

가상 증강현실에서의 OpenXR과 WebXR

□ 강진규 / 공간의파티

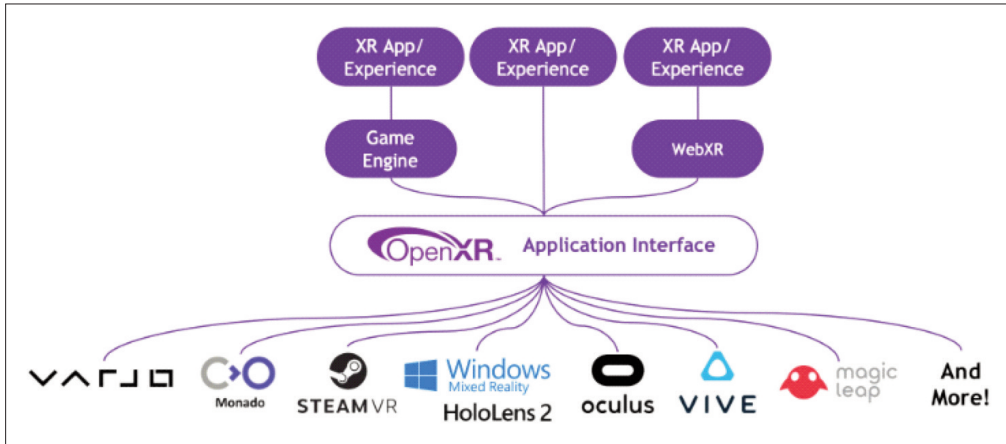
요약

최근 3D 그래픽스 기술과 관련된 기기들의 비약적 발전을 통해 가상, 증강, 확장 현실이라는 기존의 현실 세계에 새로운 정보를 더하는 연구가 활발하게 진행되고 있다. 우선 앱 기반의 가상, 증강, 확장 현실을 구현하는 과정에서 기기별 또는 브라우저별, OS별로 개발 환경이 다름으로 인한 개발비용이 상승하는 등의 애로사항들이 있었고, 이로 인해 Web 환경에서의 가상, 증강, 확장 현실을 구현하기 위한 표준형 통합 플랫폼의 필요성이 대두하였다. OpenXR, WebXR이 가상 증강현실에서의 새로운 표준을 제시하고 기술을 공개하여, 하드웨어와 소프트웨어 산업 모두에서 좋은 호응을 얻기 시작하였다. 따라서 본 고에서는 가상 증강현실에서의 OpenXR과 WebXR 표준에 대한 정의 및 기술에 대하여 살펴보고, 가상 증강 환경에서의 기술 동향과 발전방향에 대하여 살펴보고자 한다.

1. 개방형 표준 OpenXR과 WebXR

1. OpenXR

크로노스 그룹은 3차원 그래픽스, 증강, 가상현실, 컴퓨터 비전, 기계학습 병렬처리 등의 첨단 분야의 개방형 표준을 제정하기 위해 약 140개 이상의 하드웨어 및 소프트웨어 관련 기관, 기업들이 결성한 산업체 컨소시엄으로 다양한 표준들을 제정하고 있다. 크로노스 그룹에서 제정한 대표적인 표준에는 Vulkan, OpenGL, WebGL, OpenCL, OpenVX, NNEF, COLLADA, glTF 등이 있다. 2019년 7월 LA에서 열리는 SIGGraph 2019 행사에서 크로노스 그룹은 OpenXR 1.0을 Web 환경에서의 가상 증강현실 표준으로 최종 인준하고, 이를 공개한다고 선포하였다. 더불어 OpenXR과 함께

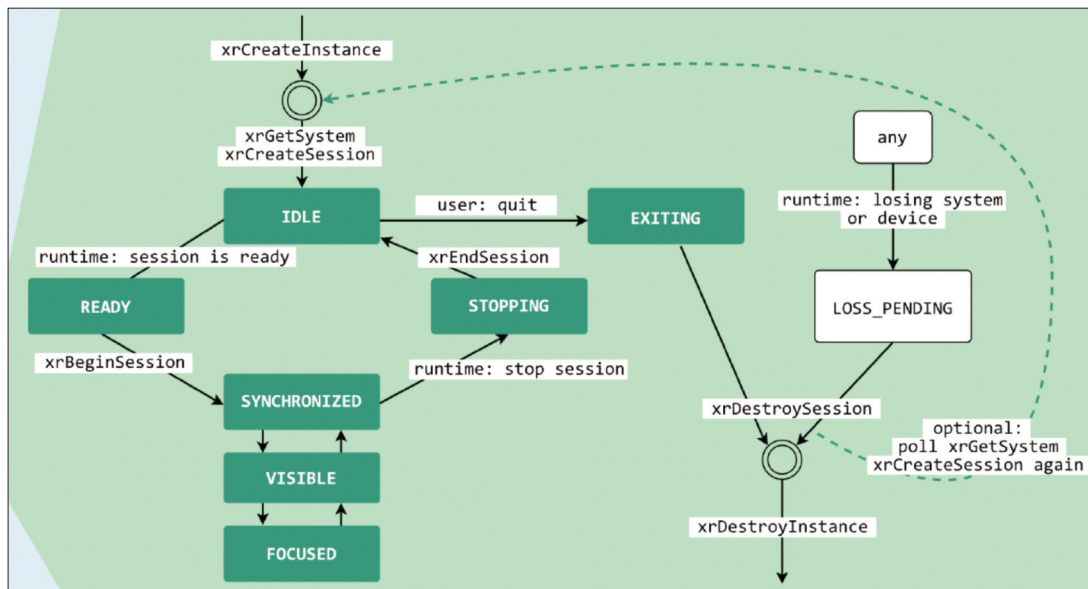


<그림 1> OpenXR 표준 흐름도

(출처 : <https://www.khronos.org/openxr/>)

사용할 수 있는 제품들을 공개하고, 가상 증강현실 생태계를 구성하는 다양한 개발 툴에 대한 내용들도 발표하였다. OpenXR은 무료로 사용할 수 있는 개방형 통합 표준으로써 가상 증강현실의 구현에 필요한 빠르고 높은 성능의 필요성을 충족하고, 서

로 다른 플랫폼 상에서 가상현실 및 증강현실을 통합하는 확장 현실(Extended Reality) 플랫폼과 장치들을 운용할 수 있도록 지원한다. 이를 통해 다양한 어플리케이션 및 구현 엔진들을 OpenXR API를 지원하는 모든 시스템에서 실행할 수 있다. 또한,



<그림 2> Open XR API lifecycle

(출처 : <https://www.khronos.org/openxr/>)

현재 확장현실 개발에 다양한 툴(라이브러리, 플랫폼 등)이 사용되고 있는 환경을 궁극적으로 통합할 수 있는 방법을 제시한다. OpenXR의 표준은 현재 1.0버전까지 개발되어있고, 이후 OpenXR 입력 하부 시스템 개선, 게임엔진 에디터 지원 및 로더 개선 등이 이루어지고 있다.

2. Web XR

WebXR Device API는 W3C 그룹에서 2018년에 처음 공개하였으며, 최근까지도 활발하게 개발 중인 API이다. 2020년 6월에는 WebXR 앵커 모듈과 WebXR 증강현실 모듈의 최신 버전이 공개되었다. 이전에는 WebVR(가상현실용)과 WebAR(증강현실용)은 API가 별도로 개발되었으나, WebXR Device API는 웹에서 가상현실 및 증강현실을 모

두 지원하고, WebXR 증강현실 모듈을 통해 증강현실에 대한 다양한 기능을 지원한다. WebXR Device API는 웹 환경에서 확장 현실을 서비스하기 위해 개발된 API로, 출력 장치의 선택을 관리하며 선택된 장치에 적절한 프레임 속도로 3D 장면을 렌더링한다. WebXR 호환 장치에는 동작 및 방향 추적 기능이 있는 완전 몰입형 3D 헤드셋, 프레임을 통과하는 실제 장면 위에 그래픽을 오버레이하는 안경, 카메라로 환경을 캡처하고 컴퓨터로 해당 장면을 확대하여 현실을 강화하는 스마트폰이 포함된다.

WebXR이 크로노스 그룹에서 개발된 OpenXR SDK와 자주 비교가 되기에 이 둘의 차이점을 살펴 보겠다. OpenXR은 네이티브 어플리케이션용 WebXR Device API와 동일한 기본 기능을 다루기에 WebXR과 OpenXR은 WebGL과 OpenGL의 매



<그림 3> WebXR 활용 예시

(출처 : https://www.youtube.com/watch?v=nGkxpBSGJI&feature=emb_logo)

칭 관계를 갖는 것처럼 보인다. WebGL과 OpenGL은 모두 크로노스 그룹에서 발표 및 관리하는 오픈 소스 라이브러리다. 하지만 OpenXR은 크로노스 그룹에서 발표 및 관리함에 비해, WebXR은 W3C 그룹에서 발표 및 관리한다. 서로 다른 표준 기관에서 개발 중인 별개의 API이기에 동일한 개념 중 많은 부분이 서로 다른 방식으로 표현된다. 그러나 확장 현실 서비스를 구축할 때, OpenXR을 사용하여 WebXR의 기능을 구현하는 것이 가능하다.

II. 주요 기술 소개 및 한계점

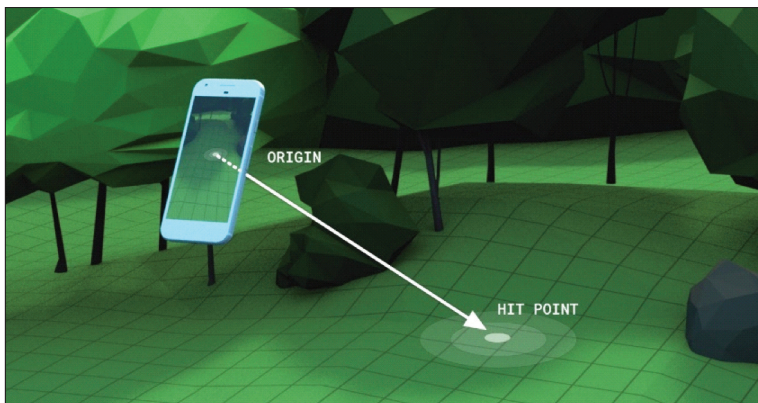
1. 주요 기술

WebXR은 웹에서 가상 환경을 표시하거나 그래픽 이미지를 실제 환경에 추가 하기 위해 설계된 하드웨어에 3D 장면 렌더링을 지원하는 데 사용되는 표준이다. WebXR 장치 API는 가상환경을 구현하는 출력 디바이스의 선택을 관리하고, 적절한

프레임 속도로 장치에 3D 장면 렌더링, 출력을 2D 디스플레이로 미러링, 입력 컨트롤의 움직임 을 나타내는 벡터 생성과 같은 주요 기능을 제공한다. 가장 기본적인 수준에서 장면은 각 눈의 위치를 계산하고 그 위치에서 장면을 렌더링 하여 사용자의 각 눈의 관점에서 렌더링 하기 위해 장면에 적용할 원근에 대한 내용을 계산하여 3D로 표현된다.

Hit-test는 컴퓨터 그래픽 프로그래밍에서 사용자가 제어하는 커서가 화면에 그려진 특정 그래픽 개체(점, 선, 면)와 교차 여부를 결정하는 과정이다. 여기서 커서는 마우스 커서나 터치스크린의 인터페이스 또는 터치 포인트와 같이 다양한 종류가 있다. Hit-test는 특정 포인팅 장치의 움직임 또는 활성화에 대해 수행할 수 있다. WebXR 환경에서는 기기의 카메라에서 광선을 뿜어 인식된 주변 환경과의 충돌 지점을 구한다. 이를 통해 바닥이나 벽, 천장과 같은 주변 환경을 인식하고 그 위에 3D 오브젝트를 표시하는 등의 방식으로 활용된다.

DOM-overlay는 WebGL 레이어 위에 DOM 엘리먼트를 렌더링 하는 기능으로, 웹 AR의 큰 장점



<그림 4>Hit-test 주변 인식 예시

(출처 : <https://codelabs.developers.google.com/codelabs/ar-with-webxr#4>)

중의 하나이다. 이를 통해 AR 세션의 GUI를 WebGL 없이 CSS + JS 이벤트 핸들러를 이용해서 간단하게 구현하거나, CSS3의 3D transform을 이용하여 비디오나 이미지 등 다양한 엘리먼트를 실제 환경 위에 표시할 수 있다.

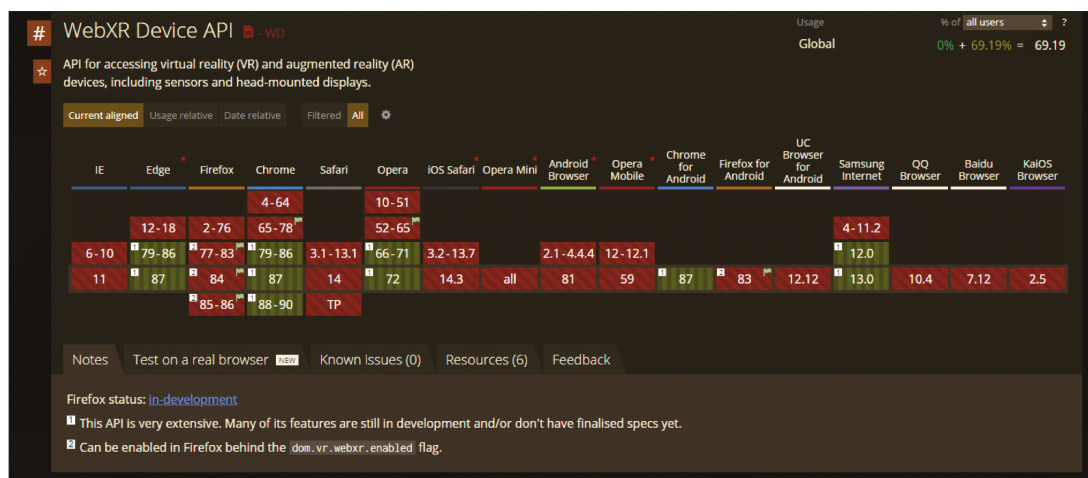
2. 한계점

현재 WebXR 적용에 가장 큰 걸림돌은 브라우저 커버리지 문제일 것이다. 현재 완벽하게 지원되는 브라우저는 Android Chrome뿐으로, 다른 브라우저에서 모든 사용자를 대상으로 WebXR 콘텐츠를 서빙하기에는 어려움이 있다. 물론 Chrome의 기반이 되는 Chromium 엔진을 사용하는 브라우저는 존재한다. 그런 브라우저에 AR 기능이 서서히 추가될 것은 자명해 보이지만, Firefox나 iOS용 브라우저의 경우 WebXR Device API가 언제 추가될지 알 수 없기에 이를 하염없이 기다려야만 하는 게 현상이다. 반면, Android 기기의 경우에도 브라우저

버전만을 충족한다고 해서 WebXR을 사용할 수 있는 것이 아니다. Chrome의 WebXR Device API 구현의 근간이 되는 ARCore는 설치 가능한 기기가 정해져 있기 때문에 모든 기기를 대상으로 WebXR을 사용하게 하는 것은 불가능하다.

III. 국내외 활용 동향

마이크로소프트와 페이스북은 각각 다수의 OpenXR 장치를 통해서 OpenXR이 다양한 플랫폼에서 이식성을 지원하고 있다. PC 환경의 가상현실 장치인 마이크로소프트 혼합현실(Microsoft's Windows Mixed Reality)과 오쿨러스 리프트(Oculus Rift)와 같은 헤드셋, HMD 그리고 추후 발표될 OpenXR 호환 장치들에서, 윈도우 환경에서 동일한 응용을 실행할 수 있을 것이다. 이와 함께 마이크로소프트는 OpenXR-호환 홀로 렌즈 헤드셋(HoloLens 2 headset)을 위한 런타임 환경을



<그림 5> WebXR Device API 기기 리커버리 정보

(출처 : <https://caniuse.com/webxr>)

공개하였으며, 페이스북은 안드로이드 기반 오쿨러스 퀘스트(Oculus Quest) 런타임 인증 제품을 출시하였다. 이들 제품들은 다양한 운영 체제에서도 다양한 종류의 독립형 혹은 테더링 XR 장치에서 VR 및 AR 응용의 호환성을 보여 준다. Valve사는 개발자 프리뷰 구현(developer preview implementation)을 SteamVR의 새로운 기능으로 발표하였다. 이를 통해 OpenVR API가 아니라 OpenXR을 통해서 SteamVR이 실행될 것으로 기대된다. 또한 Varjo사의 개발자 프리뷰 구현(developer preview implementation) 역시 유사하게 OpenXR 응용을 Varjo 헤드셋에서 사용할 수 있도록 해 준다. Collabora가 설립한 Monado 오픈소스 XR 런타임도 호환성과 기능이 향상되고 있으며 OpenXR 1.0과 호환된다. OpenXR 1.0의 인증 테스트를 마무리 지음과 동시에, OpenXR 워킹 그룹은 API 진화 과정의 하나로, 눈과 손 추적을 위한 두 개의 멀티 벤더 OpenXR 확장판을 발표하였다. 이들 새로운 확장판의 기능은 다양한 플랫폼과 벤더들을 지원하는 API를 통해 이식성을 제공하는 고급 UI 기술의 범위를 넓혀준다. 실제로 Ultraleap은 Ultraleap 추적 장치를 위한 손 추적 기술을 위한 개발자 프리뷰 OpenXR 통합을 발표하였다. Google은 Chrome 브라우저에서 WebXR 애플레이터를 통해 사용자와 개발자가 실제 XR 장치를 사용하지 않고도 데스크톱 브라우저에서 WebXR 콘텐츠를 실행하고 테스트할 수 있는 환경을 제공한다. 구글의 확장 프로그램은 아직 지원하지 않는 브라우저에서 WebXR API를 에뮬레이트하고 애플리케이션 할 컨트롤러와 함께 XR 장치 목록을 제공한다. Jeeliz사는 WebXR Device API 및 다양한 웹 환경에서의 API를 지원하고 비디오 피드에서 객체를 감지하고 정확한 위치를 제공할 수 있는 자바스크립트

라이브러리를 제공한다. 이를 통해 웹 개발자에게 실시간 컴퓨터 비전을 제공한다. 얼굴을 감지 및 추적하고 표정을 인식하거나 3D 개체를 감지할 수 있으며, WebGL을 사용하여 GPU에서 실행되는 딥러닝 엔진으로 구동된다. 모바일 장치의 약한 GPU에서도 실시간으로 비디오 스트림을 분석할 수 있는 프레임 워크를 개발하였다.

국내에서는 WebXR표준이 공개된 시점이 오래되지 않았기 때문에, 본격적으로 WebXR로 가상 증강현실을 구현하는 기업들을 손꼽을 수 있을 것이다. 공간의 파티 역시 2020년 WebXR을 활용하여 실내 도슨트 서비스를 구축하였고, 서울 스마트 시티센터에 이 기술을 국내 최초로 시범 적용하였다. 아직까지 WebXR의 모든 부분들이 공개된 것은 아니기에 많은 기업들이 쉽게 이용할 수 없는 부분들이 있고, 원하는 기능을 구현하기 위하여 WebXR만이 아닌 다양한 기술들을 혼합하여 사용해야 하는 불편함도 있지만, 앱을 설치하지 않고 웹만으로 AR을 구현할 수 있는 편리함과 가상 증강현실을 위한 기본 인프라(5G)가 구축되고 있고, 증강현실에 대한 장미빛 전망 등을 고려해볼 때, 우리나라에서도 WebXR이 표준으로 자리잡는 데에는 오래 걸리지 않을 것으로 예상된다.

IV. 결론

최근 활발하게 연구되고 있는 다양한 가상·증강현실 환경에 발맞춰 이를 위한 표준들이 공개되고 있다. 이러한 표준들을 이용하여 기존의 각기 다른 개발 환경에서 벗어나 하나의 통일된 환경에서 개발 및 서비스가 이루어질 수 있다. 즉, 개발비용도 낮아지고, 더 쉽고 편하게 사용자에게 다가가는 가

상 증강현실 산업이 발전할 수 있는 시장이 열리고 있다. 아직까지 이러한 표준들이 실험적 버전이거나 잠정 버전으로 앱에서 구현하는 것과 같은 한계점들도 존재한다. 웹에서의 가상 증강현실의 구현

이 세계적 흐름이라는 점을 고려할 때, 이러한 문제점을 분석하고 해결해 나갈 수 있는 방안을 연구하는 것이 필요하다고 사료된다.

참고 문헌

- [1] 크로노스 그룹, <https://www.khronos.org/>
- [2] WebXR Device API explainer, <https://immersive-web.github.io/webxr/explainer.html>
- [3] WebXR, 예시 <https://codelabs.developers.google.com/codelabs/ar-with-webxr#0>
- [4] Immersive Web, <https://github.com/immersive-web>
- [5] Openxr 활용동향, <https://kr.khronos.org/news/press/2020/07>
- [6] Jeeliz AR, <https://jeeliz.com/>
- [7] WebXR Device API, https://developer.mozilla.org/en-US/docs/Web/API/WebXR_Device_API

필자 소개



강진규

- UPenn 건축석사
- 고려대학교 지구환경과학과 학사
- 전) 성균관대학교 서피스학과 디지털디자인 스튜디오 겸임교수
- 세종대학교 건축학과 스튜디오 강사
- SHAPE UAE 건축설계사무소 건축디자이너
- 현재 : TheSpatialParty 공간의 파티 대표
- 주관심분야 : 3D 그래픽스, 스캐닝, 3D 공간정보 데이터, AR AE(AR Architectural Environments) 등