Virtual Reality Contents using the OculusLift and Kinect

Dongik Lee, Giyeol Baek, Yangwon Lim and Hankyu Lim(corresponding author)
Department of Multimedia Engineering
Andong National University
Andong-si, Republic of Korea
yoha3567@naver.com, qkkndq@naver.com, limyw@anu.ac.kr, hklim@anu.ac.kr

Abstract—Recently, in the game market, new kinds of peripheral devices in the controller and display area are being developed and games using them are being produced. Controllers diversely utilized in recent times are motion recognition controllers such as Kinect and Leap Motion, and as a display there is head mounted display (HMD). There have been various games and contents utilizing such peripheral devices but development of games and contents using them in combination is low. This paper designed and embodied virtual reality contents using Kinect, a motion recognition controller most actively used recently, and Oculus Lift, an HMD.

Keywords—*VR contents; HMD; oculus lift; kinect*

I. Introduction

Recently, motion recognition controllers and relevant controllers are being developed and sold through many home console game devices and personal computers (PCs) [1]. Among them, Kinect supports both Xbox and PCs and is utilized in diverse areas such as medicine and education as well as the game area. However, they are not being actively developed yet in other than the game area owing to the low recognition rate and level of awareness.

In the case of HMDs, at an early stage, wearing them was uncomfortable because of their size and weight and they had many problems such as low resolution and expensive price of displays, but thanks to the development of displays and improvement in their performance, diverse HMDs such as Sony's HMD, Oculus' Oculus Lift, and Samsung's Gear VR have been developed and the growth of the relevant market has been continuing [2].

This paper designed and embodied virtual reality contents using together Kinect, a motion recognition controller, and Oculus Lift, an HMD. The contents were to embody prototypes which moved characters utilizing Kinect and made display with Oculus Lift using C++ and DirectX11. Through this, developing the virtual reality game and contents proposed in the paper is expected to be possible.

II. The Related Research

When the motion of the user is recognized using Kinect, the location information values of the user's skeleton may be derived. Information on the skeleton is composed of a total of 20 joints including humans' hands, feet, and head [3]. We utilized location information values of these joints and used them in order to move characters.

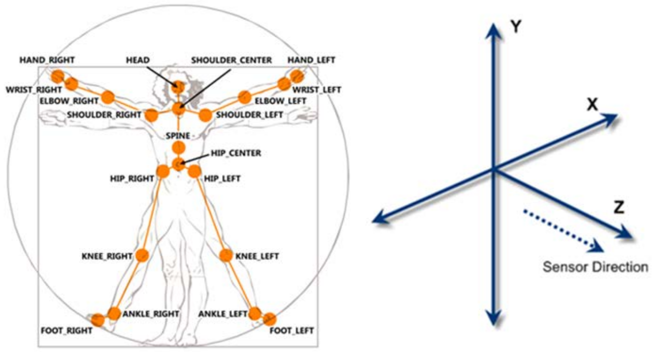

Fig. 1. Skeleton and Joint of Kinect

Although countless HMDs have been developed since the 1990s until now, the unique characteristics of Lift are wide field of view and rapid response speed [4]. There are two methods to create a game supporting Oculus Lift. When developing a game, the rendering screen of framework is set as Oculus Lift, and View Matrix and Projection Matrix, corresponding with the left and right eyes, are calculated, and fish eye rendering is made for each of the left and right screens, and there is a method of rendering by porting the existing game to be suitable for the Oculus Lift. vorpX is not supported for all games and in order to utilize it, the relevant program should be purchased separately. Here, fish eye rendering is to distort the screen and perform rendering in order to express as if one looks with the eyes, and wide field of view may be obtained through the convex lens. However, owning to the problem of pixel density by fish eye rendering, the pixel of the center area has relatively small image quality decline but the periphery has degraded image quality and during rendering the calculation ability to make fish eye rendering of the screen is needed [5].

ISBN: 978-1-61804-327-6

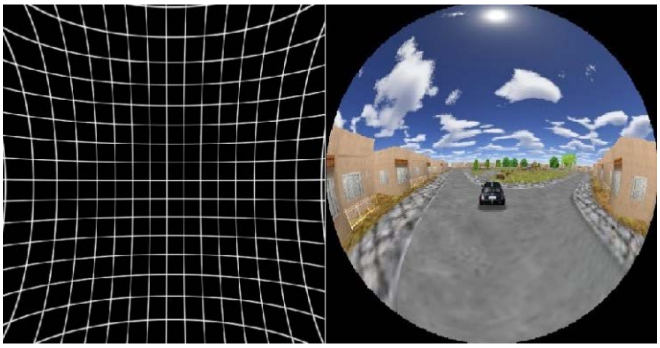

Fig. 2. Fish Eye Rendering

Another function of the Oculus Lift is head tracking, which is a function of applying changes in field of vision within the game and contents by sensing the head's rotation and forward and backward and up and down movements by the Oculus Lift.

III. Function and Implementation

A. *Function Analysis*

The functions of the program are divided into three in order to perceive a user's motions through this Kinect and utilize the perceived motions as data to move characters. First, Kinect has a function of perceiving a user's motions and manipulating characters of the game and is divided into head tracking of Oculus Lift, display, and game and contents. The figure 3 below is the perspective plan of virtual reality contents utilizing the Oculus Lift and Kinect.

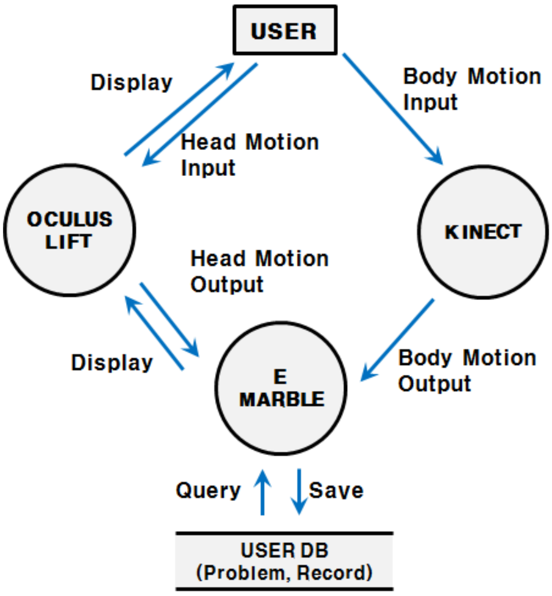

Fig. 3. DFD Structure_perspective plan

In the perspective plan of Figure 3, E Marble is the name of contents this paper designed and embodied. It is a main function of the game which is to bring information on the skeleton of the Kinect, manipulate characters and the game, bring information on head tracking from the Oculus Lift, calculate View Matrix and Projection Matrix, and perform rendering them to the Oculus Lift. Owing to the characteristics of the controller and the display, each function has high degree of mutual coupling.

B. *Kinect Implementation*

In order to move characters in the same way as the movements of the user, information on skeleton joints read from the Kinect should be applied to the character mesh's bones by calculating rotational matrix of each joint based on the information on the skeleton's joints. The Figure 4 below represents composition to correspond the character's bones with the Kinect's joints, and Kinect Tutorial and Avaterring Sample Source were referred to in order to derive the formula to calculate each joint's rotational matrix with information on joints.

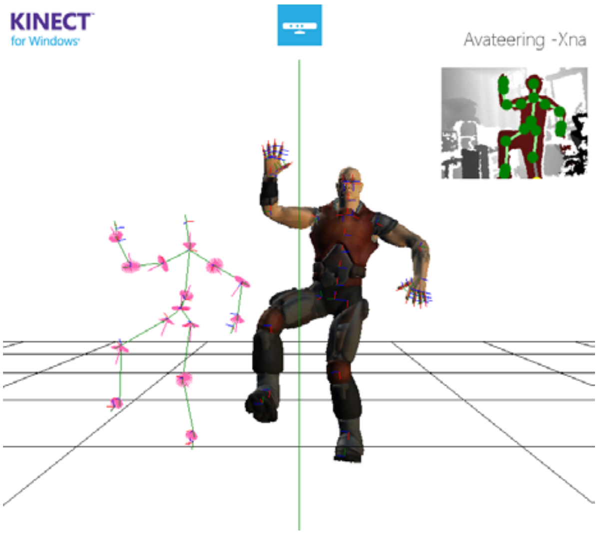

Fig. 4. Screen of Avateering C# Sample

(Left) Skeleton & Joint, (Middle) Character, (Right) Player

The Figure 5 below represents Kinect Skeleton Joint and Implementation Character Bone. Bone matrix calculated in such a way may be applied to character mesh through Skinning Animations.

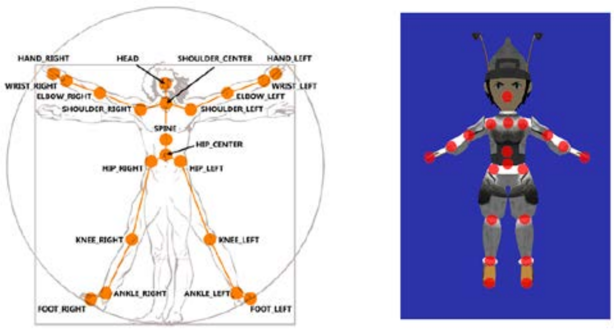

Fig. 5. Kinect Skeleton Joint and Implementation Character Bone

ISBN: 978-1-61804-327-6

C. *Event Handling*

The game embodied in this paper enabled virtual reality experience by using Kinect and Oculus Lift, but it is impossible to employ a mouse or keyboard for game manipulation. Therefore, manipulation should be made using Kinect as well. Such event was processed by setting a collider on the hands in the character's mesh and composing an object for manipulation and then detecting mutual collision.

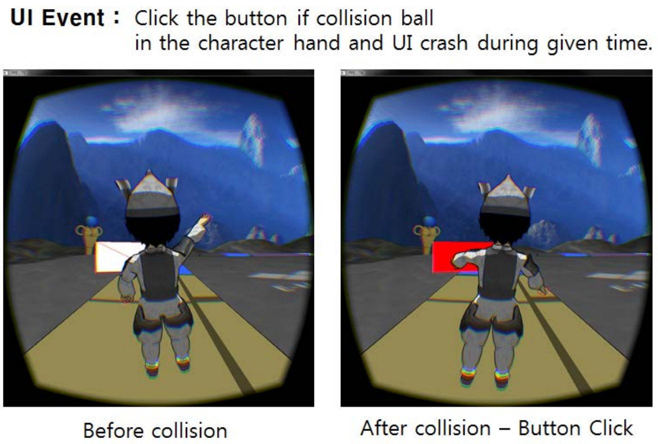

Fig. 6. UI Event Handling

Figure 6 above shows establishment of OBB(Oriented Bounding Box) aimed at event processing to the plan for UI output and event processing by testing collision with the sphere in the hands of the characters. The relevant character moves to the movement of a user and it was embodied for the user to click the button by extending the hand in order to trigger an event.

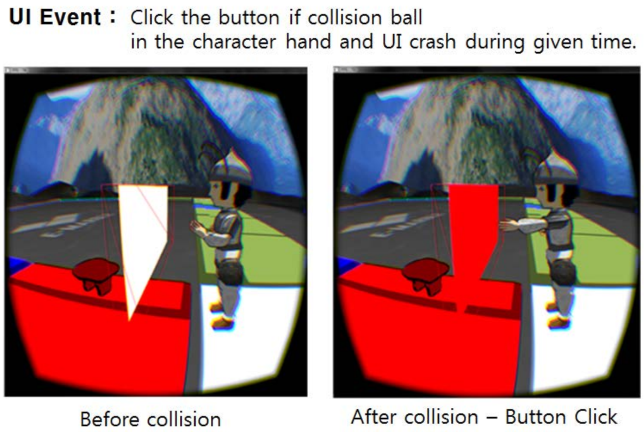

Fig. 7. UI Event Handling2 : View from Side

IV. CONCLUSION

At present, HMD and motion recognition controller have grown enough to produce games and contents, but still they have problems of high prices, low performance, and lack of key contents. Nonetheless, research and development on the area of virtual reality is continuously being made and performance of peripheral devices is being improved day by day and therefore they are good enough to be established as future new contents. In addition to the combination of Kinect and Oculus Lift proposed in this paper, utilization of other controllers will be possible to produce various virtual reality games and contents.

References

[1] Daegeun Kwon, Hyeongeun Shin, Yongje Kim, Jinyeong Jung, Seola Kim, Suncheon Jang, Daejun Lim, Yangwon Lim and Hankyu Lim, “Action simulation game design using the leap motion controller”, Proceeding of ICMEM 2014, Turkey, 2014, pp.381-391.

[2] Davis, Simon, Keith Nesbitt, and Eugene Nalivaiko. “Comparing the onset of cybersickness using the Oculus Rift and two virtual roller coasters.” Proceedings of the 11th Australasian Conference on Interactive Entertainment (IE 2015). Vol. 27. 2015, pp.30-33.

[3] Guo, B., Sun, J., Wei, Y. C., Meekhof, C., amd Leyvand, T. “Kinect identity: Technology and experience”, Computer, (4), 2011, pp. 94-96.

[4] Oculus Lift, https://en.wikipedia.org/wiki/Oculus_Rift

[5] Avateering C# Sample, https://msdn.microsoft.com/en-us/library/jj13104 1.aspx