



UNIVERSIDAD
POLITECNICA
DE VALENCIA



Máster Universitario
en Tecnologías, Sistemas y
Redes de Comunicaciones

Creación de Interfaces de Realidad Virtual Móvil en Unity 3D y Validación Experimental de sus Requisitos de Red

Autor: Danaisy Prado Alvarez

Director 1: Jose Francisco Monserrat del Río

Director 2: David García Roger

Fecha de comienzo: 26/09/2016

Lugar de trabajo: Grupo de Comunicaciones Móviles del iTEAM

Objetivos

El objetivo general de esta Tesina de Máster es determinar valores críticos de latencia y ancho de banda de la red móvil para soporte de aplicaciones VR (*virtual reality*, realidad virtual) y evaluar la calidad de la experiencia VR percibida por el usuario para dichos valores críticos. Este objetivo general se puede desarrollar en los siguientes objetivos específicos:

- Configurar las gafas de VR para que apliquen diferentes valores de latencia dentro de un rango determinado.
- Diseñar un sistema de realidad virtual embebido en UNITY-3D.
- Implementar un mundo virtual a partir de información real de un escenario con movilidad.
- Diseñar un procedimiento para estimar el ancho de banda necesario para transmitir contenido VR en directo con una tasa de bit determinada a través de la red móvil.
- Evaluar la calidad de experiencia percibida para diferentes latencias, parámetros de resolución y tasa de codificación.

Metodología

Para alcanzar los objetivos planteados en esta Tesina de Máster se diseñaron dos procedimientos experimentales. En la realización de ambos experimentos se aplicaron los métodos subjetivos de evaluación de la QoE (*Quality of Experience*, calidad de experiencia) descritos en la recomendación ITU-T P.913 [25].

Para realizar el Experimento I fue necesario identificar previamente las fuentes de latencia en las aplicaciones VR y realizar una selección de las herramientas a utilizar, en este caso, Unity 3D y el *kit* VR Oculus Rift. Posteriormente, a partir de un entorno 3D utilizado en el proyecto METIS II, se creó una interfaz de usuario adaptada para VR, así como reglas para interactuar con dicha interfaz a través de controles. Llegados a este punto, se procedió a la estimación experimental de la QoE percibida.

Para realizar el Experimento II, se seleccionó un vídeo 360° de YouTube que cumpliera con los requisitos de calidad especificados y estuviera disponible en toda la gama de resoluciones. El vídeo seleccionado a tal fin fue descargado en sus diferentes resoluciones y se estimó el consumo de ancho de banda en caso de que el vídeo en cuestión fuera transmitido en *streaming*. Posteriormente los participantes visualizaron el vídeo a diferentes resoluciones, mediante las gafas Samsung Gear VR y el teléfono móvil Samsung S7, y evaluaron la QoE.

Desarrollo de prototipos y trabajo de laboratorio

Para el Experimento I ha sido necesario desarrollar un proyecto con el software Unity 3D. El proyecto consiste en programar, mediante código software adicional, para que una herramienta software desarrollada en el marco del proyecto europeo METIS II funcione bajo VR y pase a soportar los requisitos exigidos por una interfaz VR. Para ello se introdujeron las llamadas necesarias a bibliotecas software disponibles para interactuar con VR y se incorporaron una serie de objetos y scripts que substituyen la interfaz clásica de ratón por una interacción natural del usuario con el mundo virtual a través de controles nativos VR. En el Experimento II se obtuvieron vídeos prototípicos codificados a diferentes tasas y resoluciones.

Resultados

A partir de los experimentos realizados se pudo comprobar que las aplicaciones VR demandan estrictos requisitos de latencia, pues para que sea soportable su uso en términos de comodidad del usuario la latencia no

debe superar los 20 ms como máximo. Estos requisitos imposibilitan el desarrollo de dichas aplicaciones sobre las redes móviles actuales, aunque están muy lejos del objetivo de 1 ms marcado para las redes 5G. Por otra parte, contenidos con mayores resoluciones requieren anchos de banda más elevados. Además, se ha visto que las tecnologías inmersivas no solo demandan redes con mejores prestaciones, sino también mejoras en la parte de procesado y en los dispositivos móviles. Asimismo, mediante el análisis de la información recopilada en las encuestas, ha sido posible evaluar la QoE percibida por los usuarios bajo diferentes condiciones de la red y hacer un perfil estadístico de la proporción de usuarios que tendrán una opinión satisfactoria de la aplicación de VR en dichas condiciones

Líneas futuras

Una vez acabada esta tesina de máster, y puesto que recientemente se ha adquirido un sistema completo de LTE, se plantea la opción de desplegar dicha red y experimentar una ejecución real de experiencia de VR móvil y en la nube. Esto servirá para determinar si los efectos del jitter agravan la experiencia y hacen que la latencia tenga que ser menor en comparación con el caso de jitter constante, que es el utilizado en este trabajo.

Publicaciones

[1] Danaisy Prado-Alvarez, David García-Roger, Jose F. Monserrat, “5G Network Resources Requirements for Mobile Immersive Digital Environments”, INNOV 2017, Atenas, Octubre 2017.

Resumen

El incremento acelerado en la adopción de tecnologías inmersivas como la realidad virtual o los vídeos 360°, aumenta la presión sobre las redes móviles celulares al demandar menores latencias y elevados anchos de banda para garantizar una experiencia de usuario grata y de calidad. En la presente Tesina de fin de Máster se hace una estimación de los valores críticos de latencia y ancho de banda de la red móvil para el soporte de aplicaciones VR a una QoE determinada en entornos de realidad virtual Oculus Rift y Samsung Gear VR. Asimismo, se evalúa la calidad de experiencia percibida por un grupo de usuarios para diferentes parámetros de resolución y tasa de codificación. La conclusión es que el cumplimiento de los estrictos requisitos de latencia máxima (< 20 ms) supone un importante desafío a solucionar por las futuras redes móviles celulares 5G.

Abstract

The accelerated increase in the adoption of immersive digital technologies like virtual reality and 360-degree video escalates the pressure on mobile cellular networks. Its higher bandwidth demands and minimum latencies are crucial for enjoying the contents with a satisfactory quality and comfort. This master thesis describes a study that estimates the minimum critical bandwidth, and latency prerequisites, as well as video resolutions and bitrate needed in a mobile network so as to support immersive applications with a specific subjective quality of experience, using the consumer-ready hardware platforms Oculus Rift and Samsung Gear VR. One of the main conclusions is that the strict requirements of around 20 milliseconds of minimum latency highlights the important challenge that must be addressed by future 5G mobile cellular networks.

Autor: Danaisy Prado Alvarez, email: dapraal@teleco.upv.es

Director 1: Jose Francisco Monserrat del Río, email: jomondel@iteam.upv.es

Director 2: David García Roger, email: dagarro@iteam.upv.es

Fecha de entrega: 6 de Septiembre de 2017

ÍNDICE

I. Introducción	4
I.1 Motivación	4
I.2 Estado del arte de la realidad virtual	7
I.3 Objetivos de la Tesina de Máster	9
II. Latencia en VR	10
II.1 Fuentes de latencia.....	10
II.2 Realidad virtual y su generación con Unity 3D	11
II.2.1 Habilitando el soporte VR de Unity	11
II.2.2 Qué pasa cuando VR está habilitado	11
II.2.3 Oculus + Unity (para Windows).....	12
II.2.4 Requisitos del PC.....	12
II.2.5 Especificaciones Oculus Rift	13
II.3 Desarrollo del entorno 3D.....	14
II.3.1 Visualización de las interfaces de usuario en modo VR.....	15
II.3.2 Interacción con controles	16
II.4 Configuración para el experimento I (Oculus Debug Tool).....	18
III. Determinación del ancho de banda necesario en la VR móvil en tiempo real	21
III.1 Herramientas utilizadas y procedimiento experimental	23
III.2 Configuración para el experimento II.....	24
IV. Resultados y discusión	25
IV.1 Experimento I.....	25
IV.2 Experimento II.....	27
V. Conclusiones y líneas futuras	29
Agradecimientos	30
Bibliografía	30

I. Introducción

El vertiginoso desarrollo de las tecnologías asociadas a la Realidad Virtual (VR, *Virtual Reality*) ha hecho posible que en los últimos años se hable de la misma como algo tangible y no como elemento futurista que aparece solo en películas de ciencia ficción. La VR es un entorno creado computacionalmente que imita la realidad. En él, el usuario puede disfrutar de una experiencia donde se sumerge del modo más completo posible e interactúa con el mundo virtual que le rodea como si fuera real. La VR tiene tres características principales:

- Inmersión, el usuario solo percibe los estímulos generados en el entorno virtual;
- Interacción, el usuario es capaz de interactuar con el entorno virtual, el objetivo final es que se corresponda el tiempo real con el tiempo virtual;
- Imaginación, a través del entorno virtual el usuario es capaz de concebir realidades que no existen.

La VR, un concepto inicialmente popularizado en el mundo de los videojuegos ha ganado relevancia en sectores como la medicina, la arqueología, la creación artística, el entrenamiento militar, simulaciones de vuelo, u oficinas virtuales, entre otros. Cuando se trabaja de forma inalámbrica y en movilidad, la imagen VR se transmite al visor desde una entidad remota. Entonces, para garantizar una buena experiencia de usuario en los escenarios anteriormente citados, se hace imprescindible tener en cuenta si la red inalámbrica es capaz de satisfacer las necesidades de conectividad del servicio. Así pues, diversos parámetros de las redes de comunicaciones, como pueden ser la latencia y el ancho de banda disponible, deberán estar garantizados. Estimar cuáles serían esos valores críticos constituye el principal objetivo de la presente Tesina de Máster.

1.1 Motivación

En los últimos años, la disponibilidad en la nube, tanto de contenidos como de recursos técnicos, ha permitido a los usuarios de la red disfrutar de una capacidad de procesamiento que se extiende más allá de las prestaciones limitadas de sus dispositivos, lo cual ha abierto enormes posibilidades para el desarrollo de la VR. Sin embargo, la extensión a un servicio inalámbrico en movilidad se ha tenido que posponer por las restricciones existentes tanto del ancho de banda de las redes móviles como de los períodos de latencia en la transferencia de datos desde y hacia los servidores en la red.

El siguiente esquema explica el procedimiento desde la adquisición de vídeo en el escenario del evento hasta que el mismo llega al usuario final. Cada una de las etapas será explicada con más detalle en el [capítulo III](#).

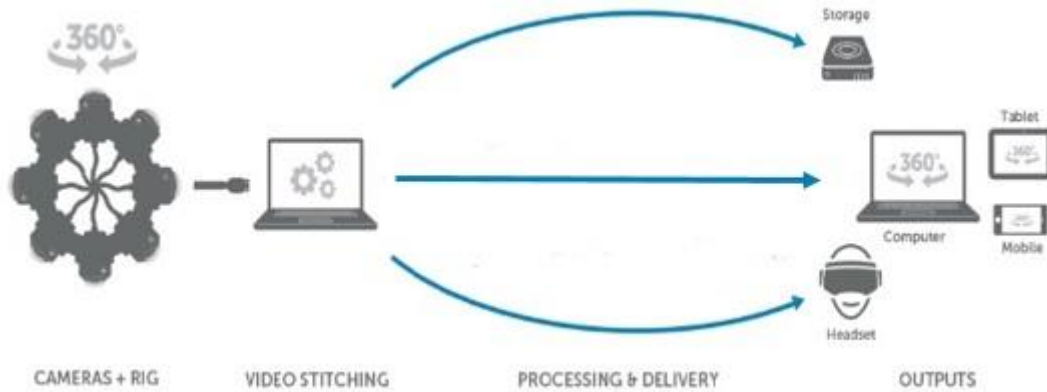


Fig.1. Esquema de transmisión VR [1].

La latencia experimentada en la etapa de *video stitching* y la latencia intrínseca de las redes móviles actuales hacen imposible que se pueda hablar de VR en tiempo real. Así, la tolerancia máxima de las aplicaciones VR para tener una experiencia confortable es de 20 ms de latencia [2] y en el esquema anterior, solo el procesado de vídeo, supone un tiempo de entre 25 y 35 ms. Tampoco es posible tener una experiencia VR completamente móvil y de calidad, pues la red 4G con una latencia de 50 ms [3] no cumple con los requisitos necesarios. Esto constituye un punto sensible pues son precisamente los dispositivos móviles aquellos que resultan más apropiados para el disfrute masivo de contenidos VR, ya que resultan más asequibles comparados con otros como los HMDs (*Head-Mounted Displays*). Llegados a este punto, se hace evidente que para conseguir hablar de *live VR streaming* es imprescindible contar tanto con novedosas soluciones en el procesamiento de vídeo como con redes que ofrezcan mayores prestaciones. El presente trabajo pretende centrarse en aspectos relativos a la red móvil.



Fig.2. Ejemplo de servicio VR móvil.

Si bien es cierto que los servicios VR móviles aún están en un estado embrionario, foros de investigación acerca de la futura red 5G, como el proyecto Europeo METIS [4], citan como caso de estudio aplicaciones donde se hace necesaria la VR en movilidad. En [4] se describe un espacio de oficinas con ambiente moderno que se encuentra situado en el interior de un edificio del siglo XIX reformado, clasificado como patrimonio cultural. El edificio es alquilado por una empresa que trabaja con tele-presencia 3D y VR. El trabajo implica interacción con escenas 3D de alta resolución y se realiza típicamente en equipos de 5 a 10 individuos interactuando simultáneamente con una misma escena. Algunos de los miembros del equipo están ubicados dentro del edificio, otros están trabajando remotamente desde otros edificios de oficinas. Cada escena puede incluir una representación de los miembros del equipo o caracteres (avatares) y elementos generados computacionalmente. La resolución de alta calidad de la escena proporciona una experiencia evocadora de la realidad, o como se conoce en inglés *as if you were here feeling*. Como cada miembro del equipo puede afectar la escena, todos deben actualizar continuamente la escena transmitiendo datos a los demás. Para proporcionar la interacción en tiempo real, esta experiencia de conectividad 5G se soporta por flujos bidireccionales de muy alta tasa de transmisión y latencias bajas.

Siguiendo con el análisis de requisitos que marca la 5G para las redes que soporten VR [4], los usuarios deberán ser capaces de experimentar una tasa binaria de transmisión sostenida de, al menos, 1 Gbps en la comunicación con otros miembros del equipo y con los servidores de red. Con el fin de sincronizar grandes cantidades de datos, se deberá ofrecer una tasa de transmisión de usuario aún mayor, en concreto de 5 Gbps, aunque se puede aceptar una menor disponibilidad para este caso de uso. La latencia promedio de paquete no debe ser mayor de 10 ms. Es decir, el tiempo desde que un paquete se envía hasta que se recibe un ACK no debe exceder 10 ms en promedio. Además, se espera una tasa de pérdida de paquetes inferior al 5%.

Viendo los requisitos marcados para la 5G, se puede concluir que las tecnologías inalámbricas de hoy en día no son capaces de proporcionar, a un coste razonable, los altos requisitos de capacidad y transmisión de datos que este tipo de aplicaciones plantea.

Es por ello que resulta imprescindible contar con una evaluación de Calidad de Experiencia (*QoE*, *Quality of Experience*) para aplicaciones VR inalámbricas operando en las actuales redes móviles. Se pretende evaluar hasta qué punto estos requisitos de tasa binaria y latencia son imprescindibles para que el usuario tenga una experiencia satisfactoria, o si por el contrario, la limitación principal está en las capacidades de visualización del propio terminal. Esto daría una medida de dónde estamos y cuánto se debe mejorar para disfrutar de una VR lo más realista posible.

¿Es de verdad la red celular móvil un impedimento para la explosión del servicio de VR móvil o se deberá prestar más atención a los procesos de grabación/codificación y a las capacidades de los propios dispositivos móviles?

A la pregunta anterior se le dará respuesta en la presente Tesina de Máster.

1.2 Estado del arte de la realidad virtual

A pesar de su desarrollo acelerado y prometedor, en la VR todavía existen varios problemas, y algunos muy importantes, que requieren de una investigación exhaustiva en los próximos años [5].

Entre las cuestiones persistentes destaca, por ejemplo, el conflicto de *Accommodation-Vergence* [6], [7]. Las pantallas estereoscópicas crean conflictos entre la distancia a la cual los ojos deben converger y la distancia a la que los ojos deben acomodarse. Tales conflictos requieren que el espectador supere la distancia usual a la que esto ocurre, y este esfuerzo puede causar molestias al espectador [8]. Huang et al.[9] han demostrado recientemente que es posible reducir este problema, pero su solución se produce a expensas de la resolución y la velocidad de transmisión de los fotogramas.

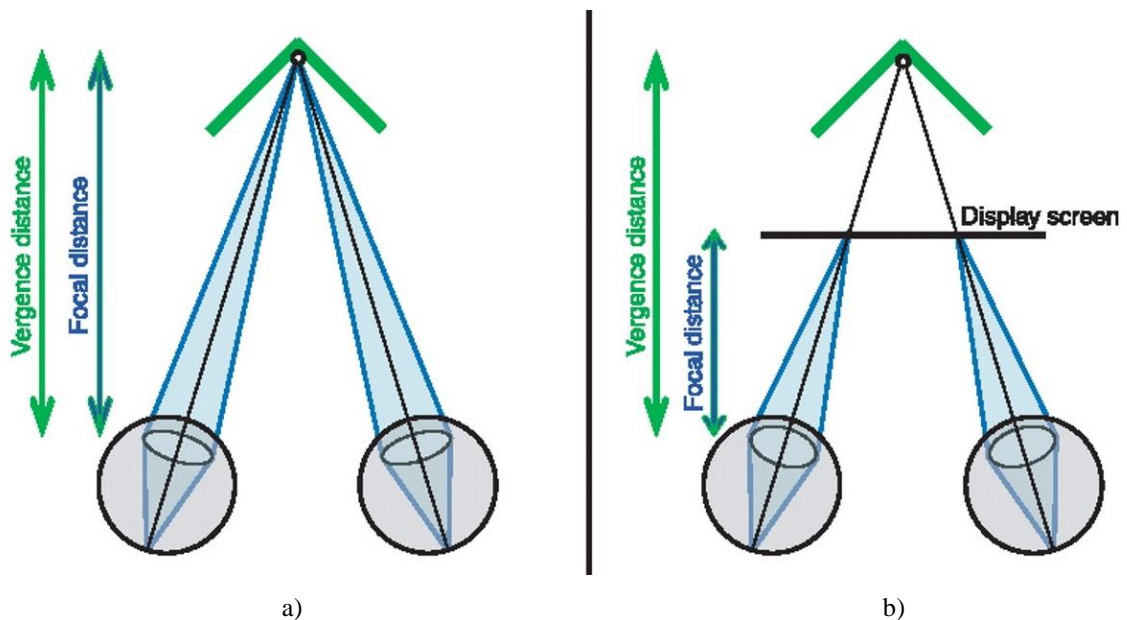


Fig.3. Conflicto Convergencia – Acomodación: a) Mundo real, b) Mundo 3D [6].

Por otro lado, la representación del usuario en HMDs es a veces problemática, ya que el motor del servidor de VR no tiene suficiente información sobre la postura completa del usuario. En particular, los usuarios se quejan de ver sus manos y no poder moverlas. Se están realizando algunos avances con el uso de cámaras, la estimación de pose, etcétera, pero las soluciones actuales distan de ser satisfactorias. En algunos casos, el problema se evita usando la vista de tercera persona, por lo que el usuario tiene una vista incorpórea divina de la escena, como si estuviera flotando.

Otra área que también representa un problema importante es el *Cybersickness*. A menos que la latencia sea muy baja (entre 7 ms y 15 ms, según criterios más conservadores [10]), los usuarios se sentirán incómodos cuando muevan la cabeza. Mover a los usuarios dentro del mundo virtual también puede causar sensación de mareo, de manera similar a lo ocurrido al viajar en un coche con los ojos vendados.

La teoría generalmente aceptada explica que el mareo viene causado por la disparidad entre la

información sobre la gravedad y la rotación procedente del vestíbulo del oído interno y la información visual procedente de los ojos. Aparte de eso, hay otra opinión que considera que la información táctil, percibida como única información sensorial, proveniente del mundo real podría contribuir adicionalmente a la sensación de malestar [11]. Poco se ha hecho hasta la fecha para cuantificar o intentar resolver los problemas de mareo en la inmersión virtual. Dentro de las pocas contribuciones existentes, en [12] se propone un sistema de prueba que permite a los usuarios evitar el mareo por movimiento en el momento en que se producen los contenidos VR. Dicho sistema evalúa el grado de riesgo de sufrir sensación de mareo por parte del usuario y genera una advertencia adecuada a modo de imagen, la cual aparece en el HMD.

Pasando a aspectos más tecnológicos, en el último año la tecnología de exhibición de HMD se ha vuelto asequible y ha mejorado significativamente. En el ámbito de los dispositivos de entrada, los enfoques parecen prometedores. Las pantallas para dispositivos móviles se supone que alcanzarán la resolución de 11k en 2018 [13]. Mientras que el ojo no puede resolver esa resolución en un uso convencional de dispositivos móviles, en el caso de aplicaciones VR se requieren pantallas de incluso mayor calidad, pues la óptica de aumento hace que los píxeles sean visibles, lo que crea una sensación de baja resolución en el visionado de grabaciones reales.

Algunas cuestiones que requieren mayor investigación y desarrollo respecto al *hardware* incluyen: aumento de la resolución, mejora de la óptica y añadir capacidades de posicionamiento de lentes por usuario. El paso a una conectividad inalámbrica y en la nube, donde el dispositivo de VR se limita al propio visor, también mejorará las limitaciones actuales con los cables que implica los peligros de tropezar y los ordenadores grandes que limitan el movimiento. Sin embargo, se necesitarán algunas mejoras en los procesadores móviles para poder hacer frente a la carga computacional necesaria.

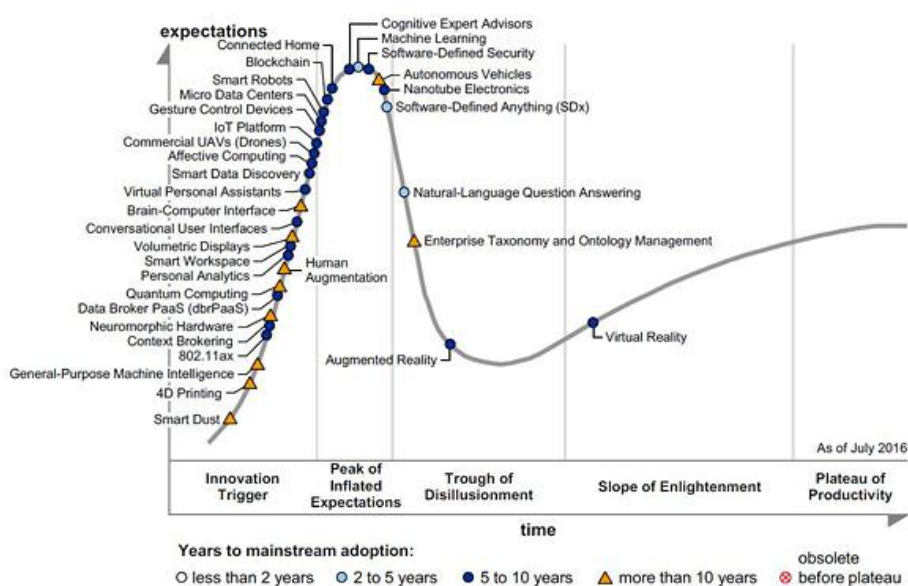


Fig.4. Hype Cycle para Interfaces Humano-Máquina, 2016 [14].

Haciendo un análisis del *hype cycle* de Gartner, para Interfaces Humano – Máquina [14], se puede llegar a importantes conclusiones sobre el estado actual de la VR. En la Fig.4. se observa que las tecnologías de VR se encuentran en la fase “Rampa de Consolidación” (*Slope of Enlighment*) y se estima que en un horizonte temporal de 5 a 10 años alcancen el estado de “Meseta de Productividad” (*Plateau of Productivity*) y los productos sean accesibles para usuarios comunes. Este marco temporal de 5 años cuadra asimismo con el desarrollo de la 5G, por lo que se espera que ambos ámbitos sean coetáneos y altamente interdependientes. Sin la 5G es probable que la VR no tenga un uso masivo y alcance al gran público. Por otra parte, será probablemente la VR la que motive la adopción de la nueva tecnología 5G, como su principal aplicación.

1.3 Objetivos de la Tesina de Máster

Estamos presenciando un crecimiento de aplicaciones que utilizan dispositivos móviles, como redes sociales, tutoriales virtuales, juegos en línea multijugador masivos y realidad aumentada (RA, *Reality Augmented*), por mencionar sólo algunos. Además, las aplicaciones basadas en los entornos virtuales (VE, *Virtual Environment*) han atraído recientemente a un gran número de usuarios. Las aplicaciones que aplican la arquitectura cliente-servidor convencional requieren que el VE se almacene en el cliente, lo cual no resulta normalmente viable debido a las limitaciones de almacenamiento del cliente [15].

Una alternativa a este problema es la transmisión de contenidos en *streaming*, donde los recursos de almacenamiento se encuentran en la nube y se reproducen directamente en el terminal sin necesidad de descarga previa. En esta dirección, existen varios aspectos que requieren de mayor investigación, sobre todo en lo que respecta a la QoE del usuario. Entre ellos, determinar el ancho de banda necesario para realizar dichas transmisiones y el nivel de latencia en la red de comunicaciones tolerable por las aplicaciones de *360-degree VR* para que el usuario no perciba ninguna degradación en su experiencia.

El objetivo de esta Tesina de Máster es, por tanto, crear un puesto experimental de VR móvil, incluyendo la programación de una escena de inmersión utilizando Unity 3D, para luego determinar experimentalmente los valores críticos de ancho de banda y latencia de la red móvil para el soporte de aplicaciones VR y ver la QoE percibida por los usuarios para dichos valores.

Este objetivo se puede descomponer en los siguientes objetivos específicos:

- Desarrollar un mundo virtual a partir de información real de un escenario con movilidad en Unity 3D.
- Configurar el Software Oculus para experimentar diferentes latencias en el entorno virtual creado.
- Evaluar la QoE para las diferentes latencias en el entorno virtual creado.
- Diseñar un procedimiento para estimar el ancho de banda necesario para transmitir contenido VR en vivo a determinada tasa de bits a través de la red móvil.

- Evaluar la QoE de contenidos VR a diferentes tasas y resoluciones.

Para determinar los valores críticos de funcionamiento y la QoE percibida por los usuarios ante dichos valores se va a plantear la realización de dos experimentos. El primero de ellos consiste en la evaluación de QoE percibida, por un grupo de personas aleatoriamente escogido, ante la existencia de distintas latencias durante el disfrute de una experiencia VR. En este caso ha sido generado un escenario 3D con interfaces asociadas con las cuales el usuario puede interactuar. En el segundo experimento se realiza una estimación del ancho de banda necesario para transmitir en *streaming* un vídeo VR 360°, conociendo la tasa a la que ha sido codificado. A cada tasa de codificación se le asocia una resolución de vídeo. El mismo grupo de personas evaluará la QoE al visualizar el vídeo a diferentes resoluciones.

II. Latencia en VR

Los sistemas VR tienen el potencial de revolucionar la forma en que interactuamos con los ordenadores. Sin embargo, el mareo y el malestar causado por el movimiento están actualmente impidiendo su rápida adopción. Incluso, en algunos casos, personas que se han iniciado en su uso lo abandonan por la incomodidad asociada a los problemas de latencia y sus mareos consiguientes.

Tradicionalmente, la optimización de los sistemas VR se ha centrado en mejorar la tasa de fotogramas. Sin embargo, el retardo y la velocidad de fotogramas no son equivalentes. La latencia puede ocurrir en varios pasos en el procesamiento de imágenes, y una medida de velocidad de fotogramas sólo capta algunos de ellos [16].

II.1 Fuentes de latencia

¿Cuánta latencia es demasiada? Sobre la base de la investigación VR durante la década de 1990, 60 ms se ha citado comúnmente como un límite superior para VR aceptable. No obstante, definitivamente se nota un retraso incómodo cuando la latencia es mayor de 40 ms [2]. Algunas fuentes aseguran que si la latencia es inferior a 20 ms [17], entonces este retraso ya no es perceptible. Aun así, existen criterios más conservadores. Haciendo una búsqueda bibliográfica se han encontrado algunos trabajos relacionados con este tema [18], [19], [20]. Sin embargo no se ha hallado, hasta la fecha, ningún estudio exhaustivo sobre un escenario VR dinámico en que se pruebe de manera experimental el límite de latencia necesario para una muestra significativa de sujetos.

Cuando hablamos de latencia se están incluyendo los siguientes parámetros: tiempo de sensado (ver [II.2.5 Especificaciones Oculus Rift](#)), llegada de datos por USB, combinación de datos sensoriales, simulación de juegos, renderizado y salida de vídeo. Otros factores, como los tiempos de conmutación de píxeles de LCD, la complejidad del juego y el almacenamiento involuntario de fotogramas en el búfer, pueden aumentarla, pero es importante señalar que el período de latencia es generalmente más corto que hace décadas [2].

II.2 Realidad virtual y su generación con Unity 3D

Unity 3D es uno de los motores de juegos más potentes que existe en la actualidad [21]. Su versatilidad para compilar en diferentes plataformas (móviles, de VR, de escritorio, consola y de TV, además de la Web) y la facilidad para obtener la licencia de operación, han asegurado el éxito de esta aplicación que está detrás de un gran porcentaje de los juegos que actualmente se están lanzando al mercado. Unity 3D permite programación utilizando lenguajes como Javascript3, Boo y C#, y dispone de un editor que agiliza el desarrollo del videojuego. También posee una amplísima comunidad, que ayuda a resolver dudas y crea *plugins* para cubrir aspectos puntuales del desarrollo, lo cual constituye un apoyo valioso para los desarrolladores.

La VR inicialmente se soportó en Unity a través de *plugins* externos. Esto tiene ciertos defectos [22]:

- Cada dispositivo VR tiene un *plugin* diferente.
- Los *plugins* pueden entrar en conflicto el uno con el otro.
- Cambiar entre *plugins* de VR para soportar múltiples dispositivos implica una gran cantidad de trabajo extra.
- Cada lanzamiento nuevo SDK (*Software Development Kit*) de VR puede corromper juegos antiguos.
- Cambiar entre modo VR y no VR no es trivial.
- Optimizaciones de bajo nivel del motor no son posibles.

Sin embargo, Unity solventó ya en la versión 5.5.1p2 la extensión de sus juegos a un escenario de VR. En la actualidad, la VR de Unity incorpora prestaciones a Unity dedicadas exclusivamente a dispositivos VR y elimina la necesidad de utilizar *plugins* externos. Además, Unity proporciona una interfaz básica para programación de aplicaciones con el objetivo de ir mejorando la compatibilidad con dispositivos y softwares de manera progresiva.

II.2.1 Habilitando el soporte VR de Unity

Para habilitar el soporte VR en un proyecto de Unity se debe habilitar la opción “Virtual Reality Supported” en Player Settings de la siguiente forma: abrir el menú, **Edit -> Project Settings -> Player**, seleccionar **Other Settings** y marcar la casilla de verificación **Virtual Reality Supported**.

II.2.2 Qué pasa cuando VR está habilitado

Cuando se habilita el modo VR, se activan automáticamente las siguientes funcionalidades:

a) Visualización estereoscópica automática

No se requiere tener dos cámaras para visualizaciones estereoscópicas. Cualquier cámara que no tenga una textura renderizada es automáticamente renderizada en estéreo para el dispositivo.

Las matrices de vista y proyección se ajustan para tener en cuenta el campo de visión y el seguimiento de la cabeza. Además, se optimiza el funcionamiento de la herramienta para que sea menos costoso dibujar cada *frame* dos veces (una para cada ojo).

b) *Input headtracked automático*

Un *Head Tracking* (Seguimiento de Cabeza) y un Campo de Visión (FOV, *Field of View*) adecuado se aplican automáticamente a la cámara (si el dispositivo está montado en la cabeza).

II.2.3 Oculus + Unity (para Windows)

A continuación se citan los pasos a seguir para poner a punto el binomio Oculus + Unity en un ordenador con sistema operativo Windows. Hay que tener en cuenta los requisitos para el ordenador que se mencionan en el apartado [II.2.4 Requisitos del PC](#).

1. Instalar Oculus Runtime
2. Reiniciar
3. Abrir Oculus Configuration Utility de la bandeja del sistema y conectar su Oculus Rift Development Kit 2. En caso de no tener usuario Oculus se debe agregar uno nuevo.
4. Abrir Unity y habilitar el soporte VR.
5. Al ejecutar el proyecto se debe observar en el HMD un *skybox* completamente guiado por la cabeza.

II.2.4 Requisitos del PC

Para poder disfrutar de todas las prestaciones de Oculus Rift, se recomienda que el PC reúna o supere los siguientes requisitos del sistema.

Tabla 1. Requisitos del PC

Característica	Recomendación	Mínimo	Utilizado
Tarjeta gráfica	NVIDIA GTX 1060/AMD Radeon RX 480 o superior	NVIDIA GTX 1050 Ti/AMD Radeon RX 470 o superior	GeForce GTX 1070 G1 Gaming 8GB GDDR5
Tarjeta gráfica alternativa	NVIDIA GTX 970/AMD Radeon R9 290 o superior	NVIDIA GTX 960 4GB/AMD Radeon R9 290 o superior	...
CPU	Equivalente a Intel i5-4590 o superior	Intel i3-6100/AMD FX4350 o superior	Intel i7 6700K
Memoria	8 GB de RAM o más	8 GB de RAM o más	32 GB de RAM
Salida de vídeo	Salida de vídeo HDMI 1.3 compatible	Salida de vídeo HDMI 1.3 compatible	Salida de vídeo HDMI 1.3

Puertos USB	3 puertos USB 3.0 más 1 puerto USB 2.0	1 puerto USB 3.0 más 2 puertos USB 2.0	1 puerto USB 3.1, 6 puertos USB 3.0, 4 puertos USB 2.0
SO	Windows 8.1 o posterior	Windows 7 SP1 de 64 bits o posterior	Windows 10 Enterprise

II.2.5 Especificaciones Oculus Rift

Oculus Rift es el último HMD desarrollado por Oculus. A través de un sistema de sensores de seguimiento y leds infrarrojos se identifica la posición del usuario mientras interactúa en el mundo virtual. El usuario se puede mover siempre que su entorno físico lo permita, sin embargo, dichos movimientos son limitados pues el HMD se encuentra conectado al ordenador mediante un cable. Se han diseñado diferentes controladores para facilitar la interacción y hacer más agradable la experiencia, entre ellos, el *Xbox One*, el *Remote Controller* y los *Touch Controllers*. A continuación se listan algunas de sus especificaciones:

- Display: OLED
- Resolución: 2160x1200
- Tasa de refrescamiento: 90 Hz
- Plataforma: Oculus Home
- Campo de visión: 110 grados
- Área de seguimiento: 5,11 m²
- Peso: 470 gramos
- Controladores: Oculus Touch, Xbox One Controller
- Sensores: Acelerómetro, Giroscopio, Magnetómetro, seguimiento posicional 360°
- Precio: 850€ (Octubre de 2016)



Fig.5. Oculus Rift.

II.3 Desarrollo del entorno 3D

Con el propósito de evaluar el impacto de la latencia en la QoE para aplicaciones VR, se ha utilizado un escenario de inmersión desarrollado en Unity 3D. El escenario, conocido como “Madrid Grid” en el proyecto METIS II, simula las variaciones dinámicas del tráfico (tanto en espacio como en tiempo) en un entorno urbano denso típico.

El escenario incluye la escena real y una capa de información lógica que se puede superponer a la escena, lo que permite representar distintos aspectos de comunicaciones sobre los que se pretenda incidir de alguna manera. Para la visualización lógica, el proyecto METIS-II desarrolló una interfaz de usuario simple y flexible, la cual presenta el contenido estructurado en forma de árbol (categorías y subcategorías) y otros elementos adicionales como control de la línea de tiempo, deslizador de velocidad de reproducción de la traza, paneles de datos laterales, etc.



Fig.6. Madrid Grid, plataforma de visualización e interfaz de usuario (2D).

Si bien para visualizar el escenario generado en Unity 3D con dispositivos VR como las Oculus Rift solo era necesario seguir las instrucciones descritas en apartados anteriores, la interfaz de usuario (UI, *User Interface*) no funciona de forma directa al pasarlo a VR. Para mantener las funcionalidades de la UI en VR ha sido necesario realizar algunas acciones, las cuales se describen en el siguiente apartado.

II.3.1 Visualización de las interfaces de usuario en modo VR

Utilizando las capacidades VR de Unity 3D y las propias funciones de Oculus, ha sido posible no sólo transportar el escenario Madrid Grid a VR, sino también incluir la información de control y la capa de información lógica en el escenario.

Además, se ha diseñado una versión espacial de la UI que posiciona la interfaz dentro del propio entorno utilizando el objeto *canvas* de tipo UI para ello. Este objeto se adjunta a la *UI camera* que a su vez es hija de la *Main Camera*. Con esto se consigue que cuando el usuario mueva la cabeza en las diferentes direcciones la UI permanezca en una posición fija.

Al no contar con el cursor del ratón que tenemos cuando interactuamos desde el ordenador, se hizo imprescindible diseñar una retícula que indicara al usuario la posición exacta del centro de la pantalla, sobre todo cuando se desea seleccionar algún elemento en la UI. La retícula de orientación es un excelente ejemplo de la adaptación de viejos paradigmas a la VR. La retícula resulta imprescindible para apuntar con precisión, sin embargo, con pegarla simplemente sobre la escena en un plano de profundidad fijo no se produce el comportamiento esperado.

Para que la retícula funcione correctamente, primero se crea un objeto de tipo *quad*, al cual se le van agregando componentes tanto para personalizar su apariencia como para gestionar su comportamiento. A través del script creado *Reticle* se configura el posicionamiento de la retícula, la escala y el comportamiento de la misma al colisionar con otros objetos. Para que siga la rotación de la *Main Camera*, se ha añadido como componente un nuevo script *Follow Rotation*, el cual simula que la retícula se encuentra siempre en el centro de nuestro display. La Fig. 7 muestra el aspecto final del escenario VR creado.

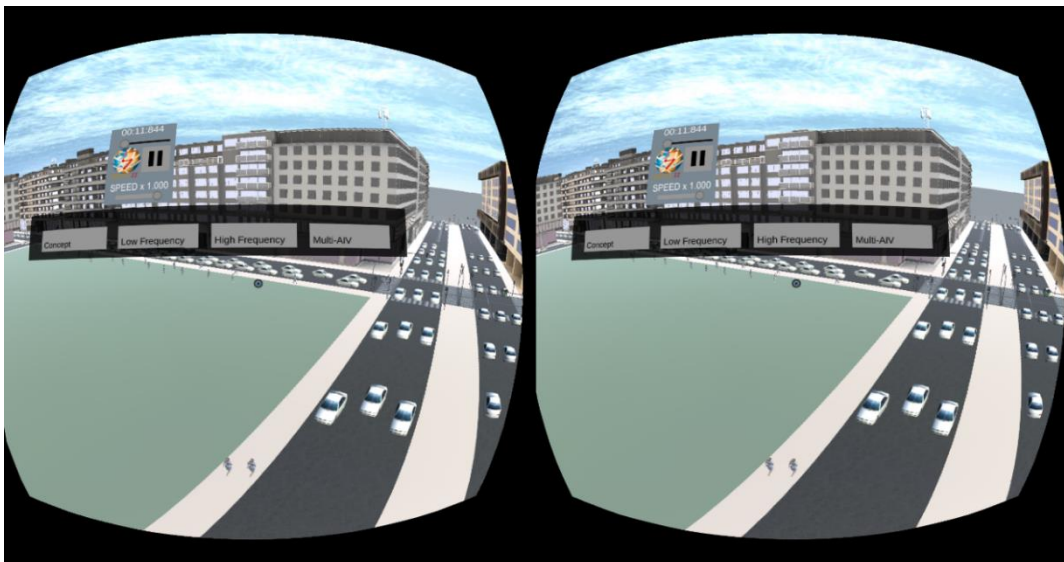


Fig.7. Escenario Madrid Grid creado en VR, menú de interacción y retícula.

II.3.2 Interacción con controles

Para la interacción con la interfaz de usuario han sido diseñadas reglas de interacción a través del mapeo de controles y creación de scripts en Unity. De los controles para Oculus Rift disponibles, ha sido seleccionado el *Oculus Remote Controller*, pues resulta adecuado para la interacción con la UI, donde solo es necesario mostrar y ocultar el menú y seleccionar botones.

A partir de proyecto ejemplo facilitado por la comunidad de Oculus [23] se han creado dos scripts que gestionan el comportamiento del menú y las acciones a desarrollar según el botón que sea seleccionado.

El primero de los scripts llamado *Basic Look Input Module* es agregado como componente al objeto de tipo UI, *Event System*. Este objeto se encarga de procesar y gestionar eventos en la escena de Unity. El código desarrollado en el script permite tomar el punto medio de la pantalla como la ubicación fija del cursor permitiendo que este se comporte como un *mouse* convencional. Asimismo, contiene otros métodos que llaman a funciones *updates* cada cierto tiempo para detectar cuando un botón ha sido seleccionado y en caso de estar seleccionado dicho botón se ilumina, lo cual facilita la interacción para el usuario.

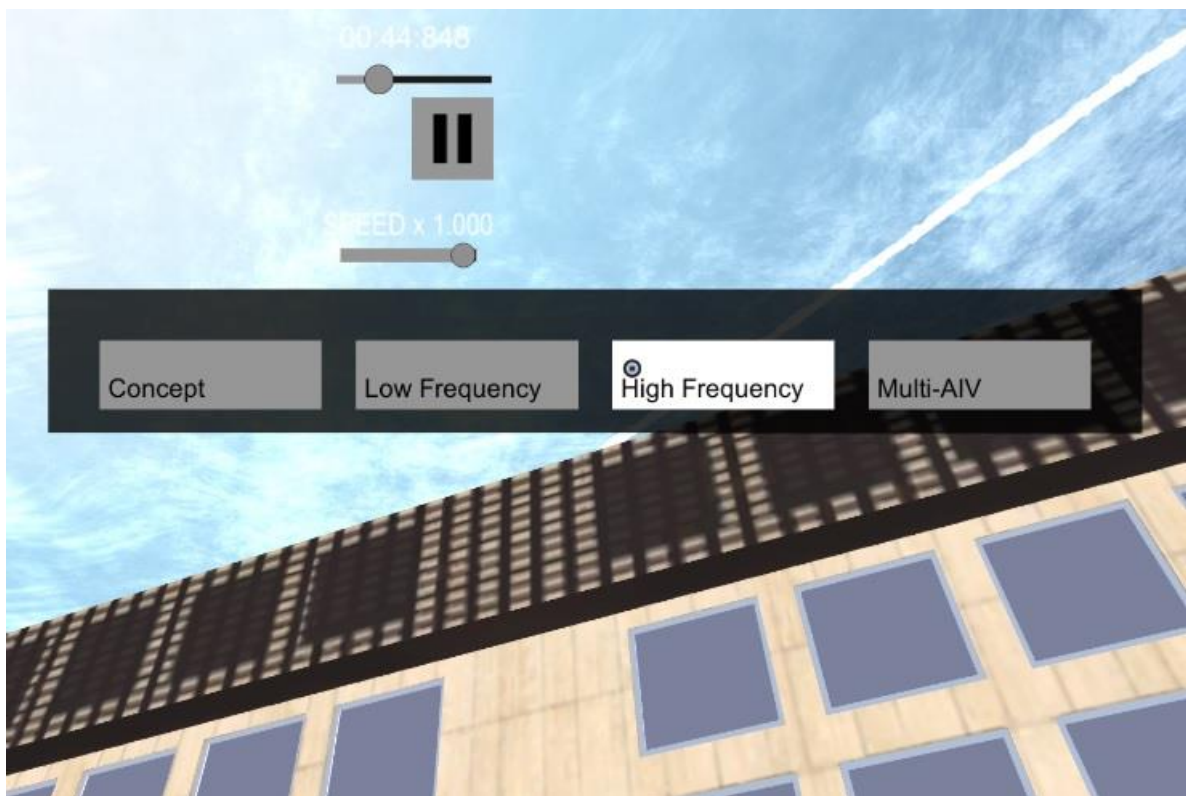


Fig.8. Botón seleccionado.

El segundo script llamado *Look Input Module*, puede ser asignado a cualquier objeto que permanezca durante todo el tiempo en la escena, en este caso ha sido seleccionada la *Main Camera*, pues cumple el requisito. En este script, a través del método *guiRaycastHit*, se comprueba si el

raycaster de la UI ha hecho contacto con algún elemento UI. En el código se describen dos modos de operación: *pointer* y *submit*. En el caso de *pointer*, se comporta como un *mouse*, ejecutando los eventos *pointerDown/pointerUp/pointerClick* con gran precisión, por lo que se utiliza para gestionar los *sliders*. El modo *submit* permite seleccionar y manipular con control de los ejes axiales solamente. Al seleccionar determinado botón, se llama a los métodos que gestionan el comportamiento del mismo y que dan paso a la acción deseada.

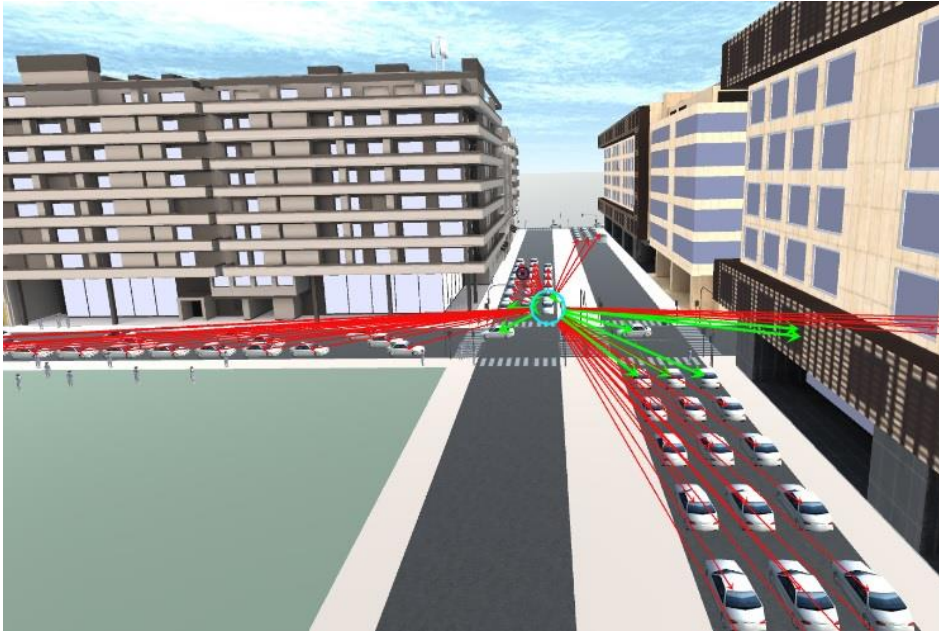


Fig.9. Acción correspondiente al botón seleccionado en Fig.8.

El mapeo de los botones del Remote Controller que permite al usuario interactuar con la UI aparece también en el primer script donde, *Button.DpadUp* y *Button.DpadDown* realizan las acciones de mostrar y ocultar el menú respectivamente, mientras que *Button.One* permite seleccionar cualquier botón del menú.

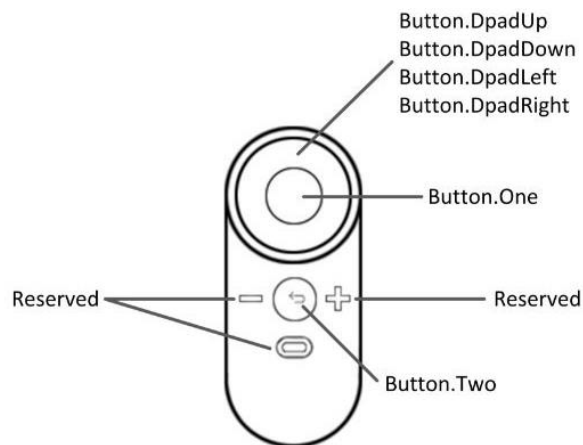


Fig.10. Mapa de botones para *Oculus Remote Controller* [24].

Llegados a este punto, contamos con un escenario VR con el cual se puede interactuar a través de las interfaces de usuario y los accesorios de control de Oculus. La siguiente tarea consiste en concretar el experimento y desarrollar las herramientas necesarias para poder fijar la latencia experimentada por el usuario, así como medir la QoE del usuario para la experiencia VR diseñada. La siguiente sección se centra en describir este trabajo.

II.4 Configuración para el experimento I (Oculus Debug Tool)

Para estimar la latencia y el efecto de esta en la experiencia del usuario se ha utilizado la herramienta Oculus Debug Tool. La herramienta de depuración de Oculus permite estudiar en tiempo real el rendimiento dentro de un juego o experiencia VR que se esté ejecutando. A esta herramienta se accede yendo al directorio Tools de Oculus SDK, donde se encuentra el archivo ejecutable. Una vez dentro de la aplicación, se selecciona Visible HUD, que incluye las opciones: *None*, *Performance*, *Stereo Debug*, o *Layer*. De estas opciones se selecciona *Performance*.

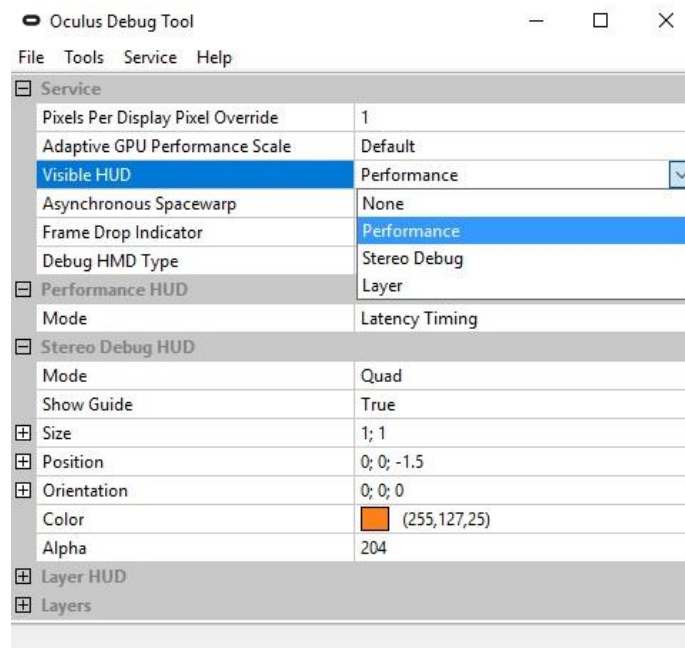
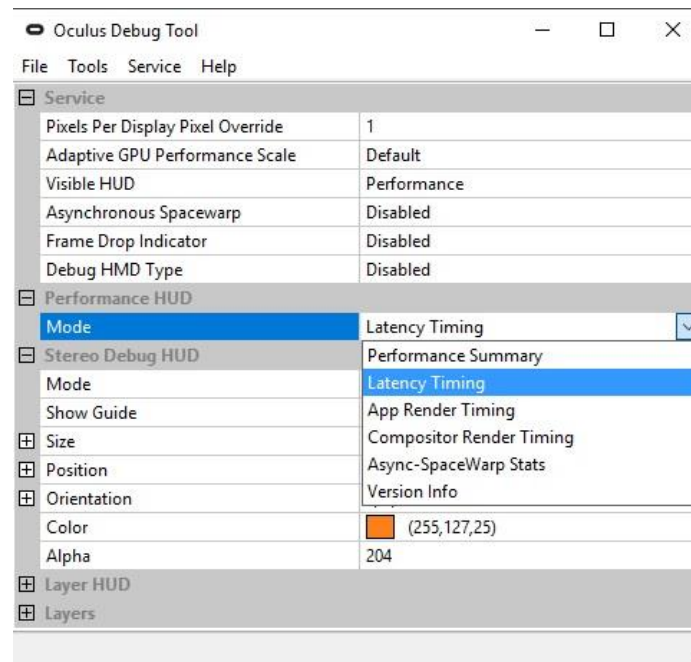
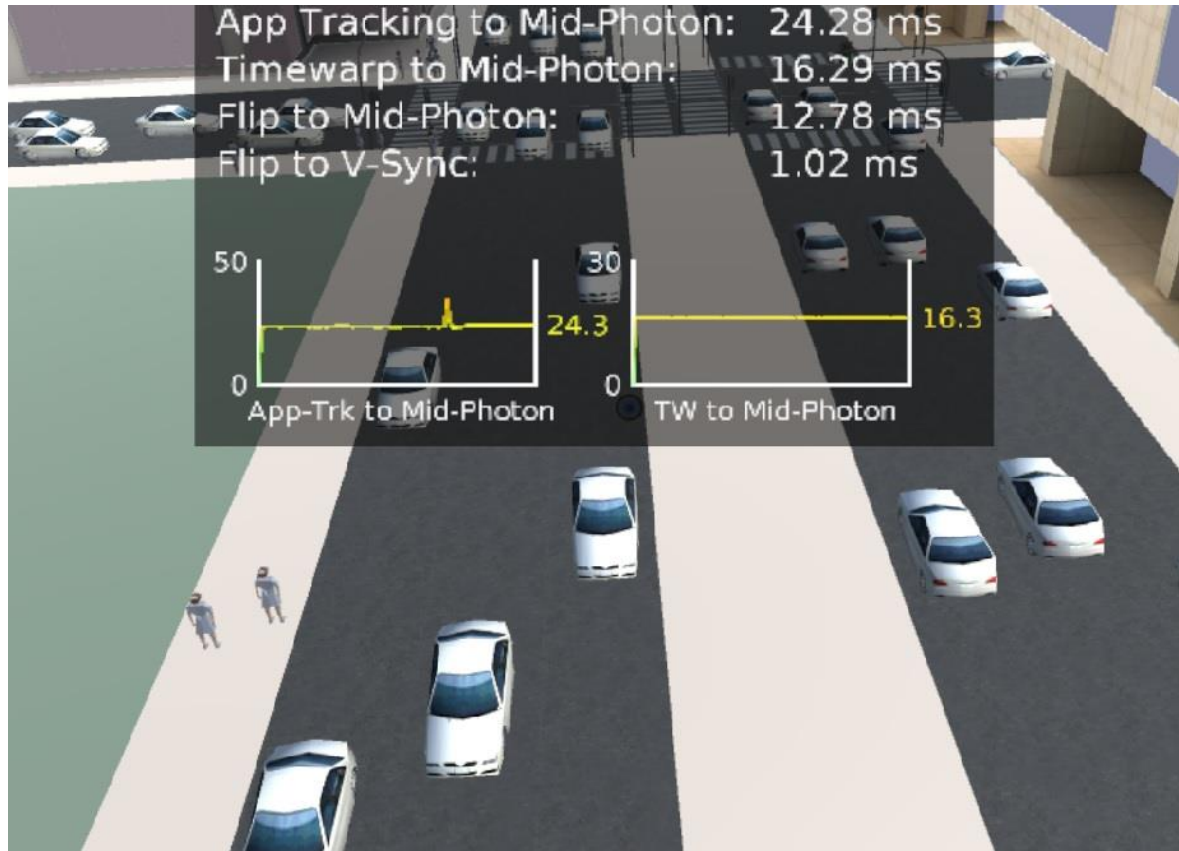


Fig.11. Selección Visible HUD-Performance.

A continuación, para *Performance*, existen las siguientes opciones disponibles: *Latency Timing*, *Render Timing*, *Performance Headroom*, y *Version Information*. Para conocer la latencia se selecciona *Latency Timing*.

Fig.12. Selección *Mode-Latency Timing*.

Con dicha configuración se puede observar a través de las Oculus Rift, además del escenario 3D en ejecución en Unity3D, una pantalla que muestra el rendimiento de los parámetros *App Tracking to Mid-Photon*, *Timewarp to Photon – Start*, y *Timewarp to Photon – Start* de forma gráfica.

Fig.13. *Display Latency Timing*.

La tabla a continuación describe cada uno de los parámetros.

Tabla 2. Latency Timing HUD

<i>Parámetro</i>	<i>Descripción</i>
<i>App Tracking to Mid-Photon</i>	Latencia desde que la aplicación <code>ovr_GetTrackingState ()</code> es llamada hasta que la <i>frame</i> objetivo es eventualmente mostrada con una iluminación media del punto medio del <i>display</i> .
<i>Timewarp to Mid-Photon</i>	Latencia desde que la última información de seguimiento predicha se alimenta a la unidad de procesamiento gráfico hasta que la línea media de exploración se ilumina en la pantalla HMD.
<i>Flip to Photon Start</i>	Diferencia de tiempo desde que se llena el búfer hasta que la primera línea de exploración se ilumina en la pantalla HMD.

Para este experimento nos centraremos en el parámetro *App Tracking to Mid – Photon*, pues este toma en cuenta los dos restantes. Este parámetro sufre variaciones al cambiar la densidad del pixelado (*Pixels per Display Pixel Override*), modificable desde el menú principal de Oculus Debug Tool.

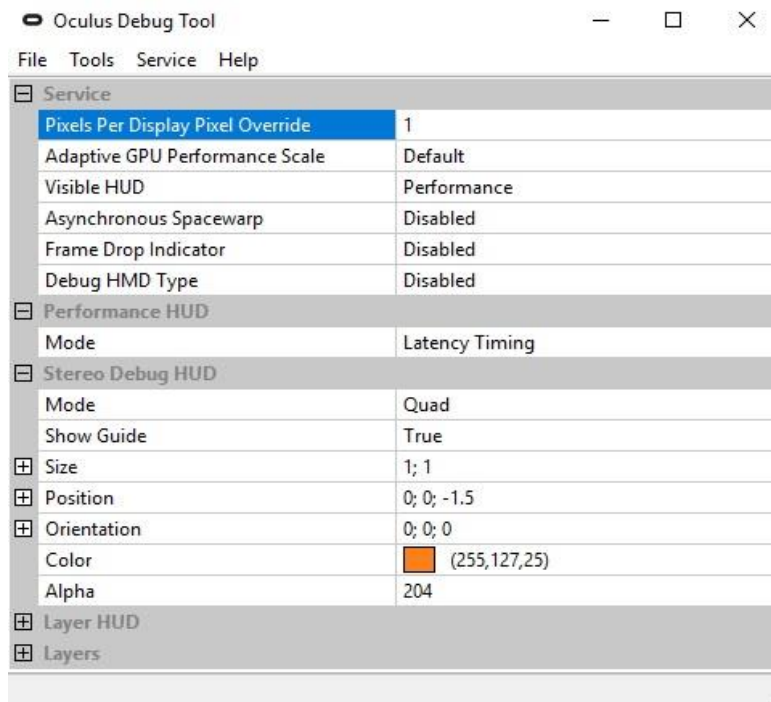


Fig.14. Selección Pixelado.

Incrementar la densidad del pixelado en cierto rango de valores (1-2), produce mejoras en la resolución de la imagen. Sin embargo, por encima de 2, no se observan cambios, pues la imagen tiene limitaciones de origen. No obstante, al incrementar dicho valor, se observa un incremento de la latencia produciéndose efectos desagradables para el usuario. Este efecto permite controlar a través del pixelado los valores de latencia experimentados.

Para este experimento los participantes son expuestos a una experiencia VR a través de las Oculus Rift. Los participantes repiten la experiencia para diferentes valores de latencia y evalúan la experiencia según el método *Degradation Category Rating (DCR)* [25]. El DCR evalúa la QoE empleando la métrica *Mean Opinion Score (MOS)*. Este método presenta los estímulos en pareja, donde el primero de los estímulos presentados se toma como referencia para evaluar el deterioro del segundo en una escala de 1 a 5 de la siguiente manera: 5 (Imperceptible), 4 (Perceptible pero no molesto), 3 (Un poco molesto), 2 (Molesto) y 1 (Muy molesto).

Durante el experimento el sujeto permanecerá de pie usando el HMD, mientras una tercera persona va cambiando de forma aleatoria los valores de pixelado (que aumentan o disminuyen la latencia). Cada vez que finalice la experiencia (15 segundos aproximadamente), y para un valor de latencia determinado, el individuo debe valorar la experiencia y notificar si le ha provocado sensación de mareo o no.

A continuación, se muestran los valores de pixelado seleccionados y las latencias asociadas a los mismos. Cabe destacar que la latencia mínima es de 22 ms, que se corresponde con el tiempo necesitado por el propio PC y la tarjeta gráfica para determinar el movimiento de la HMD y actualizar el vídeo mostrado.

Tabla 3. Latencias

<i>Valores de pixelado</i>	<i>Latencia asociada (ms)</i>
0 *	22
1	22
2	25
3	30
4	40
5	50
5.2	60

*Utilizado como referencia inicial.

III. Determinación del ancho de banda necesario en la VR móvil en tiempo real

La conocida como *360-degree VR* es una simulación audiovisual de la realidad. Durante este tipo de experiencia el usuario se encuentra rodeado por un mundo virtual y tiene la posibilidad de mirar en todas las direcciones. Existen diferentes tipos de *360-degree VR*, entre ellos los *live* vídeos capturados previamente, esto es, vídeos de la realidad que tras su procesamiento se pueden reproducir

a posteriori tantas veces como se desee, y los vídeos en tiempo real que pudieran utilizarse, por ejemplo, para difundir en tiempo real un partido de fútbol en VR.

VR vs 360-degree VR

360-degree VR no es VR. En una experiencia VR los usuarios tienen la posibilidad de moverse e interactuar con el mundo virtual que le rodea. Este mundo virtual necesita ser creado, lo cual implica contar con una capacidad de procesamiento muy importante. Por otra parte, *360-degree VR* se trata de un vídeo que ha sido grabado en todas las direcciones al mismo tiempo. En este último caso, los usuarios observan el mundo virtual que le rodea, pero no pueden interactuar con él. Entonces, se puede decir que, mientras en VR el usuario controla la experiencia, para *360-degree VR* se trata solo de una experiencia visual inmersiva. No obstante, dicha experiencia visual puede resultar muy agradable.

Procesamiento para *streaming* de vídeo 360°

Primeramente, a diferencia de un vídeo convencional, un vídeo 360° necesita múltiples cámaras para ser grabado. Cuando en el mundo real movemos la cabeza la percepción que tenemos del sonido cambia, por ello, otro aspecto a tener en cuenta durante la grabación es la utilización de micrófonos omnidireccionales y que el sonido sea adecuadamente procesado para garantizar que el sonido correcto llega al usuario en el momento preciso.

Después de la grabación, el vídeo debe ser “compuesto” en una etapa que se conoce como *Video Stitching*. Esto se debe a que el vídeo ha sido grabado con múltiples cámaras individuales. Para ello existen diferentes softwares (*VideoStitch Vahana VR*, *Kolor Autopano*) que permiten realizar la composición de los múltiples vídeos en un único vídeo esférico.

Posteriormente el vídeo debe ser codificado para su transmisión en *streaming*. La codificación se puede realizar vía software o a través de dispositivos codificadores. Esta etapa cobra vital importancia dado que este tipo de vídeos, por lo general, son de gran tamaño y la selección de una codificación óptima resulta sumamente necesaria, así como la posibilidad de codificar a diferentes tasas para ajustarse a la capacidad de la conexión del usuario.

Por último, el vídeo debe ser alojado en un servidor. El ancho de banda para ello dependerá de la resolución del vídeo, así como de la velocidad de fotogramas a la que ha sido grabado y el codificador empleado.

El *streaming* de vídeo es uno de los más grandes y populares consumidores de ancho de banda en Internet en la actualidad. El ancho de banda necesario para la transmisión se eleva más aún si se trata de vídeos 360°, lo cual constituye un inconveniente. Para resolver este problema, la calidad del vídeo se ajusta de acuerdo al estado de la conexión y características del dispositivo final. Las soluciones existentes utilizan búferes poco profundos, lo que hace que durante la reproducción del vídeo este se congele frecuentemente afectando destructivamente a la QoE.

El experimento II se plantea una estimación del ancho de banda necesario para transmitir un vídeo

360° en tiempo real en función de su tasa de codificación y su resolución. Asimismo, se evaluará la QoE percibida por un conjunto de sujetos para cada una de las resoluciones utilizadas.

III.1 Herramientas utilizadas y procedimiento experimental

Para el segundo experimento ha sido seleccionado un vídeo 360° con resolución máxima 4K de la galería de YouTube. Posteriormente, se procede a descargarlo con el programa 4K Video Downloader, software multiplataforma para la descarga de vídeos desde sitios web como Facebook, Vimeo y en este caso, YouTube. El vídeo será descargado para diferentes resoluciones.

Después de la descarga, se comprueba con la herramienta FFmpeg (<https://ffmpeg.org>) las características del vídeo para las diferentes resoluciones. El programa FFmpeg permite, transcodificar y hacer *streaming* de audio y de vídeo. Dicho programa es un software libre desarrollado en GNU/Linux, pero puede ser compilado en la mayoría de los sistemas operativos incluyendo Windows. Debe ser ejecutado desde la consola de comandos de Windows.

Una vez en la consola se ejecuta el comando *ffprobe* de la siguiente manera: *ffprobe video_in.mp4*. El comando *ffprobe* es un simple analizador de flujos multimedia. También se puede utilizar para generar todo tipo de información sobre una entrada, incluyendo duración, velocidad de fotogramas, tamaño de fotograma, etc.

Para la visualización de los vídeos se utilizará el Samsung Gear VR HMD, el cual requiere de un móvil Galaxy S8, S8+, S7, S7 edge, Note5, S6 edge+, S6 o S6 edge. En este caso se le conectará un teléfono Samsung S7, con 4GB de RAM y sistema operativo Android Nougat 7.0. Esto brinda una experiencia VR completamente móvil, con una resolución máxima de 2560x1440 píxeles, un campo de visión de 96 grados y una velocidad de fotogramas máxima de 60 *frames* por segundo (fps).



Fig.15. Terminal móvil Samsung S7 y gafas Samsung Gear VR.

La tasa de codificación del vídeo es un parámetro objetivo que repercute directamente en la QoE percibida por el usuario. Siendo así, procede elaborar un test donde cada uno de los usuarios encuestados sea capaz de evaluar la calidad de los vídeos en cuestión. Por otra parte, conocida la tasa

de codificación de cada uno de los vídeos, se estimará un valor del flujo de datos que generaría transmitir vídeos de la misma índole a través de la red móvil.

III.2 Configuración para el experimento II

A partir del vídeo original, se han generado versiones del mismo en las que solo varía la tasa de codificación y su resolución (Tabla 4). Los vídeos se han codificado con H.264 a 30 fps. La tasa de codificación se ha medido siguiendo el procedimiento descrito en el apartado anterior. Con la tasa de codificación asociada a cada resolución podemos calcular de forma aproximada el ancho de banda necesario para transmitir a través de la red en cada uno de los casos, asumiendo que 8 cámaras *rigs* han sido utilizadas en la grabación del vídeo. Para ello solo debe ser multiplicada la tasa de codificación por el número de cámaras asumido.

Tabla 4. Características de los vídeos

Vídeo	Resolución	Bitrate (kbps)	Acho de banda requerido (Mbps)
1	3840x2048	12672	101.7
2	2560x1440	8217	65.7
3	1920x1080	3458	27.7
4	1280x720	1572	12.6
5	854x480	767	6.2
6	640x360	459	3.7
7	426 x 240.	296	2.4
8	256x144	161	1.3

En el caso de los vídeos 360, estos se deben copiar en el directorio `sdcard/Oculus/360Videos` en el terminal móvil para que puedan ser visualizados en el mismo desde la aplicación Oculus Video previamente instalada y disponible en Google Play.

Para evaluar la QoE se utilizará el método *Absolute category rating* [25]. Este método se considera un método de estímulo simple y emplea una métrica MOS. El usuario evalúa la calidad de imagen del vídeo en cuestión sobre una escala de 1 al 5 de la siguiente manera: 5 (Excelente), 4 (Buena), 3 (Regular), 2 (Pobre) y 1 (Mala). Para la elección del método se ha considerado que la mayoría de las personas evalúan diariamente, de forma consciente o no, la calidad de imagen de vídeo en múltiples pantallas y, por lo tanto, cuentan con una experiencia previa que sirve de referencia.

En el Capítulo IV se llega a las conclusiones sobre el ancho de banda requerido para transmitir vídeos 360° que ofrezcan una QoE “aceptable”.

A los sujetos que participan en el experimento se les indica cómo ajustar la distancia focal y cómo iniciar o detener la reproducción de un vídeo a través del panel lateral de las Samsung Gear VR.

IV. Resultados y discusión

Para realizar los experimentos se ha tomado una muestra de 25 sujetos, valor promedio utilizado en estudios similares [26], [27]. A los sujetos se les ha aplicado una encuesta previa con el propósito de conocer edad y si han tenido o no experiencias previas con dispositivos VR, dado que estas podían ser consideradas como referencia.

Según la encuesta realizada, los sujetos encuestados se encuentran entre los 25 y 40 años de edad. Dentro de la muestra, 5 sujetos habían visionado vídeos 360° con las Samsung Gear VR y 5 habían tenido otras experiencias VR con otros HMDs, como las Oculus Rift.

En los apartados siguientes se detallan los resultados obtenidos en los dos experimentos, así como conclusiones sobre los mismos.

IV.1 Experimento I

A continuación se muestran las gráficas obtenidas a partir de los resultados experimentales. Cabe señalar que en la evaluación otorgada por los participantes a cada experiencia, el aumento del tiempo de exposición influye negativamente en la QoE. Los criterios de evaluación en cualquiera de los casos se encuentran sujetos al nivel de tolerancia de los usuarios.

La Fig. 16 muestra la QoE percibida por cada uno de los sujetos para las diferentes latencias. En el eje de las abscisas se encuentran los valores de latencia testeados mientras que la escala de colores va de verde a rojo, correspondiéndose el primero con la mejor experiencia y el segundo con la peor. Incluso para la mejor configuración posible, el valor de latencia obtenido es 22 ms, por eso se ha decidido tomar este como valor inicial.

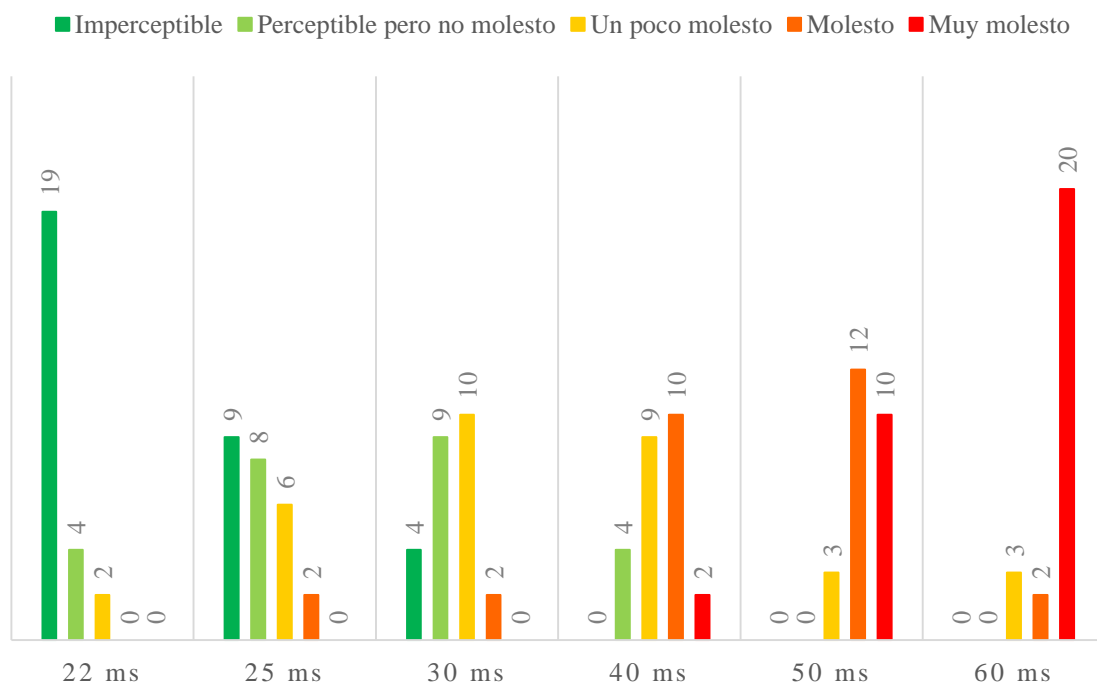


Fig.16. Resultados de la evaluación de QoE en el primer experimento.

- Para 22 ms de latencia el 76% de los encuestados considera que la latencia es imperceptible, lo cual hace pensar presumiblemente que con unos pocos milisegundos menos todos coincidirían con este criterio. Esto parece corroborar la conclusión de [15], donde se indicaba que por debajo de 20 ms no existe percepción de latencia.
- A partir de 25 ms el efecto desagradable provocado por la latencia no solo es perceptible, sino que empieza a ser molesto para algunas personas, concretamente, el 32% de los encuestados lo consideró “un poco molesto” o “molesto”.
- De 30 ms en adelante la latencia es claramente perceptible, según asegura el 84% de los encuestados.
- Para el resto de valores, los encuestados coinciden en que la latencia hace que la experiencia empiece a ser desagradable. Para 60 ms de latencia el 80% de los encuestados consideró la experiencia “muy molesta” y manifestó su deseo de abandonarla.

La Fig. 17 describe el comportamiento de la sensación de mareo con respecto a la latencia. En el eje de las ordenadas aparece el porcentaje de personas encuestadas, mientras que el eje de las abscisas aparecen los diferentes valores de latencia testeados.

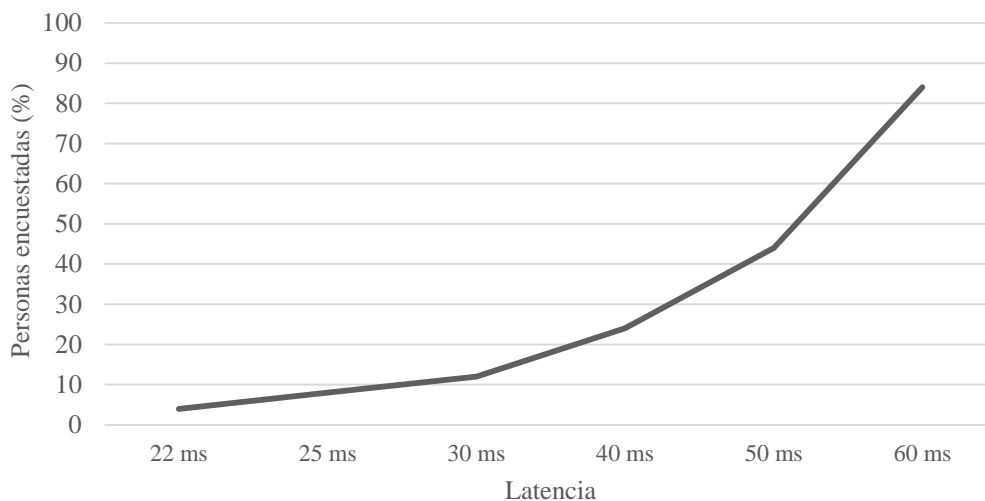


Fig.17. Sensación de mareo respecto al aumento de la latencia.

- Se observa un incremento aproximadamente exponencial del porcentaje de usuarios con sensación de mareo respecto al aumento de la latencia.
- Para 60 ms los encuestados notificaron que la latencia era intolerable y manifestaron sentirse agotados visualmente y mareados.
- Existe un porcentaje de la población, en nuestro experimento del 8%, que incluso sin una latencia que sea objetivamente molesta, sienten que la experiencia no es satisfactoria. Son claramente reacios al uso de la VR o experimentan cierta incomodidad al utilizar unas HMD, que eviten la visión cierta de la realidad.

IV.2 Experimento II

A continuación se muestran gráficamente los resultados obtenidos en este experimento.

La figura 18 muestra los resultados obtenidos para la QoE a partir de evaluar cada uno de los 8 vídeos. Los sujetos evaluaron los vídeos en posición de pie y con audio, excepto el segundo, el cual vieron sentados y con el volumen en el mínimo. Cabe destacar que, aunque el vídeo de mayor calidad tiene resolución 4K, la máxima resolución con la que puede ser visto es 2K (2560x1440 píxeles) pues esta es la resolución del teléfono Samsung S7 utilizado en el experimento.

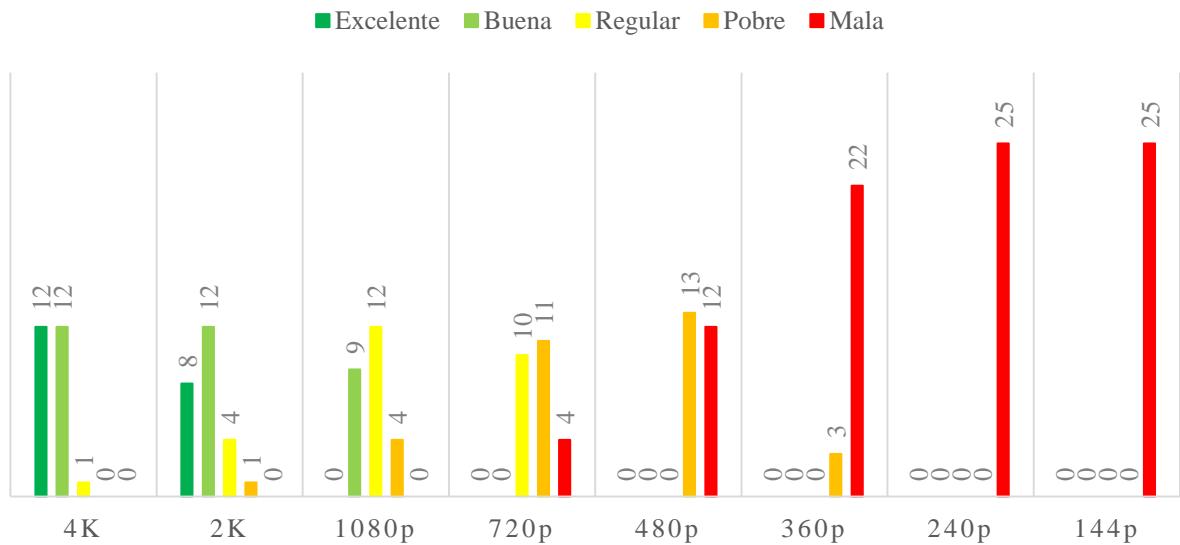


Fig.18. Resultados de la evaluación de QoE en el segundo experimento.

- Se hace evidente que la disminución de la resolución afecta negativamente la QoE.
- El 36% de los encuestados notificó haber detectado el pixelado de la pantalla del móvil.
- Incluso para la mejor configuración posible, el 52% de los sujetos no consideraron que la calidad del vídeo fuera excelente.
- Para 1080p de resolución, el 64% de los usuarios muestran su disconformidad con la calidad del vídeo visualizado evaluándola como “regular” o “pobre”.
- Si comparamos con las aplicaciones convencionales, las aplicaciones VR requieren mayor resolución para que una QoE similar sea percibida. Esto se debe a la óptica de aumento utilizada por las Samsung Gear VR.

La Fig. 19 muestra el comportamiento de la sensación de mareo, según la percepción de los usuarios encuestados.

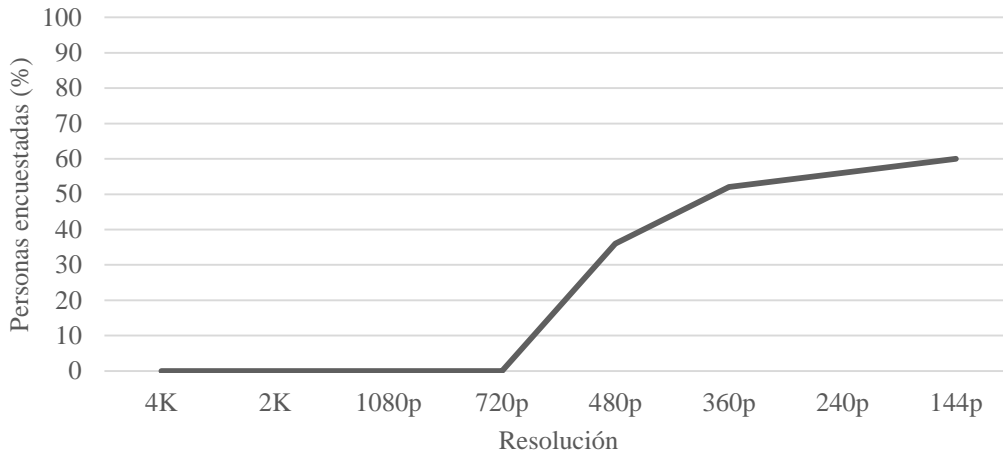


Fig.19. Sensación de mareo respecto a la resolución del vídeo.

- La disminución de la resolución provoca sensación de mareo en los usuarios, que realizan un esfuerzo visual extra para distinguir los objetos que aparecen en el vídeo.
- A partir de 720p la sensación de mareo empieza a aumentar de forma aproximadamente logarítmica.
- A partir de 480p más de la mitad de los encuestados (52%) manifiestan sentirse mareados, llegando a ser el 60% para 144p.
- De forma complementaria, los encuestados que manifestaron no sentir mareo dijeron que se debía a que no se habían esforzado en distinguir los objetos en el vídeo, pues a priori daban por hecho que sería en vano.

La Fig. 20 muestra el comportamiento de la sensación de inmersión con respecto a la resolución del vídeo. El eje de las ordenadas representa el porcentaje de personas encuestadas, mientras que en el eje de las abscisas aparecen las diferentes resoluciones.

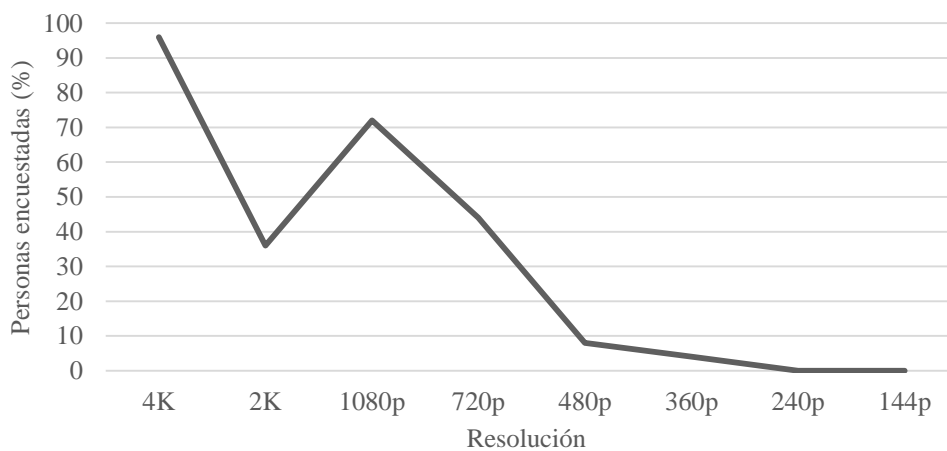


Fig.20. Experiencia inmersiva respecto a la resolución del vídeo.

- Los encuestados manifestaron que al disminuir la resolución la experiencia es cada vez menos inmersiva, pues se pierde profundidad visual y la experiencia se empieza a volver algo desagradable.
- El segundo vídeo, a pesar de ser percibido con la misma resolución que el primero, recibió menor calificación. Se supone que esto es debido a que la ausencia de audio y el hecho de verlo sentado deteriora la experiencia inmersiva, lo cual fue notificado por la mayoría de los usuarios.
- A partir de 480p de resolución, la calidad del vídeo resulta inaceptable y el 92% de los participantes coinciden con que se pierde completamente la experiencia inmersiva.

Para transmitir vídeos 360° con resolución 4K, sería necesario tener un ancho de banda mínimo de 11 Mbps aproximadamente. Sin embargo, esta resolución aún deja mucho que desear, por lo que la misma debe ser mejorada. Asimismo, mayores resoluciones se corresponderán con tasas de codificación más altas, lo cual demandará a las redes un aumento de la capacidad.

V. Conclusiones y líneas futuras

Tras haber realizado los experimentos podemos derivar importantes conclusiones sobre el rendimiento de las redes móviles actuales y de los dispositivos móviles que operan.

En el primer experimento se ha comprobado que, a priori, la experiencia VR tiene una latencia mínima de 22 ms derivada del propio procesamiento del vídeo a generar, incluso empleando una máquina de trabajo de altas prestaciones. Si se considera que hasta 25 ms es una latencia tolerable, eso dejaría como máximo 3 ms de latencia de la red para tener una experiencia análoga de forma inalámbrica. Las redes LTE actuales tienen una latencia de 50 ms, por lo que se hace evidente la necesidad de contar con redes con latencias más bajas. Sin embargo, no todo el peso debe caer encima del diseño de las redes móviles, también se debe considerar aumentar la capacidad de procesamiento de los dispositivos VR, colaborando así a la disminución de la latencia.

Por otra parte, en el segundo experimento se demostró que incluso para la mejor configuración posible, más de la mitad de los usuarios encuestados no se mostró conforme con la calidad percibida, por lo que se puede concluir que una meta necesaria para el soporte de aplicaciones VR sería diseñar dispositivos móviles con mayor resolución de pantalla. En este sentido, Samsung ha anunciado para 2018 dispositivos móviles con pantallas 11K [13]. Mayores resoluciones también traerán consigo requisitos más exigentes de capacidad de la red y generación de contenidos con la resolución correspondiente.

Durante los experimentos se preguntó a los encuestados cuál cómodo a lo que el 100% contestó: Samsung Gear VR. Esto llama la atención, dado que el HMD de Oculus Rift es mucho más caro. Sin embargo, Oculus Rift no se ajusta adecuadamente a la fisionomía del individuo. Los encuestados

notaron como inconveniente para la sensación de inmersión el de los dos HMD les resultaba más hecho de que se colara luz por la cavidad correspondiente a la nariz. En cuanto a las Samsung Gear VR, la principal crítica fue al sistema de audio, ya que no posee sistema de audio embebido sino que depende del terminal móvil que se utilice.

Como respuesta a la siguiente pregunta realizada en la introducción de la presente Tesina de Máster,

¿Es de verdad la red celular móvil un impedimento para la explosión del servicio de VR móvil o se deberá prestar más atención a los procesos de grabación/codificación y a las capacidades de los propios dispositivos móviles?

Tras la investigación desarrollada, podemos concluir que cuando se habla de aplicaciones VR y transmisión de las mismas en remoto, se debe prestar atención en ambas direcciones. El aumento de capacidad de la red móvil no tiene sentido si no existen aplicaciones que lo requieran y terminales que lo soporten, por otra parte, tampoco tiene sentido mejorar los terminales si la red no es capaz de transmitir los contenidos con una calidad y latencia aceptables.

Respecto a las líneas futuras, una vez acabada esta tesina de máster, y puesto que recientemente se ha adquirido un sistema completo de LTE, se plantea la opción de desplegar dicha red y experimentar una ejecución real de experiencia de VR móvil y en la nube. Esto servirá para determinar si los efectos del jitter agravan la experiencia y hacen que la latencia tenga que ser menor en comparación con el caso de jitter constante, que es el utilizado en este trabajo.

Agradecimientos

A mi querida familia, mis apreciados directores y a todos los que me han apoyado en esta aventura transoceánica.

Bibliografía

- [1] <https://www.wowza.com/solutions/streaming-types/virtual-reality-and-360-degree-streaming> Último acceso 2-09-2017.
- [2] LaValle, S. (2013). The latent power of prediction. *Oculus VR*.
- [3] 3GPP TR 36.913, "Requirements for further advancements for E-UTRA (LTE-Advanced)".
- [4] Monserrat, J. F., Martín-Sacristán, D. (2016). D1.1 Refined scenarios and requirements, consolidated use cases, and qualitative techno-economic feasibility assessment.
- [5] Anthes, C., García-Hernández, R. J., Wiedemann, M., & Kranzlmüller, D. (2016, March). "State of the art of virtual reality technology," *Aerospace Conference, 2016 IEEE* (pp. 1-19). IEEE.
- [6] Hoffman, D. M., Girshick, A. R., Akeley, K., and Banks, M.S., "Vergence-accommodation conflicts hinder visual performance and cause visual fatigue," *Journal of Vision*, vol. 8, no. 3, pp. 1–, March 2008.

- [7] Lambooij, M., IJsselsteijn, W., Fortuin, M., and Heynderickx, I., “Visual discomfort and visual fatigue of stereoscopic displays: A review,” *Journal of Imaging Science and Technology*, vol. 53, no. 3, pp. 1–, 2009.
- [8] Kim, J., Kane, D., & Banks, M. S. (2014). “The rate of change of vergence–accommodation conflict affects visual discomfort,” *Vision research*, 105, 159-165.
- [9] Huang, F.-C., Chen, K, and Wetzstein, G. “Light Field Stereoscope: Immersive Computer Graphics via Factored Near-Eye Light Field Displays with Focus Cues,” *ACM Trans Graph. (SIGGRAPH)*, vol. 34, no. 4, p. 60, 2015.
- [10] Abrash, M. (2012). Latency the sine qua non of AR and VR. *Ramblings in Valve Time, a blog by Michael Abrash*.
- [11] Hettinger, L. J., & Riccio, G. E. (1992). “Visually induced motion sickness in virtual environments,” *Presence: Teleoperators & Virtual Environments*, 1(3), 306-310.
- [12] Jung, S. M., Oh, S. H., Whangbo, T. K., “360° Stereo image based VR motion sickness testing system,” *2017 International Conference on Emerging Trends & Innovation in ICT (ICEI)*, Pune, India, 2017, pp. 150-153.
- [13] <https://www.extremetech.com/computing/209740-samsung-is-reportedly-working-on-an-11k-screen-claims-it-can-create-3d-illusions> Último acceso 1-09-2017
- [14] Gartner Source (2016).
- [15] Maamar, H. R., Boukerche, A., & Petriu, E. (2013). “Streaming 3d meshes over thin mobile devices,” *IEEE Wireless Communications*, 20(3), 136-142.
- [16] Raaen, K., & Kjellmo, I. (2015, September). “Measuring Latency in Virtual Reality Systems,” *International Conference on Entertainment Computing* (pp. 457-462). Springer, Cham.
- [17] Carmack, J. (2013). Latency mitigation strategies. *Twenty Milliseconds*.
- [18] Adelstein, B. D., Lee, T. G., & Ellis, S. R. (2003, October). “Head tracking latency in virtual environments: psychophysics and a model,” *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* (Vol. 47, No. 20, pp. 2083-2087). Sage CA: Los Angeles, CA: SAGE Publications.
- [19] Mania, K., Adelstein, B. D., Ellis, S. R., & Hill, M. I. (2004, August). “Perceptual sensitivity to head tracking latency in virtual environments with varying degrees of scene complexity,” *Proceedings of the 1st Symposium on Applied perception in graphics and visualization* (pp. 39-47). ACM.
- [20] Meehan, M., Razzaque, S., Whitton, M. C., & Brooks, F. P. (2003, March). “Effect of latency on presence in stressful virtual environments,” *virtual reality, 2003. Proceedings. IEEE* (pp. 141-148). IEEE.
- [21] Página web UNITY-3D <https://unity3d.com/es> Último acceso 26-08-2017.
- [22] Unity Manual <https://docs.unity3d.com/es/current/Manual/index.html> Último acceso 26-08-2017.
- [23] <https://forums.oculus.com/developer/discussion/16710/new-unity-ui-ovr-look-based-input-howto> Último acceso 26-08-2017.

[24] Manual Oculus <https://product-guides.oculus.com/en-us/documentation/rift/latest/concepts/book-rug/>
Último acceso 28-08-2017.

[25] Recommendation ITU-T P.913 Methods for the subjective assessment of video quality, audio quality and audiovisual quality of Internet video and distribution quality television in any environment, 03/2016

[26] Schatz, R., Sackl, A., Timmerer, C., & Gardlo, B. (2017, May). "Towards subjective quality of experience assessment for omnidirectional video streaming," *Quality of Multimedia Experience (QoMEX), 2017 Ninth International Conference on* (pp. 1-6). IEEE.

[27] Keighrey, C., Flynn, R., Murray, S., & Murray, N. (2017, May). "A QoE evaluation of immersive augmented and virtual reality speech & language assessment applications," *Quality of Multimedia Experience (QoMEX), 2017 Ninth International Conference on* (pp. 1-6). IEEE.