

3D 지도와 결합된 실시간 증강현실 건물 안내 시스템의 설계 및 구현

김상준⁰

배운민¹

최유주^{0,2}

⁰서울미디어대학원대학교 뉴미디어학부 ¹세종대학교 만화애니메이션학과

²서울미디어대학원대학교 실감미디어연구소

gogo5911@naver.com , ymy4276@naver.com , yjchoi@smit.ac.kr

Design and Implementation of Real-time Augmented Reality Building Information System Combined with 3D Map

Sang-Joon Kim⁰

Yoon-Min Bae¹

Yoo-Joo Choi^{0,2}

⁰Dept. of New Media, Seoul Media Institute of Technology ¹Dept. of Cartoon Animation, Sejong University

²Immersive Media Lab. Seoul Media Institute of Technology

요 약

최근 스마트폰을 이용한 증강현실 기반 건물 안내 어플리케이션들은 사용자의 위치와 목표 건물의 위치 간의 거리에 상관없이 동일한 고정 형태로 정보를 보여주고 있고, 특히 다수의 목표 건물들이 근접하여 위치하고 있을 때 제공 정보들이 중첩하여 디스플레이 됨에 따라 정보 전달력이 떨어지는 제한점이 있다. 또한, 기존 증강현실 기반 어플리케이션에서 사용자의 현재 위치를 직관적으로 파악하는 데는 어려움이 있다. 본 논문에서는 이러한 제한점을 해결하고자 사용자와 건물의 위치에 따라 정보객체의 위치와 크기가 적응적으로 제공되고, 사용자의 위치가 실시간으로 표시되는 3D 지도를 제공함으로써 사용자들이 직관적으로 자신의 위치와 건물의 위치를 파악할 수 있도록 하는 안내 시스템을 설계 구현하였다. 본 논문에서 제시한 증강현실 기반 건물 안내 시스템은 서울시 중구에 위치한 덕수궁내의 건물 안내에 초점을 맞추어 제작되었다.

Abstract

Recently, augmented reality(AR) based building information applications using a smart phone provide information in the static form irrespective of the distance between a user and a target building. If many target buildings are located close to each other, discrimination of information is reduced due to overlapping information objects. Furthermore, it is difficult to intuitively grasp the current position of the user in the previous AR-based applications. In this paper, to solve these limitations, we have designed and implemented a novel building information system in which the location and size of information objects are adaptively displayed according to locations of a user and target buildings, and which allows users to intuitively understand their location by providing a 3D map that displays the user's location and all target buildings within a given distance in real-time. The AR-based building information application proposed in this paper focuses on the building guide in Deoksu Palace in Jung-gu, Seoul.

키워드: 증강현실, 위치기반서비스, 모바일 앱, 유니티

Keywords: Augmented Reality, Location-Based Service, Mobile Application, Unity, Information System

*corresponding author: Yoo-Joo Choi/Immersive Media Lab. Seoul Media Institute of Technology(yjchoi@smit.ac.kr)

Received : 2018.08.23./ Review completed : 1st 2018.08.31. / Accepted : 2018.08.31.

DOI : 10.15701/kcgs.2018.24.4.39

ISSN : 1975-7883(Print)/2383-529X(Online)

1. 서론

스마트폰의 보급과 증강현실(augmented reality)의 발전으로 모바일 증강현실 기술이 다양한 분야에 적용되고 있다[1,2,3]. 그 중 모바일 위치기반서비스와 증강현실을 결합한 어플리케이션[4,5,6]이 소개되어 사용되고 있으며 이들 어플리케이션은 두 가지 형태를 띠고 있다. 첫 번째는 실외 건물 혹은 목표지점의 위치 정보를 제공하는 안내 어플리케이션으로 대표적으로 2018년 평창에서 선보인 AR ways[5]가 있다. AR ways는 올림픽 방문객의 공항입국부터 올림픽 경기관람, 출국까지의 이동시 증강현실 화면으로 실감나는 길 안내 서비스를 제공하는 어플리케이션이다. 두 번째는 특정한 위치에 가상의 객체(Object)를 정합하여 보여줌으로써 사용자와 상호작용하는 어플리케이션으로 대표적으로 포켓몬고(Pokemon Go)[6]가 있다. 포켓몬고는 현실 영상에 몬스터가 나타나고 사용자가 터치 상호작용을 이용하여 포켓몬을 잡을 수 있는 구조로 개발된 증강현실 게임이다. 본 논문은 모바일 위치 기반 서비스의 첫 번째 유형인 정보제공 어플리케이션에 대한 사용 편리성 개선을 목표로 하고 있다.



Figure 1. AR GUI focused on information provision purpose[7]

기존 위치기반 증강현실 어플리케이션은 실외 공간에 위치한 관광명소, 유적지, 문화재, 건물 등의 정보를 이미지 혹은 텍스트로 제공하는 안내판, 가이드북, 팸플릿, 애플리케이션 보다 정보를 쉽고 빠르게 제공해준다. 하지만 제공되는 정보가 Figure 1과 같이 사용자의 현재 위치에서의 방위각 정보에 따라 일정한 크기로 제공되기 때문에 정보가 제공되어야 할 목표 실외 건물이 다수인 경우, 각 실외 건물들에 대응되는 가상의 정보 객체들이 서로 겹쳐져서 디스플레이되어 정보 확인에 어려움이 발생하는 경우가 대부분이다. 또한, 증강현실 화면에서는 현재 위치로부터 목표 건물까지의 거리를 직관적으로 인지하기 어렵다. 이에 목표 건물까지의 위치파악을 위하여 2D 나침반이 제공되고 있지만, 실제 공간과의 매치가 어려워 목표 건물의 위치를 혼동할 수 있고, 몰입감을 제공하기 위한 가시적 요소가 부족하

다.

본 논문에서는 이러한 제한점을 해결하고자 목표 건물 대비 스마트폰 화면 방향, 목표 건물 및 사용자의 위치에 따라 가상 정보 객체의 크기와 위치가 적응적으로 변경되어 제시되고, 사용자의 현재 위치를 보다 쉽게 파악 할 수 있도록 하는 3D 지도 기능을 포함한 건물 안내 시스템을 설계 구현하였다.

2. 관련연구

2.1 증강현실

R. Azuma 정의에 따르면, 증강현실(augmented reality) 시스템이란, 현실(real-world elements)의 이미지와 가상의 이미지를 결합한 것으로 실시간으로 인터랙션(interaction)이 가능하고 3차원 공간 안에 놓인 것이다[8]. 이렇듯 증강현실은 실세계의 이미지나 배경에 3차원 가상 물체를 결합해 하나의 영상으로 보여주는 컴퓨터 그래픽 기법이다. 증강현실은 가상세계를 전제로 하는 가상현실과는 달리 현실 세계의 환경위에 가상의 대상을 결합시켜 현실의 효과를 더욱 증강시켜 몰입감을 준다[9]. 증강현실 콘텐츠는 제공되는 정보의 유형에 따라, 목표 객체 추적 방법에 따라 그리고 증강현실 콘텐츠의 디스플레이 형태에 따라 유형을 분류할 수 있다.

1. 정보의 유형에 따른 분류

증강현실에서는 어떤 정보를 어떻게 제공하느냐에 따라 두 가지로 분류된다. 첫 번째는 시각에 중점을 두고 정보를 제공하는 시각형 증강현실 유형으로 콘텐츠가 주로 그래픽 작업으로 이루어지는 게임, 증강현실 기반의 이러닝(e-learning)시스템 등이 있다. 주로 현실 공간에 기구축된 3D 그래픽 객체를 합성하여 사용자들에게 제공함으로써 몰입감을 높이는 형태이다. 두 번째는 정보 서비스형 증강현실로 그래픽 객체 보다는 현실 객체에 대한 텍스트 기반 혹은 방향 정보등의 전달이 목적인 유형이다. 대표적인 예로 증강현실 네비게이션 및 GPS 정보를 이용한 증강현실 위치기반 서비스 등이 있다. 사용자의 위치에 따라 GUI 정보를 제공하기 때문에 모바일 디바이스의 각종 센서 및 네트워크를 기반하여 콘텐츠가 제작된다.

2. 추적 방법에 따른 분류

증강현실에서는 어떤 대상을 어떤 방식으로 추적하느냐에 따라 크게 세 가지로 분류된다. 첫 번째로는 컴퓨터 비전기반 증강현실로 목표 객체를 컴퓨터 비전 기술을 이용하여 인식하여 객체 혹은 정보를 가시화 시키는 형태이며 실내외에서 적용이 가능하고 몰입감이 높다. 하지만 거대한 객체는 인식이 어렵고 조명에 민감하게 영향을 받는다. Figure 2는 목표 이미지를 인식하여 3D 모델을 증강시킨 사례를 보여주고 있다. 두 번째로는 센서 기반의 증강현실로 GPS, 방

위각 센서등 센서 정보를 기반으로 목표 위치를 인식하는 방법이다. 실외 공간에 위치한 건물, 물체 등에 적용하는 방법으로 넓은 범위를 절대좌표로 추적할 수 있어 개발이 용이하지만, 큰 거리 오차가 있어 실재와 가상 객체의 정교한 정합이 어렵고, 실내 증강현실에 적용이 불가능하다. Figure 3은 GPS 정보 기반 증강현실 게임 사례를 보여 주고 있다. 세 번째로는 하이브리드 기반의 증강현실로 센서 정보와 비전 알고리즘을 동시에 사용하는 방법이다. 비전 기술과 센서 기술의 결합을 통하여 실재와 가상객체간 정교한 정합이 가능하나 개발 난이도가 높다.



Figure 2. Vision-based Augmented Reality[10]



Figure 3. Sensor-based Augmented Reality[6]

3. 디스플레이 형태에 따른 분류

증강현실은 디스플레이의 형태에 따라 크게 네 가지로 분류된다. 첫 번째로는 일반 모니터의 형태이며 개인 컴퓨터 디스플레이, 공공장소 디스플레이 등이 있다. 일반 모니터 기반의 시스템은 데스크탑에 설치된 카메라를 이용하여 사용자 혹은 객체를 추적하여 증강하는 형태이다. 개발이 쉽고 사용자들이 접하기 쉬우나 몰입감이 낮고 이동과 공간이 제한된다. 두 번째로는 HMD기반의 형태이며 머리에 착용하여 증강현실을 경험할 수 있다. 공간 제약이 없고, 헤드트래킹을 이용하여 사용자의 움직임에 따라 디스플레이 시점이 직관적으로 변화됨에 따라 높은 몰입감을 제공한다.

세 번째로는 모바일 디스플레이 형태로 여러 가지 센서 정보와 모바일 카메라 정보를 기반으로 모바일 디바이스의 스크린 상에 실제 객체와 가상 객체의 정합 가시화가 이루어지는 형태이다. 네 번째로는 사전에 설치된 대상 물체(건물 외벽, 인테리어 공간, 오브제 등) 표면에 가상 객체를 증강시키는 프로젝션 기반의 증강현실이다. 일반적으로 프로젝션 기반 공간 증강현실(spatial augmented reality)을 프로젝션 매핑이라 부른다. 다른 형태에 비해 저렴하고 짧은 시간에 화려한 퍼포먼스를 구현할 수 있으며 다수의 사용자가 동시에 증강현실 콘텐츠를 경험할 수 있다. 하지만 관람시점에 따른 왜곡을 고려하여야 하며 피사체에 투사되는 이미지를 다른 공간에 재사용하고자 할 때 그 모양이나 크기가 다르므로 단순 재사용이 어렵다. Figure 4는 대형 건축물 외관에 맞추어 예술적 영상물을 프로젝션 시킨 프로젝션 매핑사례를 보여주고 있다.



Figure 4. Projection mapping[11]

2.2 위치기반 서비스

위치기반 서비스(LBS : Location Based Service)는 고객의 위치정보를 기반으로 상품 정보, 교통 정보등 생활 전반에 필요한 다양한 정보를 제공하여 삶의 질을 향상시키는 서비스를 말한다. 위치기반 서비스는 크게 세 가지로 분류된다. 첫 번째는 RFID(Radio-Frequency Identification) 기반의 서비스로 RFID 태그와 RFID 판독기 사이에 흐르는 전파를 통해 정보를 얻는 기술이다. 휴대폰의 GPS를 이용한 LBS서비스가 해결하지 못하는 지하, 건물 내 등의 장소에서 RFID를 이용하여 LBS 연동을 지원한다. 두 번째는 Cell 기반 서비스로 이동 통신 기지국을 이용한 방식이다. 가입자와 연결된 기지국을 중심으로 500~1500m의 오차범위에서 사용자의 위치를 파악할 수 있다. 이 방식의 단점은 오차범위가 매우 넓은 편이기 때문에 대략적인 위치파악만 가능하다. 그에 반해 중계기 등을 이용하여 건물 내 및 지하 등의 위치도 찾을 수 있으며 단말기에 하드웨어, 소프트웨어의 추가 없이 이용이 가능하다. 세 번째 방법은 GPS기반의 서비스이다. GPS는 27개의 인공위성을 이용하여 위치를 추적하는

방식으로 위치파악 정확성이 높다. 하지만 GPS 수신기가 별도로 필요하며 위성 신호의 특성 때문에 고층건물이나 실내에서는 사용이 불가능하다.

2.3 모바일 센서

대표적인 모바일 디바이스인 스마트폰에는 다양한 종류의 센서가 포함되어 있다. 이러한 센서들은 사용자의 위치 및 스마트폰의 방향등을 파악하는데 사용된다. 대표적인 센서의 종류 및 특성은 다음과 같다.

1. GPS 센서

GPS란 위성에서 보내는 신호를 수신해 사용자의 현재 위치를 계산하는 위성항법시스템이다. 과거 군사용으로 개발한 시스템이지만 현재에는 실생활의 다양한 응용분야에서 사용하는 기술이 되었다[12-14]. GPS는 인공위성의 시간/위치 오차, 전리층과 대류층의 오차, 수신기 잡음, 다중경로, SA(selective availability)이유로 오차가 존재한다[15]. 이런 여러 가지 오차를 보정하기 위한 연구들이 진행되고 있다[15-17].

2. 지자기 센서

지자기 센서(geo-magnetic Sensor)는 지구의 자기장의 흐름을 이용해 나침반처럼 방위각을 탐지하는 센서다. 기존에는 2축 지자기 센서가 주였으나, 지금은 X, Y, Z축 방향의 3축 지자기 센서가 보편화되었다. 지자기 센서는 축의 출력 합으로 자기장 방향을 탐지한다. 지자기 센서는 GPS센서와 결합해 위치기반 서비스에서 자주 사용된다. 하지만 센서에 영향을 미치는 요소들이 다수 존재하여 이를 보정하기 위한 연구들이 진행되었다[18, 19].

3. 자이로스코프 센서

자이로스코프 센서(Gyroscope Sensor)는 가속도 센서에 회전축을 더해 총 6축을 인식할 수 있도록 만든 센서이다. 자이로스코프 센서는 물체의 속도변화나 충격의 세기를 감지하여 측정하는 센서로 가속도 센서보다 훨씬 정밀한 동작을 인식할 수 있다.

2.4 위치기반 정보 서비스의 인터페이스 한계점

스마트폰을 이용한 모바일 증강현실은 대부분 가상 객체에 대한 터치 동작을 통하여 사용자와 인터랙션이 이루어진다. 증강현실 기반 주변 정보를 제공하는 정보 서비스의 경우 인터페이스에 몇 가지 한계점이 있다.

첫 번째 한계점은 객체의 중첩현상으로서 많은 위치 정보를 한 번에 디바이스 화면에 보여주어야 하는 경우 발생한다. 중첩된 객체들은 정보를 식별하기 힘들고 거리를 느낄 수 없게 된다. 이런 중첩 현상을 방지하기 위해 투명도의

차이를 주거나 카테고리, 거리에 따라 객체를 분산하여 보여준다. 하지만 투명도 차이를 주면 객체가 명확하게 판별되지 않고, 객체를 분산 배치시킨다 하더라도 가까운 거리에 많은 객체가 있을 경우 중첩 현상을 피해갈 수는 없다. Figure 5의 왼쪽 그림은 안내되어야 할 많은 정보 객체가 중첩되어 해당 객체를 선택하기 어려운 사례를 보여 주고 있다. 두 번째 한계점은 증강된 정보객체가 매칭되는 실제 객체까지의 거리를 반영하여 가시화 하는 요소가 부족하다는 점이다. 다시 말해 정보객체가 가시화된 것을 보고, 현재 위치에서 목표 위치까지 어느 정도의 거리가 남아 있는지 직관적으로 거리감을 파악하기 어렵다[20]. SLAM(simultaneous localization and mapping)기술[21]등을 통해 실세계에 존재하는 목표 객체 상에 가상의 객체를 정확하게 정합시킬 수 있지만, 이 기술은 계산량이 많고 광범위한 실외 공간에 적용하기 어렵기 때문에 모바일 디바이스를 이용한 광범위한 지역 대상 증강현실 정보 서비스에는 적합하지 않다. Figure 5의 오른쪽 사진과 Figure 6의 사진은 정보 객체의 크기가 거리에 따라 적응적으로 조정되지 않아 목표 위치까지의 거리를 직관적으로 파악하기 어려운 사례를 보여주고 있다.

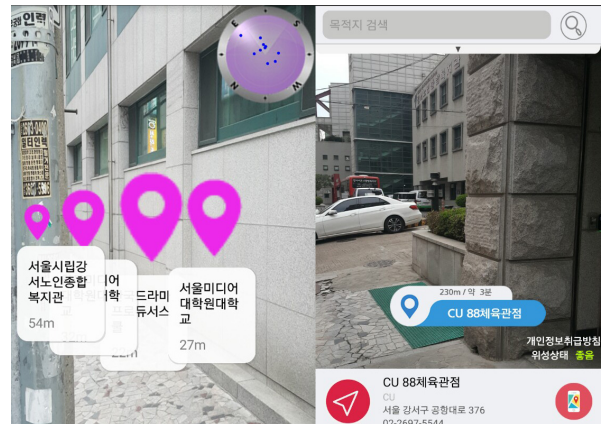


Figure 5. Overlapping virtual objects to be visualized without considering the distance factor[22,4]



Figure 6. Virtual objects displayed in ambiguous locations[23]

3. GPS 기반 실시간 증강현실 건물 안내 시스템의 설계

3.1 제안된 시스템 개요

제안하는 건물 안내 시스템은 실외 건물의 정보와 위치를 직관적으로 제공하여 실외 건물을 쉽고 빠르게 찾아 갈 수 있도록 제공하는 것으로 시스템의 Figure 7에서 보여주는 것과 같이 위치 건물안내 데이터베이스, 디바이스 센서 확인 및 보정, 2D 나침반, 위치기반 증강현실 정보 가시화, 3D 지도 모듈로 구성하였다. 위치 기반 건물 안내 데이터베이스 모듈은 건물들의 위치정보, 정보 이미지 안내 데이터베이스 모듈은 건물들의 위치정보, 정보 이미지 및 텍스트, 가상 3D 모델 등 증강현실 정보 가시화를 위해 필요한 데이터를 데이터베이스에서 검색하여 디바이스로 받아 오는 모듈이다. 디바이스 센서 확인 및 보정 모듈은 디바이스 센서의 유, 무와 오류 여부를 판단하고 지자기 센서와 자이로 센서 정보를 보정하는 모듈이다. 2D 나침반 모듈은 데이터베이스에서 받은 건물의 GPS정보와 실시간 측정 GPS 값을 이용하여 나침반 GUI 상에 목표 객체들을 표시하는 역할을 담당한다. 위치 기반 증강현실 정보 가시화 모듈은 데이터베이스에서 받은 건물의 GPS 정보, 이미지, 텍스트, 가상 객체 정보, 실시간 GPS정보를 이용하여 3D 공간의 위치 값과 매칭 하여 가상 객체를 나타내는 역할과 가상 객체를 터치하였을 때 건물의 정보를 GUI를 통해 보여주는 역할을 한다. 3D 지도 구성 및 가시화 모듈은 실시간 GPS 정보를 계산하여 미리 만들어진 3D 지도에 사용자의 위치를 실시간으로 보여주는 역할을 한다.

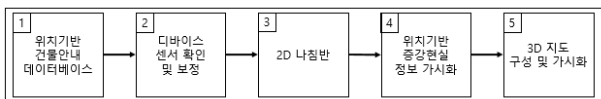


Figure 7. System configuration modules

3.2 제안 시스템 사용절차

제안 시스템의 사용절차는 Figure 8과 같은 순서로 이루어진다. 첫 번째로 증강현실 정보 제공이 지원되는 여러 지역 중에 안내를 받고 싶은 지역을 메인 메뉴에서 선택한다. 한 개의 관심지역을 선택하고 두 번째로 GPS를 이용하여 사용자의 위도 경도 정보를 받아온다. 세 번째로 디바이스의 필요한 센서가 있는지와 1단계에서 선택한 관심 지역 내에 있는지를 검사한다. 만약 센서가 없고 관심 지역 내에 있지 않다면 사용자는 애플리케이션을 사용할 수 없으므로 서비스가 가능한 다른 지역을 선택하도록 유도한다.

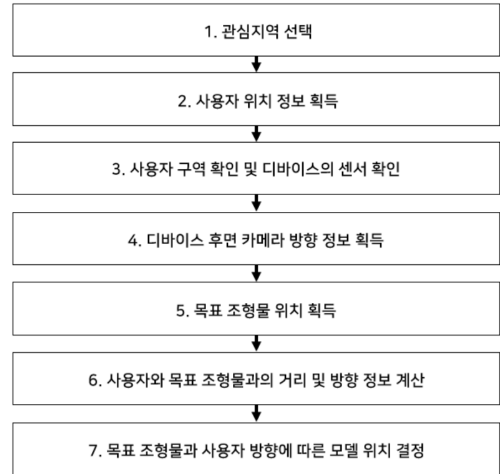


Figure 8. System workflow

네 번째로 자이로스코프와 지자기 센서를 이용하여 현재 사용자의 위치 및 디바이스 후면 카메라가 향하는 방향 정보를 계산하여 획득한다. 다섯 번째로 목표 건물들의 경도, 위도 값을 데이터베이스로부터 읽어 들인다. 여섯 번째로 사용자의 현재 위치로부터 각 목표 건물까지의 거리와 상태적인 방향을 계산한다. 일곱 번째로 목표 건물을 표현하는 3D 모델의 위치를 결정하여 카메라 화면상에 보여준다. 이때, 목표물까지의 거리와 방향에 따라 적응적으로 3D 모델의 크기 및 방향이 변경되도록 한다. 이렇게 배치된 모델은 사용자의 위치에 따라 3D 모델의 크기가 원근감 있게 변경되므로 직관적으로 목표 건물까지의 거리를 쉽게 파악할 수 있고, 2D 나침반 GUI 상에 표현된 사용자의 위치와 증강현실로 보여지는 모델의 가시화 위치에 따라 목표물로 가는 방향을 쉽게 인지할 수 있다. 또한 3D 지도에 사용자의 위치가 표시되어 목표물까지 가는 길을 한눈에 파악할 수 있다.

3.3 데이터베이스 구성

건물 안내 시스템을 위한 데이터베이스에는 관광지 선택을 위한 관광지 정보(tourist spot information) 테이블과 관광지 안의 건물들의 정보가 들어 있는 건물 정보(building information) 테이블로 구성된다. 관광지 정보 테이블과 건물 정보 테이블의 세부 필드는 Table 1과 Table 2와 같다.

Table1. Configuration of the tourist spot table

Field name	Description
Spot_No	Unique number of the spot
Spot_Name	Name of the spot
Spot_Image	Representative image for tourist spot
Latitude_From	Starting latitude for the tourist spot area
Latitude_To	Ending latitude for the tourist spot area
Longitude_From	Starting longitude for the tourist spot area
Longitude_To	Ending longitude for the tourist spot area

Table2. Configuration of the building information table

Field name	Description
Spot_No	Unique number of the spot
Building_No	Unique number of the building
Building_Name	Name of the building
Building_Latitude	Latitude of the building
Building_Longitude	Longitude of the building
Building_Image	The image to represent the building in the 2D compass
Building_Material	Virtual indicator object for the building
Building_Modeling	3D model to represent the building
Building_Infor_Name	Title image of the building to be used in the information window
Building_Infor_Image	Description image for the building to be used in the information window
Building_Infor_Movie	Description video for the building

3.4 디바이스 센서 확인 및 보정

제안 시스템은 대표적 게임 엔진인 유니티(Unity)를 이용하여 구현되었다. 센서의 유무 확인 및 센서값 보정기능은 아래와 같이 유니티의 센서 입력(input) 클래스를 이용하여 구현하였다.

1. 디바이스의 센서 확인

유니티에서는 모바일 디바이스의 경우 입력 클래스를 통해 접근할 수 있다. 지자기 센서를 기반으로 방위각 정보를 전달해 주는 콤팩스 센서(Compass Sensor)는 Input.compass[Variables]를 이용하여 접근할 수 있고 자이로 센서는 Input.gyro[Variables]를 이용하여 접근할 수 있으며 GPS 센서는 Input.Location[Variables]를 이용할 수 있다. 이렇게 접근한 각각의 센서 정보가 값을 반환하는지를 체크

하여 센서의 유, 무를 확인한다.

2. 디바이스의 후면 카메라 방향 보정

사용자의 디바이스 후면 카메라 방향 정보는 유니티의 콤팩스 센서와 자이로 센서를 이용한다. Figure 9는 디바이스 자세에 따라 변환된 콤팩스 방향을 보여주고 있다. 좌측 사진은 디바이스를 세로로 잡았을 경우이고 우측사진은 디바이스를 가로로 잡았을 경우이다. 여기서 빨간색 화살표는 콤팩스의 X축 방향, 초록색 화살표는 Y축 방향, 파란색 화살표는 Z축 방향을 나타내는데 디바이스의 자세 변화로 콤팩스 센서의 축 방향이 변경된다. 이로 인해 센서의 값은 정상적인 값이 아닌 축 변환이 이루어진 값이 들어오게 된다. 그렇기 때문에 방향에 따라 콤팩스 센서로부터 받은 X, Y, Z 값을 보정하고 사용한다.

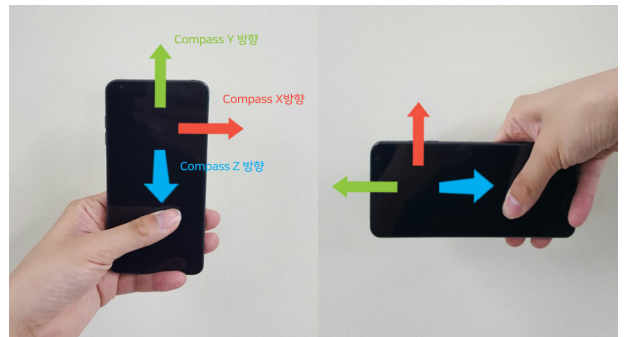


Figure 9. Compass orientation according to device pose

3.5 2D 나침반

2D 나침반은 사용자 주변에 등록된 실외 건물이 있을 경우 원형의 나침반 GUI에 목표 건물 객체들과 사용자를 나타내는 표식(indicator)을 가시화 시켜 사용자가 직관적으로 목표 건물들의 위치를 알 수 있도록 하는 역할을 한다. 2D 나침반은 사용자 GPS 위치와 실외 건물의 GPS 위치를 이용하여 사용자와 건물의 실제 거리를 계산하고, 이를 원형 나침반 상의 상대적인 거리로 변환하여 표시한다. 일반적인 네비게이션이 표현하는 것과 같이 사용자의 위치는 항상 원형 나침반의 중앙에 위치하도록 하고, 방향은 왼쪽(전진방향)을 향하도록 한다. 사용자가 좌우 위치를 바꾸는 경우, 2D 나침반이 회전하여 가운데 위를 향하는 사용자 표식은 그대로 있고, 목표 건물의 위치가 상대적으로 변경하도록 함으로써 스마트폰을 손에 들고 정면을 향하여 걸으면서 사용자의 위치 중심에서 진행 방향을 인지하기에 편리하도록 구성한다. Figure 10은 제안하는 2D 나침반 화면을 보여주고 있다. 가운데 삼각형이 사용자의 위치와 방향을 표시하는 표식이고, 원형 숫자들이 목표 건물들을 표현하고 있다. 이때 목표 건물들의 번호는 Figure 11에서 확인할 수 있다. Figure 11은 2D 나침반과 3D 지도와는 다른 일반 이미지

형태로 사용자가 실외 건물들의 위치와 번호를 확인 하는 정보 이미지이다.

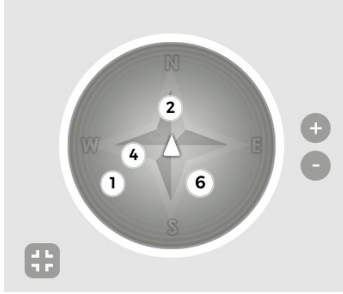


Figure 10. 2D Compass GUI



Figure 11. 2D map image

2D 나침반 상에 목표 건물 객체와 사용자 표식을 가시화 하는 세부 방법은 아래와 같다.

1. 구에서 두 점 사이 거리 구하기

구에서 사용자와 실외 건물의 거리를 계산하기 위해서는 사용자의 위도/경도값과 실외 건물의 위도/경도값을 받아온 후 식(1)을 통하여 사용자와 실외 건물간의 거리를 구한다.

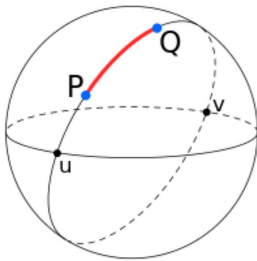


Figure 12. Distance between two points on a sphere[24]

$$DISTANCE1 = \arccos[\sin(LAT_1) \times \sin(LAT_2) + \cos(LAT_1) \times \cos(LAT_2) \times \cos(LONG_2 - LONG_1)] \quad (1)$$

식(1)에서 DISTANCE1은 구에서 두 점 사이의 거리(미터단위)를 의미하고 LAT1, LONG1, LAT2, LONG2는 사용자의 위치(위도, 경도), 실외 건물의 위치(위도, 경도)를 나타낸다.

2. 나침반 객체 가시화

실제 목표 건물과 대응되는 2D 나침반의 원형 숫자를 가시화시키기 위해서는 먼저 나침반 영역에 대응되는 실제계의 영역을 정의해 주어야 한다. 즉 Figure 13에서 보여지는 2D 원형 나침반의 반지름(S_r)에 해당하는 실제 거리(R_r)를 정의해 주어야 한다. 실제 거리는 미터단위로 표현된다. 제안 시스템의 인터페이스에서 2D 나침반의 크기 즉, S_r 은 고정하여 사용하고, 여기에 매핑되는 실제 공간의 크기 R_r 은 목표 건축물의 밀집도에 따라서 나침반 상의 +, - 버튼을 이용하여 조정할 수 있다. 밀집도가 높은 경우, R_r 의 값을 작게 하여 나침반 상에 목표 건축물의 표식(원 번호)의 겹침을 줄인다. Figure 13에서 사용자 위치와 목표 건물 A 사이의 실제 거리 d 를 기반으로 나침반 상의 상대적인 거리 O_b 는 식(2)를 사용하여 구한다.

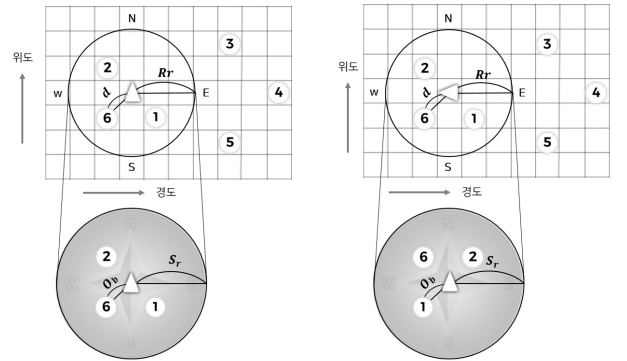


Figure 13. Matching between the distance in the real world and the relative distance on 2D compass

$$O_b = \frac{d^* S_r}{R_r} \quad (2)$$

식(2)에 사용된 값을 다시 정리하여 설명하면 O_b 는 2D 나침반 상에서의 목표 건물과 사용자간 상대적 거리, d 는 현재 위치와 목표 건물 사이의 실제 거리 값, R_r 은 실제 영역에서 나침반의 반지름에 매핑되는 실제 거리, S_r 은 2D 나

침반의 반지름을 의미한다. 여기서 R_r 은 나침반 상의 + 버튼과 -버튼을 이용하여 사용자에게 의해서 정의된다. R_r 의 값이 커지면 보다 넓은 실제 영역이 동일 크기의 나침반 상에 매핑이 되어 목표 건축물의 표식(원 번호)의 겹침이 커지고, R_r 의 값이 작아지면, 실제의 작은 영역이 나침반 영역에 매핑됨으로 건축물의 겹침이 줄어든다. R_r 을 이용하여 나침반에 보이는 목표 건축물의 밀집도를 조정할 수 있다.

2D 나침반에서 사용자의 현재 위치가 나침반의 중심에 위치하고, 사용자가 손에 들고 있는 스마트폰의 후면 카메라의 방향이 항상 나침반의 위쪽을 향하게 가시화 한다. 이를 위해서 사용자의 현재 위치에서 후면 카메라의 방향과 실제 지구의 북점을 향하는 진북방향과의 각도 α 를 계산하고 계산된 각도 α 만큼 동서남북을 표시한 나침반 이미지와 상대적인 거리로 조정된 목표 건축물의 위치 표시 이미지를 회전하여 가시화함으로써 사용자 위치가 중심이 된 2D 나침반을 구현한다. 이때, 후면 카메라의 방향은 자이로 센서와 콤팩스 센서의 정보를 받아 인식한다. 사용자 위치, 후면 카메라 방향, 진북의 방향, 나침반 회전을 위한 각도 α 의 관계는 Figure 14에 표현되어 있다. 회전각도 α 는 [25]에서 제시한 방법에 따라 계산한다.

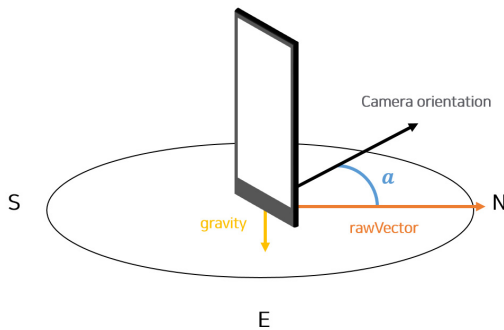


Figure 14. Relation between camera orientation and north direction of compass

[25]에서 제시한 회전각을 계산하는 방법은 동서남북의 정확한 값을 찾아 낼 수 없고 중형이 반대인 오류를 해결하기 위하여 자이로센서의 값을 보정하는 방법이다. 콤팩스 센서를 추가하여 동서남북의 값을 찾고 중형 오류를 수정하고 있다. [25]의 방법에서 α 의 값을 찾기 위해서는 자이로 센서의 자세를 받아와 스마트폰의 자세에 따라 보정한다. 그리고 콤팩스 센서에 새로운 데이터가 들어 왔는지의 여부를 체크한다. 새로운 데이터가 들어 왔다면 자이로센서 정보 Input.gyro.gravity와 콤팩스 센서 정보 Input.compass.rawVector의 값을 가져온다. 콤팩스 센서에서 실제 방위를 알기 위해서는 Input.compass.magneticHeadin g(자력 센서가

가리키는 북쪽(자북)에서 각도), Input.compass.trueHeading(지구의 북극의 방향의 각도)을 사용한다. 하지만 Heading 방법을 사용하기 위해서는 GPS를 이용해야 하는데 Heading의 경우 GPS가 자동으로 내장되어 값을 계산한다. 하지만 그 값이 올바른 값의 여부인지를 보장 할 수 없고 자체의 오차도 있으므로 Heading 대신 rawVector(자북의 방향 벡터)를 사용하여 방향을 잡아낸다. 그리고 방향을 스마트폰 자세에 따라 다시 보정한다. 이런 작업을 거치면 유니티의 세계의 z 방향이 남쪽을 향하게 된다. 그렇기 때문에 편의를 위해 Quaternion.Euler(0, 0, 180)값을 곱해줘 +z 방향을 북쪽으로 변경 해준다. 그리고 rawVector는 가끔 큰 오류 값을 가져올 수 있기 때문에 검사 절차를 거친다. rawVector가 오류를 일으키는 시점은 스마트폰이 시작한 직후이며 오류 값을 가져와 계산할 경우 회전이 NaN이 되고 아무것도 표시되지 않는다. 오류 검사까지 완료되면 계산된 값을 나침반 회전에 필요로 하는 α 로 사용한다.

3.6 위치기반 증강현실 정보 가시화

증강현실 정보 가시화는 먼저 사용자가 실의 건물을 직관적으로 알아 볼 수 있도록 실제 건물과 사용자 위치간 거리, 사용자의 스마트폰 카메라가 향하는 방향에 맞게 크기와 방향이 적응적으로 조정되는 3D 가상객체를 카메라 화면에 증강하여 표시해 주는 역할을 한다. 이때, 사용자를 중심으로 주어진 거리 안에 위치한 목표 건물들만이 카메라 화면에 증강되어 표시되고, 건물을 표시하는 3D 가상 객체를 터치 인터랙션으로 선택하였을 때, 선택된 건물에 해당하는 세부 설명 화면이 보여 지도록 설계하였다.

후면 카메라에 비추는 카메라 영상에 사용자 위치를 중심으로 범위 내에 들어 온 목표 건물의 상대적인 위치를 표현하기 위하여 다음과 같은 세부 처리가 필요하다.

1. 3D 가상객체 생성과 배치

실제 실의 건물에 대응되는 가상객체를 가시화시키기 위해서는 3.5절 2D 나침반 제작의 나침반 객체 가시화 섹션과 마찬가지로 가상의 평면에 대응되는 실제영역을 정의해 주어야 한다. 가상의 평면에서 사용자의 현재 위치가 평면의 중심에 위치하며, 상대적인 거리에 있는 가상의 객체들을 생성하여 배치한다. 배치된 가상 오브젝트들은 사용자가 일정한 영역 안으로 들어오기 전까지 비활성화 모드를 유지하며 사용자가 영역 안으로 들어 왔을 경우 활성화되며 후면 카메라 방향에 따라 유니티 3D 공간을 보여주는 객체 카메라의 회전을 변경하여 실제 실의 건물에 증강되도록 한다. Figure 15의 경우 가상의 평면에 위치한 비활성화, 활성화된 객체를 보여주고 있다. 초록색 정육면체 객체는 비활성화된 가상 객체이며 노란색 정육면체 객체는 활성화된 가상 객체이다. 파란색 원은 평면의 중심 즉 사용자의 현재 위치를

나타내며 파란색 화살표는 후면 카메라 방향을 나타낸다.

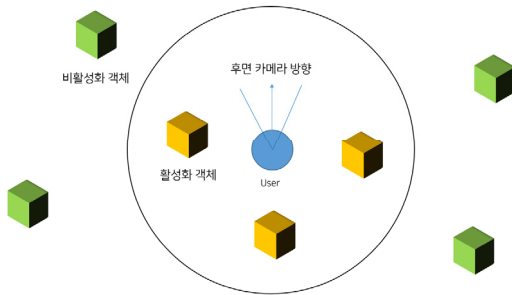


Figure 15. Active and inactive objects on the virtual object plane

2. 실외 건물과 사용자의 위치에 따른 정보 제공

식(1)을 이용하여 실외건물과 사용자간의 거리를 구하게 되면 Figure 16와 같은 방법으로 사용자의 위치와 범위에 따른 정보를 제공할 수 있다. Figure 16에서 r 은 실외 목표 건물을 중심으로 개발자가 미리 정해 놓은 영역을 나타낸다. 사용자와 실외 건물의 거리가 $r_1, r_2, r_3, \dots, r_n$ 영역보다 크다면 사용자는 미리 지정해 놓은 실외 건물 영역 보다 밖에 있는 것을 나타내고 작다면 사용자는 미리 지정해 놓은 실외 건물 영역 안에 있다는 것을 나타낸다. 사용자가 실외 건물 영역 안에 있을 경우 실외 건물의 이름과 몇 번째 영역에 들어와 있는지의 정보를 하단 GUI에 제공하여 보여준다. 이와 같이 사용자와 목표 건물간의 거리를 몇 개의 단계로 구분하여 정보를 제공하는 이유는 거리 계산을 위하여 사용되는 센서의 오차에 의해 가시화 되는 정보의 흔들림 현상을 방지하기 위함이다. Figure 17은 화면 하단에서 현재 영역의 건물이름과 단일 목표 건물 기준 몇 번째 영역에 위치하고 있는지를 보여준 화면이다.

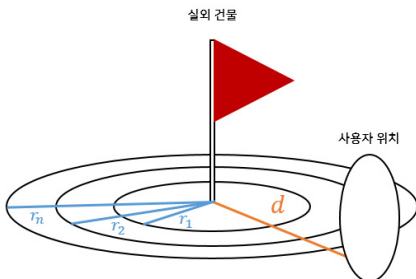


Figure 16. Information area differentiated according to the distance between the user and the target building

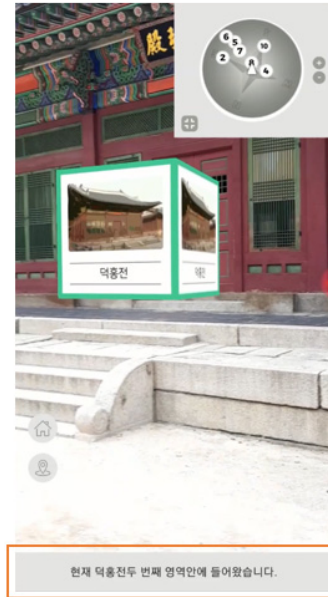


Figure 17. Screenshot showing the information for a current region

3.7 3D 지도

3D 지도는 관심지역 전체를 모델링하고 3D 모델링된 지도에 실시간으로 들어오는 디바이스의 GPS 정보를 이용하여 사용자를 나타내는 객체를 모델링된 지도에 위치시킴으로 사용자가 자신의 위치가 어디인지 직관적으로 알 수 있도록 해주는 역할을 한다.

1. 2D 지도 제작

3D 지도를 만들기 위해서는 밑그림이 되는 실외 건물이 위치한 관심지역의 2D 지도가 필요하다. 2D 지도를 만들기 위해서는 3번의 과정을 거친다. 첫 번째로는 관심지역 위치를 지정하고 구글 지도를 통해 관심지역의 중심 위도, 경도를 찾는다. 두 번째로는 관심지역에 방문하여 구글 지도와 대조하여 누락된 길 또는 건물 구분이 불가능한 화장실 및 기타 건물과 대표적인 자연경관을 찾아 표시한다. 세 번째로는 등록될 건물의 GPS 정보를 획득한다.

2. 3D 지도 모델링

3D 지도를 모델링하기 위해서는 3D 모델링 소프트웨어를 사용해야 한다. 이번 섹션의 작업은 모두 수작업의 단계로 3D 소프트웨어에 관련된 전문인이 필요하다. 3D 모델링 소프트웨어는 Cinema 4D, Maya, 3d Max 등의 소프트웨어를 이용하며 3D 제작이 완료되면 .fbx, .obj 확장자로 익스포트(Export)하여 유니티에 임포트(Import)

한다. 3D 지도를 만들기 위해서는 3단계 과정을 거친다. 첫 번째는 2D 지도를 3D 소프트웨어에 임포트한다. 두 번째는 2D 지도를 기반으로 3D 모델 객체(길, 나무, 주요 건물 등)를 생성한다. 세 번째는 2D 지도상에 3D 모델 객체를 위치시킨다. 본 논문에서 사용된 3D 모델 객체는 Maya 소프트웨어를 사용하여 제작되었다. 목표 건물은 원칙상 3D 스캔 과정을 통해 정확한 3D 모델 정보를 획득하여야 하나 본 연구에서는 실제 건축물 모양과 유사한 모양으로 수작업을 통하여 3D 모델을 제작하였다. Figure 18은 덕수궁의 주요 건물 중 하나인 중화문을 모델링하고 건물 외관의 질감과 색을 입히는 작업의 모습이다. Figure 19은 완성된 모델들을 2D 지도상에 객체로서 배치시키는 작업의 모습이다.

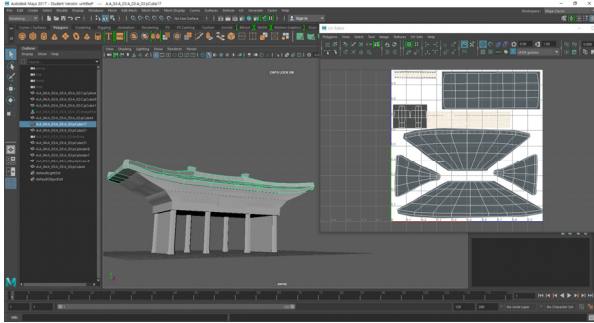


Figure 18. Modeling and texturing a target building

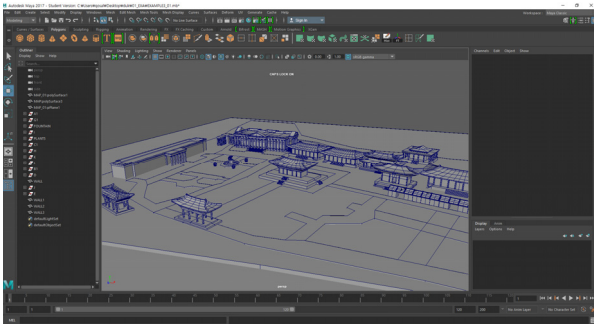


Figure 19. Positioning 3D model objects on the 2D map

3. 3D 지도

유니티에서 3D 지도를 완성하기 위해서는 3D 모델링 소프트웨어에서 제작한 3D 건물 모델링을 유니티에 임포트한다. 그리고 유니티에서 3D 모델링 객체를 위치시킬 가상의 평면(Plane)을 만든다. 그리고 1. 2D 지도 제작 섹션에서 사용한 관심지역 2D 지도를 기반으로 모델링 객체를 배치하고 가상의 평면의 중심을 2D 지도의 중심 위도 경도로 매칭하여 중심점을 기반으로 관심 관광명소의 위도 경도 공간과 3D 맵(map)의 위도 경도 공간을 일치시키고, 사용자의 현재 위치에 따라 입력되는 사용자 위도 경도 위치를 3D

맵 상에 표시한다. Figure 20은 유니티에서 덕수궁 건물 모델링과 자연경관의 모델 객체 배치 작업을 완료한 3D 지도 모습이다. Figure 21은 가상의 평면 중심점을 기반으로 배치된 3D 맵과 사용자의 위도 경도를 이용하여 배치된 사용자 모델의 모습이다.

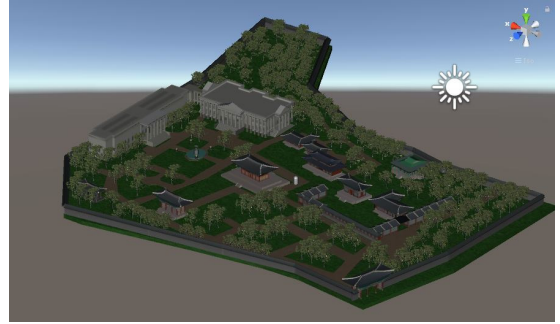


Figure 20. 3D Map for Deoksu Palace

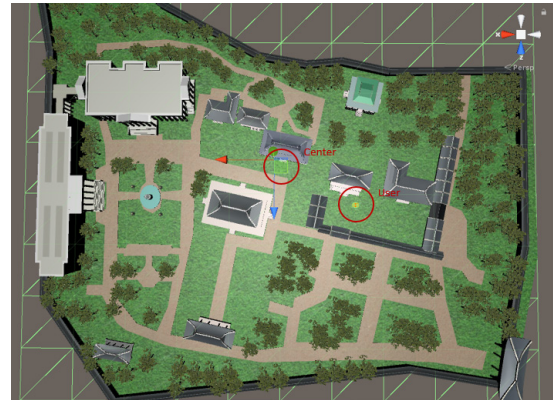


Figure 21. A user model on 3D Map

4. 구현 및 실험 결과

4.1 구현환경

유니티(Unity)는 3D 및 2D 비디오 게임의 개발 환경을 제공하는 프로그래머, 3D 애니메이션과 건축 시각화, 가상 현실 등 인터랙티브 콘텐츠 제작을 위한 통합 저작 도구이다[26]. 유니티를 사용하는 이유는 대표적으로 8가지가 있다. 첫 번째는 렌더링(Rendering)을 지원한다. 렌더링은 모델로부터 영상을 만들어 내는 과정을 말하며 텍스처 매핑(texture mapping), 조명 셰이딩(shading) 정보 등을 지니고 있다. 두 번째는 물리엔진(Physics Engine)을 지원한다. 물리엔진은 현실 세계에서 일어나는 물리적 현상을 컴퓨터로 재현한 것으로 현실감을 제공한다. 세 번째는 사운드 기술이다. 단순히 기본적인 재생, 정지 등을 제어하는 것이 아닌 입체음향과 믹싱효과 등을 지원한다. 네 번째는 애니메이션 기술을 지원한다. 2D 및 3D 애니메이션 기술을 쉽게 로

직으로 제작 할 수 있다. 다섯 번째는 일반적인 디자인 을 한다. GUI는 Compass Controller, Compass , Bulletin

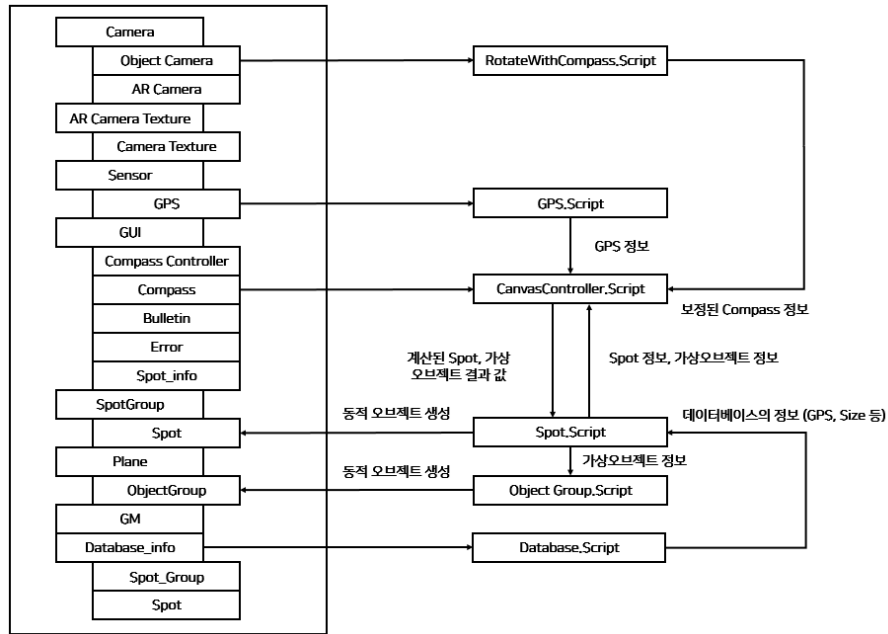


Figure 22. Data flow among scene scripts and objects for the 2D compass

툴과 비슷한 레이어와 같은 개념의 장면 디자인이다. 여섯 번째는 대부분의 메뉴가 복잡한 코딩 없이 사용법 학습과 로직으로 간단한 기능을 구현할 수 있다. 일곱 번째는 에셋 스토어 운영으로 혼자서 개발하기 어려운 부분을 다른 분야의 전문가가 올려놓은 유료, 무료 에셋을 다운받아 사용할 수 있다. 여덟 번째는 멀티 플랫폼 빌드이다. 다양한 운영체제에서 작동 할 수 있는 소프트웨어를 하나의 프로젝트에서 제작이 가능하다.

4.2 장면(Scene) 구성

건물 안내 시스템을 유니티에서 구현하기 위해 (Table 3, Table 4)와 같은 계층구조를 만들었다. Table 3은 2D 나침반의 계층구조로 상위 객체에는 Camera, AR Camera Texture, Sensor, GUI, Spot Group, Plane, GM, Database_info가 있다. Camera 객체는 Object Camera, AR Camera를 하위 객체로 가지고 있으며 Object Camera는 유니티에서 3D 공간에서 만들어진 Object를 보여주는 Camera 객체이고 AR Camera는 디바이스 카메라의 실시간 이미지를 보여주는 기능을 가진 Camera 객체이다. AR Camera Texture는 Camera Texture를 하위 객체로 가지고 있다. Camera Texture는 실시간으로 디바이스 카메라에서 받아오는 이미지를 보여주는 객체이다. Sensor의 하위객체 GPS는 디바이스의 GPS 정보를 실시간으로 받아들이고 보정하고 필요한 객체에 전달하는 역할

, Error , Spot_info 하위 객체를 가지고 있다. Compass Controller는 나침반의 제어 역할을 하는 객체들의 집합이다. Compass는 나침반을 구성하는 객체들의 집합이다.

Table 3. Object hierarchy for 2D Compass

Game Object	Definition
1. Camera	Camera Group
1.1 Object Camera	Unity Object Camera
1.2 AR Camera	Device Camera
2. AR Camera Texture	Texture Group
2.1 Camera Texture	Device Camera Texture
3. Sensor	Sensor Group
3.1 GPS	GPS Sensor
4. GUI	GUI Group
4.1 Compass Controller	Controller Object Group
4.2 Compass	Compass Object Group
4.3 Bulletin	Bulletin Object Group
4.4 Error	Error Object Group
4.5 Spot_info	Spot_info Group
5. Spot Group	Spot Group
5.1 Spot	Spot
6. Plane	Plane
6.1 Object Group	Object Group
7. GM	Game Manager
8. Database_info	Database
8.1Spot Group	Database Spot Group
1) Spot	Database Spot

Bulletin은 상황판을 구성하는 객체들의 집합이다.

Error는 센서정보, 위치정보 등의 오류가 발생할 경우 보여 지는 객체들의 집합이다. Spot_info는 사용자가 실외 건물의 가상객체를 클릭했을 경우 건물의 정보를 보여주는 객체들의 집합이다. Spot_Group은 동적으로 만들어진 Spot 객체를 담고 있는 객체로 하위 객체인 Spot은 데이터베이스에서 받아온 실외 건물들의 정보를 가지고 있다. 평면은 가상의 평평한 바닥 객체로 동적으로 만들어진 가상객체를 담고 있는 Object Group 객체를 하위로 두고 있다. GM은 게임 매니저역할로 Scene의 이동, Touch 유무 등을 관리한다. Database_info 객체는 데이터베이스에서 정보를 받아오는 역할을 하며 하위객체인 Spot Group은 데이터베이스에서 받아오는 정보들을 각각의 실외 건물 별로 나눠 동적으로 Spot 객체를 만드는 역할을 한다.

Table 4. Object hierarchy for 3D map

Game Object	Definition
1. Main Camera	Unity Object Camera
2. Plane	Plane
2.1 Player Object	My Player Object
3. 3D Map	3D Map Object Group
4. My Player	My Player info
5. GPS	GPS Sensor
6. Player Controller	Controller
7. GUI	GUI Group

Table 4는 3D 지도의 계층구조로 상위 객체에는 Main Camera , Plane , 3D Map, My Player, GPS, Player Controller, GUI가 있다. Main Camera는 Table 3의 Object Camera와 같이 유니티 3D 공간에서 만들어진 Object를 보여준다. Plane은 Table 3의 Plane과 같이 가상의 평평한 바닥 객체로 동적으로 만들어진 Player Object를 하위 객체로 가지고 있다. 3D Map은 3D 모델링된 객체들을 담고 있는 객체 그룹이다. My Player는 Player의 정보를 필요한 객체들에게 전달해주는 역할을 한다. GPS는 Table 3과 같이 GPS 정보를 실시간으로 받아들이고 보정하고 필요한 객체에 전달한다. Player Controller는 사용자의 GPS정보를 이용하여 Player Object를 평면의 공간에 배치하는 역할을 한다. GUI는 Main Camera의 이동과 Scene의 이동에 필요한 버튼들을 담고 있는 객체이다.

4.3 스크립트(Script) 구성

센서와 여러 객체들의 상호작용과 가상 객체를 배치하고 움직이기 위해서는 (Table 5 Table 6)과 같은 스크립트가 필요하다. Table 5는 2D 나침반 장면의 스크립트 구성도로 Rotate With Compass , Device Camera , GPS, Compass Controller, Canvas Controller, Spot Group,

Object Group, Database, GM 스크립트들을 구현하였다.

Rotate With Compass Script는 자이로스코프와 지자기 센서를 이용하여 디바이스 후면 카메라 방향을 예측하고 보정하여 필요한 객체에 전달해준다. Device Camera 스크립트는 디바이스의 카메라의 실시간 이미지를 받아와 Camera Texture 객체에 넘겨주는 역할을 한다. GPS 스크립트는 GPS 정보를 실시간으로 받아와 보정하고 필요한 객체에 전달하는 역할을 한다. Compass Controller 스크립트는 나침반의 펼침, 접기 기능과 레이더의 반경을 제어하는 객체에 접근하여 사용자의 행동에 따라 변경해주는 역할을 한다. Canvas Controller Script는 GPS 정보, 보정된 Compass 정보, Spot정보와 가상 객체 정보를 받아 식1, 식2을 이용하여 계산하고 Spot과 가상객체의 위치정보와 가시화 여부를 전달해주는 역할을 한다. Spot Script는 데이터베이스에서 받아온 실외 건물들의 정보들을 Canvas Controller Script에 전달하는 역할과 계산된 결과 값을 전달 받아 가상 오브젝트정보를 Object Group에 전달하는 역할을 한다. Object Group Script는 Spot Script에서 전달 받은 정보를 이용하여 가상객체를 동적으로 생성하는 역할을 한다. Database_info는 데이터베이스에서 정보를 받아오는 역할을 하며 Spot Group 객체 하위에 각각의 실외 건물 별로 객체를 동적으로 만드는 역할을 한다. GM 스크립트는 게임 매니저역할로 Scene의 이동, Touch 유무 등을 관리한다.

Table 5. Scripts for the 2D compass

Game Object	Script
Object Camera	Rotate With Compass
Camera Texture	Device Camera
GPS	GPS
Compass Controller	Compass Controller
Compass	Canvas Controller
Spot	Spot
Object Group	Object Group
Database_info	Database
GM	GM

Table 6은 3D 지도 Scene의 Script 구성도로 Camera Controller , Player Controller, GPS, My Player, GUI 스크립트로 구성되어 있다. Camera Controller 스크립트는 GUI 스크립트로부터 각각의 버튼 클릭의 유무를 전달 받아 Camera의 위치와 확대 감소시키는 역할을 한다. Player Controller 스크립트는 GPS정보를 받아와 식1, 식2을 이용하여 My Player Script에 전달해주는 역할을 한다. My Player 스크립트는 Player Controller 스크립트에서 지정된 영역에서의 계산된 거리 값을 받아 Player Object의 위치 값을 지정해줘 실시간으로 이동하게 하는 역할을 한다. GUI 스크립트는 Camera의 위치를 조정하는 버튼들의 클릭 유무를 확인하여 Camera Controller

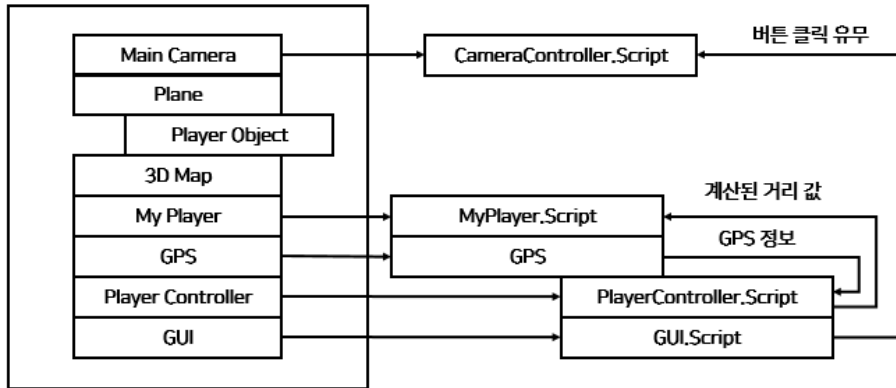


Fig 23. Data flow among scene scripts and objects for the 3D map

스크립트에 전달 해주는 역할을 한다. Figure 22과 Figure 23는 각각 2D compass와 3D 지도를 위한 스크립트와 게임 객체간 데이터 흐름을 표현하고 있다.

Table 6. Scripts for the 3D map

Game Object	Script
Main Camera	Camera Controller
Player Controller	Player Controller
GPS	GPS
My Player	My Player
GUI	GUI

4.4 구현화면

본 논문에서 구현한 건물안내시스템은 유니티5.6.2f1 (64bit)를 이용하여 구현하였고, 갤럭시 노트8 스마트폰을 이용하여 테스트하였다. Figure 24은 제안 시스템의 시작화면으로 관심 관광명소를 선택하는 화면이고, Figure 25는 사용자가 건물안내 시스템을 이용하여 건물을 찾는 과정의 이미지이다. Figure 26은 사용자의 위치가 3D 지도에 실시간으로 보여지는 이미지이다. 사용자는 3D 지도를 통해 자신의 위치를 정확히 파악할 수 있다. Figure 27는 사용자의 위치와 목표건물까지의 거리에 따라 서로 다르게 가시화 되는 2D 나침반과 3D 모델을 보여주고 있다. 가상객체는 개발자가 설정한 건물의 주변에 사용자가 위치했을 때 나타난다. Figure 28은 3D 가상객체가 사용자의 위치와 건물간 거리에 따라 크기가 변화하는 것을 보여준다. Figure 29은 사용자가 가상객체를 터치하여 선택하였을 때 해당 건물의 상세 정보를 보여주는 모습이다.

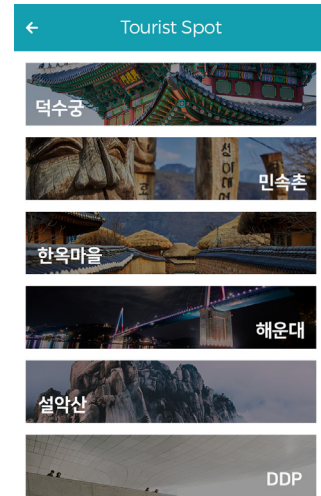


Figure 24. The start page of the proposed system



Figure 25. Finding a building using the building guide system

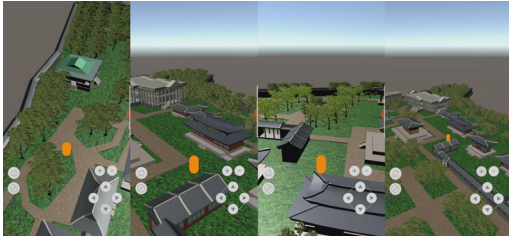


Figure 26. 3D map with the user indicator



Figure 27. 3D virtual object and 2D compass changed according to the user's orientation and the distance between the user and a target building



Figure 28. The size of a virtual 3D object changed by distance

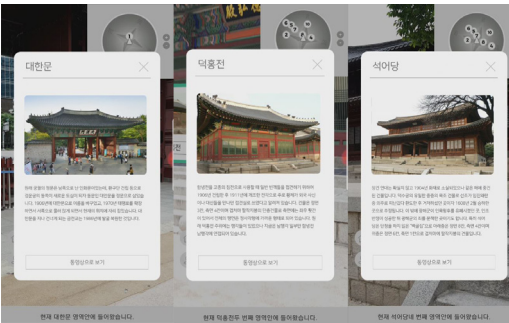


Figure 29. Detail information window activated by touch interaction to a 3D virtual object

5 결론

위치기반 서비스의 증강현실 어플리케이션의 등장으로

실외 공간에 위치한 건물 등의 정보를 이미지 혹은 텍스트로 제공하는 안내수단 보다 직관적이고 쉽고 빠르게 정보를 습득할 수 있게 되었다. 하지만 정보제공을 목적으로 초점을 맞춰 실외의 건물을 가리키는 가상의 객체가 여러 개일 경우 겹침 현상이 나타나서 원하는 목표 건물에 대한 세부 정보를 보기 위하여 해당 건물을 선택하는 것에 어려움이 발생하곤 한다. 또한, 증강현실 정보 가시화 화면에서는 현재 사용자 위치로부터 목표 건물까지의 거리를 직관적으로 예측하는 데 어려움이 있다. 사용자의 편의성과 직관성을 위해 2D 나침반을 제공하지만 실제 공간과의 매치가 어려워 사용자의 위치와 건물의 위치를 혼동할 수 있으며 화면상의 몰입감을 주기 위한 가시적 요소가 부족하다.

본 논문에서는 이러한 제한점을 해결하고자 스마트폰의 후면 카메라 방향, 목표 건물의 위치, 방향, 사용자의 위치에 따라 3D 가상 모델의 크기와 위치가 변경되어 제시되고, 사용자가 주변의 어느 영역에 있는지 쉽게 파악할 수 있도록 위치에 따라 반응하는 지표를 3D 지도상에 제시하였다. 이를 반영한 증강현실 기반 건물 안내 시스템을 설계 구현하였다. 향후 3D 지도에 네비게이션 기능을 추가하여 사용자가 쉽고 편하게 위치를 찾아가 수 있도록 하며 3D 가상객체의 흔들림을 최소화하고 사용자가 실외 건물에 가까이 갔을 때 3D 가상객체가 카메라 화면 전체를 차지하는 문제를 보완할 예정이다.

감사의 글

본 연구는 한국연구재단 이공학개인지초연구지원사업 (NRF-2017R1D11A1B03035718)에 의하여 수행됨

References

- [1] Eunha Park, jinwoo jeon, Developing Korean Learning Contents Using Augmented Reality, The Korea contents Association vol.13 , no.4, 2013.4 ,459-468(10pages).
- [2] Hee-Young Kim, Chang-Ok Shin, "Availability of Mobile Art in Smart phone Environment of Augmented Reality Content Industrial Technology, The Korea contents Association vol.13, no5, 2013.5, 48-57(10pages)
- [3] Jae-In Lee, Jong-Soo Choi, Making Contents of the Science Education for the Element Schoolchildren based on the AR(Augmented Reality), The Korea contents Association vol.11, no11, 2011.11, 514-520(

7pages)

- [4] ttubeog-i , Augmented reality pedestrian directions navigation [Internet],<https://play.google.com/store/apps/details?id=com.binari.project.walkingthrough>
- [5] AR Ways, 2018 Pyeongchang Winter Olympics and Paralympics [Internet], <https://play.google.com/store/apps/details?id=com.m4nc.arnavi>
- [6] Pokémon GO [Internet], <https://play.google.com/store/apps/details?id=com.nianticlabs.pokemongo>
- [7] Street Lens [Internet] https://play.google.com/store/apps/details?id=com.trackyapps.street_lens
- [8] Ronald T. Azuma, “A Survey of Augmented Reality”, Presence: Teleoperators and Virtual Environments, Vol. 6. Iss. 4, pp. 355-385, 1997.
- [9] doopedia Augmented reality [Internet] <http://terms.naver.com/entry.nhn?docId=1227952&categoryId=32828>
- [10] Vuforia [Internet] <https://www.vuforia.com/>
- [11] Sydney [Internet] <https://www.sydney.com/>
- [12] Navercast GPS [Internet] <http://terms.naver.com/entry.nhn?docId=3571552&cid=59088&categoryId=59096>
- [13] Kakao navigation [Internet] <https://play.google.com/store/apps/details?id=com.locnall.KimGiSa>
- [14] Runtastic [Internet] <https://play.google.com/store/apps/details?id=com.runtastic.android>
- [15] Kyungtae Kim, Dongsu Seong, Keonbae Lee , Implementation of the Accuracy Improvement Alogorithm of GPS Location Information in Sight-Seeing Guidance System, The Korean Institute of Information Technology vol.10, no5, 2012.5, 189-195(7 pages)
- [16] Hwa-seon Kim, Chang-Young Kim, Imgeun Lee , Implementation of Sluice Valve management systems using GPS and AR, KIICE vol.21, no1, 2017.1 151-156(6 pages)
- [17] Jae-Heung Park, Sun-Hee Kang, Yeong-Geon Seo , Advanced Bus Information System Using Smart Phone GPS , Korean Society of Computer and Information vol.19, no12, 2014.12, 247-255(9pages)
- [18] Tac-Yeon Kim, Chang-Ju So, Joon Lyou , On-line Magnetic Distortion Calibration Method for a Magnetometer , ICROS, vol.18, no9, 2012.9 , 819-822(4pages)
- [19] Seong Yun Cho , Biaxial Accelerometer-based Magnetic Compass Module Calibration and Analysis of Azimuth Computational Errors Caused by Accelerometer Errors , ICROS vol.20, no2, 2014.2, 149-156(8pages)
- [20] Kim Hyesung , A Study on the Virtual Objects of Augmented Reality – AR Mobile LBS Application , Journal of Korea Design Knowledge Vol.18, 2011.6, 72-83(12pages)
- [21] T. Bailey, H. Durrant-Whyte, “Simultaneous Localization and Mapping(SLAM): part II”, IEEE Robotics & Automation Magazine, Vol. 13, Iss. 3, pp. 108-117, 2006
- [22] Augmented Reality Map [Internet] <https://play.google.com/store/apps/details?id=gklapp.armac>
- [23] Jae-pil Kim, Dong-cheol Lee, Development of Mobile Location Based Service App Using Augmented Reality , KIICE vol.18, no6, 2014.06, 1481-1487 (7 pages)
- [24] Wikipedia Great-circle distance [Internet] https://en.wikipedia.org/wiki/Great-circle_distance
- [25] Implementation of camera operation according to the actual orientation on the smart tablet[Internet] <http://vr-cto.hateblo.jp/entry/2016/05/02/070000>
- [26] Wikipedia Unity [Internet] [https://ko.wikipedia.org/wiki/%EC%9C%A0%EB%8B%88%ED%8B%B0_\(%EA%B2%8C%EC%9E%84_%EC%97%94%EC%A7%84\)](https://ko.wikipedia.org/wiki/%EC%9C%A0%EB%8B%88%ED%8B%B0_(%EA%B2%8C%EC%9E%84_%EC%97%94%EC%A7%84))

〈저자소개〉



김 상 준

- 2017년 ~ 현재 서울미디어대학원 대학교
뉴미디어학부 미디어공학 석사과정
- 관심분야: Computer Graphics, AR,
Projection Mapping, etc



배 윤 민

- 2015년 ~ 현재 세종대학교 만화애니메이션
학과 학사과정
- 관심분야: 3D modeling, character, design,
storyboard



최 유 주

- 1989년 이화여자대학교
전자계산학과(이학사)
- 1991년 이화여자대학교
전자계산학과(이학석사)
- 2005년 이화여자대학교
컴퓨터공학과(공학박사)
- 1991년 (주)한국컴퓨터 기술연구소 주임연구원
- 1994년 (주)포스테이타 기술연구소 주임연구원
- 2005년 서울벤처정보대학원
컴퓨터응용기술학과 조교수
- 2010년 ~ 현 재 서울미디어대학원대학교
뉴미디어학부 부교수
- 2015년 ~ 현 재 서울미디어대학원대학교
실감미디어연구소 교수
- 관심분야: Computer Graphics, Computer
Vision, HCI, Augmented Reality