International Journal of New Computer Architectures and their Applications (IJNCAA) 6(4): 150 -155
The Society of Digital Information and Wireless Communications, 2016 ISSN 2220-9085 (Online); ISSN 2412-3587 (Print)

Development of a City Planning Simulation System Using Leap Motion in the AR Space

Toshiharu Katahira and Hiroki Imamura
Department of Information Systems Science, Graduate School of Engineering, Soka University
Mailing Address: 1-236, Tangi-machi, Hachioji-Shi, Tokyo, Japan 192-8577
E-mail: e15m5207@soka-u.jp, imamura@soka.ac.jp

ABSTRACT

Some simulation systems of the city planning in VR space have been proposed. For example, there are landscape simulation system and sunshine simulation systems. However, the conventional simulation system in VR space are limited in the 3DCG of buildings modeled in advance and don't assume that discussion of city planning with an architect in remote locations. In order to solve these issues, we have proposed methods that superimposition virtual models on actual models using AR and object recognition and using COMSAS. In this paper, we report the development and evaluation of a prototype of this city planning simulation system using Leap Motion in AR space which becomes a cornerstone of the proposed system. In this prototype system provides users intuitive manipulation of AR buildings.

KEYWORDS

Virtual Reality, Augmented Reality, City Planning, Simulation, Natural User Interface, Leap Motion, Unity.

1 INTRODUCTION

Recently, city renewal projects are carried out in various area in Japan because of decrepit buildings, anti-disaster and change of the inhabitant's needs. A city planning, such as the city renewal project, advances through consultations between an architect and government officials. They develop an idea of the city project with either making an accurate model of the city or drawing the illustrations of it. Moreover, recently they use a 3DCG model of the city for landscape simulation in virtual reality (VR) space [1][2]. In addition, research on sunshine simulation [3] and wind simulation with visualizing wind [4] have been proceeded. However, the conventional city simulation systems in VR space are limited in the 3DCG of buildings prepared in advance. Furthermore, the conventional system does not assume that discussion of city planning in remote locations. As a solution to these issues, we use two methods. The first method is superimposition virtual models on actual models using Augmented Realty (AR) and object recognition. The second method is using a 3D video communication system in synchronized AR space (COMSAS) [5]. In COMSAS, multiple users at remote locations are reconstructed both each other in the AR space. Owing to these two methods, users can add information of texture and shadow to actual models. Furthermore, it is possible to share virtual models in a remote location. To implement the above system, in this study, we develop a city planning simulation system using Leap Motion [6] in AR space which becomes a cornerstone of the proposed system.

2 PROPOSED SYSTEM

2.1 Equipments in the Proposed System

2.1.1 Leap Motion

We describe about Leap Motion. Fig. 1 shows the appearance of the Leap Motion. The Leap Motion has optical sensors and infrared light. The sensors are directed along the y-axis upward when the controller is in its standard operating position and have a field of view of about 150 degrees. The effective range of the Leap Motion Controller extends from approximately 25 to 600 millimeters above the device. The Leap Motion can obtain information of user's hands (such as three-dimensional coordinates, postures, and gestures) and track user's hands. To realize an intuitive user interface, we use the Leap Motion Controller extends from approximately 25 to 600 millimeters above the device. The Leap Motion can obtain information of user's hands (such as three-dimensional coordinates, postures, and gestures)

Fig. 1: Leap Motion

Fig. 2: The structure of HMD

and track user's hands. To realize an intuitive user interface, we use the Leap Motion.

2.1.2 HMD and Stereo Camera

We describe an outline of the HMD and the stereo camera. We use an Oculus Rift CV1 [7] that is a headset as the HMD. The Oculus Rift CV1 has a stereoscopic OLED display, 1080* 1200 resolution per eye, a 90 Hz refresh rate, and 110 degrees' field of view. The Rift has full 6 degree of freedom rotational and positional tracking. This tracking is performed by a USB stationary infrared sensor that is picking up light that is emitted by IR LEDs. To convert the HMD from VR to AR, we combine Oculus Rift CV1 and Ovrvision Pro [8]. Fig. 2 shows the structure of combined these devises. Ovrvision Pro is a stereo camera that can obtain two RGB images. Ovrvision Pro captures video at up to 60 frames per second.

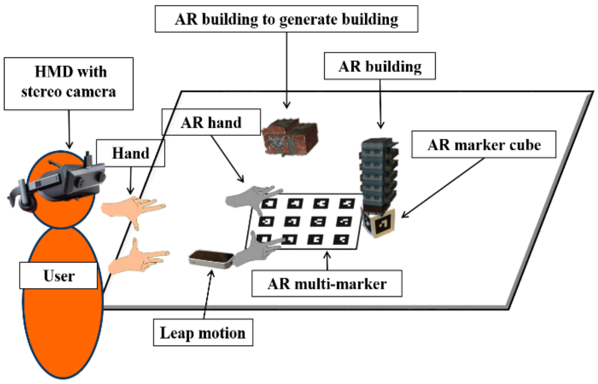


Fig. 3: The configuration of the prototype system

2.2 System Overview

Fig. 3 shows the configuration of the prototype system. The prototype system uses an Oculus Rift CV1 with Ovrvision, an AR multi-marker, an AR marker cube and a Leap Motion[7]. While the Leap Motion is detecting user's hands, the prototype system displays AR hands (means virtual objects represent user's hands). In addition, the prototype system uses Unity [9] to implement the physics simulation. By using collision decision program of Unity, the user can manipulate AR buildings (means virtual objects of buildings) with only the AR hands.

2.3 Processing of the Proposed System

Fig. 4 shows the flow of processing in this prototype system. The descriptions of each step are in following sections.

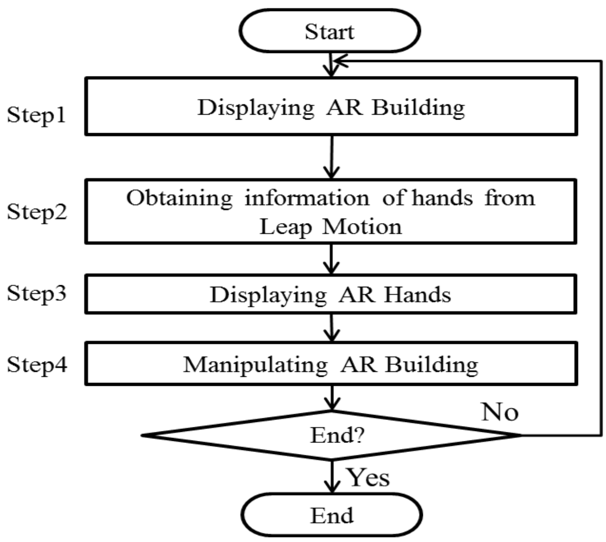


Fig. 4: The flow chart of this system

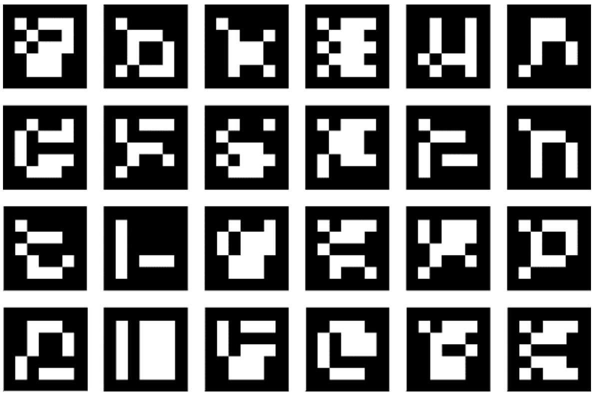


Fig. 5: AR multi-marker.

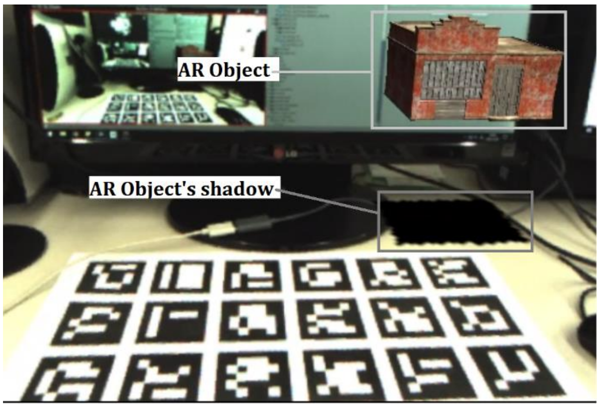


Fig. 6: The displayed AR object

2.3.1 Displaying of AR Buildings

The prototype system displays virtual models based on the position and the inclination of AR multi-maker. Fig. 5 shows the AR multi-marker. The center of AR multi-marker is origin of the virtual models, so that each virtual model is displayed based on the center. Thereby the user can observe the virtual models from various angles by moving own viewing position. In addition, AR multi-marker is robust to occlusion. The prototype system can recognize the AR multi-marker if only one or more markers of it are detected. Fig. 6 shows the displayed AR object. A shadow is drawn under the AR object by the directional light in function of Unity.

2.3.2 Obtaining Information of Hands from Leap Motion

The prototype system obtains information of user's hands from the Leap Motion. Out of those, positions of Fingertips, palms and pinch are needed to display AR hands, translate AR buildings, and change the scale AR buildings.

2.3.3 Displaying AR Hands

While the Leap Motion is detecting hands, the prototype system displays AR hands based on obtained user’s hands information. Fig. 7 shows AR hands.

2.3.4 Manipulating AR Buildings

A collision detecting function between virtual objects is implemented as basic function of Unity. Based on that function, the users can touch and manipulate virtual objects with the AR hands. In following section, we describe 5 manipulations implemented in the prototype system.

2.3.4.1 Generating of AR Buildings

When the user touches a virtual model to generate building with the tip of the forefinger of the AR hand (Fig. 8-a), a new virtual model is generated (Fig. 8-b).

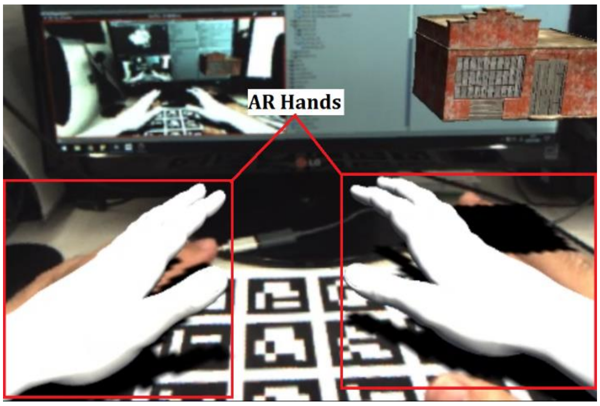
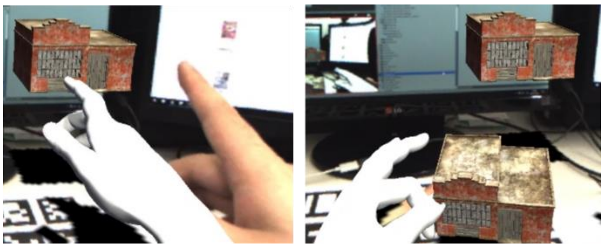
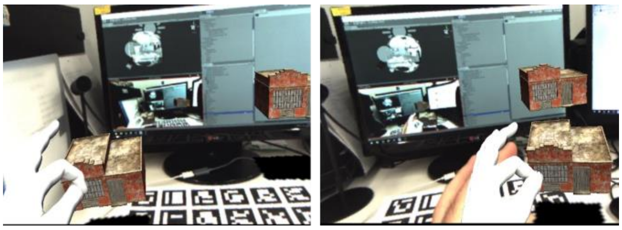


Fig. 7: Displaying the AR hands



(a)Before generate (b) After generate
Fig. 8: Generating AR building



(a) Before translation (b) After translation

Fig. 9: Translating AR building

2.3.4.2 Translating of AR Buildings

The user can pick up an AR building and translate it by pinching with thumb and forefinger of the AR hand. When the user releases the AR building, it falls to ground plane following the gravity. Fig. 9 shows this manipulation. T, a moving amount of the AR object, is determined by using

$$T = \begin{pmatrix} x'_a - (x_a - x_0) \\ y'_a - (y_a - y_0) \\ z'_a - (z_a - z_0) \end{pmatrix}, \qquad (1)$$

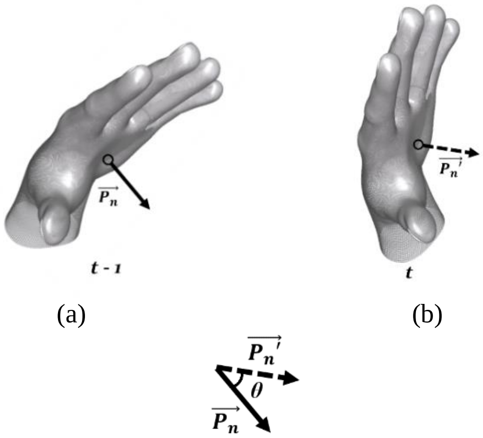
where (x_a, y_a, z_a) is a picking position, (x'_a, y'_a, z'_a) is a position after movement and (x_0, y_0, z_0) is a position before movement of the AR hands.

2.3.4.3 Rotating of AR Buildings

When the user picks up an AR building, the user can rotate the AR building with user' s palm of pinching hand. Fig. 10 shows outline in calculation of the rotation angle of the palm. Firstly, as shown in Fig. 10-a, the proposed system calculates the normal vector $\vec{P_n}$ to the palm from the function of Leap motion when the time is t-1. Secondly, as shown in Fig. 10-b, the proposed system calculates the normal vector $\vec{P}'_n$ to the palm from the function of Leap motion when the time is t. Finally, to calculate the rotation angle of the hand, the proposed system converts 3-dimensional vectors $\vec{P_n}$ and $\vec{P}'_n$ into 2-dimensional vectors P of a X-Z plane and P' respectively and calculates the rotation angle θ of the hand by using

$$\theta = \sin^{-1} \frac{P \cdot P'}{|P| \cdot |P'|}. \qquad (2)$$

Fig. 11 shows this manipulation.



(a) (b)

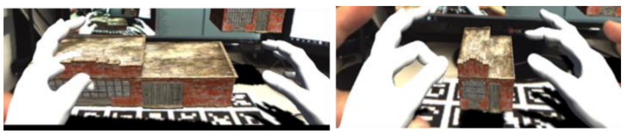
(c)

Fig. 10: Calculation the angle of rotation of the palm

Fig. 11: Rotate AR building

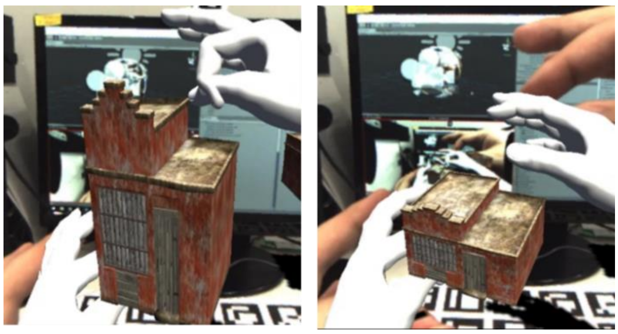
2.3.4.4 Changing Scale of AR Buildings

In a state of picking AR building, the user can change the scale of AR building such as expanding (Fig. 12-a, 13-a, 14-a) and shrinking (Fig. 12-b, 13-b, 14-b) a rubber band with both hands. The scale of AR buildings can be changed in each xyz-axis direction.



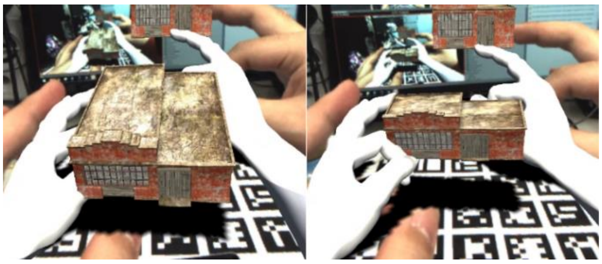
(a) Scale up (b) Scale down

Fig. 12: Changing scale AR building in x axis

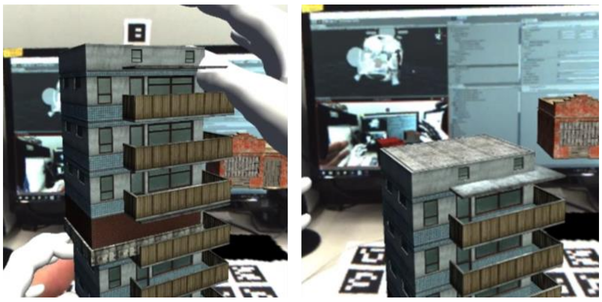


(a) Scale up (b) Scale down

Fig. 13: Changing scale AR building in y axis



(a) Scale up (b) Scale down
Fig. 14: Changing scale AR building in z axis



(a) Increasing story (b) Decreasing story
Fig. 15: Changing Story of AR Buildings

The positional relationship between both AR hands determines which scale of axis will be changed. The change amount of the scale of AR buildings is determined by varying of the distance of AR hands. When the distance of AR hands at the picked moment is larger than the distance of AR hands at the current moment, the AR building is expanding. When the distance of AR hands at the picked moment is smaller than the distance of AR hands at the current moment, the AR building is shrinking. *S*, a change amount of the scale of AR buildings, is determined by using

$$S = \begin{pmatrix} |(x'_{l2} - x'_{r1}) - (x_{l2} - x_{r1})| \\ |(y'_{l2} - y'_{r1}) - (y_{l2} - y_{r1})| \\ |(z'_{l2} - z'_{r1}) - (z_{l2} - z_{r1})| \end{pmatrix}, \qquad (3)$$

where (x_{r1}, y_{r1}, z_{r1}) is position of right AR hand picking a AR building,(x_{l2}, y_{l2}, z_{l2}) is position of left AR hand picking the same AR building, $(x'_{r1}, y'_{r1}, z'_{r1})$ and $(x'_{l2}, y'_{l2}, z'_{l2})$ are position of after movement of each right and left AR hand.

2.3.4.5 Changing Story of AR Buildings

A displayed AR building on the AR marker cube can be changed the story of it, such as in Fig. 15-a and Fig. 15-b. When user performs expanding manipulation in height direction, the same AR building is added on the top of the AR building. Conversely, when the user performs shrinking manipulation in height direction, the added AR building on the top is deleted.

3 EVALUATION EXPERIMENTS

3.1 Outline of the Experiment

We had an evaluation experiment on the proto type system. We evaluated whether the implemented function behave correctly and this system is intuitive or not. A user wears a HMD with a Ovrvision on and manipulates AR hands with his/her hands above the Leap Motion, and then, the user performs manipulation of translating and changing story of AR buildings. 10 subjects performed the experiment. After that, they evaluated following items. 10 people performed the experiment. After that, they evaluated following items.

① Could you pick up AR buildings?
② Could you translate AR buildings?
③ Could you rotate AR buildings?
④ Could you expand AR buildings?
⑤ Could you shrink AR buildings?
⑥ Could you increase story of AR building on the AR marker cube?
⑦ Could you decrease story of AR building on the AR marker cube?
⑧ Did you feel this system is intuitive?

Each evaluation is 5 levels as the highest score is 5.

3.2 Results of Experiments

Table 1 shows the result of the evaluation experiment. Average scores from ① to ③ are higher than 4.0. Among all the items, items ① and ② has the highest average score and the lowest standard deviation, conversely, the item ⑦ has the lowest average score and the highest standard deviation.

3.3 Discussion of Experiments

Average scores of items ⑥ and ⑦ are lower than 3.5 and the standard deviation is higher than 1.0.

Table 1. Evaluation experiments

No.	①	②	③	④	⑤	⑥	⑦	⑧
Average Score	4.60	4.60	4.10	3.70	3.70	3.30	3.10	4.00
Standard Deviation	0.66	0.66	0.70	0.46	0.46	1.10	1.22	0.77

It is considered that the cause is unstable displaying the AR building on the AR marker cube in manipulation of the hanging the story. An area of AR marker cube visible with a Ovrvision is smaller than the area of AR multi-marker, therefore, the AR marker cube is easy to be invisible with the Ovrvision by hands of the subject in manipulation of changing the story. Accordingly, it is seeming to be difficult for the subject to perform manipulation of changing the story. In order to solve this problem, we will implement a method of object recognition with 3D sensor. The average score of the item ⑧ is higher than 4.0 and the standard deviation of that is lower than 0.8, therefore, it can be said that most subjects feel that the prototype system is highly intuitive.

4 CONCLUSION

In this paper, we showed the outline of the city planning simulation system using a Leap motion, Ovrvision and Oculus Rift CV1. Moreover, we implemented and evaluated the intuitive manipulations of AR buildings. As the result, we found the prototype system is intuitive. However, we found a problem of the changing story of AR building displayed on AR marker cube. In the future, we will solve this problem by implementing shape recognition with 3D sensor. As future work, we will try to implement shape recognition with 3D sensor. In addition, we will try to implement sunshine simulation and wind simulation. Moreover, we will consult an expert on the city planning and determine the finally function of this system.

REFERENCES

[1] Construction of the creation and design consultation system of vision using landscape simulation in around Kashiwa Station, http://www.mlit.go.jp/common/000112023.pdf [Accessed 12 January 2017]

[2] KAWAI Yasuo, SANO Masami, IKEDA Takeshi, and MASUOKA Ryo, “Landscape simulation system using a game engine for landscape planning” Summaries of technical papers of annual meeting 2013, 93-94, 2013-08-30

[3] Noriyoshi Fujisawa, Takamasa Miyazaki, and Kiminori Nakazawa “A Study on Landscape Architecture Design using BIM and GIS” Architectural Institute of Japan, Vol. 21, No. 47, pp.355-360, 2015

[4] Visualization by stereoscopic vision based on VR technology, http://www.civil.chuou.ac.jp/lab/keisan/vr1.html [Accessed 12 January 2017]

[5] Hideyuki Hashimoto, Yuki Fujibayashi and Hiroki Imamura, "3D Video Communication System by Using Kinect and Head Mounted Displays", International Journal of Image and Graphics, Vol.15, No.2, pp.1540003-1-1540003-12, 2015

[6] Leap Motion. Inc. https://developer.leapmotion.com/ [Accessed 12 January 2017]

[7] Oculus Rift CV1. https://www3.oculus.com/en-us/rift/ [Accessed 12 January 2017].

[8] Ovrvision Pro. http://ovrvision.com/ [Accessed 12 January 2017].

[9] Unity Technologies. http://unity3d.com/jp/unity [Accessed 12 January 2017]