

Design of 3D Oculus VR Action Game using Silhouette Outline

Ho-Ryel Kim*, Chang-Min Han*, Syoungog An*, Soo Kyun Kim**

*Student, Dept. of Game Engineering, PaiChai University, Daejeon, Korea

*Student, Dept. of Game Engineering, PaiChai University, Daejeon, Korea

*Professor, Dept. of Game Engineering, PaiChai University, Daejeon, Korea

**Professor, Dept. of Computer Engineering, Jeju National University, Jeju, Korea

[Abstract]

Lately the VR (Virtual Reality) game genre is becoming increasingly more popular and it has been cementing its place in the market as its own independent game genre. The key advantage of VR is that it lowers the barrier between player and the virtual world, thus creating an immersive experience. The suggested method develops a game that allows the player to experience what it is like to be visually impaired using the unique characteristics of VR. A distinctive feature of this game is that the character is provided only a limited range of sight, which is created using silhouette outlines. This restrictive visual field is then grafted onto VR and the player can indirectly experience blindness in a highly immersive manner. The silhouette outline along with the particle system is created using Oculus Rift, a headset highly used in VR game development, and Unity 3D game engine. We will also explain in detail regarding the removal of borders between the objects.

▶ **Key words:** Finite-State Machine, Shader, Oculus Rift, VR Action Game, Virtual Reality

[요 약]

최근 VR(Virtual Reality) 게임 장르가 점점 더 대중화되고, 시장에서는 자체적인 독립 게임 장르로 자리 매김하고 있습니다. VR의 주요 장점은 플레이어와 가상세계 간의 장벽을 낮추어 몰입형 경험을 제공한다는 것이다. 제안방법은 플레이어가 VR의 고유한 특성을 사용하여 시각적으로 몰입하고, 새로운 가상세계를 경험할 수 있는 게임을 개발하는 것을 목적으로 한다. 제안게임의 특징은 캐릭터가 제한된 범위의 시야만을 제공한다는 것이다. 이를 위해 윤곽선(Outline)을 이용하며, 이는 실루엣 윤곽선을 사용하여 만들게 된다. 이러한 제한적인 시야를 VR에 접목하고, 이를 통해 플레이어는 매우 몰입적인 방식으로 가상세계를 경험하게 된다. 또한 파티클 시스템과 함께 실루엣 윤곽을 만들기 위해 오쿨러스 리프트(Oculus Rift)와 유니티(Unity 3D) 게임 엔진을 기본으로 하여 개발한다. 또한 객체 사이의 경계선 제거에 대해 자세히 설명을 하며 게임 개발 과정에 대해 소개한다.

▶ **주제어:** 유한상태기계, 셰이더, 오쿨러스 리프트, VR 액션 게임, 가상현실

-
- First Author: Ho-Ryel Kim, Corresponding Author: Soo Kyun Kim
 - *Ho-Ryel Kim (ghfuf12@naver.com), Dept. of Game Engineering, PaiChai University
 - *Chang-Min Han (gksckdals234@naver.com), Dept. of Game Engineering, Pai Chai University
 - *Syoungog An (sungohk@pcu.ac.kr), Dept. of Game Engineering, PaiChai University
 - **Soo Kyun Kim (kimsk@jejunu.ac.kr), Dept. of Computer Engineering, Jeju National University
 - Received: 2020. 09. 29, Revised: 2020. 10. 30, Accepted: 2020. 11. 04.

I. Introduction

VR게임은 HMD(Head Mounted Display)[1]와 트래킹 센서를 통해 실시간으로 캐릭터를 조작하며, 정해진 목표를 향해 진행되는 게임이다. HMD와 트래킹 센서를 통해 조작이 가능하여 다른 플랫폼과는 다르게 세심한 조작이 필요한 요소가 다소 포함된다. VR은 사용자 자신이 주인공이 되어 그 행동을 실제로 진행하기 때문에 더욱 공감할 수 있고 상호작용을 할 수 있다. 요즘 PC게임 동향을 살펴보면 1인칭 액션 및 1인칭 슈팅 장르의 게임들을 VR에 접목시킨 게임들이 다수 존재한다. 이러한 시장에서 VR 액션 게임으로써 주목을 받기 위해서는 독창적, 창의적 아이디어 및 특수 효과가 있어야 한다.

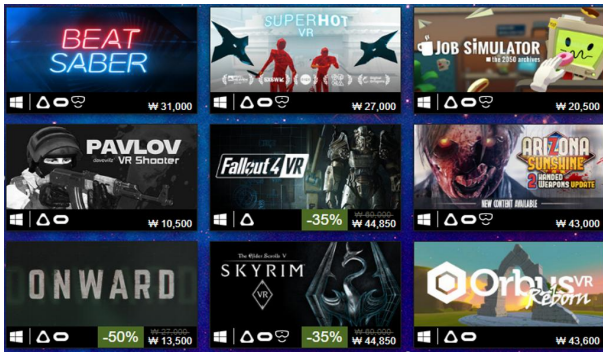


Fig. 1. 2018 Top Selling VR Games - Steam[2]

그림 1은 Steam[2]에서 2018년 높은 판매율을 보인 VR 게임들이다. 이러한 VR 게임들 중 6개 이상이 1인칭 장르의 액션 게임이며, 특수한 시각적 효과를 사용한 게임이 많다. 사용자의 움직임이 없으면 게임에서의 시간이 느리게 흐르든지 아니면, 근접무기로 노드를 자르며 리듬 게임을 하던지 화려한 파티클 이펙트 등의 다양한 시각적 효과를 사용한 것이 특징인 게임이 많다.

이와는 달리, 교육용 콘텐츠로서의 VR/AR/MR이 각광 받고 있다. 특히 AR[3]에 대한 개념과 응용까지의 예를 적절하게 잘 보여주고 있다. 혼합현실[4]에 기반을 둔 가상 화학 실험은 오클러스를 이용하여, 사용자 혼자서 간단한 실험을 할 수 있게 하였고, 이를 통해 실제 실험에서의 오류나 위험성을 통제할 수 있는 방법에 대해 제시하였다. 관련연구[5]는 라오스 학생을 위한 학습 미디어에 대한 다양한 콘텐츠 방법을 제시한 것이 특징이다.

본 제안 방법에서는 시각 장애인의 캐릭터 설정에 맞는 특수한 시각적 효과를 VR 액션 게임에서 개발하는 방법에 대해 다음 몇 가지 절을 통해 설명한다.

II. GAME DESIGN

1. Lighting

본 절은 게임에서 제일 중요한 조명에 대해 설명한다. 게임에서의 시각적 효과는 매우 높은 비중을 차지하며, 그 중에 조명은 일상생활에서 우리에게도 중요한 요소이며, 시각적인 효과 사용에 필수 불가결한 부분이다. 본 절에서는 사용자에게 특수한 시각적 효과를 제공하기 위해 조명(빛)의 제한적 사용에 관해 설명한다.

제안 방법은 플레이어 캐릭터를 제외한 모든 객체(Object)의 레이어(Layer)를 방향성 광원(Directional Light)의 컬링 마스크(Culling Mask) 기능에서 제외하여 플레이어 캐릭터 외에 다른 객체들의 명암을 제거하였다. 유니티[6,7,8,9]에서 제공하는 컬링 마스크의 기능은 카메라 혹은 빛에서 선택적으로 렌더러(Render)를 할 수 있는 기능이다. 그렇기 때문에 방향성 광원의 컬링 마스크에서 맵 객체들을 제외시켜 해당 객체들이 조명 렌더러에서 제외되도록 한다. 이러한 방법을 이용하면, 그림 2와 같이 객체 간의 경계를 없어지고, 사용자들의 객체들에 대한 구분이 어려워지게 된다.

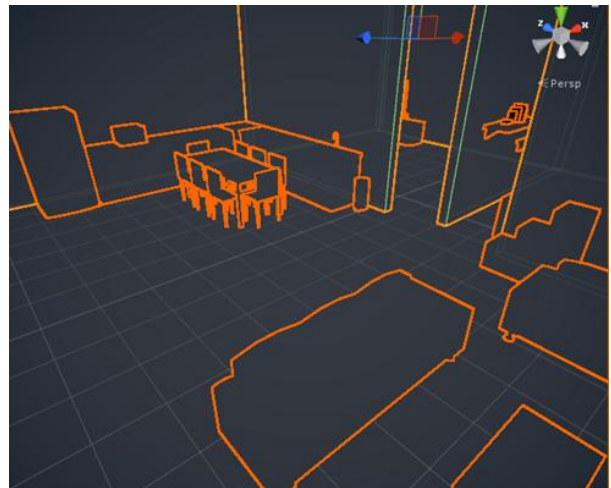


Fig. 2. Removing contrast of objects

2. Outline Shader

외곽선[10,11]을 통해 사용자에게 하여금 객체들에 대한 정보를 제공하게 된다. 렌더링(Rendering) 되는 모든 픽셀(Pixel)에 대해 법선 벡터(Normal)를 적용하여 확장시킨 후 객체의 기본 재질에 대해 렌더링 하게 된다.

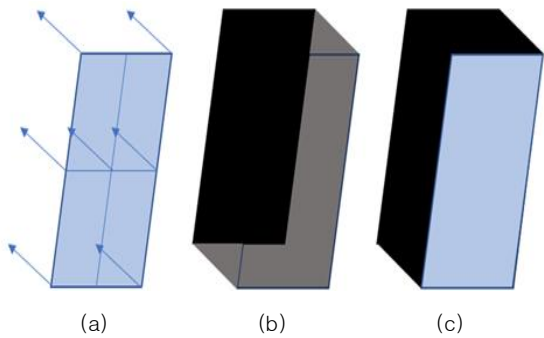


Fig. 3. (a) Normal vector, (b) Extend to normal vector and render, (c) Outline Render After Object Render

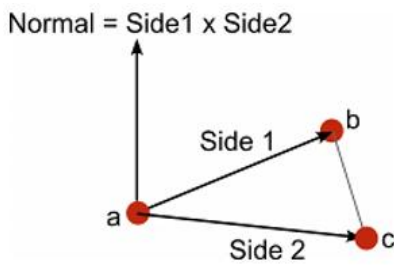


Fig. 4. Normal vector

그림 4와 같은 간단한 법선 벡터 공식을 이용하여 각 물체 픽셀에 적용시켜 그림 3과 같이 외곽선을 구현하여 사용자에게 제공한다. 이를 통해 단순히 빛만 뺏아 객체들의 경계를 구분하지 못하게 하고 정보를 감추기만 하는 것이 아닌 객체의 위치, 외형 등을 그림 5와 같이 제공하게 된다. 이러한 특수한 시각적 효과로 몰입도와 분위기를 높여 줄 수 있다.

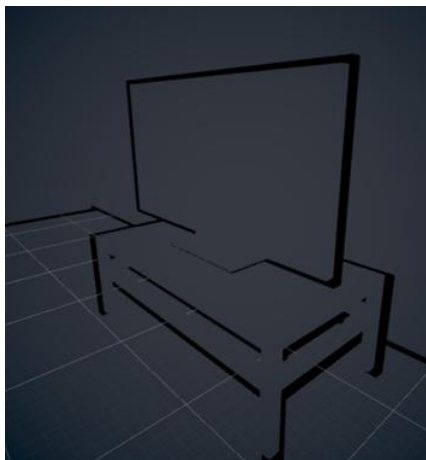


Fig. 5. Outline Shader

3. Object Shader

적 객체의 정보를 외곽선이 아닌 다른 방법을 사용하여 플레이어에게 알려주기 위하여 투시 기능[12]을 사용한다.

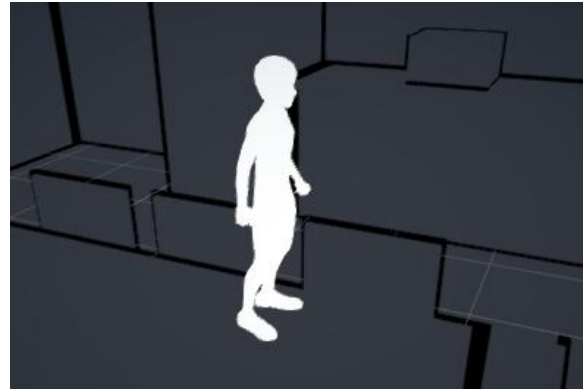


Fig. 6. See through the Enemy Object

쉐이더(Shader)의 깊이 테스트(Depth testing) 기능을 사용하여, 그림 7과 같이 깊이 테스트[12]으로 사용 중인 객체에 대한 정보를 사용자에게, 그림 6과 같이 맵 객체 너머로 표시해 준다. 그림 7은 깊이 테스트에 대한 간단한 설명으로, 어떻게 실행되는지를 보여준다. 기본 값은 LEqual로 객체로부터 보다 가까운 것은 그려주고 먼 곳은 그리지 않는 방식으로 객체 뒤에 객체가 존재할 경우 뒤에 객체의 가려지는 부분은 그려지지 않는 것이다.

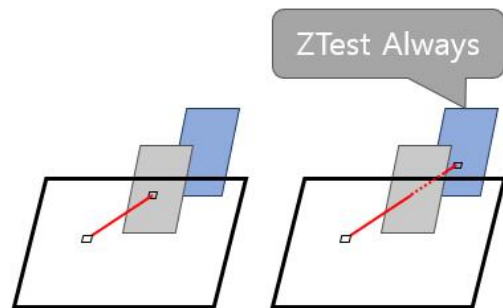


Fig. 7. Z-Testing[12]

4. Finite-State Machine

유한 상태 기계[13]는 유한한 개수의 상태(State)를 가지고 있는 기계라는 의미로, 기계는 한 개의 상태만 유지할 수 있고, 그 상태에 맞는 특별한 이벤트나 상황이 되었을 때 다른 상태로 전이하여 자신의 일을 수행하는 특징을 가진다. 유한 상태 기계를 사용하는 이유는 첫 번째로 가능한 상태들을 규정할 수 있다. 두 번째로 상태 중복을 피할 수 있고, 세 번째로는 전이들을 명확하게 규정할 수 있으며 그에 따라 기계의 동작을 분명하게 규정할 수 있다. 마지막으로 프로그래밍에서 유한 상태 기계에 기반을 둔 객체를 만든다면, 안정적인 작동을 보장할 수 있다. 이러한 유한 상태 기계의 특징으로 [그림 8]과 같이 캐릭터의 상태를 한 개의 상태만 유지하도록 구현한다.

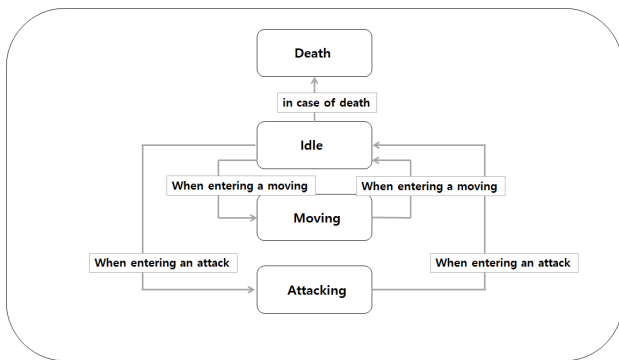


Fig. 8. Character FSM

5. Player Animation

캐릭터 애니메이션의 경우, 게임의 요소 중 매우 높은 비중을 차지하고 있다. 유니티 3D 엔진에서는 각각의 캐릭터 애니메이션을 애니메이션 컨트롤러를 이용하여 구현할 수 있다.

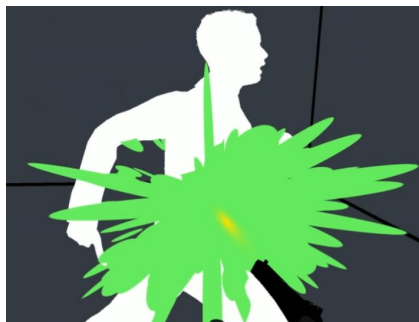


Fig. 9. Particle System

컨트롤러 내에서 각각의 애니메이션을 서로 이어준 후 캐릭터의 아바타에 적용하는 형식으로 구성된다. 애니메이션 컨트롤러를 사용하여 게임의 몰입감, 흥미를 높이는 것이 가능하며, 캐릭터의 공격 애니메이션 도중 적과 충돌(Collision) 할 때, 애니메이션 컨트롤러에서 애니메이션의 속도에 잠시 값을 0으로 주어 동작이 끊겨 보이게 할 수 있다. 또한 충돌 시 충돌 위치에서 파티클 시스템(Particle System)[14, 15] 등을 이용하여 그림 9와 같이 혈흔이 튀기는 듯 한 특수 효과를 연출할 수 있다. 이러한 특수 효과들은 사용자에게 하염금 타격감을 제공하며 분위기와 몰입도를 높여준다.

6. Monster Algorithm

몬스터의 AI(Artificial Intelligence) 부분은 사용자에게 직접적으로 체감이 되어 게임에서 가장 중요한 부분에 해당한다. 그림 10과 같은 방법으로 몬스터 알고리즘을 설계하였다. 첫 째로 알고리즘 시작 시, 플레이어 캐릭터가 주

변에 있는지 부터 검사를 한다. 주변에 존재하지 않을 경우 몬스터는 순찰 및 자신의 임무를 수행하며 대기 상태로 들어간다. 대기 상태 중 플레이어 캐릭터가 주변에 접근 시 플레이어를 인식하고 공격 가능 범위 내에 있는지 검사한다. 공격 가능 범위 외일 경우 플레이어에게 이동하게 되며 공격 가능 범위 내일 경우 플레이어를 공격하게 된다. 공격 종료 시 플레이어와 몬스터 중 사망한 캐릭터가 있는지 검사한 후 존재하지 않다면 해당 알고리즘을 반복하게 된다. 존재한다면 알고리즘은 프로세스를 종료하게 된다.

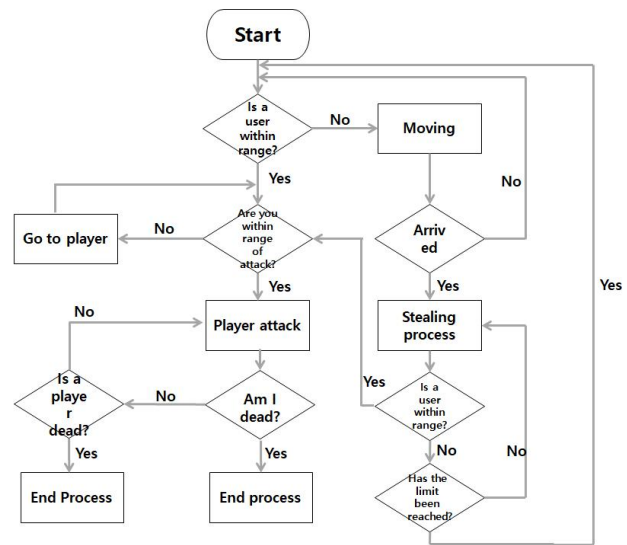


Fig. 10. Monster Algorithm

7. Item

VR에서 아이템을 잡기 위해선 오쿨러스[16]에서 지원하는 OVR Grabble 스크립트와 OVR Grabber를 통해 기능을 만들 수 있다. 기본적으로 컨트롤러에 Grabber 스크립트가 포함되어 있어야 하고 잡을 아이템에는 OVR Grabble 스크립트와 콜리더가 포함되어 있어야 한다. 그림 11과 같이 플레이어가 상호작용할 Grab콜리더와 몬스터와 상호작용할 어택(Attack) 콜리더를 만들어준다. Grab 콜리더는 오쿨러스에서 기능을 지원해주어 스크립트만 포함시켜주면 아이템을 잡을 수 있게 된다. 어택 콜리더 같은 경우 콜리더와 부딪힌 오브젝트가 몬스터일 경우 아이템이 가지고 있는 데미지 값을 몬스터에 넘겨 몬스터의 HP(Health Point)를 줄여주는 방식으로 근접 무기를 구현한다. 원거리 무기 같은 경우 OVR Grabber가 자기 잡은 오브젝트를 grabbed Object 함수에서 가지고 있어, 지금 현재 잡고 있는 아이템의 정보를 가지고 원거리 무기일 경우 지정한 버튼을 통해 상호작용을 하는 방식으로 구현한다. 총 같은 경우 트리거가 당겨지면 총구 방

향으로 레이 캐스트(Raycast)를 보내 몬스터가 맞으면 총의 데미지를 넘기는 방식으로 구현하고, 플레이어에게 어느 방향으로 총을 쏘는지 알 수 있도록 라인 렌더러(LineRenderer)를 이용해 총구 방향으로 부터 레이 캐스트가 도착한 지점까지 라인 렌더러를 그려 플레이어에게 직관적 사용자 인터페이스를 제공한다.



Fig. 11. Item

8. User Interface

VR 화면 속의 사용자 인터페이스(User Interface)는 PC처럼 마우스를 사용하여 UI를 직접 누를 수 없기 때문에 컨트롤러부터 멀리 떨어진 UI와 상호작용을 위해선 컨트롤러로부터 레이 캐스트[17]를 내보내 UI를 작동시켜야 한다. VR 화면 상에서 레이캐스트를 사용하기 위해선 오쿨러스에서 지원하는 프리팹(prefab)인 UIHelpers와 OVR 물리 레이캐스터(Physics Raycaster)를 사용하여 VR 상에서의 레이 캐스트를 구현할 수 있다. UIHelper안에는 레이(Ray)를 쏘아주고, 그 라인(Line)을 그려주는 레이저 포인터(Laser Pointer)와 레이와 컨트롤러의 상호작용을 설정할 수 있는 이벤트 시스템(EventSystem)으로 구성 되어있다.



Fig. 12. Raycast and MainUI

그림 12와 같이 OVR 물리 레이캐스터는 레이가 공간상에 존재하는 물체 간의 접촉이 있는지 확인해주는 역할을 하며 카메라에 추가하여 사용한다.

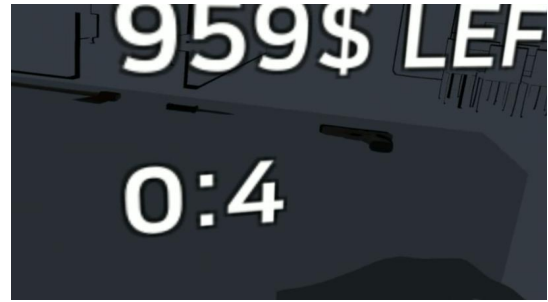


Fig. 13. Game UI(Score)

게임 안에서의 UI의 경우 캔버스(Canvas)의 렌더 모드(Render Mode)를 세계 좌표로 변경한 뒤 왼손 위치인 LeftHandAnchor에 추가해 주어 게임 진행에 필요한 UI를 그림13과 같이 왼손에 띄워주어 플레이어가 어디 있는지를 알 수 있도록 하였다.

9. Oculus VR

오쿨러스 VR은 적외선 센서를 이용하여 범위 내에 있는 사용자의 머리와 손의 회전이나 위치를 추적하여 해당 방향의 모습이 화면에 나타나거나 손의 움직임을 보여준다. 그리고 헤드 트래킹과 양쪽 눈의 오목하게 굽어진 디스플레이를 통해 사용자가 가상현실에 들어와 있다는 착각을 하게 만드는 역할을 한다.

유니티에서 VR환경을 구축하기 위해서는 오쿨러스에서 지원해주는 애셋(Asset)인 오쿨러스 Integration을 사용하여 유니티 내에서 트래킹을 위해 기본 카메라를 지우고 OVRCamera Rig를 사용해 플레이어가 머리를 움직이면 유니티 안에서 카메라가 움직이는 헤드 트래킹 기능을 설정하여 사용하게 된다. 플레이어의 자연스러운 상하 움직임을 위해서 OVRCameraRig에 있는 트래킹 원점 타임을 바닥 레벨(Floor Level)로 변경해 플레이어의 시점이 내려가면 캐릭터의 시점도 같이 내려가도록 만들었다. 바닥 레벨은 처음 사용자가 지정한 눈높이가 아닌 바닥으로부터 오쿨러스의 높이를 보정한 위치로 지정해 게임 속 시점의 높낮이를 조절해준다. 또한 캐릭터 이동은 OVRPlayerContorller를 통해 HMD의 위치와 카메라 위치를 일치할 수 있도록 지원해준다.

카메라가 가상의 벽이나 물체에 닿으면 카메라를 동적으로 조정해줘 카메라가 걸리거나 캐릭터의 위치로부터 멀어지는 현상을 줄여 쾌적한 VR 체험을 도와주게 된다.

VR 내의 손을 구현하기 위해선 플레이어의 구심점이 되는 OVR CameraRig의 자식 오브젝트로 지역 아바타(Local Avatar)라는 가상의 캐릭터를 만들어 게임 내에서 사용할 손을 만들어 컨트롤러와의 매핑을 통해 그림 14와 같이 자연스러운 손 움직임을 구현할 수 있다.



(a) Hand Tracking in Game



(b) Hand Tracking

Fig. 14. Hand-Tracking

III. Experimental Environment

본 연구에서 사용하는 하드웨어 및 소프트웨어는 다음과 같다. CPU는 AMD사의 라이젠 7세대 2,700을 사용하고, 그래픽 카드는 지포스(GeForce) GTX 1060 6GB를 사용하고, 메인 메모리는 16GB이다. 소프트웨어는 유니티 2018.3.10 버전과 오쿨러스 1.37.0.2451 88버전을 사용하고, C#: .Net 4.x을 사용하였다.



(a) Monster Death Animation



(b) Game Result

Fig. 15. Result Scene

IV. Conclusions

본 논문에서는 오쿨러스 리프트와 유니티 3D 엔진을 이용하여 VR 액션 게임을 개발하였고, 개발을 위한 조명, 셰이더 및 파티클 등의 기능 구현에 대해 자세히 살펴보았다. 특히 플레이어 캐릭터와 몬스터 사이의 상호작용에 필요한 구체적인 구현 방법에 대해 설명하였다. 게임을 위한 이러한 다양한 기능들을 이용하여 사용자들이 쉽게 몰입을 할 수 있게 하였고, 게임에 효과적으로 몰입할 수 있도록 하는 효과적인 기법으로 개발자들에게 있어 유용한 기능이 될 것이다. 또한 본 논문에서의 기능들은 높은 몰입도와 분위기를 자아내기 좋은 기능들로써 사용자에게 접근하고 있는 것이 특징이다.

향후 연구에서는 몰입감에 대한 평가 방법을 도입하고, 사용자들이 느끼는 멀미를 최소화 할 수 있는 방법에 대해 다각적 방법으로 접근할 예정이다.

ACKNOWLEDGEMENT

이 논문은 2020학년도 제주대학교 교육·연구 및 학생지도비 지원에 의하여 연구되었음.

REFERENCES

- [1] Jason Jerald, "The VR Book: Human-Centered Design for Virtual Reality", ACM Books, Morgan & Claypool Publishers, October 16, (2015)
- [2] Steam, <https://store.steampowered.com/>
- [3] Soo Kyun Kim et al., "Augmented-Reality Survey: from Concept to Application", KSII TRANSACTIONS ON INTERNET AND

INFORMATION SYSTEMS, Vol(11), Issue(2), pp. 98, doi: 10.3837/tiis.2017.02.019

- [4] Xiaoyun Duan, Shin-Jin Kang, Jong In Choi , Soo Kyun Kim, “Mixed Reality System for Virtual Chemistry Lab”, KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS, Vol. 14, No. 4, pp.1673-1688., Apr. 2020 SCI(E) doi: 10.3837/tiis.2020.04.014
- [5] Kalaphath Kounlaxay and Soo Kyun Kim, “Design of Learning Media in Mixed Reality for Lao Education”, Computers, Materials & Continua, Accepted SCI(E), doi:10.32604/cmc.2020.09930
- [6] Unity, <https://unity.com/kr>
- [7] JaehYeon Lee, “Absolute Class! Unity5”, Wiki Books 2015.
- [8] Patrick Felicia, “Unity 5 From Zero to Proficiency (Foundations): A step-by-step guide to creating your first game with Unity”, Amazon Kindle, October 2015.
- [9] Matt Smith, “Unity 5.x Cookbook”, Packt Publishing; 1 edition ,October 2015.
- [10] Gitlab, <http://www.gitlab.com/>
- [11] Outline Shader, <https://roystan.net/articles/outline-shader.html>
- [12] Patrick Felicia, Unity From Zero to Proficiency (Beginner): A step-by-step guide to coding your first game with Unity in C#, Patrick Felicia; 3rd Edition (December 19, 2015)
- [13] Jason Wang, Understanding of Finite State Machine, February 4, 2017
- [14] Alan Thorn, “Unity Animation Essentials”, Packt Publishing, June 2015
- [15] Particle System, <https://docs.unity3d.com/Manual/ParticleSystems.html>
- [16] Oculus Rift, <https://www.oculus.com/>

Authors



Ho-Ryel Kim is attending in Game Engineering in Paichai University, Korea



Chang-Min Han is attending in Game Engineering in Paichai University, Korea.



Syoungog An received the Ph.D. degrees in Computer Science and Engineering from Korea University, Korea, in 1989. She is currently a Professor in the Department of Game Engineering, Paichai University.



Soo Kyun Kim received Ph.D. in Computer Science & Engineering Department of Korea University, Seoul, Korea, in 2006. He joined the Telecommunication R&D center at Samsung Electronics Co., Ltd., from 2006

and 2008. and He worked as a professor at the Department of Game Engineering at Paichai University until 2020. He is now a professor at the Department of Computer Engineering at Jeju National University, Korea. His research interests include multimedia, pattern recognition, image processing, mobile graphics, geometric modeling, and interactive computer graphics. He is a member of ACM, IEEE, IEEE CS, KACE, KMMS, KKITS, and KIIT.