

증강 현실상 오브젝트 정합성 고도화 연구

컴퓨터 그래픽스 연구실

지도 교수

이인권

지도 조교

김현정

포켓몬고

항보정 최지웅 도회린

차례

1. 연구 개요

2. 기존 연구와 한계

- 1) 기존 연구의 한계점
- 2) 해결하고자 하는 문제 및 방법

3. 연구 및 실험 방법

- 1) 용어 설명
- 2) 실험 목표
- 3) 실험 방법 및 과정

4. 실험 결과 및 역할 분담

- 1) 실험 결과 및 기존 기술과의 비교
- 2) 개선 사항
- 3) 팀의 구성 및 팀원의 역할

1. 연구 개요



Figure 1. 증강 현실 상에 정보를 오브젝트로 표현한 인포테인먼트 시스템의 예시, WayRay¹

전기/수소 연료를 사용한 자동차가 등장함에 따라 차량 내/외부 전원 공급이 용이해지고, 자율주행 기술이 점차 발전하며 운전자 및 탑승객의 모빌리티 내에서의 활동 범위와 방식이 바뀌어 가고 있다. 최근 출시되고 있는 자율주행 레벨 3 이상에서는 운전자의 차량 제어가 의무가 아니게 되어 차량 내부에서의 다양한 활동이 가능해지고 있다. 이런 기술 발전 속에서 외부가 보이는 디스플레이를 통해 광고나 주변 건물과 운행에 대한 정보를 확인하고, 인터넷 이용, 동영상 시청 등의 다양한 오락거리를 제공할 수 있는 인포테인먼트 시스템에 대한 관심이 높아지고 있다.

오늘날 보편화된 스마트폰이나 헤드셋 등과 같은 기기들은 한정된 공간에서 느린 속도로 움직이는 환경에서 증강 현실을 구현한다. 하지만 차량과 같이 빠르게 이동하는 환경에서는 부정확한 위치 정보, 시야를 방해하는 주변 환경 요인과 이용자가 바라보는 방향에 대한 대상의 위치를 바탕으로 오브젝트를 배치하는 등 개선이 필요한 부분이 많다. 정리하자면, 본 연구에서는 외부에서 빠르게 이동하는 환경에서 증강현실 상의 오브젝트 배치의 정합성을 높여 사용자의 경험을 개선하고자 한다.

¹ WayRay (<https://wayray.com>)

2. 기존 연구와 한계

1) 기존 연구의 한계점

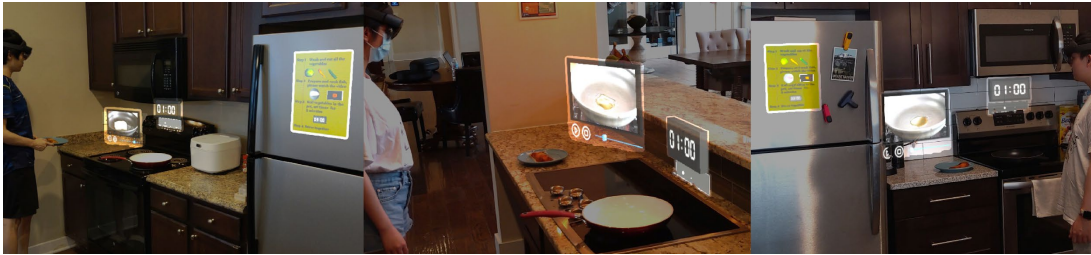


Figure 2. 제한된 증강 현실 환경에서의 오브젝트 배치²

실내와 같이 제한된 공간에서 특정 물체를 인식하여 가상 현실 환경을 구성하고, 그 위에 오브젝트를 배치하는 것은 이미 상용화 단계에 접어들고 있다. 하지만 기존의 연구는 화면의 인식 범위가 좁고, 조도가 비교적 일정한 조건 등 제한적인 상황으로 한정된다.

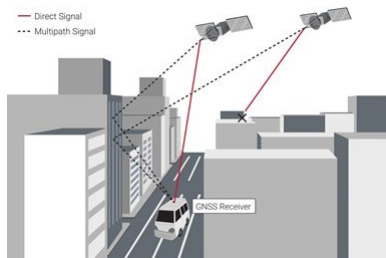


Figure 3.1. 빠른 이동속도로 인한 부정확한 위치 정보³

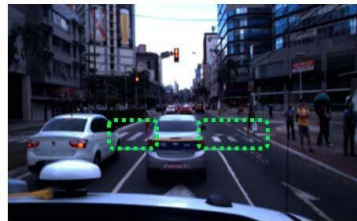


Figure 3.2. 도로 환경에 따른 폐색 영역⁴

오브젝트를 렌더링하기 위해서는 현재의 위치를 정확히 파악하고, 오브젝트의 상대 위치를 계산하는 과정이 중요하다. 그러나 차량 등 모빌리티를 통해 실외에서 빠르게 이동하는 경우 주위의 지형이나 건물 등으로 인해서 현재의 위치 정보가 부정확해지게 되고, 이를 바탕으로 계산되어 배치된 오브젝트의 위치에도 오차가 발생한다.

이뿐만 아니라 장애물(앞선 차량과 행인 등), 여러 환경 요인으로 인한 그림자의 변화 그리고 제한된 시야와 같은 변수들로 인해서 이미지에서 객체를 탐지하는 방법은 필요 시간이 길어져 실시간으로 적용하는 데에는 한계가 있다.

² Qian, Xun, et al. "ScalAR: Authoring Semantically Adaptive Augmented Reality Experiences in Virtual Reality." CHI Conference on Human Factors in Computing Systems. 2022.

³ GNSS Error (<https://www.vectornav.com/resources/inertial-navigation-primer/specifications--and--error-budgets/specs-gnsserrorbudget>)

⁴ Ryu, Seong-Eun, and Kyung-Yong Chung. "Detection Model of Occluded Object Based on YOLO Using Hard-Example Mining and Augmentation Policy Optimization." Applied Sciences 11.15 (2021): 7093.

2) 해결하고자 하는 문제 및 방법

우리는 건물의 지리적 위치 정보와 사용자의 위치 정보를 얻고, 이 정보들을 기반으로 가상 오브젝트 배치하는 방법으로 앞서 기술한 문제점을 해결하려 한다. 건물의 위치 정보들을 통해서 건물의 둘레를 계산하고, 그 둘레를 오브젝트가 이동할 수 있는 경로로 지정한다. 오브젝트의 위치는 경로 중에서 사용자와 가장 가까운 곳으로 설정되며, 경로의 바깥쪽 법선 벡터 방향을 바라보도록 회전시킨다. Figure 4 는 건물을 감싸고 있는 박스 형태의 모델링을 바탕으로, 건물의 위치와 이동 중인 대상의 위치 및 각도를 고려하여 3D 공간 상에 하얀색의 가상 오브젝트를 배치한 예시이다.

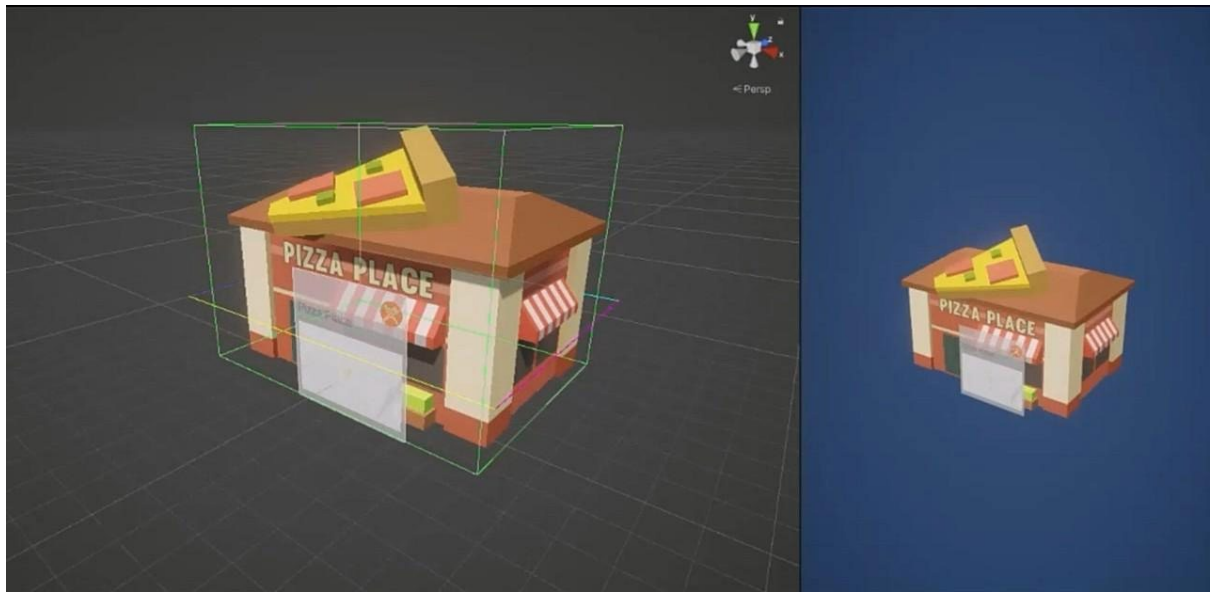


Figure 4. 건물 표면의 수직 벡터 방향으로 가상 오브젝트를 배치한 예시



Figure 5. 증강 현실상 실제 건물에 표현된 가상 오브젝트

3. 연구 및 실험 방법

1) 용어 설명

[1] RTK(Real-Time Kinematic) 포지셔닝

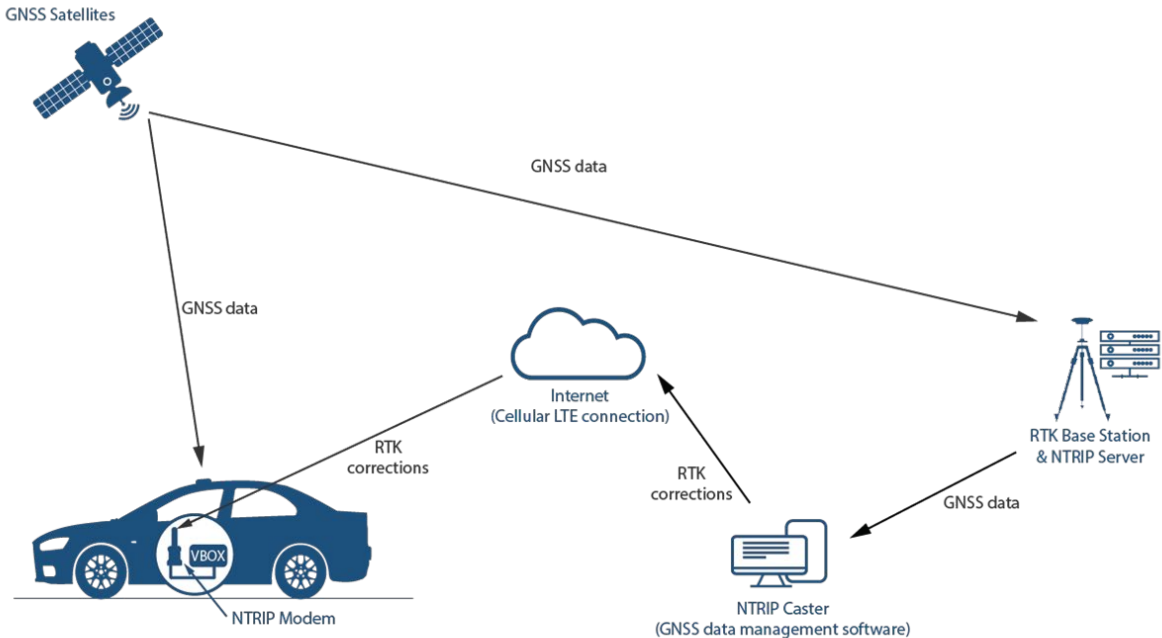


Figure 6. RTK 기술의 원리⁵

위성과의 거리를 통해 현재 위치를 파악하는 GPS 기술은 대기권 및 전리층 등의 존재, 그리고 인공위성 자체의 위치 오차로 인해 최소 수 미터에서, 많게는 100 여 미터의 오차가 발생한다. RTK는 이러한 위성 기반 위치 정보를 보정하는 기술이다.

정확한 위도, 경도 및 고도를 알고 있는 위치에 기지국을 설치하고, 해당 기지국에서 GPS 신호를 수신하면 해당 위성에서 발생한 오차를 인지할 수 있다. RTK 장비는 주변의 기지국들과 통신하여 각 위성의 오차 데이터를 파악하고, 자신이 수신한 GPS 데이터에 오차만큼 역계산 하여 현재 위치를 파악한다. 이러한 과정을 통해 RTK 장비는 최소 수 센티미터 단위의 오차로 위치를 파악할 수 있다.⁶

[2] 모델링

모델링이란 가상 오브젝트가 이동하는 경로를 결정하기 위한 위치 정보의 집합이다. 특정 건물 상의 오브젝트의 이동 경로는 각 좌표를 순차적으로 연결한 간선들로 구성된 닫힌 다각형의 형태를 이룬다. 또한 해당 간선과 현재 차량의 진행 방향을 통해 오브젝트의 회전 각도를 구한다.

⁵ VBOX Automotive – Real Time Kinematic (RTK)

<https://www.vboxautomotive.co.uk/index.php/en/component/content/article/2-products/109-real-time-kinematic-rtk>

⁶ Edwards, S. J., et al. "A methodology for benchmarking real time kinematic GPS." Survey Review 35.273 (1999): 163-174.

2) 실험 목표

실험을 진행하며 아래 요소들을 중점적으로 확인하고자 하였다.

1. RTK, AR 디스플레이 등의 장비가 정상적으로 연동되는지 확인한다.
2. RTK 장비에서 제공된 위치 정보를 통해 현재 위치가 올바르게 인식되는지, 인식된 위치를 바탕으로 건물 렌더링이 정상적으로 동작하는지 확인한다.
3. 렌더링 된 건물을 기준으로, 특정 건물을 기준점으로 하여 교정(calibration)을 진행한다. 교정 대상으로는 건물의 위치, 방향, 시야각 등이 있다.
4. 마지막으로, 렌더링된 건물 상에 현재 위치를 바탕으로 자연스러운 위치에 오브젝트가 정상적으로 표시되는지 확인한다.

3) 실험 방법 및 과정

[1] 실험 경로 및 대상 건물 목록 선정

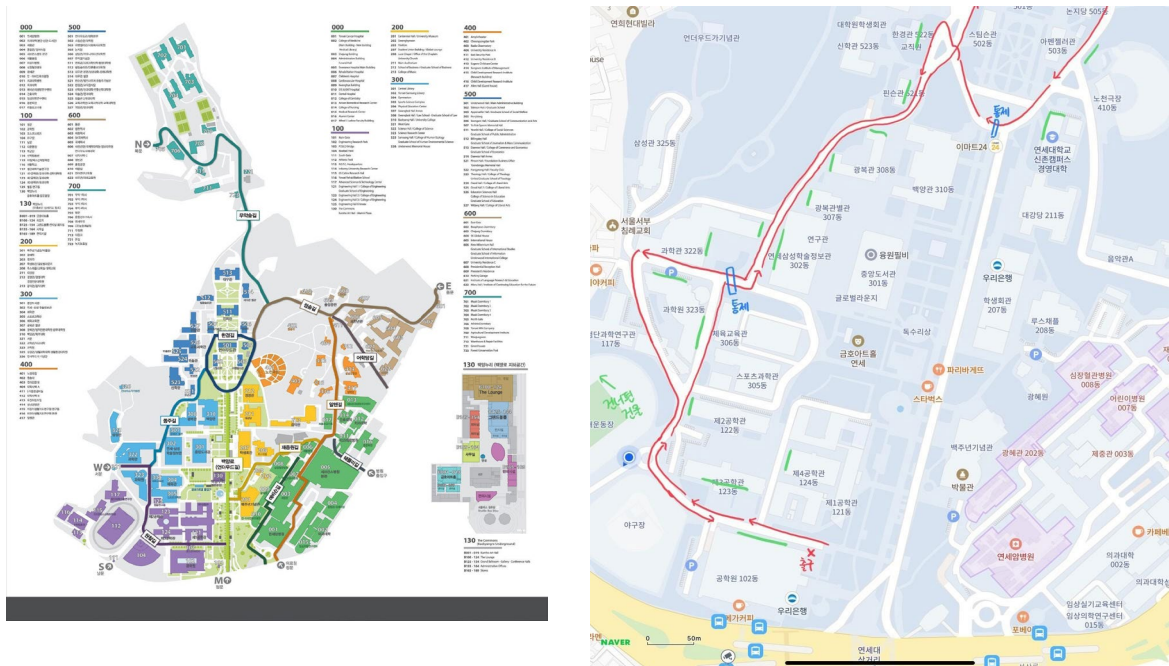


Figure 7. 연세대학교 신촌 캠퍼스 지도와 차량에서 사용할 디스플레이가 이동 방향을 기준으로 좌측에 위치한 경우의 경로⁷

상암동 자율주행 자동차 시범운행지구, 연세대학교 신촌캠퍼스 내부 등의 경로에 대한 답사를 통해 실험이 용이한 환경을 탐색하였다. 초기에는 상암동을 검토하였으나, 통제되지 않은 공도라는 환경의 특성상 속도 제한, 주변 차량의 방해, 교통 체증의 제약 사항이 실험을 진행하기에는 적합하지 않다고 판단하였다. 이에 비교적 교통량이 적고 반복적 실험이 용이한 연세대학교 캠퍼스 내에서 경로를

⁷ 연세대학교 신촌캠퍼스 지도, <https://www.yonsei.ac.kr/sc/intro/openmap.jsp>

구성하였다.

최종적으로 경로를 공학원 주차장 출구를 시작으로 캠퍼스 서부의 도로를 따라 이동한 후, 캠퍼스 중앙의 언더우드관에서 회차하여 다시 서쪽 도로를 통해서 공학원으로 돌아오게 구성하였다.

[2] 데이터 수집 및 가공

건물 상의 오브젝트를 배치하기 위해서는 해당 건물의 좌표(위도, 경도) 및 고도 정보가 필요하다.

[1]에서 선정한 건물의 표면을 계산하기 위한 좌표 정보를 수집한다. 모든 건물의 표면은 지표면에 수직하다고 가정하고, 건물의 단면, 즉, 위에서 수직으로 바라본 형상의 경로를 이용한다. 예를 들어 대상 건물 단면의 모습이 사각형이라면 4 개의 모서리를 표현할 수 있는 각 꼭짓점의 좌표를 구한다.

수집한 좌표와 고도 데이터의 가공을 위한 간단한 웹 에디터와 서버를 제작하여 실시간으로 데이터를 수정하고, 이를 즉시 클라이언트에서 렌더링에 필요한 정보를 로드할 수 있도록 구성한다. 수집한 좌표 데이터를 에디터에 입력하면 Google Maps SDK 를 이용하여 지도상에 시각화하고, Google Maps API 를 이용하여 획득한 해당 좌표의 고도 데이터를 종합하여 서버에 저장한다. 아래는 웹 에디터의 모습과 이러한 과정을 구체적인 단계별 시퀀스 다이어그램으로 표현한 것이다.

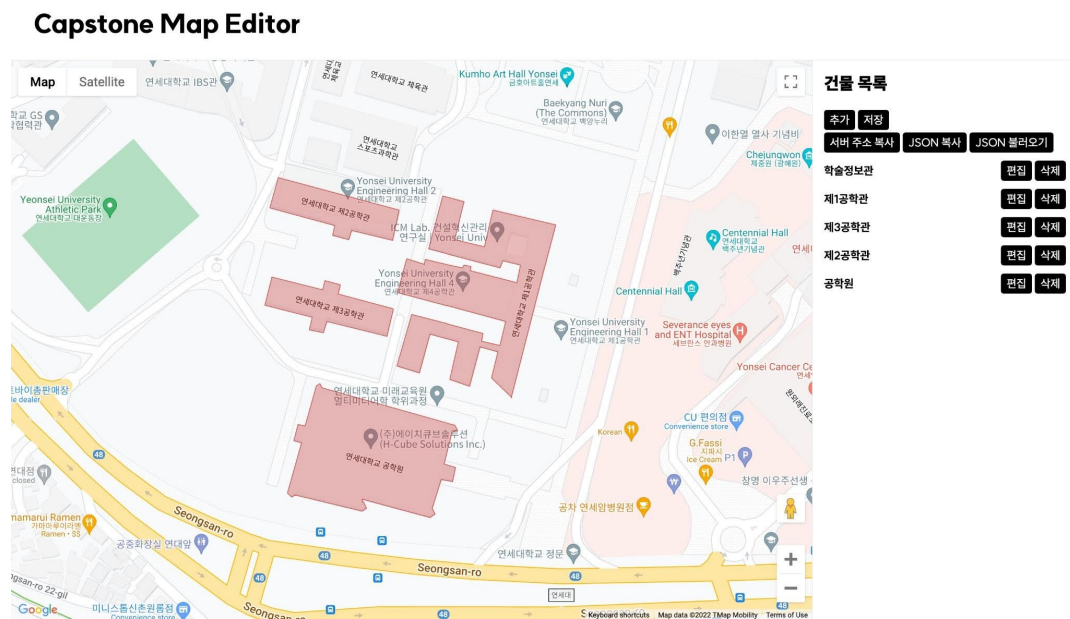


Figure 8. 데이터 가공을 위해 제작한 웹 에디터 스크린 샷

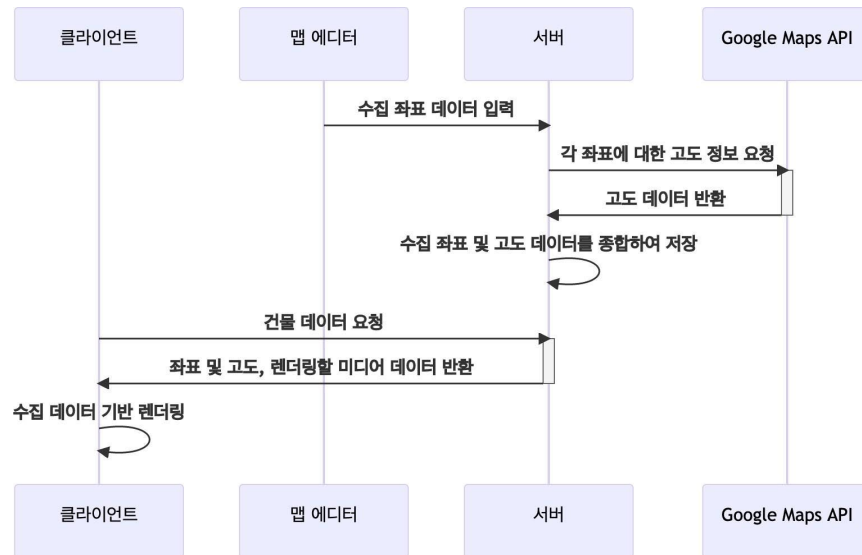


Figure 9. 수집 데이터의 저장 및 가공, 로드 절차에 대한 시퀀스 다이어그램

[3] 모델링 구현

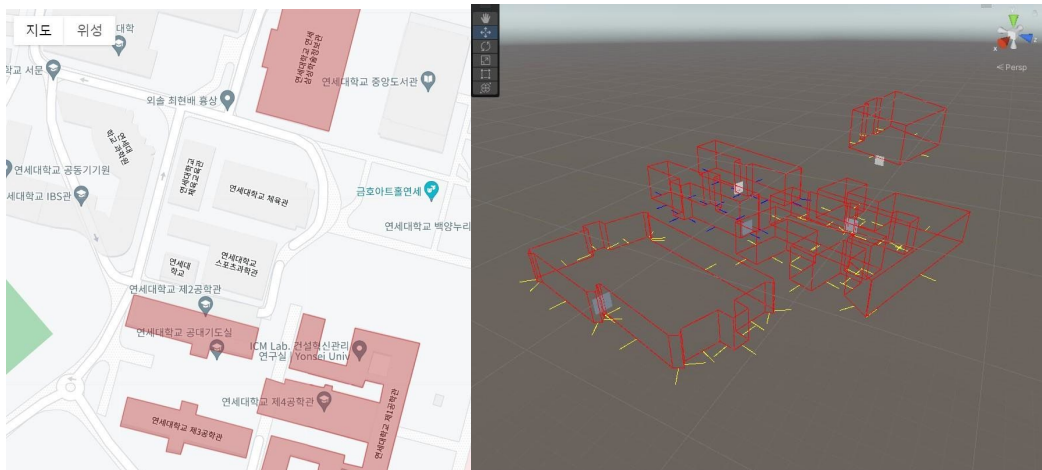


Figure 10. 모델링할 건물들의 목록과 Unity에서의 모델링 결과

앞의 [2]에서 확보한 좌표 데이터를 바탕으로 Unity 엔진을 이용하여 모델링을 생성한다. AR 디스플레이에 적용하기에 앞서 PC 프로그램 및 안드로이드 애플리케이션 형태로 빌드하여 테스트를 진행한다.

서버로부터 내려받은 건물의 좌표 및 고도 데이터를 클라이언트의 좌표계와 거리 단위에 맞춰 변환한 후, 각 건물의 좌표 목록을 순서대로 이은 간선으로 오브젝트의 이동 경로를 생성한다. 이 과정에서 위도, 경도 등 지리 좌표계를 기반으로 구성된 데이터를 Unity 엔진에서 이용할 수 있도록 Unity 좌표계 및 거리 단위로 변환하여 사용한다.

클라이언트의 월드에 건물 모델링 데이터를 그려주는 기능을 이용하여 빨간 선을 통해서 건물 모델링의 형태를 확인할 수 있고, 바닥에 표시된 노란 선과 파란 선을 통해 간선들의 법선 벡터를 확인할 수 있다. 또한 해당 모델링 상에 흰색 사각형 형태의 가상의 평면 오브젝트를 배치하여, 오브젝트가 이동 경로 내에서 대상과 가장 가까운 위치로 이동되면서, 현재 위치를 포함하는 간선과 엣터를

외적해서 나오는 법선 벡터 중 바깥으로 향하는 벡터를 바라보게 회전하는 모습을 확인할 수 있다.
(Figure 10의 우측 이미지)

[4] RTK 장비 연동

정확한 위치에 정보를 렌더링하기 위해서는 현재 위치를 정확하게 파악하는 것이 중요하다. 따라서 외부에서 빠른 속도로 이동하는 상황에 따른 현재 위치의 부정확성을 보완하기 위하여, RTK 센서를 이용하여 높은 정확도의 위치 및 방향 정보를 획득한다.

실험에 사용된 장비는 u-blox사의 ZED-F9P이다. 실험에는 아래 프로토콜 및 패킷을 사용하였다.

- NMEA 프로토콜 - GGA 패킷(Global positioning system fix data)
- UBX 프로토콜 - NAV-HPPOSLLH 패킷(High precision geodetic position solution)

본래 RTK 장비와 스마트폰 간에 블루투스를 통한 시리얼 통신을 통해 위치 정보를 수신하도록 개발을 진행하고 있었다.

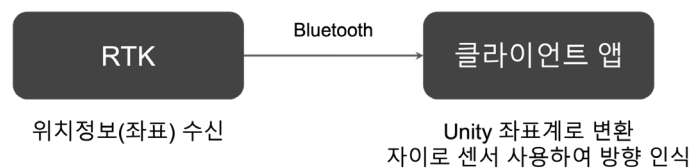


Figure 11. RTK - 클라이언트 앱 연동 방안 (기존)

그러나 우리가 사용하는 RTK 장비가 블루투스를 통해서는 제한된 정보만을 제공한다는 문제점이 있었다. 또한 안드로이드 스마트폰에서 직접 통신하려는 경우, 드라이버 수준의 개발이 필요하여 상당히 오랜 시간이 소요되는 것으로 확인하였다. 그러한 이유로 Windows PC를 이용한 데이터 서버를 추가하고, 해당 서버에서 TCP 통신을 통해 앱에 위치 정보 데이터를 제공하는 방식으로 구현을 변경하였다.

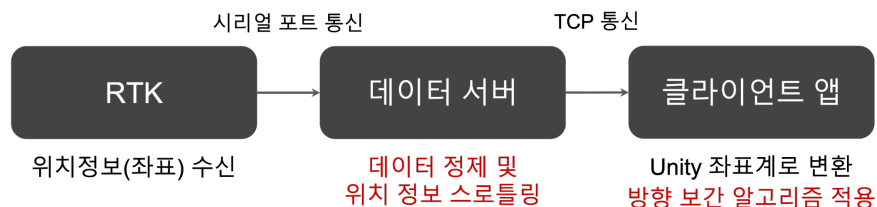


Figure 12. RTK - 클라이언트 앱 연동 방안 (최종)

RTK 장비의 민감도가 높다 보니 너무 많은 양의 패킷이 발생하여 데이터 통신에 지연이 발생하는 문제가 있었다. 이는 데이터 서버에서 버퍼 풀을 두어, 버퍼를 초과하는 패킷은 드랍하는 방식으로 구현하여 해결하였다.

또한, 기존에는 차량의 진행 방향 계산에 있어 자이로 센서를 사용하려고 하였다. 짧은 거리를 이동하며 테스트했을 때는 허용할 수 있는 정도의 오차인 것으로 추정했으나, 테스트가 길어질수록 기존 위치의 bias가 누적되어 오차가 증가하는 모습을 보였고, 특히 회전 교차로 등 급격히 회전하는 경우 센서의 인식 속도가 실제 방향 변화를 따라오지 못하는 현상이 있었다.

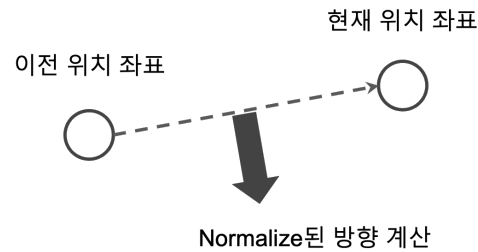


Figure 13. Normalize 된 방향 계산 방법

때문에 우리는 센서값을 이용하지 않고, 위치 좌표를 이용해 현재 위치와 이전 좌표 간의 법선 벡터 방향을 이용하였고, 여기에 역치값을 설정하여 outlier 값을 완화하는 조치를 취하였다.

[5] 실제 경로 주행 및 검증

위의 단계에서 확보한 데이터와 정합성 향상 알고리즘을 기반으로 렌더링 프로그램을 구현하고, 이를 실험 차량 창문에 탑재된 증강 현실 디스플레이에 투영한다. 선택한 경로를 차량으로 주행하여 이동하는 환경에서 시각적으로 어떻게 구현되었는지 검증하고, 부자연스러운 부분이 있다면 앞서 구현한 알고리즘을 휴리스틱하게 개선한다.



Figure 14. 증강 현실 디스플레이

4. 실험 결과 및 역할 분담

1) 실험 결과 및 기존 기술과의 비교

RTK 장비를 이용해서 사용자의 위도와 경도를 cm 단위의 오차 수준으로 획득할 수 있었다. 획득한 데이터를 통해서 건물 모델링도 의도한 위치에 맞게 생성되었다는 것을 확인하였다. 그러나 사용한 RTK 장비의 고도 정밀도는 위도와 경도에 비해서는 많이 떨어지는 수준이었고, outlier 값으로 인해서 위치와 회전값이 간헐적으로 튀는 현상을 발견할 수 있었다. 상기한 이유들로 인해서 하나의 오브젝트를 기준으로 시야각, 위치, 방향을 증강 현실 디스플레이와 일치하도록 교정해주는 작업은 진행할 수 없었다.

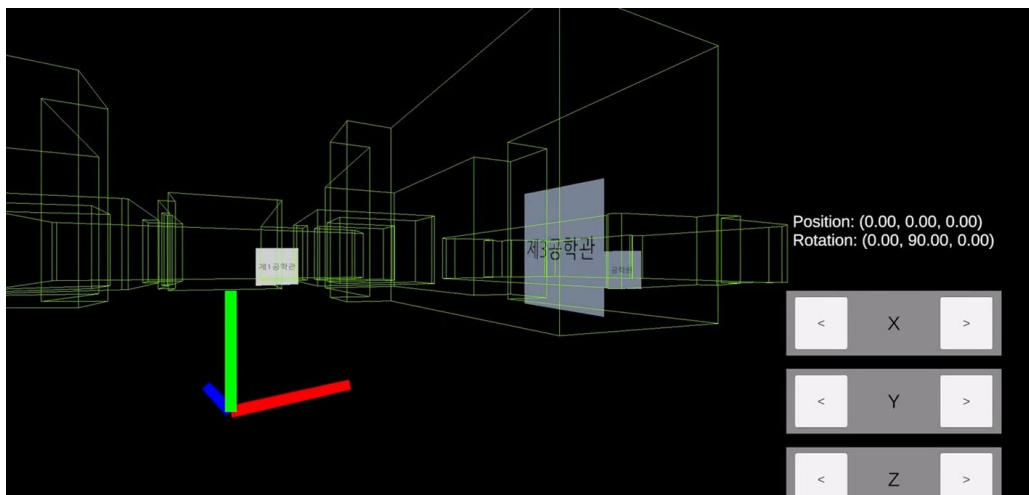


Figure 15. 실험 진행 클라이언트 스크린샷

비록 증강 현실 디스플레이와 교정하는 작업은 진행되지 않았지만, 사용자의 위치에 따라서 건물 상의 오브젝트가 정상적으로 움직이는 점과 모델링된 건물들의 위치가 현실에서의 위치와 일치한다는 점을 통해서 조금 더 정확한 위치를 확보할 수 있다면, 우리의 구현이 실제로 동작하며 유의미한 성능을 발휘할 수 있음을 확인할 수 있었다.

건물 모델링을 사용함으로써 성능적으로 이득을 볼 수 있는 부분도 있었다. 현재 상용화된 AR 엔진인 Vuforia를 Android에서 구동하기 위한 최저 API level이 26인 반면, 우리의 방식에서는 Unity 클라이언트 구동을 위한 최저 API level인 19에서도 구동 가능한 것을 확인할 수 있었다. 또한 기존의 방식과는 달리 이미지 처리가 이루어지지 않고 위치 좌표만 주어진다면 렌더링이 가능하기 때문에 갤럭시 S6 등 낮은 성능의 단말기에서도 원활히 동작하는 모습을 확인할 수 있었다.

2) 개선 사항

위치 정보의 outlier 값을 정제하기 위하여 역치값을 두어 데이터를 정제하는 방법을 이용하였다. 그러나 짧은 시간에 급격히 변하는 데이터의 경우 예상과 다르게 정제되지 못하는 현상이 있었다.

이러한 문제는 실시간으로 좌표를 이용하는 것이 아닌, 주어진 경로를 한정하고 해당 경로상의 좌표로만 현재 위치를 계산한다면 개선될 것으로 생각된다. 현존하는 내비게이션의 경우에도, GPS의 오차를 고려하여 실제 좌표를 사용하는 것이 아닌 가장 가까운 도로의 위치를 현재 위치로 표시하고 있다. 이러한 원리로 맵 에디터에 추가로 경로 정보 및 현재 위치 좌표로부터 가장 가까운 경로상의 좌표를 계산하는 기능을 추가하였다.



Figure 16 위치 오차 개선을 위한 경로상의 좌표 지정

에디터 상의 파란색 얇은 선은 우리의 실험 경로를 모사한 것이며, 그림 상부 과학원 우측의 굵은 선은 특정 위치로부터 경로상에 가장 가까운 좌표를 계산하여 그림으로 나타낸 것이다. 특정 위치를 지정하면 해당 위치로부터 가장 가까운 경로상의 좌표가 자동으로 계산, 표시된다. 이를 우리가 개발한 애플리케이션에 적용하면 outlier 데이터에 의해 렌더링이 급격히 변하는 현상을 방지할 수 있을 것이다.

그러나 이러한 알고리즘을 적용하면 항상 지정된 경로만을 이용해야한다는 단점이 있다. 우리의 연구는 차량 뿐만이 아닌 다양한 모빌리티에서도 이용하는 것이 목표였던 만큼, 장기적으로는 경로를 지정하지 않더라도 RTK 데이터 자체를 보정할 수 있는 알고리즘을 개발해야 할 것이다.

3) 팀의 구성 및 팀원의 역할

- 황보정 : 오브젝트 배치 알고리즘 구현 및 프로토타입 데모 프로그램 제작
- 도회린 : RTK 데이터 처리 서버 및 로직 구현, 데모 프로그램 제작 보조
- 최지웅 : RTK 데이터 수집 및 데이터 후처리