

디지털 트윈을 활용한 실시간 모니터링 및 원격제어 시스템의 테스트베드 구현

윤정은[†], 김원석^{**}

Implementation of Real-time Monitoring and Remote Control System Testbed based on Digital Twin

Jung-Eun Yoon[†], Won-Suk Kim^{**}

ABSTRACT

Digital twin has the advantages of quality improvement and cost reduction, so it is widely applied to various industries. In this paper, a method to implement the major technologies of digital twin easily and quickly is presented. These include data management and relay servers, real-time monitoring applications including remote control interfaces, and direct connection protocols for video streaming. In addition, an algorithm for controlling a two-wheeled vehicle with a 2D interface is also proposed. The implemented system performs near real-time synchronization between the real environment and the virtual space. The delay time that occurs in remote control of the vehicle in the real environment was compared with the results of applying the proposed delay time reduction method. In addition, in the case of 2D interface-based control, an algorithm that can guarantee the user experience was implemented and applied to the actual environment and verified through experiments.

Key words: Cloud computing, Remote control system, Monitoring application, Digital twin

1. 서 론

디지털 트윈(Digital Twin) 기술은 2014년 GE가 처음 제안한 이후 인공지능, 블록체인, 빅데이터 등과 함께 4차 산업을 주도하는 2020 가드너 미래 10대 전략기술에 선정되었으며, 2021년 역시 주요 기술로 선정된 바 있다[1]. 디지털 트윈이란 우리 주변에 있는 다양한 기기와 통신 장비로부터 얻은 데이터를 가공하여 컴퓨터에 현실과 똑같은 가상현실을 구현하는 기술이며, 이를 활용하면 현실에서 발생할 수 있는 상황을 시뮬레이션하여 결과를 예측할 수 있다. 네트워크에 연결된 기기들의 상태나 실내 환경정보

를 모니터링하여 가상공간에 보여주거나, 가상공간에서의 제어를 실제 기기에 반영하는 등 양방향 제어가 가능하다[2]. 이는 품질향상 및 비용 절감의 장점이 있어 제조업, 물류, 스마트 시티 등 다양한 산업군에 폭넓게 적용되고 있다.

디지털 트윈을 실제 환경에 구현하고 활용하기 위해서는 몇 가지 핵심 기술이 요구된다. 현실 세계 센싱 및 센싱 데이터 수집 기술, 수집된 데이터 기반 가상공간 구축 기술, 현실 센싱 데이터와 가상공간 간 데이터 동기화 기술, 가상공간을 시각적으로 표현할 수 있는 시각화 기술, 가상공간을 활용하여 미래 예측 등을 수행할 수 있는 시뮬레이션 기술이다. 시뮬

※ Corresponding Author : Won-Suk Kim, Address: (36729) Gyeongdong-ro 1375, Andong-si, Gyeongsangbuk-do, Korea, TEL : +82-54-820-7968, FAX : +82-54-820-6257, E-mail : wonsukkim@anu.ac.kr
Receipt date : Feb. 9, 2022, Approval date : Feb. 23, 2022

[†] Dept. of Multimedia Eng., Andong National University (E-mail : yje3058@naver.com)

^{**} Dept. of Multimedia Eng., Andong National University
※ This work was supported by a grant from 2020 Research Fund of Andong National University

레이션 결과를 바탕으로 사용자에게 서비스를 제공하거나 현실 세계의 객체를 제어하는 응용서비스 기술이 필요하다. 즉 디지털 트윈을 구현하기 위해서는 데이터 관리 및 수집, 동기화, 시뮬레이션, 응용 서비스 등의 기술들이 구현되어야 하며 이들이 서로 유기적으로 동작할 수 있어야 한다[3].

디지털 트윈 기술을 활용하여 건물 관제 시스템, 자동차 부품 생산라인, 스마트 시티 등과 같은 다양한 분야에서 관련 연구들이 진행되었다[4,5,6]. 그러나 디지털 트윈은 가상현실, IoT, Big Data 등의 신기술들이 종합된 형태로 다양한 분야의 전문가들이 협업하여 시스템을 구축해야 하는 한계점을 가지고 있어 시스템 구축에 많은 시간과 비용이 소요된다[7]. 이러한 문제를 해결하기 위해 디지털 트윈 시스템 및 서비스를 저비용으로 단시간에 구축할 수 있는 도구로 게임엔진을 활용한 연구들이 수행되고 있다[3,8,9].

본 논문에서는 디지털 트윈 구축에 요구되는 핵심 기술을 기반으로 시스템을 설계하고 디지털 트윈을 쉽고 빠르게 구현할 수 있도록 게임엔진을 활용한다. 게임엔진을 활용하면 물리 세계에 존재하는 사물, 시스템, 환경 등을 가상공간에 쉽게 배치할 수 있으며 하나의 프로젝트를 윈도우, 맥OS, 안드로이드, iOS 등 다양한 플랫폼에 배포할 수 있고 이식이 용이하다[1].

본 연구에서 제작한 디지털 트윈 테스트베드는 다음과 같이 구성된다. 우선 실제 환경 구성을 위해 Wi-Fi 통신을 수행하는 RC(Remote Control) 차량을 제작하였으며 차량 위치 추적을 위해 UWB(Ultra Wide Band) 기반 실내 측위 기술을 활용하였다[14]. 가상공간은 대표적인 게임엔진인 Unity를 활용하여 구현하였으며, 실제 환경과 동기화되며 영상 스트리밍 및 차량 제어가 가능하도록 구성하였다. 본 논문에서는 가상공간 및 서버 구축에 집중하여 서술한다. 구축한 가상공간과 실제 환경의 동기화 및 데이터 수집을 위한 서버를 구현하여 양방향 제어 및 정보 교환을 지원한다. 추가로, 원격제어 애플리케이션에서 저 지연으로 실제 환경의 RC 차량을 제어하기 위해 중계 프로토콜을 설계하였으며, RC 차량의 속도 및 방향을 정교하고 직관적으로 제어하기 위한 2D 조이스틱 기반 이륜차량 제어 알고리즘을 제안한다.

2. 관련 연구

2.1 중계 프로토콜

RC 차량을 원격제어하는 연구는 다음과 같다. 라즈베리파이 기반 다기능 RC 차량 컨트롤러[10]는 애플리케이션과 RC 차량이 동일한 Wi-Fi 환경을 기반으로 RC 차량에 카메라를 부착하였다. 애플리케이션에서 직접 고유 IP를 입력한 후, RC 차량의 카메라 화면을 시청하며 버튼을 눌러 RC 차량을 제어하도록 설계하였다. 실시간으로 영상을 시청하며 차량을 제어할 수 있는 장점이 있으나 RC 차량과 애플리케이션이 동일한 환경에서 제어하도록 설계되어 있으므로 공간제약에 한계가 있다.

메카넘 휠을 사용한 촬영용 RC 차량[11]은 기존 RC 차량에 초음파 센서를 부착하였으며 IP 통신을 통해 애플리케이션과 라즈베리파이 기반 차량이 통신한다. 카메라 영상을 시청하며 버튼을 통해 방향을 제어할 수 있도록 설계하였다. 전진, 후진뿐만 아닌 대각선, 제자리 회전 등 10가지의 구동 방향으로 주행할 수 있으며 초음파 센서를 통해 일정 거리 이상 가까워지면 정지하는 장점이 있으나 IP 통신을 이용하기 때문에 일정 거리 이상 멀어지면 통신이 끊기는 문제점이 있다.

본 논문에서는 클라우드 서버를 활용하여 RC 차량과 애플리케이션이 통신하도록 설계하여 공간의 제약 없이 제어할 수 있으며 직접 통신을 위한 중계 프로토콜을 구축하였기 때문에 RC 차량과 애플리케이션 간 저 지연 원격제어가 가능하다.

2.2 원격제어 알고리즘

음성인식을 이용한 RC 차량은[12] 블루투스를 기반으로 통신하고, 사용자의 음성으로 RC 차량을 제어할 수 있도록 설계하였다. 해당 논문은 다중 명령어를 인식하여 처리할 수 있도록 구현하여 실제 버튼만큼의 정확성을 가지고 있으나 앞, 뒤, 좌, 우 4가지 기능만 제공하므로 속도를 제어하거나 대각선으로 주행하는 등의 방향을 세밀하게 제어하는 데 어려움이 있다.

탐색 알고리즘을 이용한 초음파 센서 기반 무선조종 RC 차량[13]은 자이로 센서와 초음파 센서의 출력 값을 기반으로 새로운 4bit 데이터를 활용해 RC 차량의 모터를 제어하였다. 장애물 감지 횡수, 충돌 횡수

실험을 통해 안정도 있는 장애물 감지를 검증하였으나 RC 차량을 구동하는 제어 데이터가 0, 1로 처리되어 있으므로 세밀하게 속도를 조절할 수 없는 문제점이 있다.

본 논문의 원격제어 시스템은 2D 조이스틱 기반 이륜차량 제어 알고리즘을 제안한다. 조이스틱 핸들 위치에 따라 RC 차량을 전진, 후진, 좌회전, 우회전 등과 같은 자유로운 방향 조절이 가능하며 0~255 사이의 세밀한 속도 제어가 가능하다.

3. 본 론

3.1 디지털 트윈 동기화 및 데이터 수집 클라우드 서버 구축

본 논문에서 제안하는 디지털 트윈 기반 위치추적 및 원격제어 시스템의 테스트베드 구성은 Fig. 1에 나타나 있다. 실제 환경에는 실내 측위 시스템과 RC 차량이 있다. 실내 측위 시스템은 UWB를 기반으로 하며, TWR(two-way ranging) 방식을 통해 UWB 앵커(Anchor) 3개와 RC 차량에 부착된 UWB 태그(Tag) 간 거리를 측정한다[14]. 각 앵커는 차량과의 거리 정보를 실내 측위 GW(Gateway)로 전송하고 GW는 거리 정보를 종합하여 차량의 위치를 계산한

후 REST API를 통해 서버로 실내공간 내 기준점 대비 상대 위치 좌표를 전송한다.

원격제어 모니터링 애플리케이션은 시각화된 가상공간, 실제 RC 차량과 동일한 위치에 배치된 차량 오브젝트, 영상 스트리밍 화면, RC 차량을 제어하기 위한 조이스틱, RC 차량의 ID와 좌표 값, 모드 변환 버튼으로 구성되어 있다. 그리고 실제 환경과 가상공간의 동기화를 위한 서버는 GW와 RC 차량으로부터 위치 좌표 및 차량 정보 데이터를 수신하는 수신부, DB(Database)와 연결하는 연결부, 가상공간으로 데이터를 송신하는 송신부로 구성되어 있다.

모니터링 데이터 및 RC 차량의 정보를 수집하고 관리하는 클라우드 서버의 흐름은 Fig. 2와 같다. RC 차량이 클라우드 서버에 최초 연결되면 차량의 외부 IP와 포트 번호를 RC 차량 연결정보 DB에 저장한다. 해당 연결정보는 추후 애플리케이션이 저 지연 통신을 위해 RC 차량과 직접 WebSocket 연결을 수립하기 위해 DB에 유지된다. 실내 측위 GW로부터 전달 받은 차량 위치정보는 실시간 모니터링뿐만 아니라 차후 이동기록 데이터로 활용될 수 있어 서버는 timestamp별 위치를 기록함과 동시에 애플리케이션으로 위치정보를 전달한다.

언급한 바와 같이, 애플리케이션이 클라우드 서버

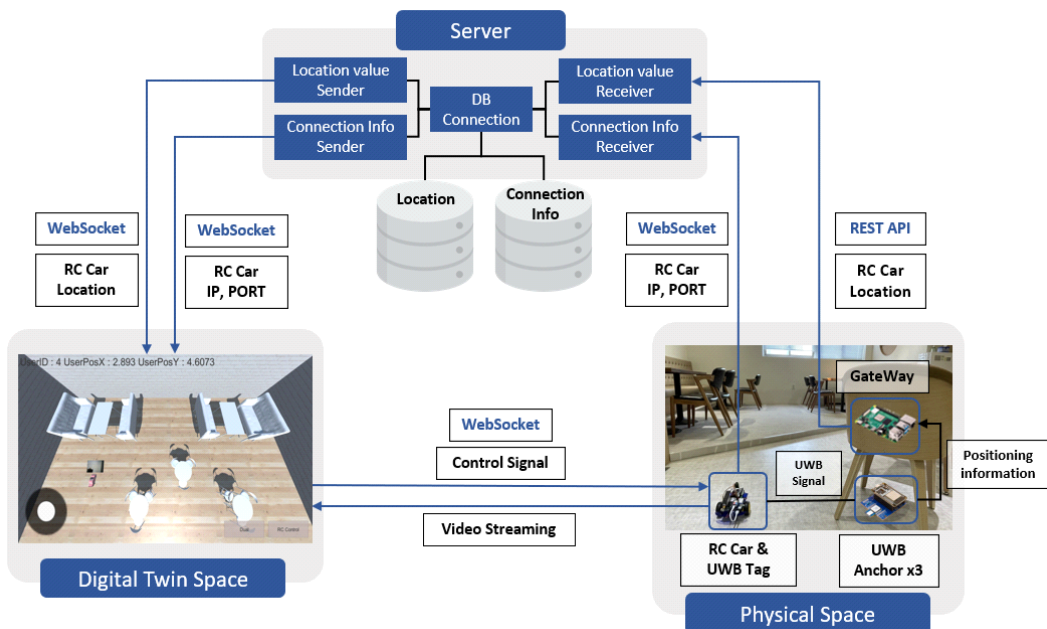


Fig. 1. System overview for Real-time location tracking and Remote Control System Testbed.

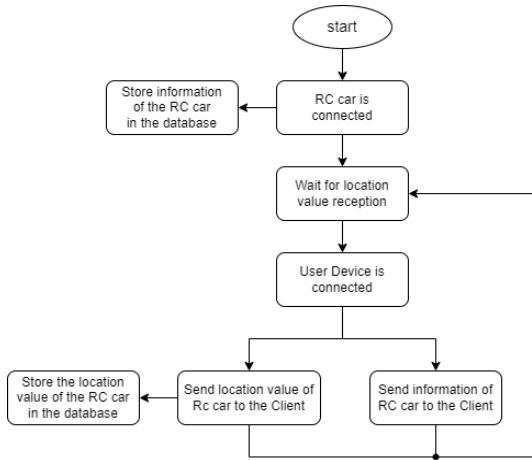


Fig. 2. A flowchart of a cloud server for collecting monitoring data and information on an RC vehicle.

를 통해 RC 차량을 제어하거나 영상 스트림을 수신하게 되면 대역폭이 불필요하게 낭비될 뿐만 아니라 지연시간도 다소 발생하게 된다[15]. 제안하는 시스템에서는 애플리케이션과 RC 차량이 직접 데이터를 주고받을 수 있도록 지원하며, Fig. 3은 이 절차를 도식화하여 보여준다. 우선 RC 차량이 클라우드 서버에 연결되면 RC 차량은 IP, 포트 번호와 같은 연결 정보를 클라우드 서버로 전송한다. 서버는 해당 IP와 포트 번호를 RC 차량 연결정보 DB에 저장한다. 그런 다음, 애플리케이션이 서버에 연결되면 목표하는 차량의 연결정보를 요청하며 서버는 해당 정보를 응답한다. 애플리케이션은 차량의 IP와 포트 번호를 사용하여 RC 차량과 WebSocket 연결을 수립한 후 직접 통신을 수행한다. 이를 기반으로 애플리케이션은 제어 신호를 전송하고 RC 차량은 부착된 카메라 영상을 스트리밍한다.

지연시간을 검증하기 위해 간단한 실험을 진행하

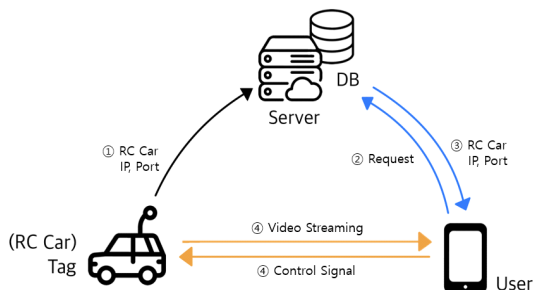


Fig. 3. Remote Control Communication Protocol.

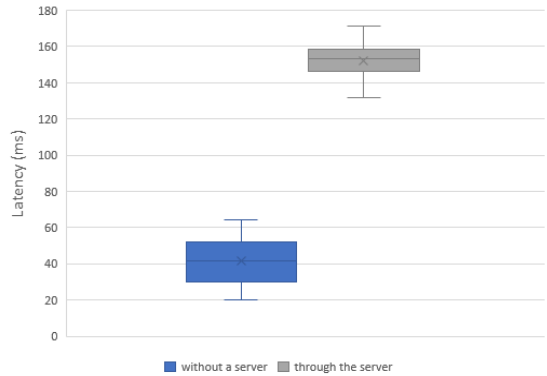


Fig. 4. Remote control Latency experiment.

였다. RC 차량과 애플리케이션이 서버를 거쳐 통신하는 경우와 직접 통신하는 경우의 지연시간을 비교하였다. 실험을 위해 5분간 연결을 유지하였으며, 통신 지연시간을 시간 동기화 없이 측정하기 위해 RTT (Round Trip Time)를 계산할 수 있도록 추가적으로 구현하였다. 좀 더 상세하게, 애플리케이션에서 RC 차량으로 현재 timestamp를 기록하여 제어 신호를 보내며 차량은 해당 신호를 그대로 반환한다. 애플리케이션이 반환된 신호를 수신하면 해당 신호에 기록된 timestamp와 현재 timestamp의 차이를 반으로 나누어 편도 지연시간을 측정하였다. 측정한 지연시간은 Fig. 4와 같은 Box Plot 그래프로 나타났다. 본 논문에서는 클라우드 서버로 국내에 서버가 위치하는 AWS Lightsail 인스턴스를 사용하였다[16]. 애플리케이션과 RC 차량이 서버를 통해 통신할 때는 지연시간이 평균 152ms로 나타났으며, 직접 통신을 한 경우에는 지연시간이 평균 41ms로 개선된 것을 볼 수 있다.

3.2 2D 조이스틱 기반 이륜차량 제어 알고리즘

테스트베드를 구성하는 애플리케이션은 2D 조이스틱으로 RC 차량을 제어한다. 조이스틱의 핸들은 -1과 1 사이의 좌표값을 가지며, 모터는 -255와 255의 속도값을 가진다. 단순하게 각 좌표를 각 모터의 회전 속도에 선형적으로 대응하면 해결될 수 있어 보인다. 하지만 RC 차량은 좌우에 하나씩의 모터가 장착된 이륜 차량이므로, 2D 조이스틱의 핸들 위치 좌표와 RC 차량의 좌우 모터의 회전 속도는 간단하게 대응되지 않는다. 본 논문에서는 2D 조이스틱으로 좌우 이륜 차량을 제어할 때 사용자가 차량의 이

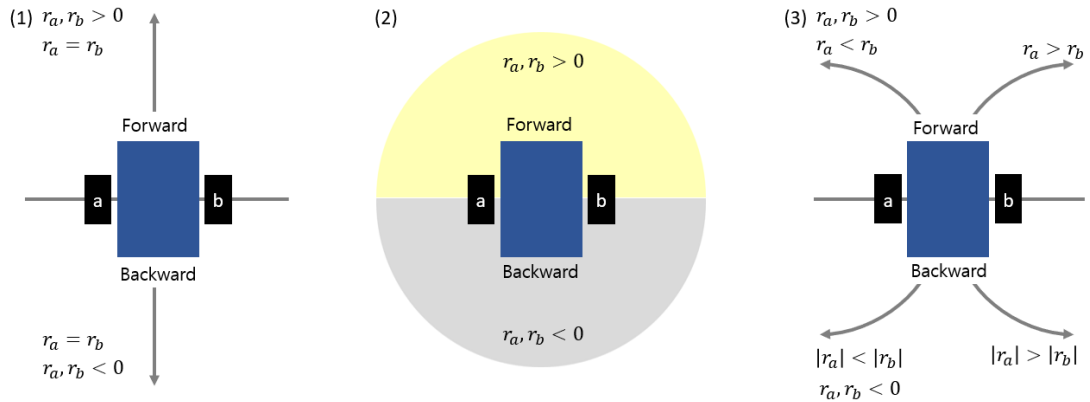


Fig. 5. Considerations for mapping 2D joysticks to left and right two-wheeled vehicles.

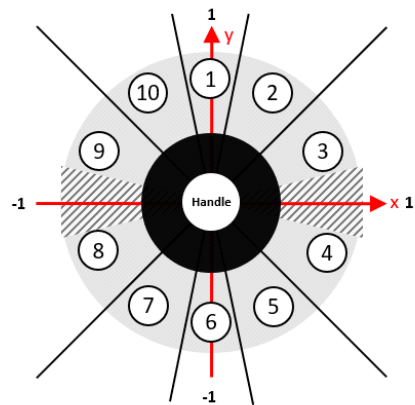
동을 직관적으로 예상할 수 있고 좀 더 정교한 제어가 가능하게 하는 조이스틱과 좌우 모터의 매핑 알고리즘을 제안한다.

조이스틱 위치 좌표와 좌우 모터 회전 속도를 매핑하기 위해 Fig. 5와 같이 3가지 사항을 고려해야 한다. Fig. 5의 r_a 는 좌측 모터(a)의 회전 속도이고 r_b 는 우측 모터(b)의 회전 속도이다. 편의상 각 모터의 회전 속도는 전진 방향일 때를 양수로 간주한다. 첫째, 똑바로 전진하는 경우 r_a, r_b 는 동일해야 하며 양수 값을 가진다. 똑바로 후진하는 경우에는 r_a, r_b 는 동일해야 하며 음수 값을 가진다. 둘째, RC 차량이 전방으로 주행할 경우 r_a, r_b 는 둘 다 양수 값, 후방으로 주행할 경우 r_a, r_b 는 둘 다 음수 값을 가진다. 즉, 보다 직관적인 제어를 위해, 두 모터가 서로 반대 방향으로 회전하는 경우는 없다고 가정한다. 셋째, RC 차량을 회전시키는 경우, 회전 방향의 반대쪽 모터가 더 빠르게 회전해야 한다. 예를 들어 RC 차량을 우측 전방으로 제어한다면 r_a 는 r_b 보다 커야 한다. 이때 r_a, r_b 는 전진하며 회전하는 경우 양수 값을, 후진하며 회전하는 경우에는 음수 값을 가진다.

매핑 알고리즘은 사용자가 차량의 제어 결과를 직관적으로 인지할 수 있도록 상기 3가지 사항을 고려하여 조이스틱 핸들 좌표를 좌우 모터 회전 속도로 변환하는 방법을 제시한다. 조이스틱은 Fig. 6에 나타나 있으며, 핸들의 위치 좌표는 2D 좌표로 나타난다. 편의상 가로축을 x축, 세로축을 y축이라 한다. 핸들의 위치 좌표를 각 모터의 회전 속도로 변환하는 경우, 얼핏 보기에 핸들의 y 좌표는 RC 차량의 속도 제어를 위한 값으로 대응될 수 있다고도 볼 수 있

며, 좌우 모터 회전 속도에 각각 대입할 수 있다. 이때 핸들의 위치가 x축에 가까이 위치하면 y 좌표는 0에 가까운 값을 가지는 문제점이 발생한다. 즉, 차량의 속도는 핸들의 y좌표값이 아니라 핸들의 위치벡터의 크기로 정의되어야 한다.

RC 차량의 방향은 좌우 모터의 회전 속도 차이로 결정되므로 핸들 방향에 따라 좌우 모터의 회전 속도를 지정해야 한다. 여기서 x 좌표는 좌우 모터의 회전 속도 차이를 나타내기 위한 값이 될 수 있다. RC 차량을 제어하는 방향에 따라 좌우 모터의 회전 속도 차이를 두어야 하며 Fig. 5의 고려사항에 따라 더 빠르거나 느리게 회전하는 모터가 결정된다. 알고리즘의 복잡도를 완화하기 위해 핸들이 위치할 수 있는 구역을 Fig. 6과 같이 나누었으며 구역별로 핸들의 x, y 좌표를 기반으로 r_a, r_b 을 연산한다.



Joystick handle movable area
Fig. 6. Coordinate plane of joystick control.

Fig. 6에서, 1, 6구역은 똑바로 직진 또는 후진하는 경우에 해당하며, 사용자 제어 오차를 감안하여 일정 각도 이하에서는 똑바로 직진 또는 후진한다고 간주하였다. 즉, 해당 구역 내에서는 r_a 와 r_b 가 동일한 값을 가진다.

Fig. 6의 2, 3구역은 차량이 전진하면서 우측으로 회전해야 하므로, r_a 와 r_b 가 0보다 크고 r_a 가 r_b 보다 큰 값을 가져야 한다. 여기서는 차량이 최대 속도로 이동할 때, 즉, 핸들이 이동 가능 구역의 테두리에 위치할 때 r_a 와 r_b 를 결정하는 방법을 예시를 들어 설명한다. 이때 핸들의 벡터 각 θ 를 활용하는데, 이는 양의 x축을 기준으로 현재 핸들의 위치벡터까지의 라디안 각을 의미한다. 우선 핸들이 2사분면에 위치하는 경우, 핸들의 벡터 각 θ 가 0.45π 이상이면 1구역에 위치하므로 회전 속도 차이가 0이다. θ 가 0.05π 이하면, 핸들이 서로 반대 방향으로 이동하는 구역 3과 4 사이를 쉽게 이동할 수 있으므로 완충 효과를 위해 회전 속도 차이를 더 증가시키지 않는다. 즉, 회전 속도 차이는 $\theta \geq 0.45\pi$ 일 때 최소이며, $\theta \leq 0.05\pi$ 일 때 최대값을 가진다.

우측 방향으로 회전하는 경우, 우측 모터의 회전 속도를 결정하기 위해서는 몇 가지 회전 방식을 고려해야 한다. 즉, 회전 결과가 사용자에게 직관적으로 받아들여지려면 몇 가지 방식 중 하나를 선택해야 한다. 만약 핸들이 $\theta = 0.05\pi$ 인 (0.9848, 0.1736)에 위치한다고 할 때, 차량은 첫째, 좌우 모터는 최고속도로 서로 반대로 회전하여 제자리에서 회전하거나, 둘

째, 좌측 모터는 최고속도로 회전하고 우측 모터는 정지하여 우측 모터를 기준으로 회전하거나, 셋째, 좌측 모터는 최고속도로 회전하고 우측 모터는 일정한 속도로 순방향 회전하여 전진하면서 회전하는 경우가 있을 수 있다. 본 논문에서는 제어 결과의 직관성을 위해 세 번째 방식을 채택하였으며 우측 모터의 최소 회전 속도는 마찰력을 기반으로 계산된 80으로 간주하였다[14].

최대 속도를 가정한다면, 즉, 핸들의 위치벡터 크기가 1이고 벡터 각 $\theta = 0.45\pi$ 일 때의 회전 속도 차이는 0이며 $\theta = 0.05\pi$ 일 때의 회전 속도 차이는 175가 될 수 있다. 회전 속도 차이가 θ 에 선형적으로 대응된다고 단순화하면 우측 모터의 속도는 $r_b \leftarrow r_a - ((0.45\pi - \theta) \times 175 / 0.4\pi)$ 로 구할 수 있다. 이와 유사한 방식으로 다른 사분면 구역도 핸들의 위치벡터 크기와 벡터 각에 따라 좌우 모터의 회전 속도를 결정할 수 있다. 이를 통해 사용자는 RC 차량의 속도 및 방향을 정교하게 제어할 수 있으며 직관적인 경험을 할 수 있다.

매핑 알고리즘의 흐름도는 Fig. 7과 같다. 조이스틱 핸들의 위치 좌표에 따라 구역을 구분한 후, 해당 구역에 적절한 변환을 수행한다. 우선 핸들의 위치벡터가 x축과 이루는 각도가 0.45π 에서 0.55π 사이, 1.45π 에서 1.55π 사이인 경우에는 1, 6구역에 해당하며 r_a 와 r_b 를 동일한 값으로 설정한다. 그 외의 경우, 벡터 각이 0에서 0.45π 사이인 경우에는 2, 3구역에 해당하며, r_a 는 r_b 보다 큰 값을 가져야 한다. 이때 좌우 모터 회전 속도는 핸들의 위치벡터에 따른 차량

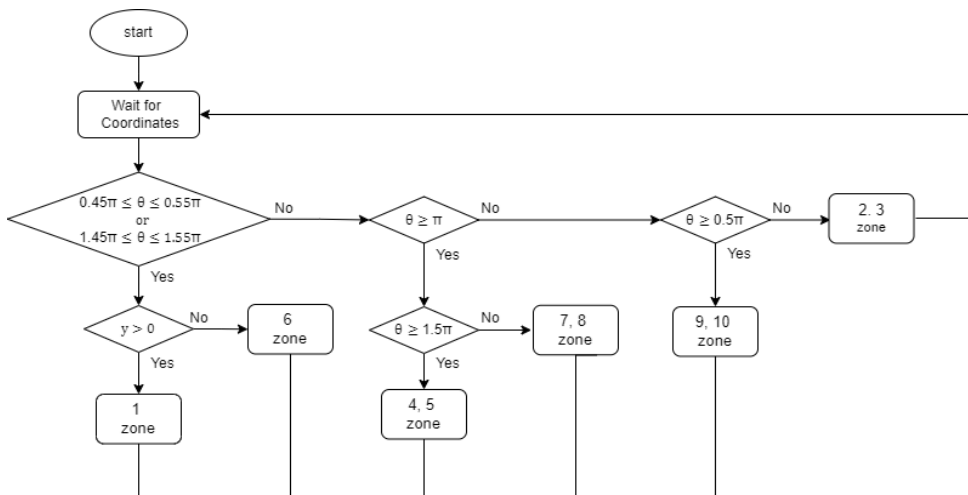


Fig. 7. Flow chart of mapping algorithm for 2D joystick and left and right two-wheeled vehicles.

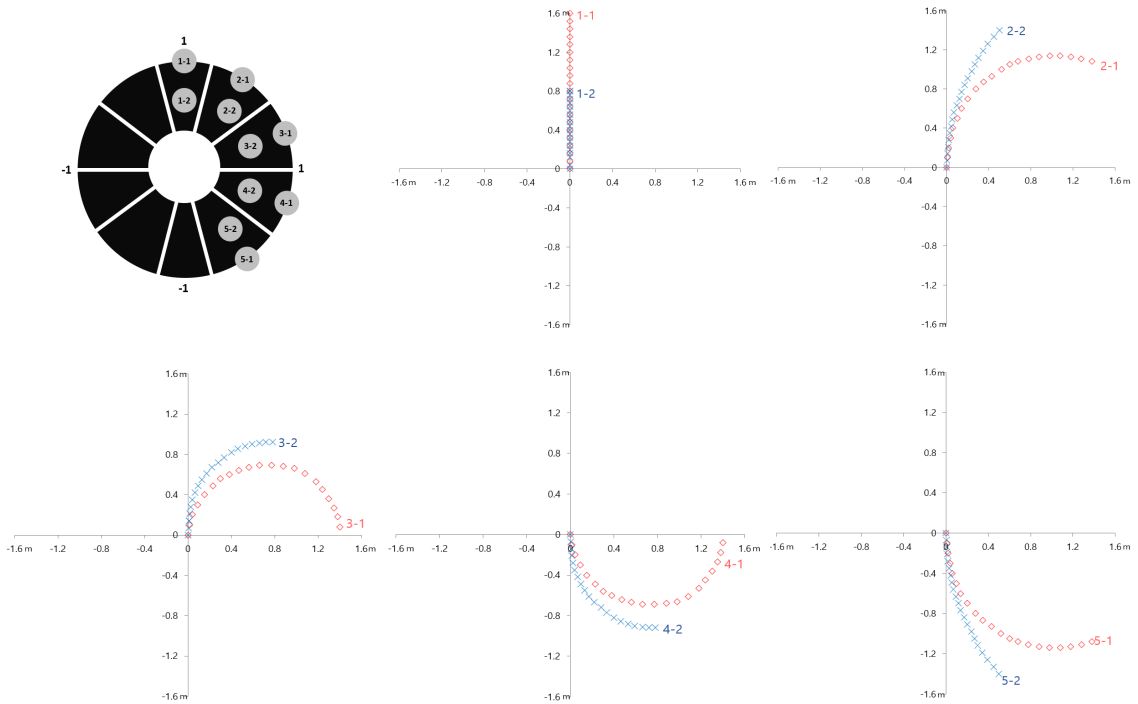


Fig. 8. The RC vehicle's movement path according to the joystick handle.

의 속도를 먼저 구한 후, 벡터 각에 따라 우측 모터의 속도 r_b 를 감소시킨다. 나머지 구역도 이와 유사하게 동작한다.

매핑 알고리즘을 검증하기 위한 RC 차량 주행 실험을 진행하였다. 실험을 위해 Fig. 8과 같이 RC 차량을 제어하는 방향과 속도를 고려하여 총 10개의 좌표에 핸들이 위치하는 경우로 분류하였다. 실험은 가로, 세로 3.2 m 공간의 중심 지점에 RC 차량을 배치하여 4초 동안 차량을 제어함으로써 수행되었다. 제어하는 동안 RC 차량의 위치를 0.2초 간격으로 측정하여 좌표 평면상에 RC 차량의 경로를 나타냈다. RC 차량은 1번 구역 직진, 2, 3번 구역 우측 전방, 4, 5번 구역 우측 후방으로 각각 주행하였으며 핸들 위치벡터 크기에 의해 계산된 속도에 따라 경로가 달라진 것을 볼 수 있다. 이를 통해 조이스틱과 좌우 모터의 매핑 알고리즘을 통해 사용자가 RC 차량의 속도 및 방향을 정교하고 직관적으로 제어할 수 있음을 확인할 수 있었다.

3.3 원격제어 및 실시간 모니터링 애플리케이션

가상공간의 시각화를 위해 Unity를 활용하여 가

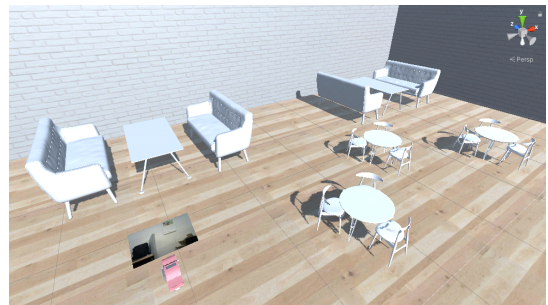


Fig. 9. A virtual space built with game engines.

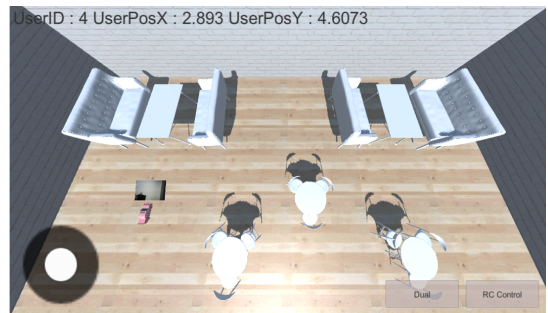


Fig. 10. A application implemented through game engines.

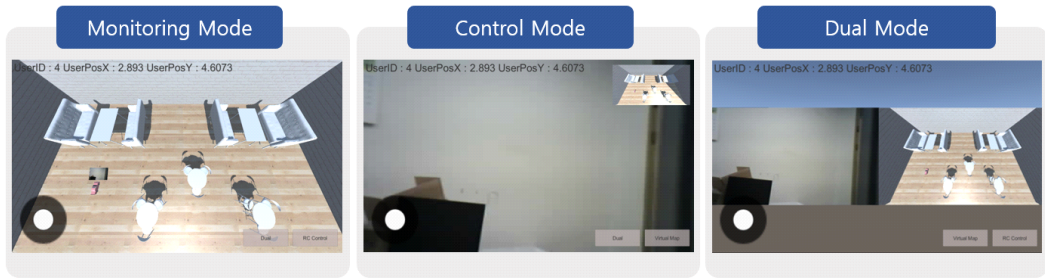


Fig. 11. Three modes of application.

상공간을 구성하였다. 실제 환경을 가상공간에 재현하기 위해 Fig. 9와 같이 RC 차량, 영상 스트리밍 화면, 의자, 소파, 테이블을 배치하였다. Fig. 10과 같이 조이스틱 인터페이스, 모드 변환 버튼, RC 차량의 ID와 실시간 위치 좌표를 애플리케이션의 UI로 구성하였다. 상단의 위치 좌표는 서버로부터 받은 차량의 좌표가 표시되며 0.1초마다 갱신된다.

애플리케이션은 RC 차량과 직접 통신하기 위해 우선 서버로부터 RC 차량의 IP와 포트 번호와 같은 차량 연결정보를 JSON 포맷으로 전달받는다. 해당 연결정보를 기반으로 RC 차량과 WebSocket 연결을 수립한 후 직접 통신을 수행한다. 직접 통신은 양방향으로 수행되는데, 애플리케이션에선 조이스틱의 핸들 좌표를 기반으로 좌우 모터 회전 속도를 전송하여 RC 차량을 제어한다. RC 차량은 부착된 카메라의 영상을 실시간으로 스트리밍하며 애플리케이션은 수신한 영상 스트림을 가상 RC 차량 상단에 재생하여 실제 환경을 관찰할 수 있도록 구현하였다.

실제 RC 차량과 가상 RC 차량의 위치 동기화를 위해 WebSocket을 기반으로 현재 위치 좌표를 전달받는다. JSON 포맷으로 전달받은 위치 좌표는 애플리케이션에서 활용할 수 있도록 C# 객체로 변환하여 사용하였다. 실시간 모니터링을 위해 가상 RC 차량은 수신한 위치 좌표로 이동해야 한다. 하지만 실제 차량의 위치 좌표는 0.1초마다 수신되므로 선형 보간을 적용하여 차량을 부드럽게 이동시켰다.

Fig. 11은 애플리케이션의 모드를 나열한 것으로 제어 목적에 따라 3가지 모드를 제공한다. 모니터링 모드는 사용자가 RC 차량의 위치를 실시간으로 모니터링하는 데에 목적을 두며, 실제 환경을 가상공간에 재현함과 동시에 RC 차량의 카메라 영상이 차량 상단에 부분 화면으로 구성된다. 제어 모드는 카메라 영상을 전체화면으로 구성하였으므로 사용자는 RC

차량을 제어하면서 실제 환경을 자세히 관찰할 수 있다. 우측 상단에 가상공간이 미니 맵 형태로 구성되어 있어 RC 차량의 위치를 대략적으로 파악할 수 있다. 듀얼 모드는 실제 환경과 가상공간을 동시에 모니터링하기 위한 화면으로 위의 2가지 모드를 혼합하여 구성한 모드이다. 모든 모드는 화면의 좌측 하단 버튼을 통해 다른 모드로의 변경이 가능하다.

4. 결 론

본 논문에서는 디지털 트윈 구축에 요구되는 핵심 기술을 기반으로 원격제어 및 모니터링 시스템을 구현하였다. 범용 게임엔진을 기반으로 실제 환경을 가상공간에 시각화하였으며 실시간 모니터링을 위해 가상공간 차량은 실제 RC 차량과 위치 동기화되어 동일한 좌표에 위치하도록 구현하였다. 제안된 시스템은 가상공간과 실제 환경 간 다양한 정보를 서버 기반 및 직접 통신으로 주고받을 수 있도록 구현되었다. 애플리케이션에서 저 지연으로 원격지 RC 차량을 제어하기 위한 중계 프로토콜을 구현하였으며, RC 차량의 속도 및 방향을 정교하고 직관적으로 제어하기 위해 2D 조이스틱 핸들 위치를 RC 차량의 좌우 모터 회전 속도로 변환하는 알고리즘을 구현하였다.

본 논문에서 제시한 디지털 트윈 구현은 디지털 트윈 시스템을 구축하는데 요구되는 핵심 기술들을 저비용으로 단시간에 구현한 것에 의의가 있다. 여기에는 게임엔진 기반 가상공간의 시각화, 가상공간과 실제 환경의 동기화, 센싱 데이터 수집 및 관리, 저지연 원격제어 및 모니터링 애플리케이션이 포함된다. 디지털 트윈 시스템을 모니터링하고 제어하기 위한 직관적이고 사용성이 높은 애플리케이션을 구현하였으며, 수집된 데이터를 기반으로 시뮬레이션, 미

래 예측 등을 수행할 수 있는 소프트웨어로 사용될 수 있다.

향후 우리는 이와 같은 원격제어 및 모니터링 디지털 트윈 시스템의 구현을 지속하여, 데이터베이스에 축적된 정보를 활용하는 시뮬레이션 시스템을 설계할 것이며 해당 시스템을 가상공간에 시각화하기 위한 연구를 지속적으로 진행할 예정이다.

REFERENCE

- [1] T.H. Ko, S.B. Noe, D.H. Noh, J.H. Choi, and T.B. Lim, "Study of Implementation as Digital Twin Framework for Vertical Smart 1 Far," *Journal Of Broadcast Engineering*, Vol. 26, No. 4, pp. 377-389, 2021.
- [2] D. Lee, S. Song, C. Lee, S.D. Noh, S. Yun, and H. Lee, "Development and Application of Digital Twin for the Design Verification and Operation Management of Automated Material Handling Systems," *Korean Journal of Computational Design and Engineering*, Vol. 26, No. 4, pp. 313-323, 2021.
- [3] J. Kim, W. Choi, M. Song, and S. Lee, "Design and Implementation of IoT Platform-based Digital Twin Prototype." *Journal of Broadcast Engineering*, Vol. 26, No. 4, pp. 356-367, 2021.
- [4] Y. Park, H.S. Kang, J. Kim, S. Choi, and J.Y. Lee. "Digital Twin Testbed in Cyber Physical Systems Towards Smart Manufacturing of Small and Medium-sized Manufacturers," *Korean Journal of Computational Design and Engineering*, Vol. 25, No. 3, pp. 298-306, 2020.
- [5] J. Choi, J. Yang, J. Lym, S.D. Noh, S.H. Lee, J.T. Kang et al, "Development of a CPS Platform and Digital Twin Applications for Production Lines of Automotive Parts," *Korean Journal of Computational Design and Engineering*, Vol. 26, No. 4, pp. 285-298, 2021.
- [6] J. Shin, S. Lee, and S. Lee, "Development of 3-dimensional Solar Photovoltaic Simulation System based on Digital Twin in the Smart City," *The transactions of The Korean Institute of Electrical Engineers*, Vol. 69, No. 12, pp. 1834-1841, 2020.
- [7] J. Ryu, "Proposal of Essential Components for Construction of Architectural Digital Twin Using Game Engine," *Proceeding of Annual Conference of the Architectural Institute of Korea*, Vol. 41, No. 1, pp. 518-519, 2021.
- [8] Y. Nam, S. Lee, D. Lee, S. Im, and S.D. Noh, "Digital Twin-based Application for Design of Human-Machine Collaborative Assembly Production Lines," *Journal of the Korean Institute of Industrial Engineer*, Vol. 46, No. 1, pp. 042-054, 2020.
- [9] S. Park, J. Yang, H. Kim, M. Cho, and Y. Kwon, "Safety Design of Smart Children's Pedestrian Space Based on Digital Twin Using Unreal Engine - Focused on the elementary school located in Uijeongbu -," *Proceeding of Annual Conference of the Architectural Institute of Korea*, Vol. 41, No. 1, pp. 651-652, 2021.
- [10] M. Lee, Y. Lee, and J. Kim, "Raspberry-based Multi-function RC Car Controller," *The Journal of The Institute of Internet Broadcasting and Communication*, Vol. 18, No. 2, pp. 149-156, 2018.
- [11] Y. Shin, K. Kim, B. Kim, S. Park, and Y. Lee, "RC Car for Shooting Using a Machine Wheel," *Proceedings of KIIT Conference*, pp. 379-380, 2019.
- [12] S. Hong, E. Lee, S. Jang, C. Bae, H. Shin, Y. Kang et al, "Development of Speech Recognition Radio Control Car," *Proceedings of the Korean Information Science Society Conference*, pp. 1999-2001, 2018.
- [13] M. Shin, S. Lim, and D. Kim, "Ultrasonic Sensor-based Radio Controlled RC Car Using Search Algorithm," *The Journal of Korean Institute of Information Technology*, Vol. 18, No. 3, pp. 49-57, 2020.
- [14] D. Yu and W. Kim, "Implementation of UWB Indoor Positioning and Real-time Remote Control System for Disaster Monitoring based on DigitalTwin," *Journal of Korea Multime-*

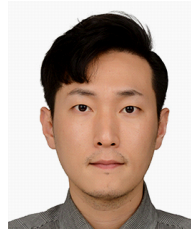
dia Society, Vol. 24, No. 12, pp. 1682-1692, 2021.

- [15] W. Kim, "Edge Computing Server Deployment Technique for Cloud VR-based Multi-User Metaverse Content," *Journal of Korea Multimedia Society*, Vol. 24, No. 8, pp. 1090-1100, 2021.
- [16] Introduction to AWS(2022), <http://aws.amazon.com/about-aws/> (accessed January 28, 2022)



윤 정 은

2019년~현재 국립안동대학교 멀티미디어공학과 학사 과정 재학
관심분야: VR/AR, 디지털트윈, 메타버스



김 원 석

2010년 부산대학교 컴퓨터공학과 졸업(공학사)
2017년 부산대학교 컴퓨터공학과 졸업(공학박사)
2016년~2019년 (주)플론베이 대표이사

2019년~2020년 영산대학교 문화콘텐츠학부 조교수
2020년~현재 국립안동대학교 멀티미디어공학과 조교수
관심분야: 컴퓨터네트워크, 엣지컴퓨팅, VR/AR, 디지털트윈, 메타버스