

Informe Técnico: Estado del arte de Realidad Aumentada (AR) en teléfonos móviles para montajes teatrales

Por [Aarón Montoya-Moraga](#) y [Guillermo Montecinos](#)

28 de marzo de 2018

Introducción

El siguiente informe técnico aborda el estado del arte de Realidad Aumentada (AR, por su sigla en inglés).

Por Realidad Aumentada se entiende una visión en vivo del mundo físico, cuyos elementos son aumentados por información generada por computador. Permite la incorporación de información digital - imágenes, datos, modelos 3D, sonido, entre otros - sobre un entorno real, mediante el uso de software y hardware que realizan reconocimiento espacial de patrones.

El objetivo de este estudio es describir el panorama actual de la AR, caracterizar las tecnologías disponibles y las posibles soluciones que se pueden lograr en función de cumplir con los requisitos del montaje teatral "Monga". Para ello se enumeran y describen las tecnologías y estrategias más comunes para desarrollar aplicaciones de AR. Adicionalmente, se revisan cuáles teléfonos móviles son capaces de implementar las tecnologías revisadas, y cuál soporte técnico se requiere para implementar la aplicación en escena.

Realidad Aumentada

La generación de experiencias AR en dispositivos móviles consiste en el despliegue de información, datos o modelos 3D virtuales en un entorno real. Para que el despliegue de esta información sea coherente con el entorno, los sistemas de AR requieren un análisis del espacio de modo de detectar planos, profundidad, bordes y otros atributos que entreguen información espacial, y el movimiento del equipo respecto de este espacio. Esto se llama odometría y corresponde al análisis de la trayectoria de puntos en el espacio.

Actualmente existen dos métodos para realizar odometría con un dispositivo móvil orientada a AR: la primera consiste en el uso de referencias, particularmente etiquetas, que pueden ser seguidas y permiten el reconocimiento de planos a través de la cámara del equipo. La segunda se denomina Odometría Visual Inercial (VIO, por su nombre en inglés *Visual Inertial Odometry*) y corresponde a un sistema más complejo que estima la posición y orientación de un equipo en el espacio, mediante el reconocimiento de planos, bordes y

puntos de anclaje a través de la cámara, sumado al análisis de la información inercial entregada por los sensores integrados en el equipo.

La calidad de la experiencia generada depende mayoritariamente de la tecnología elegida, ya que una aplicación basada en VIO al contar con mayor información espacial puede obtener muchos mejores resultados, eso sí con un costo mayor de recursos de procesamiento, mientras que una aplicación basada en etiquetas puede lograr resultados menos complejos y con una exigencia de procesamiento menor.

Las aplicaciones AR pueden ser instaladas de distintas formas en teléfonos móviles:

- Publicación en *Google Play* o *Apple App Store* para descarga del usuario.
- Instalación directa desde un computador a cada teléfono.
- Sitio web que es accedido por el usuario en el navegador de su teléfono.

Entornos de desarrollo de aplicaciones AR

Un SDK (Kit de Desarrollo de Software, en inglés *Software Development Kit*) es una plataforma que permite el desarrollo de software en un entorno particular, y que tiene una serie de complementos orientados a ciertas funciones específicas. Durante el año 2017 Apple y Google lanzaron kits de desarrollo de aplicaciones de AR en los sistemas operativos iOS y Android, respectivamente. Estos kits son exclusivos, por lo que para programar una aplicación que funcione en iOS y Android, hay que programarlas paralelamente.

Existen alternativas que permiten programar para ambos sistemas operativos y luego exportar a ambas. La más popular es Vuforia, creado por la compañía Qualcomm y en 2015 adquirido por PTC, que se ha posicionado como la plataforma de desarrollo de aplicaciones AR con más impacto en el mercado.

Apple iOS

En junio de 2017 Apple lanzó [ARKit](#), una plataforma para desarrollar aplicaciones de AR en iOS. ARKit es un sistema de Odometría Visual Inercial (VIO) incorporado en los dispositivos móviles de Apple a partir del iOS 11.0, cuya función es capturar la información del espacio en que está ubicado el equipo y procesarla para generar los inputs necesarios para el modelamiento de una experiencia AR. Es compatible con herramientas de desarrollo de modelos 3D externas como Unity y Unreal. A la fecha de entrega de este estudio ARKit se encuentra disponible en su versión 1.5 que es compatible con iOS 11.3 beta.

Google Android

En 2017 Google anunció la plataforma [ARCore](#), un SDK para desarrollar aplicaciones móviles de AR basado en la experiencia del proyecto [Tango](#), plataforma AR experimental creada por la compañía en 2015. Tras el anuncio de septiembre pasado, la versión 1.0 fue lanzada a fines de febrero de 2018 para ser descargada desde las plataformas de Google.

ARCore cuenta con un motor que analiza los datos espaciales mediante un sistema de VIO que recoge la información de la cámara del dispositivo y de los sensores inerciales integrados. Además, cuenta con un algoritmo para detectar la intensidad lumínica del espacio. ARCore es compatible con sistema operativo Android (Nougat) 7.0 o superior, como el Samsung Galaxy S7 (o superior), el Google Pixel (o superior), y otros especificados [aquí](#). Además, es compatible con softwares para desarrollo de VR como Unity y Unreal.

Multiplataforma

Vuforia

Es la plataforma de desarrollo de experiencias AR más utilizada en el mercado. Permite desarrollar aplicaciones compatibles con los sistemas iOS y Android, basado en la odometría por reconocimiento de patrones - etiquetas - a través de la cámara del dispositivo. Es compatible con Unity para desarrollo de modelos 3D, y con ARKit y ARCore. Un ejemplo de aplicación AR realizada con Vuforia y Unity puede encontrarse [aquí](#).

La licencia estándar de Vuforia tiene un costo de \$499.00, pero existe una licencia especial gratuita para desarrolladores que trabajan con [Unity](#).

Requerimientos dispositivo móvil

OS	Versión
Android	4.4+
iOS	9+

Fuente: [Vuforia Supported Versions](#)

Holokit

Es una plataforma open source de realidad mixta (MR), que permite generar experiencias inmersivas utilizando el teléfono móvil. La realidad mixta es una variante que intenta mezclar los mejores atributos de VR y AR, generalmente creando experiencias mediante el uso de visores VR en el que se puede ver el mundo real (mostrado mediante una cámara) sobre el cual se montan modelos de 3D para crear una experiencia MR.

Holokit consiste en un kit de visores de cartón en el que se monta un teléfono móvil, y un [SDK](#) basado en entornos de AR. Es compatible con ARKit y ARCore, por lo que tiene las mismas limitaciones técnicas y de costo que dichas plataformas.

Cuadro comparativo SDK

Plataforma	Empresa	Año	\$ USD	Android	iOS	Unity
AR Kit	Apple	2017	0	N/A	11+	Sí
AR Core	Google	2017	0	7.0+	N/A	Sí
Vuforia	PTC	2011	499	4.4+	9+	Sí
Holokit	Amber Garage	2017	30 (1)	7.0+	11+	Sí

(1): Costo del visor de cartón, el código fuente del SDK es open source y está disponible en GitHub.

Fuentes:

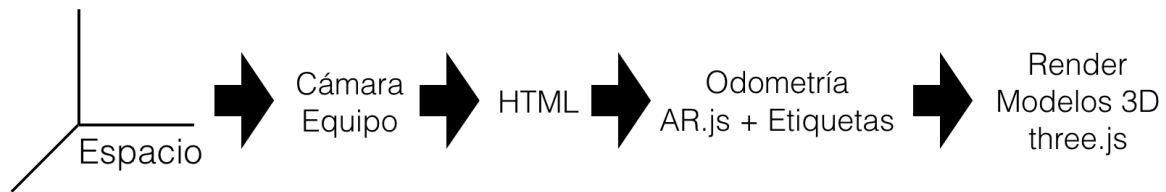
- [Versión OS Android para AR Core](#)
- [Versión iOS para AR Kit](#)
- [Versiones iOS, Android y Windows compatibles con Vuforia](#)
- [Licencia Vuforia](#)

AR en formato web

Existen dos formas de desarrollar aplicaciones de AR en la web: la primera consiste en ejecutar en HTML un script JavaScript que mediante la biblioteca AR.js realice odometría utilizando la cámara del dispositivo, mediante visión por computador (CV), para luego renderizar modelos 3D utilizando A-frame u otros entornos para crear modelos VR. La segunda consiste en utilizar las plataformas VIO ARKit o ARCore para analizar los datos espaciales, y ejecutar en un navegador adaptado a estas plataformas, ya sea WebARonARKit o WebARonARCore, un script que analice dicha información mediante la biblioteca three.ar.js para luego renderizar los modelos 3D utilizando three.js.

AR web con odometría por CV

El desarrollo de experiencias AR web con odometría por CV requiere la conexión de algunos componentes en el entorno HTML. Generalmente en estos casos la imagen es capturada mediante la cámara integrada del equipo a través de un objeto DOM de HTML. Luego esta imagen es analizada utilizando una biblioteca que realiza la odometría mediante algoritmos de visión por computación, con AR.js. Este tipo de complementos también crean objetos AR como escenas, sobre las cuales se renderizan modelos 3D utilizando plataformas como A-frame o simplemente three.js.



AR.js

Es una biblioteca de JavaScript que permite realizar aplicaciones de AR en un entorno HTML mediante odometría por seguimiento de etiquetas. Está basada en [three.js](#) y [jsartoolkit5](#) y pensada para correr aplicaciones AR web desde cualquier teléfono móvil que posea WebGL y webRTC, complementos incluidos desde la versión de Google Chrome 64 o superior. Un ejemplo de esta biblioteca es el [siguiente](#).

Ventajas

- AR.js corre en la mayoría de los teléfonos móviles
- Al utilizar odometría mediante etiquetas utiliza menor cantidad de recursos
- No requiere navegadores WebARonARCore o WebARonARKit

Desventajas

- Requiere etiquetas para anclar el modelo al mundo real
- No utiliza sensores espaciales del teléfono
- Espacialización depende sólo de la cámara del equipo
- Estabilidad en localización del modelo depende de la luminosidad del espacio

A-frame

Es un entorno HTML diseñado originalmente por Mozilla para desarrollar aplicaciones de VR en la web basada en [three.js](#). Es un entorno VR compatible con la mayoría de los [headsets](#) disponibles en el mercado, y puede ser utilizado para generar modelos 3D en aplicaciones de AR-web al incorporar la biblioteca [AR.js](#).

[A-frame](#) puede ser conectada con AR.js de modo que esta última administre la información obtenida a través de la cámara del dispositivo, basándose en odometría mediante etiquetas, de modo de dar el marco espacial para que A-frame renderice los modelos 3D, como en siguiente [ejemplo](#). Es compatible con bibliotecas de desarrollo web como [D3.js](#) y [React](#).

AR web con VIO (ARKit o ARCore)



Una forma de desarrollar experiencias AR web más complejas es utilizando los navegadores web desarrollados por Google que permiten ejecutar aplicaciones AR sobre ARKit en iOS y ARCore en Android. Estos navegadores, llamados [WebARonARKit](#) y [WebARonARCore](#) respectivamente, han sido liberados como aplicaciones móviles experimentales para desarrollo. Al ser ejecutados sobre ARKit y ARCore, este navegador exige los mismos requerimientos técnicos.

El rol de estos navegadores es conectar la información espacial procesada por los sistemas VIO con la biblioteca que analiza dichos datos, que en este caso corresponde a la biblioteca experimental three.ar.js diseñada por Google para el prototipado de aplicaciones AR web que usen ARKit o ARCore. Al igual que en AR web con odometría por CV, los modelos 3D o la información desplegada en el entorno AR se renderiza utilizando una biblioteca como three.js.

[three.ar.js](#)

Es una biblioteca de JavaScript desarrollada por Google para crear aplicaciones AR web. Cumple el rol de asistir a los objetos de la clase three.js para conectarse con la información obtenida por el sistema VIO y generar los modelos VR 3D. Los modelos generados en esta operación son conectados con los navegadores web AR mediante la [extensión WebVR](#) para smartphones, disponibles en los navegadores WebARonARKit and WebARonARCore, para luego ser desplegados en el espacio AR.

Requerimientos técnicos para montaje

El montaje de la obra "Monga" considera el uso de 50 o más dispositivos móviles conectados a una red WiFi para la descarga de una aplicación móvil o la descarga de datos a través de una aplicación o un navegador web. La siguiente tabla resume los teléfonos móviles soportados por las tecnologías Apple ARKit y Google ARCore.

Marca	Modelo	Fecha	\$ USD	OS	WiFi
Apple	iPhone 6s	2015	449	iOS 11.2.6	802.11ac with MIMO
Apple	iPhone SE	2016	349	iOS 11.2.6	802.11ac
Apple	iPhone 7	2016	549	iOS 11.2.6	802.11ac with MIMO
Apple	iPhone 8	2017	699	iOS 11.2.6	802.11ac with MIMO
Apple	iPhone X	2017	999	iOS 11.2.6	802.11ac with MIMO
Google	Pixel	2016	549	Android Oreo 8.0	802.11 a/b/g/n/ac
Google	Pixel 2	2017	649	Android Oreo 8.0	802.11 a/b/g/n/ac
Samsung	Galaxy S7	2016	469	Android Nougat 7.0	802.11 a/b/g/n/ac
Samsung	Galaxy S7 edge	2016	569	Android Nougat 7.0	802.11 a/b/g/n/ac
Samsung	Galaxy S8	2017	599	Android Oreo 8.0	802.11 a/b/g/n/ac
Samsung	Galaxy S8+	2017	689	Android Oreo 8.0	802.11 a/b/g/n/ac
Samsung	Galaxy Note 8	2017	950	Android Oreo 8.0	802.11 a/b/g/n/ac
Samsung	Galaxy S9	2018	719	Android Oreo 8.0	802.11 a/b/g/n/ac
Samsung	Galaxy S9+	2018	839	Android Oreo 8.0	802.11 a/b/g/n/ac

Como apreciamos, todos los equipos soportados por los sistemas VIO para AR utilizan el mismo protocolo WiFi para la transmisión de datos, por lo que no presentan diferencias entre ellas. Por este motivo, la transmisión de datos está limitada al router en uso.

Los routers más nuevos del mercado funcionan con dos bandas, 2.4G y 5G. La banda 2.4G es más lenta y es capaz de alcanzar mayores distancias. La banda 5G es más rápida, aunque posee un menor radio de cobertura. Se puede usar un sistema 5G con repetidores, aumentando así el radio de cobertura en el caso que sea necesario.

Los routers de marca Linksys por defecto soportan hasta 50 aparatos. Además tienen software que permite darle prioridad a ciertos dispositivos. Tienen temporizador para poder reiniciarlos cada día, semana, o mes. Los routers Asus también poseen excelentes reseñas y hay profesionales para uso en una instalación como esta.

Conclusiones

En este informe técnico se han descrito las tecnologías disponibles para el desarrollo aplicaciones de Realidad Aumentada en dispositivos móviles que corran localmente en el equipo y aplicaciones que puedan ser ejecutadas en un navegador web. Estas aplicaciones pueden desarrollarse para ser ejecutadas localmente sobre los entornos de Odometría Inercial Visual (VIO) ARCore para Google Android o ARKit para Apple iOS, las que deben

ser programadas en el código fuente de cada OS, Java o Swift, respectivamente. En este sentido se concluye que:

- Es conveniente para la creación de experiencias AR complejas desarrollar dichas aplicaciones en las plataformas AR específicas ARCore o ARKit, desarrolladas para cada sistema operativo.
- Es más eficiente programar una aplicación que corra exclusivamente en Google Android o en Apple iOS, ya que la transpilación de una plataforma a otra y el desarrollo en paralelo en ambos OS son opciones poco viables.
- En el caso desarrollar la aplicación "No apagues la luz" con Apple ARKit o Google ARCore, solamente ser ejecutada en celulares Apple con iOS 11 o Android Oreo, respectivamente.
- Es más fácil y rápido desarrollar aplicaciones AR basadas en odometría mediante seguimiento de etiquetas que programar con el enfoque VIO. Vuforia es la plataforma más utilizada en el mercado para desarrollar este tipo de experiencias.
- Si bien las aplicaciones AR basadas en odometría mediante etiquetas incurre en un menor uso de recursos del equipo y es soportada por una gran cantidad de dispositivos, la calidad y complejidad de la experiencia AR lograda es mucho menor que en el caso de las aplicaciones desarrolladas sobre sistemas VIO.
- Para todos los casos anteriores, Unity es una buena opción para el desarrollo de modelos 3D y entornos VR, ya que posee integración con Google ARCore, con Apple ArKit y Vuforia, aunque está mejor soportado para Google ARCore.

En el caso en que las aplicaciones sean diseñadas para ejecutarse desde un navegador web, pueden ser desarrolladas utilizando como soporte odométrico la biblioteca AR.js, la que alimenta plugins generadores de modelos 3D, como A-frame (basada en three.js) o three.js. AR.js utiliza odometría mediante seguimiento de etiquetas a través de visión por computador. Por otro lado, se pueden desarrollar aplicaciones que trabajen sobre los entornos VIO ARCore y ARKit, las que requieren la existencia de los navegadores experimentales para AR desarrollados por Google WebARonARKit y WebARonARCore, para ARKit y ARCore respectivamente. En este sentido se concluye que:

- El uso del sistema webAR con AR.js tiene limitaciones en cuanto a la complejidad de la experiencia AR que se pueda lograr, pero permite correr la aplicación en cualquier celular mediante odometría por seguimiento de etiquetas, la que es más fácil de implementar.
- El desarrollo de experiencias AR web basadas en los sistemas VIO puede lograr resultados más elaborados pero en un espectro acotado de equipos, ya que por un lado requiere los mismos OS que ARCore (Android 7.0+) y ArKit (iOS 11.0+), y por otro lado requiere la instalación de los navegadores WebARonARCore o WebARonARKit respectivamente, aplicaciones que no han sido liberadas oficialmente por encontrarse únicamente disponibles para desarrollo.

Finalmente todos los equipos soportados por ARCore y ArKit utilizan el mismo protocolo WiFi, por lo que la problemática de dar conexión a 50 o más equipos durante una función de

"Monga" se reduce a la velocidad de transferencia, la que depende de la banda utilizada: 2.4G o 5G. Respecto de esto se concluye que:

- En cuanto a internet, todos los celulares poseen la misma tecnología y pueden usar tanto bandas 2.4G como 5G. La banda 2.4G es más lenta y alcanza distancias más grandes. La banda 5G es más rápida pero funciona a menor distancia, aunque se pueden instalar repetidores.
- Se debe usar un router MIMO (multiple input multiple output), que permite que cada conexión no haga más lenta la conexión de los otros dispositivos. Se recomiendan las marcas Asus y Linksys que son las más avanzadas del mercado.

Referencias

- [David Bowie in Three Dimensions - New York Times](#): experiencia AR hecha con Apple ARKit.

Artículos de interés

- [Realidad Aumentada. Wikipedia](#)
- [Web-Powered Augmented Reality: a Hands-On Tutorial. Uri Shaked](#)
- [Web AR with smart plugs using A-Frame. Gemma Vincent](#)
- [How to create Augmented Reality posters with Unity & Vuforia. Charlie Gerard](#)
- [How is ARCore better than ARKit?. Matt Miesnieks](#)
- [Why is ARKit better than the alternatives?. Matt Miesnieks](#)
- [A \\$30 AR Headset?](#)

Teléfonos

- [Apple iPhone SE specs](#)
- [Apple iPhone SE wiki](#)
- [Apple iPhone 6s specs](#)
- [Apple iPhone 6s wiki](#)
- [Apple iPhone 7 specs](#)
- [Apple iPhone 7 wiki](#)
- [Apple iPhone 8 specs](#)
- [Apple iPhone 8 wiki](#)
- [Apple iPhone X specs](#)
- [Apple iPhone X wiki](#)
- [Google Pixel specs](#)
- [Google Pixel wiki](#)
- [Google Pixel 2 specs](#)
- [Google Pixel 2 wiki](#)
- [Samsung Galaxy S7](#)
- [Samsung Galaxy S7 specs](#)
- [Samsung Galaxy S7 wiki](#)
- [Samsung Galaxy S8](#)

- [Samsung Galaxy S8 specs](#)
- [Samsung Galaxy S8 wiki](#)

Routers

- [Apple Airport Extreme](#)
- [Linksys Velop Whole Home Mesh Wi-Fi](#)
- [Linksys - 10 consejos](#)