig. 2.

Construction of patient face model

The face model was constructed based on the optimum condition found for combining STL and DFA data files. Each part of the face model was reconstructed from CT data sets by using volume rendering software (VG Studio Max 2.0, Volume Graphics, Heidelberg, Germany). With CAD software (FreeFormModeling, SensAble Technologies), an upper part of the face was removed and an opened mouth constructed. The dentition model was scanned from a prosthetic restoration jaw model (D18FE-500E, NISSIN, Kameoka, Japan) using a three-dimensional measurement system (Rexcan ARX, Solutionix, Seoul, Korea) and exported into an STL data format. As a cutting target in the VR simulator, regions of the mandibular left first molar (C10) were extracted from micro CT data²⁵⁾ sets using volume rendering software and reconstructed as a three-dimensional tooth model. A corresponding part of the dentition was replaced by the tooth model. A face, lip, oral mucosa, tongue, and dentition were composed of STL data files. We included the DFA data files required for collision detection. The tooth model was also composed of DFA data files. To realize a run at the minimum frequency, a mesh optimization technique using reverse modeling software (LEIOS, Data Design, Nagoya, Japan) was applied. Finally, we measured the computational costs of the do-nothing, collision, cutting, and collision-cutting operation.

RESULTS

Determination of optimum conditions to develop patient face model

Figure 7 shows the computational costs as a function of STL data file sizes. The step time did not change from the control (0 MB) up to the 30 MB file size and increased dramatically beyond 30 MB.

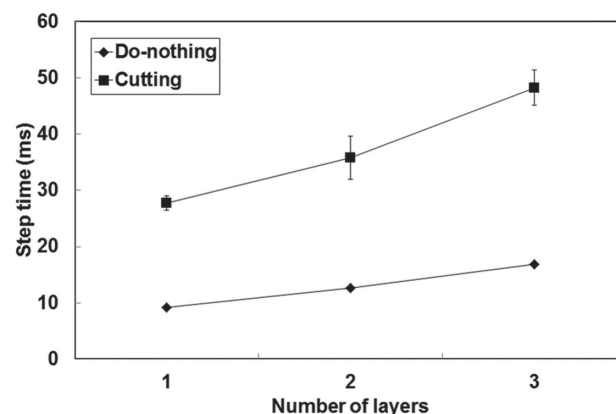


Fig. 8 Step time vs. computational cost in number of layers for do-nothing and cutting operations.

For DFA files, the step time increased as the number of layers increased (Fig. 8). In the cutting operation, the computational costs other than for one layer were over 33 ms.

For three octree levels of DFA files, computational costs for octree levels 7 and 8 did not differ among the do-nothing, collision, and cutting operations (Fig. 9a). However, computational costs slightly increased for octree level 9, showing over 33 ms particularly for cutting operation (Fig. 9a). For the three different types of combined DFA data files, there were clear differences between octree levels 9 and 10 for both do-nothing and cutting operations (Fig. 9b).

The computational cost of non-existent, visible or invisible states of DFA data files for cutting operations increased in comparison with do-nothing operations (Fig. 10a). Figure 10b shows the difference in computational costs between 40-size, 80-size, and 160-size for all three states. The computational cost of the invisible state decreased, though that of the visible state increased as

the thickness of the experimental model increased.

Construction of patient face model

Figure 11 shows the face model constructed. The model consisted of five STL and two DFA data files. The STL data files consisted of the face (3.2 MB), lip (1.2 MB), oral mucosa (4.4 MB), tongue (2.5 MB), and dentition (2.1 MB). After applying the mesh optimization technique, the total file size of the STL data files was 13.4 MB. The DFA data files consisted of a C10 tooth model and an oral region. The octree level of C10 was 9, and its octree space scale was 51 mm. Thus, C10's voxel resolution was 0.0996 mm (less than 100 μ m). The octree level of the oral region, consisting of the lip, oral mucosa, dentition, and tongue was 8, and its octree space scale was 181. Thus, the voxel resolution of the oral region was 0.707 mm. The computational costs (average \pm standard deviation) of the do-nothing, collision, cutting, and collision-cutting operations were 11.3 \pm 0.1, 18.3 \pm 0.1, 30.3 \pm 0.9, and 33.5 \pm 0.7 ms, respectively.

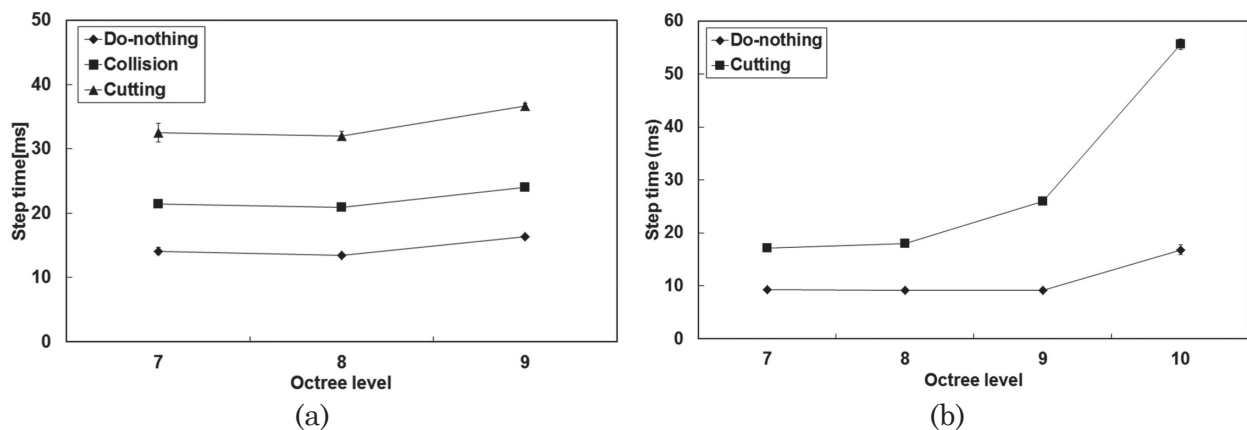


Fig. 9 (a) Step time vs. computational cost in octree level for do-nothing, collision, and cutting operations. (b) Step time vs. computational costs in octree level for do-nothing and cutting operations.

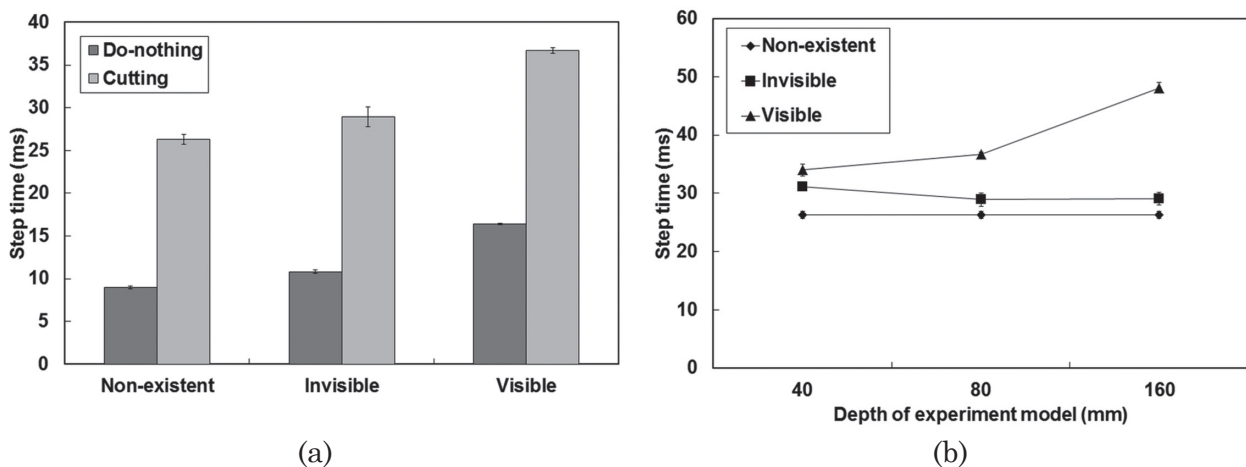


Fig. 10 (a) Computational cost for non-existent, invisible, and visible states for do-nothing operation. (b) Step time vs. computational cost in thickness of experimental model for non-existent, invisible, and visible states.

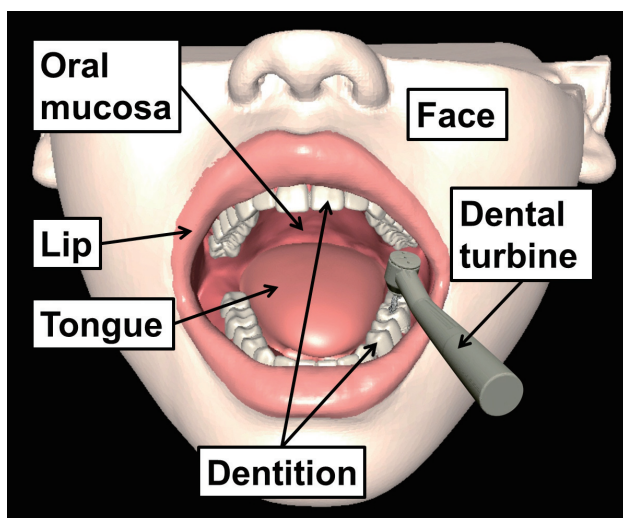


Fig. 11 Developed patient face model.

DISCUSSION

As described by Rhiemora *et al.*, the graphics loop of the VR simulator requires adherence to run at a minimum frequency of 30 Hz (less than 33 ms)²⁴. Thus, assessment of the dependence of computational costs on the STL data file size revealed a file size limit for the patient face model, suggesting that the total STL data format size should not be over 30 MB to realize a real-time simulation under the specifications as shown in Table 1. One of the reasons for increasing step time beyond 30 MB is to account for the “cache problem”²⁶ that a thread 2 as shown in Fig. 2 accessed to huge octree nodes, which were located randomly in the physical memory.

The computational costs increased for the cutting operation for multi-layered DFA models. Even with do-nothing operation, the computational costs increased just due to the increment of loop time for both Threads depending on the multi-layering. This suggests that a few-layered model is better for real-time simulation. The reason of maintaining 10 ms is to consider the running collision detection.

Based on the results for the dependence of computational costs on the voxel resolution DFA data files, it was found that a part for collision detection should adopt octree level 8. This is due to the fact that computational costs of level 8 was similar to level 7 and less than 33 ms. In addition, surface resolution related with smoothness of level 8 should be smaller than level 7 by following equation (1). Furthermore, there was a difference between octree levels 9 and 10 in the size of the required physical memory. The maximum required memory sizes of level 8, level 9, and level 10 were 86, 690, and 5,522 MB, respectively. These results suggest that a size of physical memory had a bottleneck effect on the system because level 10 was beyond 3,072 MB as shown in Table 1. This means the “cache problem” cannot be

solved even if the specification as shown in Table 1 was updated by using a high-performance computer. Thus, it is necessary to implement an algorithm that will locate octree nodes efficiently and quickly. Achieving the required voxel resolution of the patient face model in a DFA data file needed an octree level of 9 and octree space scale of 51 or octree level of 10 and octree space scale below 102, as required by equation (1). Since the computational cost during cutting operation of octree level 10 was over 33 ms (Fig. 9b), the former octree level and space scale should be adopted.

Using DFA data files for invisible parts of cutting targets did not reduce the computational costs for any of them (Fig. 10a). To further reduce the costs, DFA data files should not be used for unnecessary parts. Thus, DFA data files such as the lip, oral mucosa, tongue, and dentition were prepared as the invisible parts. Also, we used STL data files with DFA files (without a collision detection) to represent the face model that must not be touched by dental instruments. In addition, we used a single-layered DFA file for the lip, oral mucosa, tongue, dentition, and tooth of the cutting target. The computational costs of the cutting operation for three different types of combined DFA models suggest that reducing the computational cost causes the invisible process to increase as the size of the DFA data format increases (Fig. 10b).

The patient face model was constructed using the optimum condition for combining STL and DFA data files. As a result, the face model was successfully constructed with low computational costs for do-nothing, collision, cutting, and collision-cutting, respectively. The do-nothing, collision, and cutting states achieved real-time simulation. Collision-cutting required computational costs over 33 ms for real-time simulation. However, the collision detection and cutting operation seldom occur at the same time during training. From this viewpoint, we can say that real-time simulation of collision-cutting was almost achieved.

The collision detection algorithm needs to search a contact area for the dental instrument within a huge number of voxels in the patient face model including the oral cavity. Reducing computational costs will expand the possibilities of future VR dental haptic simulators. With the patient face model, various training tasks will be available. Moreover, the patient face model not only enhances visual reality but also has great potential to train people to sense a patient’s reactions such as painful and worried looks²⁷ in VR environments.

To further improve the patient face model developed in this study, the parallel computing algorithm that available on a general-purpose graphics processing unit (GPGPU)²⁸, which is an expanded graphics processing unit for three-dimensional calculation, may be useful to reduce the computational costs. The GPGPU is available on the personal computer instead of supercomputer and easy to implement with the C language. In addition, the parallel computing algorithm with the GPGPU has already been implemented in many applications including octree algorithms^{29,30}. We are working on

implementing GPGPU architecture for octree algorithms in a developed VR simulator.

CONCLUSION

We investigated the optimum conditions of STL and DFA data files to develop a three-dimensional patient face model that enables real-time collision and cutting operation. Less than 30 MB of STL data files was needed to achieve real-time simulation. An octree level of 8 was optimum for non-cutting parts such as the lip, oral mucosa, tongue, and dentition other than the tooth of a cutting target, for which level 9 was optimum. Setting an invisible state for the DFA data files was effective in reducing the computational costs. By using the conditions we determined, we successfully constructed a patient face model with real-time simulation.

ACKNOWLEDGMENTS

This research was supported by Good Practice Project 2007 of the Ministry of Education, Culture, Sports, Science and Technology (MEXT) of Japan and Grants-in-Aid for Scientific Research Nos. 22360109 and 24592955 from the JSPS. We are grateful to Associate Professor Sakae Nagasawa at Matsumoto Dental University for the use of micro-CT data.

REFERENCES

- 1) Suvinen TI, Messer LB, Franco E. Clinical simulation in teaching preclinical dentistry. *Eur J Dent Educ* 1998; 2: 25-32.
- 2) Kuhl J, Evans D, Papelis Y, Romano R, Watson G. The Iowa Driving Simulator: an immersive research environment. *IEEE Comput* 1995; 28: 35-41.
- 3) Cremer J, Kearney J, Papelis Y. Driving simulation: challenges for VR technology. *IEEE Comput Graph Appl*, 1996; 16: 16-20.
- 4) Ambroz M, Prebil I. i3Drive, a 3D interactive driving simulator. *IEEE Comput Graph Appl* 2010; 30: 86-92.
- 5) Yavrucuk I, Kubali E, Tarimci O. A low cost flight simulator using virtual reality tools. *IEEE AES Magazine* 2011; 26: 10-14.
- 6) Chaudhry A, Sutton C, Wood J, Stone R, McCloy R. Learning rate for laparoscopic surgical skills on MIST VR, a virtual reality simulator: quality of human-computer interface. *Ann R Coll Surg Engl* 1999; 81: 281-286.
- 7) Kallstrom R, Hjertberg H, Svanvik J. Construct validity of a full procedure, virtual reality, real-time, simulation model for training in transurethral resection of the prostate. *J Endourol* 2010; 24: 109-115.
- 8) Colt HG, Crawford SW, Galbraith O. Virtual reality bronchoscopy simulation—A revolution in procedural training. *Chest* 2001; 120: 1333-1339.
- 9) Maschuw K, Hassan I, Bartsch DK. Surgical training using simulator. *Chirurg* 2010; 81: 19-24.
- 10) Panait L, Merrell RC, Rafiq A, Dudrick SJ, Broderick TJ. Virtual reality laparoscopic skill assessment in microgravity. *J Surg Res* 2006; 136: 198-203.
- 11) Rosen JM, Long SA, McGrath DM, Greer SE. Simulation in plastic surgery training and education: The path forward. *Plast Reconstr Surg* 2009; 123: 729-738.
- 12) Quinn F, Keogh P, McDonald A, Hussey D. A study comparing the effectiveness of conventional training and virtual reality simulation in the skills acquisition of junior dental students. *Eur J Dent Educ* 2003; 7: 164-169.
- 13) Quinn F, Keogh P, McDonald A, Hussey D. A pilot study comparing the effectiveness of conventional training and virtual reality simulation in the skills acquisition of junior dental students. *Eur J Dent Educ* 2003; 7: 13-19.
- 14) Thomas G, Johnson L, Dow S, Stanford C. The design and testing of a force feedback dental simulator. *Comput Methods Programs Biomed* 2001; 64: 53-64.
- 15) Steinberg AD, Bashook PG, Drummond J, Ashrafi S, Zefran M. Assessment of faculty perception of content validity of PerioSim, a haptic-3D virtual reality dental training simulator. *J Dent Educ* 2007; 71: 1574-1582.
- 16) Kim L, Park SH. Haptic interaction and volume modeling techniques for realistic dental simulation. *Vis Comput* 2006; 22: 90-98.
- 17) Marras I, Nikolaidis N, Mikrogeorgis G, Lyroutdia K, Pitas I. A virtual system for cavity preparation in endodontics. *J Dent Educ* 2008; 72: 494-502.
- 18) Johnson L, Thomas G, Dow S, Stanford C. An initial evaluation of the Iowa Dental Surgical Simulator. *J Dent Educ* 2000; 64: 847-853.
- 19) Suebnukarn S, Phatthanasathiankul N, Sombatwerroje S, Rhiemora P, Haddawy P. Process and outcome measures of expert/novice performance on a haptic virtual reality system. *J Dent* 2009; 37: 658-665.
- 20) Suebnukarn S, Haddawy P, Rhiemora P, Gajananan K. Haptic virtual reality for skill acquisition in endodontics. *J Endod* 2010; 36: 53-55.
- 21) Yoshida Y, Yamaguchi S, Kawamoto Y, Noborio H, Murakami S, Sohmura T. Development of a multi-layered virtual tooth model for the haptic dental training system. *Dent Mater J* 2011; 30: 1-6.
- 22) Lucas SM, Reynolds TJ. Learning deterministic finite automata with a smart state labeling evolutionary algorithm. *IEEE Trans Pattern Anal Mach Intell* 2005; 27: 1063-1074.
- 23) Kato A, Ohno N. Construction of three-dimensional tooth model by micro-computed tomography and application for data sharing. *Clin Oral Invest* 2009; 13: 43-46.
- 24) Rhiemora P, Haddawy P, Khanal P, Suebnukarn S, Dailey MN. A virtual reality simulator for teaching and evaluating dental procedures. *Methods Inf Med* 2010; 49: 396-405.
- 25) Nagasawa S, Yoshida T, Tamura K, Yamazoe M, Hayano K, Arai Y, Yamada H, Kasahara E, Ito M. Construction of database for three-dimensional human tooth models and its ability for education and research—Carious tooth models. *Dent Mater J* 2010; 29: 132-137.
- 26) Smelyanskiy M, Holmes D, Chhugani J, Larson A, Carmean DM, Hanson D, Dubey P, Augustine K, Kim D, Kyker A, Lee VW, Nguyen AD, Seiler L, Robb R. Mapping high-fidelity volume rendering for medical imaging to CPU, GPU and many-core architectures. *IEEE Trans Vis Comput Graph* 2009; 15: 1563-1570.
- 27) Tanzawa T, Futaki K, Tani C, Hasegawa T, Yamamoto M, Miyazaki T, Maki K. Introduction of a robot patient into dental education. *Eur J Dent Educ* 2012; 16: e195-199.
- 28) Nickolls J, Dally WJ. The Gpu computing era. *IEEE Micro* 2010; 30: 56-69.
- 29) Dick C, Georgii J, Westermann R. A hexahedral multigrid approach for simulating cuts in deformable objects. *IEEE Trans Vis Comput Graph* 2011; 17: 1663-1675.
- 30) Laine S, Karras T. Efficient sparse voxel octrees. *IEEE Trans Vis Comput Graph* 2011; 17: 1048-1059.