

Story Creation And Design Algorithm In Unity

Woo-Hyun Park*, Ki-Nam Park**

*College of Information & Communication Engineering, Sungkyunkwan University, Korea

**College of Software, Sungkyunkwan University, Korea

*cgvt@skku.edu, **knpark2008@skku.edu

Abstract—In this study, we propose intelligent narrative story creation systems using artificial intelligence plan technology and the created story implemented using Unity game engine is expressed by an animation module in detail.

The program object implemented is separated into generic object and action objects. When one action is done, other actions are carried out by automatically recognizing the end event. When different virtual modules conflict, the action objects interact effectively with each other. We generate the story using the action object and surrounding objects. Based on the needs of user, we make it possible to implement actions through reference.

Keywords— Story Telling, VR(Virtual Reality), Game, Artificial Intelligent, Unity

I. INTRODUCTION

Recently, research is being carried out not only smart phones, but also on home appliances and consumer electronics. In particular, there is an active study of satisfying the user's five senses by providing indirect experience in virtual space. The satisfying factors include the fulfillment of the user's immersion with the game story, the fulfillment of the user's desire, and the rewards and responsibilities within the game. The immersion can be increased by experiencing the variety of stories and backgrounds in virtual reality implemented as narrative form using VR and Unity. This paper aims to design a story telling system to enhance satisfaction of user through VR.

II. RELATED STUDIES

A. Oculus Rift

Developers often use Oculus Rift and HTC VIVE, which are already on the market. The reason for using Oculus Rift is that it is possible to use two modes of standing and sitting type. In other words, it doesn't differentiate between two modes.

Oculus Rift fixes 90frame which is the primary frequency and can reduce overload by creating intermediate frames with a rendering speed of 45 fps. Types of controller according to contents include a keyboard type, a joystick type, and a touch type. Several VR products were released based on mobile and PC applications.

B. Cardboard

typical example is Google Card. It is impossible to connect to the PC and can be used for a mobile phone.

C. Unity

The Unity engine supports Oculus Rift and Mobile Application VR like Cardboard and Galaxy Gear VR currently.



Figure 1. A game implemented in Unity

It started support for the Will and iPhone in 2008.

In September 2010, it launched the NBA 3.0 version and supported Android, X-Box 360, and PlayStation3.

In February 2012, Flash was supported by 3.5 version.

Currently, it is a multiplatform engine that supports the Linux environment, and provides a basic interface with a common set of tools.

III. IMPLEMENTATION DETAILS

We developed this system in PC (Windows10 Home) platform using C# language, Oculus Rift(CV1) and Unity (version: 6.4.0). We implement two things significantly. The first is to suggest a trailer creation and the second is sequent UI system based on AI planner in virtual reality. Using these, we can add character's action scene. Figure 1 shows the processing flow of trailer production. In this paper, there are two major implementations. The first is to suggest an artificial intelligent generation

system and the second is to suggest minimizing movements of character and automatic movement.



Figure 2. Plan Structure Flow

We plan the example test with three actors, a 50-second trailer. First of all, if you look at the act of the count above figure 1, you will be able to walk within 0.4 seconds of the arrival point. The count is to give Hideko a ring in the box. The count at the box location takes action to pick up the ring in one second. After that, the count goes to another location for 0.8 seconds and then moves to meet Hideko to give her the ring. Each actions of actors follow the same procedure as the above. We generated Plane Material as the first step to implement the program.

We set up the coordinates of the place which characters (Man 1, Man 2, 1 Woman 1) walk around to (X:0, Y:15, Z:0) and we set up the scale value to (X:1, Y:1, Z:).

To set up the group of interact action object, we opened the Animation window to create a clip.

After selecting an object, we open Animator editor and created the transition from any state to each state. Transitions that are directed to each sequential action are selected and executed according to the condition for each action.

Users wearing VR can view the story in the third person according to each camera location. The Transform of Inspector view is used to control the positions of objects placed in the scene. The Camera applied to Perspective Projection to give the feeling of perspective. The viewing angle (Field of View) was set to 30, set Near to 0.03, and set Far to 300. Occlusion Culling was checked to avoid locations outside the observation area.

Also, to smooth the level of vision as camera is changed, it fixed RGB and change Alpha color to process the camera's fade.

Each character that made on 3D Max was used by importing in Asset store. Elsewhere, various objects such as tree, chair, building, grass, rock, etc. were placed around the plane to generate the story.

Before playing, the initial coordinate value of the male 1 character is set to (X:1.8, Y:0, Z:1.2), that of the male 2 Character is set to (X:2.7, Y:0, Z:-5.7), that of the female is set to (X:0, Y:0, Z:-1.5).

We set the directional light type to release the same amount of light in the plane. And the lighting position value is set to (X:10.5, Y:100, Z:-59.). The intensity value of lighting is set to 2.08.

When we start playing, in camera1, female character sits on the bench. The character of the male 1 comes to the female. And then he sings a song for proposal after rushing into a box to give the female a ring while the character of male 2 complain about at position (X:2.5, Y:0, Z:-2.6). The coordinate of the bench object is fixed to (X:0, Y:0, Z:-1.5), that of the Box object to (X:0, Y:0, Z:14). The action speed of the male 1 who ran to the box was set to 0.6.

The speed of action that after the man reaches the box and picks up the ring in the box is set to 1 second, the action of getting the ring set to 7 seconds at speed. The delta time represents the time gap between the previous frames and the current frame. The delta time was set to 1/60 to remove the difference about movement caused by the difference in the frame based on the device.

Each character is generated each animation controller and added various animation for each unique action. In the case of the male 1 character, he acted as a walking motions, picking motions, moving motions, moving hands, giving motion, telling motion, and proposing motion.

In the case of the male character2, he acted like walking, shouting, complaining, and kicking motion with foot.

In the case of the female character, the action was performed on the bench like sitting on the bench, the action of the walking, and the action of taking a ring.

Each action is allowed to control the speed and time of the action to arrive efficiently at the target point.

The animation is recorded on the animators according to the number of frames after selecting the character prefab composed of Bone and Rigid.

We exported the animation that worked in 3D Max and then imported into the Unity Program.



Figure 3. The male 1 who give the woman a ring.

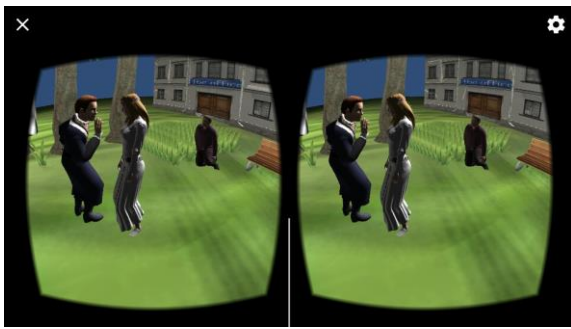


Figure 4. The scene of Cardboard VR

Four sets of camera objects were created as child objects and placed in order. We set the coordinates of the first camera to (X:0, Y:2, Z:17), the second camera to (X:-3.7, Y:112, Z:0), the third camera to (X:-1.3, Y:1.5, Z:-1), and the fourth camera to (X:-1.7, Y:2, Z:9.4). The position of the first camera with wide viewing angle allows all characters (male 1, male 2, female) to appear. The position of second camera is arranged at the side so that user can see the conflict scenes of characters. The third camera zooms in to show a scene effectively where a male 1 character gives a ring to a female character. The camera's locations are arranged to keep the characters from getting out of the cameras so that they can't be seen efficiently.

When the camera was rotated and a Light source reflected by the map was bright, symptoms of motion sickness appeared. Therefore, this paper has fixed the rotation value of the camera to zero and lowered the intensity of the light source to 2.08.

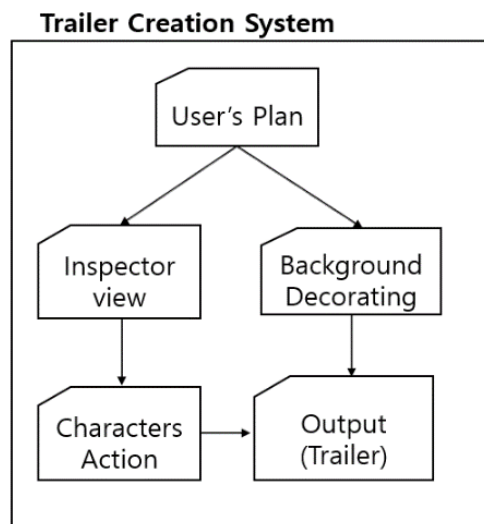


Figure 5. Trailer Creation Acting Model

Figure 2 shows the execution screen; when the user makes plan, it shows this in inspector view. And then its output is

trailer scene.

TABLE 1. IMPLEMENTATION OPTION MENU

Option name	Option Description
State Name	The user can set Action (give, talk, pick up...)
Goal	The location where the result of action is finished.
Speed	For actor's action, the user can set action's speed in detail.
Animation Length	The user can control animation length using this option.

The main option implemented in the system are described in Table 2. The user can set action name that actor should behave. The goal option is the location where the result of action is finished. In speed option, for actor's action, the user can set action's speed in detail. In animation length, the user can control animation length using this option.

IV. CONCLUSIONS



Figure 6. Trailer Start Scene

In this paper, the narrative generation and the movement optimization system were implemented using the Oculus Rift.

In order to implement the narrative generation system, it was possible to obtain the desired things by placing the character movements, generic objects, and four cameras.

Specifically, we give the actions speed and latency to play an animation. At the result of this, it is possible to interact between the character and the camera efficiently. Also, the sequence was organized organically between the generic object and the action object. In addition, in order to set the frame numbers consistent with the frame of the user's computer, we set up delta time to implement the same performance that were not disconnected in any VR environment. It was able to integrate with 3D Max on Unity Program, thereby generating a full-scale virtual reality world. We are confident that inexperienced individuals who lack the knowledge of Unity will be able to implement the system easily through this paper. However, it was difficult to identify the interaction between characters in a fixed camera location. Also, the texture of the map and the strength of the light source caused motion sickness.

In the future, we will automatically position the camera so that we can better see the interactions between the characters. In addition, when feeling fatigue or motion sickness, the user is recommended to relax by controlling Oculus, turn off the

image by user's choice and turn on the music for relaxation. we will also create an optimal movement algorithm for passing obstacles. Now, the characters move according to the defined algorithm, but in the future, the characters will move randomly, and the number of general objects will be increased so that the characters react according to the random situation.

ACKNOWLEDGMENT

This research was supported as a result of the research findings of the Center for Science and Technology Development Projects of the Center for Future and Technology Advancement Promotion (2015-0-00914)

REFERENCES

- [1] <http://terms.naver.com/entry.nhn?docId=841112&cid=42344&categoryId=42344>
- [2] http://navercast.naver.com/contents.nhn?rid=122&contents_id=108007
- [3] <http://www.inven.co.kr/webzine/news/?news=152580>.
- [4] APA styleIjaein, yuseunghan, Augmented reality assimilation based on the Web and smartphones, Information of the Information Education Society 16(2) (2012), 201-209.Home network technology, *IT/information technology magazine*.
- [5] <http://terms.naver.com/entry.nhn?docId=841112&cid=42344&categoryId=42344>
- [6] http://navercast.naver.com/contents.nhn?rid=122&contents_id=108007
- [7] <http://www.inven.co.kr/webzine/news/?news=152580>



Woo Hyun Park, was born in Republic of Korea, July 4, 1990. Woo Hyun Park is Master degree from Sungkyunkwan University in Korea. His main research interests include Artificial Intelligent Game, User Interface and Big data in virtual reality.



Ki Nam Park, was born in Republic of Korea, July 18, 1989. Ki Nam Park is Master degree from Sungkyunkwan University in Korea. His main research interests include Security of Artificial Intelligence.