

Metaverse-based hybrid WebXR development and service proposal

Byong-Kwon Lee*

*Professor, School of media contents, Seowon University, Chungbuk, Korea

[Abstract]

Extended Reality (XR) is a standardized technology encompassing virtual reality, augmented reality, and mixed reality. In virtual reality, it is common to use a Head Mounted Device (HMD) for the experience. However, this poses accessibility challenges for users who cannot use an HMD or experience discomfort, such as dizziness. In this study, we propose a hybrid WebXR service that minimizes discomfort and provides an immersive experience for everyone in 3D virtual reality. The hybrid 2-Way approach allows users to simultaneously experience PC-based WebGL interactions and immersive HMD-based virtual reality services through WebXR. Furthermore, to overcome the inconvenience of installing programs within HMD devices, we implement the proposed method using web-based technology, enabling device-independent services. Additionally, we maximize user accessibility by offering WebXR services through 360 panoramas for users without access to an HMD.

▶ **Key words:** Hybrid-WebXR, HMD, 360 panaram, 2-way WebService, Uinty WebGL

[요 약]

확장현실(XR: Extended Reality)은 가상현실, 증강현실, 혼합현실을 포괄하는 용어로, 표준화된 기술로서 사용된다. 가상현실의 경우 일반적으로 HMD(Head Mounted Device)를 사용하여 체험하는 것이 일반적이다. 그러나 HMD를 사용할 수 없거나 어지러움을 느끼는 사용자들은 접근성이 떨어지는 문제가 있다. 본 연구에서는 3차원 가상현실 체험에서의 불편함을 최소화하고 누구나 체험형 서비스를 이용할 수 있는 하이브리드형 WebXR 서비스를 제안했다. 하이브리드형 2-Way 이용방법은 PC 기반의 체험을 위한 WebGL을 활용한 서비스와 HMD를 사용한 가상현실 서비스인 WebXR을 동시에 체험할 수 있는 방식을 연구했다. 또한, 기존의 가상현실 콘텐츠 서비스는 HMD 장치 내부에 프로그램을 설치해야 하는 불편함을 해소하기 위해 웹 기반으로 구현하여 어떤 장치에서든 서비스할 수 있도록 했다. 또한, 제안된 방식은 HMD를 사용할 수 없는 사용자들을 위해 360 파노라마를 통한 WebXR 서비스를 제공하여 사용자의 접근성을 극대화했다.

▶ **주제어:** 하이브리드- WebXR, 가상현실장치(HMD), 360도 파노라마, 2-Way 웹서비스, 유니티 WebGL

• First Author: Byong-Kwon Lee, Corresponding Author: Byong-Kwon Lee
*Byong-Kwon Lee (sonic747@daum.net), School of media contents, Seowon University
• Received: 2023. 06. 07, Revised: 2023. 06. 20, Accepted: 2023. 06. 26.

I. Introduction

가상현실 장치(HMD:Head Mounted Device)의 불편함(무거움)과 어지러움으로 가상현실 콘텐츠 및 서비스는 점점 더 사용자가 줄어들고, 관심도 약해지는 추세이다[1]. 이러한 이유는 앞에서 언급했듯이 사용상의 불편함이다. 대표적인 것이 어지러움과 사용상의 불편함이다[2]. 본 연구에서는 HMD 장치를 꺼리거나 불편함을 호소하는 사용자를 위해 장치 없이도 가상현실을 체험할 수 있도록 웹 기반의 3차원 360 파노라마 서비스를 제공하고, 실감형(체감형) 서비스 사용자를 위한 HMD 서비스를 제공하는 하이브리드형 WebXR 서비스를 연구했다[3]. Fig.1은 본 연구에서는 제안하는 하이브리드형 WebXR 개념도이다. 본 연구에서는 Fig.1과 같이 하이브리드형 웹 XR 서비스를 제안한다.

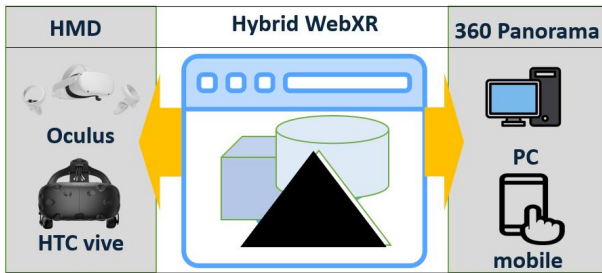


Fig. 1. Concept of Hybrid WebXR Service

II. Preliminaries

2.1 OpenXR and WebXR

OpenXR은 Khronos[4]의 개방형으로 로열티 없는 API 표준이며, 혼합 현실 스펙트럼 서비스로 다양한 HMD 디바이스에 대한 기본 액세스를 엔진에 제공한다. Fig.2는 OpenXR의 서비스 플랫폼이다.

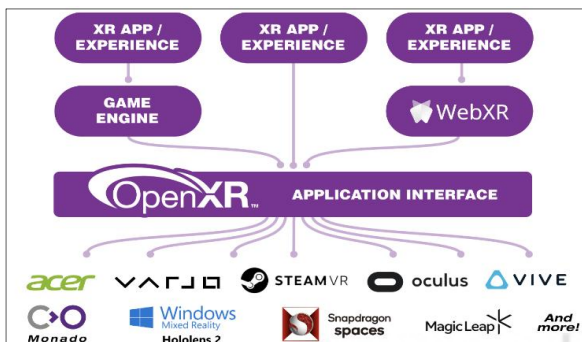


Fig. 2. OpenXR provides a single cross-platform

OpenXR API는 홀로그램 및 몰입형 디바이스를 대상으로 엔진을 빌드해, 핵심 포즈 예측, 프레임 타이밍 및 공간 입력 기능을 제공한다. 또한, OpenXR를 관리하는 크로노스 그룹은 3차원 그래픽스, 컴퓨터 비전, 증강현실, 가상현실, 병렬처리 등의 개방형 표준을 제정했으며, 140개 이상의 하드웨어 및 소프트웨어 관련 기관과 산업체가 모여, 컨소시엄 통해 표준을 제정했다. 이때, 제정한 대표적인 표준에는 WebGL, OpenCL, Vulkan, OpenGL, OpenVX, NNEF, COLLADA, glTF 등이 있다[5].

WebXR는 W3C에서 2018년에 공개하였으며, 2020년에는 WebXR 모듈과 WebXR 증강현실 모듈 공개했다. 또한, WebVR(가상현실)과 WebAR(증강현실)은 API가 별도로 개발되었으나, WebXR는 웹에서 가상현실 및 증강현실을 모두 지원한다. WebXR과 OpenXR은 확장현실(XR) 및 가상현실(VR)과 관련된 기술 표준이다[6]. 그러나 두 기술은 서로 다른 분야에서 사용되며 목표와 플랫폼에서 차이가 있다. WebXR은 웹 기반의 확장현실과 가상현실을 지원하기 위한 표준이며, HTML, CSS, JavaScript와 같은 웹 기술을 사용하여 브라우저에서 AR 및 VR 콘텐츠를 실행할 수 있도록 개발되었다. 이것은 웹 개발자들이 브라우저에서 AR 및 VR 콘텐츠를 개발하고 배포할 수 있도록 지원한다. WebXR은 크로스 플랫폼 호환성이 있어, 다양한 HMD 기기로 웹 브라우저(크롬,오페라,엣지)에서 동작한다. 반면에 OpenXR은 XR 디바이스(AR 및 VR 헤드셋 등)와 XR 애플리케이션 간의 표준 인터페이스를 제공하는 표준으로, 하드웨어 및 소프트웨어 제조업체 간의 호환성을 향상하기 위해 개발되었다. Fig. 3과 같이 OpenXR은 다양한 플랫폼에서 XR 애플리케이션 개발을 단순화하고, 애플리케이션이 다양한 XR 디바이스에서 작동하도록 지원한다. OpenXR은 렌더링 엔진, 입력 및 센서 시스템 등과 같은 XR 장치와의 상호작용을 표준화한 것이다[7].

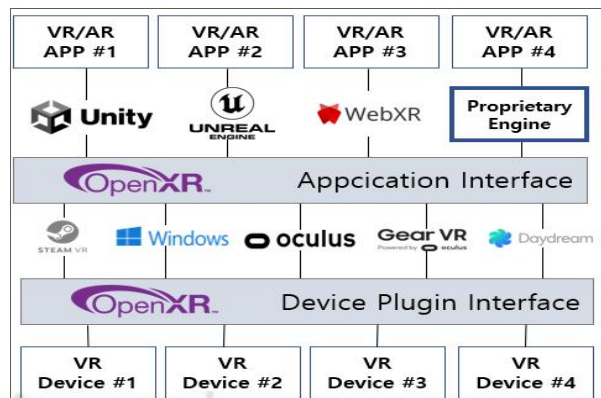






Fig. 3. Wide interoperability of XR apps and device

다시말해, WebXR은 웹 브라우저에서 XR 콘텐츠를 실행하기 위한 표준이며, OpenXR은 XR 디바이스와 XR 애플리케이션 간의 인터페이스를 표준화한 것이다. WebXR은 웹 개발자를 대상으로 하고, OpenXR은 하드웨어 및 소프트웨어 제조업체를 대상으로 한다.

2.2 Web of XR Service

Table 1. Pros and Cons Web based XR Service

| XR Service | Pros : Advantages Cons : Disadvantages |
|---|---|
|  WebXR | <ul style="list-style-type: none"> Platform-ınd and accessible through web browsers. No need for users to download or install separate apps. Being a relatively new technology, some of the latest features may not be supported by all browsers. VR and AR experiences may be limited due to reliance on browser and hardware performance. |
|  A-Frame | <ul style="list-style-type: none"> Provides an easy-to-use entity component system that enhances development speed. Familiar to web developers due to its HTML-like syntax. Limitations in implementing complex interactions or advanced functionalities. Slightly lower performance compared to other frameworks. |
|  AR.js | <ul style="list-style-type: none"> Enables the implementation of AR experiences using web technologies. Accessible through web browsers across different devices. Some features may be limited, especially on older browsers. Accessibility may be restricted for some users due to high-performance requirements and limited device support. |
|  Babylon.js | <ul style="list-style-type: none"> Provides a powerful 3D graphics engine for high-quality VR and AR experiences. Works on various platforms and devices. Steeper learning curve and may require additional development work for complex interactions or advanced features. Demands higher performance, thus may not deliver optimal performance on some devices. |

현재, 공개된 웹기반 서비스는 WebXR, A-Frame, AR.js 및 Babylon.js가 대표적이다. 표1은 웹에서 XR 서비스를 제공하는 서비스 플랫폼의 장·단점을 비교한 것이다. **WebXR API:** WebXR API는 웹 브라우저에서 가상현실(VR) 및 증강현실(AR)을 지원하기 위한 Table준 API이다. 이를 사용하여 웹 기반의 VR 및 AR 콘텐츠를 개발할 수 있다. WebXR API는 웹 브라우저에서 VR 헤드셋, 모바일 기기의 내장 카메라, AR 마커 등 다양한 기기와 상호

작용할 수 있는 기능을 제공한다. **A-Frame[8]:** A-Frame은 웹 기반의 가상현실(VR) 콘텐츠를 개발하기 위한 프레임워크이다. HTML과 유사한 문법을 사용하여 VR 콘텐츠를 구축할 수 있으며, WebGL을 기반으로 3D 그래픽을 렌더링한다. A-Frame은 WebXR API와 연동하여 다양한 디바이스에서 실행할 수 있는 VR 경험을 제공한다. **AR.js:** AR.js는 웹 브라우저에서 증강현실(AR) 콘텐츠를 개발하기 위한 오픈소스 라이브러리이다. 이를 사용하여 웹 카메라를 통해 실시간으로 환경을 탐지하고, 3D 모델이나 이미지 등을 기반으로 증강된 콘텐츠를 표시할 수 있다[9]. AR.js는 WebGL을 기반으로 동작하며, 스마트폰의 내장 카메라를 사용하여 웹 브라우저에서 AR 경험을 제공한다[9][10]. **Babylon.js:** Babylon.js는 웹 기반의 3D 게임 및 가상현실(VR) 콘텐츠를 개발하기 위한 JavaScript 프레임워크이다. Babylon.js는 WebGL을 사용하여 3D 그래픽을 렌더링하며, WebXR API와 통합하여 VR 콘텐츠를 제작하고 실행할 수 있다.

본 연구에서는 W3C표준 협회에서 관리하고 있고 WebGL서비스 및 유니티 게임엔진 컴퍼넌트 그대로 사용하면 W3C 표준인 WebXR 서비스가 가능한 게임엔진 기반의 하이브리드형 WebXR 서비스를 연구했다. 여기에서 게임엔진을 사용하는 이유는 게임엔진에는 다양한 기술 및 API를 제공하고 있고, 현재 가상현실 개발자들이 대부분이 게임엔진을 이용한 콘텐츠를 만들고 있다.

III. The Proposed Scheme

본 논문에서는 하이브리드형 WebXR 지원을 목표로 연구했다. Web에서 VR서비스는 다양한 플랫폼(Aframe, WebXR, AR.js, Babylon.js)을 제공하고 있지만, 게임엔진(유니티)을 이용해서 Web상에 퍼블리쉬(Publish)하는 방법을 연구했다.

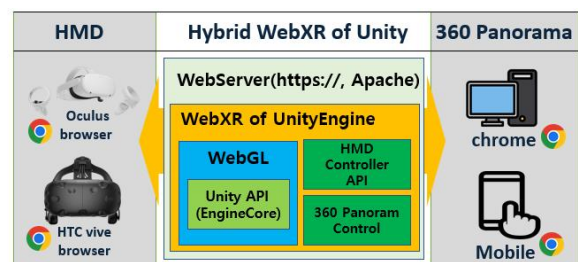


Fig. 4. Hybrid WebXR of Unity

또한, HMD가 없는 사용자에게도 PC 브라우저(오페라, 크롬, 사파리, 엣지) 기반으로 간접 체험할 수 있도록 2-way 방식을 연구했다. Fig. 4는 본 연구에서 제안하는 게임엔진(unity) 기반 하이브리드 WebXR 플랫폼 구성도이다.

3.1 Unity content production and WebGL

기존 표준화된 WebXR API를 사용할 경우 웹 스크립트 (JavaScript)를 3D VR형태로 개발하면 게임엔진인 유니티 엔진에서 제공하는 다양한 형태의 에셋(기능 및 도구)을 사용하지 못하는 단점이다[11]. 본 연구에서는 유니티의 모든 기능을 사용할 수 있으면서, Web 서비스할 수 있도록 WebGL기반의 XR을 연구했다. 유니티 기반의 WebGL를 로딩해오기 위해서는 유니티 플랫폼의 경우 Build Setting에서 HTML5 기반의 WebGL을 가져올 수 있도록 컴파일 플랫폼을 “Switch Platform”으로 변경해야 한다. 유니티 포함된 WebGL은 Web 퍼블리쉬 기술로, 웹 브라우저에서 2D 및 3D 그래픽을 렌더링하기 위한 API(응용 프로그래밍 인터페이스)이다[12]. WebGL은 웹 기반 그래픽 애플리케이션을 개발하기 위해 사용되며, HTML5와 JavaScript와 함께 작동한다. 유니티에서 WebGL을 사용하면 Unity 게임을 웹 브라우저에서 실행할 수 있다. WebGL 렌더링은 유니티의 그래픽 엔진을 통해 수행되며, 유니티의 다양한 기능과 도구를 활용하여 웹 기반 게임을 개발하는데 응용된다.

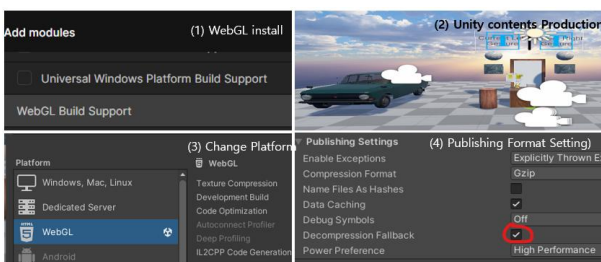


Fig. 5. Install WebGL and Setting of Unity

Fig.5는 유니티에서 제공하는 웹 기반 그래픽 플러그인 서비스로 WebGL를 설치하고, 설정하는 과정이며, Fig.5의 (2)의 실험을 위해 3D 공간을 구성한 것이고, (4)는 Web Publishing을 위한 압축해제 옵션을 설정한 것이다 (이유:WebGL의 단점인 로딩속도가 떨어지는 문제해결). 이로써 웹-상에서 GL(Graphic Library)을 사용할 수 있도록 구성됐다.

3.2 HMD Controller API Interface for WebXR

웹(web)에서 가상현실 장치(HMD)를 제어(컨트롤러)하기 위해서는 OpenXR 이용해서 장치를 로딩하고 WebXR 표준 API를 사용해 헤드셋 및 컨트롤러를 제어하는 프로그램을 작성해야 된다.

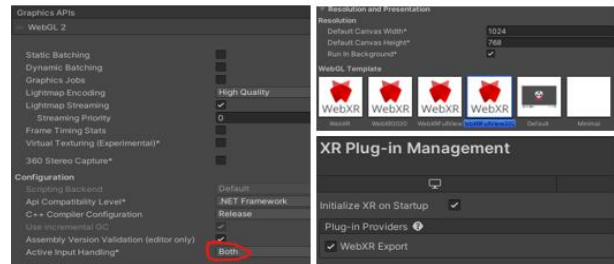


Fig. 6. WebXR Setting of Unity

Fig.6은 WebXR API device를 버전별로 설정하는 과정으로 “Active input Handling”을 “Both”로 설정하고 WebXR 장치 설정을 마무리한다. 또한, HMD 컨트롤러의 사용을 “WebXRSettings.cs” 스크립트 파일을 통하여 환경설정을 했다.

Table 2. WebXR Settings for Unity XR

| Functions | Value |
|----------------------|---|
| Reference SpaceTypes | local = 1, local_floor = 2, bounded_floor = 4, unbounded = 8, viewer = 16 |
| ExtraFeature Types | hit_test = 1, hand_tracking = 2 |
| VR Settings | ReferenceSpaceTypes.local_floor; ExtraFeatureTypes.hand_tracking; |
| AR Settings | ReferenceSpaceTypes.local_floor; (ExtraFeatureTypes)(-1); |

Table 2는 WebXR 설정으로 C# 코드로 작성해서 환경 설정으로 사용하는데. VR 및 AR를 동시 지원하도록 설정되었다.

3.3 Interface implements for Unity WebXR

HMD 장치를 이용한 가상현실은 기본 원리가 카메라 두 개를 사용하는 방식인 side-by-side 형태의 양안 카메라를 Import해야한다[13]. 이를 위해, 유니티에서 제공하는 “Main Camera”는 side-by-side 형태의 화면을 제공하기 위해 카메라 두 개를 배치했다.

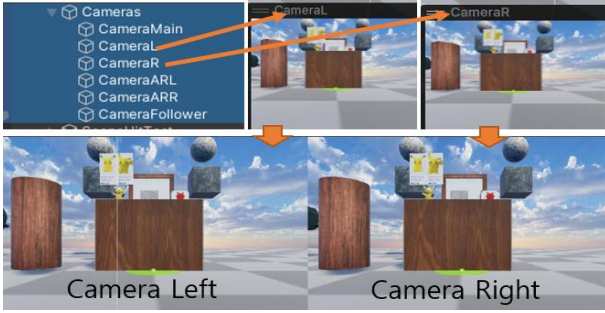


Fig. 7. Side By Side Camera Setting for VR

Fig.7은 가상의 “Main Camera” 카메라 2개를 이용해서 카메라를 배치했다. 기존의 카메라를 삭제하고 새롭게 CameraMain 만들었고, 하위 두 개의 카메라인 CameraL과 CameraR 생성해 양안카메라를 만들었다. 또한, 유니티의 WebGL API도 Import해 HMD 없이도 PC 브라우저 사용 할 수 있도록했으며, 만일, HMD가 있으면 VR모드로 변경 가능하도록 인터페이스를 개발했다.

3.4 Hybrid(Mouse, HMD Controller) contents Production process for Unity WebXR

본 절에서는 Cube 객체를 웹 기반에서 마우스 이용해 Drag and Drop 구현하는 과정과 HMD Controller에서 Drag and Drop하는 방법을 소개한다.

Table 3. Hybrid Control of Web and HMD(VR)

| Hybrid Control of Web and HMD | |
|--|--|
| Mouse (Web) Drag and Drop | HMD Controller(VR) Drag and Drop |
| Import File : MouseDragObject.s Script Code: m_rigidbody = GetComponent<Rigidbody>(); Vector3(Input.mousePosition.x, Input.mousePosition.y, screenZ) | Import File : InteractableFacade.cs Script Code: Interactables.Interactables.Grab.Provider; Interactions.Interactables.Interactables.Grab.Receiver; Interactions.Interactables.Interactables.Interactors; |

표3은 하이브리드형 2-Way 방식으로 웹에 접근하기 위해 웹(web)과 HMD Controller를 동시에 지원하도록 구현한 코드로 Web에서는 Mouse 이용해 객체를 움직이고, 가상환경에서는 HMD controller를 통해 움직이도록 구현하는 주요코드이다. 두 가지 경우 모두 객체의 충돌범위를 설정하는 콜라이더(collider)와 래지드바디(Rigidbody)를

기본적으로 설정했다. 하지만, 유니티에서 제공하는 객체는 기본적으로 콜라이더가 생성되어 외부에서 가져오는 객체의 경우는 콜라이더 설정을 추가적으로 해야한다. Fig. 8은 객체를 잡고 움직이기 위한 콜라이더를 추가한 것으로 콜라이더의 유형은 BoxCollider, CircleCollider, CapsuleCollider, MashCollider등이 있고 객체의 모양에 따라 선택적으로 적용한다.



Fig. 8. Collider Creation for Object Drag

3.5 Implement of Around View and Movement

전시관 및 행사장을 구현할 때는 가상공간에 전.후.좌.우 및 전체조망(All Around View)과 같은 움직임을 마우스 제어 및 HMD 컨트롤러에 구현해야 한다. 이 역시 2-Way(Web and HMD) 방식으로 동작해야 한다. 마우스를 이용한 Web 기반의 동작 구현은 카메라의 이동을 이용해서 구현했다. 키보드의 “A”, “W”, “D”, “S” 또는 방향키로 전.후.좌.우로 움직이도록 했다. 또한, 마우스의 휠을 이용해 Zoom In, Zoom Out으로 동작하도록 했다.

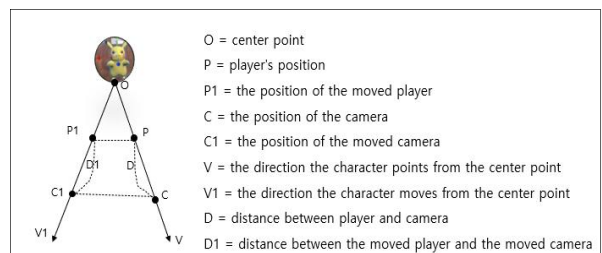


Fig. 9. Camera Movement Method

Fig.9는 카메라 동작원리로 O = 중심점(이 점을 중심으로 캐릭터가 움직이고, 카메라는 이 중심점을 바라본다.), P = 플레이어의 위치, P1 = 이동한 플레이어의 위치, C = 카메라의 위치, C1 = 이동한 카메라의 위치, V = 중심점에서 캐릭터를 가리키는 방향, V1 = 중심점에서 이동한 캐릭터를 가리키는 방향, D = 플레이어와 카메라 간의 거리, D1 = 이동한 플레이어와 이동한 카메라 간의 거리이다. 이를 시스템상에서 캐릭터의 이동으로 적용하면 다음

과같은 단계를 거치게 된다. (1) 캐릭터가 앞으로 간다는 것은 중심으로 다가가는 것. (2) 뒤로 간다는 것은 중심점에서 멀어지는 것, (3) 왼쪽으로 간다는 것은 중심점을 기준으로 시계 방향으로 움직이는 것을 의미, (4) 오른쪽으로 간다는 것은 중심점을 기준으로 반시계 방향으로 움직이는 것을 의미한다. 카메라의 위치를 캐릭터의 이동에 따라 움직일 경우 $V, V1$ 의 값을 찾아야 한다. $V = P - O$:: 중심점에서 플레이어를 가리키는 방향은 플레이어 위치값에서 중심점의 위치값을 빼면 구할 수 있다. $V1 = P1 - O$:: 중심점에서 이동한 플레이어를 가리키는 방향 역시 위와 같다. 또한 카메라의 위치 $C = P + (||V|| * D)$:: 카메라의 위치는 플레이어의 위치에서 $||V||$ 방향으로 설정하고자 하는 거리 D 만큼 곱한 것으로 구할 수 있다. $C1 = P1 + (||V1|| * D1)$:: 와 같다. ($||V||$ 는 V 라는 벡터의 단위 벡터, 즉 길이가 1인 V 벡터를 의미한다.) 이와 같은 방법으로 카메라를 이용시키면 All Around View 구현할 수 있다. Fig. 10은 카메라가 특정 객체를 따라가면서 장체를 조망할 수 있도록 구현된 것이다.



Fig. 10. Implementation of Movement

또한, HMD 기반의 XR은 HMD Controller에 카메라와 컨트롤러를 연동해서 화면을 이동하도록 구현했다. Fig. 11은 HMD Controller의 버튼 구성도이다.

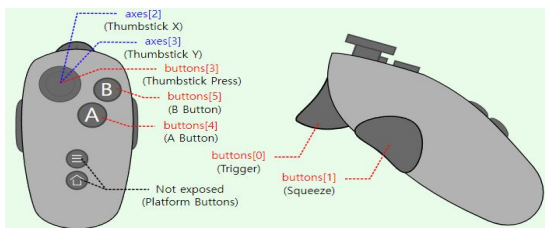


Fig. 11. Complex Controller for HMD Device

HMD 컨트롤러를 이용해 객체를 짚어서 움직이(Drag and Drop)는 기능을 구현하기 위해서는 객체를 선택하는 기능과 짚는 동작할 수 있는 컨트롤러 버튼 설정 작업이

필요하다. 일명 “Pointer Grabber Facade”라고 하고, WebXR API에서 장치별(Oculus, HTC Vive) 기본 API를 제공하고 있다. Fig. 11에 정의된 각 컨트롤러의 설정하는 부분이다.

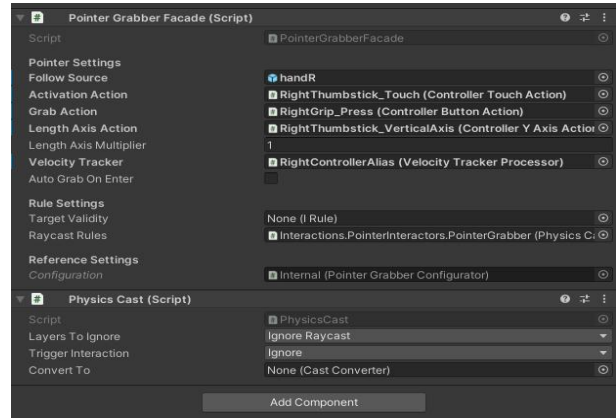


Fig. 12. Pointer Grabber Facade for Object Grab

Fig. 12는 객체를 짚기 위한 컨트롤러 설정으로 “Follow Source”: 컨트롤러 오른쪽선택, “Activation Action”: Thumbstick_Touch로 스틱에 손가락을 대면 동작하는 방식, “Grab Action”: 오른쪽 그립 버튼으로 객체 짚기, “Length Axis Action”: Raycast 방향설정, “Velocity Tracker”: 스택킹 속도 조절이다.

IV. Experiment and analysis

본 절에서는 게임엔진(Unity)을 기반으로 하이브리드형 WebXR을 퍼블리쉬(Publishing)하고, 시험하는 과정이다. Fig.13은 WebGL로 export 하는 과정으로 웹서버에서 사용할 수 있도록 index.html 파일을 생성하는 과정이다. 이렇게 export 된 파일은 웹서버(아파치, IIS, WAS etc)에 FTP로 업로드해 확인할 수 있다.

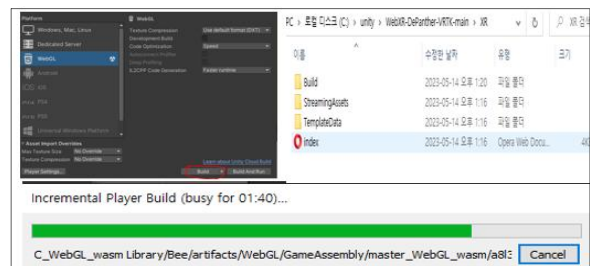


Fig. 13. Pointer Grabber Facade for Object Grab

Fig.14는 웹상에서 유니티로 제작된 콘텐츠를 360도 파노라마 동작 여부로 HMD 장치 없이도 All Around View를 확인한 것이다. Fig. 14의 하단의 “AR,VR”은 HMD 장치를 통해서 가상현실 또는 증강현실로 입장하는 인터페이스를 따로 제공한다(“A”,“W”,“S”,“D”:전후좌우, 마우스 휠:ZoomIn/out, 마우스 왼쪽:Object Movement).

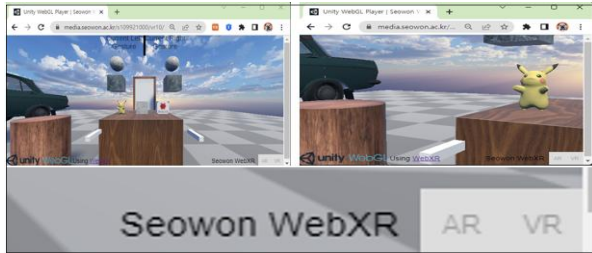


Fig. 14. Web Based for Hybrid WebXR

Fig. 15는 HMD(Oculus Quest2)를 이용해 가상환경에 진입한 것이다. 가상현실 장치인 HMD 사용하면 실감형 콘텐츠에 대하여 직접 체험할 수 있다.



Fig. 15. HMD Based for Hybrid WebXR

Table 4은 XR 솔루션을 제공하는 타 서비스와의 비교 분석한 것으로, Web Service 퍼블리쉬 지원, HMD device 지원여부, Install APP: 개발한 앱을 장치에 설치 지원, 360 Panorama View: All Around View 지원, Mouse Control : HMD 사용하지 않고 마우스제어 지원, Unity Asset : 유니티에서 사용하는 각종 도구 지원 등으로 구분해서 지원 여부를 비교 분석한 것이다. 본 연구의 목표는 막강한 유니티 게임엔진의 다양한 에셋을 사용하면서 웹서비스를 지원할 수 있고, HMD 없이도 가상환경으로 접근할 방법을 연구했다.

Table 4. Comparative analysis with other platform services

| Div | Unity XR | AframeXR | WebXR | Hybrid XR |
|----------------|------------|-------------|-------------|------------|
| WebService | NotSupport | Support | Support | Support |
| HMD device | Support | Support | Support | Support |
| Install APP | Support | No Install | No Install | No Install |
| 360Panora View | Support | Support | Not Support | Support |
| Mouse Control | Support | Support | Not Support | Support |
| UnityAsset | Support | Not Support | Not Support | Support |

V. Conclusions

한동안 코로나19로 인해 가상현실에 대한 관심이 많았으나, 체감형과 실감형 가상현실의 고질적인 문제인 어지러움과 사용상의 불편함 때문에 가상현실 서비스는 점차 위축되고 있는 분위기이다. 이러한 문제들은 대부분 HMD 장치가 필요하다는 점에서 비롯되었다. 본 연구에서는 HMD 없이도 간접 체험할 수 있는 Hybrid형 XR 서비스를 제공하는 방법을 연구했다. 또한, 기존의 웹 기반 XR 서비스는 개발자가 자체 솔루션인 API를 학습하고 개발하는 과정을 거치기 때문에 접근성이 낮으며[14][15], 게임 엔진의 다양한 기능을 활용하지 못하는 단점이 있다. 이러한 단점을 보완하기 위해 마우스를 사용하여 웹에서 XR를 체험할 수 있으며, HMD를 가진 사용자는 WebXR를 통해 체험할 수 있는 하이브리드형 웹 서비스를 제공했다. 이를 위해 기존의 WebXR 서비스에 Unity 플랫폼 서비스를 WebGL을 통해 통합하고, Unity WebXR 서비스를 기본 플랫폼으로 사용하여 연동하는 방법을 연구했다. 이를 통해 사용성과 사용자 접근성을 극대화하고, 가상현실 콘텐츠가 침체기에 접어들어 있는 상황에서 새로운 도약의 기회를 제공할 수 있을 것으로 기대된다. 향후 연구로, 인공지능 기술 특히 GPT 기술을 활용한 자동화된 WebXR 서비스에 관한 연구가 앞으로 필요할 것으로 예상된다.

REFERENCES

[1] J. Xu and W. Song, "Study on Multi-Scene VR Service Mode of Library in 5G Environment," 2020 IEEE 3rd International Conference on Information Systems and Computer Aided Education (ICISCAE), Dalian, China, 2020, pp. 257-261, doi: 10.1109/ICISCAE51034.2020.9236799.

- [2] Byong-Kwon Lee, "A study on the analysis of virtual reality platform API for virtual reality (VR) development", *Journal of The Korea Society of Computer and Information (JKSCI)*, 25, no.8 (2020) : 23-30. doi: <https://doi.org/10.9708/jksci.2020.25.08.023>
- [3] Byong-Kwon Lee, "A Study on Realistic 360 Degree Panorama Webtoon-Metaverse Service", *Journal of The Korea Society of Computer and Information (JKSCI)*, 27, no.10 (2022) : 147-153. doi: <https://doi.org/10.9708/jksci.2022.27.10.147>
- [4] Khronos, <https://www.khronos.org/>
- [5] Z. Long, H. Dong and A. E. Saddik, "Interacting With New York City Data by HoloLens Through Remote Rendering," in *IEEE Consumer Electronics Magazine*, vol. 11, no. 5, pp. 64-72, 1 Sept. 2022, doi: 10.1109/MCE.2022.3165961.
- [6] B. MacIntyre and T. F. Smith, "Thoughts on the Future of WebXR and the Immersive Web," 2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct), Munich, Germany, 2018, pp. 338-342, doi: 10.1109/ISMAR-Adjunct.2018.00099.
- [7] M. Engberg, J. David Bolter and B. MacIntyre, "RealityMedia: An Experimental Digital Book in WebXR," 2018 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct), Munich, Germany, 2018, pp. 324-327, doi: 10.1109/ISMAR-Adjunct.2018.00096.
- [8] P. Sinthong and M. J. Carey, "AFrame: Extending DataFrames for Large-Scale Modern Data Analysis," 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 2019, pp. 359-371, doi: 10.1109/BigData47090.2019.9006303.
- [9] Z. Li, H. Shu, W. Song, X. Li, L. Yang and S. Zhao, "Design and implementation of traditional kite art platform based on human-computer interaction and WebAR technology," 2021 2nd International Conference on Intelligent Computing and Human-Computer Interaction (ICHCI), Shenyang, China, 2021, pp. 209-212, doi: 10.1109/ICHCI54629.2021.00051.
- [10] R. Miao, J. Song and Y. Zhu, "3D geographic scenes visualization based on WebGL," 2017 6th International Conference on Agro-Geoinformatics, Fairfax, VA, USA, 2017, pp. 1-6, doi: 10.1109/Agro-Geoinformatics.2017.8046999.
- [11] A. Sheik Abdullah, A. Manoj, G. T. Tarun Kishore and S. Selvakumar, "A New Approach to Remote Health Monitoring using Augmented Reality with WebRTC and WebXR," 2021 22nd International Arab Conference on Information Technology (ACIT), Muscat, Oman, 2021, pp. 1-5, doi: 10.1109/ACIT53391.2021.9677324.
- [12] S. Boustila, T. Guégan, K. Takashima and Y. Kitamura, "Text Typing in VR Using Smartphones Touchscreen and HMD," 2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR), Osaka, Japan, 2019, pp. 860-861, doi: 10.1109/VR.2019.8798238.
- [13] X. Zhang, Y. Zhao, N. Mitchell and W. Li, "A New 360 Camera Design for Multi Format VR Experiences," 2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR), Osaka, Japan, 2019, pp. 1273-1274, doi: 10.1109/VR.2019.8798226.
- [14] N. T. Banerjee, A. J. Baughman, S. -Y. Lin, Z. A. Witte, D. M. Klaus and A. P. Anderson, "Side-by-Side Comparison of Human Perception and Performance Using Augmented, Hybrid, and Virtual Reality," in *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 12, pp. 4787-4796, 1 Dec. 2022, doi: 10.1109/TVCG.2021.3105606.
- [15] Martin Maier, "6G Post-Smartphone Era: XR and Hybrid-Augmented Intelligence," in *6G and Onward to Next G: The Road to the Multiverse*, IEEE, 2023, pp.101-130, doi: 10.1002/9781119898566.ch5.

Authors



Byong-Kwon Lee received the B.S., M.S. and Ph.D. degrees in Computer Science and Engineering from Hanbat, Hannam and Chungbuk University Korea, in 2000, 2003 and 2007, respectively.

My main areas of interest are embedded systems, virtual and augmented reality (VR, AR), and artificial intelligence (AI). The field currently being studied is the construction of an exhibition hall using virtual reality. It is a technology that combines AI with cultural uniform restoration technology as a future research field.