

스마트폰을 이용한 저가 VR 시스템 구현에 관한 연구

임은수·윤성의·고영석·정하영·최홍섭*
대전대학교 전자공학과

Implementation of a Low-cost Virtual Reality System Using Smart Phone

Eun-Su Lim · Sung-Yi Yun · Yong-Suk Ko · Ha-Young Jung · Hong-Sub Choi*

Department of Electronic Engineering, Daejin University, 1007 Hoguk-Ro, Pocheon-Si, Gyonggi-Do 11159, Korea

[요 약]

현재 가상현실 상용화의 문제점은 사람들이 가상현실을 체험하기 위해서는 수십만 원을 호가하는 가상현실 장비들이 필요하다는 것과 가상현실을 즐길 만한 콘텐츠가 부족하다는 것이다. 이러한 문제를 해결하기 위하여 스마트폰과 PC를 활용하여 별도의 비용 없이 HMD(Head Mounted Display)의 기능을 구현하고 무선으로 가상현실 시스템을 구축하는 것이 본 논문의 목표이다. 가상현실은 많은 양의 데이터 처리를 요구하기 때문에 PC에서 데이터를 처리하고, 스마트폰의 디스플레이와 내장 센서를 이용하여 PC와의 데이터 송수신을 통해 HMD의 기능을 구현하였다. 또한 무선 컨트롤러를 제작하여 사용자의 움직임에 자유롭게 하였다. 본 논문에서 구현한 시스템의 테스트를 위해 콘텐츠를 제작하여 실험하였다. 이를 통하여 사용자는 고가의 장비 없이 기존에 보유하고 있던 스마트폰과 PC를 이용하여 가상현실을 체험할 수 있으며, 기존의 유선환경에서 벗어나 무선 환경에서 가상현실을 체험할 수 있다. 이렇게 가상현실을 쉽게 접함으로써 사용자가 증가하면 개발에 많은 투자가 됨에 따라 가상현실 기술 발전과 상용화에 많은 진전이 있을 것으로 기대된다.

[Abstract]

Currently the bottleneck in virtual reality's commercialization might be that high cost VR equipment is needed and interesting VR contents are not enough. The purpose of this paper is to solve these problems by implementing low-cost VR system with wireless controller and HMD(Head Mounted Display) using a smart phone and PC. The functions of real HMD are simulated by utilizing the display and embedded sensors of a smart phone. In that situation PC is in charge of processing huge data and is communicated with smart phone. And wireless controller is designed to make VR user's movement to be free. In addition, we made the several VR contents for testing our prototype system directly. Easy access to VR equipment could induce increasing number of users and investment. And it will be a big step toward improving VR technology and its commercialization.

색인어 : 가상현실, 스마트폰, HMD

Key word : VR(Virtual Reality), Smart phone, HMD(Head Mounted Display)

<http://dx.doi.org/10.9728/dcs.2018.19.7.1237>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 28 June 2018; **Revised** 18 July 2018

Accepted 25 July 2018

***Corresponding Author; Hong - Sub Choi**

Tel: [REDACTED]

E-mail: hschoi@daejin.ac.kr

1. 서론

우리는 지금까지의 삶의 방식을 근본적으로 바꿀 기술 혁명의 흐름 속에 있다. 정보통신기술(ICT)의 융합으로 이루어낸 제 4차 산업혁명은 빅 데이터, 인공지능, 로봇, 사물인터넷, 가상현실과 같은 새로운 기술이 혁신의 중심에 있다.

이중 가상현실은 현재 구글, 오픈GL, 페이스북, HTC, 소니 등 IT 기업의 선두주자들을 비롯하여 중소기업을 포함한 많은 기업들이 가상현실 관련 디바이스를 출시하면서 가상현실 시장이 본격적으로 확산되고 있다.

그림 1의 Statista에서 발표한 가상현실 시장 현황에 따르면 전 세계 가상현실 시장 규모는 2014년 90백만 달러에서 2016년에는 3,800백만 달러로 성장하였으며 2018년에는 5,200백만 달러까지 성장할 것으로 예상하고 있다.

그러나 현재 가상현실은 체험 할 수 있는 기기가 매우 비싸고, 사람들이 다양하게 즐길 수 있는 콘텐츠가 아직까지는 많이 부족하다는 문제가 있다. 대부분의 HMD(Head mounted Display) 장비들이 고가로 대중들의 경제적 부담감을 증가시켜 가상현실의 접근성이 감소하여 가상현실 시장의 성장이 느려지는 원인이 될 수 있다. 또한 장비들이 유선으로 되어있기 때문에 움직임에 제한을 받는다[1].

본 논문은 이러한 문제에 대해 일반인들이 쉽게 가상현실을 즐길 수 있도록 경제적 부담감을 감소시키고 접근성을 높이기 위해 기존에 보급된 PC와 스마트폰을 이용하여 고가의 HMD 없이 가상현실 환경을 구현하는 것을 목적으로 한다.

기능을 구현하려고 한다.

또한 그에 상응하는 콘텐츠를 개발하여 대중들이 가상현실에 쉽게 접근하고 재미있게 즐길 수 있는 가상현실 환경을 구현하기 위하여 다음과 같은 세부 목표를 구현하였다.

2.1.1 스마트폰과 PC를 이용하여 디스플레이 영상 제공

가상현실을 구현하기 위해서는 많은 연산량이 요구되기 때문에 이러한 작업은 PC에서 처리해서 화면을 스마트폰에 디스플레이하여 사용하였다.

2.1.2 무선 컨트롤러 제작과 헤드트래킹

조이스틱 중심으로 이루어진 무선 컨트롤러를 제작하여 현재 가상현실에서의 유선의 제약을 벗어나 무선 환경을 구축하였다. 그리고 안드로이드 스튜디오로 스마트폰 어플리케이션을 제작하여 PC에서 스마트폰의 센서값을 받아 헤드트래킹을 구현하였다.

2.1.3 멀미현상을 최소화하기 위한 이동방식

기존 이동방식인 W(상), S(하), A(좌), D(우)에서의 문제점인 멀미현상을 최소화하기 위한 방법으로 텔레포트와 빠른 이동 방식을 구현하였다. 또한 이동 중에 화면의 초점을 순간 흐리게 하여 순간적인 화면 전환에 대한 멀미현상을 최소화하였다.

2.1.4 콘텐츠 구현

구현한 시제품에 적용하기 위해 언리얼 엔진4를 이용하여 박물관 체험, 방 탈출, 틀린 방 찾기 콘텐츠 등을 제작하였다.

2.2 시스템 구현 방식

2.2.1 전체 시스템

전체 시스템은 스마트폰, PC, 컨트롤러 총 세 개의 구성요소로 이루어지며, 스마트폰은 내장되어 있는 가속도, 자이로 센서의 값을 이용한 HMD 기능의 구현과 액정을 이용한 화면 디스플레이 기능을 수행한다. PC는 가상현실 구현을 위한 데이터를 처리하며 처리된 영상 결과를 스마트폰으로 전송한다. 컨트롤러는 사용자의 조작을 조이스틱과 자이로, 가속도 센서를 이용해 감지하여 PC로 데이터를 전송한다.

전체적인 시스템의 블록도는 그림 2와 같다. 스마트폰에서 PC로 블루투스를 이용하여 자이로 센서, 가속도 센서의 값을 송신하면 PC에서는 이를 수신하여 스마트폰의 움직임과 가상현실 내에서의 움직임을 일치시킨다. 컨트롤러에서 조이스틱을 조작하게 되면 블루투스를 통해 HID(Human Interface Device) 형식의 통신으로 PC에서 키보드, 마우스의 동작으로 인식하게 된다. 이를 통해 컨트롤러로 가상현실내의 캐릭터의 움직임을 조종할 수 있게 하였다. 최종적으로 PC에서 구현된 가상현실의 화면은 스마트폰으로 전송되며 사용자는 이를 통해 가상현실을 체험하게 된다.

Global Virtual Reality Market Size



Statista(2016b)

그림 1. 가상현실 시장 현황
Fig. 1. Virtual reality market status

2. 본론

2.1 연구의 세부목표

본 논문은 기존의 보급된 PC와 스마트폰을 이용하여 고가의 가상현실 기기 없이도 쉽게 즐길 수 있도록 스마트폰의 디스플레이와 센서를 이용하여 PC와의 무선통신을 통해 HMD의

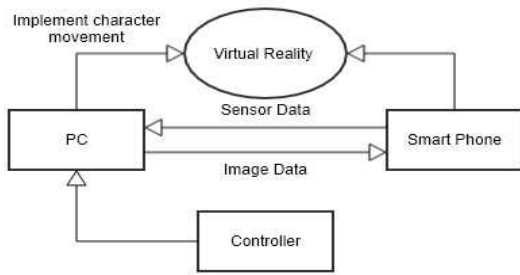


그림 2. 전체 시스템 블록도
Fig. 2. Block diagram of overall system

2.2.2 스마트폰 어플리케이션

어플리케이션은 스마트폰의 내장 센서들을 이용하여 HMD의 Head Tracking 기능을 구현하는 것을 목표로 두었다.

스마트폰 어플리케이션은 스마트폰의 내장 센서인 자이로, 가속도 센서를 이용하여 해당 값을 블루투스를 통하여 PC로 전송한다. 이 값을 통하여 PC에서는 가상현실 내에서 캐릭터가 보는 시점을 움직이며, 사용자는 HMD를 착용하는 것과 같은 효과를 얻을 수 있다.

스마트폰 어플리케이션은 Android Studio를 이용하여 제작하였다[2]. 어플리케이션은 센서 매니저를 이용하여 안드로이드의 자이로, 가속도 센서 값을 읽어오고 Thread를 이용하여 일정시간마다 이 값을 블루투스를 통하여 PC의 가상 시리얼 포트(Virtual Serial Port)로 전송되도록 하였다. 또한 어플리케이션의 동작 중지 시점을 onDestroy()로 설정하여 onStop()이나 onPause()시에도 동작이 중지되지 않도록 하여 백그라운드 상태에서도 지속해서 데이터를 전송할 수 있도록 하였다. 그리고 사용자가 어플리케이션이 동작 중이라는 것을 인식하도록 하기 위하여 백그라운드 상태가 될 경우 알림바를 통하여 어플리케이션이 실행 중이라는 것을 알리도록 하였다. 센서 어플리케이션에서 사용하는 클래스는 총 3가지이다.

BluetoothChatService 클래스는 MainActivity 클래스의 onCreate() 메소드에서 실행되며 블루투스를 통하여 데이터를 전송하는 과정을 담당한다.

DeviceListActivity 클래스는 MainActivity 클래스에서 화면 레이아웃의 ‘연결’버튼을 눌렀을 때 실행되며 주변 블루투스 장치의 목록을 가져오고 선택된 블루투스 장치와의 연결을 담당한다.

MainActivity 클래스는 프로그램이 실행되는 클래스로 액티비티의 라이프 사이클에서의 동작을 담당한다. 레이아웃은 어플리케이션의 메인화면을 담당하는 activity_main 레이아웃과 블루투스 연결 시 블루투스 장치의 목록을 표시하는 device_list 레이아웃 두 가지를 사용하였다. 메인화면 레이아웃은 ‘검색키’, ‘연결’, ‘최소화’, ‘끄기’ 4개의 버튼을 가지며 현재 스마트폰의 자이로, 가속도 센서의 값을 화면에 표시하도록 하였다. 어플리케이션은 그림 3의 순서도와 같이 동작한다.

‘검색키’ 버튼을 누르면 스마트폰의 블루투스 검색기능이 켜짐으로 바뀌어 다른 디바이스에서 스마트폰을 블루투스 장치로 검색할 수 있도록 한다. ‘연결’ 버튼을 누르면 device_list 레이아웃이 화면에 표시되며 연결 가능한 블루투스 장치들이 표시된다. 목록에서 블루투스 장치를 선택하면 해당 장치로 연결 대기를 한다. ‘최소화’ 버튼을 누르면 어플리케이션이 백그라운드로 이동하며 메인화면으로 이동한다. 그리고 알림창에 ‘VR 어플리케이션이 실행중입니다.’ 라는 알림이 표시된다. ‘끄기’ 버튼을 누르면 블루투스 연결이 해제되며, 센서매니저의 동작을 중지한다.

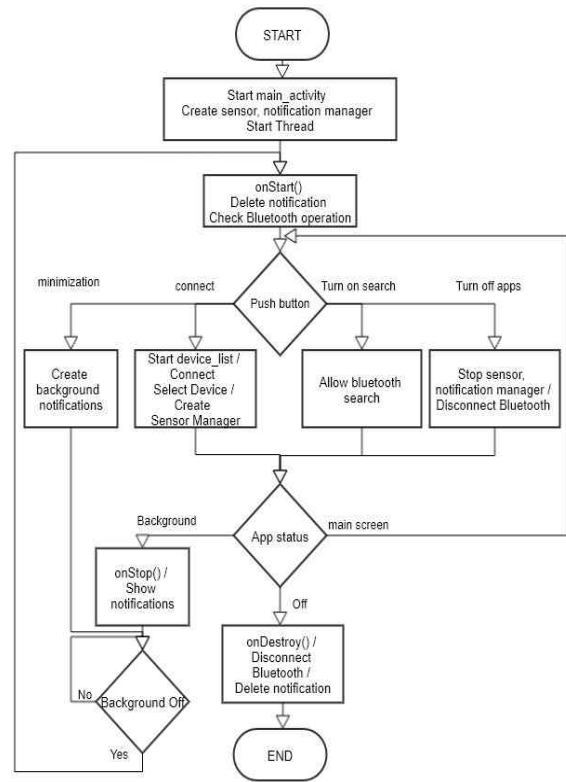


그림 3. 스마트폰 어플리케이션의 전체 순서도
Fig. 3. Flowchart of smart phone application

2.2.3 컨트롤러

컨트롤러는 HID 블루투스 모듈을 이용하여 연결 장치가 컨트롤러를 HID로 인식하게 하여 조작에 따라 키보드, 마우스의 동작을 수행하도록 하였다. 조이스틱의 왼손과 오른손의 움직임에 따라 게임에서 주로 사용되는 상하좌우, 누름 키가 눌리도록 하였으며, 조이스틱이 가운데 위치하면 키를 뺐 것으로 하였다. 또한 오른쪽 컨트롤러에 자이로, 가속도 센서를 이용하여 컨트롤러의 공간상 움직임이 마우스를 제어하도록 하여, 게임 내에서 손이 움직이는 것과 같은 효과를 주었다[3].

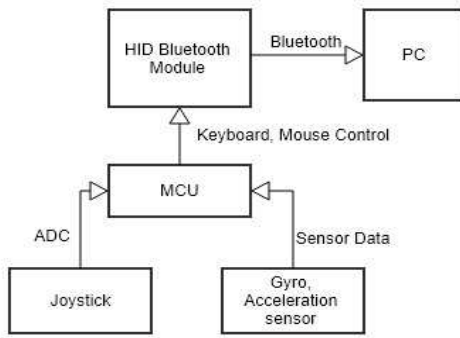


그림 4. 컨트롤러 전체 블록도
 Fig. 4. Block diagram of controller

컨트롤러의 전체 블록도는 그림 4와 같다. MCU(Micro Controller Unit)와 자이로, 가속도 센서, 조이스틱 2개, HID 블루투스 모듈을 사용하였다. MCU는 조이스틱과 자이로, 가속도 센서의 값을 읽어오고, 값의 변화에 따라 키보드, 마우스의 제어 값을 선택하여 HID 블루투스 모듈로 전송하게 된다. HID 블루투스 모듈은 수신된 값을 PC로 전송하며, PC는 수신된 값을 바탕으로 키보드, 마우스의 동작을 제어한다.

MCU로는 ARM-Cortex-m3 기반의 STM32F103C를 사용하였다. 전원부에 별도의 스위치를 넣어 전원을 쉽게 켜고 끌 수 있도록 하였고, LED를 연결하여 전원 상태를 확인할 수 있도록 하였다. 컨트롤러의 전체 순서도는 그림 5와 같다.

MPU-6050은 6축 자이로, 가속도센서로 각각 X, Y, Z축에 대한 자이로, 가속도 값을 가져온다. SCL, SDA인 PB6, 7번 핀을 이용하여 센서 모듈과 I2C통신을 하여 센서 값을 가져온다. 조이스틱의 오른쪽 컨트롤러에는 자이로, 가속도 센서가 내장되어 있으며, 가져온 센서 값 중 자이로 센서 값을 이용하여 그 값의 변화량 만큼을 마우스 좌표에 더해주어 컨트롤러의 공간상 움직임이 마우스를 제어할 수 있도록 하였다.

조이스틱은 3개의 출력 값을 가지며 X축, Y축, 버튼 총 세 개의 출력을 내보낸다. A0~A5까지 왼손, 오른손 총 6개의 ADC핀에 연결된다. 조이스틱은 X축, Y축 상하 움직임에 따라 전압 값이 달라진다. 이 값이 일정 값 이상이 되면 상, 일정 값 이하가 되면 하 방향으로 움직였다고 판별하며, 이에 따라 상하좌우로 움직인다. 조이스틱을 누르면 버튼이 눌리게 되며 버튼은 버튼을 누를 때 접지와 연결되어 출력 값이 0이 된다. 조이스틱의 동작은 상, 하, 좌, 우, 버튼 총 5개이며, 양손을 합하여 10개의 동작으로 분류할 수 있다. 각각의 동작에 따라 왼손 ‘W(상)’ ‘S(하)’ ‘A(좌)’ ‘D(우)’ ‘E(버튼)’와 오른손 ‘I(상)’ ‘K(하)’ ‘J(좌)’ ‘L(우)’ ‘O(버튼)’ 의 키보드를 누른 것과 같은 동작을 한다.

사용자가 조이스틱의 조작을 중지하여 스틱이 중앙에 위치하게 되면 전압 값이 기준 값 이하가 되며, 이 경우 블루투스 모듈은 키 누름을 해제하는 신호를 MCU로 보낸다.

MCU는 사용자가 조이스틱을 제어할 때 그 값을 읽어 조이

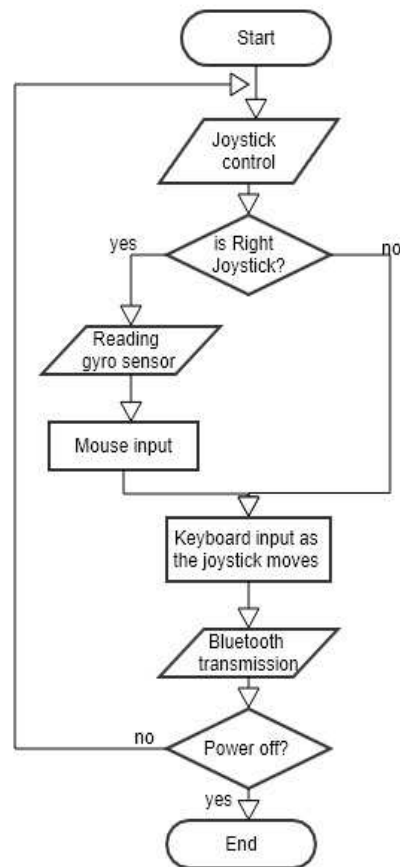


그림 5. 컨트롤러 전체 순서도
 Fig. 5. Flowchart of controller

스틱 제어에 따른 지정된 값을 HID 인터페이스 코드로 변환하여 블루투스 모듈로 전송하고, HID 인터페이스 코드를 수신 받은 블루투스 모듈은 이를 PC로 전송하며, PC는 전송받은 데이터에 따라 키보드, 마우스 동작을 제어한다. 블루투스 모듈은 HID 프로파일을 내장하고 있어 외부 장치와 HID 무선 장치로 연결되며 별도의 드라이버 설치 없이 키보드, 마우스 등 인터페이스 제어가 가능하다.

2.2.4 Head Tracking

스마트폰의 자이로 센서와 가속도 센서 값을 블루투스를 이용한 시리얼 통신을 통해 값을 받아 이 값을 처리하여 가상현실 내의 사용자의 눈이 될 카메라의 각도를 사용자의 머리의 각도와 일치시켜 Head Tracking을 구현한다.

전체적인 흐름은 그림 6과 같다. 먼저 VRmode라는 변수와 각 X, Y, Z라는 변수를 선언한다. VRmode는 Head Tracking의 사용 여부를 결정한다. 앞뒤를 X축, 좌우를 Y축, 상하를 Z축으로 정의할 때, 각각 3차원 상의 X축의 회전각도(Roll), Y축의 회

전각도(Pitch), Z축의 회전각도(Yaw)의 값을 저장한다.

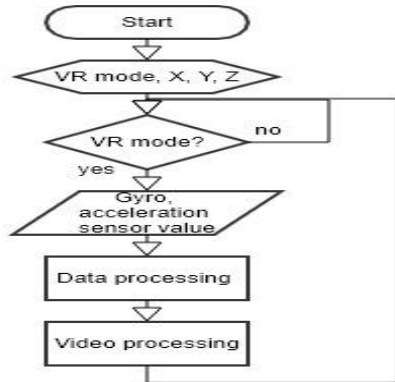


그림 6. Head Tracking 전체 순서도
Fig. 6. Flowchart of Head Tracking

Head Tracking이 시작되면 반복문을 이용하여 0.01초 단위로 Head Tracking을 반복한다. 반복문의 지연시간이 0.01초로 적용한 이유는 가상현실 내에서 Head Tracking은 20ms이하의 지연을 가져야 어지러움을 최소화 할 수 있다고 한다. 그렇기 때문에 스마트폰에서 센서 값을 받아올 때의 지연과 나중에서 있을 영상 스트리밍의 지연시간을 고려하여 지연시간을 0.01초 즉 10ms로 적용하였다. VRmode이고 스마트폰과 시리얼 통신이 연결이 되었다면 스마트폰의 자이로 센서와 가속도 센서 값을 읽는다. 센서 값은 바로 사용하는 것이 아니라 상보필터를 사용하여 값을 처리 후 사용하게 된다.

자이로 센서는 각속도를 측정하는 센서로 특정 축을 기준으로 회전하는 속도를 나타낸다. 그러므로 자이로 센서 값을 적분하여 각도를 얻을 수 있다. 즉 초기 값에서 매 기준 시간마다 자이로 센서 값을 적분하여 더하면 현재의 각도를 계산할 수 있다. 그러나 센서의 오차와 적분 시 소수점 아래 몇 자리 이하는 버리면서 계산을 하기 때문에 측정된 각도가 틀어지게 되는 드리프트 현상이 발생하게 된다. 그리고 가속도 센서는 중력 가속도를 이용하여 각 축이 중력을 어느 정도 받는지에 따라 각도를 측정한다. 또한 가속도 센서는 이동과 진동에 취약하여 노이즈가 심한 단점이 있다. 이 센서 값들을 보충하여 더 정확한 각도를 계산하는 것이 바로 상보필터이다.

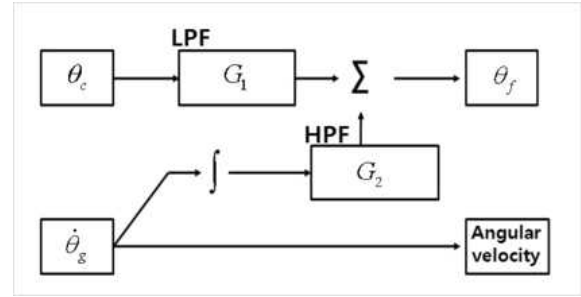


그림 7. 상보필터
Fig. 7. Complementary filter

그림 7은 상보필터의 기본개념이며, 저역필터와 고역필터 부분은 아래의 수식에서처럼 상수로 하여 적용하였다.

$$\angle = 0.98 * (\angle + gyrr Data * dt) + 0.02 * (accData) \quad (1)$$

식(1)은 가장 일반적으로 사용되는 상보필터 수식과 계수 값이며, X, Y, Z는 위 수식에서 angle이며, K가 저역필터와 고역필터를 담당하는 계수가 되며 그 범위는 저역필터와 고역필터의 합이 1이 되기 위해 0에서 1사이의 값을 갖는다. 다시 정리해보면 자이로 센서 값은 고주파 영역의 값을 받아오고 가속도 센서 값은 저주파 영역의 값을 받아오는 것이다. 상보필터는 아래와 같이 사용하였다. (X, Y, Z의 초기 값은 0)

- X축의 각도(Roll) :
 $X = K * (X + gyrr Data * (-1.1)) + (1 - K) * (accData) \quad (2)$
- Y축의 각도(Pitch) :
 $Y = K * (Y + gyrr Data * (-1.1)) + (1 - K) * (accData) \quad (3)$
- Z축의 각도(Yaw) :
 $Z = Z + (-1) * gyrr Data \quad (4)$

Roll은 사용자 고개의 기울기, Pitch는 고개의 상하방향, Yaw는 좌우방향이다. K값은 계수이며 gyrrData는 자이로 센서 값, accData는 가속도 센서 값이다. 좌우방향인 Yaw의 경우 가속도 센서의 값이 의미가 없으므로 자이로 센서만 사용하였다. K값은 사용자의 환경에 따라 조절이 가능하도록 추가하였다. 특히 무선 환경에 따라 달라 질 수 있는데, 무선 환경인 와이파이와 블루투스 환경이 좋을 경우엔 더 높은 해상도와 더 낮은 지연을 가지기 때문에 상보필터에서 자이로 센서의 적분 연산의 신뢰도가 높은 반면 무선 환경이 좋지 못할 경우엔 더 높은 지연을 가지기 때문에 반대로 자이로 센서의 신뢰도가 더 낮아지게 된다. 그러므로 전자의 경우 K값을 높여서 자이로 센서에 중점을 두며, 후자의 경우 K값을 낮춰 가속도 센서에 중점을 두어 사용자의 무선 환경에 맞춰 조절한다.

2.2.5 Movement

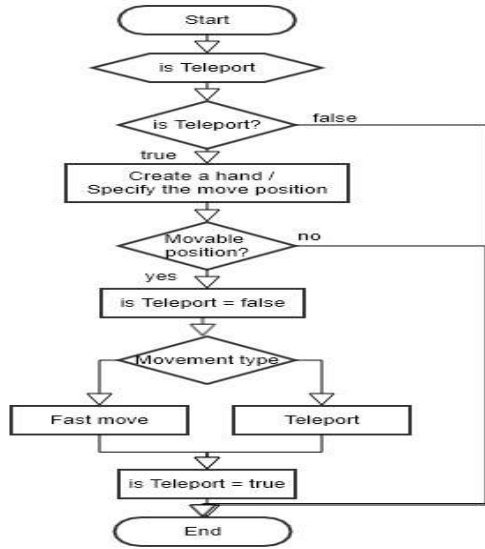


그림 8. 움직임의 전체 순서도
 Fig. 8. Flowchart of Movement

그림 8은 가상현실 내에서 움직임을 제어하는 부분의 순서도이다. 본 콘텐츠에서의 이동방식에는 3가지가 있다. 첫 번째로는 기본적인 이동방식인 걷기가 있다. 3D 멀미 현상을 겪을 수 있기 때문에 이를 방지하기 위해 두 번째 방식은 순간이동이다. 그리고 이동시에 3D 멀미 현상을 낮추기 위해 시선을 분산시키기 위한 Blur처리를 하였다.

이 방식의 문제는 갑자기 순간이동을 하기 때문에 사용자가 방향감각을 상실하는 문제가 발생하기 때문에 이를 해결하기 위한 세 번째 방식은 빠른 이동이다. 이동할 위치로 1초 내로 빠르게 이동하여 이동하는 방향을 인식하면서 빠르게 이동할 수 있으며 마찬가지로 이동시 Blur처리를 하였다. 하지만 이 방식의 경우는 두 번째 방식보다 3D 멀미 현상을 좀 더 유발 시킬 수 있기 때문에 사용자가 선택하여 이동할 수 있도록 3가지의 이동방식을 제공한다. 첫 번째 방식은 기본적인 이동방식이기에 생략하고 두 번째와 세 번째의 이동방식을 설명하도록 하겠다.

먼저 Is Teleport가 True인 경우에만 순간 이동 및 빠른 이동이 가능하며, True일 경우 이동할 위치를 지정하기 위한 가상의 손(오른손)을 생성한다. 일반적으로 인식하는 손의 위치처럼 배치하기 위하여 사용자의 시선에서 약간 앞, 그리고 약간 오른쪽에 위치하도록 좌표를 계산하여 가상의 손을 생성한다.

그림 9에서 빨간 둥근 점은 캐릭터의 손에서 시작된 벡터가 벽과 만나는 지점이며 이 위치가 이동할 위치가 된다. 생성된 벡터의 크기보다 멀거나 이동이 불가능한 지점이라면 빨간 점이 생성되지 않아 시각적으로 이동 가능 여부를 확인 할 수 있다. 그리고 이제 이동할 위치가 지정 되면 Is Teleport 값은 False가 되는데, 이는 한 번에 너무 많은 순간 이동이나 빠른 이동을



그림 9. Teleport 이동 화면
 Fig. 9. Display of Teleport movement

할 경우에 오히려 3D 멀미를 더 유발 할 수 있기 때문에 이를 방지하기 위함이다.

다음으로 그림 10과 같이 화면을 Blur 처리를 하여 사용자가 순간 이동 및 빠른 이동 시에 3D 멀미를 줄이기 위해 시선을 분산 처리한다. 그리고 이 Blur가 지속되는 동안 현재 설정된 이동 방식이 순간 이동인지 빠른 이동인지를 확인하여 해당하는 방식으로 이동을 하게 된다. 이동이 완료 되면 다음 이동이 가능하도록 Is Teleport 값을 True로 설정한다.

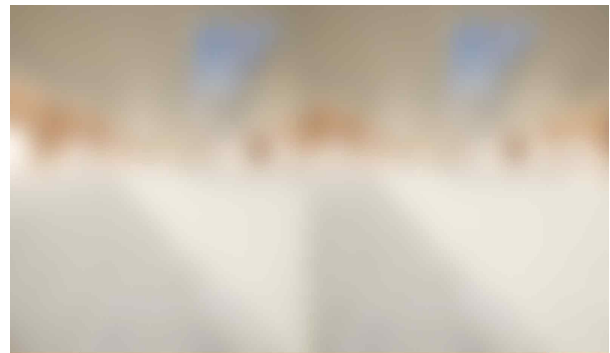


그림 10. Blur 처리 후 이동
 Fig. 10. Movement with blurring effect

3. 시스템 구현 결과

그림 11은 PC의 화면이 스마트폰으로 전송되는 모습이다. 스마트폰으로 전송된 화면은 렌즈를 통하여 확대되어 가상현실을 볼 수 있게 해준다.

그림 12는 콘텐츠의 내부 모습이다. 콘텐츠는 루브르 박물관을 모델로 하여 실제 박물관을 체험하는 것과 같은 느낌을 받도록 제작하였다.



그림 11. 디스플레이 구현
Fig. 11. Implementing the display

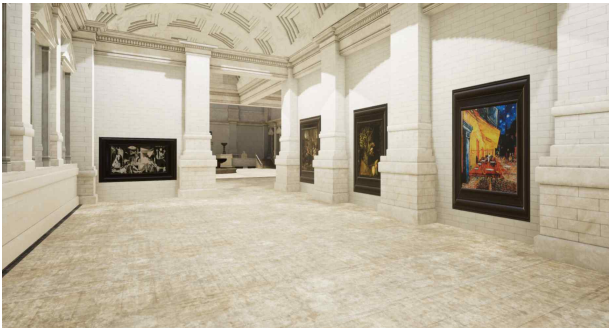


그림 12. 콘텐츠 모습
Fig. 12. a scene of the contents

4. 결론 및 향후 연구방향

본 논문은 고가의 가상현실 기기 없이 PC와 스마트폰 등이 보급화 된 기기를 응용하여 가상현실을 체험하는 시스템을 구현하였다. 표 1과 같이 기존 기기보다 가격은 대폭 낮추면서 무선 환경을 구현하였고, 기존의 기기와 마찬가지로 PC의 그래픽 처리 능력을 사용하여 높은 품질의 그래픽을 얻을 수 있었다. 다만 무선 환경에 따라 영상 데이터가 끊기거나 전송 딜레이가 증가하는 등의 단점이 있었다. 이는 원활한 환경의 와이파이를 이용하고 추후 무선통신 기술이 발전되면 해결 될 것이다.

본 연구를 통하여 가상현실의 경제적 부담감을 감소시켜 접근성을 증가시킬 수 있을 것이며, 가상현실에서 무선 환경을 구축하여 유선 환경에서 생기는 움직임의 제약을 감소시킬 수 있을 것이다. 이렇게 가상현실을 쉽게 접함으로써 가상현실의 사용자가 증가하면 많은 투자가 있을 것이고 이에 따라 더욱 양질의 콘텐츠가 제작될 것이다. 게임분야[4] 외의 교육·문화 등 다양한 분야에 대한 콘텐츠가 더욱 많이 개발된다면 현재 느린 성장세의 가상현실 시장에 더 큰 활력을 불어 넣을 수 있으며, 우리나라의 가상현실 기술을 한층 발전시킬 것으로 기대된다.

표 1. 기존 기기와 제작된 기기 비교

Table 1. Comparisons between existing devices and implemented device

	Existing devices	Implemented device
Prise (\$)	300~500	10~30
Connection	Cable	Wireless
Performance	Very High	High

참고문헌

- [1] S. A. Jung, "A study on the gesture UI to tele-presence in VR game", *Dept. of Image Design Major, Hanyang University (Master)*, pp.1-32, 2017
- [2] I. K. Cheon, "Android programming to illustrate easily", *saengneung Publishing Co*, vol. 3, pp.1-976, 2015.
- [3] H. S. Kim, " Implementation of VR object Control system using Tilt/Campass Sensors", *Dept. of Information and Communication Engineering, Dongshin University (Master)*, pp.22-30, Feb 2012.
- [4] J. H. Hwang, "Real-time VR Strategy Chess Game using Motion Recognition", *Journal of DCS*, Vol.18, No.1, pp.1-7, Feb. 2017

임은수(Eun-Su LIM)



2018년 : 대전대학교 전자공학과 (공학학사)

※관심분야 : 가상현실, 임베디드 시스템

윤성익(Sung-Yi YUN)



2018년 : 대전대학교 전자공학과 (공학학사)

※관심분야 : 가상현실, 게임개발



고영석(Yong-Suk KO)

2018년 : 대진대학교 전자공학과 (공학학사)

※관심분야 : 가상현실, 게임개발



정하영(Ha-Young JUNG)

2018년 : 대진대학교 전자공학과 (공학학사)

※관심분야 : 가상현실, 임베디드 시스템



최홍섭 (Hong-Sub CHOI)

1994년 : 서울대학교 대학원 전자공학과 (공학박사)

1995년~ 현 재: 대진대학교 전기전자공학부 교수

※관심분야 : 멀티미디어 통신, 음성 및 음향 신호처리