최종보고서

**VR과 360도 카메라를 이용한**

**실시간 스트리밍 원격제어 차량**

**(360°Avatar Driver)**

Ver. 2.0

2018.12.05

한국외국어대학교 정보통신공학과

1팀(VICER)

**문서 정보**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **구 분** | **소 속** | **성 명** | **날 짜** | **서 명** |
| **작성자** | 한국외국어대학교 | 박지훈 | 2018. 12. 05 |  |
| 한국외국어대학교 | 나윤호 | 2018. 12. 05 |  |
| 한국외국어대학교 | 류형오 | 2018. 12. 05 |  |
| 한국외국어대학교 | 유한석 | 2018. 12. 05 |  |
| 한국외국어대학교 | 유정현 | 2018. 12. 05 |  |
| **검토자** | 한국외국어대학교 | 박지훈 | 2018. 12. 05 |  |
| 한국외국어대학교 | 나윤호 | 2018. 12. 05 |  |
| 한국외국어대학교 | 류형오 | 2018. 12. 05 |  |
| 한국외국어대학교 | 유한석 | 2018. 12. 05 |  |
| 한국외국어대학교 | 유정현 | 2018. 12. 05 |  |
| **사용자** |  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| **승인자** | 한국외국어대학교 | 홍진표 | 2018. 12. 05 |  |

**개정 이력**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **버전** | **작성자** | **개정일자** | **개정내역** | **승인자** |
| **1.0** | 유정현 | 2018.11.05 | 초안 및 시나리오 작성 |  |
| **검토자** | 유정현, 박지훈 | |  |
| **1.2** | 유정현 | 2018.11.05 | 적용방안 및 기대효과 작성 |  |
| **검토자** | 유정현, 박지훈 | |  |
| **1.3** | 유한석 | 2018.11.15 | 상세설계 및 기능동작 작성 |  |
| **검토자** | 유한석, 류형오, 박지훈 | |  |
| **1.6** | 나윤호 | 2018.11.15 | 제품 설명, 시스템 구성도 작성 |  |
| **검토자** | 나윤호, 류형오, 박지훈 | |  |
| **2.0** | 박지훈 | 2018.12.05 | 최종 보완 및 수정 |  |
| **검토자** | 박지훈, 나윤호, 류형오, 유한석, 유정현 | |  |

**머리말**

본 문서는 VR 기술과 360도 카메라를 이용하여 실시간 스트리밍 원격제어 차량 운전을 제공하는 해당 시스템 구현에 대한 최종 결과를 기술한 것이다. 시스템에 대한 설명부터 시작하여, 설계 및 기능 동작, 적용방안까지 자세하게 다루고 있다.

**목차**

**1. 개요**9

**1.1. 목적**9

**1.2. 관련문서**9

**1.3. 용어 및 약어**10

**2. 제품 소개**11

**2.1. 제품 개요**11

**2.2. 제공 서비스**11

2.2.1. Unity 엔진을 이용한 App 제공11

2.2.2. 360도 영상제공11

**3. 시스템 구성도**13

**3.1. 전체 시스템 구성**13

**3.2. 세부 시스템 구성**13

3.2.1. 통신 시스템13

3.2.2. 세부 시스템14

**3.3 네트워크 구성**14

3.3.1. 블루투스 통신: Controller와 VR기기에 장착된 App과의 통신14

3.3.2. 와이파이 통신: 차량과 스트리밍 서버 간의 통신15

3.3.3. TCP 통신15

**4. 제품기능 및 기능동작**16

**4.1 본인인증 및 차량연결**16

4.1.1. 본인인증을 위한 회원가입 및 로그인 서비스 16

4.1.2. 차량 및 컨트롤러 연결을 위한 서비스 17

**4.2 차량 제어**18

4.2.1. 컨트롤러와 유저 어플리케이션 간의 블루투스 통신 기능 18

4.2.2. RC카의 Network Access 기능 18

4.2.3. 컨트롤러를 이용한 차량 제어 기능 19

**4.3 영상 스트리밍**19

**4.4 Sequence Diagram**22

4.4.1. Using VR Application sequence diagram 22

4.4.2. Control RC Car Sequence diagram22

**4.5 User Flow chart**23

**4.6 예상 시나리오**23

4.6.1. 유해 물질 위험 구역에서의 VICER23

4.6.2. 자율 주행 차량 테스트를 위한 VICER23

4.6.3. 문화 산업 분야에서의 VICER24

**5. 시스템 상세 설계**25

**5.1. Streaming**25

5.1.1. Data Encoding25

5.1.2. Media Transmission TCP Protocol26

5.1.3. Streaming Server26

5.1.4. Demension Converting26

**5.2. Controller RC CAR**28

5.2.1. Message Processing28

5.2.2. VR Application28

5.2.3. RC Controll Server29

5.2.4. RC CAR30

**6. 실제적용방안 및 기대효과**31

**6.1. 키덜트 족을 타깃으로 하는 VICER 차량 시스템**31

**6.2. 무인 차량 시스템의 테스트를 위한 VICER 차량 시스템**31

**6.3. VICER 차량 시스템을 통한 기대 효과**32

**7. 프로젝트 세부추진계획 및 세부일정**33

**8. 팀원 담당 업무**34

**그림 목차**

[Figure 1] Unity Engine을 이용한 App 11

[Figure 2] 스티칭 기법11

[Figure 3] Wrapping Frame11

[Figure 4] 전체 시스템 구성도13

[Figure 5] 통신 시스템 구성도13

[Figure 6] 세부 시스템 구성도14

[Figure 7] Baud Rate14

[Figure 8] TCP 통신14

[Figure 9] 본인인증을 위한 회원가입 및 로그인 창 16

[Figure 10] 차량 및 컨트롤러 연결을 위한 창 17

[Figure 11] 블루투스 연결 모듈18

[Figure 12] RC 카 통신 모듈18

[Figure 13] 차량제어 시스템 기능흐름 19

[Figure 14] Gear360과 Gear360 Application 블루투스 연결 20

[Figure 15] 영상 스트리밍 시작 20

[Figure 16] 중계 Server Thread 21

[Figure 17] 영상 스트리밍 결과 21

[Figure 18] Using VR Application sequence diagram22

[Figure 19] Control RC Car Sequence diagram22

[Figure 20] User Flow Chart23

[Figure 21] 영상 프레임 압축 흐름도25

[Figure 22] Server Architecture26

[Figure 23] 스티칭 된 Frame27

[Figure 24] wrapping frame27

[Figure 25] 사용자가 보는 화면27

[Figure 26] VR application28

[Figure 27] Plugin29

[Figure 28] RC Car 시스템 구성요소30

[Figure 29] 헤더파일30

**표 목차**

[Table 1] 관련 문서9

[Table 2] 용어 및 약어10

[Table 3] 영상 압축 기술종류25

[Table 4] Message Processing28

[Table 5] 프로젝트 세부 추진 일정33

[Table 6] 팀원 담당 업무34

1. 개요

해당 시스템은 VR 기술과 360도 카메라를 이용한 실시간 스트리밍 원격제어 차량 운전을 제공한다.

즉, 사용자가 VR기기를 장착하고, 360도 카메라가 부착된 차량과 원격으로 연결된 컨트롤러를 조작함으로써, 실시간으로 스트리밍 되는 차량 주변의 상황을 보며 차량을 제어할 수 있도록 구현된 시스템이다.

* 1. 목적

본 프로젝트는 VR 기술과 360도 카메라를 이용한 실시간 스트리밍 원격제어 차량 시스템을 구축하는데 그 목적이 있다.

즉, 기존의 무선조종 RC카가 제공하던 2D영상의 한정된 시야를 극복하고, 360도 카메라를 이용하여 4D영상을 실시간으로 송신 받아 운전자에게 사방의 상황을 현장감 있게 제공할 수 있는 시스템을 만들고자 하였다.

또한, 소켓서버를 통한 실시간 제어를 구현함으로써 거리의 제약 없이 지구 반대편에서도 원격으로 이용 가능한 시스템을 만들고자 하였다. 현재 시장에 출시되어 있는 원격제어 제품(드론, 무선조종 RC카)들은 P2P 통신방식을 사용하고 있기 때문에, 거리에 따른 제한이 있기 때문이다.

프로젝트를 진행하기 위해 다음 사항들을 구체적으로 명시하고 구현하였다.

(1) Unity Engine을 활용하여 app device를 원격 운전자에게 제공

(2) Samsung gear 360 SDK를 활용하여 영상 스트리밍을 구현하고 원격 운전자는 gear VR를 장착하여 이를 보게되므로, 현장감 넘치는 실시간 3D 운전 영상을 제공

(3) 원격 운전자는 컨트롤러(핸들, 페달, 기어)를 사용하여 실내에서 원격으로 차량을 제어

(4) 차량 소유주와 원격 운전자에게 App 제공

* 1. 관련 문서

|  |  |
| --- | --- |
| **출판사** | **문서 제목** |
| 위키북스 | 절대강좌! 유니티 5 |
| 한빛아카데미 | 안드로이드 프로그래밍 |
| 한빛미디어 | 초보자를 위한 유니티 5 입문 : 설치에서 3D와 2D 게임까지 |

[Table 1] 관련 문서

* 1. 용어 및 약어

|  |  |  |
| --- | --- | --- |
| **용어 및 약어** | **풀이** | **비고** |
| WAS | Web Application Server, 웹 애플리케이션과 서버환경을 만들어 동작시키는 기능을 제공하는 소프트웨어 프레임워크. 웹 애플리 케이션 서버는 동적 서버 콘텐츠를 수행하는 것으로 일반적인 웹 서버와 구별됨 |  |
| MTU | 어떤 데이터링크에서 하나의 프레임 또는 패킷에 담아 운반 가능한 최대 크기  - 수용 가능한 상위 계층 데이터(헤더 포함된 전체 싸이즈)의 크기 |  |
| 스티칭 기법 | 영상을 이어 붙이는 기법, 앞, 뒤, 좌우, 위, 아래의 모든 모습이 나오는 360도 영상을 위한 방법 |  |
| VICER | VR(Vritural Reality)과 ICE(Information and Communication Engineering)의 합성어, 원격 차량 제어 시스템의 이름이자 팀명 |  |

[Table 2] 용어 및 약어

1. 제품 소개
   1. 제품 개요

키덜트’(Kidult)란 어린이를 뜻하는 ‘키드’(Kid)와 어른을 의미하는 ‘어덜트’(Adult)의 합성어로, ‘아이같은 감성과 취향을 지닌 어른’을 지칭한다. 이를 기반으로 하여, 최근, RC카 무선 조종 장남감의 수요가 증가하는 것을 확인 할 수 있었고, 본 시스템이 키덜트 장난감의 수요가 증가하는 흐름에 알맞다고 생각하였다. 따라서 이 제품은 2D 영상만 제공하는 기존 RC 카들과 차별화하여 360도 카메라를 장착한 RC 카를 활용해 생동감 있게 그것을 원격제어하는 컨텐츠를 제공한다.

* 1. 제공 서비스
     1. **Unity 엔진을 이용한 App 제공**



[Figure 1] Unity Engine을 이용한 App

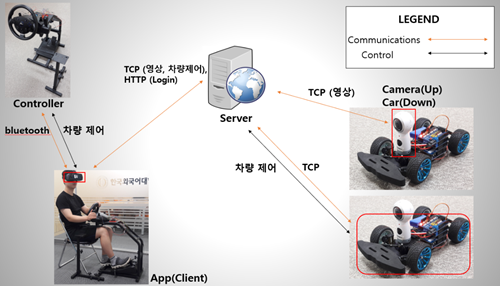
사용자에게 어플리케이션 이용을 위해 입력하는 정보들, 이를테면 이름, 전화번호, 차량 번호, 시리얼 번호 등은 요청 시, 어플리케이션과 연동된 AWS에서 DB에 접속하여 그것들을 저장하거나, 다른 정보들과 비교하거나, 자신이 원격제어 할 차량에 대해 접속하는 서비스를 제공한다.

* + 1. **360도 영상제공**

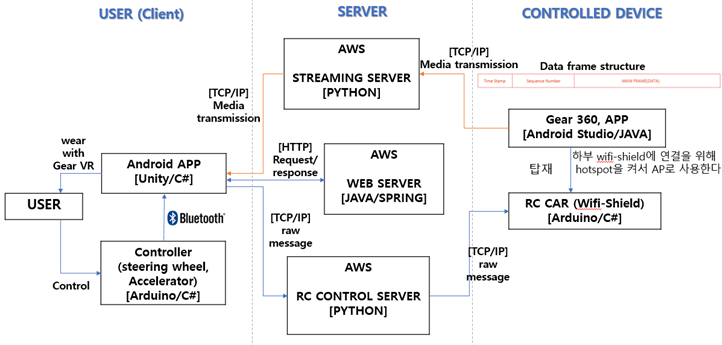
****

[Figure 2] 스티칭 기법 [Figure 3] Wrapping Frame

차량의 부착된 삼성 기어 360을 이용해 3D 영상을 실시간 스트리밍 하여 스티칭 프레임을 구체의 Texture로 Wrapping시킨후 씌워진 프레임을 VR기기에 연결된 App으로 전송해 원격으로 차량을 제어하는 서비스를 제공한다.

1. 시스템 구성도
   1. 전체 시스템 구성

[Figure 4] 전체 시스템 구성도

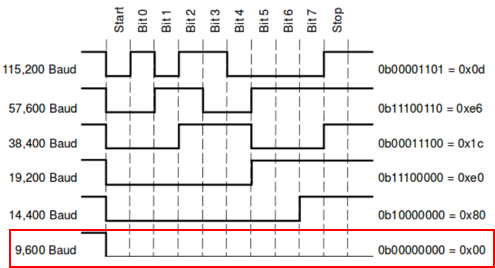
* 1. 세부 시스템 구성
     1. **통신 시스템**

[Figure 5] 통신 시스템 구성도

* + 1. 세부 시스템



[Figure 6] 세부 시스템 구성도

* 1. 네트워크 구성
     1. **블루투스 통신: Controller와 VR기기에 장착된 App과의 통신**

[Figure 7] Baud Rate

Built-in 된 Serial 통신용 0,1번 핀 외에 다른 디지털 핀으로 Serial 통신을 원활하게 해주기 위해 SoftwareSerial 라이브러리 사용. 블루투스 모듈과 통신을 위해 UART 통신을 사용함. 블루투스 근거리 무선통신 규격의 하나로 2.45HZ 주파수 반경 안에서 작동. 비동기 통신이므로 송, 수신간의 Baud Rate 정함.

* + 1. **와이파이 통신: 차량과 스트리밍 서버간의 통신**

Phpoc(와이파이 쉴드)와 아두이노를 연결하기 위해 Clock을 통하여 동기화하는 동기식 통신방식으로써 SPI 라이브러리를 이용한다. SPI는 최고 70MHZ에 이르는 빠른 통신 속도와 양방향 통신이 가능함으로써 데이터 단위 제약이 없는 장점을 가졌다. SPI 라이브러리를 통해 내장된 TCP Protoco로 통신한다.

* + 1. **TCP 통신**



[Figure 8] TCP 통신

RTSP Protocol을 참조하여 독자적으로 Media Transmission TCP protocol을 만들었다. Sender는 프레임의 길이를 측정하여 해당 정보를 4Byte ‘Frame Data Size’로 추가해준다. 마지막으로 Sender identification을 위해 1Byte Header를 추가하여 캡슐화를 완료한다 Receiver는 수신 받은 Pcaket을 역캡슐화하고 수신 받을 프레임의 길이를 예측한다. 프레임 데이터가 들어오게 되면 프레임 데이터를 조합하여 화면에 띄워준다

1. 제품기능 및 기능동작
   1. 본인인증 및 차량연결

해당 시스템은 원격으로 차량을 제어하기 위해서, 가장 먼저 어플리케이션을 통한 간단한 회원가입과 본인확인 및 차량연결 서비스를 제공한다. 회원 및 차량 정보는 Oracle 기반의 DBMS(Database Management System)에 저장하였고, 서비스 처리를 위한 엔진으로는 Tomcat Server를 이용하였으며, 서버와 데이터베이스 간 통신을 위한 언어로는 Spring Framework를 이용하였다.

* + 1. 본인인증을 위한 회원가입 및 로그인 서비스

|  |  |
| --- | --- |
| EMB000043ec5ee1  ( 회원가입을 위한 창 ) | EMB000043ec5ee0  ( 로그인을 위한 창 ) |

[Figure 9] 본인인증을 위한 회원가입 및 로그인 창

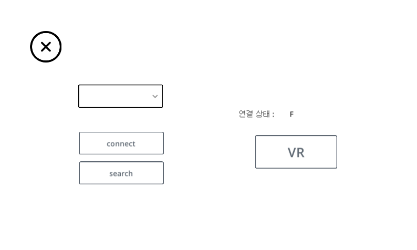
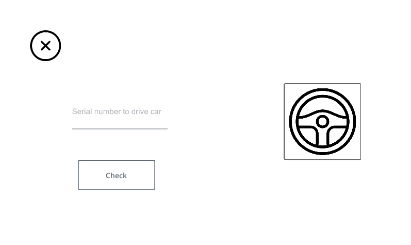
1. **회원가입 서비스**

① 어플리케이션 첫 화면의 ‘Sign up’ 버튼을 누르면, 로그인을 위한 계정을 생성하는 회원가입 창으로 넘어간다.

② 사용자는 자신의 ID, Password, E-mail, 생년월일, 차량의 시리얼 번호 등 계정 생성을 위한 정보를 입력하고 ‘SIGN UP’ 버튼을 누르면, 어플리케이션은 해당 정보를 담은 메시지를 HTTP POST 방식을 통해 서버로 전송한다.

③ 사용자로부터 계정 생성을 위한 정보를 받으면, 그것을 Oracle 기반의 데이터베이스에 저장하는데, 기존에 데이터베이스에 저장된 정보와 비교하여, 중복되는 데이터가 있으면 회원가입을 다시 할 수 있도록 response 하고, 중복되는 데이터가 없으면 회원가입을 처리하여 로그인 화면으로 넘어가도록 하였다.

1. **로그인 서비스**
2. 어플리케이션의 첫 화면에서, 사용자가 자신의 ID와 Password를 입력하고, Login 버튼을 누르면, 해당 정보는 HTTP POST 방식을 통해 Server로 request 된다.
3. Server는 사용자로부터 request 받은 정보를 데이터베이스의 회원 정보 테이블에 있는 데이터들과 비교한다.
4. 해당 정보가 테이블에 등록 되어있는지 그 유효성이 확인되면, 차량의 시리얼 번호를 입력할 수 있는 창으로 넘어가도록 처리해준다.
   * 1. 차량 및 컨트롤러 연결을 위한 서비스



(차량 연결을 위한 시리얼 넘버 인증 창) (블루투스 통신을 통한 컨트롤러 연결 창)

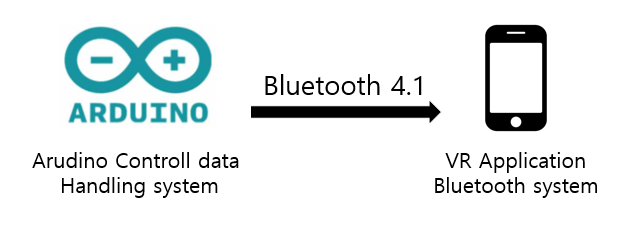
[Figure 10] 차량 및 컨트롤러 연결을 위한 창

1. **차량 인증 서비스**
2. 로그인을 하게 되면 차량을 인증할 수 있는 창으로 넘어간다.
3. 사용자는 자신의 차량의 시리얼 번호를 입력하고 check 버튼을 누르게 되면, 해당 시리얼 번호 정보를 HTTP POST 방식을 통해 서버로 전송한다.
4. 전송 받은 시리얼 번호와 데이터베이스 회원 정보 테이블에 저장된 시리얼 번호를 비교하여, 로그인 된 사용자 차량의 시리얼 번호임이 확인되면, 차량에 부착된 360도 카메라가 촬영하는 영상을 받을 수 있는 환경을 구성해주고, 컨트롤러 연결을 위한 창으로 넘어가도록 처리해준다.
5. **컨트롤러와 차량 연결 서비스**
6. 사용자의 차량과 연결이 되면, 어플리케이션은 컨트롤러 연결을 위한 창으로 전환된다.
7. 사용자는 휴대폰 상의 블루투스 기능을 켜고, 어플리케이션 상의 ‘Search’ 버튼을 통해 블루투스 모듈이 장착된 컨트롤러를 찾을 수 있다.
8. ‘Connect’ 버튼을 누르면, 해당 차량을 운전하기 위한 컨트롤러와 연결되고, VR 버튼을 통해 차량으로부터 촬영되는 영상을 받아볼 수 있다. 이를 통해 시스템 이용을 위한 본인인증 및 차량연결이 완료된다.
   1. 차량 제어
      1. 컨트롤러와 유저 어플리케이션 간의 블루투스 통신 기능

1. 유저는 어플의 블루투스 선택화면에서 Search 버튼을 통해 Connect 가능한 컨트롤러를 찾는다

2. 자신이 사용할 컨트롤러의 MAC 번호를 확인 후 Connect 버튼을 눌러 어플리케이션과 컨트롤러가 연결됨을 확인한다.

3. 연결되는 순간부터 컨트롤러의 제어정보를 블루투스 통신으로 받아와 서버로 전송해주게 된다.



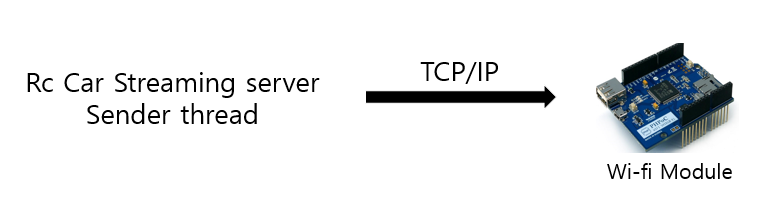
[Figure 11] 블루투스 연결 모듈

* + 1. RC카의 Network Access 기능

1. RC카에 전원을 공급하여 작동을 시작하게 한다.

2. RC카에 부착된 Wi-fi 기능을 이용하여 Network 설정을 진행한다.

3. 설정을 완료한 뒤 다시 전원을 공급하게 되면 서버에 자동으로 접속되어 제어정보를 받게 된다.



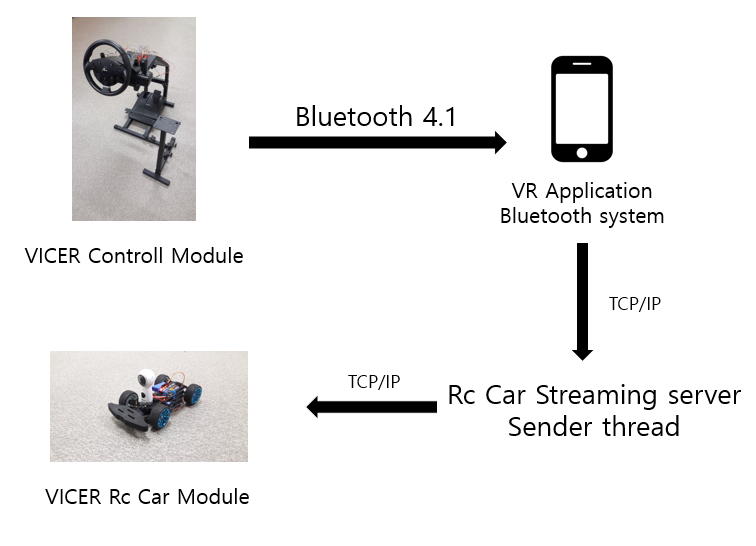
[Figure 12] RC 카 통신 모듈

* + 1. 컨트롤러를 이용한 차량 제어 기능

1. 유저가 블루투스로 컨트롤러를 어플리케이션과 연결시킨다.

2. RC카에 전원을 공급하면 자동으로 사전에 설정한 IP로 접속하게 된다.

3. 컨트롤러를 제어함에 따라 차량이 움직이게 된다.



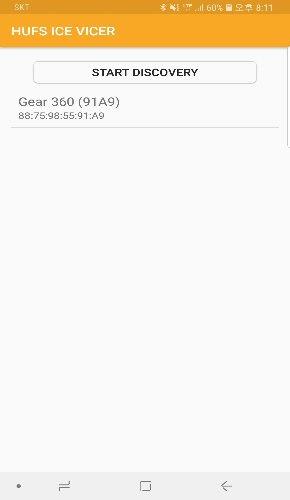
[Figure 13] 차량제어 시스템 기능흐름

* 1. 영상 스트리밍

영상 스트리밍은 사용자 Application에서 영상 받을 준비를 하고 Gear360 Application에서 Gear360과 블루투스 통신으로 연결 후 영상을 보내주면 시작된다.

영상 스트리밍은 360˚카메라인 Gear360, Gear360 Application, 사용자 Application 그리고 중계 Server로 이루어진 시스템에서 동작한다.

1. Gear360과 Gear360 Application 블루투스 연결

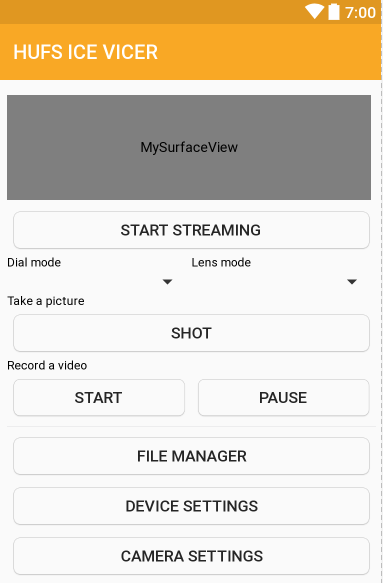
 

BlueTooth 4.1

[Figure 14] Gear360과 Gear360 Application 블루투스 연결

Gear360에서 찍은 영상을 중계 Server로 보내기위해 Gear360 Application과 블루투스로 연결한다.

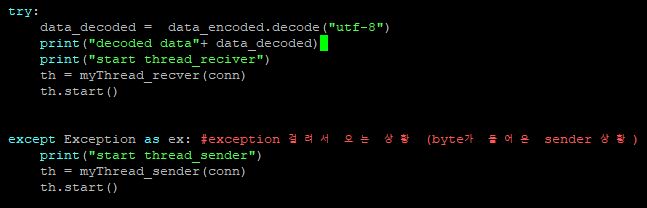
1. 중계 Server로 영상 프레임을 전달.



[Figure 15] 영상 스트리밍 시작

Gear360과 Gear360 Application을 연결 후 위의 그림의 Red Box인 Gear360 Application의 Start Steaming 버튼을 누르면 중계 Server로 Gear360에서 찍은 영상 프레임을 보내준다.

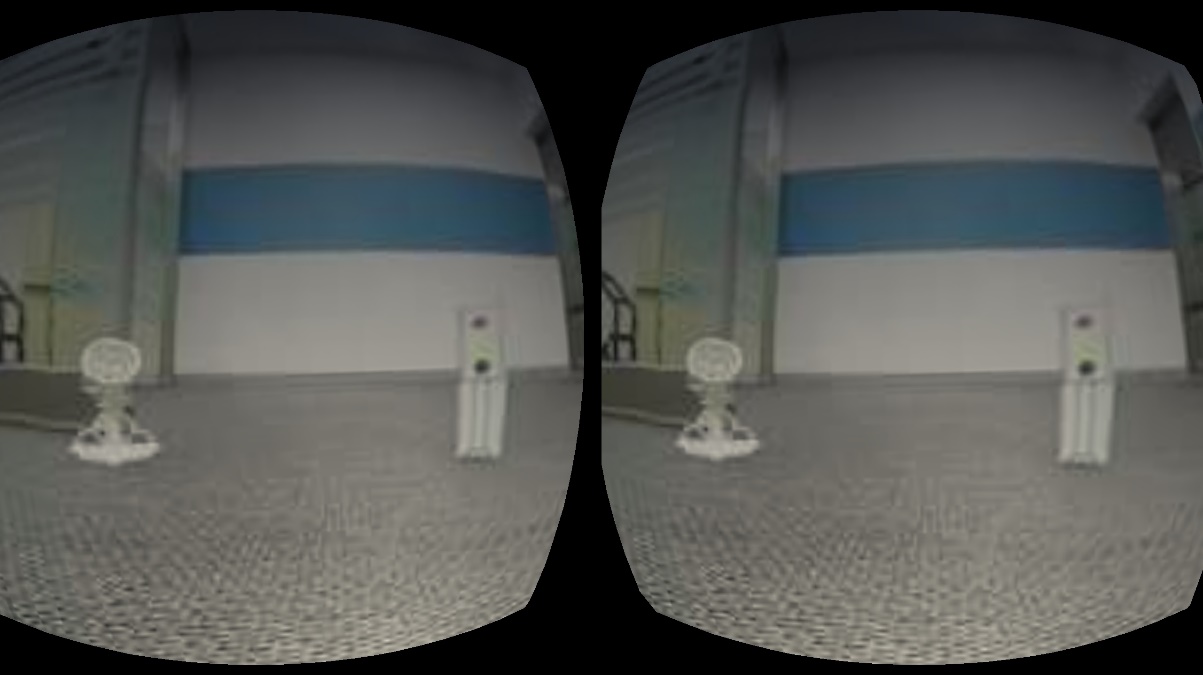
1. 영상 프레임을 사용자 Application으로 전달



[Figure 16] 중계 Server Thread

위의 보시는 코드처럼 처음에 Gear360 Application(Sender)과 사용자 Application(Receiver) 으로 접속할 때 서로 다른 Sender Thread, Receiver Thread 를 실행시켰고 Spinlock기법을 사용하여 공통변수인 프레임데이터 변수를 동시에 사용하는 일을 방지하였다. 이를 통해 Gear360 Application을 통해 받은 영상 프레임 데이터를 사용자 Application으로 전달해준다.

1. 영상을 사용자 Application에서 볼 수 있음

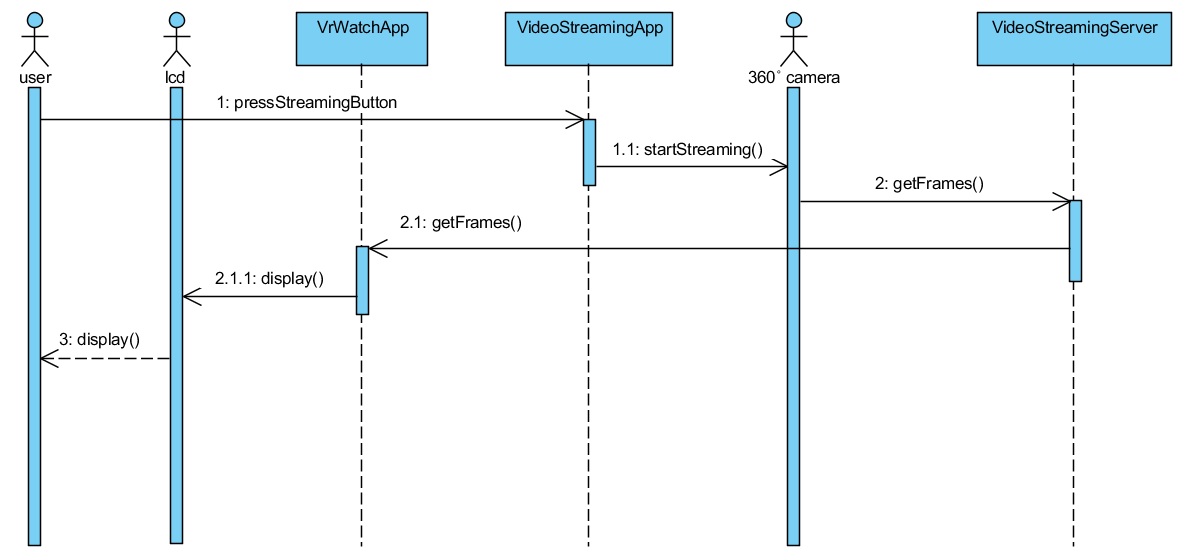


[Figure 17] 영상 스트리밍 결과

중계 Server로부터 받은 영상을 Unity에서 VR을 통해 360도전체를 볼 수 있도록 구체에 씌워 사용자 Application에 띄워준다.

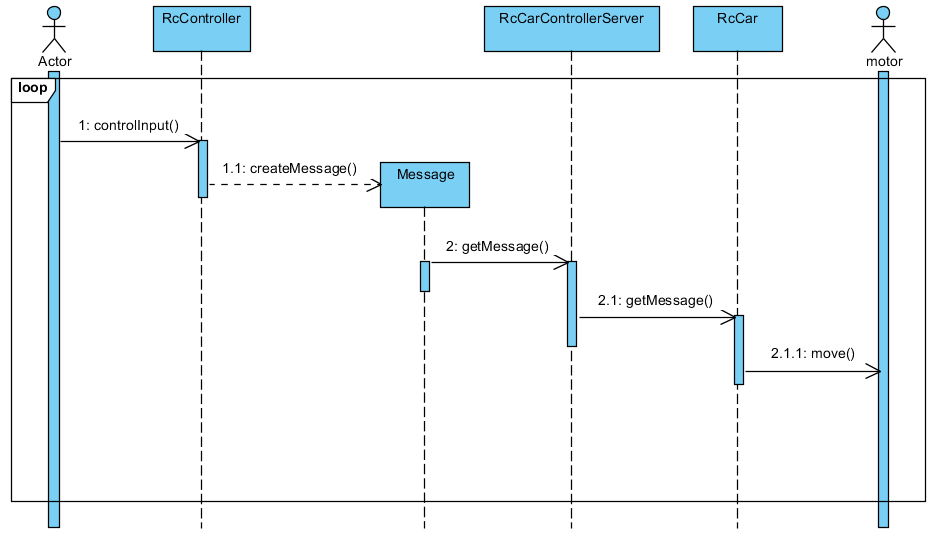
위의 4가지 절차를 통하여 영상 스트리밍 기술을 구현하였다.

* 1. Sequence Diagram
     1. **Using VR Application sequence diagram**



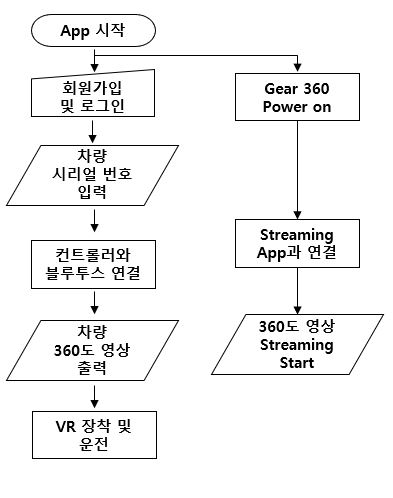
[Figure 18] Using VR Application sequence diagram

* + 1. **Control RC Car Sequence diagram**



[Figure 19] Control RC Car Sequence diagram

* 1. User Flow chart



[Figure 20] User Flow Chart

* 1. 예상 시나리오
     1. 유해 물질 위험 구역에서의 VICER

Tony는 산업 폐기물 현장에서 현장의 방사능을 체크하고 정리하는 작업자이다. 그는 매 주 작업복을 단단히 갖춰 입고서 현장에 투입되지만, 그는 그의 건강이 여간 의심스럽다. 회사는 이에 대한 해결 방안으로써 작업자들의 건강과 좀 더 편리한 현장 작업을 위해 방사능 체크를 하는 자율주행 자동차를 구입하기로 하였다. 그러나 자율주행 자동차가 작업을 대신하게 되면, Tony는 일자리를 잃게 된다. 회사는 고민 끝에 VICER의 원격 차량 제어 제품을 이용하기로 하였다. VICER의 제품은 원격으로 차량을 제어하면서 방사능 수치 체크를 해주는 자동차였다. 산더미로 쌓여 있는 산업 폐기물들을 이리저리 잘 피해 갈 수 있는 것은 360도 카메라로 촬영되는 영상을 실시간으로 받아보면서 작업자들이 직접 원격으로 운전을 하기 때문이었다. Tony는 더 이상 작업복을 입으면서도 유해 물질에 노출될 걱정을 할 필요가 없게 되었다. 그리고, 기계로 인해 일자리를 잃게 된 것이 아니라, 훨씬 더 편안하고 쾌적한 환경에서 작업을 할 수 있게 되었다. 회사는 사원들의 만족도와 일의 효율성을 종합하여 볼 때, VICER의 제품이 쓰일 또 다른 곳은 없는 지 곰곰히 생각하게 되었다.

* + 1. 자율 주행 차량 테스트를 위한 VICER

자율 주행 차량을 개발한 회사 King은 자율 주행 차량 테스트를 위해 고민 중이다. 앞 다퉈 자율 주행 차량을 내 놓은 타 회사들의 테스트 현장에서, 자동차 테스트를 위해 인간을 탑승하게 하는 것에 대한 윤리적 문제와 그로 인한 사망 사고가 많았기 때문이다. 그러던 중, VICER가 개최한 신기술 포럼에서, 자동차에 부착된 360도 카메라가 촬영하는 차량 주변 영상을 실시간으로 스트리밍하여 VR기기를 통해 볼 수 있는 기술을 소개하는 것을 보게 되었다. King은 이것을 자율 주행 차량 테스트에 접목시키기로 하였다. 자율 주행 차량 테스트를 위해서, King은 차량에 360도 카메라를 달았고, 핸드폰 어플리케이션을 통해서 360도 카메라가 촬영하는 자율 주행 차량 운전 영상을 보면서 그것을 테스트할 수 있게 된 것이다. 차량의 알고리즘이 사고를 냈다면, 그것을 바로 해결할 수 있다는 장점이 있었기 때문에 , King은 자신들이 차량을 테스트 하는 것으로 인하여 또 다른 인명피해나 사고가 일어나는 것에 대한 두려움과 불안함을 덜 수 있었다. 결과는 성공적이었다. King은 자율 주행 차량에 사람을 탑승 시키지 않은 것에 대한 윤리적 만족감을 느꼈고, 자율 주행 차량이 운전하는 모습을 운전자의 입장에서 확인해 볼 수 있어서, 다음 자율 주행 차량 설계를 위한 학습으로의 효과를 느끼기도 했다.

* + 1. 문화 산업 분야에서의 VICER

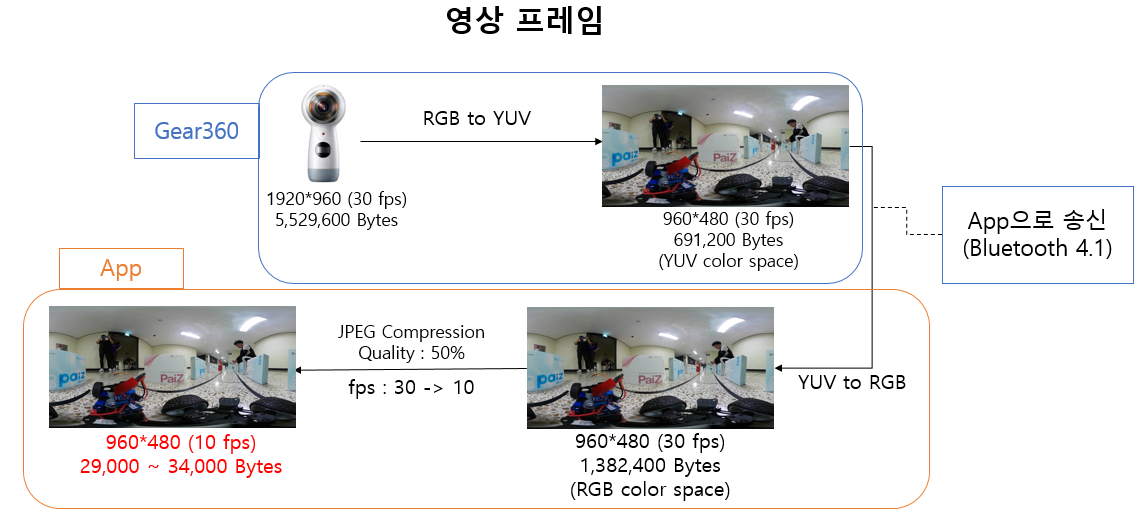
평소, 카레이싱에 관심이 많았던 학생 Justin은 무서운 속도와 함께 그와 관련된 카레이싱 사고에 대한 두려움 때문에 쉽게 도전하지 못하고 있었다. 그러던 어느 날, Justin은 테마파크에서 VR 기기를 이용한 원격 자동차 주행 놀이기구를 접하게 되었다. 자동차와 원격으로 연결된 컨트롤러를 이용하면서, 차량 주변의 상황을 360도 방면에서 촬영한 영상을 VR기기를 통해 실시간으로 확인하며 운전하는 방식이었다. 이것은 Justin이 이전에 이용했던 ‘범퍼카’나, 다른 VR 운전 놀이기구와는 확연히 다른 방식의 놀이기구였다. Justin은 이 놀이기구를 통해서, 자신이 실제로 운전하는 것과 같은 긴장감과 재미를 느꼈다. Justin은 소셜 커머스 사이트에서, 티켓을 싸게 구입하여 놀이기구를 자주 이용하였고, 아무리 이용하여도 큰 부상과 같은 사고가 일어나지 않는다는 점에서 큰 매력을 느꼈다.

1. 시스템 상세 설계
   1. Streaming
      1. **Data Encoding**

|  |  |
| --- | --- |
| **기술** | **설명** |
| JPEG | 정지 화상(단일 프레임) 압출 |
| Mothon-JPEG | 각 프레임을 JPEG으로 압축한 뒤 이어 붙이는 방식으로 압축률은 JPEG과 동일하다. |
| H.264/AVC | 저비트율 전송을 강화한 동영상 압축의 표준으로 모바일 동영상 서비스 시장에서 가장 많이 활용되는 방식이다. |

[Table 3] 영상 압축 기술종류

수신 받는 영상의 화질을 개선하기 위해 전송하는 데이터의 크기를 줄이기 위해서 영상 압축을 시도하였다. 기존에 사용되는 JPEG, Motion-JPEG 방식은 압축 효율면에서 본 프로젝트에 맞지 않는 효율을 보여 본 프로젝트에서는 H.264/AVC 방식을 통하여 압축을 진행하였다.



[Figure 21] 영상 프레임 압축 흐름도

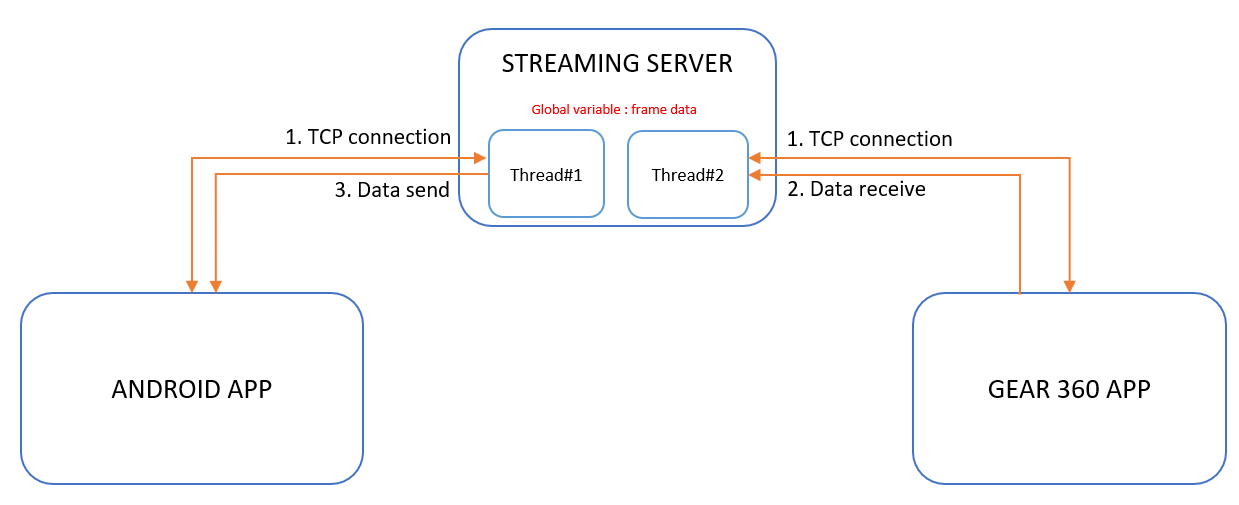
1차적으로 Gear 360 카메라에서 영상을 촬영하게 되면 내부에서 영상 프레임을 YUV 색공간에서 1/2 Size로 압축하게 됩니다. 이를 다시 블루투스 통신으로 Gear 360 App으로 전송합니다. App에서는 YUV Frame을 RGB-Bitmap으로 변환시켜주고 JPEG Compression(Quality 50%)을 통해 최종 데이터 크기를 30KB 정도로 압축시켜 주었습니다. 또한 만약의 Traffic congestion을 대비하여 FPS를 30에서 10으로 낮추었습니다.

* + 1. **Media Transmission TCP Protocol**

RTSP protocol을 참조하여 독자적으로 Media Transmission TCP protocol을 만들었다. Sender는 프레임의 길이를 측정하여 해당 정보를 4byte 'Frame Data Size'로 추가해준다. 마지막으로 Sender identification을 위해 1Byte Header를 추가하여 캡슐화를 완료한다. Receiver는 수신 받은 Packet을 역캡슐화하고 수신 받을 프레임의 길이를 예측한다. 프레임 데이터가 들어오게 되면 프레임 데이터를 조합하여 화면에 띄워 준다.

* + 1. **Streaming Server**

수신 받은 데이터를 사용자가 사용하는 Application으로 Forwarding 해주기 위한 Server이다.



[Figure 22] Server Architecture

본 서버는 TCP Protocol을 기반으로 하는 relaying(중계) 서버로 gear 360 app 으로부터 전달받은 frame data를 어떠한 처리도 하지도 않고 사용자 앱(Android App) 으로 포워딩 해주는 역할을 담당하고 있다.

* + 1. **Dimension Converting**

서버로부터 수신 받는 영상 Frame은 180도 광각 렌즈로부터 촬영된 두개의 화면이 스티칭 기법을 통해 하나의 프레임으로 나오게 된다. 이를 유니티 상에서 하나의 구를 만들어 스티칭된 화면을 구의 안쪽에 랩핑 해준다. 이렇게 되면 사용자는 구의 중심에서 사방을 바라볼 때 전방향의 이미지를 볼 수 있게 된다.



[Figure 23] 스티칭 된 Frame



[Figure 24] wrapping frame

사용자는 Figure \*\*에서 보이듯이 구체에 씌워진 이미지 한 가운데에서 사방을 둘러볼 수 있다.



[Figure 25] 사용자가 보는 화면

이렇게 구의 중심에서 사용자가 전방을 보는 화면을 나타낼 수 있다

* 1. Controller RC CAR
     1. **Message Processing**

실제 자동차는 핸들, 페달과 바퀴가 물리적으로 연결 되어있다. 하지만 본 작품에서는 멀리 있는 RC Car와 컨트롤러가 통신을 통해 제어된다. 따라서 컨트롤러의 물리적 변화와 RC Car가 동기 되어야 한다.

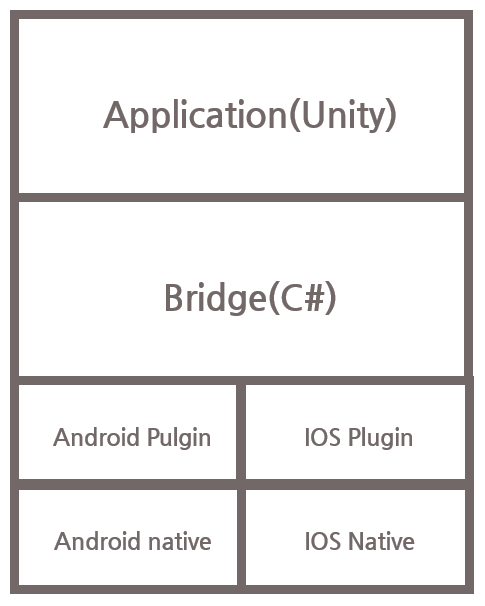
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 핸들의 가변저항 값 | 970 | 570 | | 170 |
| 각 차이 | 400 | | 400 | |
| 모터의 각도 | 57° | 90° | | 123° |
| 각 차이 | 33° | | 33° | |
| Mapping된 결과 1°당 12 | | | | |

[Table 4] Message Processing

기존의 문자 값만 보내던 방식에 비하여 모터가 돌아가는 각도마다 핸들의 값을 사상시켜 보다 자연스럽게 제어하도록 하였다.

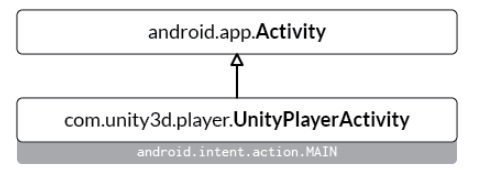
* + 1. **VR Application**

컨트롤러로부터 제어정보를 블루투스 통신으로 전달받아 RC Streaming server로 전송해주는 역할을 한다.



[Figure 26] VR application

유니티 상에서 블루투스를 비롯한, 유니티에서는 제공하지 않는 안드로이드 기능을 사용하기 위해서는 Java Library를 .jar 파일로 만들어 플러그인이라는 하나의 형태로 사용하여야 한다. 본 Application에서는 블루투스 및 화면전환을 위한 기능을 추가하기 위해 플러그인으로 만들어 구현하였다.



[Figure 27] Plugin

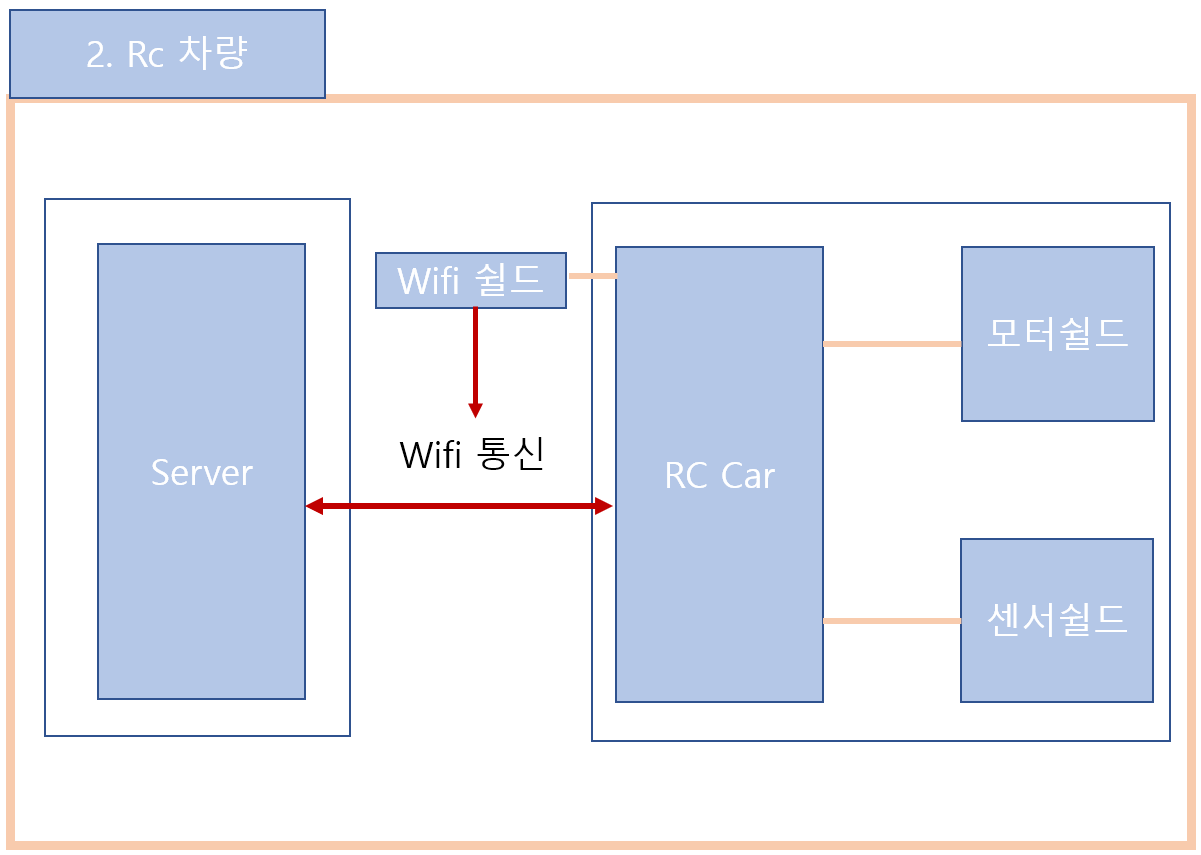
결론적으로 유니티를 안드로이드로 빌드 시키게 되면 유니티 프로젝트가 하나의 Activity가 되어 라이브러리의 Method들을 사용하는 것처럼 작동한다.

이러한 방식으로 컨트롤러의 블루투스 모듈과 Bluetooth로 통신한다. 컨트롤러로부터 제어정보를 받게 되면 TCP 통신 방식을 이용하여 서버로 전송한다.

* + 1. **RC Controll Server**

TCP Protocol을 기반으로 하는 relaying(중계) 서버로 사용자가 제어하는 Controller로부터 전달받은 조작 정보를 어떠한 처리도 하지도 않고 RC Car으로 포워딩 해주는 역할을 담당하고 있다. 구체적인 구현은 Streaming Server에서 설명한 것과 동일하다.

* + 1. **RC CAR**



[Figure 28] RC Car 시스템 구성요소

RC Car에 전력이 공급되면 wifi – shield를 이용해 사전에 설정한 무선네트워크에 접속한다.



[Figure 29] 헤더파일

네트워크에 접속하고 나면 서버로부터 제어정보를 받아 모터를 제어하여 차량이 동작하게 된다.

1. 실제적용방안 및 기대효과
   1. ‘키덜트 족’을 타깃으로 하는 VICER 차량 시스템

요즘 ‘키덜트(kidult: Kid와 Adult, 아이와 어른의 합성어)족’이 새로운 트렌드로 떠오르고 있다. 성인이 되었는 데도 여전히 어렸을 적의 분위기와 감성을 간직한 사람들을 일컫는다. 이들은 어린 시절에 경험했던 향수들을 여전히 잊지 못하고 그 경험들을 다시 소비하고자, RC카나 아날로그 게임 등과 같은 잡화 등을 이용한다. 키덜트족 관련 시장 규모는 2014년 5000억원대에서 해마다 20%씩 성장해 지난해 1조원을 넘어섰을 것으로 추정되고 있다. 올해는 시장 규모가 더욱 커질 것이라는 예상이다. 이와 함께, 입지를 늘려가고 있는 기술이 바로 ‘실시간 스트리밍 기술’이다. 이 기술에 대한 연구는 계속 되고 있으며, 미래에 가상현실, 증강현실 콘텐츠에서 쓰일 핵심 기술이 될 것으로 기대되고 있다.

해당 시스템은 이러한 트렌드적 가치관과 미래 유망 기술들을 기반으로 하고 있다. VR기기를 장착한 후, 자동차에 부착된 360도 카메라가 촬영하는 영상이 실시간으로 스트리밍 되는 것을 받아볼 수 있는 시스템 구조를 가지고 있는데, 이를 통해 사용자는 실제로 운전하는 듯한 현장감과 생동감을 느낄 수 있게 된다.

키덜트족은 경제적으로 여유가 있어 고가의 상품에도 돈을 아끼지 않는 성향을 가지고 있다. 또한, 소비가 본인에게 집중되는 1인가구들은 자신의 취미 생활이나 관심있는 물건에 대해 과감하게 투자하는 경향을 강하게 보인다. 이러한 사회적인 분위기는 해당 시스템이 보유한 기술들이 사용자로 하여금 소비되는 것에 어려움이 없는 환경을 만들 것이다.

* 1. 무인 차량 시스템의 테스트를 위한 VICER 차량 시스템

우리는 현재, 사람이 운전하지 않고도 알고리즘만으로 주행하는 ‘완전 무인 자동차’의 시대를 기다리고 있다. 국내∙외 자동차 회사들은 계속해서 자율 주행 자동차를 연구하고 있는 중이며, 이미 몇몇 회사들은 출시를 위한 테스트까지 마친 시점이다. 그들은 차를 움직이는 알고리즘이 정상적으로 작동하는지 확인하기 위해서 무인 자동차에 사람을 탑승 시켰다. 아무 일이 없었다면 다행이지만, 이로 인해 실험자의 사고가 빈번히 발생하고 있으며, 사망사고까지 일어난 상태다. ‘사람이 없는 자동차’가 과연 정상적으로 주행할 것인지 어떻게 확인할 수 있을까?

해당 시스템은 SAMSUNG Gear 360 카메라를 이용하였는데, 이는 자동차와 연결된 컨트롤러의 조작과 VR기기의 장착만으로 실제 자동차에 탑승한 듯 운전자의 시야를 실시간으로 확보해준다. 이것은 효율적인 무인 자동차의 테스트를 위한 몇 가지 이점을 가지고 있다. 기계의 작동을 확인하기 위해 사람을 이용하지 않는다는 점, 그러면서 실시간 영상 스트리밍 기술로 자동차가 주행하는 경로를 확인할 수 있다는 점, 이로 인해 잘못된 알고리즘으로 인한 사고 발생 시 재빠른 조치가 가능하다는 점, 출시 전의 테스트 뿐만 아니라 시스템 고장 확인 및 업데이트 버전의 차량 출시 시에도 단순한 연결만으로 재사용이 가능하다는 점, 운전자의 입장에서 운행을 확인함으로써 더욱 발전된 형태의 자율 주행 자동차 설계가 가능하다는 점 등이다.

사람이 없는 자동차의 출현으로 인하여, 해당 차량 시스템이 도태될 것이라고 예측한다는 것은 옳지 않다. 이것은 둘 중 누가 더 어떤 장점을 가지고 있는 지, 견주어 비교할 문제가 아니라, 자율 주행 차량을 위해서 꼭 필요한 시스템이라고 할 수 있다.

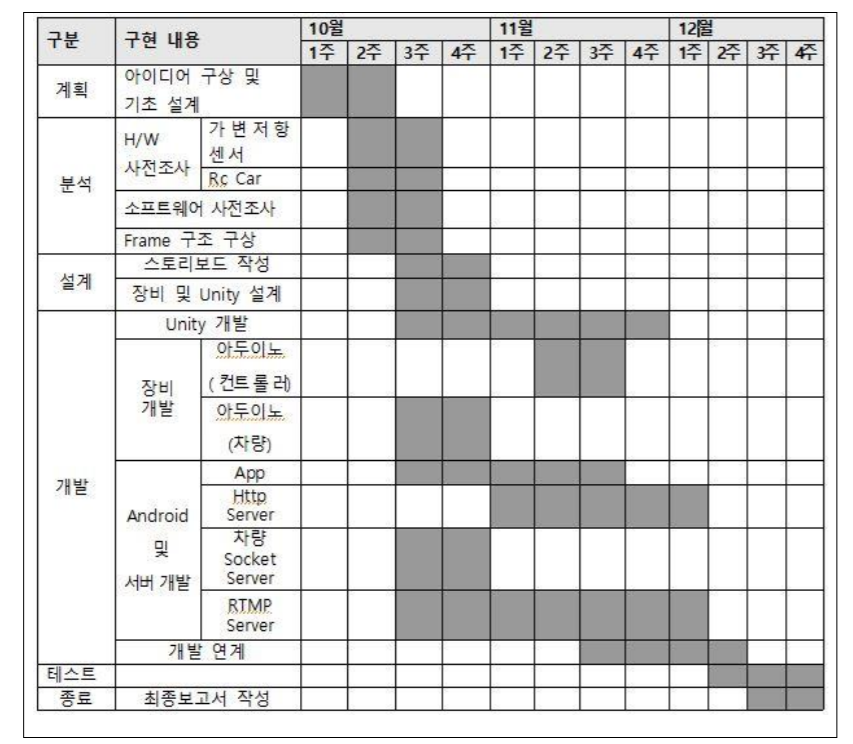
* 1. VICER 차량 시스템을 통한 기대 효과

해당 시스템을 이용하면, 운전을 하기 위해서, 자동차에 직접 탑승하지 않아도 된다. 이것이 가져오는 가장 큰 효과는 인명피해를 최소화 시킬 수 있다는 점이다. 자동차로 인한 사망 및 사고는 자신이 조심한다고 해서 일어나지 않는 것이 아니다. 다른 운전자들의 부주의를 통해서도 빈번히 일어난다. 우리는 원격으로 차량을 조정함으로써 사고로 인한 인명피해를 방지할 수 있게 된다.

또한 사용자는 실제로 차량이 있는 위치에 의존하지 않아도 된다. 사용자가 동작시키는 것은 컨트롤러 하나이다. 목적지가 어디든지 상관없다. 먼 거리이던 짧은 거리이던 간에, 사용자는 현 위치에서 원격으로 그 곳에 도착할 수 있다. 이것은 사용자에게 시간적으로 엄청난 경제적 효과를 가져다 줄 것이다.

이 시스템은 컨트롤러와 실제 자동차를 연결시킴으로써 운전이 가능하다. 연결은 시리얼 번호를 인증하는 방법으로 이루어지는데 이것은 꼭 한 대의 자동차에만 국한된다고 볼 수 없다. 적으면 한 대에서 많으면 수십대까지, 하나의 컨트롤러를 가지고 여러 대의 자동차를 운전하고 관리할 수 있다는 점에서 확장성과 용이성을 기대할 수 있다.

1. 프로젝트 세부추진계획 및 세부일정



[Table 5] 프로젝트 세부 추진 일정

1. 팀원 담당 업무

|  |  |  |  |
| --- | --- | --- | --- |
| **번호** | **이름** | **역할** | **담당업무** |
| 1 | 박지훈 | 팀장 | 전체 프로젝트 관리 및 개발 |
| 2 | 류형오 | 팀원 | Amazon Cloud Server를 통한 Database/ Back-end 개발 |
| 3 | 유한석 | 팀원 | VR Application 개발 |
| 4 | 나윤호 | 팀원 | Module 및 무인차량 제작 |
| 5 | 유정현 | 팀원 | SPRING SERVER 개발 |

[Table 6] 팀원 담당 업무