

클라우드 기반 지리공간정보를 활용한 작전상황 혼합현실 가시화 시스템 설계

장영찬^{*,1)} · 박재일¹⁾ · 조은지¹⁾ · 곽송연¹⁾ · 신상헌¹⁾

¹⁾ 한화시스템(주)

Design of Mixed Reality Visualization System for Operational Situation Using Cloud-based Geospatial Information

Youngchan Jang^{*,1)} · Jaeil Park¹⁾ · Eunji Cho¹⁾ · Songyun Kwak¹⁾ · Sang Heon Shin¹⁾

¹⁾ *Hanwha Systems Corporation, Ltd., Korea*

(Received 8 November 2023 / Revised 5 December 2023 / Accepted 3 January 2024)

Abstract

The importance of geospatial information is increasingly highlighted in the defense domain. Accurate and up-to-date geospatial data is essential for situational awareness, target analysis, and mission planning in military operations. The use of high-resolution geospatial data in military operations requires large storage and fast image processing capabilities. Efficient image processing is required for tasks such as extracting useful information from satellite images and creating 3D terrain for mission planning. In this paper, we designed a cloud-based operational situation mixed reality visualization system that utilizes large-scale geospatial information distributed processed on a cloud server based on the container orchestration platform Kubernetes. We implemented a prototype and confirmed the suitability of the design.

Key Words : Cloud Server(클라우드 서버), Geospatial Information(지리공간정보), Mixed Reality(혼합현실), Web Map Tile Service(웹 맵 타일 서비스), Socket Communication(소켓 통신)

기 호 설 명

zoom : Web Map Tile Service zoom level

I_x : 타일 맵 서비스(TMS) X 인덱스

I_y : Web Map Tile Service Y 인덱스

Target_{lat} : 3D 객체 위도

Target_{lon} : 3D 객체 경도

V_t : 객체의 ECEF 좌표

V_c : 중심 타일의 ECEF 좌표

Center_{lat} : 지형데이터 중심 타일 위도

Center_{lon} : 지형데이터 중심 타일 위도 경도

* Corresponding author, E-mail: youngchan84.jang@hanwha.com
Copyright © The Korea Institute of Military Science and Technology

PoS_c : 중심 타일의 3D 공간 상 위치 벡터
 PoS_i : 3D 객체의 3D 공간 상 위치 벡터
 tileSize_{zoom} : zoom 별 타일 사이즈

1. 서론

정보통신과 컴퓨터 기술의 발전으로 인해 길 안내, 3차원 지도, 메타버스 등 지리공간정보를 활용한 다양한 서비스들이 급증하고 있으며, 이와 관련된 연구 개발도 활발히 진행되고 있다. Google은 Google Earth를 통해 3차원 지형 및 건물을 시각화한 지도 서비스를 제공하고 있으며, 우리나라도 공간정보산업진흥원에서 국민 누구나 이용 가능한 오픈 플랫폼 방식의 VWorld를 운영 중이다. 미국 국가지리공간정보국(National Geospatial-intelligence Agency)은 지리공간정보를 영상(imagery)과 영상 정보(imagery intelligence), 지리공간정보(geospatial information)가 통합된 정보로 정의하고 있다^[1].

지리공간정보의 중요성은 국방 분야에서 증대되고 있는데 정확하고 최신의 지리공간 데이터는 군사 작전의 상황 인식, 표적 분석, 임무 계획 등에 필수적이다. 세계 각국의 군대들은 지리공간 정보의 중요성을 인식하고 있으며, 데이터의 수집, 분석, 활용을 담당하는 부대를 운영하고 있다.

미국은 국가지리공간정보국에서 국방부 전체의 지리공간 데이터의 수집, 분석, 시각화, 정책 수립 등을 담당하고 있는데^[2], 17,000명 이상의 인원이 근무하고 있다. 우리 군에서도 별도의 조직을 운영하여 지리공간 정보를 수집하고 관리하고 있다.

군사 작전을 위해 고해상도의 지리공간정보 데이터를 활용하기 위해서는 대용량의 저장 공간과 빠른 이미지 처리 능력이 요구된다. 위성 이미지로부터 유용한 정보를 추출하고 임무 계획 수립을 위한 3차원 지형을 생성하는 등의 작업에 효율적인 이미지 처리가 필수적이다.

이를 위해 본 연구에서는 쿠버네티스^[3] 기반 클라우드 서버에서 수신한 대용량 지리공간정보를 활용하여 클라우드 기반 작전상황 혼합현실 가시화 시스템을 설계하고 프로토타입 구현을 통해 설계 적합성을 검증하고자 한다. 또한 본 연구는 기존의 지리공간정보를 혼합현실로 가시화하는 연구를 확장하여 국방 분야에 혼합현실 콘텐츠 활용한 임무계획 수립, 작전 상

황 도시 등을 위해 시나리오를 통해 입력받은 위경도 데이터를 기반으로 상호작용이 가능한 3D 객체를 혼합현실 콘텐츠로 가시화하는 방법 및 3D 객체 정보 변경 시 실시간으로 공유하는 방법을 제안하였다.

본 논문의 구성은 다음과 같다. 2장 관련 연구에서는 지리공간정보 서비스의 성능개선 및 지리공간정보를 활용한 혼합현실 콘텐츠와 관련된 연구에 대해 기술하였고 3장에서는 클라우드 서버로부터 영상, 지형, 3D 건물 타일 데이터를 수신하여 Hololens2로 가시화하는 작전 상황 혼합현실 가시화 시스템 설계 내용을 기술하였다. 4장에서는 3장에서 설계한 내용의 적합성을 검증하고자 구현한 프로토타입에 관해 기술하였다. 마지막으로 5장은 본 논문의 결론과 향후 연구방향을 제시하고자 한다.

2. 관련 연구

Kim(2017)은 대용량의 3차원 지리공간정보 서비스 향상 및 상호운용성 확보를 위해 한국형 공간정보 오픈플랫폼 서비스(VWorld)^[4]에서 사용하고 있는 3차원 데이터 포맷의 개선 방안을 제시하였다^[5]. 한국형 공간정보 오픈플랫폼의 3차원 건물 데이터는 AutoDesk사의 3DS 포맷을 참고하여 만든 XDworld Object(XDO) 포맷을 사용하는데 웹 서비스 환경에서의 효율적 운영과 경량화를 위해 JSON 포맷을 제안하였으며 렌더링 테스트를 통해 속도 개선 효과를 확인하였다.

Lee(2019)는 그래픽 처리장치가 없는 컴퓨팅 환경에서도 지리공간정보 서비스가 가능하도록 웹브라우저 환경 기반의 공간정보 플랫폼을 제시하였다.^[6] 기존 쿼드트리 기반 타일구조 방식에서 서로 다른 레벨의 타일이 인접할 때 발생하는 간격을 최소화하고자 3D지형 객체모델 생성방법을 제안하였으며, 실험을 통해 타일 간 간격 감소와 GPU가 없는 컴퓨팅 환경에서도 최소 50 FPS 이상의 속도로 렌더링하는 것을 확인하였다.

Kim(2022)은 혼합현실 기술을 공간정보에 적용한 3차원 도시모델을 활용하여 다중 사용자가 공통 업무를 수행할 때 필요한 협업 기술을 제시하였다^[7]. 지리공간정보 플랫폼에서 건물, 도로 등의 공간 객체를 수신하고 이를 Hololens2에서 가시화하였으며, 가상 객체와 사용자 사이의 인터랙션을 기반으로 동일한 시점을 공유하고 협업할 수 있는 기술을 개발하였고 실험을 통해 성능저하 없이 정상 동작함을 확인하였다.

3. 작전 상황 혼합현실 가시화 시스템 설계

본 연구에서는 대용량, 고해상도의 지리공간정보 데이터를 활용한 작전 상황 혼합현실 가시화 시스템 설계와 실제 적합성을 검증하는 프로토타입 구현을 목적으로 한다. 대용량, 고해상도의 지리공간정보 데이터는 컨테이너 오케스트레이션 플랫폼인 쿠버네티스 기반의 클라우드 서버로부터 수신하며 본 연구는 수신한 데이터를 Microsoft사의 Hololens2 디바이스를 활용하여 혼합현실로 가시화하는 시스템을 대상으로 한다. 본 연구의 요구사항은 Table 1과 같다.

Table 1. Requirement of mixed reality visualization system for operational situation

요구사항	상세 내용
혼합현실 콘텐츠 생성	지리공간정보 데이터를 파싱하여 혼합현실 콘텐츠로 변환할 수 있어야 함
3D 확대/축소/회전/이동	전시된 홀로그램을 선택하여 확대, 축소, 회전, 이동할 수 있어야 함
이미지 오버레이	전시된 홀로그램 지형 위에 다른 지도 이미지를 오버레이하여 보여줄 수 있어야 함
거리 및 면적 측정	전시된 지역에서 원하는 두 지점을 선택하여 거리를 측정하거나 면적을 측정할 수 있어야 함
지리공간정보 공유	Hololens2를 착용한 다른 사용자가 동시에 접속할 때 정보가 동기화되어 동시에 볼 수 있어야 함

각각의 요구사항을 만족하면서 사용자가 작전상황 시 UI 기반으로 운용이 가능하도록 Fig. 1과 같이 MVC 디자인 패턴을 적용한 아키텍처 설계 및 클래스 다이어그램을 설계하였다. UIManager 클래스에서 UI를 통해 사용자의 입력을 받도록 설계하였으며 TrainingManager 클래스에서 입력을 모델로 전달하고 처리한 결과를 수신하여 업데이트하도록 설계하였다.

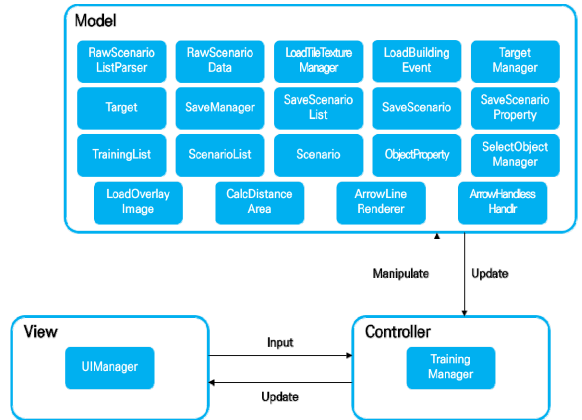


Fig. 1. Architecture of mixed reality visualization system for operational situation

3.1 혼합현실 콘텐츠 생성 모듈 설계

지리공간정보 플랫폼은 대용량의 지리공간정보 데이터를 효율적으로 서비스하고자 타일 방식으로 분할하여 인덱스와 줌 레벨을 부여하고 인덱스와 줌 레벨을 기준으로 지리공간정보 데이터를 요청 시 해당 데이터를 제공하는 Web Map Tile Service(이하 WMTS) 방식을 사용한다. 혼합현실 콘텐츠 생성 기능은 사용자가 UI를 통해 로드할 시나리오를 선택하면 해당 시나리오의 위경도 값과 도시 반경을 WMTS 인덱스와 줌 레벨로 변환 후 해당 인덱스와 줌레벨을 입력 파라미터로 서버에 전달하여 지리공간정보 데이터를 요청하며, 클라우드 서버로부터 영상, 지형, 건물 데이터를 수신하여 혼합현실 콘텐츠로 변환하는 기능이다. 본 연구에서는 이를 위해 UIManager 클래스로부터 사용자 입력을 받도록 설계하였으며, TrainingManager 클래스에서 사용자 입력을 LoadTileTextureManager 클래스로 전달하면 입력받은 위경도 값과 도시 반경을 변환한 줌 레벨을 식 (1)과 식 (2)에서 사용하여 WMTS의 X, Y 인덱스(I_x, I_y)로 변환하도록 설계하였다.

$$I_x = \left\lfloor \frac{(longitude + 180)}{360} \cdot 2^{zoom} \right\rfloor \quad (1)$$

$$I_y = \left\lfloor \frac{1}{2} \left(1 - \log \left(\tan \left(\frac{\pi \cdot latitude}{180} \right) + \frac{1}{\cos \left(\frac{\pi \cdot latitude}{180} \right)} \right) \right) \cdot 2^{zoom} \right\rfloor \quad (2)$$

계산한 WMTS의 인덱스를 사용해 클라우드 서버에서 영상, 지형, 건물 데이터 및 상세 시나리오를 다운

로드하여 파싱 후 혼합현실 콘텐츠로 변환하도록 설계하였다. 또한 상세 시나리오의 정보를 기반으로 고정, 회전, 전차, 장갑차 등의 객체(Target)를 생성하고 시나리오에 기록된 각 객체의 위경도 값을 사용하여 컴퓨터가 생성한 3차원 공간에 매핑하도록 설계하였다.

위경도 값을 사용하여 3차원 공간에 매핑하기 위해서는 몇 가지 단계를 거치는데 먼저 객체(Target)의 위경도 값(Target_{lat}, Target_{lon})을 Earth-Centered, Earth-Fixed (이하 ECEF) 좌표계로 변환하여 벡터값(V_i)을 구한다. 이를 위해서 UnityGPSConverter 오픈소스 API (Coordinates)를 사용하였다^[8]. 다음 단계로는 로드한 지형 데이터의 중심 타일(CENTER)의 위경도 값(CENTER_{lat}, CENTER_{lon})을 ECEF 좌표(V_c)를 계산하는데, 선택한 시나리오의 위경도 값을 한 번 더 사용한다. GPS 좌표계에서 ECEF 좌표계를 변환하여 사용하는 이유는 벡터 연산을 통해 객체와의 차를 구하여 로드한 지형데이터의 중심타일을 기준으로 객체를 정확하게 매핑하기 위함이다. 마지막으로 계산한 중심 타일 ECEF 벡터값과 객체의 ECEF 벡터값의 차를 줌 레벨을 기준으로 로드한 타일의 크기(TileSize)로 나눈 후 중심 타일의 컴퓨터가 생성한 3차원 공간 상의 위치 벡터(Pos_c)를 더하여서 좌표(Pos_s)를 구하였다. Pos_c는 3차원 좌표의 원점(0, 0, 0)을 사용하였다. 여기서 줌 레벨마다 타일의 크기(TileSize)가 다른데 줌 레벨 별 타일의 크기는 Open Geospatial Consortium 표준^[9]에서 기술한 값을 사용하였다. 본 과정을 식으로 표현하면 아래와 같다.

$$V_i = GPSToECEF(Target_{lat}, Target_{lon}) \quad (3)$$

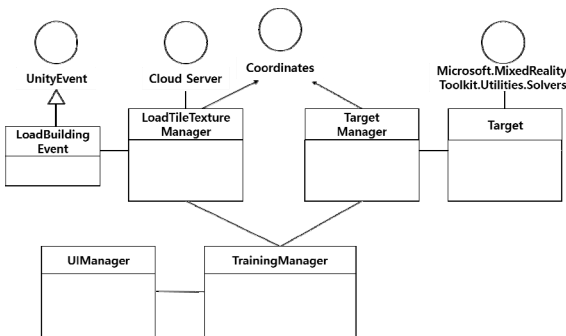


Fig. 2. Class diagram for mixed reality contents generation

$$V_c = GPSToECEF(CENTER_{lat}, CENTER_{lon}) \quad (4)$$

$$Pos_s = Pos_c + \frac{V_i - V_c}{TileSize_{zoom}} \quad (5)$$

이상의 과정을 거쳐 생성된 객체 전체는 TargetManager에서 관리하고 각 객체의 속성정보는 Target을 통해 관리하도록 설계하였다. Fig. 2는 혼합현실 콘텐츠 생성 클래스 다이어그램이다.

3.2 3D 확대/축소/회전/이동 모듈 설계

3D 확대/축소/회전/이동 기능은 사용자가 UI를 통해 확대/축소/회전 기능을 선택하거나 3D 이동 기능을 선택 후 제스처 인식을 통해 3차원 혼합현실 지형을 확대/축소/회전하거나 선택하여 다른 위치로 이동시킬 수 있다. 본 연구에서는 3D 확대/축소/회전 기능을 위해 Microsoft에서 제공하는 Mixed Reality Toolkit(이하 MRTK)^[10]의 BoundsControl API를 사용하고자 하며, MRTK의 ObjectManipulator API를 활용하여 생성한 3차원 지형을 선택하고 이동할 수 있도록 설계하였다. UIManager 클래스에서 사용자의 3D 확대/축소/회전과 3D 이동 명령을 입력으로 받도록 설계하였으며, TrainingManager 클래스에서 사용자 입력을 전달받은 후 LoadTileTextureManager 클래스의 BoundsControl API 또는 ObjectManipulator API를 호출해 해당 기능을 제공하도록 설계하였다. Fig. 3은 3D 확대/축소/회전/이동 클래스 다이어그램이다.

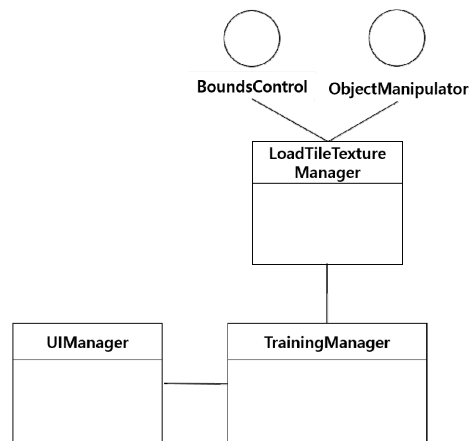


Fig. 3. Class diagram for 3D scale, rotation and move

3.3 이미지 오버레이 모듈 설계

이미지 오버레이 기능은 사용자가 UI를 통해 이미지 오버레이 기능을 선택하면 전시된 3D 지형의 재질(texture)인 위성영상 이미지 위에 레스터 지도 이미지를 덮어씌우는 기능이다. 본 연구에서는 LoadOverlayImages 클래스에서 현재 전시되고 있는 3차원 혼합현실 지형과 동일한 지형을 로드하고 위성영상이 아닌 레스터 지도 이미지를 지형의 재질로 사용하도록 설계하였다.

3.4 거리 및 면적 측정 모듈 설계

거리 측정 기능은 사용자가 UI를 통해 거리 측정 기능을 선택 후 제스처 인식으로 혼합현실 지형 위의 두 점을 선택하면 두 점 사이의 거리를 계산하여 UI로 시현하는 기능이며 면적 측정 기능은 사용자가 UI를 통해 면적 측정 기능을 선택 후 혼합현실 지형 위의 네 점을 선택했을 때 네 점을 잇는 사각형의 면적을 계산하여 시현하는 기능이다. 거리 및 면적 측정 모두 선택한 점의 3차원 좌표를 위경도로 변환 후 위경도 값을 사용하여 거리/면적을 계산하였다. 본 연구에서는 ArrowHandlessHandler 클래스에서 사용자가 지형 위의 한 지점을 선택하는 것을 감지한 후 CalcDistanceArea 클래스에서 거리 측정 시에는 두 개의 지점을 연결하는 거리를 계산하며, 면적 측정 시에는 네 개의 지점을 잇는 사각형의 면적을 계산한다. 또한 계산한 결과 값 뿐 아니라 사용자가 선택한 점을 잇는 선을 사용자가 볼 수 있도록 ArrowLineRender 클래스에서 선을 그리는 기능을 제공하도록 설계하였다.

Fig. 4는 이미지 오버레이, 거리 측정 및 면적 측정 클래스 다이어그램이다.

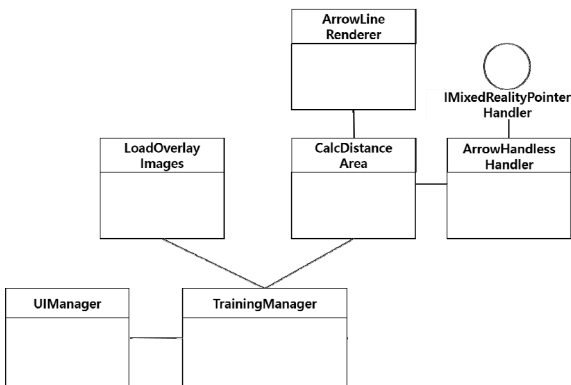


Fig. 4. Class diagram for image overlay, distance measurement and area superficial measurement

3.5 지리공간정보 공유 모듈 설계

지리공간정보 공유 기능은 Hololens2를 착용한 두 명 이상의 사용자가 클라우드 서버에 저장된 동일한 시나리오를 로드하여 작전을 수행할 때 한 사용자가 객체 정보를 변경하면 클라우드 서버를 통해 변경된 정보를 공유, 다른 사용자에게도 변경된 정보가 동기화되어 참여자 모두가 동일한 작전 상황을 볼 수 있도록 하는 기능이다.

지리공간정보를 공유하기 전에 먼저 객체 이동, 삭제 등 객체 정보의 변경이 발생하는데 사용자가 손가락으로 혼합현실 지형 위의 객체를 선택하면 제스처 인식을 통해 선택한 객체를 식별하고 상호작용할 수 있도록 SelectObjectManager 클래스를 설계하였고 정보 변경 시 TargetManager 클래스와 Target 클래스를 통해 객체들을 관리하도록 설계하였다. 3차원 공간 상의 3D 객체의 이동 시 위경도 데이터가 변경되는데 변경된 위경도 값은 식 (3), 식 (4), 식 (5)를 역으로 계산하여 구하였다.

Table 2. Design of packet protocol for sharing geospatial information

요소명	타입	크기
상호작용 유형(type)	Int	4
객체 ID(id)	Int	4
구분(forceID)	Int	4
부대규모(forceSize)	Int	4
군대부호코드(forceCode)	Char	20
단대호식별명(classification)	Int	4
부대명(unitName)	Char	20
부대약어명(unitCode)	Char	20
적아구분(isAlly)	Bool	1
부대위치 중 위도(unitLat)	Float	4
부대위치 중 경도(unitLon)	Float	4
부대위치 중 고도(unitHeight)	Int	4
군대부호 도시여부(showForceMark)	Bool	1
임무내용(mission)	Char	40
이동방향(direction)	Int	4

공유 데이터 형식은 사용의 편의성을 고려하여 개방형 교환 포맷인 JSON(Javascript Object Notation)을 사용하고자 하며, TCP/IP 기반의 소켓통신 방식으로 데이터 전송 시에는 직렬화를 수행하고 데이터 수신 시에는 역직렬화를 수행하고자 한다. 공유되는 정보는 고정익, 전차 등 객체의 이동, 삭제 및 속성 변경에 관한 정보이며 Table 2와 같이 패킷 프로토콜을 설계하였다.

패킷 프로토콜 중 상호작용 유형(type)은 객체 생성(1), 이동(2), 삭제(3) 발생 시 해당값을 나타내며, 객체 ID(id)는 운용 중인 시나리오의 각 객체들을 구분하는 고유 ID를 나타낸다. 구분(forceID)은 객체가 보병(1), 탱크(2), 항공기(3), 회전익(4) 인지를 나타낸다. 임무내용(mission)은 문자열로서 사용자가 시나리오를 작성하거나 임무 중 객체 생성 시 사용자가 부여한 각 객체별 임무를 나타낸다. 이동방향(direction)은 정북방향(0°)을 기준으로 객체의 방향을 나타내며 객체가 이동하거나 또는 객체의 이동방향만 변경되었을 시 변경된 값을 나타낸다. 상술한 데이터 공유와 관련한 데이터 전송, 수신 및 동기화는 ShareObjectInfo 클래스를 담당하여 처리하도록 설계하였다.

작전 종료 시에는 현재 상황을 저장하여 향후 로드 후 해당 상황을 이어서 진행할 수 있도록 SaveManager 클래스를 설계하였다. 이 때 저장되는 파일은 총 2개로 시나리오의 개략정보를 저장하는 ScenarioList과 시나리오 상세 정보 및 객체 속성을 저장하는 Scenario 파일인데 Json 형식으로 파일을 생성하고자 하며 저장한 파일은 클라우드 서버에 업로드하여 관리되도록 설계하였다. Fig. 5는 지리공간 공유 클래스 다이어그램이다.

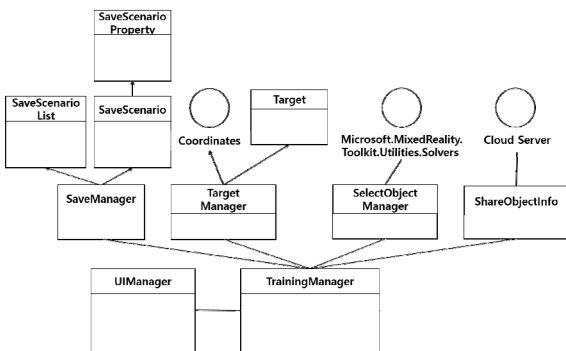


Fig. 5. Class diagram for sharing geospatial information

4. 작전상황 혼합현실 가시화 프로토타입 구현

3장에서 서술한 각 모듈에 대한 설계 내용을 토대로 설계 적합성 및 정상 동작 등을 검증하고자 상용 게임엔진인 Unity 기반으로 프로토타입을 구현하였다. 클라우드 서버로부터 영상 및 지형 데이터를 요청하고 다운로드 받아야 하지만 프로토타입 구현 단계에서는 로컬 저장소에 해당 데이터를 사전에 저장하고 데이터에 접근하여 로드하는 방식으로 구현하였다. 영상 및 지형 데이터는 인터넷에 공개된 태국 방콕 지역 데이터를 사용하였으며 건물 데이터 로드는 제외하였다. UI는 혼합현실 콘텐츠에 맞게 투명도 반영 및 사용자 시선을 추적하여 시선 내에 UI가 항상 시현되도록 시선 추적 기능을 적용하였으며, 기능별 프로토타입 동작 확인은 노트북과 홀로렌즈2에서 진행하였으며 혼합현실 콘텐츠 생성과 관련하여 테스트를 수행하여 두 기기간 성능을 비교하였다. 프로토타입 동작 실험환경은 Table 3과 같다.

Table 3. Mixed reality visualization system prototype operation test environment

Operation environment	
Specs	Laptop CPU: Intel i5-1135G7 RAM: 8 GB GPU: Intel Iris Xe Graphics
	Hololens2 AP: Qualcomm Snapdragon 850 RAM: 4 GB GPU: Adreno 630
Center Longitude/ Latitude	Longitude: 100.476685 Latitude: 13.75563
Zoom level	14
Image data	Google map data(png file) ^[11]
Terrain data	Maptiler data(terrain file) ^[12]
Map range	1 km, 3 km, 5 km
Rendering Engine	Unity3D
3D Object Asset	F-15K, K-2(.fbx file)

4.1 혼합현실 콘텐츠 생성 모듈 프로토타입 구현

시나리오 기반의 객체 생성 및 매핑을 검증하기 위해서 fbx 형식의 F-15K 고정익과 K-2 전차 객체를 사용하였다.

Table 4. Test result of mixed reality contents creation
Unit : msec

Map range	Count of Test	Latency		Map range	Count of Test	Latency	
		Lap top	Holo len2			Lap top	Holo len2
3	1	241	6176	5	1	282	7403
	2	252	6071		2	330	7312
	3	249	6315		3	314	7324
	4	245	6271		4	293	7347
	5	225	6456		5	315	7580
	6	265	6365		6	309	7159
	7	253	6860		7	424	7065
	8	223	6361		8	301	7198
	9	230	6379		9	338	7597
	10	248	6680		10	349	7415
Mean	243.1	6393.4	Mean	325.5	7340		

혼합현실 콘텐츠 생성의 성능 비교를 위해 전시 반경(Map range)를 3 km, 5 km로 변경하며 노트북과 Hololens2에서의 로딩시간을 측정하였다. 간 전시 반경 마다 10회씩 반복하여 수행하였으며 실험결과는 Table 4와 같다.

실험을 통해 노트북 환경에서는 전시 반경이 증가하더라도 로딩까지의 지연 시간이 1초 미만이 소요되나 Hololens2에서는 3 km 반경을 전시할 때 평균 6.39 초, 5 km 반경 전시 시 평균 7.34초가 지연됨을 확인할 수 있었다. 이는 하드웨어의 성능 차이로 발생한 것으로 추정된다. 또한 위경도 값을 변환하여 매핑한 객체의 위치와 Google Map에서의 위경도 값을 입력했을 때의 위치를 비교하여 위치가 일치함을 확인할 수 있었다.

Fig. 6은 Hololens2에서 혼합현실 콘텐츠 생성 프로토타입을 동작한 화면이다.

4.2 3D 확대/축소/회전/이동 모듈 프로토타입 구현

3D 확대/축소/회전과 이동을 기능을 독립적으로 선택할 수 있도록 UI를 구현하였으며, 확대/축소/회전 시에는 MRTK의 BoundsControl API를 사용하여 혼합현실 지형 전체를 박스로 묶어서 하나의 영역으로 컨트롤할 수 있도록 구현하였는데 축소 시 지형 위의 객체들이 지형 위에서 떨어지는 것을 방지하기 위해 확대/축소/회전 시에는 객체들을 비활성화한 후 확대/축소/회전에 종료 시 활성화하도록 구현하였다. 이동 시에는 MRTK의 ObjectManipulator API를 사용하여 사용자가 손가락으로 혼합현실 지형을 잡고 이동할 수 있도록 구현하였다. Fig. 7은 3D 확대/축소/회전/이동 프로토타입 중 축소/이동 동작 화면이다.



Fig. 6. Prototype for mixed reality contents generation



Fig. 7. Prototype for 3D scale, rotation and move

4.3 이미지 오버레이 모듈 프로토타입 구현

이미지 오버레이 기능은 혼합현실 지형의 재질 위에 레스터 지도 이미지를 덮어씌우는 기능으로 프로토타입 구현 시에서는 레스터 지도 대신 지형 재질과 동일한 위성영상 이미지를 사용하되 높이를 조절하여 기존 지형 재질과 비교할 수 있도록 하였다. 프로토타입을 통해 기존 지형 위에 이미지가 올라오는 것을 확인할 수 있었다. Fig. 8은 이미지 오버레이 프로토타입 동작 화면이다.

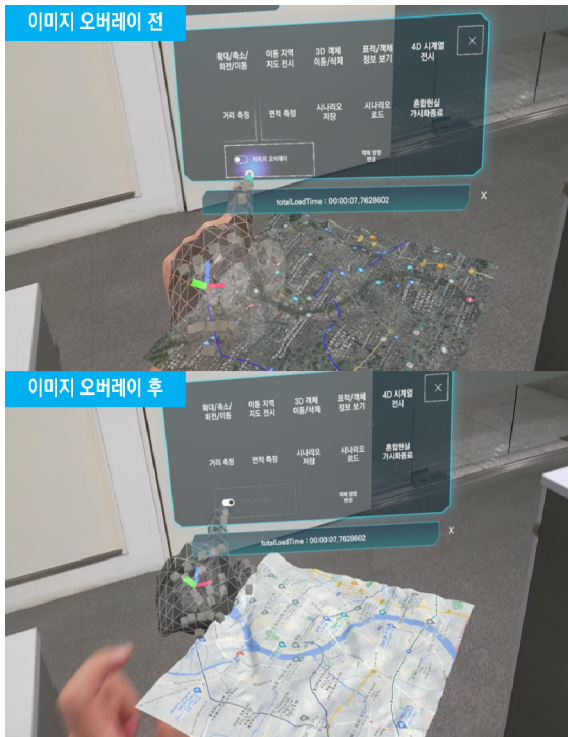


Fig. 8. Prototype for image overlay

4.4 거리 및 면적 측정 모듈 프로토타입 구현

3장에서 서술한 설계 내용을 기반으로 거리 및 면적 측정 모듈 프로토타입을 구현하였으며 이를 Google Earth에서 제공하는 거리 및 면적 측정 기능과 결과값을 비교하였는데 거리 및 면적 측정 모두 5% 이내의 차이가 있었으나 각 지도마다 점을 찍을 때 발생하는 위치 차이를 고려했을 때 정상적으로 기능이 동작함을 확인할 수 있었다. Fig. 9와 Fig. 10은 거리 및 면적 측정 프로토타입 동작 화면이다.



Fig. 9. Prototype for distance measurement



Fig. 10. Prototype for area superficial measurement

4.5 지리공간정보 공유 모듈 프로토타입 구현

지리공간정보 공유 모듈 프로토타입은 객체 정보 변경이 발생하는 객체 이동 및 삭제 기능을 구현하였으며, 패킷 전송의 동작을 확인하기 위해 테스트 서버를 구현하였다. 테스트 서버는 Python을 사용하여 멀티 쓰레드 기반으로 구현하였고 혼합현실 가시화 시스템에서 전송한 패킷을 가공하지 않고 서버와 접속한 다른 참여자에게 전달하도록 구현하였으며 정상적으로 동작함을 확인하였다.

시나리오 저장 기능은 UI를 통해 시나리오를 새로 저장하거나 기존 시나리오를 덮어쓸 수 있도록 프로토타입을 구현하였다. 저장한 개별 시나리오는 Fig. 11과 같으며 시나리오 저장 기능도 정상 동작함을 확인하였다.


```

{
  "objects": [
    {
      "objectID": 0,
      "forceSize": 1,
      "forceCode": "00-12",
      "classification": 2,
      "unitName": "**부대",
      "unitCode": "00-001",
      "isAlly": true,
      "unitLat": 13.7552919,
      "unitLon": 100.492706,
      "height": 1.03399968,
      "showForceMark": true,
      "mission": "적을 향해 돌진",
      "movingDirection": 0
    },
    {
      "objectID": 1,
      "forceSize": 2,
      "forceCode": "00-16",
      "classification": 3,
      "unitName": "**대대",
      "unitCode": "00-002",
      "isAlly": true,
      "unitLat": 13.7558136,
      "unitLon": 100.492783,
      "height": 1.004,
      "showForceMark": false,
      "mission": "진형 좌측 돌파",
      "movingDirection": 0
    }
  ],
  "scenarioID": "local_sce_002",
  "scenarioName": "2023-04-20 02:34:08",
  "folderName": "2023-04-A-MR",
  "writer": "MR",
  "mapType": "",
  "latitude": 13.75117,
  "longitude": 100.4926,
  "scale": 0,
  "range": 0,
  "objectCount": 3,
  "description": "",
  "rightToControl": "",
  "writeDay": "2022-01-01",
  "revisionDay": "2023-04-20"
}

```

Fig. 11. Scenario JSON file of mixed reality visualization system

였으며 설계 적합성을 검증하기 위해 프로토타입 구현 내용에 대해 제시하였다.

입력받은 위경도 값을 WMTS 인덱스 기반으로 변환하여 영상, 지형, 건물 데이터를 파싱 후 혼합 현실 콘텐츠를 생성하도록 설계하였으며, 기존 지도 영상, 지형, 건물 등 지리공간정보를 혼합현실로 가시화하는 연구를 확장하여 시나리오를 통해 입력받은 위경도 데이터를 기반으로 상호작용이 가능한 3D 객체를 컴퓨터가 생성한 3차원 공간상에 혼합현실 콘텐츠로 생성 및 가시화하는 방법을 제안하였다. 시나리오 정보를 기반으로 생성한 혼합현실 지형을 확대, 축소, 회전 및 이동이 가능하도록 설계하였으며, 영상, 지형 데이터 위에 레스터 지도 이미지를 덮어씌우는 이미지 오버레이 기능을 설계하였다. 두 점을 잇는 거리 측정과 네 점을 잇는 사각형의 면적을 측정할 수 있는 기능과 객체의 정보 변경 시 같은 시나리오를 수행하는 참여자에게 공유하는 TCP/IP 소켓통신 방식의 지리공간정보 공유 기능을 설계하였다. 마지막으로 설계적합성을 검증하기 위해 프로토타입을 구현하여 기능 동작을 확인하였으며, 실험을 통해 혼합현실 콘텐츠 생성 시 로딩시간을 측정하고 노트북과 Hololens2의 성능을 비교하였다.

향후 연구방향으로는 클라우드 서버 구축 및 연동을 통해 혼합현실 콘텐츠 생성 시 성능을 분석하고 이를 최적화할 예정이다. Hololens2에서 혼합현실 콘텐츠 생성 시 지연시간은 클라우드 서버 연동 시 증가할 것으로 예상되는데 이를 단축하기 위해 멀티 스레드 방식을 적용하고자 한다. 지리공간정보 공유기능은 다수의 참여자들이 통신 시 발생하는 동기화 문제를 해결하고 지연 시간을 최소화하는 논블록킹 방식의 서버를 구현할 예정이다.

이를 통해 고성능의 컴퓨터에서 동작하는 지리공간 정보 가시화 서비스를 iOS, 안드로이드, 윈도우 모바일 등 모바일 플랫폼 기반에서도 운용이 가능할 것으로 기대된다.

5. 결론

본 연구에서는 클라우드 서버에서 수신한 대용량의 지리공간정보를 혼합현실 콘텐츠로 가시화하는 작전 상황 혼합현실 가시화 시스템의 설계 방안을 기술하

후 기

이 논문은 2023년 정부(방위사업청)의 재원으로 국방 기술진흥연구소의 지원을 받아 수행된 연구임(KRIT-CT-21-040).

References

- [1] National Geospatial-Intelligence Agency, “National System for Geospatial Intelligency,” <http://www.irp.fas.org/agency/nga.doctrine.pdf>, p. 7, 2006.
- [2] National Geospatial-Intelligence Agency, <https://www.nga.mil/index.html>.
- [3] Kubernetes Documentation, <https://kubernetes.io/>.
- [4] VWorld, <http://www.vworld.kr>
- [5] T. Kim, H. Jang, S. Yoo, J. Go, “3D Tile Application Method for Improvement of Performance of V-world 3D Map Service,” Journal of the Korean Society for Geospatial Information Science, Vol. 25, No. 1, pp. 55-61, 2017.
- [6] A. Lee, I. Jang, “Spatial Information Platform with VWorld for Improving User Experience in Limited Web Environment”, Electronics, Vol. 8, No. 12, pp. 1-20, 2019.
- [7] K. Kim, Y. Chang, E. Cho, S. Shin, I. Jang, “A Study on Collaborative Support for 3D City Models using Mixed Reality”, Journal of Software Assessment and Valuation, Vol. 18 No. 2, pp. 209-219, 2022.
- [8] GPS Encoder by Michael Taylor, <https://github.com/MichaelTylor3D/UnityGPSConverter/>.
- [9] Open Geospatial Consortium, “OGC Web Map Tile Service(WMTS) Simple Profile,” <http://docs.opengeo-spatial.org/is/13-082r2/13-082r2.html>, 2016.
- [10] Mixed Reality Toolkit3, <https://learn.microsoft.com/en-us/windows/mixed-reality/mrtk-unity/mrtk3-overview/>.
- [11] <https://mt1.google.com/vt/lyrs=r&x=12766&y=7558&z=14>
- [12] <https://docs.mapbox.com/cloud/api>