Journal of Bioinformatics and Neuroscience (JBINS), 1 (1): 38-44, 2015
ISSN 2188-8116
Applied Science and Computer Science Publications

Experimental Results of 2D Depth-Depth Matching Algorithm Based on Depth Camera Kinect v1

Hiroshi Noborio, Kaoru Watanabe, Masahiro Yagi, Yasuhiro Ida, Shigeki Nankaku, Katsuhiko Onishi, Masanao Koeda, Masanori Kon, Kosuke Matsui and Masaki Kaibori

Abstract: Last year, we proposed a smart transcription algorithm in which a real liver is captured using a 3D depth camera. As opposed to this, a virtual liver is represented by a polyhedron in STL (Standard Triangulated Language) format (stereo-lithography) via DICOM (Digital Imaging and Communication in Medicine) data captured by MRI (magnetic resonance imaging) and/or a CT (computed tomography) scanner. By comparing the depth image in the real world and the Z-buffer in its virtual world, we quickly identify translation/rotation differences between real and virtual livers in a GPU (graphics processing unit). Then by a randomized steepest descent method based on the differences, we can quickly copy real liver motion to virtual liver motion. In this paper, this performance (i.e., motion precision and calculation time) of the proposed algorithm is ascertained from several kinds of experiments based on the depth camera Kinect v1. This is the first challenge to use matching real-virtual-depth-images in our algorithm running in 3D AR (augmented reality) with overlapping real and virtual environment.

Keywords: Depth camera image, Z-buffer, steepest descent method, GPU, Parallel processing

I. INTRODUCTION

Recently, authors of this paper have been developing a surgical navigation system based on a GPGPU (general purpose graphics processing unit). The purpose of using a GPU (graphics processing unit) is to have real-time navigation in which a doctor operates on a human organ in a real and virtual environment simultaneously based on AR (augmented reality) [1-2]. This is the reason why we are now preparing several types of software and hardware. One method involves exactly overlapping a real and virtual liver in the real and in a virtual 3D environment. This is an image-based position/orientation adjustment system, and therefore the CG (computer graphics) virtual world artificially captured by a Z-buffer of a GPU should be coincident with a real world image captured by a real-depth camera [3]. In addition, we have developed a fast algorithm using parallel processing of GPU for a scraper tip to calculate the Euclidean distances against cancer area and/or blood vessels [4]. In the last decade, many researchers have tried to design several types of fast and robust approaches for surface registration [5-11]. In this paper, we propose a new type of approach in which motions of a real liver in an operating room are copied onto motions of a virtual liver in a PC. This motion transfer function is quite important for constructing a surgical navigation system. In our algorithm, we successively find a position/posture which is most likely to exist by searching the minimum of differences between depth image in a real world and the Z-buffer in a virtual world. For this purpose, we propose an algorithm that evaluates its performance by master and slave virtual livers in a PC. The master liver corresponds to a real liver captured by a 3D depth camera, whereas, the slave liver corresponds to a virtual liver calculated by the Z-buffer of a GPU. In [12], we proposed an original framework with reference to a 2D depth-depth matching method based on a real depth image and the Z-buffer of a GPU (the virtual depth image), and that algorithm adequately evaluates in a master-slave liver simulation. In [13], we investigated the algorithm's performance in a master-slave liver simulation by varying the search space to a smaller or larger value which consists of 3 degrees of translational freedom and 3 degrees of rotational freedom. Furthermore in this paper, we first use the algorithm in a real world; tthat is, a real liver printed by a 3D printer and its virtual liver is modeled by STL (stereo-lithography). Then the 3D real plastic liver is

Corresponding Author: Hiroshi Noborio. Department of Computer Science, Osaka Electro-Communication University, Kiyotaki 1130-70, Shijo-Nawate, Osaka, 575-0063, Japan, Tel: +81(0)72-876-3317, Fax: +81(0)72-876-3321, E-mail: nobori@isc.osakac.ac.jp

manipulated by human such as doctor for several kinds of directions and orientations. Finally, we check the following performance of the 3D virtual liver by 2D depth-depth surface matching. Consequently, we ascertain that our algorithm has some useful possibility of translational and rotational motion in a surgical navigation.

Section 2 of this paper describes our real and virtual livers, and then we describe our real-virtual 2D depth-depth matching algorithm in section 3.In section 4, we give three kinds of experimental results when a real liver moves along the X, Y, and Z axes, and we give one experimental result when a real liver moves around the Z axis. Finally in section 5, we conclude our research.

II. LIVER MODELLING

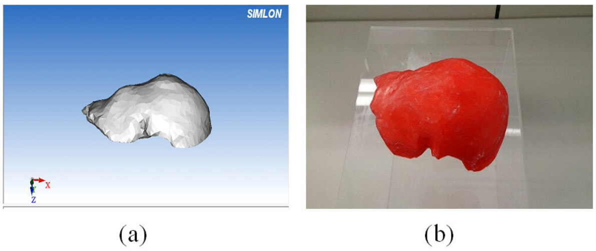


Fig.1 (a) A polyhedral liver with STL format, (b) its 3D printed plastic liver.

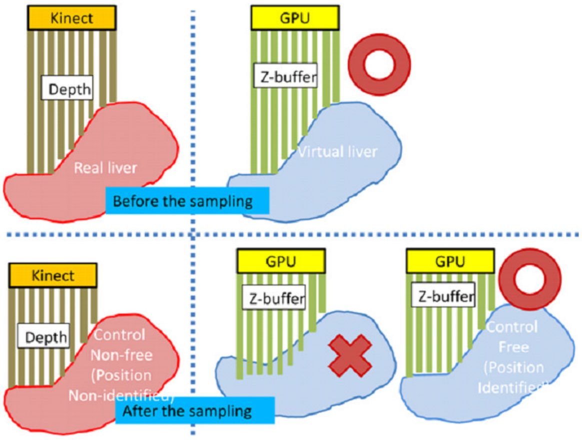


Fig.2 By matching real and virtual depth images, we can find a better position/orientation of virtual liver against that of real liver.

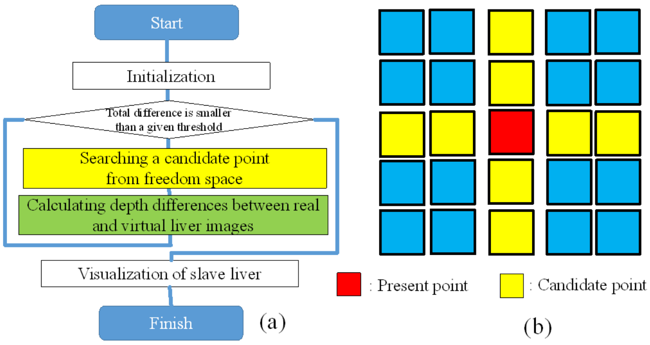

Start
Initialization
Total difference is smaller than a given threshold
Searching a candidate point from freedom space
Calculating depth differences between real and virtual liver images
Visualization of slave liver
Finish
(a)
🟥 : Present point
🟨 : Candidate point
(b)

Fig.3 (a) The flowchart of our depth-depth matching algorithm (b) 24 neighbor candidates whose dimension is 6 represented by 3 translational degrees-of-freedom and 3 rotational ones.

In this algorithm, a liver of a patient is captured as DICOM data by MRI/CT, and the DICOM data is converted into polyhedron with STL format. The reason for using STL is to efficiently maintain visible quality and also to quickly calculate a depth image as the Z-buffer in the GPU (Fig.1). The STL is used for a virtual liver. A 3D printer for the STL is used to construct a plastic liver as a real liver. Uunfortunately this does not have any elastic or viscoelastic property. Thus, in this paper, we check all motion transcription excluding the deformation of the liver.

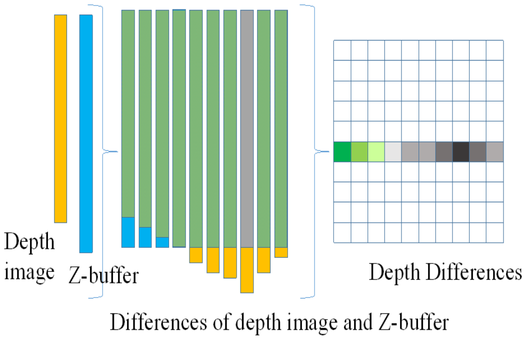


Fig.4 The sum of differences in real and virtual depth images whose pixels are

III. OUR REAL VIRTUAL DEPTH-DEPTH MATCHNIG ALGPRITHM

Our algorithm consists of the following five steps. First, a real plastic liver and its virtual STL liver are initially set at the same position/posture by using our image-based initial position/orientation adjustment system [3].

[Step 1] A human (doctor) operates a real plastic liver printed by 3D printer in the experiment.

[Step 2] The real plastic liver is captured as a rectangular parallelepiped group by the 3D depth camera Kinect v1.

[Step 3] The virtual liver STL is moved to the best position/posture of 24 ones neighboring the present position/posture (Fig.2).

Each virtual liver moves in a motion space with 6 degrees-of-freedom (X, Y, and Z translational freedoms and roll, pitch, and yaw rotational freedoms). In our algorithm (a randomized steepest descendent algorithm), we always select one of the three kinds of motions such as positive step, stop, and negative step along each freedom. Therefore, 24(= 6×2×2) candidates exist as a neighbor for a present candidate (Fig.3).

(3-1) for each candidate, the virtual liver STL is captured by the Z-buffer of the GPU.

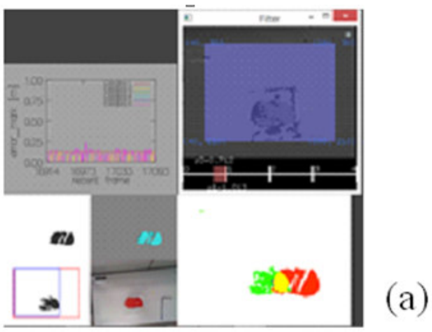
(3-2) for each candidate, we compare real and virtual depth images from the depth camera Kinect v1 and the Z buffer of the GPU. In general, the camera coordinate system (camera parameters) of Kinect v1 is known. Therefore, we adjust the CG coordinate system (CG parameters) to the Kinect's coordinate system. Here, the rendering resolutions of CG are set as 320×240 pixels which is that of Kinect v1 (Fig.4).

(3-3) we pick up 20 pairs of depth images and simultaneously select 100 pixels randomly for each image. Then, from the histogram of depth differences in 100 pixels, we select their average. In succession, from the histogram of averages in 20 images, we then select their medium.

[Step 4] The PC moves a virtual liver STL (a truly virtual liver) according to the selected position/posture.

[Step 5] We return to [Step 1].

IV. Experimental Results



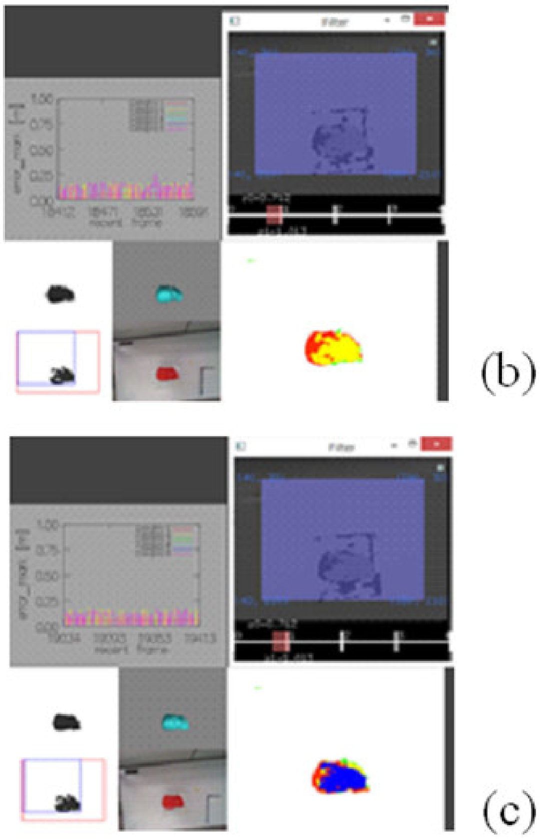


Fig.5 (a) A 2D virtual liver colored red is started to overlap with its 2D real liver colored green in the depth image, (b) the overlapping area in XY plane is colored by a set of yellow pixels, (c) the overlapping area in XYZ space is colored by a set of blue pixels.

In this section, we check that when a doctor operates a red colored 3D printed real liver, whether its yellow colored virtual liver interlocks precisely or not.

First of all, we overlap a real liver by its virtual one in 3D space by watching several kinds of status between real and virtual depth images in 2D space [3]. For this purpose, we prepare several kinds of windows. In the upper left of Fig.5 (a), (b), and (c), we can find position errors in X, Y, and Z, and orientation errors in pitch, yow, and roll coordinate systems. In the upper right of Fig.5 (a), (b), and (c), we can select captured XY image and Z direction of real and virtual livers.

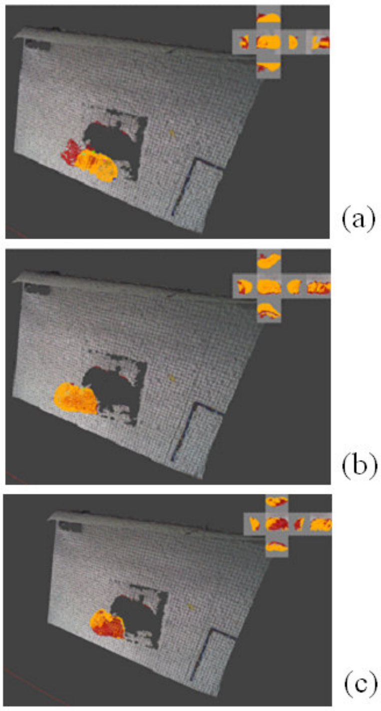


Fig.6 (a) A 3D virtual liver colored yellow is starting to overlap its 3D real liver colored red in the XY plane, (b) the overlapping is finished in the XY plane, (c) the overlapping is also finished along the Z axis.

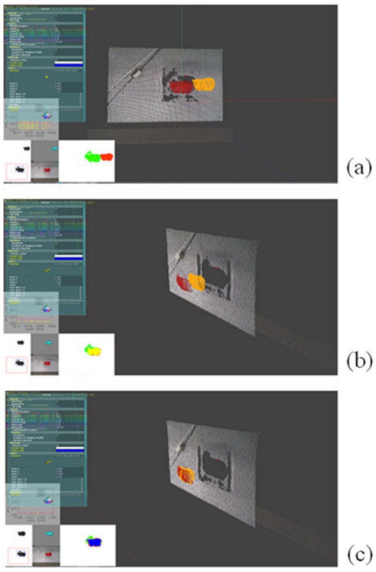


Fig.7 (a) A 3D virtual liver colored by yellow is started to overlap with its 3D real liver colored by red in the XY plane, (b) the overlapping is finished in the XY plane (the overlapping area is colored by a set of yellow pixels), (c) the overlapping is also finished along the Z axis (the overlapping area is colored by a set of blue pixels).

Furthermore, a 3D virtual liver colored yellow is starting to overlap its 3D real liver colored red in the XY plane in Fig.6 (a) and the overlapping is finished in the XY plane in Fig.6 (b); then the overlapping is also finished along the Z axis in Fig.6 (c). Finally, a 3D virtual liver colored yellow is starting to overlap its 3D real liver colored red in the XY plane in Fig.7 (a), and the overlapping in 3D is finished in the XY plane (the overlapping area in 2D is colored by a set of yellow pixels) in Fig.7 (b); then the overlapping in 3D is also finished along the Z axis (the overlapping area in 2D is colored by a set of blue pixels) in Fig.7(c). This is the initial adjustment system in which a 3D real liver is synchronized to its 3D virtual liver in a 3D environment (a surgical operation room). In the second row of Table 1 and the second column of Table 2, we show the initial identification ratio of a virtual liver against its real liver. The initial identification ratio is defined as the number of 2D blue pixels overlapping between 3D real and virtual livers to the number of 2D pixels projected from a 3D

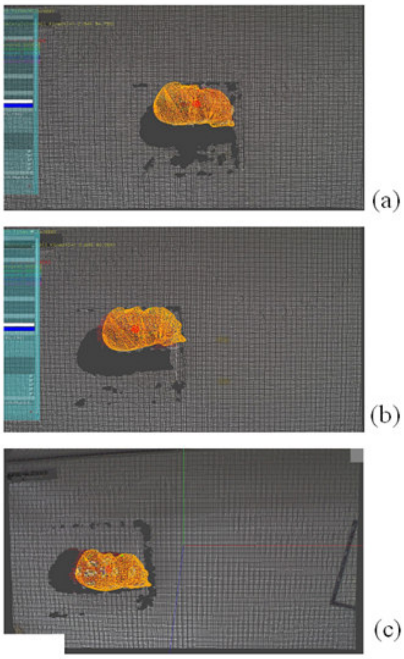


Fig.8 a 3D virtual liver colored yellow translationally follows its 3D real liver colored red which moves along X axis. The overlapping area in 3D XYZ space is colored by a set of blue pixels. (a) Initial position. (b) Final position where speed is 15 cm/10 s. (c) Final position where speed is 15 cm/20 s.

Virtual liver, which are colored by a set of green pixels. Since the initial identification ratio is to be 94% in the second row of Table 1 and the second column of Table 2, real and virtual livers are completely overlapped in a 3D environment.

After the near-optimal coincident is found, we move the real liver along the X, Y, and Z axis by using two kinds of velocities. We check whether the virtual liver follows the real liver by our proposed 2D depth-depth matching algorithm based on the depth camera Kinect v1.

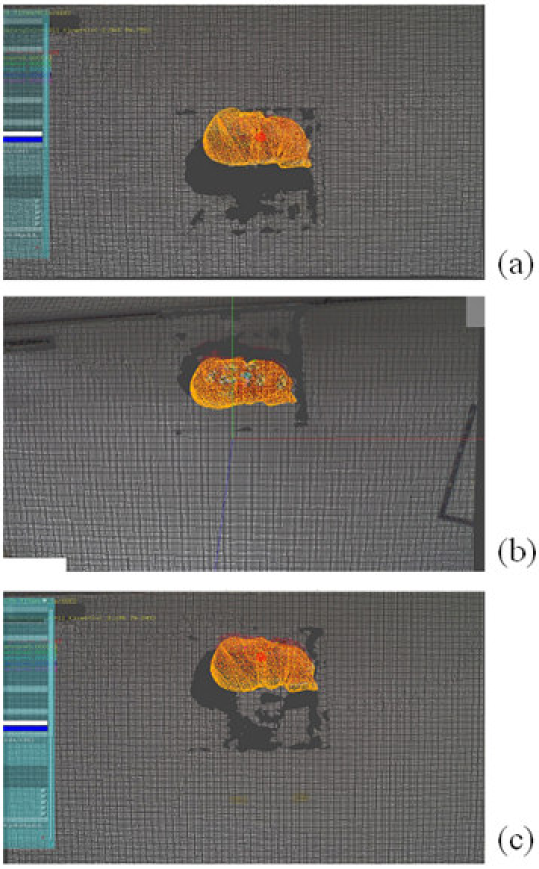


Fig.9 A 3D virtual liver colored yellow translationally follows its 3D real liver colored red which moves along Y axis. The overlapping area in 3D XYZ space is colored by a set of blue pixels. (a) Initial position. (b) Final position where speed is 15cm/10second. (c) Final position where speed is 15cm/20second.

First of all, in Fig.8 and Table 1, a 3D virtual liver colored yellow translationally follows its 3D real liver colored red which moves along the X axis. The overlapping area in 3D XYZ space is colored by a set of blue pixels. In Fig.8(a), we describe the initial position, and in Fig.8(b), we describe the final position where speed is 15 cm/10 s, and also in Fig.8(c), we indicate the final position where the speed is 15 cm/20 s. Since the final identification ratio is to be 81%, as listed in the third column of Table 1, real and virtual livers are almost overlapped in the 3D space.

Secondly in Fig.9 and Table 1, a 3D virtual liver colored yellow translationally follows its 3D real liver colored red which moves along the Y axis. Similar to the above step, the overlapping area in the 3D XYZ space is colored by a set of blue pixels. First, we show the initial position in Fig.9 (a). In Fig.9 (b), we indicate the final position where speed is 15 cm/10 s. In this case, the final identification ratio is to be 76%,

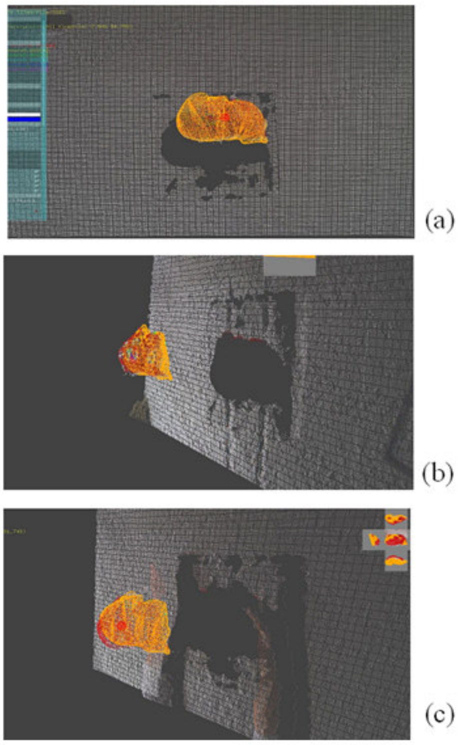


Fig.10 A 3D virtual liver colored by yellow translationally follows its 3D real liver colored by red which moves along Z axis. The overlapping area in 3D XYZ space is colored by a set of blue pixels. (a) Initial position. (b) Final position whose speed 15cm/10second. (c) Final position whose speed 15cm/20second.

as listed in the third column of Table 1. Simultaneously, we describe the final position where speed is 15 cm/20 s. In this case, the final identification ratio is to be 89% in the third column of Table 1. Therefore, the faster the translational movement speed of real liver, the lower the overlapping precision.

Third, in Fig.10 and Table 1, a 3D virtual liver colored yellow translationally follows its 3D real liver colored red which moves along the Z axis. Similar to the previous case, the overlapping area in the 3D XYZ space is colored by a set of blue pixels. First, we show the initial position in Fig.10 (a). In Fig.10 (b), we indicate the final position where the speed is 15 cm/10 s. In this case, the final identification ratio is to be 70%, as listed in the third column of Table 1. Simultaneously, we describe the final position where the speed is 15 cm/20 s. In this case, the final identification ratio is to be 83%, as shown in the third column of Table 1. Therefore, the faster the translational speed of the real liver, the lower the overlapping precision.

Table 1. Initial and final identification ratio of fast and slow translational movements along X, Y, and Z axes.

	Translation speed with 15 cm/10 s	**Transl 1**
Initial identification ratio	94%	
Final identification ratio along X axis	81%	
Final identification ratio along Y axis	76%	89%
Final identification ratio along Z axis	70%	83%

Table 2. Final identification ratio of fast and slow rotational movements around the Z axis.

	Initial identification ratio	**Final identification ratio around Z axis**
Rotation speed with 45° /10 s	94%	65%
Rotation speed with 45° /20 s	94%	81%

Finally in Fig.11 and Table 2, a 3D virtual liver colored yellow rotationally follows its 3D real liver colored red which moves around Z axis. Similar to the previous case, the overlapping area in the 3D XYZ space is colored by a set of blue pixels. First, we show the initial position in Fig.11 (a). In Fig.11 (b), we indicate the final position where speed is 45°/10 s.

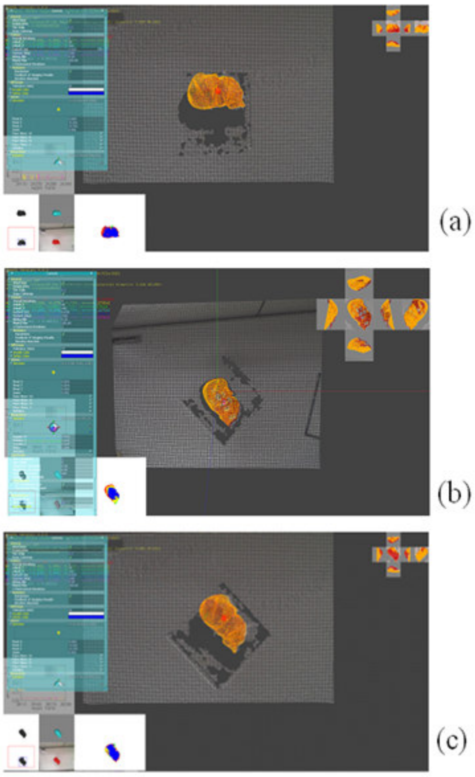


Fig.11 A 3D virtual liver colored yellow rotationally follows its 3D real liver colored red which moves around Z axis. The overlapping area in 3D XYZ space is colored by a set of blue pixels. (a) Initial position. (b) Final position where speed is 4.5°/s. (c) Final position where speed is 2.25 in the clockwise direction.

In this case, the final identification ratio is to be 65%, as presented in the third column of Table 2. Simultaneously, we describe the final position where the speed is 45° /20 s. In this case, the final identification ratio is to be 81%, as listed in the third column of Table 2. Therefore, the faster the rotational speed of real liver, the lower the overlapping precision.

In the several kinds of translation or rotational movements shown above, the real and virtual livers almost overlapped in the 3D space, and therefore the usefulness of our proposed 2D depth-depth matching algorithm based on depth camera Kinect v1 is verified in these experiments.

V. CONCLUSIONS

In this paper, by efficiently matching a real depth image captured using the popular depth camera Kinect v1 with a virtual depth image in the Z-buffer of GPU, we designed a fast motion transcription algorithm between real and virtual livers. In succession, we evaluated the algorithm's efficiency and its motion precision in several kinds of experiments. As a result, we can understand that our fast 2D depth-depth matching algorithm has the potential to execute motion transcription of human liver in surgical navigation.

ACKNOWLEDGEMENTS

This is supported in part by 2014 Grants-in-aid for Scientific Research (No. 26289069) from the Ministry of Education, Culture, Sports, Science and Technology, Japan. This is also supported in part by 2014 Cooperation research fund of graduate school in Osaka Electro-Communication University.

REFERENCES

[1] K.Onishi, H.Noborio, M.Koeda, K.Watanabe, K.Mizushino, T.Kunii, M.Kaibori, K.Matsui, M.Kwon. Virtual liver surgical simulator by using Z-buffer for object deformation. *Universal Access in Human-Computer Interaction* (In Proc. of HCII 2015), Part III, LNCS 9177, pp. 345–351, Los Angeles, CA, USA, Aug. 2015.

[2] M.Koeda, A.Tsukushi, H.Noborio, K.Onishi, K.Mizushino, T.Kunii, K.Watanabe, M.Kaibori, K.Matsui, M.Kwon. Depth Camera Calibration and Knife Tip Position Estimation for Liver Surgery Support System. In Proceedings of the 17th Int. Conf. on Human-Computer Interaction (HCII2015), Posters, Part I, CCIS 528, pp. 496–502, Los Angeles, CA, USA, Aug. 2015.

[3] H.Noborio, K.Watanabea, M.Yagi, Y.Ida, K.Onishi, M.Koeda, H.Nankaku, M.Kaibori, K.Matsui, M.Kon. Image-based Initial Position/Orientation Adjustment System between Real and Virtual Livers. *Jurnal Teknologi, Medical Engineering*, Vol.77, No.6, pp.41-45,DOI: 10.11113/jt.v77.6225, Penerbit UTM Press, 2015.

[4] H.Noborio, T.Kunii, K.Mizushino. GPU-Based Shortest Distance Algorithm For Liver Surgery Navigation. In Proceedings of the 10th Anniversary Asian Conference on Computer Aided Surgery, Kyusyu University, Japan, pp.42-43, 2014.

[5] P. J. Besl and N. D. McKay. A Method for Registration of 3-D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.14, issue 2, pp.239-256, 1992

[6] Z. Zhang. Iterative Point Matching for Registration of Free-Form Surfaces. *Int. Journal of Computer Vision*, vol.13, issue 2, pp.119–152, 1994.

[7] S. Granger, X. Pennec. Multi-Scale EM-ICP: A Fast and Robust Approach for Surface Registration. In Proceedings of the 7th European Conference on Computer Vision, No.4, pp.69-73, 2002. Y. Liu. Automatic Registration of Overlapping 3D Point Clouds Using Closest Points," *Journal of Image and Vision Computing*, vol.24, issue 7, pp.762-78, 2006.

[8] J.Salvi, C.Matabosch, D.Fofi, J.Forest. A review of recent range image registration methods with accuracy evaluation. *Journal of Image and Vision Computing*, vol.25, pp.578-596, 2007.

[9] R. B. Rusu and S. Cousins. 3D is here: Point Cloud Library (PCL). In Proceedings of the the IEEE Int. Conf. Robotics and Automation, pp.1-4, 2011.

[10] Y.F.Wu, W.Wang, K.Q.Lu, Y.D.Wei, Z.C.Chen. A new method for registration of 3D point sets with low overlapping ratios. Procedia CIRP 27 (the 13th CIRP conference on Computer Aided Tolerancing), pp.202-206, 2015.

[11] H.Noborio, K.Onishi, M.Koeda, K.Mizushino, M.Yagi, M.Kaibori, M.Kwon. Motion Transcription Algorithm By Matching Corresponding Depth Image and Z-buffer. In Proceedings of the 10th Anniversary Asian Conference on Computer Aided Surgery, Kyusyu University, Japan, pp.60-61, 2014.

[12] K.Watanabe, M.Yagi, K.Ota, K.Onishi, M.Koeda, S.Nankaku, H.Noborio, Masanori Kon, Kousuke Matsui, M.Kaibori. Parameter Identification of Depth-Depth-Matching Algorithm for Liver Following. *Jurnal Teknologi, Medical Engineering*, vol.77, issue 6, pp.35-39, DOI:10.11113/jt.v77.6224, Penerbit UTM Press, 2015.