

## 이기종 디바이스 지원을 위한 크로스 플랫폼 기반 XR 콘텐츠 개발

장희진<sup>1</sup> · 이성화<sup>1</sup> · 신정훈<sup>2</sup> · 이승민<sup>1</sup> · 김민아<sup>2</sup> · 남덕윤<sup>3\*</sup>

<sup>1</sup>한국과학기술정보연구원 오픈XR플랫폼융합연구단 선임연구원

<sup>2</sup>한국과학기술정보연구원 오픈XR플랫폼융합연구단 책임연구원

<sup>3\*</sup>경북대학교 컴퓨터학부 조교수

## Development of Cross-platform XR Content Supporting Heterogeneous Devices

Heejin Jang<sup>1</sup> · Sunghye Lee<sup>1</sup> · Jung Hun Shin<sup>2</sup> · Seungmin Lee<sup>1</sup> · Min Ah Kim<sup>2</sup> · Dukyun Nam<sup>3\*</sup>

<sup>1</sup>Senior Researcher, Korea Institute of Science and Technology Information, Daejeon 34141, Korea

<sup>2</sup>Principal Researcher, Korea Institute of Science and Technology Information, Daejeon 34141, Korea

<sup>3\*</sup>Assistant Professor, School of Computer Science and Engineering, Kyungpook National University, Daegu 41566, Korea

### [요약]

확장 현실(XR) 기술 분야에서 다자간 협업을 위한 이기종 XR 디바이스 간 호환성 및 상호운용성 지원 문제는 여전히 도전적인 과제이다. 이기종 사용자 간 상호작용을 지원하기 위해 XR 콘텐츠 개발 시 XR 디바이스들의 다양한 입력 장치와 플랫폼을 고려해야 하지만, 이를 위한 크로스 플랫폼 설계 및 XR 콘텐츠 개발에 관한 모범 사례는 찾아보기 어렵다. 본 논문에서는 OpenXR 표준과 XR 개발 도구들을 활용해 크로스 플랫폼 개발 환경을 구축하고 4종의 이기종 디바이스를 이용해 협업을 지원하는 XR 콘텐츠를 개발하였다. 또한 XR 콘텐츠와 MRTK 기반의 크로스 플랫폼 개발에 대해 소개하고, 실험을 통해 이기종 XR 디바이스 사용자들이 손동작, 터치스크린 등 다양한 입력 도구들을 활용하여 가상 객체 및 다른 사용자와 상호작용할 수 있음을 실증하였다.

### [Abstract]

Supporting multi-user collaboration across diverse heterogeneous XR devices remains challenging in extended reality technology, primarily because of compatibility and interoperability issues. XR content creators are tasked with considering various input devices and platforms across XR devices to facilitate user interactions. However, it is difficult to determine the practices for cross-platform design and XR content development. In this paper, we set up a cross-platform development environment based on OpenXR and XR development tools and develop XR collaboration content designed to support interactions across four different XR devices. Furthermore, this paper introduces XR content and discusses cross-platform development based on the Mixed Reality Toolkit. The experiment empirically demonstrates that users of diverse XR devices can interact with virtual objects and other users using various input methods such as gestures and touchscreens.

**색인어** : XR 콘텐츠, 크로스 플랫폼, 이기종 디바이스, 오픈 XR, 다중 사용자 협업

**Keyword** : XR Contents, Cross Platform, Heterogeneous Devices, OpenXR, Multi-User Collaboration

<http://dx.doi.org/10.9728/dcs.2023.24.10.2533>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

**Received** 30 August 2023; **Revised** 04 October 2023

**Accepted** 11 October 2023

**\*Corresponding Author; Dukyun Nam**

**Tel:** +82-53-950-6371

**E-mail:** dynam@knu.ac.kr

## 1. 서론

가상 현실(AR; augmented reality)과 증강 현실 (VR; virtual reality)은 컴퓨터로 생성된 가상 콘텐츠를 활용하여 새로운 체험을 제공하는 기술이다. COVID-19 이후 비대면 환경이 확대되면서 원격 협업, 의료, 교육 등의 수요가 늘어남에 따라 다양한 AR/VR 디바이스도 시장에 출시되고 있다. 최근에는 광학적 투시 디스플레이를 탑재한 HoloLens와 같은 디바이스로 인해 AR과 VR 개념이 혼합된 혼합 현실(MR; mixed reality) 기술이 큰 관심을 받고 있으며[1], AR, VR 및 MR 기술을 통칭하는 확장 현실(XR; eXtended Reality) 기술은 사용자에게 가상과 현실의 경계를 넘나드는 새로운 차원의 경험을 제공한다[2].

XR 디바이스의 보급이 확대됨에 따라 다양한 이기종 디바이스를 지원하는 XR 콘텐츠의 필요성이 점차 부각되고 있다. 기존 연구는 주로 동일한 디바이스를 사용하는 다중 사용자 간 협업에 중점을 두었으나[3],[4], 사용자들은 XR 디바이스의 가격과 협업 시 자신의 역할을 고려하여 각자에게 적합한 디바이스를 선택할 수 있어야 한다. 하지만 이기종 XR 디바이스를 활용해 다자간 협업을 가능하게 하려면 XR 콘텐츠 개발 과정에서 여러 가지 어려움이 존재한다[5].

기존의 응용들은 특정 플랫폼에 종속되거나, 범용 플랫폼에서의 활용을 위해 접근성이 뛰어난 웹 기반 서비스로 개발되었다[6]. 그러나 XR 콘텐츠는 디바이스와 체험 장치를 이용해 사용자에게 직접적인 상호작용을 지원해야 하므로 웹 기반 구현은 한계가 있다. XR 콘텐츠가 이기종 디바이스를 지원하려면 특정 플랫폼에 종속되지 않고, 디바이스 간 상호 호환성을 고려하며 다양한 입력 방식을 지원해야 한다. 예를 들어, 가상 객체를 선택하기 위해 XR 디바이스에 따라 마우스, 손동작, 시선, 컨트롤러, 터치스크린, 음성 등 다양한 입력 방법이 존재한다. 이처럼 다양한 입력 방식과 이기종 디바이스 지원의 복잡성으로 실험 환경을 구축하기 어려워 기존 연구에서는 애플리케이션 환경에서 실험을 수행하였다[7],[8]. 또한, XR 콘텐츠 제작 시 호환성 지원을 위한 구현 설계와 OpenXR과 같은 공통의 표준을 개발에 적용하는 방법[9],[10], 크로스 플랫폼 개발을 위해 XR 디바이스 제조사별로 제공되는 각종 소프트웨어 개발 도구(SDK; Software Development Toolkit)들을 효율적으로 통합하는 방법에 대한 모범 사례(Best practice)를 찾아보기 어렵다[11],[12].

본 연구팀은 XR 콘텐츠를 개발하기에 앞서, XR 콘텐츠 제작에 활용할 수 있는 XR 개발 도구들과 OpenXR 표준에 대한 사전 조사를 수행하였다. 디바이스 제조사와 관련 소프트웨어 업체에서는 XR 개발 도구로 MRTK(Mixed Reality ToolKit)[13], AR Foundation[14]과 같은 크로스 플랫폼 개발을 위한 개발 도구들을 제공하고 있으며, 현재 단계는 크로스 플랫폼 개발을 위한 공통의 표준과 개발 도구가 확립되었다기보다는, 관련 기업들 간 경쟁과 협력을 통해 신제품을 출시하고 생태계 확대 및 표준 정립 후 개발 도구들이 제시되

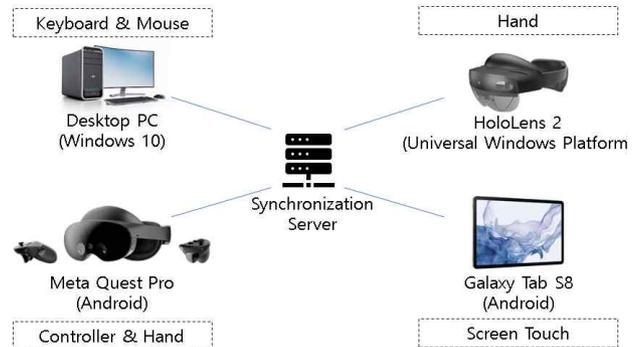


그림 1. 이기종 AR/VR 디바이스 및 사용자 입력 장치  
**Fig. 1. Heterogeneous AR/VR devices and user input method**

는 과도기적인 단계로 판단된다.

OpenXR 표준[15]은 XR 디바이스 보급이 확대될수록 이기종 디바이스를 지원하기 위한 표준화된 프로토콜과 인터페이스에 대한 필요성이 더욱 증가할 것으로 예상됨에 따라, 산업계 컨소시엄인 크로노스 그룹 (Khronos Group)의 주도하에 제정되었다. OpenXR 표준은 XR 콘텐츠가 다양한 플랫폼과 XR 디바이스에서 실행되도록 크로스 플랫폼 개발을 지원하기 위한 개방형 API 내용으로, 표준화된 API 제공과 함께 하드웨어에 독립적인 추상화된 액션 (Action) 기반의 구현을 지원함으로써 개발의 편의성과 효율성을 향상시킨다.

본 논문에서는 현시점에서 활용 가능한 각종 XR 개발 도구와 OpenXR을 활용해 4종의 XR 디바이스를 지원하는 XR 콘텐츠를 개발하여, 이기종 XR 디바이스의 활용 및 다양한 입력 장치를 이용한 사용자 상호작용을 실증하였다. 본 연구에서 활용한 디바이스는 HoloLens 2 (UWP; Universal Windows Platform), Meta Quest Pro (Android), Galaxy Tab S8, (Android), Desktop PC (Windows 10)이며 그림 1에 나타나 있다. XR 콘텐츠는 여러 명의 사용자가 XR 기반의 협업 공간에서 가상 객체인 3D 드론을 공동으로 조립하는 크로스 플랫폼 응용으로, 사용자는 이기종 XR 디바이스를 활용하여 XR 협업에 참여한 뒤 가상 환경 및 타 사용자와의 상호작용을 통해 조립 작업을 수행할 수 있다. 콘텐츠 개발에는 유니티와 MRTK를 활용하였으며, OpenXR을 기반으로 이기종 XR 디바이스와 다양한 사용자 입력 장치를 연동하였다. 또한, XR 콘텐츠의 크로스 플랫폼 지원을 실증하기 위해 3명의 AR 디바이스 사용자와 2명의 VR 디바이스 사용자로 구성된 물리적 협업 공간을 각각 구성하고 두 공간을 원격으로 연결해 원격 기반의 XR 협업 환경을 구축하여 실험을 진행하였다. 실험을 통해 구현된 XR 콘텐츠가 Android, Windows, UWP 3종의 플랫폼으로 구성된 4종의 XR 디바이스에서 정상적으로 구동되고, 각기 다른 사용자 입력 장치를 이용해 가상 객체와의 상호작용을 성공적으로 지원함을 확인할 수 있었다.

본 논문은 다음과 같은 구성으로 이루어져 있다. 2장에서는 XR 개발 도구와 OpenXR 표준에 대한 현황을 소개하고, 3장에서 MRTK와 OpenXR을 활용한 크로스 플랫폼 XR 콘텐츠

크 개발 방법에 관해 기술한다. 4장에서는 4종 XR 디바이스와 XR 콘텐츠를 이용한 실험에 관해 설명한 후, 5장에서 본 연구의 결론 및 향후 연구계획으로 마무리한다.

## II. 관련 연구

이기종 XR 디바이스를 지원하는 XR 콘텐츠는 다양한 플랫폼과 입력 장치를 지원하며 XR 디바이스 간 상호 호환성을 보장해야 한다. 본 절에서는 주요 플랫폼별 XR 개발 도구와 크로스 플랫폼 개발을 위해 제안된 XR 개발 도구에 관해 소개하고, 크로스 플랫폼 및 다수의 인터페이스를 지원하기 위해 제정된 OpenXR 표준에 대해 설명한다.

### 2-1 XR 콘텐츠 개발 도구

XR 디바이스와 플랫폼을 지원하기 위해 다양한 SDK가 존재한다. 표 1은 XR 콘텐츠 개발을 위한 주요 플랫폼 및 개발 도구, OpenXR 지원 여부를 보여준다.

**표 1.** 플랫폼에 따른 XR 개발 도구 및 OpenXR 지원 여부  
**Table 1.** XR development tools and OpenXR support by platform

	iOS	Android		UWP
Plugin	ARKit	ARCore	Oculus XR Plugin & Oculus Integration SDK MR OpenXR Plugin (Unity OpenXR Plugin)	MR OpenXR Plugin (Unity OpenXR Plugin)
Open XR	×	×	○	○
Device	iPhone iPad	Android Phone/ Tablet	Oculus Rift, Oculus Quest, Meta Quest Pro	HoloLens 2, HP Reverb

#### 1) ARCore

ARCore[16]는 구글이 개발한 안드로이드 기반의 AR 개발 도구이다. 안드로이드 7.0 이상의 디바이스를 대상으로 내장된 위치 추적 센서와 카메라를 이용해 디바이스의 실시간 위치 및 사용자 모션 추적 등의 주요 기능을 제공한다. 또한, 얼굴 추적, 평면 및 포인트 클라우드 탐지, 가상 객체와 현실 공간을 정합하기 위한 앵커 기능 등의 추가적 기능을 제공한다. ARCore는 OpenXR을 지원하지 않으며 안드로이드 디바이스에서 OpenXR 지원을 위한 Monado[17]와 같은 별도의 프로젝트가 진행 중이다.

#### 2) ARKit

ARKit[18]은 애플이 개발한 iOS 기반의 증강 현실 개발 도구로서, ARCore와 같이 디바이스에 탑재된 센서와 카메라를 이용해 XR 콘텐츠 개발에 필요한 주요 기능을 제공한다. 특히

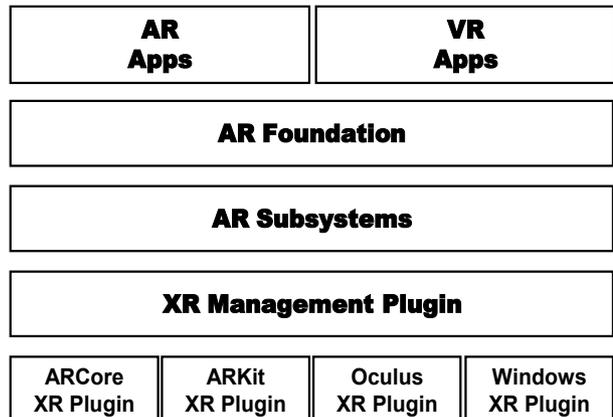
아이패드 프로 3세대 이후 탑재된 LiDAR(Light Detection and Ranging) 센서와 같은 강력한 하드웨어를 이용해 전신 추적, 객체 추적과 같이 ARCore가 제공하지 않은 고급 기능을 지원하며, 정확한 공간인식 기능과 정밀한 추적 성능을 제공하는 것으로 알려져 있다. ARCore와 마찬가지로 OpenXR을 지원하지 않으며, 자사의 생태계와 독자적인 기술을 선호하는 애플의 정책에 따라 ARKit을 중심으로 자체적인 iOS 생태계를 구축하리라 예상된다.

#### 3) Oculus Integration SDK

Oculus Integration SDK[19]는 메타가 제조하는 Oculus 디바이스를 위한 개발 도구이다. 메타는 초기부터 OpenXR 표준 개발에 적극적으로 참여하였으며, Oculus 디바이스가 안드로이드 기반이지만 자체적으로 개발한 Oculus XR Plugin[20]을 통해 OpenXR 기능을 지원한다. 2022년부터는 Oculus XR Plugin 대신 유니티 공식 OpenXR Plugin[21]을 백엔드로 활용하는 마이크로소프트의 Mixed Reality OpenXR Plugin을 이용해 OpenXR 기능을 지원한다.

#### 4) AR Foundation

AR Foundation은 유니티가 제공하는 크로스 플랫폼 AR 개발 프레임워크로서, 유니티 엔진은 XR 콘텐츠 개발을 지원하기 위해 내부적으로 그림 2와 같은 구조로 구성된다[22].



**그림 2.** 유니티 XR 콘텐츠 개발 환경  
**Fig. 2.** Development stack for Unity XR content

XR Plugin Management는 가장 저수준으로 동작하며 디바이스 제조사가 제공하는 ARCore, ARKit과 같은 특정 플랫폼에 대한 Plugin을 관리한다. 개발자는 다양한 Plugin을 쉽게 추가할 수 있으며 여러 XR Plugin을 동시에 사용하도록 설정함으로써 XR 콘텐츠를 복수의 플랫폼에서 구동시킬 수 있다. AR subsystem은 XR 디바이스와의 상호작용을 담당하는 핵심 계층으로서 XR Plugin Management를 통해 활성화된 특정 XR 디바이스와의 저수준에서 상호작용하며, 사용자 입력 처리, 위치 추적, 화면 렌더링 등 주요 XR 기능을 관리한다.

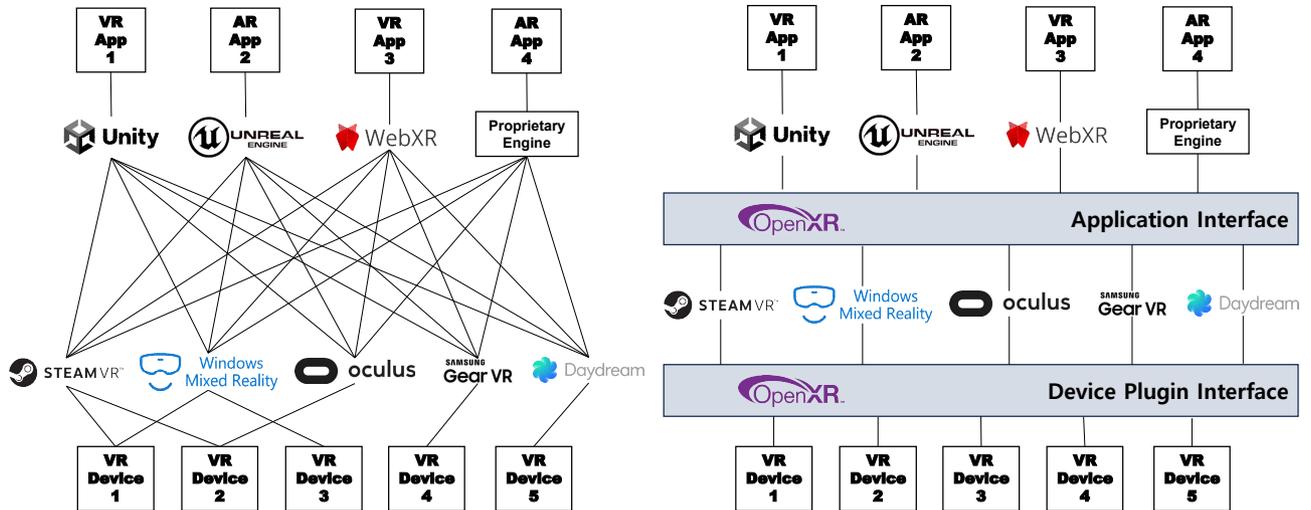


그림 3. OpenXR 활용하지 않은 구현 복잡도(좌), OpenXR 활용 구현 복잡도(우)  
 Fig. 3. Complexity of implementation with OpenXR(Left) and without OpenXR(Right)

다, 또한 XR SDK를 통해 각 기능들에 대한 추상화된 인터페이스를 상위 계층에 제공하여 표준화된 접근과 제어를 가능하게 한다.

가장 상위 계층인 AR Foundation은 이를 기반으로 XR 콘텐츠 개발에 필요한 공통된 기능을 고수준 API를 통해 개발자에게 제공함으로써, ARCore 및 ARKit 개발자가 일관된 방식으로 XR 콘텐츠를 개발할 수 있도록 한다.

### 5) Mixed Reality Toolkit

AR Foundation이 XR 콘텐츠 개발을 위한 추상화된 API를 제공하지만, ARCore와 ARKit을 중점적으로 지원하며 AR 기본 기능을 위주로 지원한다. 보다 폭넓은 XR 디바이스와 HMD (Head Mounted Display) 사용자의 신체 동작을 이용한 다양한 상호작용을 처리하려면 개발자가 추가로 직접 구현해야 한다. MRTK[13]는 마이크로소프트가 주도하는 플랫폼 중립적인 XR 개발 툴킷으로, AR Foundation의 기능을 보완하고 크로스 플랫폼 개발의 효율을 위해 아래와 같은 다양한 기능을 제공한다.

- 다관절 손동작, 음성, 6-DOF 컨트롤러 등 다양한 입력 장치를 이용한 상호작용을 지원하기 위해, GUI 기반의 MRTK 입력 시스템을 통해 액션 정의 및 다양한 입력 장치와의 매핑기능을 설정하기 위한 편리한 인터페이스를 제공한다.
- HMD 사용자를 위한 손 추적, 시선 추적, 광선을 이용한 원거리 상호작용, 음성 등의 고급 상호작용 기능을 지원하며, 주변 공간을 분석하고 사용자의 안전한 활동 영역을 보장하기 위한 경계 시스템 기능을 제공한다.
- MRTK에서 제공하는 Mixed Reality OpenXR Plugin은 Unity의 공식 OpenXR Plugin을 백엔드로 이용함으로써 OpenXR 런타임을 지원하는 여러 디바이스와

입력 장치를 통합적으로 지원한다. MRTK 입력 시스템 프로필 설정 정보는 Mixed Reality OpenXR Plugin Unity OpenXR Plugin을 통해 XR 디바이스의 OpenXR 런타임에 반영됨으로써 크로스 플랫폼 개발을 지원한다.

- 개발의 편의와 효율을 위해 진단 도구, 에디터 시뮬레이션, UI/UX 빌딩 블록, 기본 에셋 등을 제공한다.

본 연구에서는 MRTK를 활용하여 크로스 플랫폼을 지원하는 XR 콘텐츠를 개발하였다.

### 2-2 OpenXR 표준

OpenXR은 크로노스 그룹이 개발을 주도하며 XR 콘텐츠를 다양한 플랫폼과 디바이스에서 실행하기 위한 개방형 표준이다. 크로노스 그룹은 산업계 컨소시엄으로 VR/AR 디바이스 제조업체, 소프트웨어 개발 업체, 그래픽스 기술 업체 등 다양한 기업과 단체들이 OpenXR 표준에 참여하고 있으며, 2019년 7월 OpenXR 1.0[15]을 공개하였다.

XR 콘텐츠 개발을 위해 개발자들은 플랫폼과 디바이스에 따른 독자적인 API를 사용해 각각의 콘텐츠를 개발해야 하며, 유니티, 언리얼과 같은 게임 엔진에 따라서도 각기 다른 라이브러리를 활용해야 했다. 이러한 문제를 해결하고자 OpenXR은 통일된 API 표준을 제공함으로써 개발자들이 다양한 플랫폼에서 구동되는 단일 코드 베이스를 개발하고 실행할 수 있도록 지원한다. 그림 3은 게임 엔진 및 XR 디바이스에 의존적으로 콘텐츠를 개발하는 경우(좌)와 OpenXR 표준에 맞추어 개발하는 경우(우)의 복잡도를 도시하고 있다[15]. OpenXR 표준은 플랫폼에 독립적인 추상화된 API를 통해 응용 개발을 가능케 함으로써 개발 생산성을 높일 수 있을 뿐 아니라, 개발자들이 더욱 다양하고 안정적인 XR 콘텐츠를 개발·배포함으

로써 XR 산업의 성장을 촉진하고 사용자들에게 다양하고 일관된 경험을 제공할 수 있게 한다.

이를 위해 OpenXR은 이기종 디바이스를 이용하는 사용자의 상호작용을 “액션(Action)”을 기반으로 처리할 수 있도록 한다. 액션은 사용자와의 상호작용을 추상적으로 표현한 것으로, 특정한 손동작이나 버튼 클릭과 같은 사용자의 입력 동작을 일반화한 표현이다. 개발자들은 필요한 액션을 정의한 후, 이를 기반으로 XR 콘텐츠를 개발한다. XR 콘텐츠가 실행될 때, 해당 디바이스와 입출력 장치를 식별하고 동적으로 액션과 디바이스를 바인딩한다. 이로써, 개발된 XR 콘텐츠는 다양한 플랫폼을 지원하며 각종 입출력 장치와 유연하게 연동될 수 있다.

그림 4는 XR 콘텐츠 구현에서부터 실제 상호작용 이벤트가 발생하고 처리되기까지의 OpenXR에 따른 예시를 보여준다. 손으로 객체를 움켜질 때 색상을 변경하는 경우, XR 콘텐츠 구현 시 "Grip"이라는 액션을 생성한 후 해당 액션을 각각 Oculus 디바이스의 'A' 버튼 클릭 동작과 HoloLens 2의 움켜쥐는 손동작과 바인딩을 정의한다. 이후 런타임 시 실제 착용한 디바이스 및 입출력 장치가 동적으로 바인딩 되고, 콘텐츠 실행 중에 입력 장치에서 이벤트가 감지되면 바인딩 된 액션이 XR 콘텐츠로 전송되어 처리된다. 이처럼 OpenXR은 추상화된 액션과 특정 장치 간 동적 바인딩을 통해 응용 개발의 독립성을 보장하면서 다양한 상호작용 장치를 지원할 수 있다.

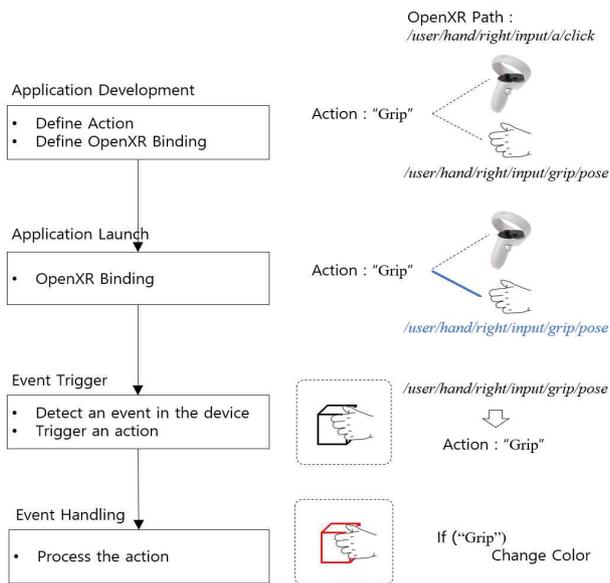


그림 4. OpenXR 기반의 다양한 사용자 상호작용 지원  
Fig. 4. Cross-platform interaction support based on OpenXR

### III. 크로스 플랫폼 지원 XR 콘텐츠 개발

본 연구에서는 MRTK 개발 도구와 OpenXR 소프트웨어를

활용해 4종 디바이스에서 구동되는 XR 콘텐츠를 개발하였다. 본 장에서는 크로스 플랫폼 개발 환경과 XR 콘텐츠의 구현에 관해 기술한다. 특히 추상화된 액션을 기반으로 4종 디바이스의 사용자 입력 장치를 이용한 상호작용을 지원하는 방법에 관해 설명하고, 액션을 이용해 새로운 사용자 입력 장치와 상호작용하는 구현 사례를 제시한다.

#### 3-1 OpenXR 기반의 크로스 플랫폼 개발 환경 구축

그림 5는 4종의 XR 디바이스를 대상으로 크로스 플랫폼 개발을 위해 구축된 유니티 기반의 개발 도구 스택을 보여준다. 구글, 메타, 마이크로소프트에서는 각 사의 디바이스 또는 플랫폼에 특화된 XR 개발 도구로서, 각각 ARCore Plugin, Oculus XR Plugin과 Oculus Integration SDK (SDK; software develop toolkit), Mixed Reality OpenXR Plugin과 MRTK를 제공한다. 2장에서 언급한 바와 같이 Oculus 디바이스의 경우 2022년부터 유니티의 OpenXR Plugin을 백엔드로 이용하는 Mixed Reality OpenXR Plugin을 지원하기 시작했으며 그림 5는 두 방법을 모두 포함하여 도시하고 있다.

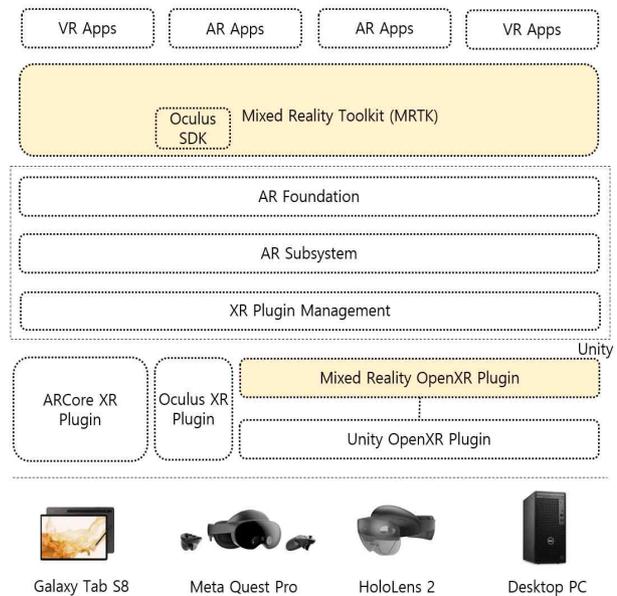


그림 5. 유니티 기반의 크로스 플랫폼 XR 콘텐츠 개발 환경  
Fig. 5. Cross-platform development stack for XR content

가장 하위계층에 Meta Quest Pro, Galaxy Tab S8, HoloLens 2 및 Desktop PC를 위한 Oculus XR Plugin, ARCore Plugin, Mixed Reality OpenXR Plugin 패키지와 이들을 통합해 관리하기 위한 XR Plugin Management 패키지를 설치하였다. 표 1에 도시된 바와 같이 ARCore는 공식적으로 OpenXR을 지원하지 않으므로 나머지 XR 디바이스와 달리 Galaxy Tab S8의 경우 OpenXR API를 활용할 수 없

다. 상위 계층에는 AR Subsystem/AR Foundation 패키지를 설치하고 MRTK Foundation 패키지를 추가함으로써 XR 콘텐츠 개발을 위한 크로스 플랫폼 개발 환경을 구축하였다.

3-2 이기종 디바이스 지원 XR 콘텐츠 개발

본 연구에서 개발한 XR 콘텐츠는 각기 다른 4종의 XR 디바이스를 이용하는 사용자들이 가상의 XR 협업 공간에 접속해 가상의 3D 드론을 함께 조립하는 응용이다. 가상 드론은 모든 사용자에게 접근 권한이 허용된 공유객체이며, 사용자는 각 디바이스의 입력 장치를 이용해 가상 객체 및 타 사용자와 상호작용을 수행한다. 그림 6은 다양한 부품으로 구성된 가상 드론 세트를 보여주며, XR 콘텐츠를 이용해 여러 명의 사용자가 협업하는 모습이 그림 7에 나타나 있다.

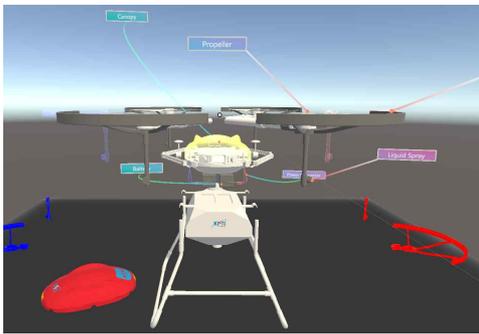


그림 6. XR 콘텐츠: 가상 드론 조립 세트  
Fig. 6. XR content: virtual drone assembly set

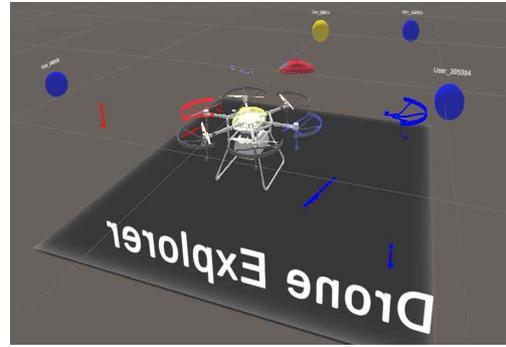


그림 7. 크로스 플랫폼 XR 콘텐츠 기반 다중 사용자 협업  
Fig. 7. Multi-user collaboration based on cross-platform XR contents

아래 그림 8은 각각 AR 디바이스와 VR 디바이스를 사용하는 사용자 1과 사용자 2가 협업에 참여한 후 가상 객체의 위치를 변경할 때, XR 콘텐츠의 내부 처리 과정 및 동기화 서버와의 연동을 도시화한 것이다. 가장 왼쪽에 가상 객체들의 상태를 동기화시키기 위한 동기화 서버가 위치하며, 5개의 주요 모듈로 구성된 XR 디바이스의 처리작업을 보여준다. 본 연구에서 동기화 서버는 멀티플레이어 게임 개발 시 많이 활용되는 Photon 서버[23]를 이용하였다. XR 디바이스의 주요 모듈 중 Sensor, 3D Rendering, Display는 주로 하드웨어 관련 작업을 처리하고, 협업을 위한 가상 객체 및 타 사용자와의 상호작용은 Analysis와 Scene Management 모듈에서 처리된다.

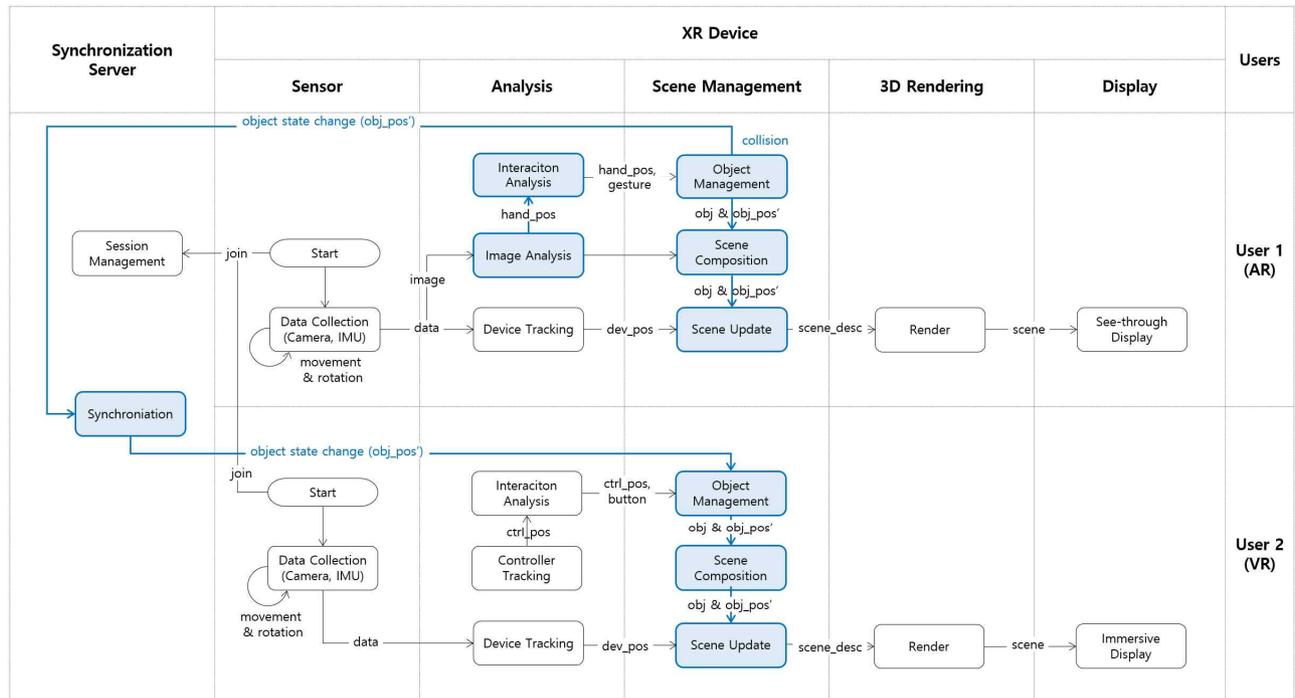


그림 8. AR 디바이스 사용자가 가상 객체 이동 시 XR 콘텐츠의 동기화 처리 과정  
Fig. 8. Synchronization process of XR content when an AR device user moves virtual objects

사용자가 XR 콘텐츠를 실행하면 각자 동기화 서버에 접속해 동일한 XR 협업 세션에 참여한다. 최초 접속자는 협업 공간에 가상 드론을 생성하고 이후의 접속자는 가상 드론을 공유한다. 사용자 1은 주변 탐색 후 QR 코드의 위치 (qr\_pos)에 드론을 배치하고 사용자 2는 주변 환경을 인식하기 어려우므로 초기 위치인 디바이스 앞에 드론을 배치한다. 이후 각 XR 디바이스는 카메라와 위치 추적 센서로부터 자신의 위치를 추적하면서 매 순간 해당 관점 (dev\_pos)에서 바라본 드론 객체 (obj)의 실시간 위치 (obj\_pos)에서의 모습으로 장면 갱신하고 렌더링하는 과정을 반복한다. 사용자 1이 부품을 이동하는 경우, 전면 카메라를 통해 추적된 손의 위치 (hand\_pos)와 부품의 위치가 근접해 충돌이 발생하고 손동작이 옮겨주는 동작과 일치하면, 부품의 위치가 손의 위치를 따라 갱신되고 자신의 디바이스에도 렌더링 된다. 갱신된 부품의 좌표 (obj\_pos')는 Photon 동기화 서버를 통해 최종적으로 사용자 2에 전달되어 사용자 2도 부품의 이동을 경험하게 된다. 사용자 2가 컨트롤러를 이용해 부품을 이동하는 경우, 컨트롤러의 내장 센서로부터 획득한 컨트롤러 위치와 부품의 위치가 근접하고 특정 버튼이 눌리면 부품 객체가 이동하게 된다. 본 논문에서는 OpenXR을 이용해 추상화된 액션을 기반으로 상호작용을 처리하므로 각 디바이스에 따른 사용자 입력 처리를 구현할 필요가 없다. 크로스 플랫폼 상호작용에 대한 설명은 3-3절에서 자세히 기술한다.

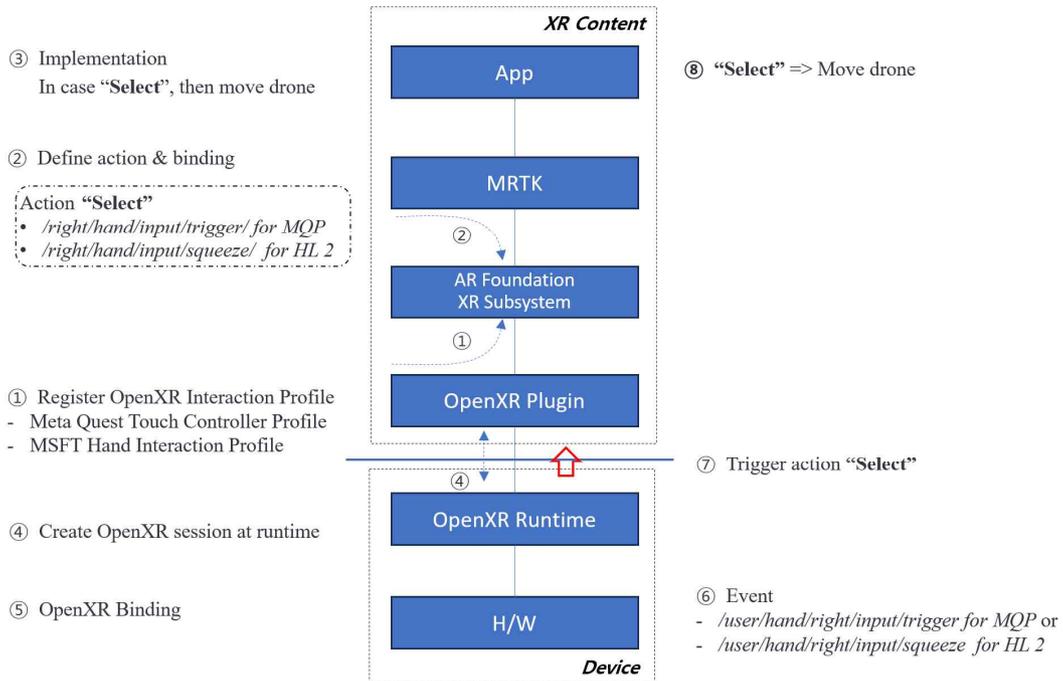
### 3-3 OpenXR 기반 크로스 플랫폼 상호작용 지원

표 2는 본 연구에서 활용한 XR 디바이스의 플랫폼 정보 및 입출력 장치를 보여준다.

**표 2.** XR 디바이스의 플랫폼 및 입출력 장치 정보  
**Table 2.** Platform and I/O information of XR devices

디바이스	플랫폼 (OS)	입력 장치	출력 방식	AR/VR
Desktop PC	Windows 10	마우스, 키보드	Immersive	VR
HoloLens 2	WMR	손동작, 음성, 시선	Optical see-through	AR
Galaxy Tab S8	Android 12	터치스크린	Video see-through	AR
Meta Quest Pro	Android 10	손동작, 컨트롤러	Immersive	VR

MRTK 입력 시스템은 XR 콘텐츠 개발에 있어 다양한 입력 장치와 상호작용을 추상화하고 편리하게 관리할 수 있는 도구이며, 이를 통해 개발자는 여러 종류의 입력 장치에 대한 처리를 일원화된 방식으로 처리할 수 있다. 개발자는 MRTK 입력 시스템을 이용해 추상화된 액션을 정의하고 터치스크린, 컨트롤러, 손동작, 마우스 등 다양한 XR 디바이스의 입력 장치와의 대응을 정의할 수 있다. 본 연구에서는 MRTK 입력 시스템을 이용해 각 디바이스와의 상호작용을 설정할 수 있다. 또한 MRTK 입력 시스템은 다양한 XR 디바이스 및 사용자 입력 장치에 대해 일반적으로 사용되는 액션과의 매핑관계



**그림 9.** OpenXR 액션 기반의 이기종 입력 장치 지원을 위한 이벤트 처리 과정

**Fig. 9.** Event processing process for supporting heterogeneous input devices based on OpenXR action

를 기본적으로 제공하므로 이를 활용해 구현하였다. 새로운 액션을 정의하고 별도의 사용자 입력 장치와 연동하는 방법은 3-4에서 자세히 소개한다.

그림 9는 OpenXR 런타임을 지원하는 HoloLens 2와 Meta Quest Pro를 착용한 사용자가 각각 손동작과 컨트롤러를 이용해 상호작용이 가능하도록 XR 콘텐츠를 개발하는 과정(①-⑤)과 이벤트의 처리 과정(⑥-⑧)을 도시화한 것이다. 3-1과 같이 크로스 플랫폼 개발 환경을 구축 과정에서 MRTK Mixed Reality OpenXR Plugin 설정 시 OpenXR 지원 입력 장치인 Microsoft Hand Interaction 프로파일과 Meta Quest Touch Controller 프로 파일을 등록해 OpenXR 표준에 따른 각 입력 장치의 OpenXR 경로를 추가한다 (①). 이후 MRTK 입력 시스템 또는 유니티 입력 시스템 통해 액션을 정의하고 다양한 입력 장치 또는 제스처와의 매핑관계를 정의한 후, 액션을 기반으로 응용을 구현한다 (②-③). 본 구현에서는 "Select"라는 액션이 HoloLens 2의 움켜쥐는 손동작을 나타내는 OpenXR 경로와 Meta Quest Pro 컨트롤러의 트리거 버튼을 클릭하는 OpenXR 경로에 각각 대응된다. 이후 XR 콘텐츠가 실행되면 콘텐츠의 OpenXR Plugin과 디바이스에 내장된 OpenXR Runtime 간 OpenXR 세션이 맺어지고, 동적으로 탐지된 XR 디바이스 및 입출력 장치와 바인딩이 생성이 완료된다 (④-⑤). 이 시점부터 HoloLens 2 사용자가 가상 객체를 손으로 움켜쥐거나 Meta Quest Pro 사용자가 컨트롤러의 트리거 버튼을 누르면 (⑥) OpenXR 런타임에서는 이에 대응되는 바인딩 된 액션 "Select"를 콘텐츠에 전달함으로써 (⑦), 최종적으로 응용에서 처리된다 (⑧). 이와 같이 액션을 이용한 상호작용의 추상화를 통해 응용에서는 하드웨어에 독립적인 구현이 가능하며, 이기종 장치를 이용한 사용자의 상호작용을 지원할 수 있다.

**3-4 추가적 입력 장치를 이용한 상호작용 구현**

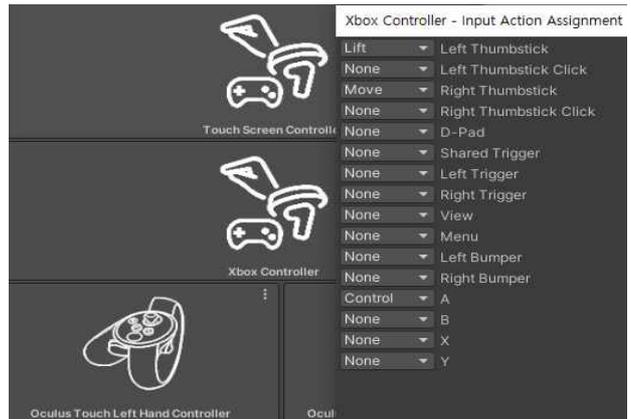
본 절에서는 사용자가 4종 디바이스 외 새로운 입력 장치인 Microsoft Xbox 컨트롤러를 이용해 상호작용할 수 있도록 구현하였다. 사용자는 실물 객체인 Xbox를 이용해 가상 드론의 비행 방향을 보다 정밀하게 조작할 수 있으며, 실감나는 체험을 할 수 있다. 이를 위해 표 3에 도시된 바와 같이, 드론의 전진, 후진, 좌우 회전, 높낮이 제어를 위한 추상화된 새로운 액션들을 정의하고, 그림 10과 같이 MRTK 입력 시스템을 이용해 Microsoft Xbox 컨트롤러에 매핑하였다.

응용에서는 추상화된 액션을 기반으로 드론의 위치를 갱신하는 코드를 개발하였다. 사용자가 Xbox를 이용해 가상 드론의 비행을 조종하는 모습이 그림 11에 나타나 있다.

**표 3.** 가상 드론의 비행 제어를 위한 액션 정의

**Table 3.** Actions for controlling flight of virtual drone

Action	Type	Axis	State	Description
Lift	Dual Axis	X	Degree	Control sensitivity
		Y	Up	Drone ascends upward
			Down	Drone descends downward
Move	Dual Axis	X	Right	Drone turns right
			Left	Drone turns left
		Y	Forward	Drone moves forward
			Backward	Drone moves backward
Control	Digital		On	Start the flight control of drone
			Off	Stop the flight control of drone



**그림 10.** MRTK 입력 시스템의 액션과 컨트롤러 간 매핑 정의  
**Fig. 10.** Define mappings between actions and controllers in the MRTK input system



**그림 11.** Xbox 컨트롤러를 이용한 가상 드론의 비행 제어  
**Fig. 11.** Flight control of virtual drone with Xbox controller

**IV. 크로스 플랫폼 XR 콘텐츠 실증**

본 장에서는 크로스 플랫폼 개발을 통해 개발된 XR 콘텐츠와 4종 디바이스를 이용해 원격 협업 환경을 구축한 후, 이기종 XR 디바이스를 이용한 사용자의 상호작용을 실증하였다.

#### 4-1 이기종 XR 디바이스로 구성된 XR 협업 환경

XR 협업 환경은 그림 12에 도시된 바와 같이 구성된다. 2개의 물리적 협업 공간 A, B가 존재하며 두 공간은 유선망을 통해 원격으로 연결된다. 각 협업 공간은 인터넷을 통해 외부의 Photon 동기화 서버와 연동되며 각 협업 공간 내에서는

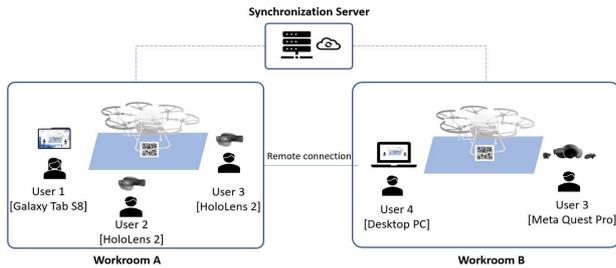


그림 12. XR 협업 테스트베드

Fig. 12. Testbed for XR collaboration

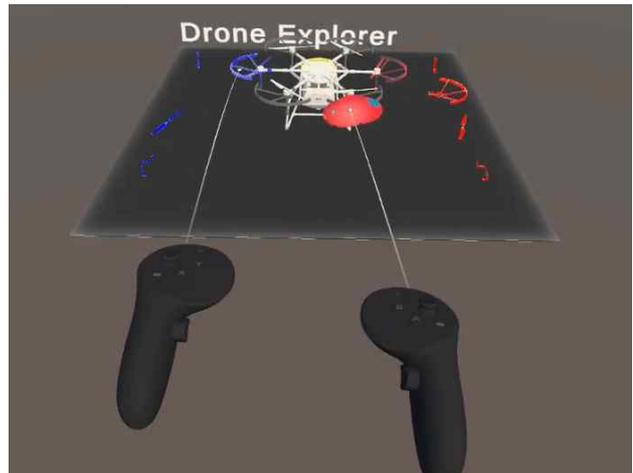
WiFi를 통해 통신한다. 총 5명의 사용자가 4종의 XR 디바이스를 이용하여 협업에 참여한다. 협업 공간 A는 AR 기반의 협업 환경으로 HoloLens 2 사용자 2명과 Galaxy Tab S8 사용자 1명이 참여하며, 실제 공간과 가상 객체를 정합하기 위한 QR 코드가 부착된다. 협업 공간 B는 VR 기반의 협업 환경으로 Desktop PC와 Meta Quest Pro를 이용하는 사용자 2명이 참여한다.

#### 4-2 이기종 XR 디바이스 기반의 XR 협업 시연 및 상호작용 분석

협업 참여자가 XR 디바이스에서 XR 콘텐츠를 실행하면 그림 8에서 기술한 바와 같이, AR 디바이스 사용자와 VR 디바이스 사용자는 각각 탐지된 QR 코드와 XR 디바이스 앞에 드론 객체와 부품을 증강한다. AR 디바이스는 QR 코드의 x, y, z 회전축에 맞추어 드론의 위치와 회전을 정렬하므로 AR 디바이스 사용자들은 물리적 공간의 동일한 지점에 배치된 드



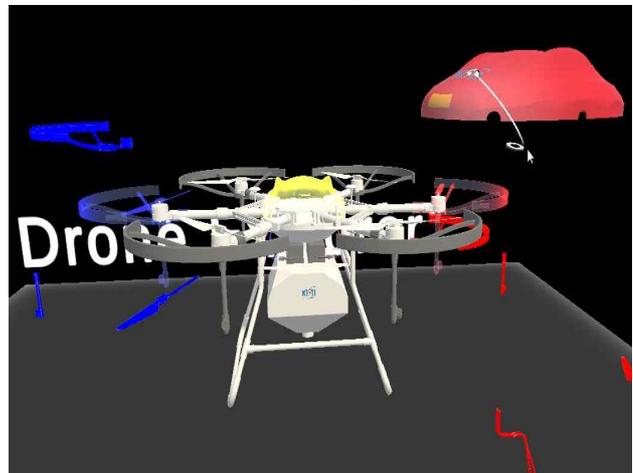
(a) 손동작을 이용한 상호작용  
(a) Interaction using hand gesture



(b) 컨트롤러를 이용한 상호작용  
(b) Interaction using controller



(c) 스크린터치를 이용한 상호작용  
(c) Interaction using screentouch



(d) 마우스와 키보드를 이용한 상호작용  
(d) Interaction using mouse and keyboard

그림 13. 4종 XR디바이스의 사용자 입력 장치를 이용한 가상 객체와의 상호작용  
Fig. 13. User interaction with virtual objects using 4 types of XR devices

론을 볼 수 있다. 또한, 각자의 관점에서 바라본 드론의 모습이 디바이스에 렌더링되고 위치가 실시간으로 동기화되므로 사용자들은 드론을 마치 협업 공간 A에 존재하는 단일한 실물 객체를 공유하는 듯한 경험을 하게 된다. 5명의 사용자는 표 1에 표기된 바와 같이 4종의 XR 디바이스와 입력 장치를 이용해 가상 객체 또는 다른 사용자와 상호작용한다.

그림 13은 사용자들이 각각 손, 컨트롤러, 터치스크린, 키보드 및 마우스를 통해 가상 객체와 상호작용하는 모습을 보여준다. 실험을 통해 크로스 플랫폼 개발을 통해 구현된 XR 콘텐츠가 Android, Windows, UWP 3종의 플랫폼으로 구성된 4종의 디바이스에서 정상적으로 구동되고, 다양한 입력 장치를 이용한 가상 객체와의 상호작용을 성공적으로 지원함을 확인할 수 있었다. 이처럼 사용자 입력 장치의 종류에 상관없이 상호작용이 가능함을 확인할 수 있었던 반면, 상호작용의 편의성 및 지원 가능한 협업 수준에는 차이가 있음을 관찰되었다. 그림 13(a)와 13(b)의 HoloLens 또는 Meta Quest와 같은 HMD 디바이스의 경우 양손을 동시에 사용하여 회전, 이동, 크기 조절 등의 제어가 가능하고 원거리 광선을 이용해 편리하게 협업할 수 있다. 그림 13(c)의 Galaxy Tab을 사용하는 경우, 한 손은 디바이스를 고정해야 하므로 상호작용이 제한되고, 터치스크린 입력은 2D 스크린 좌표 기반이므로 가상 객체가 겹쳐진 경우 가까운 객체만 상호작용이 가능하며 가상 객체의 원근 이동에 대한 제어가 쉽지 않아 정확한 조작을 위해서는 사용자의 잦은 위치 변동이 요구되었다. 그림 13(d)의 Desktop PC의 경우 WADS 키 및 마우스를 이용해 사용자 움직임과 손 위치 이동이 가능하지만, 마우스를 이용하므로 동시에 양손을 활용한 상호작용이 어려워 HMD의 경우와 같이 원활한 상호작용을 기대하기는 어려웠다.

그림 14는 Android Tablet 사용자가 조작하는 모습을 HoloLens 2 사용자의 관점에서 바라본 장면이다. 디바이스의 위치가 구 형태로 표현되어 있으며, 다른 사용자의 상호작용이 HoloLens 2 사용자에게도 동일하게 렌더링 된 것으로 보아 이기종 디바이스간 실시간 동기화가 성공적으로 이뤄진 것을 알 수 있다.



그림 14. 태블릿 사용자가 조작하는 모습을 홀로렌즈 사용자가 바라본 화면

Fig. 14. HoloLens user screen looking at the tablet user's operation

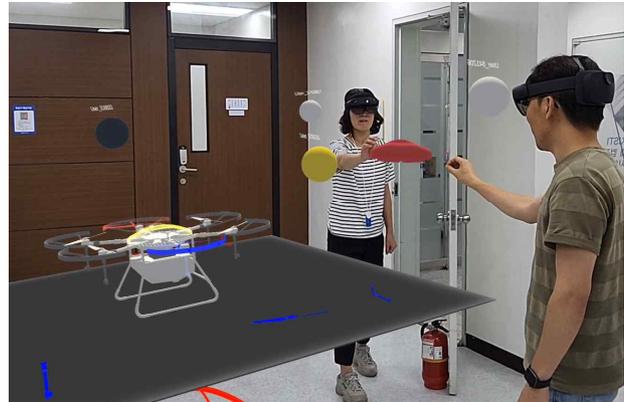


그림 15. 이기종 디바이스 기반 다중 사용자 시연

Fig. 15. Multi-user demonstration with heterogeneous devices

그림 15는 협업 공간 A에 있는 Android Tablet 사용자의 관점에서 나머지 4명의 협업을 바라본 장면으로 4명의 위치가 구 형태로 표현되어 있다. 동일 공간에 위치한 두 명의 HoloLens 2 사용자는 전신을 볼 수 있는 반면, 원격 사용자는 구를 통해 상대적인 위치를 파악할 수 있다. 그림 15에서 AR 디바이스 사용자들 간 부품을 전달하는 상호 간 상호작용 장면을 확인할 수 있다. 앞에서 살펴본 바와 같이 각 디바이스가 가상 객체와의 상호작용을 정상적으로 지원하고, 가상 객체의 위치가 QR코드를 기준으로 정합되며, 각 사용자 및 가상 객체의 상대적인 위치가 실시간으로 동기화되므로 기본적인 전달 작업은 정상적으로 수행할 수 있었다. 하지만 상대방의 상호작용에 대한 예측이 불가능한 경우 협업에 있어 어려움이 발생하였다. 상대방이 VR 디바이스 사용자이거나 원격 사용자인 경우, 현재 콘텐츠는 디바이스 위치만 표시하고 손 위치는 확인할 수 없으므로, 실제 부품 이동이 발생하기 전에는 부품을 전달하려는 상대방의 행위를 예측할 수 없어, 대응되는 행동을 미리 준비하기 어려웠다. 원활한 상호작용을 위해서는 손과 같은 주요 신체의 움직임을 반영한 아바타가 필수임을 알 수 있었다.

한편, 동일 공간에 있는 AR 디바이스 사용자 간 상호작용의 경우, 상대방의 행위를 실제 볼 수 있으므로 보다 원활한 협업이 가능하였다. 그러나 Galaxy Tab 사용자와 상호작용하는 경우, 터치스크린을 이용하는 상대방의 상호작용 내용을 예측할 수 없고, 해당 상호작용을 유발한 주체가 Tab 사용자인지 다른 원격 사용자인지 명확히 파악하기 어려운 경우가 존재하였다. 또한, 시야가 Tablet 화면으로 제한되어 타 사용자들의 위치 및 협업의 전반적인 상황을 파악하며 동시에 다른 사용자와 상호작용을 수행하기에 어려움이 있었다.

실험을 통해 다양한 XR 디바이스를 이용한 가상 객체 및 타 사용자와의 상호작용을 실증함으로써 크로스 플랫폼 개발의 유용성과 이기종 XR 디바이스의 활용 가능성을 입증하였다. 그러나 입력 장치 및 XR 디바이스의 종류에 따라 협업 수준 및 활용 가치에 차이가 있음을 알 수 있었다.

## V. 결 론

본 논문에서는 XR 개발 도구와 개발 환경을 조사하고, MRTK를 비롯한 다양한 XR 개발 도구와 OpenXR 표준을 기반으로 4종의 XR 디바이스를 지원하는 XR 콘텐츠를 개발하였다. 개발된 콘텐츠를 활용해 XR 원격 협업 환경을 구축하여 다양한 입력 장치를 이용한 사용자 상호작용을 실증함으로써 크로스 플랫폼 개발의 유용성과 이기종 XR 디바이스의 활용 가능성을 입증하고자 하였다. 본 연구에서는 OpenXR을 기반으로 추상화된 액션을 이용해 디바이스에 독립적으로 구현을 함으로써, 이기종 XR 디바이스와 다양한 사용자 입력 장치를 지원하였다. 실험을 통해 구현된 XR 콘텐츠가 Android, Windows, UWP 3종의 플랫폼으로 구성된 4종의 XR 디바이스에서 정상적으로 구동되고, 사용자가 이기종 디바이스의 다양한 입력 장치를 활용하여 가상 환경 및 다른 사용자와의 상호작용을 성공적으로 수행할 수 있음을 실증하였다. 이러한 연구 결과는 XR 콘텐츠 개발자들이 보다 효과적으로 다양한 디바이스 및 플랫폼을 활용하여 XR 콘텐츠를 제작할 수 있을 것으로 기대한다. 향후 계획으로 본 콘텐츠를 기반으로 사용자 평가를 수행하고 각 상호작용이 사용자 경험에 미치는 영향 분석과 효용성 평가를 진행하고자 한다.

## 감사의 글

본 연구는 2022년 정부 (과학기술정보통신부)의 재원으로 국가과학기술연구회 오픈XR (OXR) 융합연구사업 (No. CRC 21011)의 지원을 받아 수행된 연구로서, 관계부처에 감사드립니다.

## 참고문헌

- [1] M. Speicher, B. D. Hall, and M. Nebeling, "What Is Mixed Reality?" in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, Glasgow, Scotland, UK, Paper No. 537, pp. 1-15, May 2019. <https://doi.org/10.1145/3290605.3300767>
- [2] O.-Y. Kwon and J.-M. Yu, "Development of a Mixed Reality Systems for Damaged Artifacts via Tactile Interaction," *Journal of Digital Contents Society*, Vol. 22, No. 1, pp. 1-12, January 2021 (In Korean). <https://doi.org/10.9728/Dcs.2021.22.1.1>
- [3] M. Nebeling and K. Madier, "360Proto: Making Interactive Virtual Reality & Augmented Reality Prototypes from Paper," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, Glasgow, Scotland, UK, Paper No. 596, pp. 1-13, May 2019. <https://doi.org/10.1145/3290605.3300826>
- [4] M. Nebeling, K. Lewis, Y.-C. Chang, L. Zhu, M. Chung, P. Wang, and J. Nebeling, "XRdirector: A Role-Based Collaborative Immersive Authoring System," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, Honolulu, HI, USA, Article No. 508, pp. 1-12, April 2020. <https://doi.org/10.1145/3313831.3376637>
- [5] N. Ashtari, A. Bunt, J. Mcgrenerre, M. Nebeling, and P.K. Chilana, "Creating Augmented and Virtual Reality Applications: Current Practices, Challenges, and Opportunities," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, Honolulu, HI, USA, Article No. 593, pp. 1-13, April 2020. <https://doi.org/10.1145/3313831.3376722>
- [6] W. Bi, Y. Ma, D. Tian, Q. Yang, M. Zhang, and X. Jing, "Demystifying Mobile Extended Reality in Web Browsers: How Far Can We Go?," in *Proceedings of the ACM Web Conference*, Austin, TX, Usa, pp. 2060-2069, April 2023. <https://doi.org/10.1145/3543507.3583329>
- [7] J. Choi, S. Jeong, and J. Ko, "Emulating Your Extended World: An Emulation Environment for XR App Development," in *Proceedings of 2022 IEEE 19th International Conference on Mobile Ad Hoc and Smart Systems*, Denver, CO, USA, pp. 131-139, October 2022. <https://doi.org/10.1109/Mass56207.2022.00025>
- [8] J. Levy and D. Liu, "Extended Reality (XR) Environments for Flood Risk Management with 3D GIS and Open Source 3D Graphics Cross-Platform Game Engines: Advances in Immersive Sea Level Rise Planning Technologies for Student Learning and Community Engagement," in *Proceedings of 7th Asres International Conference on Intelligent Technologies*, Jakarta, Indonesia, pp. 271-285, December 2022. [https://doi.org/10.1007/978-981-99-1912-3\\_25](https://doi.org/10.1007/978-981-99-1912-3_25)
- [9] V. Krauß, F. Jasche, S.M. Saßmannshausen, T. Ludwig, and A. Boden, "Research and Practice Recommendations for Mixed Reality Design Different Perspectives from the Community," in *Proceedings of the 27th ACM Symposium on Virtual Reality Software and Technology*, Osaka, Japan, Article No. 24, pp. 1-13, December 2021. <https://doi.org/10.1145/3489849.3489876>
- [10] D. B. Rawat and H. El Alami, "Metaverse: Requirements, Architecture, Standards, Status, Challenges, and Perspectives," *IEEE Internet of Things Magazine*, Vol. 6, No. 1, pp. 14-18, March 2023. <https://doi.org/10.1109/Iotm.001.2200258>

- [11] M. Nebeling and M. Speicher, "The Trouble with Augmented Reality/Virtual Reality Authoring Tools," in *Proceedings of 2018 IEEE International Symposium on Mixed and Augmented Reality*, Washington, DC, USA, pp. 333-337, October 2018. <https://doi.org/10.1109/IsMAR-Adjunct.2018.00098>
- [12] V. Krauß, A. Boden, L. Oppermann, and R. Reiners, "Current Practices, Challenges, and Design Implications for Collaborative AR/VR Application Development," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, Yokohama, Japan, Article No. 454, pp. 1-15, May 2021. <https://doi.org/10.1145/3411764.3445335>
- [13] Microsoft. Mixed Reality Toolkit 2 [Internet]. Available: <https://learn.microsoft.com/en-us/windows/mixed-reality/mrtk-unity/mrtk2/>.
- [14] Unity. AR Foundation [Internet]. Available: <https://unity.com/unity/features/arfoundation>.
- [15] Khronos Group. OpenXR overview [Internet]. Available: [https://www.khronos.org/api/index\\_2017/openxr](https://www.khronos.org/api/index_2017/openxr).
- [16] Google. AR Core [Internet]. Available: <https://developers.google.com/ar>.
- [17] Monado. OpenXR Runtime [Internet]. Available: <https://monado.dev/>.
- [18] Apple. ARKit [Internet]. Available: <https://developer.apple.com/kr/augmented-reality/arkit/>.
- [19] Meta. Oculus Integration SDK [Internet]. Available: <https://developer.oculus.com/downloads/package/unity-integration/>.
- [20] Unity. Oculus XR Plugin [Internet]. Available: <https://docs.unity3d.com/kr/2023.2/Manual/com.unity.xr.oculus.html>.
- [21] Unity. OpenXR Plugin [Internet]. Available: <https://docs.unity3d.com/kr/2023.2/Manual/com.unity.xr.openxr.html>.
- [22] Unity. XR Plugin Architecture [Internet]. Available: <https://docs.unity3d.com/Manual/XRPluginArchitecture.html>.
- [23] Photon Engine [Internet]. Available: <https://www.photonengine.com/>.



**장희진(Heejin Jang)**

2001년 : POSTECH 컴퓨터공학과 (공학학사)  
2003년 : POSTECH 컴퓨터공학과 (공학석사)

2003년 ~ 2009년: 삼성종합기술원 책임연구원  
2010년 ~ 2011년: 국가보안기술연구소 선임연구원  
2012년 ~ 현 재: 한국과학기술정보연구원(KISTI) 선임연구원  
※관심분야 : Augmented Reality, Mixed Reality, Human Interaction, Object Detection



**이성희(Sunghee Lee)**

1995년 : 전남대학교 전산학과 (이학학사)  
2006년 : 한국과학기술원 텔레콤 MBA (경영학석사)

2015년 ~ 현 재: 한국과학기술정보연구원(KISTI) 선임연구원  
※관심분야 : 인간과 컴퓨터의 상호작용(HCI), UI/UX



**신정훈(Jung Hun Shin)**

1999년 : KAIST 기계공학과 (공학학사)  
2002년 : KAIST 기계공학과 (공학석사)  
2012년 : KAIST 기계항공우주시스템공학부 (공학박사-계산과학공학)

2015년 ~ 현 재: 한국과학기술정보연구원(KISTI) 책임연구원  
※관심분야 : 계산과학공학, 데이터 및 인공지능 기반 시물레이션



**이승민(Seungmin Lee)**

2003년 : POSTECH 컴퓨터공학 (공학학사)  
2006년 : POSTECH 컴퓨터공학 (공학석사)

2006년 ~ 현 재: 한국과학기술정보연구원 선임연구원  
※관심분야 : 확장현실(XR), 컴퓨터구조 시뮬레이터, 클라우드 컴퓨팅 등



**김민아(Min Ah Kim)**

1998년 : 부산대학교 전자계산학과 (이학학사)

1998년 : 부산대학교 전자계산학과 (이학석사)

1998년~2000년: LG정보통신

2000년~2003년: (주)필링크

2004년~2005년: 동명정보대 초빙교수

2005년~현 재: 한국과학기술정보연구원(KISTI) 책임연구원

※ 관심분야 : XR 협업 플랫폼



**남덕운(Dukyun Nam)**

1999년 : POSTECH 컴퓨터공학과 (공학학사)

2001년 : KAIST 대학원 (공학석사)

2006년 : KAIST 대학원 (공학박사-분산시스템)

2016년~2018년: 한국과학기술정보연구원 슈퍼컴퓨터SW연구실 실장

2018년~2021년: National Center for Supercomputing Applications, Visiting Scholar

2004년~2022년: 한국과학기술정보연구원 선임연구원

2022년~현 재: 경북대학교 IT대학 컴퓨터학부 조교수

※ 관심분야 : High Performance Computing, Distributed System, Software Optimization