



Universidad de Valladolid



ESCUELA DE INGENIERÍAS  
INDUSTRIALES

UNIVERSIDAD DE VALLADOLID

ESCUELA DE INGENIERIAS INDUSTRIALES

Grado en Ingeniería en Diseño Industrial y Desarrollo del  
Producto

**ANÁLISIS DE MÉTODOS APLICADOS A LA  
REPRESENTACIÓN ESTEREOSCÓPICA EN  
DISPOSITIVOS DE REALIDAD VIRTUAL**

**Autor:**

**Álvarez Gayoso, Darío José**

**Tutor:**

**Alarcia Estévez, María Esperanza  
Matemática Aplicada**

**Valladolid, Diciembre 2017.**



En primer lugar, quiero agradecer a mi familia todo el apoyo que me ha dado para conseguir llegar hasta aquí.

Además, quiero darle las gracias a mi tutor de prácticas, Pablo Sanchez Gatón, por proponerme este interesante proyecto y facilitarme una parte muy importante del material empleado a lo largo del trabajo.



## RESUMEN

La tecnología cada día tiene más peso en la sociedad, estamos siendo testigos de numerosos avances tecnológicos y somos propietarios de parte de este progreso.

Tal ha sido la evolución, que hoy en día podemos disfrutar de una experiencia de realidad virtual en nuestras casas gracias a un dispositivo HDM (Head Mounted Display). A pesar de que el concepto no era nuevo, no ha sido hasta el año 2012 cuando la compañía Oculus VR lanzó un dispositivo con unas prestaciones hasta la fecha inigualables y un diseño compacto que lo hacían habitable en cualquier hogar.

Además, en 2014 Google crea las Cardboard, unas gafas de cartón con unas lentes de plástico, en las que, al introducir el móvil, se convierten en unas gafas de realidad virtual con unas prestaciones más que aceptables. En este momento la realidad virtual empezó a imponerse en nuestras vidas.

Para crear una aplicación de realidad virtual se puede partir de dos unidades básicas, bien un modelo 3D o bien imágenes estereoscópicas. En este trabajo se ha realizado un análisis detallado de la tecnología existente hasta el momento, y una comparativa entre los distintos recursos y métodos, generando una guía para elegir el recurso más adecuado para cada objetivo.

## PALABRAS CLAVE

- **Realidad virtual**
- **HDM**
- **Google Cardboard**
- **HTC Vive**
- **Estereoscopía**

## **ABSTRACT**

The growing importance that new technologies are acquiring in our society is an indisputable fact. Proof of this are the numerous technological advances of which we are not only witnesses, but also owners of part of this progress

Such has been the evolution, that nowadays we are able to enjoy a virtual reality experience in our homes thanks to an HDM device (Head Mounted Display). Despite this concept was not new, it was not until 2012 when the Company Oculus VR launched a device with unrivalled features to date and a compact design which made it habitable at any home.

In addition, in 2014 Google created the Cardboard: a glasses made of paper-board with plastic lenses, in which, when you enter the phone, the become virtual reality glasses with more than acceptable benefits. At this time, virtual reality was imposed on our lives.

To create a virtual reality application is possible to start from two basic units, either a 3D model or stereoscopic images. In this work a detailed analysis of the existing technology has been made up to now, and a comparison between the different means and methods, generating a guide to choose the most suitable resource for each objective.

## **KEYWORDS**

- **Virtual reality**
- **HDM**
- **Google Cardboard**
- **HTC Vive**
- **Stereoscopy**

## ÍNDICE

Índice .....	1
Índice de ilustraciones .....	5
Índice de tablas .....	11
1- Motivaciones .....	13
2- Objetivos del proyecto .....	13
3- Introducción a la realidad virtual .....	14
3.1- Introducción a la visión estereoscópica .....	14
3.2- Métodos auxiliares de observación estereoscópica .....	16
3.3- Historia de la realidad virtual .....	24
3.4- Dispositivos de realidad virtual .....	32
3.4.1- HMD para smartphone .....	32
3.4.2- HMD integrales .....	34
4- Fundamentos de la visión humana .....	36
4.1- Anatomía y fisiología del ojo humano .....	36
4.2- Defectos de la visión .....	37
4.3- Ergonomía de los HMD .....	39
4.3.1- Ergonomía de los HMD para usuarios con problemas de visión ...	40
5- Software .....	42
5.1- Unity .....	42
5.1.1- Software complementario a Unity .....	46
5.1.2- Realidad virtual en Unity .....	47
5.1.2.1- Visualización estereoscópica a partir de imágenes .....	47
5.1.2.2- Visualización estereoscópica a partir de modelo 3D .....	50

5.2- Software de apoyo .....	51
5.2.1- Panoweaver .....	51
5.2.2- Google Street View .....	54
5.2.3- Photoshop .....	54
5.2.4- Premiere .....	57
5.2.5- 3ds Max .....	58
6- Hardware .....	59
6.1- Análisis de hardware en el mercado .....	59
6.1.1- Smartphone .....	59
6.1.2- Cámara réflex .....	60
6.1.3- Cámaras 360° .....	64
6.1.4- Cámaras estereoscópicas .....	67
6.1.5- Cámaras estereoscópicas 360° .....	68
6.1.6- Trípode con rótula medidora de ángulos .....	71
6.1.7- Rótula panorámica .....	72
6.2- Desarrollo de un soporte estereoscópico para smartphone .....	77
6.2.1- Planos del soporte .....	80
6.2.2- Planos de soporte con diseño adaptativo .....	85
7- Creación de aplicaciones de realidad virtual en Unity .....	87
7.1- Creación de entornos estáticos en realidad virtual .....	88
7.1.1- Métodos basados en imágenes .....	88
7.1.1.1- Fotografía esférica .....	88
7.1.1.2- Fotografía semiesférica .....	92
7.1.1.3- Fotografía cilíndrica .....	93
7.1.1.4- Fotografía plana .....	95
7.1.2- Métodos basados en modelos 3D .....	96

7.2- Análisis de métodos de recreación de entornos estáticos .....	97
7.2.1- Métodos basados en fotografía esférica .....	98
7.2.1.1- Cámara réflex .....	98
7.2.1.2- Smartphone .....	103
7.2.1.3- Cámara 360° .....	109
7.2.1.4- Vuze camera .....	113
7.2.1.5- Renders .....	118
7.2.2- Métodos basados en fotografía semiesférica .....	123
7.2.2.1- Cámara réflex .....	123
7.2.2.2- Smartphone .....	127
7.2.2.3- Cámara 180° .....	130
7.2.2.4- Renders .....	133
7.2.3- Métodos basados en fotografía cilíndrica .....	135
7.2.3.1- Cámara réflex .....	135
7.2.3.2- Smartphone .....	138
7.2.3.3- Renders .....	141
7.2.4- Métodos basados en fotografía plana .....	144
7.2.4.1- Cámara réflex .....	144
7.2.4.2- Smartphone .....	146
7.2.4.3- Renders .....	148
7.2.5- Métodos basados en modelos 3D .....	150
7.3- Creación de entornos dinámicos en realidad virtual .....	151
7.3.1- Métodos basados en imágenes .....	151
7.3.1.1- Vídeo esférico .....	152
7.3.1.2- Vídeo semiesférico .....	154
7.3.1.3- Vídeo cilíndrico .....	156
7.3.1.4- Vídeo plano .....	157
7.3.2- Métodos basados en modelos 3D .....	159
7.4- Análisis de métodos de recreación de entornos dinámicos .....	160
7.4.1- Métodos basados en vídeo esférico .....	161
7.4.1.1- Vuze camera .....	161
7.4.1.2- Cámara 360° .....	166
7.4.1.3- Renders .....	169

---

7.4.2- Métodos basados en vídeo semiesférico .....	172
7.4.2.1- Cámara 180° .....	173
7.4.2.2- Renders .....	177
7.4.3- Métodos basados en vídeo cilíndrico .....	180
7.4.3.1- Renders .....	180
7.4.4- Métodos basados en vídeo plano .....	183
7.4.4.1- Smartphone .....	183
7.4.4.2- Renders .....	188
7.4.5- Métodos basados en modelos 3D .....	190
8- Conclusiones .....	191
8.1- Conclusiones sobre aplicaciones de entornos estáticos .....	191
8.1.1- Aplicaciones esféricas .....	192
8.1.2- Aplicaciones semiesféricas .....	193
8.1.3- Aplicaciones cilíndricas .....	194
8.1.4- Aplicaciones planas .....	194
8.1.5- Comparativa general de aplicaciones .....	196
8.2- Conclusiones sobre aplicaciones de entornos estáticos .....	197
8.2.1- Aplicaciones esféricas .....	198
8.2.2- Aplicaciones semiesféricas .....	199
8.2.3- Aplicaciones cilíndricas .....	199
8.2.4- Aplicaciones planas .....	200
8.2.5- Comparativa general de aplicaciones .....	201
9- Bibliografía .....	202
9.1- Bibliografía referenciada .....	203
9.2- Bibliografía gráfica .....	203
9.3- Vídeos extraídos para crear aplicaciones .....	203

## ÍNDICE DE ILUSTRACIONES

Figura 1- Esquema visión estereoscópica .....	14
Figura 2- Efectos tridimensionales no estereoscópicos .....	15
Figura 3- Esquema de estereoscopio de visión cruzada .....	16
Figura 4- Esquema del estereoscopio de visión paralela .....	17
Figura 5- Estereoscopio antiguo .....	17
Figura 6- Esquema del funcionamiento de un anaglifo .....	18
Figura 7- Esquema de la polarización de la luz .....	19
Figura 8- Esquema del visionado 3D mediante polarización lineal .....	20
Figura 9- Esquema del visionada 3D mediante polarización circular .....	20
Figura 10- Esquema del visionado 3D mediante obturación .....	21
Figura 11- Esquema del visionado 3D mediante monitor lenticular .....	22
Figura 12- Esquema del visionado 3D mediante estereograma .....	23
Figura 13- Usuaría disfrutando de las gafas de realidad virtual HTC Vive ..	23
Figura 14- The stereoscope (Charles Wheatstone) .....	24
Figura 15- The lenticular stereoscope (David Brewster) .....	25
Figura 16- Película estereoscópica anaglífica .....	25
Figura 17- View Master (William Gruber) .....	26
Figura 18- Sensorama (Morton Heiling) .....	26
Figura 19- Telesphere Mask (Morton Heiling) .....	27
Figura 20- Sword of Damocles (Ivan Sutherland y Bob Sproull) .....	28
Figura 21- Jaron Lanier en su laboratorio de investigación .....	28
Figura 22- EyePhone 1 y Dataglobes (VPL) .....	29
Figura 23- Virtuality para 4 jugadores .....	29
Figura 24- Virtual Boy (Nintendo) .....	30
Figura 25- Oculus Rift (Oculus VR) .....	30
Figura 26- Cronograma de la historia de la realidad virtual .....	31
Figura 27- Gafas Cardboard de Google .....	32
Figura 28- Samsung Gear VR .....	33
Figura 29- Gafas Google Daydream .....	33
Figura 30- HTC VIVE .....	34
Figura 31- Esquema del funcionamiento de los sensores de las HTC VIVE ..	34

Figura 32- Usuario utilizando las Oculus Rift .....	35
Figura 33- Usuario utilizando las gafas PlayStation VR .....	35
Figura 34- Esquema de las partes principales del ojo humano .....	36
Figura 35- Esquema visión ojo sano .....	37
Figura 36- Esquema visión ojo hipermetrope y lente convexa .....	38
Figura 37- Esquema visión ojo miope y lente cóncava .....	38
Figura 38- Esquema visión ojo astigmático y lente a medida .....	38
Figura 39- Ángulo de visión humana y gafas VR .....	39
Figura 40- Esquema acomodación HDM .....	40
Figura 41- Esquema del ajuste para problemas de visión en un HDM .....	40
Figura 42- Esquema de la Ley de Snell .....	41
Figura 43- Esquema de la adaptación de un HDM a usuarios con problemas de visión .....	41
Figura 44- Esfera en la Scene de un proyecto de Unity .....	43
Figura 45- Esfera con un Mesh Filter de 32.768 polígonos .....	43
Figura 46- Camera de Unity, con área de enfoque y previsualización de enfoque .....	44
Figura 47- Script base de Unity en C# .....	45
Figura 48- Imagen esférica .....	47
Figura 49- Imagen esférica proyectada sobre una esfera en Unity .....	48
Figura 50- Imagen semiesférica proyectada sobre una semiesfera en Unity .....	48
Figura 51- Imagen cilíndrica proyectada sobre un cilindro en Unity .....	49
Figura 52- Imagen plana proyectada sobre un plano en Unity .....	49
Figura 53- Modelo 3D de una bodega en Unity .....	50
Figura 54- Fotos necesarias para completar el panorama esférico de la Catedral de Zamora .....	51
Figura 55- Resultado de la panorámica esférica de la Catedral de Zamora .....	52
Figura 56- Resultado del cosido con datos guardados del mirador del Troncoso de Zamora .....	52
Figura 57- Resultado del cosido del mirador del Troncoso de Zamora sin las imágenes de cielo .....	53
Figura 58- Panorama esférico del mirador del Troncoso de Zamora obtenido con la aplicación Google Street View .....	54

---

Figura 59- Vista que nos facilita eliminar el trípode de la imagen .....	55
Figura 60- Panorama esférico sin trípode .....	55
Figura 61- Par estereoscópico con diferencias en el cielo .....	56
Figura 62- Panorámica con el cielo modificado .....	57
Figura 63- Objetos exportados a Unity .....	58
Figura 64- Calculadora de distancia hiperfocal en <a href="http://www.dofmaster.com/dofjs.html">http://www.dofmaster.com/dofjs.html</a> .....	61
Figura 65- Esquema del campo de visión de una cámara .....	62
Figura 66- Calculadora online del campo de visión en: <a href="http://www.nodalninja.com/products/field-of-view-and-panorama-production-calculator/14">http://www.nodalninja.com/products/field-of-view-and-panorama-production-calculator/14</a> .....	63
Figura 67- Zonas de visión estereoscópica utilizando dos cámaras 360° .....	64
Figura 68- Cámara Panono, integrada por 36 cámaras .....	65
Figura 69- Zonas de visión estereoscópica utilizando dos cámaras 360° .....	65
Figura 70- Cámara Ricoh Theta .....	66
Figura 71- Esquema de visión de una pareja de cámaras modulares .....	66
Figura 72- Cámaras Kodak PixPro SP360, en una combinación para capturas estereoscópicas .....	67
Figura 73- Cámara SONY PMW-TD300 del año 2011 .....	68
Figura 74- Esquema funcionamiento cámara 3D 360° .....	68
Figura 75- Esquema zonas de visión estereoscópica en una cámara 3D 360° .....	69
Figura 76- Vuze Virtual Reality Camera .....	69
Figura 77- Cámara Project Beyond de Samsung .....	70
Figura 78- Configuración de la Facebook Surround 360 .....	70
Figura 79- Esquema del funcionamiento de un trípode con rótula medidora de ángulos .....	71
Figura 80- Esquema de los movimientos permitidos por una rótula panorámica .....	72
Figura 81- Esquema del ajuste de la cámara en el eje vertical .....	73
Figura 82- Esquema del ajuste en el eje transversal .....	73
Figura 83- Esquema del ajuste en el eje longitudinal .....	74
Figura 84- Esquema del giro en el eje vertical .....	75
Figura 85- Esquema del giro en el eje transversal .....	75
Figura 86- Esquema comparativo entre métodos .....	76

Figura 87- División de la esfera en la app Google Street View .....	77
Figura 88- Soporte estereoscópico para smartphone montado sobre un trípode .....	78
Figura 89- Esquema de rotación entre soporte y apoyo .....	79
Figura 90- Soporte utilizando 2 smartphones simultáneamente .....	79
Figura 91- Medidas necesarias del soporte para smartphone .....	85
Figura 92- Polo de una esfera compuesta por 32768 polígonos, en contraposición con otra integrada por 2600 .....	89
Figura 93- Shader utilizado para invertir las normales .....	90
Figura 94- El siguiente script contrarresta el efecto del tracking .....	91
Figura 95- Elementos que integran la escena semiesférica .....	93
Figura 96- Elementos que integran la escena cilíndrica .....	94
Figura 97- Elementos que integran la escena plana .....	96
Figura 98- Esquema de la calidad estereoscópica del panorama esférico utilizando una cámara réflex .....	100
Figura 99- Par estereoscópico del interior de la Catedral de Zamora .....	101
Figura 100- Par estereoscópico del Mirador del Troncoso (Zamora) .....	102
Figura 101- Esquema de la calidad estereoscópica del panorama esférico utilizando un smartphone .....	104
Figura 102- Par estereoscópico esférico del interior de la Catedral de Zamora .....	105
Figura 103- Par estereoscópico esférico del Mirador del Troncoso (Zamora) .....	106
Figura 104- Par estereoscópico esférico de la Plaza de la Marina Española (Zamora) .....	107
Figura 105- Par estereoscópico esférico del Parque del Oeste (Madrid) .....	108
Figura 106- Esquema de la calidad estereoscópica del panorama esférico utilizando una cámara 360° .....	110
Figura 107- Par estereoscópico esférico de Rueda (Valladolid) .....	111
Figura 108- Esquema de la calidad estereoscópica del panorama esférico utilizando la Vuze Camera .....	114
Figura 109- Par estereoscópico del Times Square (New York City) .....	115
Figura 110- Par estereoscópico de una plaza en el distrito financiero de Zuidas (Amsterdam) .....	116
Figura 111- Par estereoscópico de un lago (Amir Weiss) .....	117
Figura 112- Modelo 3D de la bodega en 3ds Max .....	118

Figura 113- Esquema multimaterial 3ds Max .....	119
Figura 114- Esquema self-illumin material .....	119
Figura 115- Esquema de la calidad estereoscópica del panorama esférico utilizando renders .....	120
Figura 116- Par estereoscópico esférico del modelo 3D de una bodega .....	122
Figura 117- Esquema de la calidad estereoscópica del panorama semiesférico utilizando una cámara réflex .....	124
Figura 118- Par estereoscópico semiesférico de la Catedral de Zamora .....	125
Figura 119- Par estereoscópico semiesférico del Mirador del Troncoso (Zamora) .....	126
Figura 120- Esquema de la calidad estereoscópica del panorama semiesférico utilizando un smartphone .....	128
Figura 121- Par estereoscópico semiesférico del Parque del Oeste (Madrid) .....	129
Figura 122- Esquema de la calidad estereoscópica del panorama semiesférico utilizando una cámara 180° .....	131
Figura 123- Par estereoscópico semiesférico de un bosque .....	132
Figura 124- Esquema de la calidad estereoscópica del panorama semiesférico utilizando renders .....	133
Figura 125- Par estereoscópico semiesférico del modelo 3D de una bodega .....	134
Figura 126- Esquema de la calidad estereoscópica del panorama cilíndrico utilizando una cámara réflex .....	136
Figura 127- Par estereoscópico cilíndrico de la Catedral de Zamora .....	137
Figura 128- Par estereoscópico cilíndrico del Mirador del Troncoso (Zamora) .....	137
Figura 129- Esquema de la calidad estereoscópica del panorama cilíndrico utilizando un smartphone .....	139
Figura 130- Par estereoscópico cilíndrico del Parque del Oeste (Madrid) .....	140
Figura 131- Esquema de la calidad estereoscópica del panorama cilíndrico utilizando renders .....	142
Figura 132- Par estereoscópico cilíndrico del modelo 3D de una bodega .....	143
Figura 133- Par estereoscópico plano de la estatua de Viriato (Zamora) .....	145
Figura 134- Par estereoscópico plano de la estatua de Viriato (Zamora) .....	147
Figura 135- Par estereoscópico plano del modelo 3D de una bodega .....	149
Figura 136- Elementos que integran la escena dinámica esférica .....	154
Figura 137- Elementos que integran la escena dinámica semiesférica .....	155

Figura 138- Elementos que integran la escena dinámica cilíndrica .....	157
Figura 139- Elementos que integran la escena dinámica plana .....	158
Figura 140- Aspecto del objeto 3D en la escena de Unity .....	159
Figura 141- Esquema de la calidad estereoscópica del vídeo esférico utilizando la Vuze Camera .....	162
Figura 142- Fotograma de la escena con la cámara estática .....	163
Figura 143- Fotograma de la escena en la que la cámara sigue una trayectoria lineal .....	164
Figura 144- Fotograma de la escena en la que la cámara sigue una trayectoria libre .....	165
Figura 145- Esquema de la calidad estereoscópica del vídeo esférico utilizando una cámara 360° .....	167
Figura 146- Fotograma del vídeo esférico utilizando una cámara 360° estática .....	168
Figura 147- Esquema de la calidad estereoscópica del vídeo esférico utilizando renders .....	170
Figura 148- Fotograma esférico del recorrido en la bodega .....	171
Figura 149- Shader que permite aplicar tiling y offset a nuestro MovieClip ....	172
Figura 150- Opciones añadidas por el Shader a nuestro MovieClip .....	173
Figura 151- Esquema de la calidad estereoscópica del vídeo semiesférico utilizando una cámara 180° .....	174
Figura 152- Fotograma de la escena en la que la cámara permanece estática .....	175
Figura 153- Fotograma de la escena en la que la cámara sigue una trayectoria lineal .....	176
Figura 154- Esquema de la calidad estereoscópica del vídeo semiesférico utilizando renders .....	178
Figura 155- Fotograma semiesférico del recorrido en la bodega .....	179
Figura 156- Esquema de la calidad estereoscópica del vídeo cilíndrico utilizando renders .....	181
Figura 157- Fotograma cilíndrico del recorrido en la bodega .....	182
Figura 158- Fotograma de la escena en la que la cámara permanece estática .....	185
Figura 159- Fotograma de la escena en la que la cámara gira .....	186
Figura 160- Fotograma de la escena en la que la cámara sigue una trayectoria libre .....	187
Figura 161- Fotograma plano del recorrido en la bodega .....	189

---

## ÍNDICE DE TABLAS

Tabla 1- Comparativa de aplicaciones de entornos estáticos .....	196
Tabla 2- Comparativa de aplicaciones de entornos dinámicos .....	201



## 1- MOTIVACIONES

La evolución tecnológica es abrumadora en nuestros días, uno de los últimos avances llega con las gafas de realidad virtual. A pesar de que el concepto VR (virtual reality) nació hace más de 50 años, no ha sido hasta el año 2012 cuando se han creado las primeras gafas de realidad virtual lo suficientemente ergonómicas y manejables como para que supusiese un boom a nivel de usuario. Estos acontecimientos están suponiendo una revolución en el ocio tecnológico, las compañías de videojuegos están ahora centradas en sacar al mercado unas gafas de realidad virtual a un precio razonable, sin perder en jugabilidad.

A partir de aquí, estamos viendo como numerosos campos de la ciencia se están nutriendo de estas posibilidades para crear nuevas formas de visualizar prototipos, diseñar entrenamientos mediante simulador más inmersivos u ofrecer terapias más eficaces.

Personalmente siempre me ha fascinado como los avances tecnológicos expanden nuestros horizontes, pudiendo afrontar desafíos de una forma nunca imaginada hasta la fecha. Numerosas ramas del conocimiento van a experimentar un desarrollo que ampliará y mejorará sus herramientas

Me parece que las gafas de realidad virtual, tienen mucho que ofrecernos en todos los campos de la ciencia. Por ello, investigar sobre las diversas fórmulas que permiten representar entornos de forma tridimensional, es un área de gran interés personal.

## 2- OBJETIVO DEL PROYECTO

En el siguiente estudio, se analizan tanto el software como el hardware involucrado en los procesos de creación de aplicaciones de realidad virtual. También se explorarán los diferentes métodos de recreación de entornos mediante realidad virtual, analizándose las máximas técnicas posibles para la obtención de imágenes estereoscópicas y dirigir los resultados a las plataformas principales de VR.

Además, se aprenderá a utilizar estas técnicas con el objetivo de generar el material necesario para construir aplicaciones estereoscópicas. Llegando a diseñarse un soporte estereoscópico para smartphone que permite utilizar nuestro dispositivo móvil como una herramienta para la generación de panoramas estereoscópicos.

Una vez realizadas las aplicaciones estereoscópicas, se analizarán cualitativamente en función del método por el cual han sido generadas. A partir de este análisis se generará una guía que permita elegir el método más conveniente en función de los objetivos del proyecto.

### 3- INTRODUCCIÓN A LA REALIDAD VIRTUAL

#### 3.1- Introducción a la visión estereoscópica

La visión estereoscópica es una facultad físico - psicológica que posee el ser humano, que le permite ver en tercera dimensión aquellos objetos que contempla mediante su visión binocular. [1]

Consiste en la fusión que realiza el cerebro de las imágenes percibidas por los ojos izquierdo y derecho. Se percibe una sensación de profundidad por la disparidad de las imágenes, al estar los ojos separados unos 65mm, las imágenes que percibe cada uno son ligeramente diferentes.

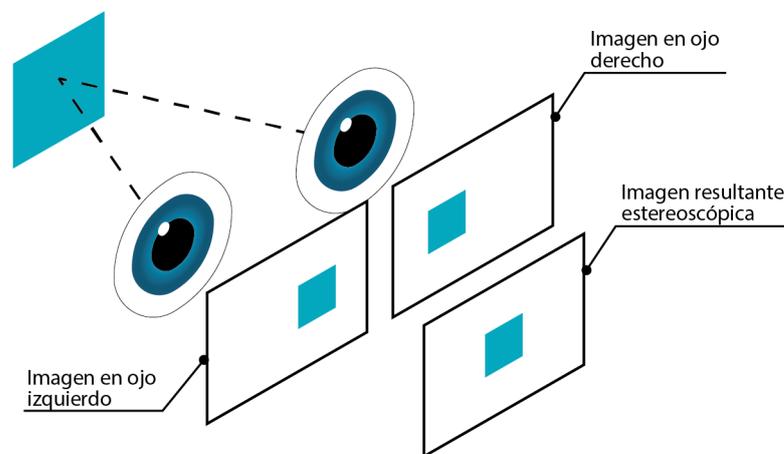


Figura 1. Esquema visión estereoscópica

De esta forma y de manera inconsciente, el cerebro mediante triangulación, se hace una idea de la distancia a la que se encuentran los objetos que enfocan los ojos, dentro de su campo de visión. Pero no toda la percepción de profundidad está relacionada con la visión estereoscópica.

Medios como la televisión, la pintura o la publicidad utilizan imágenes planas que proporcionan sensación de profundidad. Esto se explica porque el cerebro también obtiene la información tridimensional a través de otras técnicas, que ya no se basan en la visión binocular y que han sido utilizadas en pintura durante muchos años.

Estas propiedades están ligadas a la fisonomía de nuestra visión y a la forma en que atmósfera interacciona con la luz. Son las siguientes:

•**Perspectiva cónica:** propiedad intrínseca a nuestra a visión por la cual la proyección de todos los objetos converge en un punto (nuestra retina). Fruto de esta visión vemos los objetos lejanos más pequeños y las líneas paralelas tienden a converger en un punto.

---

·**Interposición o solapamiento:** por la cual un objeto que está detrás de otro no puede verse.

·**Iluminación y sombreado:** la iluminación y sombreado de los objetos da información sobre la geometría de los objetos.

·**Atenuación y desplazamiento cromático:** los objetos más lejanos pierden saturación y luminosidad, es decir tienden al gris. Además, su tonalidad cromática se desplaza hacia tonos azules.

·**Enfoque:** vemos menos nítidos los objetos lejanos.

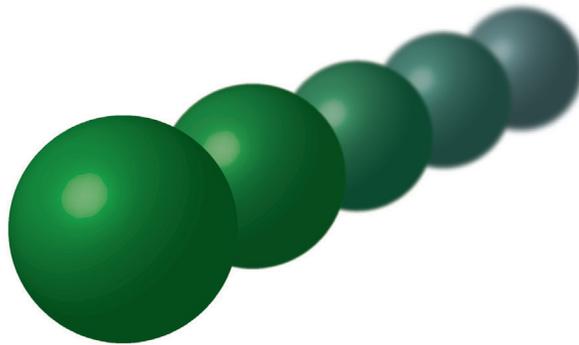


Figura 2. Efectos tridimensionales no estereoscópicos

### 3.2- Métodos auxiliares de observación estereoscópica

Como ya hemos visto, para poder ver de forma estereoscópica, necesitamos ver dos imágenes similares, una con cada ojo. Para que esto sea posible, es necesario preparar ambas imágenes para tal fin, y en la mayoría de casos utilizar un instrumento que dirija nuestra mirada a cada una de las imágenes. Vamos a ver el funcionamiento de dichos métodos:

**Estereoscopio visión cruzada:** este método no requiere ningún instrumento para poder ver las imágenes. Consisten en colocar las 2 imágenes una al lado de la otra, la correspondiente al ojo izquierdo se coloca a la derecha y viceversa. Para ver la imagen estereoscópica es necesario algo de entrenamiento, el método consiste en dirigir la mirada de cada ojo a su correspondiente imagen. De esta manera conseguimos que el cerebro superponga las imágenes.

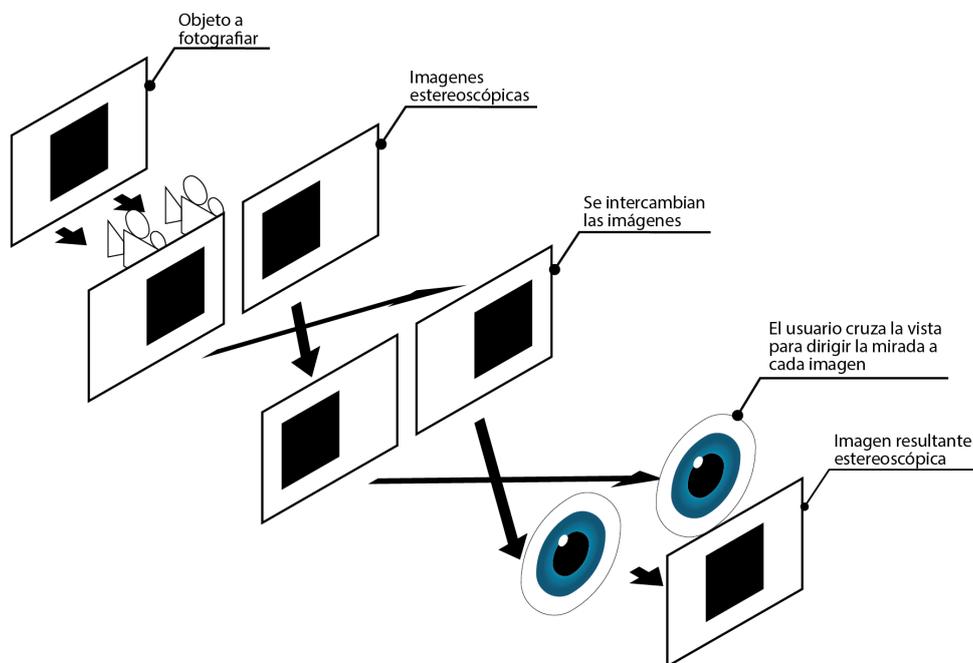


Figura 3. Esquema del estereoscopio de visión cruzada

**Visión paralela:** el principio es el mismo que el procedimiento anterior, pero las imágenes se cambian de lugar, de esta forma que cada ojo observa la imagen que está enfrente suya. Se llama paralela, porque en este tipo de visión ambos ojos miran de frente, sin cruzar sus caminos. Es un tipo de visión muy compleja de lograr sin la ayuda de un estereoscopio que, gracias a unas lentes, logra acomodar cada imagen a su respectivo ojo.

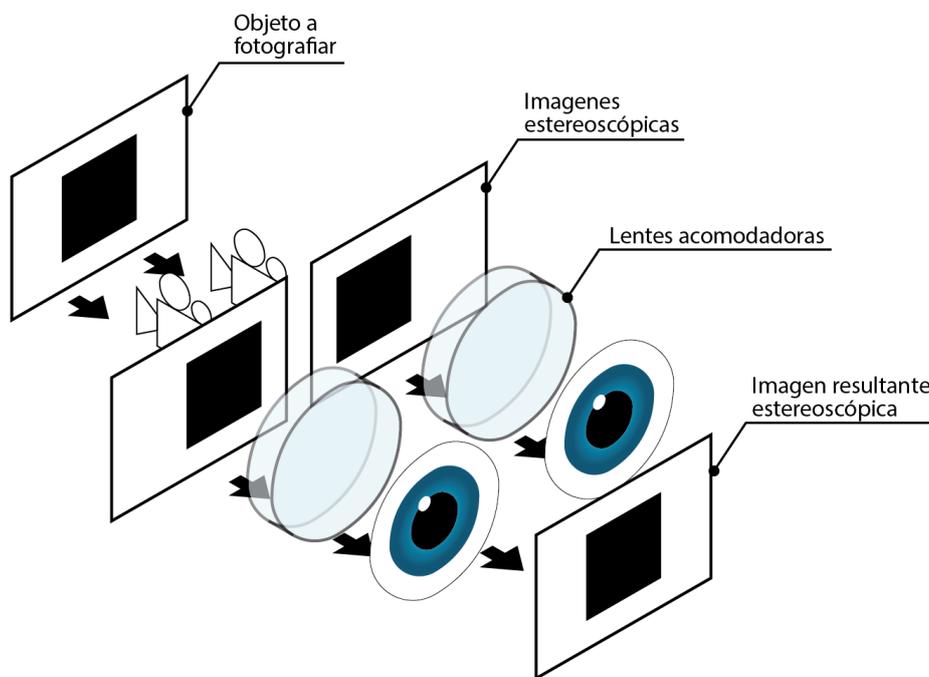


Figura 4. Esquema del estereoscopio de visión paralela



Figura 5. Estereoscopio antiguo

**Anaglifos:** un anaglifo es una imagen tridimensional compuesta por 2 imágenes superpuestas, una correspondiente al ojo derecho y la otra al izquierdo. Cada ojo percibe una de las imágenes de forma independiente gracias a un tratamiento de color que recibe cada imagen y unas gafas con filtros de color en sus lentes.

El método más utilizado es el de filtros rojo y cian, que son colores complementarios, premisa indispensable para que funcione dicha técnica. El procedimiento es el siguiente, se obtienen dos imágenes tomadas desde diferentes puntos en el espacio, estos puntos deben distar 65 mm, que es la distancia de separación de nuestros ojos. Cada imagen recibe un tratamiento de color, bien mediante filtros o con herramientas de retoque digital. La imagen izquierda, que será vista a través de un filtro rojo, recibirá un tratamiento mediante el cual eliminaremos los canales verde y azul. El resultado será una imagen en tonos rojos. Con la imagen derecha hacemos lo opuesto, eliminamos el canal rojo y la vemos a través de un filtro cian.

El cerebro a partir de las dos imágenes, creará una imagen virtual en la que los puntos separados parecerán estar en un plano anterior o posterior a la imagen, obedeciendo al siguiente esquema:

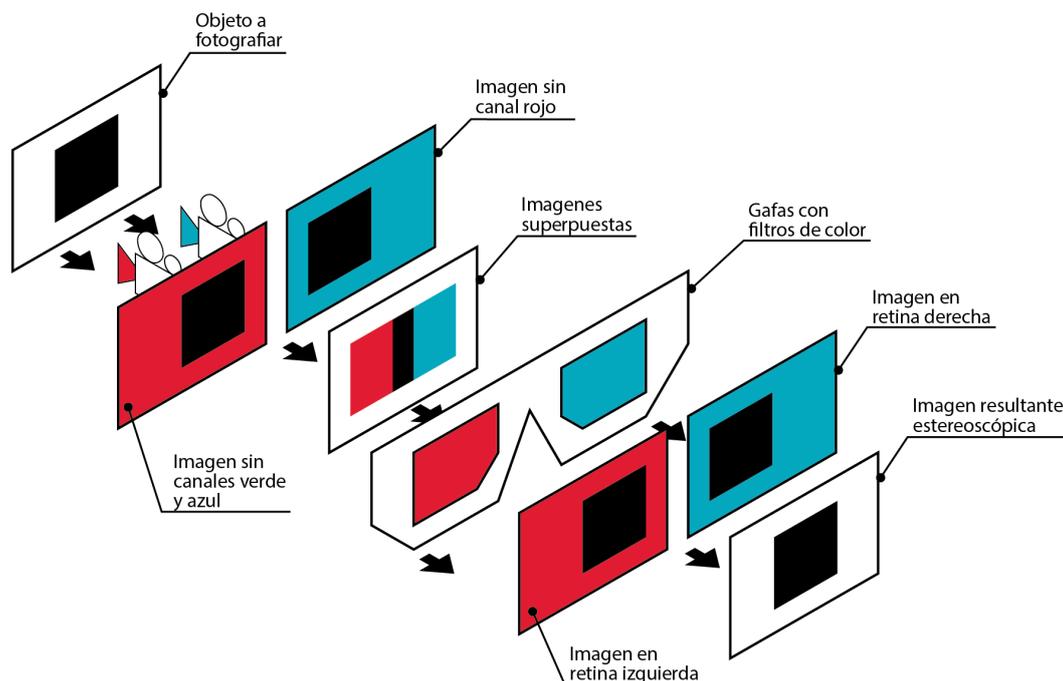


Figura 6. Esquema del funcionamiento de un anaglifo

**Polarización lineal y circular:** es un sistema de visión 3D pasivo, para entender como funciona esta tecnología, es necesario entender qué es la polarización de la luz.

La luz es una onda electromagnética cuyo campo eléctrico vibra en todas las direcciones del espacio. Cuando podemos predecir el campo de vibración en función del tiempo, decimos que la luz se encuentra polarizada. Existen dos tipos de polarización, que se aplican a este procedimiento, la lineal y la circular.

La lineal se consigue haciendo pasar la luz a través de un filtro polarizante. La circular, una vez que la luz se encuentra polarizada, se la hace pasar a través de una lámina retardadora de cuarto de onda. Vemos un detalle de ambos tipos en la siguiente ilustración:

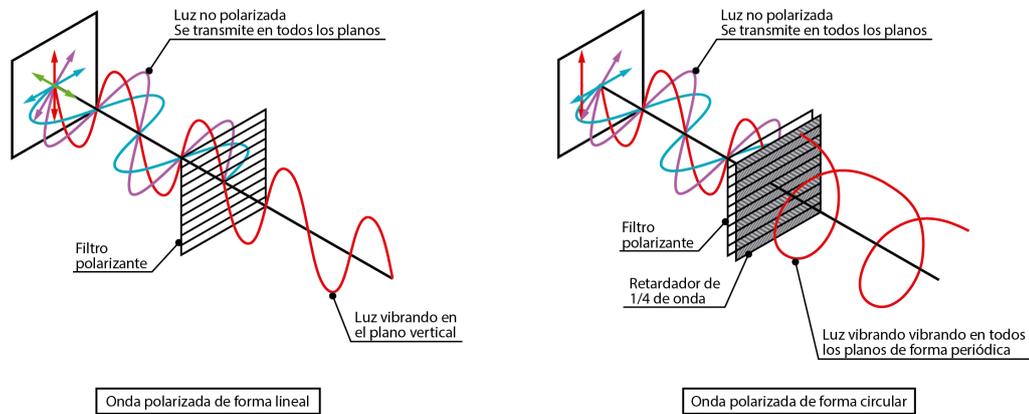


Figura 7. Esquema de la polarización de la luz

Los métodos de estereoscopía que utilizan la polarización, polarizan cada imagen en un sentido, de esta forma utilizando las gafas correspondientes, cada ojo ve una imagen individualmente.

La pega es que las gafas polarizadas linealmente son sensibles a los giros de nuestra cabeza, por lo que, si inclinamos la cabeza, el plano de vibración de cada onda deja de estar alineado con el de su respectiva lente, impidiéndonos ver nada.

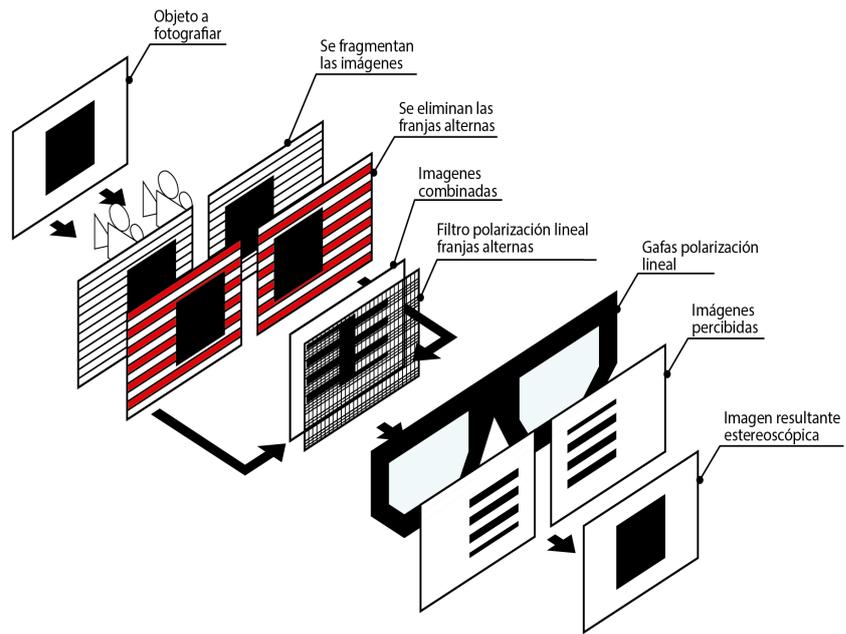


Figura 8. Esquema del visionado 3D mediante polarización lineal

El procedimiento que utiliza la polarización circular presenta más ventajas, puesto que no se ve afectado por los movimientos de cabeza. Cada imagen se polariza en sentido horario u antihorario y cada lente de las gafas permitirá el paso de su correspondiente imagen, permitiendo la visión estereoscópica.

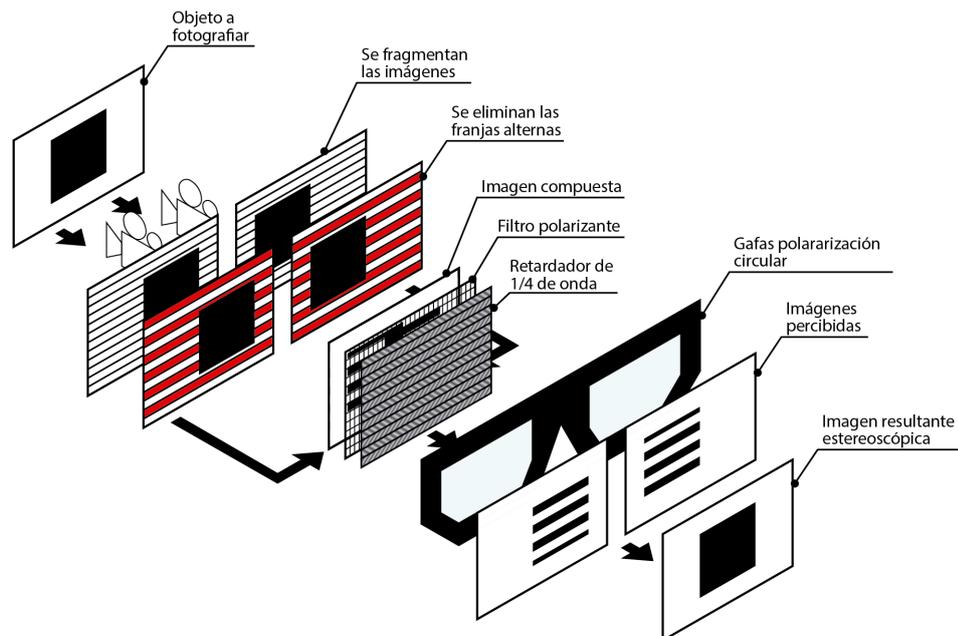


Figura 9. Esquema del visionado 3D mediante polarización circular

**Obturación:** es un sistema de visionado 3D activo, consiste en unas gafas de cristal líquido que bloquean la visión en cada ojo de manera intermitente. La pantalla emite los fotogramas de uno y otro ojo de manera alterna. Las gafas se sincronizan mediante infrarrojos con la pantalla, así se garantiza que cada ojo vea los fotogramas destinados a él.

El beneficio que presentan estas gafas es que no se pierde resolución de pantalla, pero presentan una serie de inconvenientes respecto de las gafas de 3D pasivo, ya que son más pesadas, hacen la imagen más oscura y pueden cansar la vista. Además, se necesita una frecuencia de refresco elevada para garantizar la calidad de la imagen.

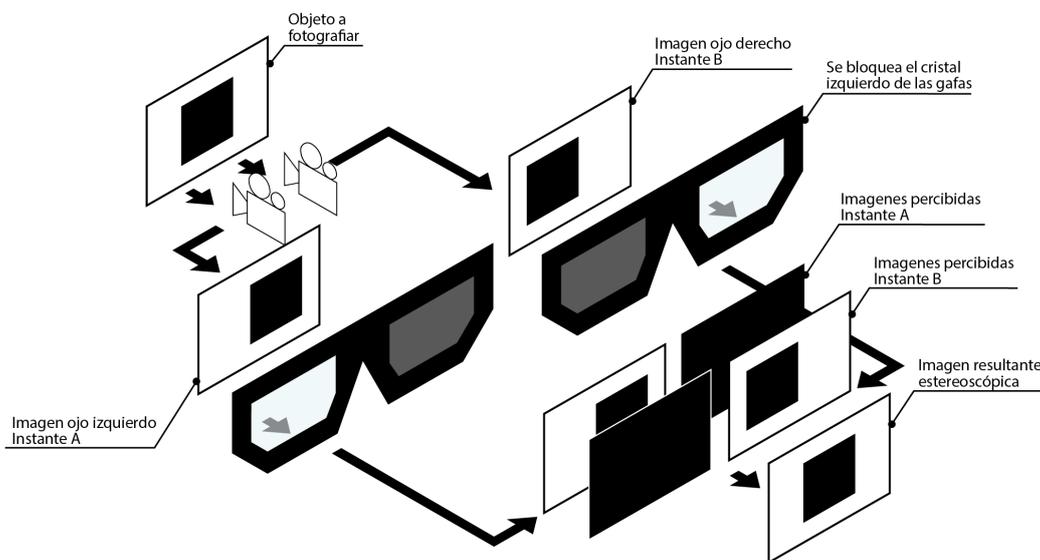


Figura 10. Esquema del visionado 3D mediante obturación

**Monitor lenticular:** consiste en un monitor que proyecta las 2 imágenes simultáneamente. Cada imagen aprovecha la mitad de la superficie de la pantalla, fragmentándose en rayas verticales alternas. Un filtro difracta cada una de las bandas en una dirección, de tal forma, que las columnas pares son proyectadas a un ojo y las impares al otro, de esta forma se consigue el efecto de estereoscopia. Vemos un esquema de su funcionamiento en la imagen de la siguiente página:

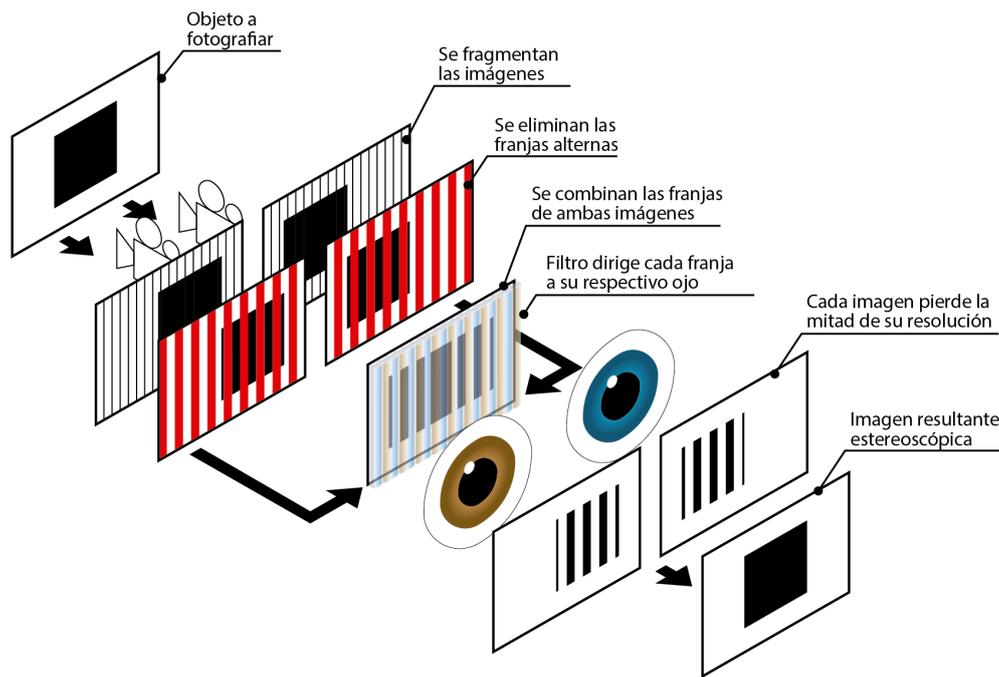


Figura 11. Esquema del visionado 3D mediante monitor lenticular

**Estereogramas:** los estereogramas son imágenes 2D que permiten ver en 3D. Se basan en la repetición de un patrón y para poder verlos es necesario un poco de entrenamiento. Es necesario poner los ojos bizcos de tal forma que la imagen de un ojo se solape con la del otro, encajando en diferentes puntos.

Los patrones que se repiten no son exactamente iguales, pero son lo suficientemente parecidos como para que el cerebro los una. De esta forma le estamos dando una imagen a un ojo y otra a otro, lo suficientemente semejantes como para que las una y lo suficientemente diferentes como para que se consiga el efecto de la estereoscopia.

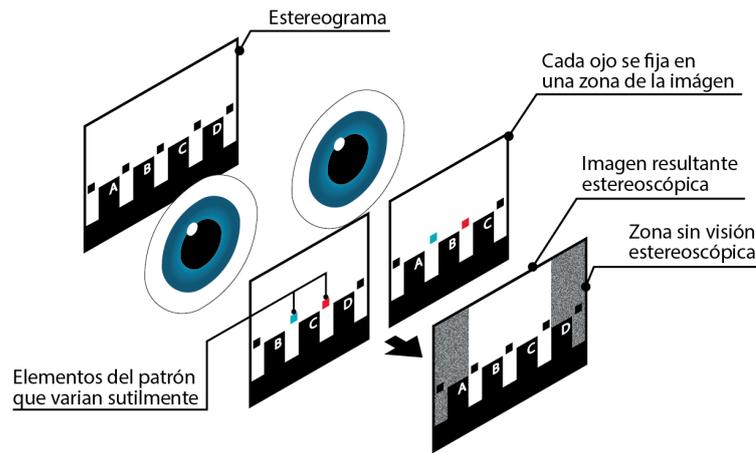


Figura 12. Esquema del visionado 3D mediante estereograma

**Gafas de realidad virtual:** es el sistema de visión estereoscópica más utilizado actualmente y el que proporciona una experiencia más inmersiva. Consiste en un dispositivo que se monta sobre los ojos a modo de gafas, y elimina los estímulos visuales externos, sustituyéndolos por los que proporciona a través de dos pantallas, una para cada ojo.

El funcionamiento es igual que el del estereoscopio de visión paralela, salvo por la diferencia de que las gafas sustituyen 2 fotografías por pantallas, haciendo posible la visualización de vídeos tridimensionales.



Figura 13. Usuaria disfrutando de las gafas de realidad virtual HTC Vive

### 3.3- Historia de la realidad virtual

Como ya hemos visto, la visión estereoscópica es la fusión que realiza el cerebro de dos imágenes obtenidas mediante cada uno de nuestros ojos. Así es como vemos la realidad y de esta manera nos deberían proporcionar las imágenes los dispositivos que quieren simular otra realidad. Esto nos proporciona dos puntos de visión, y nuestro cerebro mediante triangulación, puede apreciar inconscientemente la profundidad en su campo de visión.

Hoy en día el método auxiliar para la visualización estereoscópica más utilizado son las gafas de realidad virtual, que como ya hemos visto, aíslan nuestra visión del exterior y proporcionan una imagen a cada ojo mediante pantallas independientes.

Lograr una tecnología como esta no ha sido tarea fácil, muchos han sido los intentos por lograrlo en el siglo pasado, pero debido al desarrollo tecnológico de la época, lograr un producto de calidad a un precio asequible no fue tarea fácil como vamos a ver a continuación.

Antes de los primeros dispositivos de realidad virtual, ya había aparatos para poder ver de manera estereoscópica:

- **1838 - The stereoscope (Charles Wheatstone):** constaba de un dispositivo, que mediante 2 espejos, dirigía cada una de las imágenes a un ojo.

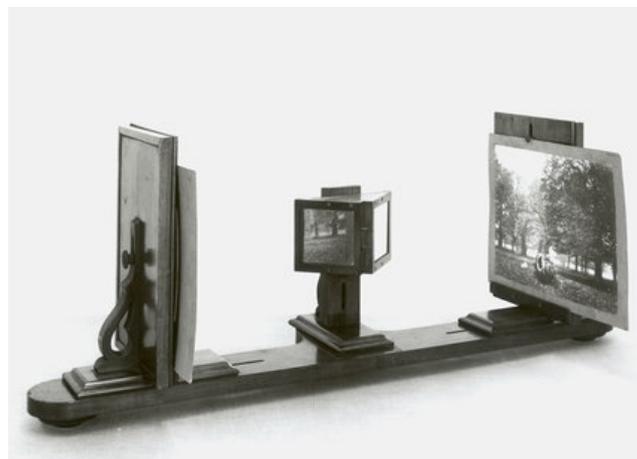


Figura 14. The stereoscope (Charles Wheatstone)

- **1849 - The lenticular stereoscope (David Brewster):** no dejaba de ser un estereoscopio de lentes paralelas, en el que se introducía una tarjeta con las 2 imágenes.

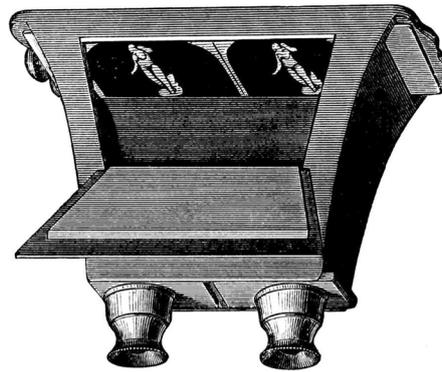


Figura 15. The lenticular stereoscope (David Brewster)

· **1891 - Anaglifo (Louis Ducos du Hauron):** el invento es patentado por su autor en 1891. Y en el año 1915 se empiezan a rodar las primeras películas que proporcionan una imagen diferente para cada ojo. Eso sí, eran necesarias unas gafas.

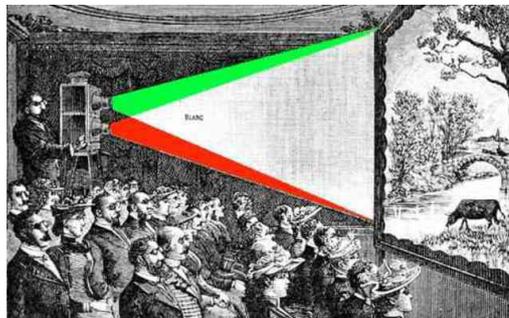


Figura 16. Película estereoscópica anaglífica

- **1939** : The View-Master (William Gruber): un estereoscopio de lentes paralelas que tenía la peculiaridad de que las imágenes estaban dispuestas en una rueda que al ir girando ofrecía diferentes entornos.



Figura 17. View Master (William Gruber)

- **1957 – Sensorama (Morton Heilig)**: su autor, un experto en tecnología, crea una cabina que pretende aislarnos de los estímulos externos y sustituirlos por los que ella genera. Disponía de una pantalla estereoscópica, sonido estéreo, generadores de olores, una silla vibradora, hasta un ventilador para simular viento. No fue hasta 1962, cuando se le concedió la patente.



Figura 18. Sensorama (Morton Heilig)

· **1960 - Telesphere Mask (Morton Heiling)**: es el primer HDM (head-mounted display) de la historia. Proporcionaba visión estereoscópica y sonido estéreo.



Figura 19. Telesphere Mask (Morton Heiling)

· **1961 – Headsight (Philco Corporation)**: dos ingenieros de dicha compañía, Comeau y Bryan, crean el primer HDM de la historia con rastreo de movimiento, lo que permitía observar a través de las gafas según la orientación de nuestra cabeza. Incorporaba una pantalla para cada ojo y un sensor magnético de movimiento, que era el responsable de conocer la orientación de la cabeza.

Era utilizado para entrenamientos inmersivos ante situaciones peligrosas por la industria militar. Y se le puede considerar como el precursor de las gafas de realidad virtual que conocemos hoy en día.

· **1968 – Sword of Damocles (Ivan Sutherland y Bob Sproull)**: es un dispositivo VR (virtual reality)/AR (augmented reality) conectado a un ordenador que genera los fotogramas según la observación del usuario. Debido a la escasa potencia computacional de la época, estas imágenes eran muy precarias a nivel gráfico.

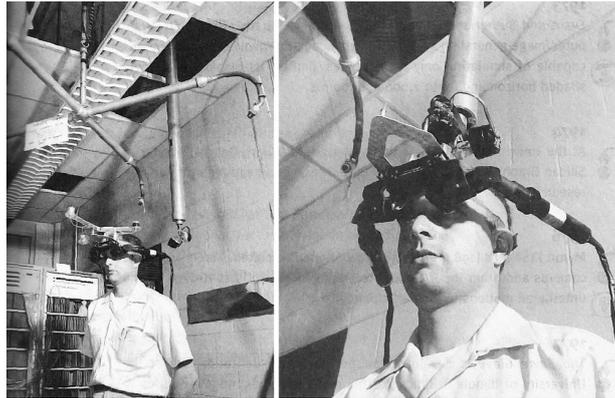


Figura 20. Sword of Damocles (Ivan Sutherland y Bob Sproull)

· **1987 – Nacimiento del término realidad virtual (Jaron Lanier):** hasta la fecha, todos los avances relacionados con este campo, no tenían un término específico para su designación. No es hasta fecha, cuando Jaron Lanier, fundador de la compañía VPL, acuñó este término a este campo tecnológico.



Figura 21. Jaron Lanier en su laboratorio de investigación

• **1989 – EyePhone (VPL):** la compañía de Jaron Lanier desarrolló equipos de realidad virtual, tales como el Dataglobe y las EyePhone. Las EyePhone eran unas gafas de visión estereoscópica, que permitían visualizar las manos del usuario si utilizaba también los guantes especiales Dataglobe.

De esta forma, VPL se convirtió en la primera compañía en vender al público gafas de realidad virtual. Aunque sus precios eran bastante privativos, las EyePhone 1 salieron a la venta con un precio de mercado de \$9400.



Figura 22. EyePhone 1 y Dataglobes (VPL)

• **1990 – Virtuality Head 4 Head (Jonathan Waldern):** es la primera aplicación de la realidad virtual a la industria del videojuego. Consistía en una consola con gafas de realidad virtual y un joystick que permitía unir hasta 4 consolas para ofrecer partidas multijugador.



Figura 23. Virtuality para 4 jugadores

- **1995 – Virtual Boy (Nintendo):** la compañía de videojuegos sacó a la venta esta consola que constaba de unas gafas de realidad virtual y un mando para controlar el juego. Fue un fracaso ya que el público no se adaptó a unos gráficos precarios en solo dos colores (rojo y negro) y su poca ergonomía.



Figura 24. Virtual Boy (Nintendo)

- **2012 – Oculus Rift (Oculus VR):** tras una etapa con poco desarrollo en este campo, la empresa fundada por Palmer Luckey, había venido para quedarse, puesto que la ergonomía y calidad de las gafas no tenía nada que ver con la de sus antecesoras. Todo esto ha sido posible gracias al enorme desarrollo tecnológico que tuvo lugar durante todo este tiempo, por un lado, las pantallas y sensores de las gafas no solo habían ganado en calidad, sino que también lo habían hecho en ligereza, además la potencia computacional que nos encontramos en un ordenador personal doméstico medio-alto, es suficiente para poder utilizar este tipo de gafas.

Gracias a estos acontecimientos, ha sido posible el desarrollo de la empresa Oculus VR y el posterior nacimiento de otras empresas y productos destinados a la realidad virtual.



Figura 25. Oculus Rift (Oculus VR)

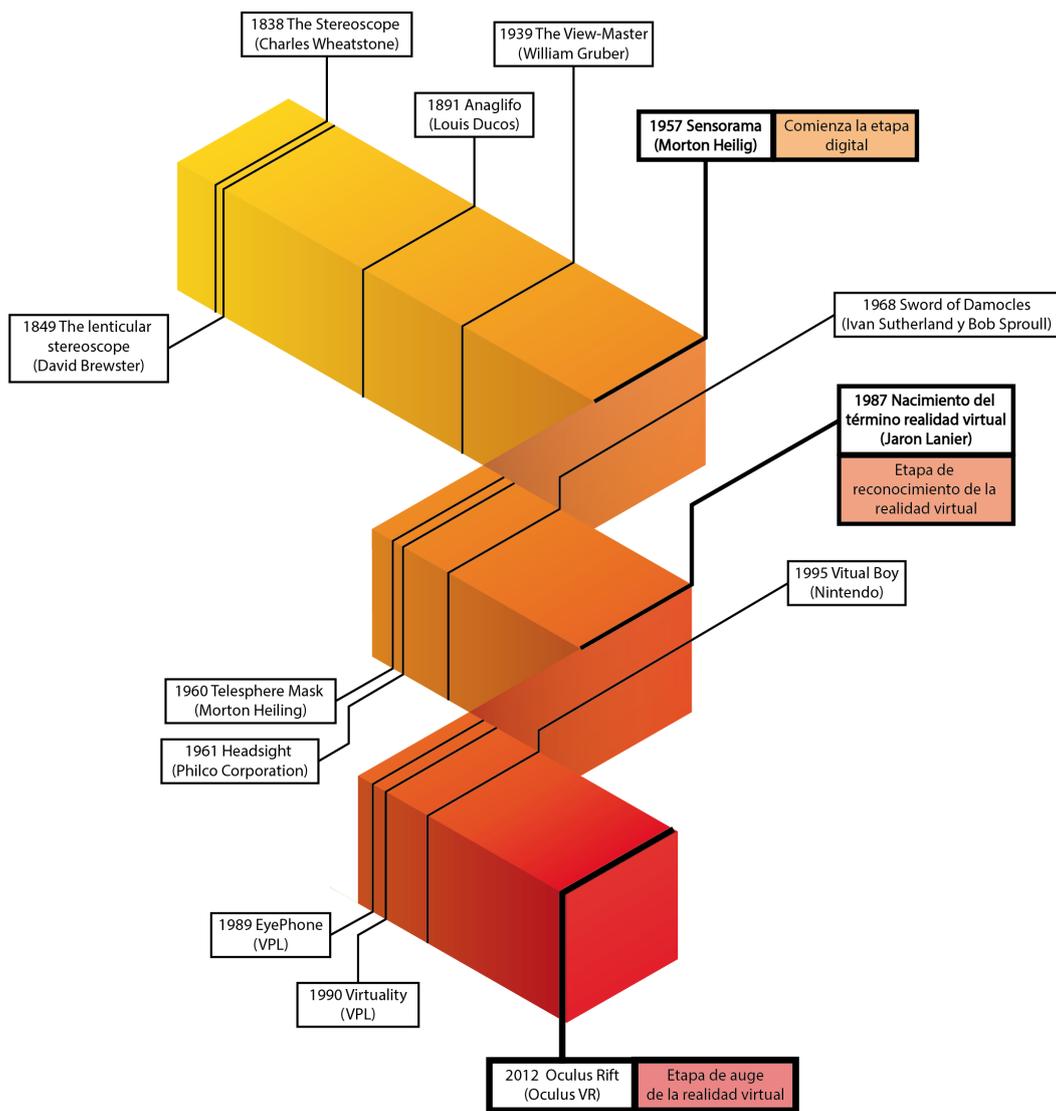


Figura 26. Cronograma de la historia de la realidad virtual

### 3.4- Dispositivos de realidad virtual

La gama de dispositivos que nos permite disfrutar de una experiencia inmersiva, no para de crecer. Desde el 2012, año en el que se crearon las Oculus, muchas compañías se han sumado a la aventura de la realidad virtual.

Actualmente podemos dividir estos dispositivos en 2 categorías, los dispositivos que cuentan con pantalla y los que se apoyan en la pantalla de nuestros teléfonos móviles para proporcionarnos las imágenes.

Vamos a ver un análisis de los dispositivos más relevantes, de ambas categorías hasta la fecha:

#### 3.4.1- HMD para smartphone

- **Google Cardboard:** es la apuesta más simple en cuanto a HMD se refiere, están compuestas por una carcasa de cartón en la que introducimos nuestro dispositivo móvil y unas lentes que acomodan la imagen a nuestros ojos. Actualmente hay muchos modelos basados en estas gafas, con carcasas de plástico más duraderas o gomas elásticas para ajustarlas a nuestra cabeza. Sus precios oscilan entre los 10-30€.



Figura 27. Gafas Cardboard de Google

- **Samsung Gear VR:** el modo de funcionamiento es muy parecido al de Google Cardboard, pero amplía un poco más su funcionalidad. Estas gafas, desarrolladas por Oculus, disponen de botones en su lateral derecho que nos permiten interactuar con el entorno. También está la opción de utilizar un mando con dichos botones.



Figura 28. Samsung Gear VR

• **Google Daydream:** es la respuesta de Google a las gafas de Samsung. También incorporan un controlador inalámbrico, mientras que se ahorran los botones en el lateral de las gafas.



Figura 29. Gafas Google Daydream

### 3.4.2- HMD integrales

• **HTC VIVE:** son unas gafas de realidad virtual que ofrecen una experiencia inmersiva de muy alta calidad. Proporcionan una resolución de imagen de 1080 x 1200 píxeles para cada ojo y un ángulo de visión de 110°. Disponen de acelerómetro y giroscopio para responder a nuestros giros y disponen de salida de audio para utilizar con unos auriculares externos.

Además, cuenta con algo que no posee ninguna gafa para smartphone, y es un rastreador de movimiento que permite al usuario desplazarse a lo largo del espacio. Esta tecnología, bautizada con el nombre de SteamVR Tracking, funciona con dos sensores láser, que se colocan en las esquinas de la habitación y rastrean la posición de las gafas. Los sensores se colocan en las esquinas del área a delimitar con una máxima separación entre sensores de 5 metros.

Todas estas características posicionan a las HTC VIVE como líderes en el mercado. La única pega, la parte económica, ya que las gafas cuestan unos 700€, a lo que hay que sumarle un equipo competente que vendría a costar en torno a 1000€.



Figura 30. HTC VIVE

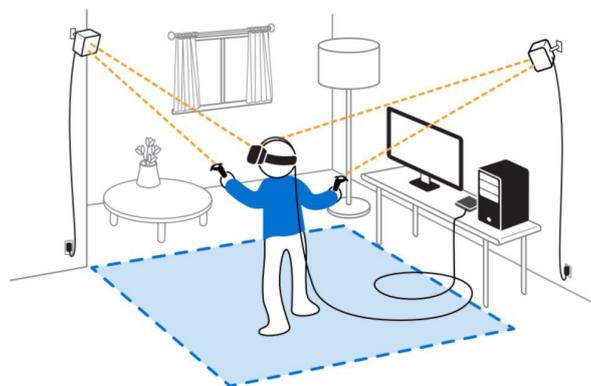


Figura 31. Esquema del funcionamiento de los sensores de las HTC VIVE

· **Rift:** este dispositivo HDM viene con auriculares integrados y también ofrecen una experiencia de una calidad a la altura de las HTC VIVE. A pesar de que ambas poseen las mismas funciones, las gafas de HTC son las favoritas por la comunidad de usuarios ya que su sistema de seguimiento cubre un área mayor.

También su precio es algo más modesto, en torno a los 550€ y no necesitan un ordenador con unas prestaciones tan elevadas.



Figura 32. Usuario utilizando las Oculus Rift

· **Playstation VR:** diseñadas para jugar con la famosa consola, no incorporan la tecnología de rastreo de movimiento, pero ofrecen muy buenas prestaciones. Con una resolución de imagen de 960 x 1080 píxeles para cada ojo y un ángulo de visión de 100° son unas buenas competidoras para las Oculus y las HTC VIVE.

Su precio, de 400€ las convierte en unas gafas relativamente asequibles, sobre todo para los usuarios que dispongan de una consola PlayStation 4.



Figura 33. Usuario utilizando las gafas v VR

## 4- FUNDAMENTOS DE LA VISIÓN HUMANA

### 4.1- Anatomía y fisiología del ojo humano

El ojo humano es el órgano encargado de la visión, su misión es convertir los estímulos visuales en señales eléctricas, que al llegar al cerebro producen el efecto que conocemos como visión. Vamos a ver de una forma superficial qué partes lo componen y su misión en el proceso visual.

Una analogía al comportamiento del ojo es el funcionamiento de una cámara de fotos, en la cual la luz atraviesa la lente, que refracta los rayos haciéndolos converger en el sensor. Este recopila toda la información visual y la dirige al procesador, que compila toda esa información creando una foto. Pues bien, nuestros ojos también tienen partes que actúan como lentes (córnea y cristalino) y como sensor (retina).

La forma de nuestros ojos es más o menos esférica, cuando la luz entra a nuestros ojos, lo primero que atraviesa es la córnea, una estructura cristalina cuyo objetivo es refractar los rayos de luz para dirigirlos a una segunda lente, el cristalino, cuya deformación es controlada por los músculos ciliares. El cristalino es el elemento clave en la tarea del enfoque ya que hace converger la luz en la retina y se deforma en función de la distancia a la que se encuentre el objeto a observar. Para acomodar la visión, el cristalino se contrae para objetos cercanos y se relaja para objetos lejanos. La facultad de enfocar objetos cercanos se va perdiendo con la edad, ya que el cristalino pierde flexibilidad y los músculos ciliares no consiguen deformarlo lo necesario. Un niño de 6 años tiene la capacidad de observar objetos a 6,3 cm de distancia, mientras que una persona adulta de 30 años necesita 15 cm para ver de forma nítida.

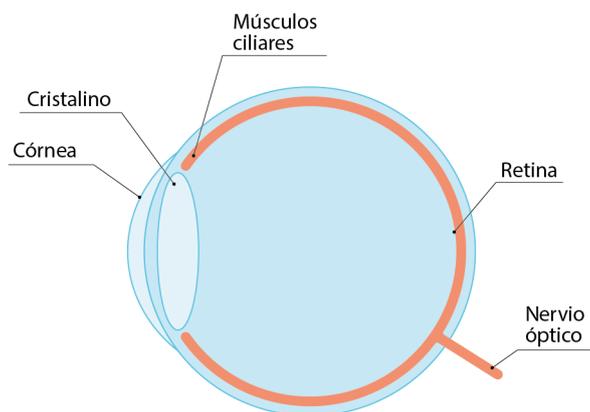


Figura 34. Esquema de las partes principales del ojo humano

A continuación, la luz llega a la retina, que se extiende por la parte posterior del globo ocular. La retina está compuesta por conos y bastones, los conos son los encargados de la visión con mucha luz, y nos proporcionan el color, los bastones funcionan cuando la luz es escasa y nos proporcionan visión en blanco y negro, ya que solo detectan la intensidad de la luz. Generalmente bastones y conos funcionan a la vez salvo en situaciones extremas de luz.

Por último, la información visual es convertida en energía eléctrica que llega al cerebro a través del nervio ocular. La parte del cerebro encargada de decodificar la información es el lóbulo occipital.

#### 4.2- Defectos de la visión

Gran parte de la población posee alguna alteración en la visión, hay 4 problemas de visión que son los más extendidos y se corrigen con lentes.

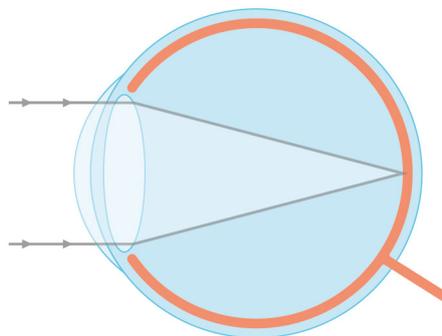


Figura 35. Esquema visión ojo sano

La hipermetropía y la presbicia tienen efectos similares, pero sus causas son diferentes. Ambos consisten en que el cristalino no se deforma lo suficiente y la imagen converge más allá de la retina. Mientras que la presbicia está relacionada con la edad y un cristalino rígido que el músculo ciliar no consigue deformar, la hipermetropía consiste en que, teniendo un cristalino sano, el músculo ciliar no tiene la fuerza necesaria para hacer converger los rayos en la retina. Se corrige mediante el uso de lentes convexas.

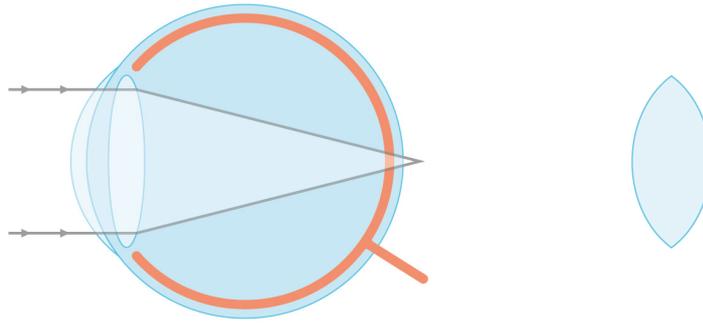


Figura 36. Esquema visión ojo hipermetrope y lente convexa

La miopía tiene el efecto contrario, el músculo ciliar tiene demasiada fuerza y los rayos visuales convergen antes de llegar a la retina. Se corrige con unas lentes cóncavas.

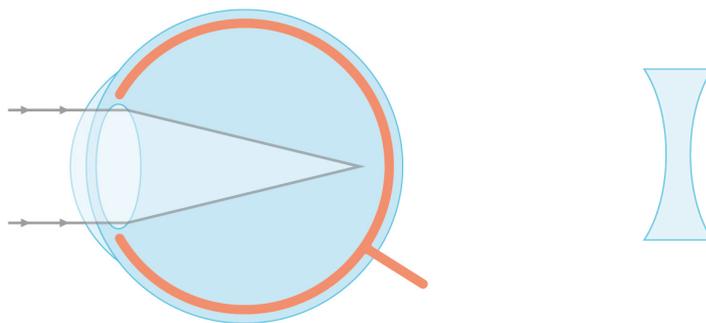


Figura 37. Esquema visión ojo miope y lente cóncava

Hay un último caso, el astigmatismo, relacionado con la geometría de la córnea. En un ojo con este defecto, la córnea no es exactamente circular, sino más bien elíptica. No se puede corregir con unas lentes simples, las lentes que se usan deberán estar hechas a medida y no son simétricas.

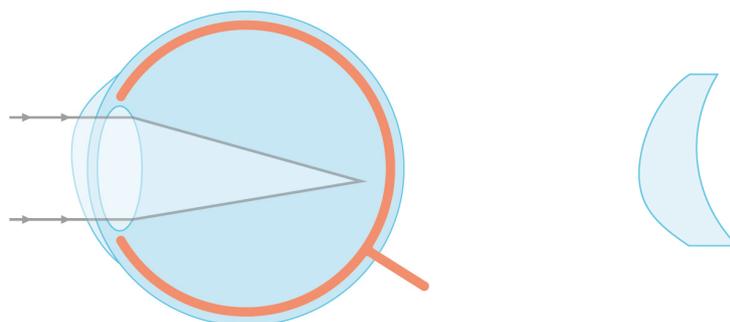


Figura 38. Esquema visión ojo astigmático y lente a medida

### 4.3- Ergonomía de los HMD

Como ya se ha comentado, las gafas de realidad virtual funcionan reemplazando los estímulos visuales de la realidad por otros creados artificialmente. El comportamiento entre unas gafas, por muy buenas que sean y la realidad no es exactamente el mismo, pues las gafas responden a nuestros movimientos con un tiempo de respuesta, sus pantallas no abarcan el mismo ángulo de visión que la visión humana y además las pantallas tienen una tasa de refresco que, si es baja, puede producir mareos.

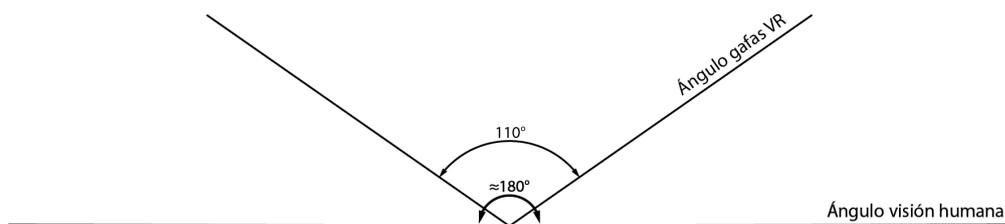


Figura 39. Ángulo de visión humana y gafas VR

Estas diferencias hacen que para nuestro cerebro sea un desafío procesar toda esta información, puesto que la visión humana responde a mecanismos biológicos que han evolucionado durante miles de millones de años. Por lo que recibir estímulos visuales alterados, puede producir mareos o dolores de cabeza. Este es un hecho que depende en parte de la tecnología, pero también nosotros podemos contribuir a minimizarlo, utilizando imágenes sin distorsión, tomando las imágenes a una altura del suelo coincidente a la altura de nuestros ojos (se suele utilizar 1,70 m como valor medio) y separando las dos cámaras a la medida de la separación entre nuestros ojos (se suele tomar 65 mm como valor medio).

Hay otro aspecto con el que lidian los dispositivos HMD, y es lo cerca que está la pantalla del ojo. La pantalla, al encontrarse a escasos centímetros, no podría ser percibida correctamente por nuestros ojos, ya que el cristalino no se puede deformar tanto. La solución viene de mano de una lente convexa, que proporciona la imagen al ojo de tal forma que el cristalino se puede relajar como si estuviera mirando al infinito. Así, el ojo permanece relajado evitando que sufra daños.

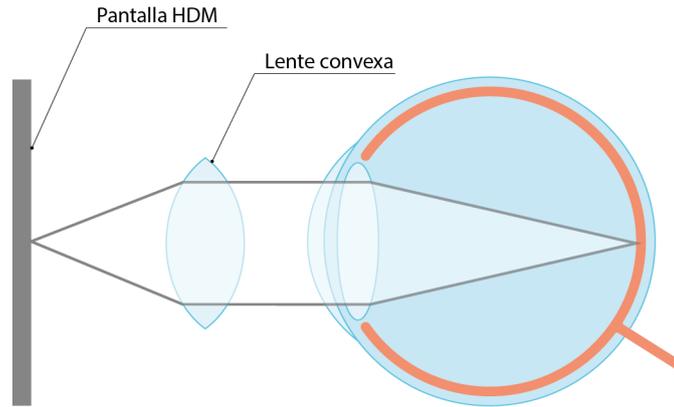


Figura 40. Esquema acomodación HDM

#### 4.3.1- Ergonomía de los HMD para usuarios con problemas de visión

Las gafas de realidad virtual pueden tener en cuenta a la gente con problemas de visión, mediante un botón de enfoque que modifique la distancia entre la lente y la pantalla. De esta forma los rayos entre la lente y la pantalla, que en un principio eran paralelos, pueden hacerse convergentes, para un ojo hipermetrope o divergentes para un ojo miope.

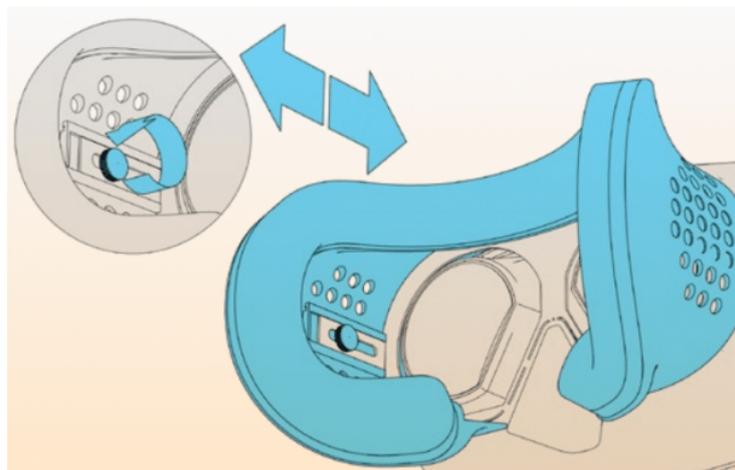
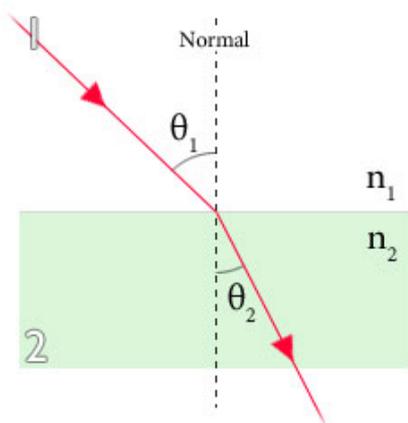


Figura 41. Esquema del ajuste para problemas de visión en un HDM

En las gafas actuales, el ajuste, se hace moviendo la pantalla respecto de la lente. Para entender como se modifica la dirección de los rayos con este gesto, es necesario comprender la Ley de Snell, que estudia la refracción de la luz al pasar de un medio a otro.



### Ley de Snell

$$n_1 \text{ sen } \theta_1 = n_2 \text{ sen } \theta_2$$

Donde:

$n_1$  : Índice de refracción del medio 1

$n_2$  : Índice de refracción del medio 2

Figura 42. Esquema de la Ley de Snell

Según la Ley de Snell, independientemente de los índices de refracción de cada medio, si aumenta el ángulo de incidencia, aumenta el ángulo de refracción. Por lo cual, al separar la pantalla de la lente, el ángulo de incidencia de los rayos sobre esta será menor, refractándose menos al salir, por lo cual como se puede ver en la ilustración, los rayos refractados serán convergentes, por lo cual la imagen podrá ser vista por un ojo hipermetrope. Por consiguiente para un ojo miope el procedimiento consistirá en acercar la pantalla a la lente.

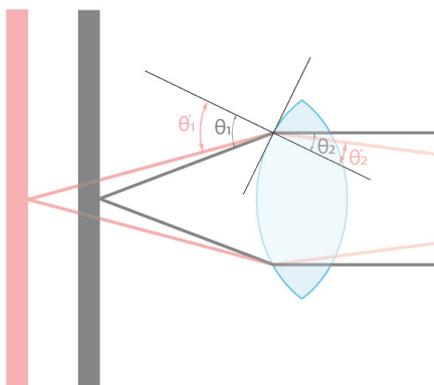


Figura 43. Esquema de la adaptación de un HDM a usuarios con problemas de visión

---

## 5- SOFTWARE

### 5.1- Unity

Unity es un motor de videojuegos desarrollado por Unity Technologies. Es un software ideal para desarrollar aplicaciones a pequeña escala ya que es gratuito para usos no comerciales o para pequeñas empresas cuyos ingresos no excedan de \$100.000 brutos anuales. Además, es una plataforma que permite exportar un mismo proyecto a multitud de plataformas sin la necesidad de hacer cambios significativos en el mismo.

Los inicios en Unity para alguien que no está familiarizado con software de desarrollo de videojuegos no son sencillos, pero rápidamente gracias a la cantidad de información que encontramos en las redes uno se familiariza rápido con todos los términos.

Vamos a ver los componentes principales del menú de Unity:

- **Scene:** muestra tridimensionalmente todos los elementos físicos que componen el proyecto de Unity. En la escena podemos seleccionar los objetos, moverlos o modificarlos, haciendo clic sobre ellos.
  
- **Game:** es la ventana en la que podemos activar la vista previa del proyecto con solo presionar el botón de Play, es muy útil para corregir posibles fallos en el proyecto sin la necesidad de exportar a la plataforma de destino.
  
- **Asset Store:** es una tienda en línea en la que podemos encontrar paquetes de Assets destinados a todo tipo de aplicaciones. Estos paquetes pueden ahorrarnos mucho tiempo en la elaboración de un proyecto.

- **3D Objects:** podremos crear objetos predeterminados como esferas, cubos, planos... Y aplicarle distintos materiales.

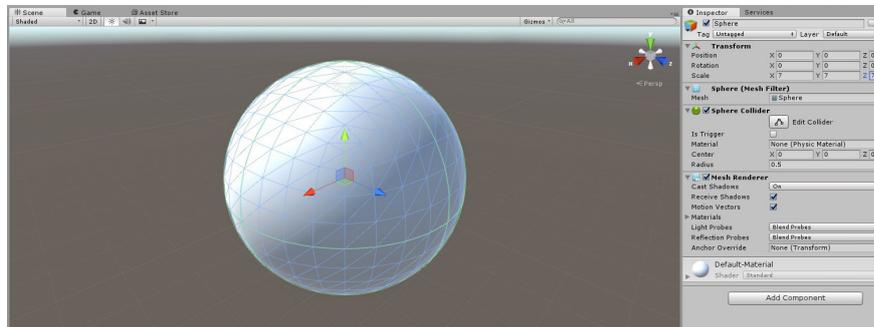


Figura 44. Esfera en la Scene de un proyecto de Unity

- **Mesh Filter:** un objeto 3D, está formado por una maya finita de puntos, la geometría y número de polígonos de dicha maya determinará la geometría del cuerpo. A continuación, vemos una esfera con un Mesh Filter compuesto por 32.768 polígonos, en contraposición con el Mesh Filter para esfera que viene con Unity, de 768 polígonos.

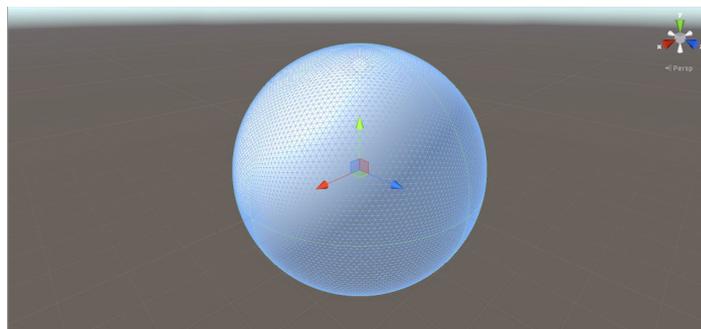


Figura 45. Esfera con un Mesh Filter de 32.768 polígonos

- **Materials:** los materiales de Unity permiten aplicar una textura, color a un objeto, modificar su tiling... Estas funciones cambian dependiendo cual sea su shader.

· **Shaders:** son scripts pequeños que contienen los cálculos matemáticos y algoritmos para calcular el color de cada píxel renderizado, basándose en el input de iluminación y la configuración del Material. Todo material tiene asociado un Shader, que delimita las posibilidades de dicho material, modificando un Shader podemos hacer visible la superficie interna de un objeto, modificar su brillo...

· **Cameras:** las cámaras de Unity serán nuestros ojos una vez creada la aplicación. Además, hay cámaras estereoscópicas predefinidas, que son ideales para exportar a VR.

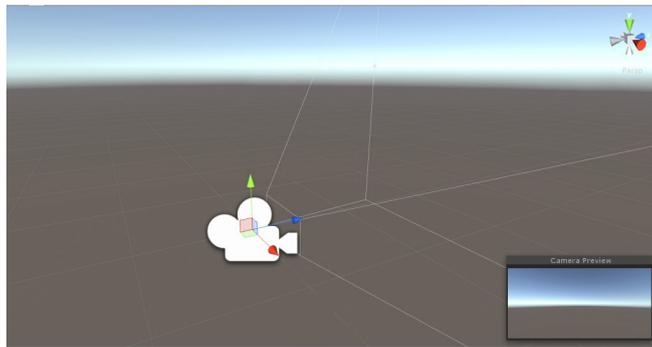


Figura 46. Camera de Unity, con área de enfoque y previsualización de enfoque

· **Assets:** son paquetes que contienen archivos que forman una escena o parte de ella. Los archivos que podemos almacenar son de todo tipo, objetos 3D, imágenes, cámaras, scripts... Podemos guardar paquetes de Assets que consideremos interesantes para posteriormente utilizarlos o importar paquetes predefinidos que contienen Assets destinados a una finalidad concreta.

· **Scripts:** es un archivo de texto escrito en lenguaje C# o UnityScript. Su función es muy versátil, ya que se comunica con el funcionamiento interno de Unity. Se pueden utilizar para que se inicien eventos ante diferentes acciones del usuario, también sirven para definir el comportamiento de un material... Los contenidos iniciales del archivo tendrán la siguiente apariencia:

```
1  using UnityEngine;
2  using System.Collections;
3
4  public class NewBehaviourScript : MonoBehaviour {
5
6      // Use this for initialization
7      void Start () {
8
9      }
10
11     // Update is called once per frame
12     void Update () {
13
14     }
15 }
16
```

Figura 47. Script base de Unity en C#

En donde la clase **MonoBehaviour** sirve para establecer conexiones con el funcionamiento interno.

La función **Start** función que se ejecuta antes de que comience a ejecutarse la aplicación.

La función **Update** contiene el código que se encargará de modificar la escena en cada fragmento de tiempo. Responde a los inputs del usuario, por lo cual es la función que permite la interacción entre el usuario y la escena.

· **Build settings:** una vez que hemos creado la escena, en Build Settings, definimos la configuración de salida de nuestra aplicación. En este menú elegimos la plataforma de salida, la resolución de la aplicación, su icono y los nombres de los desarrolladores. Además, es necesario importar a la escena un paquete de Assets correspondiente con la plataforma de destino.

### 5.1.1- Software complementario a Unity

A pesar de que Unity es una herramienta muy potente, necesita apoyarse en software externo para exprimir todas sus posibilidades. En este estudio, se exportarán las aplicaciones a las plataformas HTC y Android, por ello es necesario descargar algunas herramientas que hagan posible exportar a estos dos soportes.

No obstante, parte del software es independiente de estas plataformas y es imprescindible para cualquier proyecto, como el Microsoft Visual Studio o el Java JDK.

- **Microsoft Visual Studio:** proporciona herramientas para desarrollar código en lenguaje C# y compilarlo en Unity. Además, ofrece un análisis del código detectando errores.
- **Android Studio:** proporciona las herramientas más rápidas para crear apps en todas las clases de dispositivos Android.
- **Android SDK tools:** se incluye en el Android Studio y contiene las herramientas necesarias para crear aplicaciones Android.
- **Java JDK:** set de herramientas para crear o testear programas creados en lenguaje Java.
- **Cardboard SDK:** paquete de Assets con todos los contenidos que necesita una aplicación para Android.
- **Steam VR:** paquete de Assets que agrupa todos los archivos imprescindibles en la creación de una aplicación para las gafas HTC VIVE.

## 5.1.2- Realidad virtual en Unity

Unity es una plataforma muy cómoda para trabajar en proyectos de realidad virtual, ya que dispone en su biblioteca de cámaras estereoscópicas, lo cual hace muy sencillo el proceso de generar una imagen para cada ojo. Además, como ya se ha comentado, ofrece multitud de plataformas de destino.

Se pretende analizar las técnicas de representación estereoscópica a través de dos métodos un tanto diferentes. Mientras que un procedimiento se apoya en la fotografía tridimensional para generar aplicaciones de realidad virtual, el otro utiliza directamente un modelo 3D.

Vamos a ver por qué Unity es la plataforma ideal para ambos métodos:

### 5.1.2.1- Visualización estereoscópica a partir de imágenes

Es un método interesante, ya que no se necesita un modelo 3D previo para su desarrollo. Por lo que se convierte en una herramienta ideal cuando se trata de representar elementos orgánicos o escenas con multitud de objetos. Y es que, las imágenes estereoscópicas en las que se apoya, pueden obtenerse mediante procedimientos físicos, como la fotografía digital. No obstante, también se pueden obtener dichas imágenes a partir de un modelo 3D y un software de renderizado.

Una vez tenemos las imágenes estereoscópicas, que pueden ser esféricas, semiesféricas, cilíndricas o planas. Tenemos que conseguir que se visualicen sin deformación, debemos para ello proyectarlas sobre cuerpos geométricos afines y observarlas desde el lugar adecuado.



Figura 48. Imagen esférica (elaboración propia)

La imagen esférica se proyectará sobre una esfera y el punto de observación se situará en el centro de la esfera, de esta manera, enfocaremos solamente un pequeño área de la misma, que corresponderá con nuestro ángulo de visión, pudiendo dirigir nuestra mirada a cualquier punto. Solo así, observamos la fotografía como si nos encontrásemos en el lugar donde fue tomada.

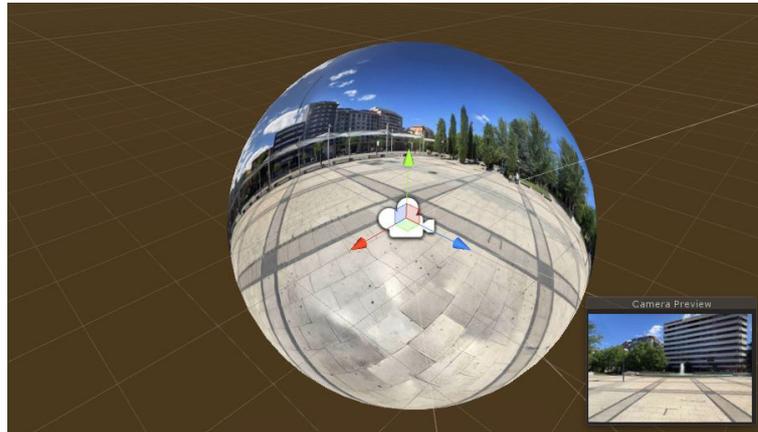


Figura 49. Imagen esférica proyectada sobre una esfera en Unity

Por consiguiente, la imagen semiesférica se proyectará sobre una semiesfera y se observará en el centro de la esfera coincidente. Cabe advertir que, si dirigimos nuestra mirada fuera de la semiesfera, no veremos imagen.



Figura 50. Imagen semiesférica proyectada sobre una semiesfera en Unity

La fotografía cilíndrica, se proyectará sobre un cilindro con las proporciones entre altura y diámetro correspondientes a las dimensiones de la imagen y se observará desde el centro del cilindro. En este caso no podremos ver fotografía hacia arriba ni hacia abajo.

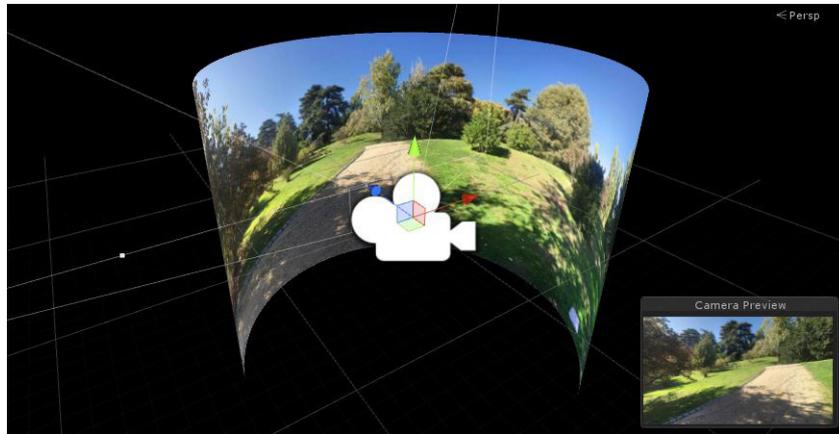


Figura 51. Imagen cilíndrica proyectada sobre un cilindro en Unity

Por último, para la fotografía plana, que es el más sencillo de los casos, proyectaremos la imagen sobre un plano, que será observado a una distancia adecuada sobre la normal al centro del plano.

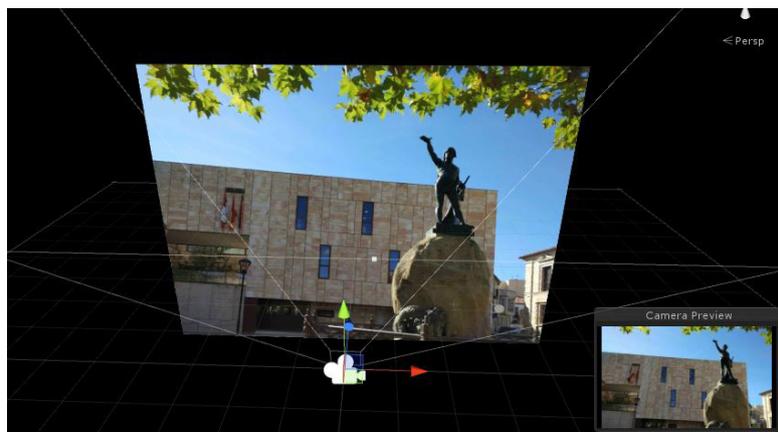


Figura 52. Imagen plana proyectada sobre un plano en Unity

Ya hemos explicado como representar las imágenes para verlas en verdadera magnitud, un procedimiento sin mayor complicación, pero no se ha comentado como representar las parejas estereoscópicas y destinar una a cada ojo.

En primer lugar, se crean dos cuerpos 3D afines a las imágenes, sobre cada uno proyectamos una de las imágenes del par estereoscópico. A continuación, creamos 2 cámaras estereoscópicas en el lugar correspondiente y hacemos que una cámara solo observe un cuerpo y la otra el otro. En la cámara que enfoca la imagen izquierda, desactivamos la visión con el objetivo derecho y hacemos lo contrario con la otra cámara, de esta forma cada ojo observa su imagen correspondiente. Pudiendo ver la escena de una forma completamente inmersiva.

Falta añadir, que si queremos visualizar vídeo, proyectaremos el vídeo, convertido a MovieClip sobre cuerpos afines a sus fotogramas. Siendo el proceso equivalente al uso de imágenes.

#### 5.1.2.2- Visualización estereoscópica a partir de modelo 3D

Este método es mucho más fácil de poner en práctica, ya que solo hay que importar el modelo 3D a Unity y crear una cámara estereoscópica para su visualización. Tiene como contra que se necesita un modelo 3D que visualizar, lo cual no compensa para todo tipo de representaciones.

La visualización se produce a través de la cámara, que renderiza a tiempo real nuestro campo de visión, por lo que se necesita una potencia computacional alta.



Figura 53. Modelo 3D de una bodega en Unity

## 5.2- Software de apoyo

Unity es el programa con el que se van a generar las distintas aplicaciones para los dispositivos HDM. Algunas aplicaciones requerirán de modelos 3D, otras de pares de imágenes estereoscópicas o incluso vídeo estereoscópico.

Todos estos componentes, antes de ser importados a Unity, han de recibir el correspondiente tratamiento. Para ello nos apoyamos en software externo.

Vamos a ver qué software se ha utilizado y cuál ha sido su función.

### 5.2.1- Panoweaver

Panoweaver es una herramienta de fotografía profesional costura, que puede coser todo tipo de fotos en alta calidad de 360 grados o parcial imágenes panorámicas. [2]

Gracias a esta herramienta podemos generar los pares estereoscópicos panorámicos, tanto cilíndricos como esféricos. Para ello, solo tenemos que importar las imágenes correspondientes a un panorama, indicarle al programa de qué tipo de panorama se trata y automáticamente realizará el cosido entre imágenes.

Las imágenes que introduceremos en este software serán obtenidas con una cámara réflex por el autor del proyecto.

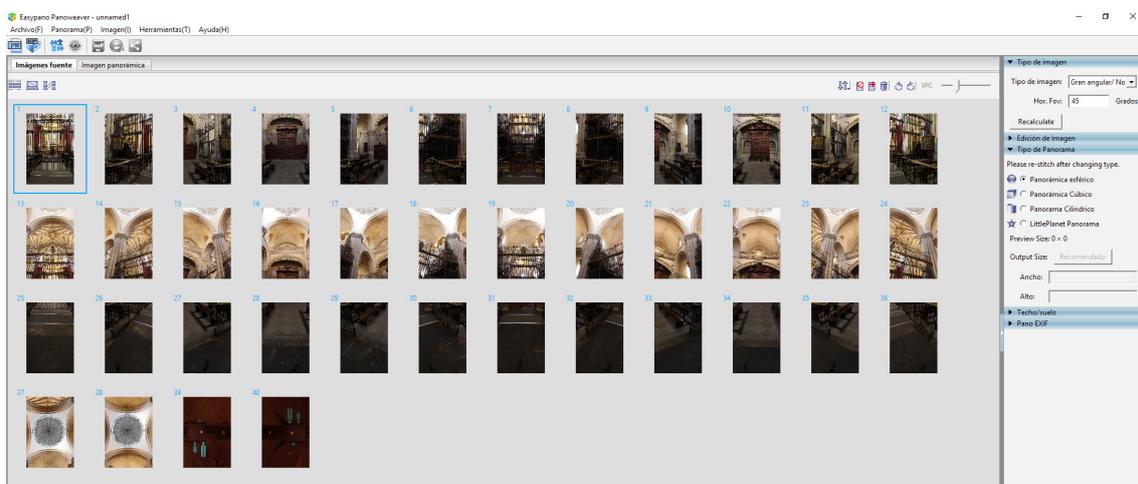


Figura 54. Fotos necesarias para completar el panorama esférico de la Catedral de Zamora



Figura 55. Resultado de la panorámica esférica de la Catedral de Zamora (elaboración propia)

Puede ocurrir que haya imágenes con tanta semejanza, como es el caso de imágenes de cielo, que Panoweaver no sepa establecer puntos de cosido entre ambas. Para ello este software tiene una opción muy interesante que permite guardar los parámetros de cosido de algún proyecto.

En el caso de la catedral, que es un escenario donde la panorámica resultante da muy buenos resultados, podemos guardar los parámetros de cosido y aplicárselo a alguna imagen donde el cielo diese fallos. Es muy importante introducir las imágenes en el orden equivalente al anterior proyecto.



Figura 56. Resultado del cosido con datos guardados del mirador del Troncoso de Zamora (elaboración propia)

Aunque, como se observa, los objetos no presentan buen cosido entre sí, esto se debe a que el programa proyecta las imágenes en el sitio correspondiente al anterior proyecto. No teniendo por qué coincidir exactamente en el mismo lugar. La zona que mejor aspecto presenta es el cielo, ya que al ser tan similar no hace falta tanta precisión para que las imágenes encajen de forma aceptable entre sí.

Una buena forma de solucionarlo es boquear las imágenes que únicamente contengan cielo, y realizar el panorama sin esas fotografías.



Figura 57. Resultado del cosido del mirador del Troncoso de Zamora sin las imágenes de cielo (elaboración propia)

El resultado será un panorama con algunas zonas en negro, pero buen cosido entre imágenes. De esta manera, si utilizamos la mejor zona de cada panorama, podemos crear un panorama esférico de muy buena calidad.

### 5.2.2- Google Street View

Google Street View es una aplicación para smartphone que permite crear fotografías panorámicas de una forma muy eficiente. Su eficiencia reside en el uso que hace de los sensores del terminal para proyectar las imágenes en el lugar correspondiente del panorama.

Una vez completada la toma de imágenes, será el propio terminal quien ponga la potencia computacional para procesar el cosido de imágenes.

Como ya se ha comentado, se utilizará un soporte para smartphone para dicha tarea.



Figura 58. Panorama esférico del mirador del Troncoso de Zamora obtenido con la aplicación Google Street View (elaboración propia)

### 5.2.3- Photoshop

La mejor aplicación de edición de imágenes y diseño del mundo constituye la base de casi todos los proyectos creativos. [3]

En este proyecto Photoshop será muy útil para remover el trípode de los panoramas. Para ello solo hay que invertir la imagen y aplicar el filtro de coordenadas polares. De esta forma podemos visualizar el trípode sin apenas deformación.

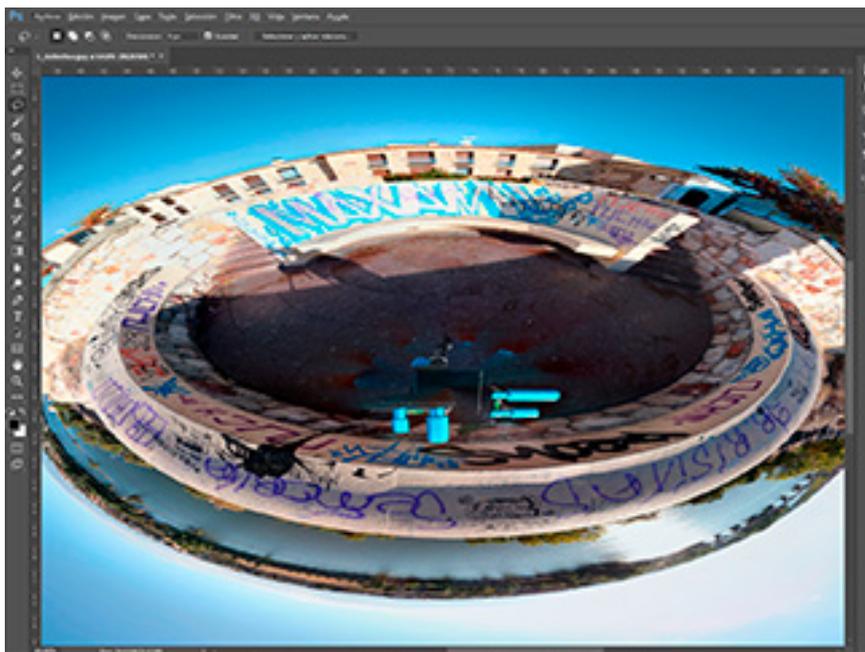


Figura 59. Vista que nos facilita eliminar el trípode de la imagen (elaboración propia)

A partir de aquí, podemos utilizar una imagen del suelo sin trípode o utilizar herramientas de clonado para conseguir el resultado deseado.



Figura 60. Panorama esférico sin trípode (elaboración propia)

Además, existe otro problema que afecta a los pares estereoscópicos tomados en el exterior. El conflicto consiste en que las nubes del cielo, muy difícilmente van a cuadrar en las dos imágenes, debido al movimiento de las mismas.

A continuación, observamos un par estereoscópico en el cual apreciamos ligeras diferencias en el cielo que harán que visualizemos dichas zonas borrosas. Incluso en imágenes que han sido tomadas simultáneamente, empleando dos smartphones, podemos apreciar estas incongruencias entre imágenes, como es el caso del par expuesto a continuación.

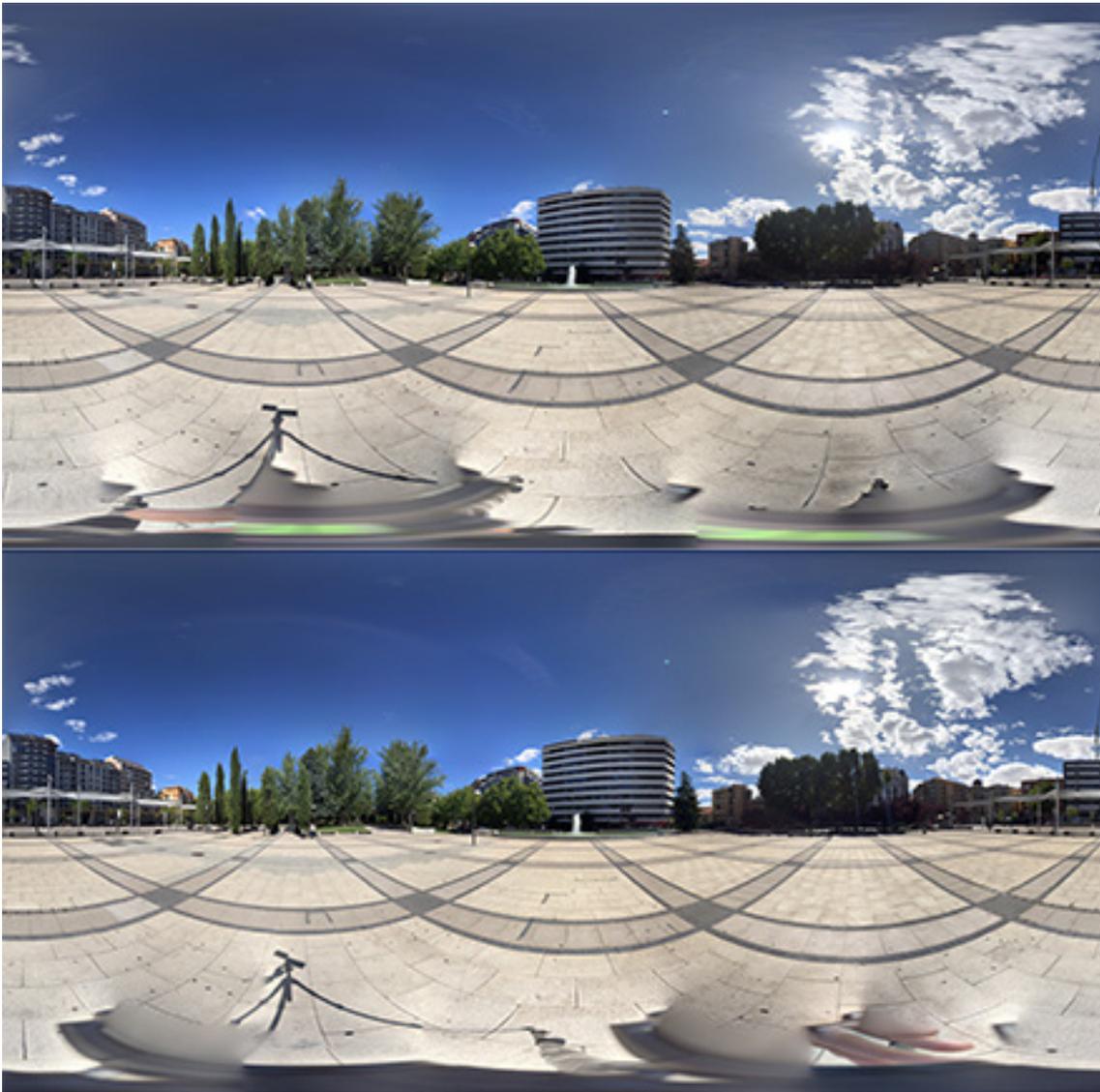


Figura 61. Par estereoscópico con diferencias en el cielo (elaboración propia)

En la siguiente imagen se muestra como se ha modificado el cielo de la misma para hacerlo coincidir con su pareja.



Figura 62. Panorámica con el cielo modificado (elaboración propia)

Es un proceso laborioso, que requiere recortar la imagen con el cielo que queramos conservar, hacer coincidir unos objetos con otros y posteriormente ir borrando dichos objetos hasta que solo nos quedemos con cielo. De esta forma la imagen resultante casará perfectamente con su pareja.

#### 5.2.4- Premiere

Es un programa de edición de vídeo de la familia de Adobe Creative Cloud. Es un componente necesario para sincronizar los vídeos que vamos a emplear, ya que en algunos casos se tratará de vídeos obtenidos con cámaras independientes.

Premiere nos permite ajustar ambos vídeos con una precisión de un fotograma, por lo cual el resultado será muy preciso. Además, nos permitirá exportar cada vídeo relativo a un ojo de forma individual y extraer la pista de audio de forma separada.

### 5.2.5- 3ds Max

3ds Max®, el software de modelado y renderización en 3D, le permite crear grandes mundos en juegos, escenas impactantes para la visualización y contagiosas experiencias de realidad virtual (VR) [4].

Este software es imprescindible en este proyecto, gracias a él podemos generar formas geométricas sobre las que proyectaremos los pares estereoscópicos. Unity, tiene una cantidad de geometrías muy reducida que no podremos modificar, gracias 3ds Max, podremos crear todas las geometrías que Unity no nos facilita.

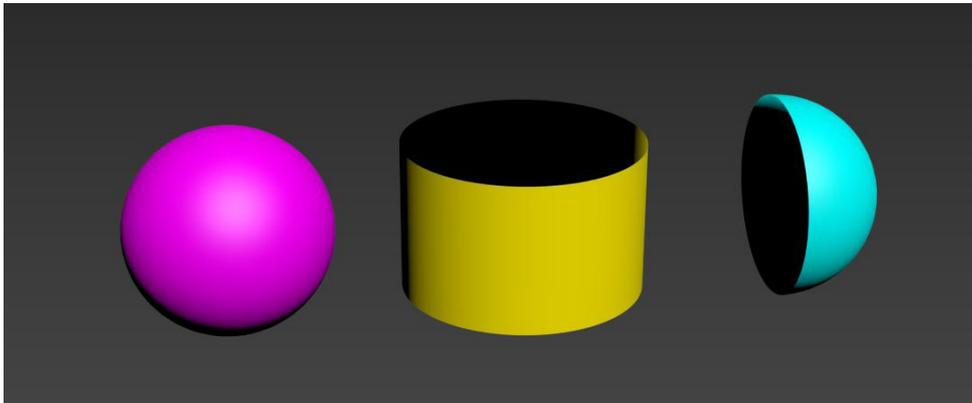


Figura 63. Objetos exportados a Unity

Además, necesitaremos instalar el motor de renderizado V-Ray, para poder generar pares estereoscópicos de los renders que vamos a analizar. Para ello tendremos que introducir el modelo 3D correspondiente, adaptar la escena y configurar V-Ray.

---

## 6- HARDWARE

Un punto importante de este trabajo de experimentación, es explorar el mayor número de medios que permitan recopilar imágenes estereoscópicas para su posterior aplicación en realidad virtual.

Por un lado, haremos un estudio de mercado con los dispositivos que nos permitan la obtención de fotografía tridimensional, desde las cámaras a los soportes para las mismas.

Además, se mostrará el desarrollo de un soporte estereoscópico compatible con smartphone. Este dispositivo ha sido diseñado específicamente para cumplir tal propósito, dado que no encontramos en el mercado opciones que permitan desempeñar la misma función.

### 6.1- Análisis de hardware en el mercado

Vamos a ver en este apartado los tipos de cámaras que permiten la obtención de imágenes estereoscópicas. Las cámaras más comunes se han podido utilizar de primera mano para su posterior análisis, mientras que las más novedosas se analizan a partir de material descargado a través de Internet.

Vamos a comenzar por el análisis de las cámaras por orden de disponibilidad y seguiremos con los soportes.

#### 6.1.1- Smartphone

Gracias a la gran difusión de los smartphones, todo el mundo sale a la calle con una cámara en el bolsillo. Ya casi ni lo valoramos, pero muchas veces no somos conscientes de las posibilidades que ofrece. Incluso en temas de realidad virtual, el uso del smartphone se convierte en una de las opciones más interesantes, ya que gracias a apps que se apoyan en el giroscopio del móvil, podemos construir una panorámica 360° de una forma muy sencilla. Eso sí, para construir 2 panorámicas estereoscópicas es necesario utilizar un elemento de apoyo como un trípode y otro mecanismo que permita preservar la disparidad de ambas imágenes.

Por si no fuera suficiente, también se convierten en la opción más inmediata de grabar vídeo 3D, ya que, al necesitarse dos dispositivos, se convierten en la opción más abundante. Para ello tenemos que disponer los móviles de tal forma que sus cámaras se encuentren sobre el mismo eje y apuntando en la misma dirección. Para conseguir, que se sincronicen ambas grabaciones, podemos grabar desde una app que enlaza los dispositivos. O bien, aunque menos aconsejable, sincronizar ambas grabaciones manualmente en algún software de edición de vídeo.

### 6.1.2- Cámara réflex

Es una buena opción, ya que, aunque no es una herramienta tan extendida como el smartphone, permite configurar multitud de parámetros para obtener la calidad de imagen deseada.

La cámara tiene multitud de modos, en los que algunas opciones se configuran de forma automática y otras de forma manual. En la obtención de fotografías 360°, que resultarán de la unión de muchas fotos, con el objetivo de mejorar esa costura, se recomienda utilizar la cámara en modo manual configurando las opciones de la cámara de tal forma que todos los objetos tengan una correcta exposición y enfoque.

Las opciones principales a tener en cuenta a la hora de sacar una fotografía son:

- **Exposición:** parámetro de la cámara que nos da una idea de la cantidad de luz que tiene una imagen. Toma valores de -3 a 3, estando los valores negativos asociados a fotografías subexpuestas y valores positivos a fotografías sobrepuestas. Aunque esto no siempre es cierto y es un valor subjetivo, relacionado con el software de la cámara. Por lo que, aunque nos da una ligera idea en cuanto a la exposición de la imagen, la opinión del fotógrafo siempre ha de prevalecer sobre dicho valor. La exposición no se modifica directamente, pues está directamente influenciada por 3 parámetros que regulan la cantidad de luz que recogerá la cámara. Estos parámetros son el ISO, la apertura y la velocidad del obturador.

- **Velocidad del ISO:** este parámetro que suele tomar los valores de 100, 200, 400, 800, 1600... configura la sensibilidad a la luz del sensor. De tal forma que cuanto mayor sea este número mayor luz recogerá el sensor. No se recomienda utilizar valores altos, ya que puede aparecer ruido en la imagen.

· **Velocidad del obturador:** regula el tiempo que permanece abierto el obturador, permitiendo la entrada de luz a la cámara. Toma valores fraccionarios, salvo para grandes exposiciones mayores de 1 segundo. De esta manera los siguientes valores irán ordenados de mayor a menor tiempo de exposición: 2", 0"5, 1/8, 1/250.

· **Apertura del diafragma:** el diafragma está formado por unas pestañas que forman un agujero, este agujero regula la cantidad de luz que entra al objetivo. La apertura de dicho diafragma, se denota mediante la expresión f/ acompañada de un número. Algunos valores comunes son: f/3.5, f/4, f/22.

De esta forma los números pequeños, irán asociados a una apertura de diafragma alta, que permite la entrada de más luz al sensor. Aunque al abrir mucho el diafragma se pierde en profundidad de campo. Por lo que se recomienda utilizar valores intermedios.

· **Enfoque:** está relacionado con la distancia a la que los objetos aparecen nítidos, y se ve influenciado por los valores de apertura de la cámara. Para realizar una fotografía esférica, se recomienda utilizar la distancia hiperfocal de la cámara, que es la distancia de enfoque a la cual conseguimos enfocar un mayor campo de visión y se calcula mediante la siguiente formula:

$$H = (F \cdot F) / (f \cdot d)$$

Donde **F** es la distancia focal del objetivo de la cámara, **f** la apertura del diafragma y **d** el diámetro del círculo de confusión. Sin entrar en detalles, **F** depende del objetivo, **f** es un parámetro que seleccionamos en cada instantánea y **d** depende de la cámara. Una vez aplicada la fórmula, obtendremos el parámetro **H**, que es la distancia hiperfocal en milímetros a la cual tenemos que enfocar para obtener el mayor campo de visión.

Además, hay calculadoras online de la distancia hiperfocal, en las cuales esta tarea se vuelve mucho más fácil y obtenemos información de forma más gráfica sobre la fotografía que vamos a realizar.

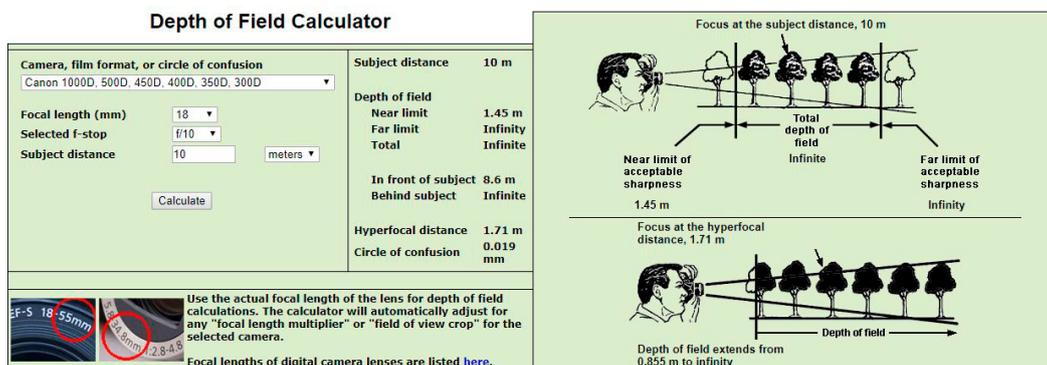


Figura 64. Calculadora de distancia hiperfocal en <http://www.dofmaster.com/dofjs.html>

En ella, introducimos la distancia focal del objetivo, la apertura del diafragma y nuestro modelo de cámara, e inmediatamente obtenemos el valor de la distancia hiperfocal. Además, si introducimos la distancia a la que se encuentra el objeto principal de la escena, nos hace una comparativa entre la fotografía enfocando directamente a dicho objeto y la fotografía hiperfocal. Siempre, elegir la distancia hiperfocal, supone enfocar más campo de visión por delante y por detrás del objeto principal.

- **Balance de blancos:** controla los niveles de los colores de la imagen. Es una función que se recomienda tener en automático, aunque se puede ajustar manualmente en función de la iluminación de la foto.

- **Objetivo:** elegir un objetivo apropiado es fundamental para obtener buenos resultados en una fotografía panorámica. Para este tipo de fotografía, lo que nos interesa conocer es el campo de visión que tendrá nuestra cámara, solo así podremos conocer el número de fotografías necesarias para cubrir los 360° de la escena.

El ángulo de visión que cubre una cámara viene definido por dos parámetros, el HFOV y el VFOV (horizontal field of view y vertical field of view, respectivamente). Estos 2 parámetros dependen de la distancia focal del objetivo y del tamaño del sensor de la cámara. Cuanto mayor sea el tamaño del sensor, mayor campo de visión tendrá la cámara y cuanto menor sea la distancia focal del objetivo, mayor será el campo de visión proporcionado por el objetivo. Las distancias focales de los objetivos oscilan entre los 8 y los 600 mm, siendo los de 8 mm de gran angular con más de 100° de HFOV, pasando por los de 18 mm con unos 45° de HFOV hasta llegar a los de 600 mm con 2° de HFOV.

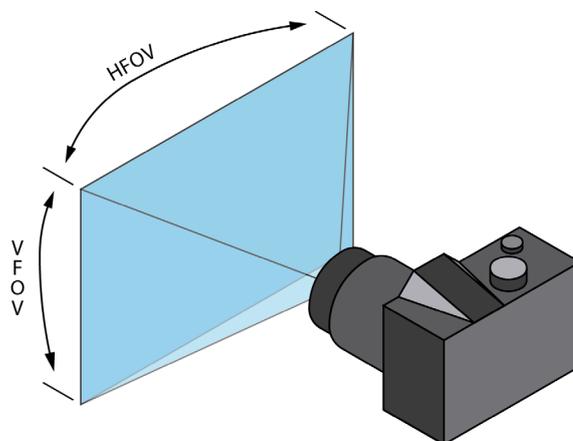


Figura 65. Esquema del campo de visión de una cámara

Como es lógico, cuanto menor sea el campo de visión de nuestra cámara, mayor número de fotos tendremos que sacar para obtener una panorámica 360°. Por ello es crucial conocer los valores del HFOV y VFOV, que determinarán cuántas fotos tendremos que utilizar. La forma más sencilla y precisa de conocer este valor es mediante una calculadora online del campo de visión.

1. Enter data						
Focal Length	<input type="text" value="18"/>	mm	Film/CCD Sensor length	<input type="text" value="22.3"/>	mm	
Size utilization	<input type="text" value="100"/>	%	Film/CCD Sensor width	<input type="text" value="14.9"/>	mm	
			Min. image circle	<input type="text" value="26.82"/>	mm	
<input type="button" value="Film, CCD Sensor Sizes (click to auto fill)"/>						
2 Calculate Field of View (FOV)						
<input type="button" value="Calculate now !"/>	For fisheye lens			For normal lens		
	VFOV (portrait)	<input type="text" value="72.17"/>	deg	VFOV (portrait)	<input type="text" value="63.55"/>	deg
	HFOV (portrait)	<input type="text" value="47.77"/>	deg	HFOV (portrait)	<input type="text" value="44.97"/>	deg
	FOV min. image circle	<input type="text" value="87.48"/>	deg	FOV min. image circle	<input type="text" value="73.37"/>	deg
3 Calculate Panorama						
<input type="button" value="Calculate now !"/>	Overlap in % of FOV	<input type="text" value="25"/>	%	Overlap in % of FOV	<input type="text" value="25"/>	%
	Desired panorama FOV	<input type="text" value="360"/>	deg	Desired panorama FOV	<input type="text" value="360"/>	deg
	No. of images landscape	<input type="text" value="6.7"/>		No. of images landscape	<input type="text" value="7.6"/>	
	No. of images portrait	<input type="text" value="10.1"/>		No. of images portrait	<input type="text" value="10.7"/>	

Figura 66. Calculadora online del campo de visión en:  
<http://www.nodalninja.com/products/field-of-view-and-panorama-production-calculator/14>

Únicamente es necesario conocer las dimensiones del sensor de nuestra cámara, la distancia focal del objetivo y si nuestro objetivo es normal o de tipo ojo de pez. Es importante darse cuenta de que los valores se nos proporcionan para la cámara en posición de retrato. Además, nos calcula el número de fotos de forma automática, en función del porcentaje de solapamiento entre imágenes deseado.

Por estas razones, lo más eficiente a la hora de realizar una panorámica esférica es utilizar un objetivo con la menor distancia focal posible. Ya que la toma de fotos nos llevará menos tiempo y el cosido entre imágenes tendrá menos aberraciones.

### 6.1.3- Cámaras 360°

Las cámaras 360° son un gadget tecnológico bastante reciente, hace unos años nadie había oído hablar de ellas y hoy en día ya hay bastantes modelos en el mercado. Se caracterizan por ser capaces de capturar todo el campo de visión gracias a varias cámaras que posteriormente unirán esas capturas. Algunos modelos pueden capturar foto y vídeo. Vamos a ver cuáles son los tipos principales de cámaras y como se pueden utilizar en la generación de aplicaciones de realidad virtual:

· **Cámara con objetivos múltiples:** consiste en una serie de cámaras (más de 2) que enfocan cada una a un punto de la escena, enfocando entre todas los 360°. Posteriormente el conjunto de imágenes se solapa con ayuda de un software dando como resultado una panorámica esférica. No tiene demasiada aplicación en realidad virtual ya que, si queremos sacar dos fotografías, una para ojo, la disparidad entre ambas solo se conservará en la dirección perpendicular al eje que une las cámaras, como se aprecia en la siguiente ilustración.

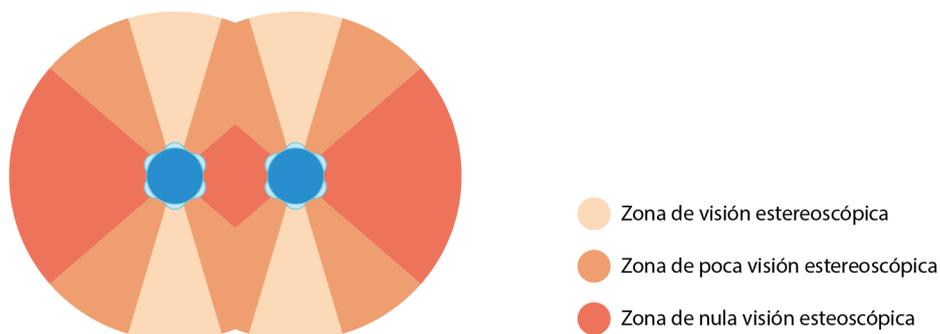


Figura 67. Zonas de visión estereoscópica utilizando dos cámaras 360°

Como se puede observar a medida que nos salimos de esta zona, la disparidad de las imágenes se empieza a reducir, reduciéndose progresivamente los 65mm de disparidad entre ambas imágenes.

Hay multitud de cámaras de este tipo y se diferencian por la cantidad de cámaras que utilizan para completar toda la escena. A mayor número de cámaras, mayor resolución que tendrá la imagen resultante. Ya que el total de la imagen se reparte entre un mayor número de sensores, aportando cada uno mayor definición.



Figura 68. Cámara Panono, integrada por 36 cámaras

· **Cámara con 2 objetivos:** crean el panorama a través de 2 objetivos situados de forma opuesta. Cada objetivo realiza una captura 180° de la escena, que posteriormente se solaparán. Esta tipo de cámaras no ofrece tan buena resolución como las descritas anteriormente, ya que toda la escena se reparte entre dos sensores, por lo cual sus resultados no son demasiado buenos.

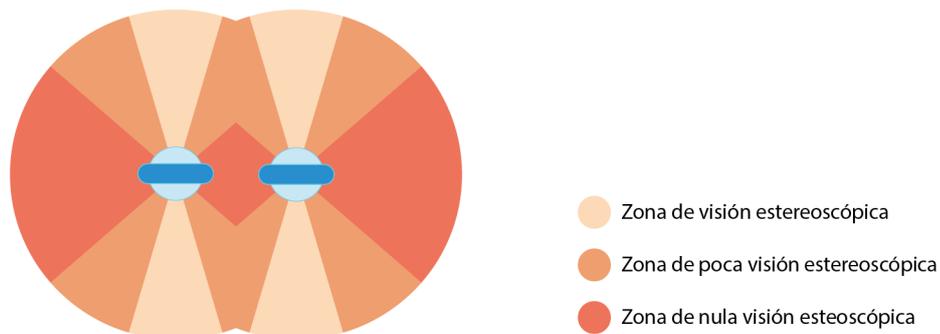


Figura 69. Zonas de visión estereoscópica utilizando dos cámaras 360°

Al igual que las anteriores, su aplicación en proyectos de realidad virtual no es demasiado buena. Ambas sufrirían el mismo problema de no conservar la disparidad de las imágenes en toda la escena. Mientras que este tipo de cámaras ofrecen menos resolución de imagen, presentan una ventaja respecto de las de múltiples objetivos. Y es que, al utilizar dos cámaras simultáneamente es inevitable que una enfoque a la otra, por lo que, al ser más compactas, se pierde menos escena.



Figura 70. Cámara Ricoh Theta

· **Cámara modular:** es una cámara de un solo objetivo de gran angular que enfoca un campo de visión de  $180^\circ$ . Se pueden agrupar de varias maneras, obteniendo diferentes resultados. Si las colocamos de forma opuesta obtendríamos un panorama esférico. También podemos agruparlas de forma paralela, separándolas la distancia deseada, 65 mm si queremos emular la visión humana. Obteniendo una escena un tanto diferente al de las anteriores. Esta escena sería semiesférica, por lo que la otra mitad de la escena no tendría imagen. Aunque presenta el mismo problema en cuanto a la conservación de la disparidad. Lo podemos ver claramente en el siguiente esquema:

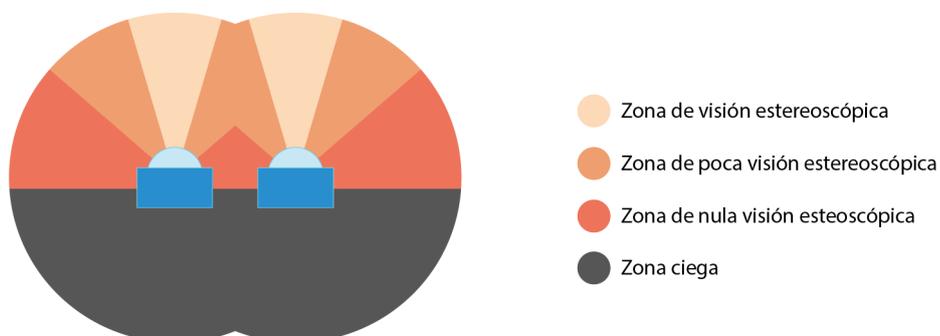


Figura 71. Esquema de visión de una pareja de cámaras modulares

Es una aplicación interesante, ya que permite representar con menos recursos una escena en la que solo nos interese representar una zona determinada, ofreciendo un campo de visión más que suficiente como para que suponga una experiencia inmersiva para el usuario.

Por último, añadir, que con este tipo de cámaras también podemos representar escenas 360° estereoscópicas mediante el uso de 4 cámaras. Con dos parejas de cámaras opuestas situadas de forma paralela.



Figura 72. Cámaras Kodak PixPro SP360, en una combinación para capturas estereoscópicas

#### 6.1.4- Cámaras estereoscópicas

Se encuentran en desuso hoy día ya que se sustituyen fácilmente por 2 cámaras modulares situadas de forma paralela, las cuales, como ya hemos visto, son más versátiles.

La cámara está compuesta por dos objetivos que envían la información visual a un sistema de sensores dual. De esta forma conseguimos recopilar una imagen para cada ojo. Hay diferentes modelos, desde cámaras en las que se puede ajustar la separación de las lentes, hasta otras con objetivos de gran angular.



Figura 73. Cámara SONY PMW-TD300 del año 2011

### 6.1.5- Cámaras estereoscópicas 360°

Es un tipo de cámaras que permite la visualización de entornos 360° con una visión estereoscópica mucho mejor repartida que la que ofrecen las cámaras 360°, es decir, ofrece su visionado de forma estereoscópica en cualquier ángulo de observación.

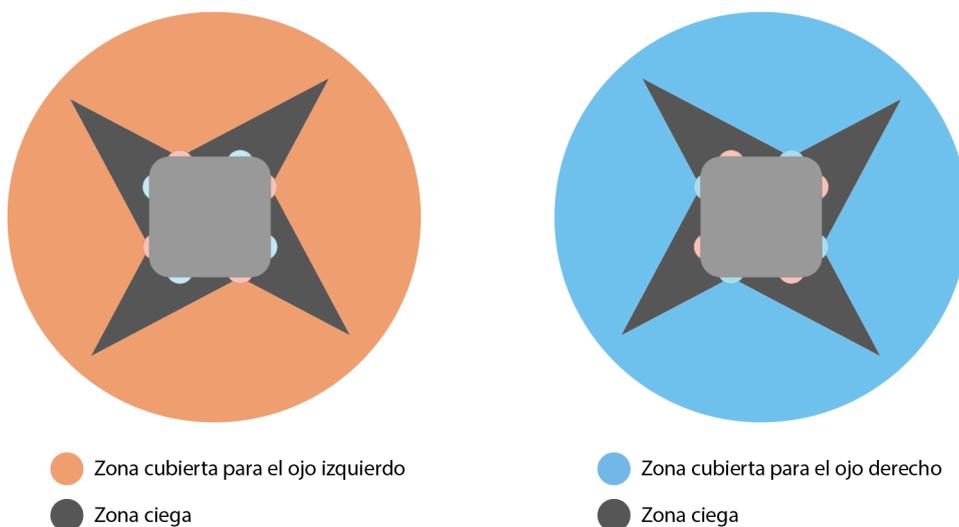


Figura 74. Esquema funcionamiento cámara 3D 360°

Está compuesta por una serie de cámaras dispuestas a lo largo de todo el dispositivo, que cubren todas las direcciones. Están agrupadas en parejas, y cada pareja de imágenes enfocará a una dirección, por lo que la disparidad de las imágenes se mantiene a lo largo de la escena. Aunque en el fondo esto no es totalmente cierto, ya que las direcciones a las que no apuntan directamente las cámaras, irán perdiendo progresivamente su disparidad, como observamos en el siguiente esquema:



Figura 75. Esquema zonas de visión estereoscópica en una cámara 3D 360°

Podemos deducir también del esquema anterior que, cuantas más parejas de cámaras se utilicen, mayor será el campo de visión estereoscópica pura. Por el momento, la única cámara comercializada que graba tridimensionalmente en los 360°, es la Vuze camera. Cuyo esquema de actividad corresponde con el expuesto anteriormente.



Figura 76. Vuze Virtual Reality Camera

Habrá que esperar para poder disfrutar de cámaras con mayor calidad estereoscópica como la Project Beyond de Samsung, la cual aún no se comercializa. Con 17 cámaras, 8 pares que cubren el plano horizontal y 1 cámara que apunta hacia arriba, debería de ofrecer una distribución estereoscópica el doble de buena que la Vuze camera.



Figura 77. Cámara Project Beyond de Samsung

Facebook, pretende revolucionar el mercado, presentando un prototipo, la Facebook Surround 360, que no se comercializa, sino que se pretende que sea el usuario quien la construya y la modifique en función de sus necesidades. Los planos de montaje se encuentran accesibles a todo el público en Internet, en la web de GitHub. Además, el software utilizado es de código libre, lo que abre las puertas a la comunidad de makers a probar otras arquitecturas de montaje. Sin lugar a dudas es una iniciativa que promueve la evolución de este tipo de cámaras, que por el momento se encuentran en fase de desarrollo.



Figura 78. Configuración de la Facebook Surround 360

### 6.1.6- Trípode con rótula medidora de ángulos

La función de este trípode es la de permitirnos rotar la cámara respecto del eje vertical, indicándonos la magnitud de la rotación.

Nos permite capturar todo el campo de visión horizontal. Además, si conocemos el valor del HFOV de nuestra cámara, sabremos, en función del porcentaje de solapamiento, el número de fotos que tenemos que tomar para capturar todo el campo de visión. Aunque, es recomendable evitar números de fotos que no sean múltiplos de las divisiones de nuestra rótula, ya que harán menos precisa la tarea. Es mejor realizar más fotos de la cuenta, siempre que escojamos una cantidad múltiple del número de divisiones, lo cual nos ahorrará tiempo y mejorará la precisión de las tomas.

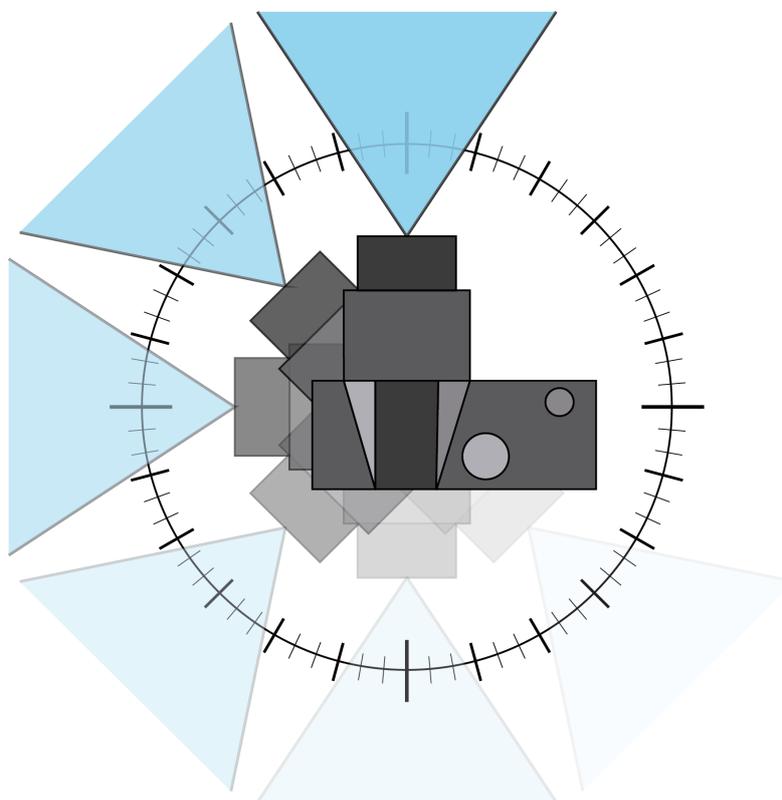


Figura 79. Esquema del funcionamiento de un trípode con rótula medidora de ángulos

### 6.1.7- Rótula panorámica

Como se ha comentado, el trípode con la rótula medidora de ángulos, nos permite capturar toda la escena en el plano horizontal, pero para capturar los 360° de visión, también se ha de poder rotar la cámara en el eje transversal. La rótula panorámica, permite a la cámara esa movilidad extra que no proporciona el trípode.

El funcionamiento es el siguiente, la cámara se acopla en posición de retrato a la rótula panorámica y esta a su vez a la rótula del trípode. Vamos a ver un esquema del conjunto montado en el que se señalan los movimientos permitidos:

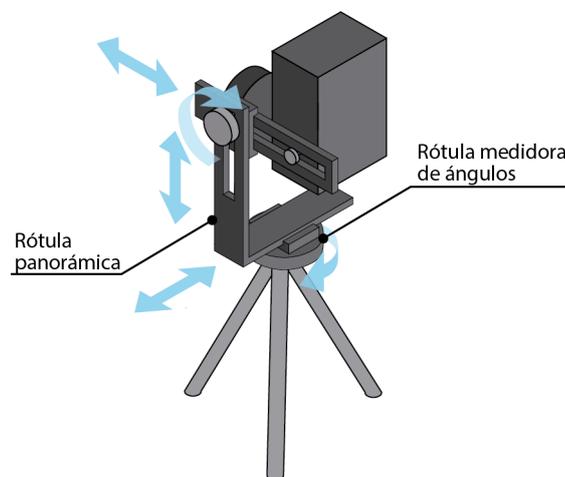


Figura 80. Esquema de los movimientos permitidos por una rótula panorámica

Este tipo de rótulas han sido ideadas para obtener panorámicas esféricas, no estereoscópicas. El procedimiento consiste en ir fotografiando el entorno, girando la cámara alrededor del eje vertical, de tal forma que la separación angular entre fotografías sea la misma. Este ángulo de separación, ha de ser calculado previamente en función de la configuración de la cámara y el ángulo de solapamiento. Cuando hayamos completado el giro, repetimos el procedimiento girando la cámara en torno al eje transversal, lo cual permitirá fotografiar las zonas cercanas al zenit y nadir. Posteriormente, utilizando un software de cosido de imágenes, podemos crear una fotografía panorámica esférica.

Pero si nos damos cuenta, podemos mover la cámara transversalmente, desplazándola de su centro. Por lo cual, siguiendo los principios de la visión estereoscópica humana, si separamos la cámara 32,5 mm del centro hacia un lado y hacia otro, obtendremos 2 imágenes con una disparidad de 65 mm. Si además seguimos el procedimiento anterior podremos unir las fotografías tomadas con la cámara a un lado del eje y al otro, creando un par de panorámicas estereoscópicas.

Antes de empezar a sacar fotos con este instrumento, es necesario colocar la cámara en el lugar correcto para que ambas fotos del par estereoscópico guarden la disparidad entre sí. En primer lugar, ajustamos la altura de la cámara en una posición lo suficientemente elevada como para que la cámara no choque con la rótula en las demás posiciones. Se recomienda elegir el punto más alto, ya que así nos aseguramos de que la cámara puede realizar todos los movimientos, además, al hacer tope, no perderemos el eje de rotación de la cámara.

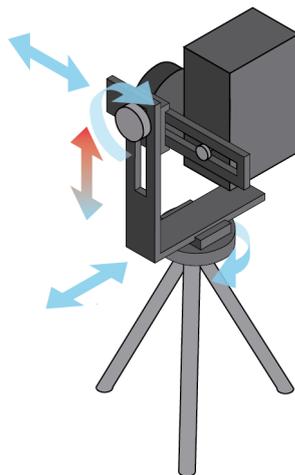


Figura 81. Esquema del ajuste de la cámara en el eje vertical

Posteriormente, tendremos que desplazar la cámara transversalmente hasta situarla en el punto correspondiente a un ojo. Para ello lo primero que tenemos que hacer será alinear el centro de la lente con el eje de rotación del trípode. A continuación, dado que la distancia promedio entre ojos es de 65 mm, desplazaremos la cámara 32,5 mm hacia cada lado para tomar las imágenes correspondientes a cada ojo.

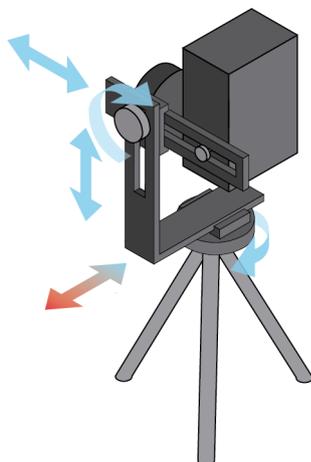


Figura 82. Esquema del ajuste en el eje transversal

Por último, ajustamos la posición de la cámara en el eje longitudinal, hasta encontrar el punto de no paralaje. El punto de no paralaje, es el punto en el cual objetos que se encuentran a distancias distintas de la cámara no se mueven unos respecto de otros al rotar la cámara. Para encontrarlo, podemos usar como referencia una pegatina pegada en una ventana, observando como se mueve respecto del fondo al rotar la cámara.

Hay una regla muy sencilla para encontrarlo, si el objeto cercano se mueve hacia el lado contrario al giro de la cámara, la cámara está adelantada. Cuando la cámara está por detrás del punto de no paralaje o punto nodal, el objeto se mueve en la misma dirección que la cámara. Jugando con esta regla finalmente encontramos el punto de no paralaje.

Cabe destacar que debido a que las posiciones que tendrá la cámara son simétricas del centro, el punto de no paralaje será el mismo tanto para la fotografía del ojo derecho como para la del izquierdo.

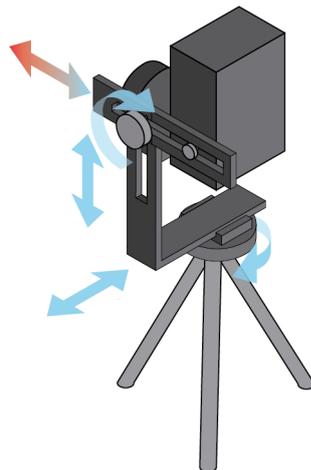


Figura 83. Esquema del ajuste en el eje longitudinal

Una vez tenemos preparada la cámara en la posición correspondiente a un ojo, rotamos la cámara en torno al eje vertical, hasta capturar los 360° de la escena. Es muy importante que cuando terminemos este paso, movamos la cámara a la posición correspondiente al otro ojo y volvamos a tomar las fotos. Es muy importante hacerlo así, ya que es más fácil ajustar la posición de la cámara que el ángulo de rotación en torno al eje transversal.

Además, se recomienda realizar la primera rotación desde la segunda foto, para así finalizar en la primera, luego mover la cámara hasta la posición del otro ojo y empezar a tomar fotos desde esa posición. De esa forma las dos primeras fotos de cada giro, que son las que el programa de cosido toma como referencia, apuntan en el mismo sentido.

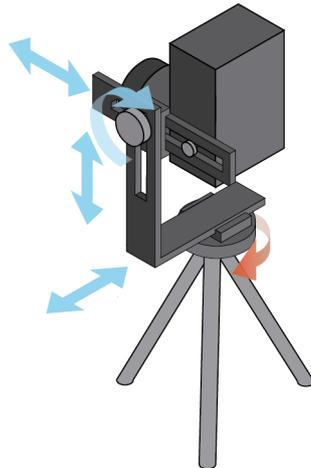


Figura 84. Esquema del giro en el eje vertical

Cuando hayamos terminado de capturar el plano horizontal, podemos pasar a tomar fotos de la cúpula de la escena. Sobre todo, si estamos en exteriores, lo mejor es continuar por la parte superior ya que puede haber nubes que es interesante fotografiar lo más rápido posible para evitar que se muevan respecto de las anteriores fotografías.

El modus operandi sería análogo al descrito anteriormente, pero con la cámara apuntando a una posición más elevada. Cuando hayamos terminado de fotografiar la parte superior, comenzamos con la inferior hasta finalizar en el nadir, punto en el cual se encuentra el trípode.



Figura 85. Esquema del giro en el eje transversal

Si analizamos este procedimiento, observamos que es similar al modo de funcionamiento de la cámara estereoscópica de Vuze.

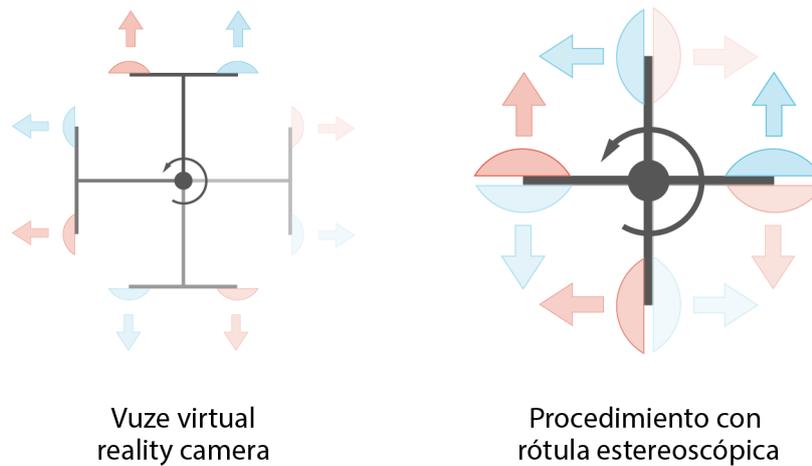


Figura 86. Esquema comparativo entre métodos

Como se puede apreciar, en ambos casos las parejas de cámaras izquierda y derecha, se exponen en torno a un eje, la diferencia es que, en el procedimiento con rótula estereoscópica, las cámaras se encuentran en su punto de no paralaje, por lo que el cosido de la imagen será de mayor calidad.

Esto se debe a que las cámaras de la Vuze, actúan todas de forma simultánea, mientras que en el método que utiliza la réflex, la cámara va ocupando las distintas posiciones de forma progresiva. La cámara Vuze nunca podría seguir el otro esquema porque las cámaras se estorbarían entre sí.

En la cámara de Vuze, al no encontrarse sus cámaras dispuestas respecto de su punto nodal, la zona de solapamiento entre imágenes puede ser conflictiva, ya que los objetos cercanos pueden aparecer movidos respecto del fondo en imágenes contiguas.

Por otro lado, la rapidez con la que la cámara Vuze obtiene las imágenes, puede compensar estos defectos de cosido. La zona de cosido solo puede optimizarse para una distancia concreta de enfoque, los objetos situados por delante y por detrás de esta zona no casarán bien con su imagen adyacente.

Los entornos orgánicos, por su condición, disimulan bastante bien las zonas de unión entre imágenes, por lo cual, este tipo de entornos es el más propicio para capturar con esta cámara. Por el contrario, los entornos arquitectónicos, suelen ser más propensos a presentar fallos de esta naturaleza.

## 6.2- Desarrollo de un soporte estereoscópico para smartphone

Tras analizar el funcionamiento de los dispositivos de obtención de imágenes estereoscópicas, nos damos cuenta de que siguen un patrón común, el uso de al menos una pareja de lentes separadas entre sí la distancia deseada para dotar a las parejas de imágenes de disparidad.

Además, un smartphone, dotado de la aplicación Google Street View, se convierte en una herramienta simple y rápida para la obtención de panoramas esféricos. Esta aplicación guarda bastante semejanza con el proceso seguido para obtener una panorámica esférica con la cámara réflex, la diferencia es que el cosido de imágenes se realiza de forma automática en la aplicación y las fotos que tenemos que realizar nos salen indicadas en la pantalla del móvil. Por lo cual, siempre vamos a saber qué fotos hemos realizado y cuales nos faltan por hacer.

La aplicación se apoya en los sensores de posicionamiento de nuestro terminal (acelerómetro y giroscopio) para proyectar sobre una esfera las diferentes imágenes que se necesitan. La esfera que utiliza la app, está dividida en una serie de puntos, que serán las distintas zonas a las que tendremos que apuntar para cerrar la esfera. Podemos ver en la siguiente imagen una distribución de los puntos para el teléfono en posición horizontal:

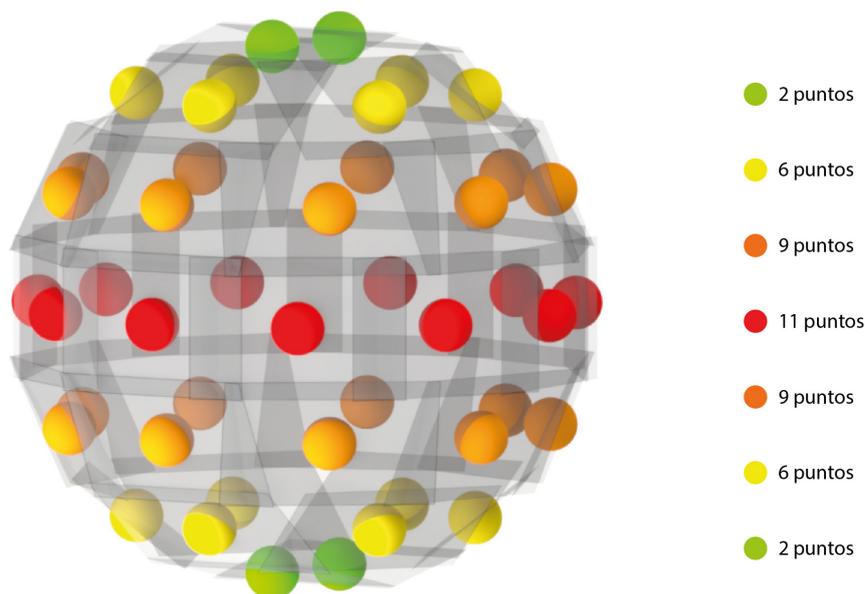


Figura 87. División de la esfera en la app Google Street View

Los puntos se distribuyen por la esfera a lo largo de planos horizontales, necesitaremos sacar más fotos cuanto mayor sea la circunferencia sobre la que se dispongan. De esta manera sacaremos 11 fotos en el ecuador, 30 repartidas entre los 4 paralelos, 2 en el zenit y 2 en el nadir. Esto hace un total de 45 imágenes, para las cuales necesitaremos colocar el móvil en 7 posiciones diferentes rotándolo sobre su eje transversal.

Conociendo la distribución de puntos de la esfera, y la separación que queremos establecer entre las 2 cámaras, podemos crear un soporte para 2 smartphones que se pueda acoplar a un trípode, para mejorar la calidad de los panoramas.

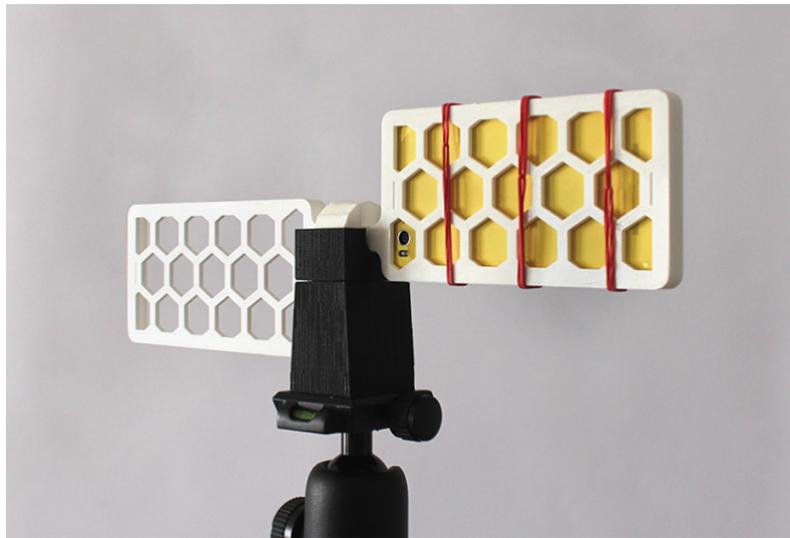


Figura 88. Soporte estereoscópico para smartphone montado sobre un trípode

El soporte se ha fabricado mediante impresión 3D y se sostiene sobre un apoyo, también impreso en 3D, diseñado para que encaje el soporte, pudiendo rotar sobre él. Sobre este eje de rotación se encuentra dispuesta la cámara del móvil, cuando ocupa las posiciones izquierda y derecha. El apoyo ha sido diseñado para encajar en el trípode del que dispongo y permitir, mediante su geometría, apuntar a todos los puntos de la esfera.

Para ello, y debido a las divisiones de la esfera, he dotado a la superficie de contacto entre soporte y apoyo, de una geometría tetradecagonal, para que cada desplazamiento entre caras contiguas suponga la rotación correspondiente al siguiente nivel de la esfera.

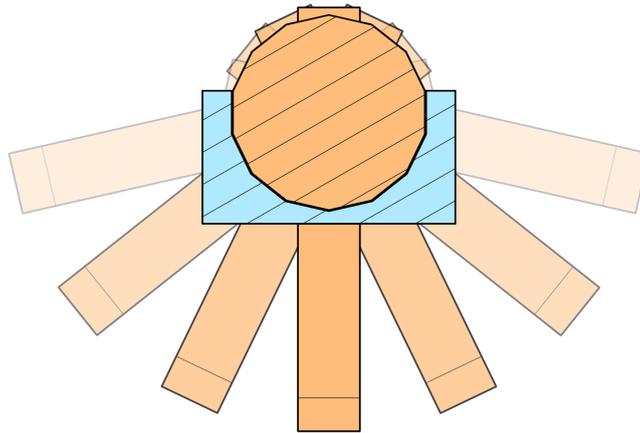


Figura 89. Esquema de rotación entre soporte y apoyo

Utilizando la configuración expuesta en la imagen 76, podemos capturar dos panoramas esféricos estereoscópicos con un solo dispositivo. El procedimiento es similar al seguido con la cámara réflex montada sobre la rótula esférica, salvo que, en este, hasta que no terminemos un panorama, no podremos tomar las imágenes del otro.

También podemos utilizar dos smartphones simultáneamente en este soporte, con ello podremos crear dos panoramas estereoscópicos en la mitad de tiempo. De esta forma nos aseguramos la concordancia entre imágenes, ya que, sobre todo en exteriores, es difícil conseguir que todos los elementos de la imagen permanezcan estáticos en el periodo que dura la toma de los dos panoramas, que se puede alargar hasta los 10 minutos.



Figura 90. Soporte utilizando 2 smartphones simultáneamente

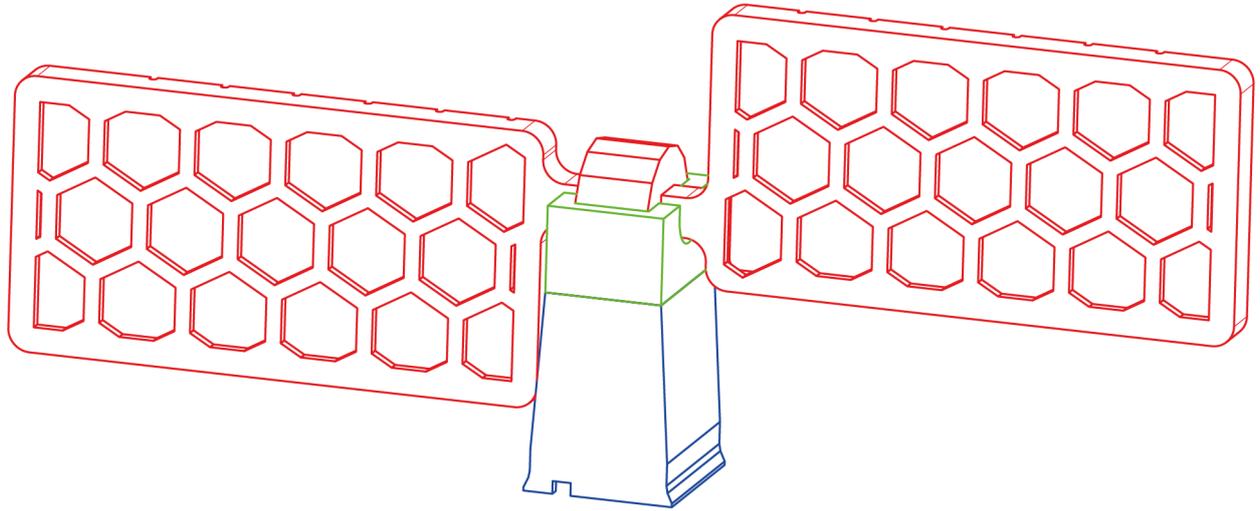
El uso de dos smartphones, también nos abre la posibilidad de grabar vídeo 3D. Para ello, simplemente tendremos que poner los dos móviles a grabar y posteriormente sincronizar ambos vídeos con un software de edición de vídeo.

### **6.2.1- Planos del soporte**

El soporte inicial, fue creado para el modelo de móvil del que dispongo actualmente, el Xiaomi Mi4C. Además se adapta al famoso iPhone 6 calzándolo con un suplemento que puede ser incluso de cartón, ya que este terminal es ligeramente más pequeño.

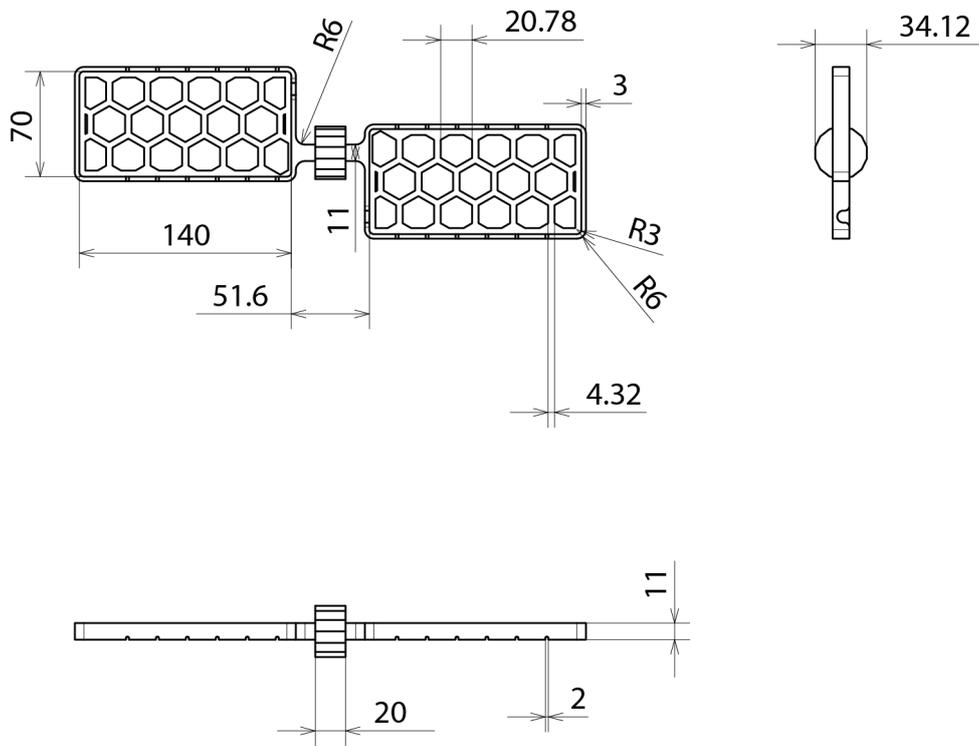
Cuento esto, porque para la aplicación Google Street View, el modelo de Xiaomi no funciona correctamente, supongo que por falta de precisión en alguno de sus sensores. A pesar de intentar calibrarlos, seguían sin dar buenos resultados. Ya que la esfera, que se supone que tiene que permanecer estática, siendo el móvil el que se mueve, no lo hacía.

A continuación, vamos a ver los planos del soporte original, del apoyo y del enganche al trípode.



<b>LEYENDA</b>
Plano 1_Soporte para smartphone
Plano 2_Apoyo
Plano 3_Enganche trípode

 ESCUELA DE INGENIERÍAS INDUSTRIALES	INVESTIGACIÓN SOBRE MÉTODOS PARA LA IMPLEMENTACIÓN DE LA REALIDAD VIRTUAL		
ALUMNO: Darío José Álvarez Gayoso	NOMBRE DEL PLANO: Plano 0_Plano de conjunto		
FECHA: 28/09/2017	FIRMA:	NÚMERO DE PLANO: 0	ESCALA: 1/2
 UNIVERSIDAD DE VALLADOLID Grado en Ingeniería de Diseño Industrial y Desarrollo del Producto	<b>A4</b>	MATERIAL: PLA	<b>81</b>



ESCUELA DE INGENIERÍAS  
INDUSTRIALES

## INVESTIGACIÓN SOBRE MÉTODOS PARA LA IMPLEMENTACIÓN DE LA REALIDAD VIRTUAL

Trabajo de Fin de Grado

ALUMNO:  
Darío José Álvarez Gayoso

NOMBRE DEL PLANO: Plano 1\_Soporte para smartphone

FECHA:  
28/09/2017

FIRMA:

NÚMERO DE PLANO: 1

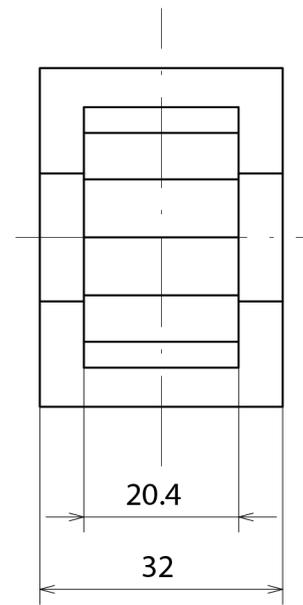
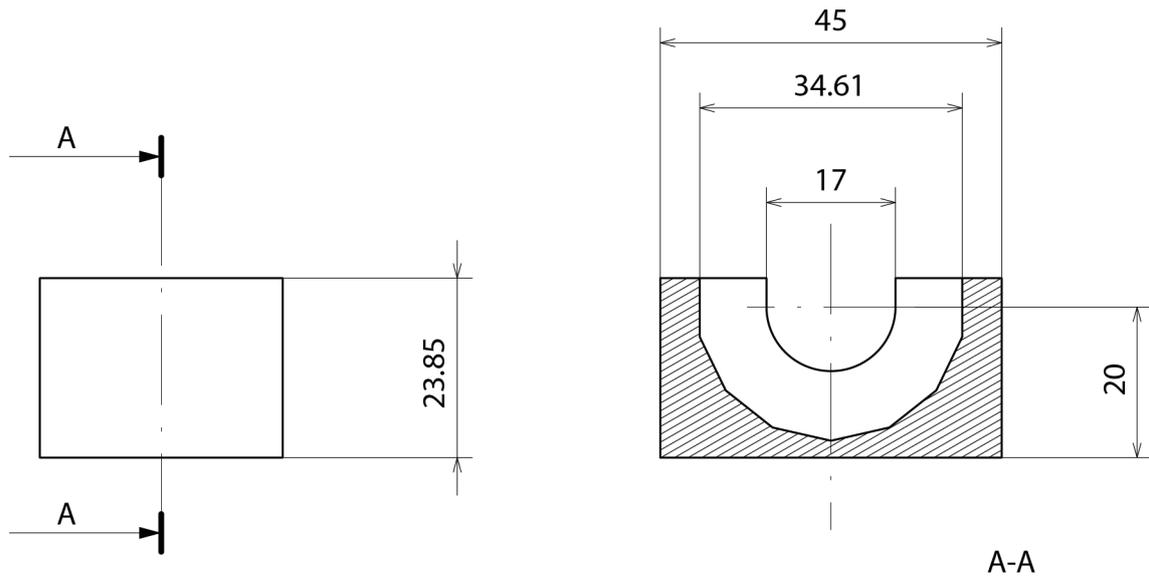
ESCALA: 1/5



UNIVERSIDAD DE VALLADOLID  
Grado en Ingeniería de Diseño  
Industrial y Desarrollo del Producto

**A4**

MATERIAL: PLA



ESCUELA DE INGENIERÍAS INDUSTRIALES

INVESTIGACIÓN SOBRE MÉTODOS PARA LA IMPLEMENTACIÓN DE LA REALIDAD VIRTUAL

Trabajo de Fin de Grado

ALUMNO:  
Darío José Álvarez Gayoso

NOMBRE DEL PLANO: Plano 2\_Apoyo

FECHA:  
28/09/2017

FIRMA:

NÚMERO DE PLANO: 2

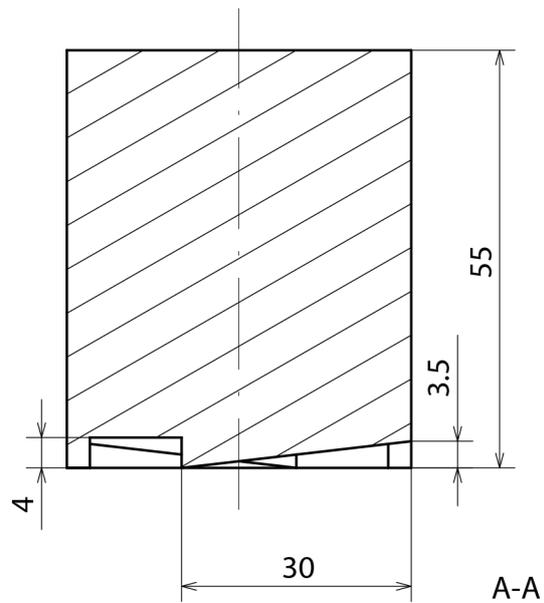
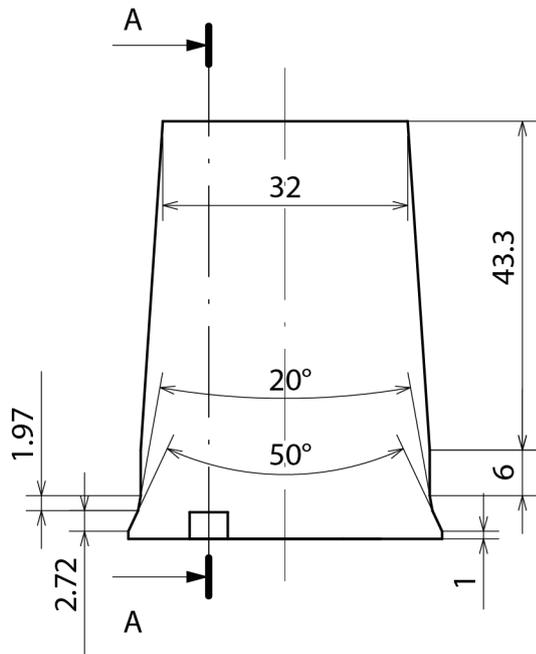
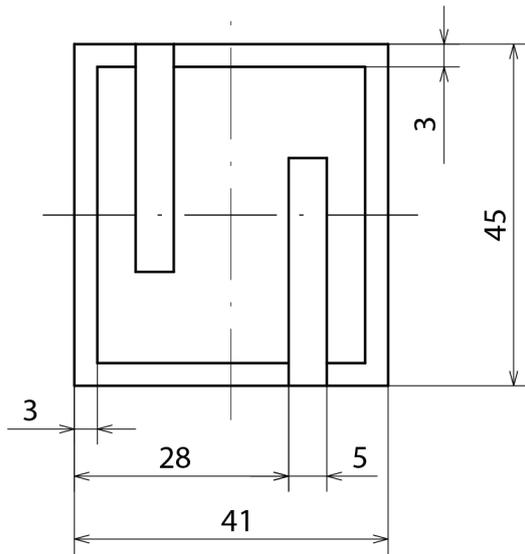
ESCALA: 1/1



UNIVERSIDAD DE VALLADOLID  
Grado en Ingeniería de Diseño Industrial y Desarrollo del Producto

A4

MATERIAL: PLA



ESCUELA DE INGENIERÍAS INDUSTRIALES

## INVESTIGACIÓN SOBRE MÉTODOS PARA LA IMPLEMENTACIÓN DE LA REALIDAD VIRTUAL

Trabajo de Fin de Grado

ALUMNO:  
Darío José Álvarez Gayoso

NOMBRE DEL PLANO: Plano 3\_Enganche trípode

FECHA:  
28/09/2017

FIRMA:

NÚMERO DE PLANO: 3

ESCALA: 1/1



UNIVERSIDAD DE VALLADOLID  
Grado en Ingeniería de Diseño Industrial y Desarrollo del Producto

A4

MATERIAL: PLA

### 6.2.2- Planos del soporte con diseño adaptativo

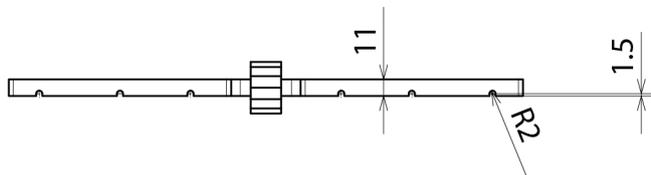
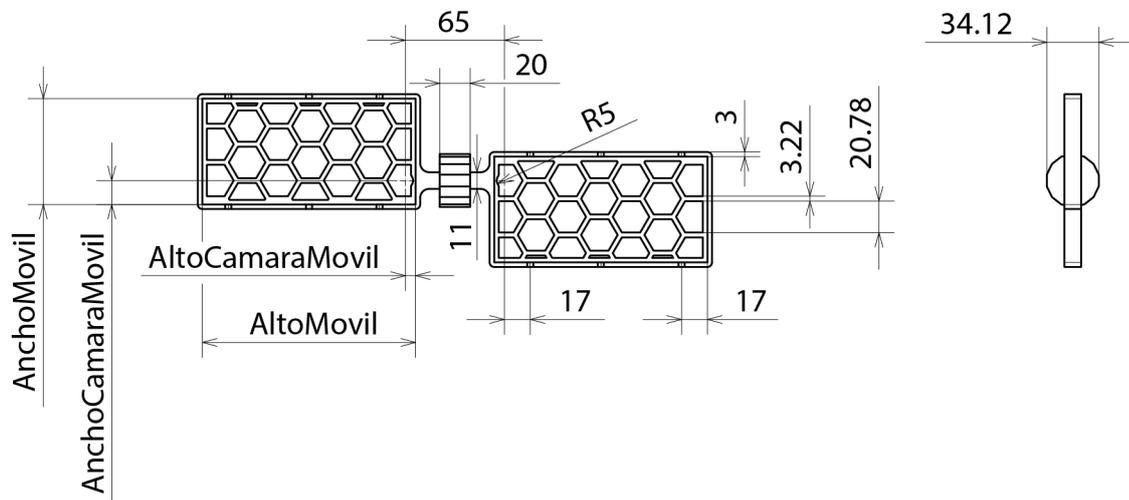
Motivado por la universalidad del soporte, se ha creado un archivo cad del soporte que se pueda modificar rápidamente. El archivo cad está diseñado en Catia, y con solo tomar 4 medidas de nuestro dispositivo, será suficiente para que el programa adapte el soporte a nuestro móvil.

El procedimiento de modificación es muy sencillo, tan solo hay que buscar en el apartado de funciones del archivo las 4 restricciones que aparecen con los nombres: AltoMovil, AnchoMovil, AltoCamaraMovil y AnchoCamraMovil. Al introducir los valores del terminal deseado, el modelo se modificará automáticamente, siendo compatible con 2 terminales iguales.

Este fichero será compatible con móviles cuya medida AltoCamaraMovil no exceda los 13mm, ya que terminales como el Samsung Galaxy S8+ cuya medida alcanza los 24,3mm, harían muy estrecha la rueda tetradecagonal, situada en el eje de rotación del soporte.



Figura 91. Medidas necesarias del soporte para smartphone



ESCUELA DE INGENIERÍAS INDUSTRIALES

## INVESTIGACIÓN SOBRE MÉTODOS PARA LA IMPLEMENTACIÓN DE LA REALIDAD VIRTUAL

Trabajo de Fin de Grado

ALUMNO:  
Darío José Álvarez Gayoso

NOMBRE DEL PLANO: Plano 4\_Soporte adaptativo para smartphone

FECHA:  
28/09/2017

FIRMA:

NÚMERO DE PLANO: 4

ESCALA: 1/5



UNIVERSIDAD DE VALLADOLID  
Grado en Ingeniería de Diseño Industrial y Desarrollo del Producto

A4

MATERIAL: PLA

---

## 7- CREACIÓN DE APLICACIONES DE REALIDAD VIRTUAL EN UNITY

A grandes rasgos, hay dos tipos de fórmulas para recrear estos entornos, una es utilizando imágenes estereoscópicas, y la otra es utilizando directamente un modelo 3D. Estas aplicaciones se podrán exportar a la plataforma de destino que deseamos.

A continuación, vamos a ver las diferentes formas de obtención de estas aplicaciones, en función de las plataformas de destino, HTC Vive o Android.

- **HTC Vive:** comenzaremos por preparar la aplicación para la plataforma de destino. Necesitaremos importar un paquete de assets denominado SteamVR Plugin, que contendrá todos los archivos necesarios para hacer compatible la aplicación con las gafas. Posteriormente, en Build Settings, nos introducimos en Player Settings para configurar los detalles de la aplicación, en este menú marcamos la casilla de Virtual Reality Supported y a continuación escogemos el Virtual Reality SDK de Open VR para poder exportar la aplicación al HDM de HTC.

Una vez preparado el proyecto de Unity, solo tenemos que generar la escena que permita la correcta visualización de las imágenes. Los componentes que integrarán el proyecto, dependerán del tipo de escena a representar.

- **Android:** los pasos a seguir para preparar la aplicación para Android son muy similares a los que se siguen para preparar la app para las HTC. En primer lugar, importaremos un paquete de assets compatible con Android, GoogleVRForUnity. A continuación, en el menú de Player Settings, dentro de Build Settings, marcamos la casilla de Virtual Reality Supported y escogemos el Virtual Reality SDK de Carboard.

Después, los pasos para generar cada una de las aplicaciones son prácticamente los mismos salvo por la configuración de las cámaras. Vamos a ver los pasos a seguir para generar los diferentes tipos de aplicaciones.

## 7.1- Recreación de entornos estáticos en realidad virtual

Vamos a analizar las diferentes formas de representar entornos con Unity, en función de los elementos de partida, bien sean pares de imágenes estereoscópicas o un modelo 3D.

### 7.1.1- Métodos basados en imágenes

La visión estereoscópica consiste en imitar nuestra experiencia cotidiana en cuanto a visión se refiere. Para ello debemos proporcionar a cada ojo su correspondiente imagen de un par estereoscópico. De esta forma conseguiremos reemplazar los estímulos visuales externos por otros equivalentes.

Vamos a ver el proceso de creación de estas aplicaciones en función del tipo de imagen que utilizemos:

#### 7.1.1.1- Fotografía esférica

El procedimiento consiste en hacer llegar una imagen a cada ojo partiendo de dos fotografías esféricas. Para conseguirlo, necesitamos proyectar cada imagen sobre una esfera. A continuación, introducimos dos cámaras estereoscópicas en el centro de las esferas. Ordenamos las esferas en capas, de tal forma que cada cámara visualice una imagen diferente. Por último, seleccionamos el ojo de destino de cada cámara, la cámara que observa la imagen izquierda solo tendrá activo el ojo izquierdo y viceversa. De esta forma ambas cámaras se comportarán como una sola, permitiendo visualizar una imagen diferente con cada ojo, con lo que conseguiremos ver en 3D.

Vamos a ver los componentes de Unity que integran esta aplicación:

- **Objects:** necesitaremos dos esferas sobre las que aplicaremos cada uno de los materiales, la imagen correspondiente a cada material será vista en verdadera magnitud desde el centro de la esfera.
- **Meshers:** la esfera que Unity trae por defecto está compuesta por muy pocos polígonos, lo que se traduce en una textura que se proyecta de forma extraña, sobre todo en los polos de la esfera. Para solucionar este problema, hay que utilizar un Mesher, que modifica la geometría de la esfera, dando como resultado una esfera con suficientes polígonos como para que la esfera no distorsione su textura.

Podemos crear una esfera en 3ds Max, importarla a la biblioteca y aprovechar su mesher, se recomienda que la esfera esté integrada por un número de polígonos en torno a 2000, ya que con muchos polígonos se produce una distorsión como la que observamos en la siguiente imagen:

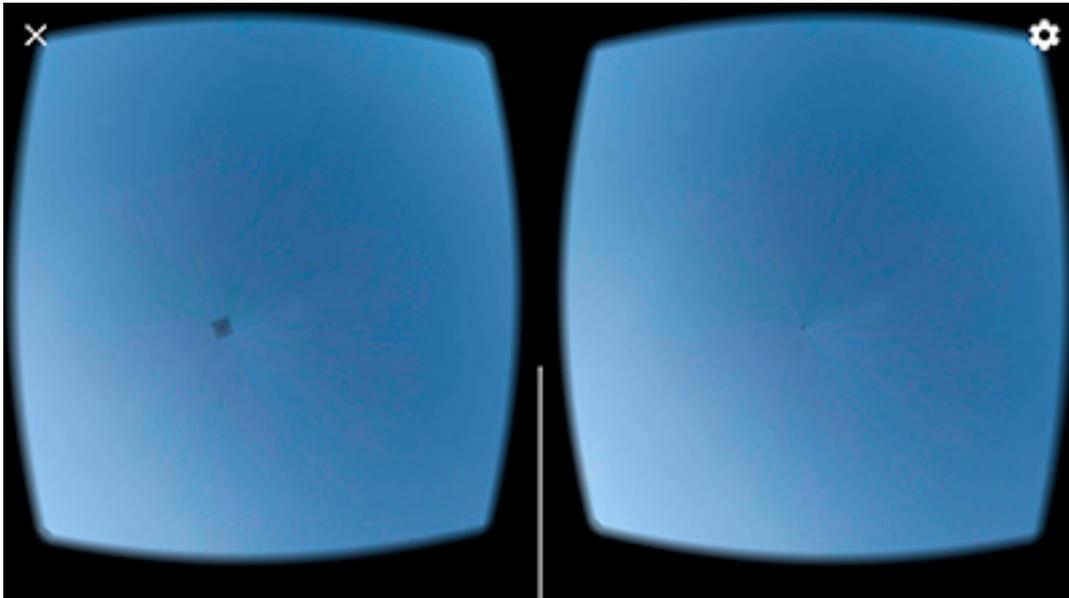


Figura 92. Polo de una esfera compuesta por 32768 polígonos, en contraposición con otra integrada por 2600

- **Materials:** aplicaremos un material diferente a cada esfera, aplicaremos el shader al material, para que invierta las normales, y a continuación, elegimos la textura que queremos proyectar.
- **Shaders:** aplicaremos al material un shader que invierta las normales de la superficie de la esfera para poder ver el material desde el interior.
- **Textures:** el par estereoscópico que vamos a proyectar, tiene que ser importado a la biblioteca de Unity, una vez dentro, podemos modificar el tamaño máximo del archivo en Unity, lo cual determinará la calidad de la aplicación.

```

1 Shader "InsideVisible" {
2   Properties {
3     _MainTex ("Base (RGB)", 2D) = "black" {}
4   }
5
6   SubShader {
7     Tags { "RenderType"="Opaque" }
8     Cull front
9     LOD 100
10
11    Pass {
12      CGPROGRAM
13        #pragma vertex vert
14        #pragma fragment frag
15
16        #include "UnityCG.cginc"
17
18        struct appdata_t {
19          float4 vertex : POSITION;
20          float2 texcoord : TEXCOORD0;
21        };
22
23        struct v2f {
24          float4 vertex : SV_POSITION;
25          half2 texcoord : TEXCOORD0;
26        };
27
28        sampler2D _MainTex;
29        float4 _MainTex_ST;
30
31        v2f vert (appdata_t v)
32        {
33          v2f o;
34          o.vertex = mul(UNITY_MATRIX_MVP, v.vertex);
35          v.texcoord.x = 1 - v.texcoord.x;
36          o.texcoord = TRANSFORM_TEX(v.texcoord, _MainTex);
37          return o;
38        }
39
40        fixed4 frag (v2f i) : SV_Target
41        {
42          fixed4 col = tex2D(_MainTex, i.texcoord);
43          return col;
44        }
45      ENDCG
46    }
47  }
48 }
49

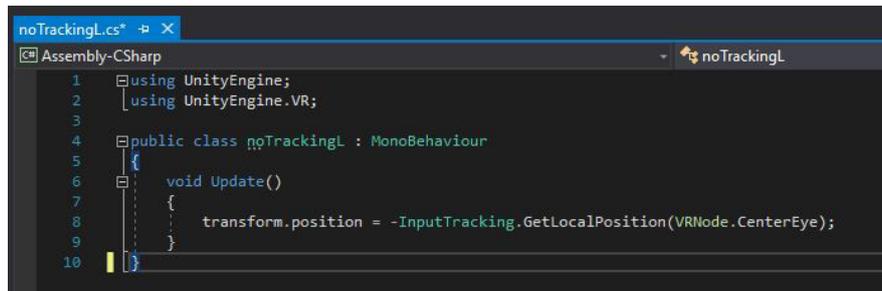
```

Figura 93. Shader utilizado para invertir las normales

· **Cameras HTC Vive:** configurar la pareja de cámaras para las gafas HTC es un proceso ligeramente más complejo. Una vez hayamos importado el paquete de assets SteamVR Plugin, podemos empezar a crear la escena, ya que necesitaremos algunos de sus componentes.

En primer lugar, arrastramos a la escena un CameraRig. A continuación, creamos 2 GameObjects vacíos que más adelante harán la función de cámaras. Añadimos a estos 3 elementos el script de SteamVR\_Camera y expandimos el script dentro de cada elemento. Introducimos los GameObjects, ahora convertidos en cámaras dentro del CameraRig y posteriormente seleccionamos una capa de visualización y el ojo de destino de cada cámara. De esta forma conseguimos que cada cámara tenga un solo ojo activo, que visualizará a una esfera diferente.

Pero en este tipo de gafas surge otro problema, y es el Tracking de la HTC, que permite que nos desplazemos por la escena. Como en este tipo de aplicación nos interesa permanecer estáticos en el centro de la esfera, para observar la imagen en verdadera magnitud, podemos bloquear el tracking añadiendo a cada cámara el siguiente script:



```
noTrackingL.cs* [X]
Assembly-CSharp noTrackingL
1 using UnityEngine;
2 using UnityEngine.VR;
3
4 public class noTrackingL : MonoBehaviour
5 {
6     void Update()
7     {
8         transform.position = -InputTracking.GetLocalPosition(VRNode.CenterEye);
9     }
10 }
```

Figura 94. El siguiente script contrarresta el efecto del tracking

La función del script es anular el tracking, añadiendo a cada cámara una función inversa a su movimiento, con lo que el efecto del tracking se hace nulo, permaneciendo en todo momento la cámara estática, en el centro de la esfera.

- **Cameras Android:** configurar las cámaras para la aplicación de Android es un proceso bastante sencillo. Solamente tenemos que crear 2 cámaras y arrastrarle a cada una el prefab GvrViewerMain. De esta forma crearemos 2 cámaras estereoscópicas. A continuación, solo tendremos que establecer en cada cámara el ojo activo y la máscara de visualización, para que cada cámara observe una imagen.

### 7.1.1.2- Fotografía semiesférica

El procedimiento es análogo al utilizado con la fotografía esférica, salvo que esta vez proyectaremos las imágenes sobre dos semiesferas, y las cámaras las colocaremos en el centro de las esferas coincidentes. Para finalizar, encerraremos en dos cubos negros todos los objetos, para visualizar en negro las zonas sin imagen, sin que sea visible en entorno de Unity.

Vamos a ver los componentes de Unity que integran esta aplicación:

- **Objects:** necesitaremos dos semiesferas sobre las que aplicaremos cada uno de los materiales, la imagen correspondiente a cada material será vista en verdadera magnitud desde el centro de la esfera coincidente con la semiesfera.

Además, dado que la semiesfera no completa todos los ángulos de visión, podemos encerrar las semiesferas y cámaras en dos cubos negros, uno para cada capa. De esta manera no visualizaremos el entorno de Unity, cuando dirijamos la vista fuera de la imagen.

- **Meshers:** Unity no tiene ningún objeto semiesférico en su biblioteca, por lo que tendremos que utilizar 3ds Max para crear una semiesfera que importaremos a la biblioteca de Unity. Para evitar la distorsión de la imagen, hay que intentar que la semiesfera esté integrada por unos 1000 polígonos.

- **Materials:** utilizaremos un material diferente para cada textura, aplicaremos el shader al material, para que invierta las normales, y a continuación, elegimos la textura que queramos proyectar.

Además, para los cubos, utilizaremos un material con el shader que invierte las normales de los objetos, modificando el código del shader para que sea negro.

- **Shaders:** aplicaremos a los materiales un shader que invierta las normales de la superficie de los objetos para poder ver el material desde el interior.

- **Textures:** el par estereoscópico que vamos a proyectar, tiene que ser importado a la biblioteca de Unity, una vez dentro, podemos modificar el tamaño máximo del archivo en Unity, lo cual determinará la calidad de la aplicación.

- **Cameras:** el procedimiento de creación de las cámaras es el mismo que el utilizado para generar la aplicación de fotografías esféricas.

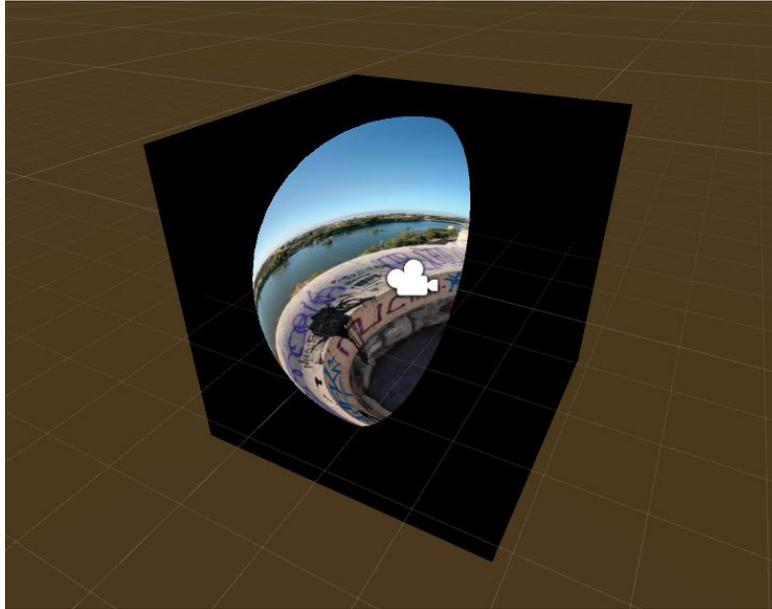


Figura 95. Elementos que integran la escena semiesférica

### 7.1.1.3- Fotografía cilíndrica

El procedimiento es análogo al utilizado con la fotografía esférica, salvo que esta vez proyectaremos las imágenes sobre dos cilindros, colocando las cámaras en el centro de dichos cilindros. Para finalizar, encerraremos en dos cubos negros todos los objetos, con el objetivo de aislarnos del entorno de Unity.

Vamos a ver los componentes de Unity que integran esta aplicación:

- **Objects:** necesitaremos dos cilindros sobre los que aplicaremos cada uno de los materiales, la imagen correspondiente a cada material será vista en verdadera magnitud desde el centro de los cilindros.

Además, dado que el cilindro no completa todos los ángulos de visión, podemos encerrar los objetos en dos cubos negros, uno para cada capa. De esta manera no visualizaremos el entorno de Unity, cuando dirijamos la vista fuera de la imagen.

- **Meshers:** Unity no tiene ningún cilindro abierto en su biblioteca, por lo que tendremos que utilizar 3ds Max para crear un cilindro que importaremos a la biblioteca de Unity. Tenemos que ser cuidadosos a la hora de crear el polígono, ajustando sus proporciones a las de la imagen.

Para evitar que se aprecien los polígonos en el borde de la imagen,

crearemos un cilindro con el máximo número de lados posible, 200 en 3ds Max, lo que se traduce en un cilindro de 400 polígonos.

- **Materials:** utilizaremos un material diferente para cada textura, aplicaremos el shader al material, para invertir las normales, y a continuación, elegimos la textura que queramos proyectar.

Además, para los cubos, utilizaremos un material con el shader que invierte las normales de los objetos, modificando el código del shader para que sea negro.

- **Shaders:** aplicaremos a los materiales un shader que invierta las normales de la superficie de los objetos para poder ver el material desde el interior.

- **Textures:** el par estereoscópico que vamos a proyectar, tiene que ser importado a la biblioteca de Unity, una vez dentro, podemos modificar el tamaño máximo del archivo en Unity, lo cual determinará la calidad de la aplicación.

- **Cameras:** el procedimiento de creación de las cámaras es el mismo que el utilizado para crear la aplicación de fotografías esféricas.

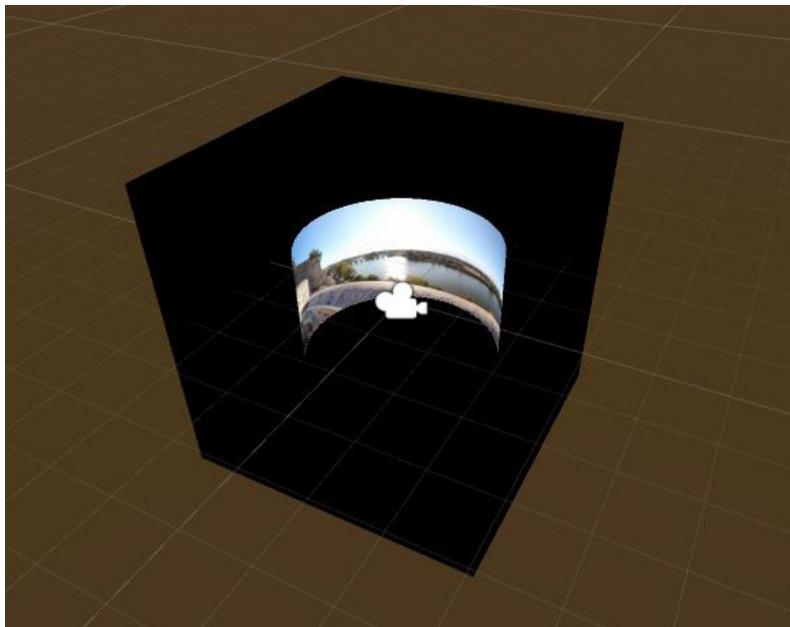


Figura 96. Elementos que integran la escena cilíndrica

#### 7.1.1.4- Fotografía plana

El procedimiento es análogo al utilizado con la fotografía esférica, salvo que esta vez proyectaremos las imágenes sobre dos planos, y las cámaras las colocaremos a una distancia prudente del centro de los planos. Para finalizar, encerraremos en dos cubos negros todos los objetos, para visualizar en negro las zonas sin imagen.

Vamos a ver los componentes de Unity que integran esta aplicación:

- **Objects:** necesitaremos dos planos de las proporciones de las imágenes a proyectar, sobre los que aplicaremos cada uno de los materiales. La imagen correspondiente a cada material será vista en verdadera magnitud desde la perpendicular al centro de la imagen.

Además, dado que el plano no completa todos los ángulos de visión, podemos encerrar los objetos en dos cubos negros, uno para cada capa. De esta manera no visualizaremos el entorno de Unity, cuando dirijamos la vista fuera de la imagen.

- **Meshers:** utilizaremos el mesher que Unity asigna por defecto a un plano.

- **Materials:** utilizaremos un material para proyectar las imágenes sobre los planos.

Además, para los cubos, utilizaremos un material con el shader que invierte las normales de los objetos, modificando el código del shader para que sea negro.

- **Shaders:** aplicaremos a los cubos un shader que invierta las normales de la superficie de los objetos para poder ver el material desde el interior. Para los planos simplemente utilizaremos un shader del tipo Unlit/Texture

- **Textures:** el par estereoscópico que vamos a proyectar, tiene que ser importado a la biblioteca de Unity, una vez dentro, podemos modificar el tamaño máximo del archivo en Unity, lo cual determinará la calidad de la aplicación.

- **Cameras:** el procedimiento de creación de las cámaras es el mismo que el utilizado para crear la aplicación de fotografías esféricas.

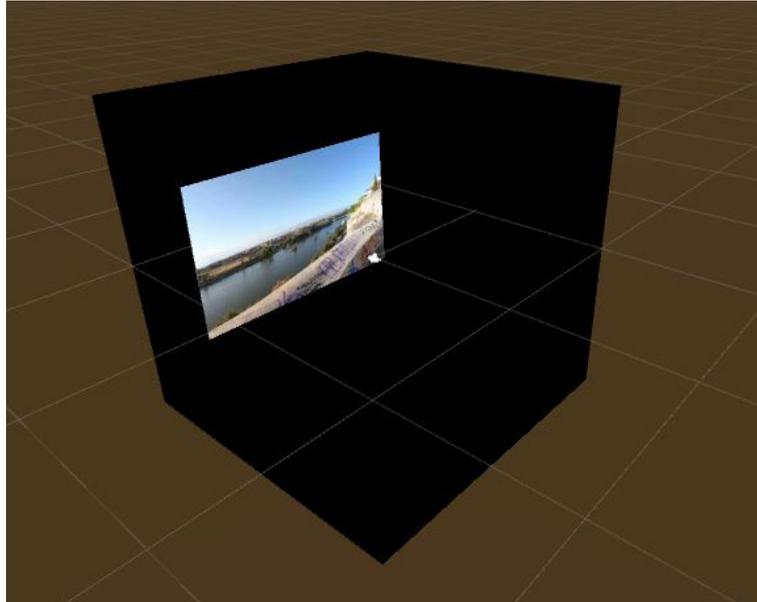


Figura 97. Elementos que integran la escena plana

### 7.1.2- Métodos basados en modelos 3D

Es el procedimiento más simple para crear una aplicación de realidad virtual. Se puede resumir en importar el modelo 3D y crear una cámara estereoscópica. Para importar de forma correcta el modelo, tendremos que copiar dentro de la carpeta de Assets del proyecto, la carpeta con el modelo 3D y sus texturas. Una vez tenemos la carpeta en Assets, aparecerá el modelo 3D en la escena. El modelo automáticamente generará un material por textura, que podemos editar posteriormente.

Vamos a ver los componentes de Unity que integran esta aplicación:

- **Objects:** una vez importado el modelo 3D junto todas sus texturas a Assets, arrastramos el modelo 3D a la escena.
- **Meshers:** los meshers se importan junto con el modelo 3D.
- **Materials:** los materiales, al importar el archivo 3D junto con sus texturas, se generarán solos.
- **Shaders:** los materiales, que se crean automáticamente, tendrán por defecto el shader Standard. Este Shader crea por defecto materiales metalizados, que no se corresponde por la apariencia natural de la textura, para solucionarlo, modificamos la textura de todos los shaders a Unlit/Texture.

- **Textures:** las texturas son importadas conjuntamente con el modelo 3D en la misma carpeta.
- **Cameras HTC Vive:** en primer lugar, para crear la cámara estereoscópica, arrastraremos el Prefab [CameraRig] dentro del paquete de Assets de Steam VR. En segundo lugar, dado que queremos crear una aplicación estática, creamos un game object vacío que contendrá al [CameraRig]. A dicho objeto le añadimos un script como el que vemos en la figura 92, para contrarrestar el efecto del tracking.
- **Cameras Android:** creamos una cámara de la biblioteca de Unity y le aplicamos el Prefab GvrViewerMain de los Assets de Google. De esta forma la cámara se convertirá en estereoscópica.

## 7.2- Análisis de métodos de recreación de entornos estáticos

En este apartado se va a proceder a analizar todos aquellos entornos que han sido creados a partir de parejas de imágenes estereoscópicas o a partir de un modelo 3D en el que permanecen estáticos todos sus elementos.

Se clasificarán según el campo de visión que ofrezcan dichos entornos, que puede ser de 360°, 180°, cilíndrico o plano. Cada tipo de entorno podrá ser representado por varios métodos, lo cual repercutirá en la calidad de la aplicación, así como su tiempo de producción.

Vamos a analizar diferentes factores dentro de cada aplicación:

- **Peso de la aplicación:** puede ser un factor limitante dependiendo del dispositivo que utilicemos para abrirla.
- **Calidad de imagen:** está relacionado con la nitidez, exposición y color de la imagen. Dependerá del dispositivo que utilicemos para fotografiar la escena.
- **Inmersividad:** lo fiel que la aplicación es a la realidad. Cuanto más fiel sea la representación, la experiencia de usuario será más real. En este punto intervienen factores como la distorsión de las imágenes o la disparidad del par. Ya que, si hay distorsión, nuestro cerebro no podrá mezclar las imágenes de forma correcta y sin disparidad no veremos tridimensionalmente.
- **Esquema de disparidad:** no todos los puntos del par estereoscópico conservan la misma disparidad. Esto ocurre porque la escena está formada por diferentes fotografías cuya perspectiva es cónica, por lo cual, solamente la vertical central de estas parejas de fotografías, conservará la disparidad deseada, a medida

que nos alejemos a los extremos de la foto, la disparidad irá disminuyendo. Por lo cual cuantas más fotos integren el panorama, mayor será la disparidad general del par estereoscópico. Estudiaremos mediante un esquema, como se reparte la disparidad a lo largo de la imagen.

- **Tiempo de generación de las imágenes:** salvo cuando utilicemos directamente un modelo 3D para representar un entorno, necesitaremos imágenes estereoscópicas para representar el entorno deseado. Como hemos visto, hay multitud de dispositivos con los que podemos capturar un par estereoscópico. El tiempo de generación de estos pares difiere mucho según el dispositivo que utilicemos para generar las imágenes.

- **Recursos utilizados:** cada método de obtención de imágenes utiliza una serie de dispositivos cuyo coste conviene analizar a la hora de asumir un proyecto.

### 7.2.1- Métodos basados en fotografía esférica

Podemos obtener pares esféricos estereoscópicos utilizando varios dispositivos, la calidad de las imágenes obtenidas, varía mucho de un método a otro, lo cual repercutirá notoriamente en las cualidades de la aplicación. Además, analizaremos el coste a nivel de recursos para obtener cada imagen.

#### 7.2.1.1- Cámara réflex

La cámara utilizada para realizar las fotografías esféricas ha sido la Canon EOS 750D, montada con un objetivo de 18mm. Dada la configuración de la cámara, se precisa un total de 40 fotografías para realizar una captura esférica del entorno.

Cabe destacar que para otra configuración de cámara cambiará el número de fotos necesario para completar la escena, repercutiendo en algún parámetro de la imagen.

- **Peso de la aplicación:** debido al tamaño de las imágenes utilizadas, el peso de la aplicación se verá influenciado por este aspecto. La aplicación de HTC Vive, pesa 57 MB, lo cual es despreciable teniendo en cuenta las características del ordenador al que conectaremos las gafas.

---

Para smartphone Android, que es dispositivo más limitado a nivel de almacenamiento, la aplicación pesa 39 MB.

· **Calidad de imagen:** la cámara réflex es la herramienta utilizada por los profesionales de la fotografía. Es por ello que el producto resultante del cosido de imágenes, es una fotografía esférica de gran calidad, con muy buena nitidez en todos los objetos de la imagen y unas cualidades de color y contraste muy realistas.

Este tipo de cámaras, permiten el control de muchos de sus parámetros, por lo que se requiere de un usuario experimentado para sacar el máximo partido a la fotografía.

La resolución final de la imagen será de 8192x4096 píxeles, aunque podría ser mayor, conservamos este tamaño, ya que Unity no admite texturas superiores a 8192 píxeles.

· **Inmersividad:** desgraciadamente la forma en la que se cosen las imágenes con este método, en la que ninguna está referenciada en el espacio, hace que la posición en la que se encuentran determinados objetos no sea análoga en su pareja. Apareciendo zonas en las que hay que hacer un esfuerzo visual mayor para poder visualizar el contenido.

Se puede decir que, aunque el contenido se logra ver en 3D, a algunos usuarios les cuesta más que a otros adaptar la visión a estas aplicaciones y al quitarnos las gafas puede aparecer la sensación de haber forzado un poco la vista.

A pesar de este problema, la sensación de inmersividad es muy buena, ya que la gran calidad de las imágenes ayuda mucho a percibir los colores y los detalles de una forma muy realista.

· **Esquema de disparidad:** al utilizar la cámara con un objetivo de 18 mm, es necesario sacar 12 fotos en el plano horizontal, lo que se traduce en una disparidad muy bien distribuida por toda la imagen.

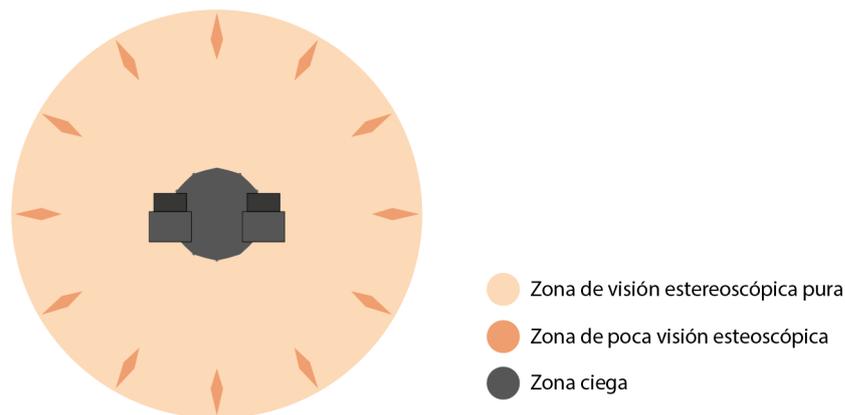


Figura 98. Esquema de la calidad estereoscópica del panorama esférico utilizando una cámara réflex

- **Tiempo de generación de las imágenes:** una de las pegas de este método es el largo tiempo que se tarda en nivelar el trípode, anclar la cámara a la rótula, ajustar la posición de la cámara, ajustar los parámetros fotográficos de la cámara y sacar las 80 fotografías (pueden variar en función del objetivo utilizado) que se requieren para obtener el par estereoscópico. Este tedioso procedimiento suele durar unos 20 minutos.

A este tiempo hay que sumarle el tiempo de preparación de las imágenes. Tendremos que coser las imágenes, eliminar el trípode e igualar el cielo cuando haya nubes movidas. Es por esto que en exteriores tardaremos algo más en conseguir las imágenes finales. El tiempo de postproducción depende mucho de las características del entorno y sobre todo de su suelo, ya que determinará la dificultad de eliminar el trípode de la imagen. El tiempo invertido en esta fase suele oscilar entre 30 minutos y 1 hora.

- **Recursos utilizados:** para realizar dicho procedimiento, necesitaremos una cámara réflex, un trípode con rótula medidora de ángulos y una rótula estereoscópica. El precio de estos 3 elementos puede rondar los 650€, pudiendo alcanzar cifras mucho mayores en función de las características de la cámara.



Figura 99. Par estereoscópico del interior de la Catedral de Zamora (elaboración propia)

Los detalles comentados sobre el mal solapamiento de estas imágenes no se aprecian a simple vista. Observándolas una por una, vemos 2 fotografías esféricas de muy buena calidad, con unos niveles de exposición, brillo y contraste muy realistas.

Aunque los efectos de la falta de concordancia entre imágenes suelen ser muy tenues, en este par encontramos puntos críticos en los pivotes de la barandilla dorada, en una zona muy cercana a la toma de la imagen.



Figura 100. Par estereoscópico del Mirador del Troncoso (Zamora) (elaboración propia)

Este par en concreto no presenta zonas de falta de concordancia espacial entre puntos que impidan la fusión entre imágenes. El efecto es tan tenue que no se produce este defecto, pero sí que en un inicio cuesta más acomodar la vista y se produce fatiga visual con estas imágenes.

Por lo demás el par estereoscópico es de altísima calidad y sorprende de la profundidad de campo que se consigue.

### 7.2.1.2- Smartphone

Se ha utilizado el iPhone 6 para generar el par estereoscópico. Ha sido elegido por disponibilidad, pero seguramente se pueda utilizar una gran variedad de modelos para la obtención de pares estereoscópicos.

- **Peso de la aplicación:** debido al tamaño de las imágenes utilizadas, el peso de la aplicación se verá influido por este aspecto. La aplicación de HTC Vive, pesa 57 MB, lo cual es despreciable teniendo en cuenta las características del ordenador al que conectaremos las gafas.

Para smartphone Android, la aplicación pesa algo más que la obtenida con la cámara réflex, unos 45 MB de media.

- **Calidad de imagen:** las imágenes obtenidas no se pueden comparar con las de la cámara réflex. A nivel de cosido, la cámara réflex supera con creces el trabajo realizado por la aplicación Google Street View, que en ocasiones crea zonas borrosas o no coincidentes. Además, en cuanto a nitidez, las fotografías tomadas con smartphone no muestran con tanta claridad los objetos lejanos.

La resolución de cada imagen es de 8704x4352 píxeles, aunque quedará reducida a 8192x4096 píxeles, ya que Unity no admite texturas mayores a 8192 píxeles.

- **Inmersividad:** la inmersividad es bastante satisfactoria, aunque de forma individual cada una de las imágenes no llega al nivel de las realizadas con la cámara réflex, al mezclar ambas usando las gafas obtenemos una escena estereoscópica de una calidad superior a las imágenes por separado.

Por otro lado, el cosido se realiza a partir de imágenes autoproyectadas sobre una esfera, por lo que, aunque haya puntos con una peor costura, las imágenes del par serán más afines entre sí, ya que cada objeto aparece representado en su sitio.

Esto se traduce en una menor fatiga visual al visualizar el contenido y una mayor facilidad para el usuario de acomodar su visión. La inmersividad por tanto será muy buena, aunque puedan aparecer pequeños detalles que no presenten la suficiente nitidez.

- **Esquema de disparidad:** la aplicación Google Street View nos obliga a sacar 11 fotos en el plano horizontal, una menos que con la cámara réflex, con lo que la disparidad estará también muy bien repartida por toda la escena.

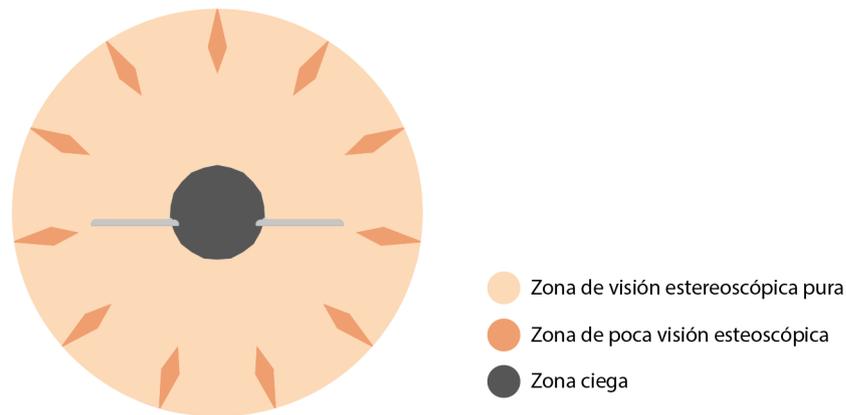


Figura 101. Esquema de la calidad estereoscópica del panorama esférico utilizando un smartphone

· **Tiempo de generación de las imágenes:** en este aspecto este método supone una ventaja respecto del anterior, ya que por un lado podemos utilizar dos móviles simultáneamente, reduciendo el tiempo de toma de fotos a la mitad. Y la aplicación de Google, hace mucho más rápida la toma de fotos, y nos ahorra el cosido de las imágenes. Con un solo dispositivo, podemos movernos en torno a tiempo de 10 minutos para la toma de fotos.

Aunque las imágenes necesitarán ser editadas para eliminar el trípode de ambas e igualar el cielo de ambas fotografías, cuando aparezcan nubes movidas. Esta tarea suele llevar alrededor de media hora.

· **Recursos utilizados:** los elementos necesarios para llevar a cabo dicho procedimiento son un smartphone (aunque podemos utilizar dos), un trípode y el soporte para smartphone impreso en 3D.

El precio de estos 3 elementos, teniendo en cuenta que el teléfono utilizado ha sido un iPhone 6 es de unos 450€. La ventaja es que, si despreciamos el precio del smartphone (que ha de ser compatible con la aplicación Google Street View), que es un útil que, si no disponemos de él, muy probablemente podamos pedir a alguien de nuestro entorno, el precio del resto de componentes no alcanza los 100€.



Figura 102. Par estereoscópico esférico del interior de la Catedral de Zamora (elaboración propia)

Esta imagen de la Catedral de Zamora, como se observa es mucho más clara que la tomada con la réflex. Esto sucede porque el smartphone, al utilizar la aplicación Google Street View, configura de forma automática los parámetros de la cámara. Como resultado, obtenemos una imagen sobreexpuesta.

En cuanto a la nitidez, algún detalle en la pared presenta menos claridad de la esperada, sin que suponga un problema mayor.

Por lo demás, el cosido de esta imagen presenta muy pocos fallos, ya que, por ser un interior, la iluminación es homogénea durante todo el tiempo de captura.



Figura 103. Par estereoscópico esférico del Mirador del Troncoso (Zamora) (elaboración propia)

En este par percibimos una escena muy bien iluminada, en la que tenemos objetos muy cercanos y otros que se pierden en el infinito.

En cuanto a la nitidez de los objetos lejanos, se echa de menos la ofrecida por la cámara réflex, pero la que ofrecen estas imágenes es más que suficiente para que la sensación de inmersión sea de muy buena calidad.

Por otro lado, en los objetos más cercanos aparece algún que otro problema de cosido, como es el caso del muro frontal del mirador. Por lo demás, el resto de la escena se ve perfectamente sin sobrecargar la vista.

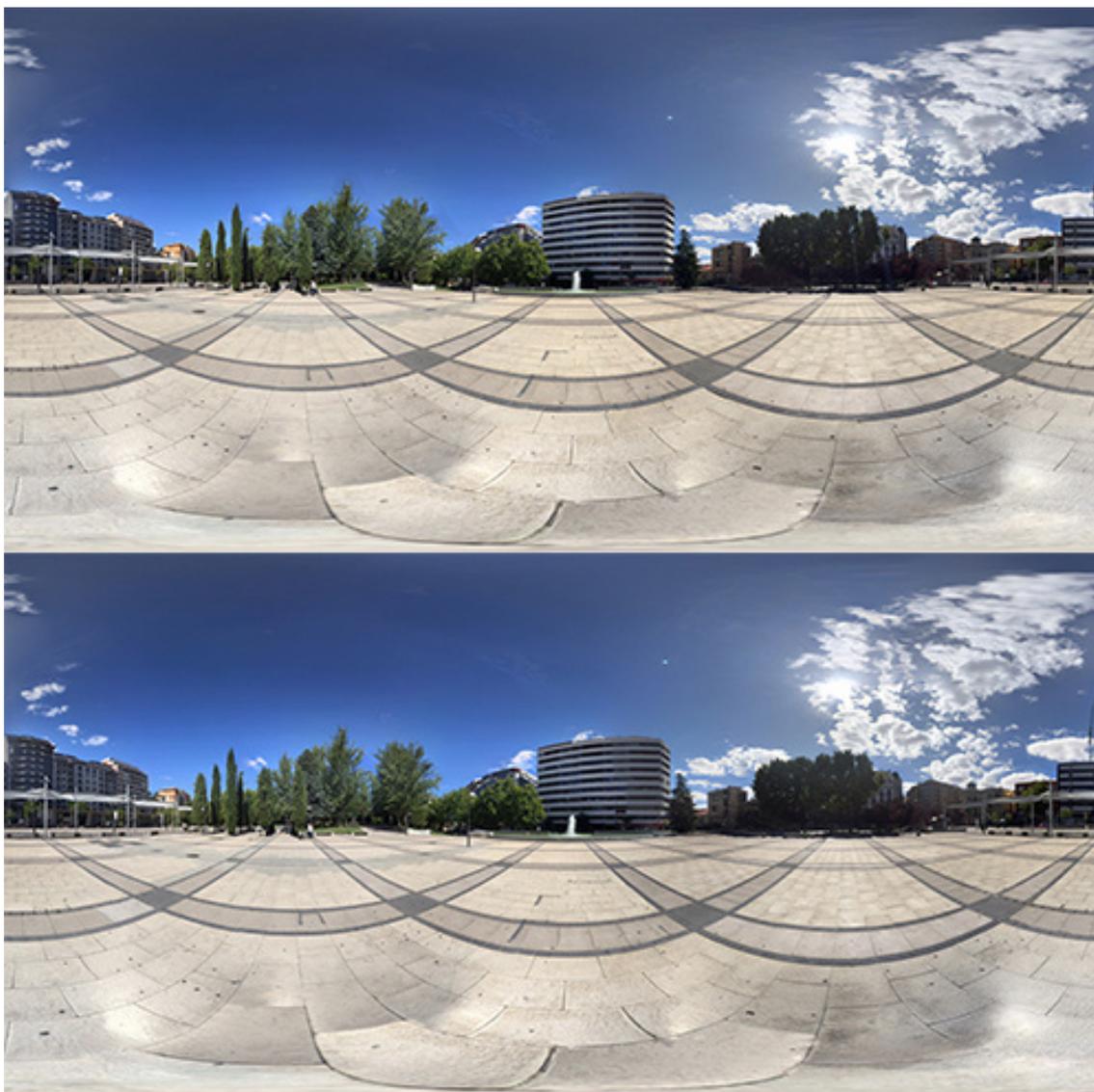


Figura 104. Par estereoscópico esférico de la Plaza de la Marina Española (Zamora) (elaboración propia)

Este par ha sido tomado de forma simultánea con dos iPhone 6, de este modo podemos obtener pares estereoscópicos de lugares públicos, en el que el movimiento de los viandantes no presente un problema.

Podemos ver que aparecen defectos en algunas zonas, como puede ser una mancha en el edificio central de la imagen o un error de cosido en la grúa que aparece en el extremo derecho de la foto. Aunque al visualizar el par con nuestro dispositivo HDM, estos defectos se atenúan, siendo la sensación de inmersividad muy buena.

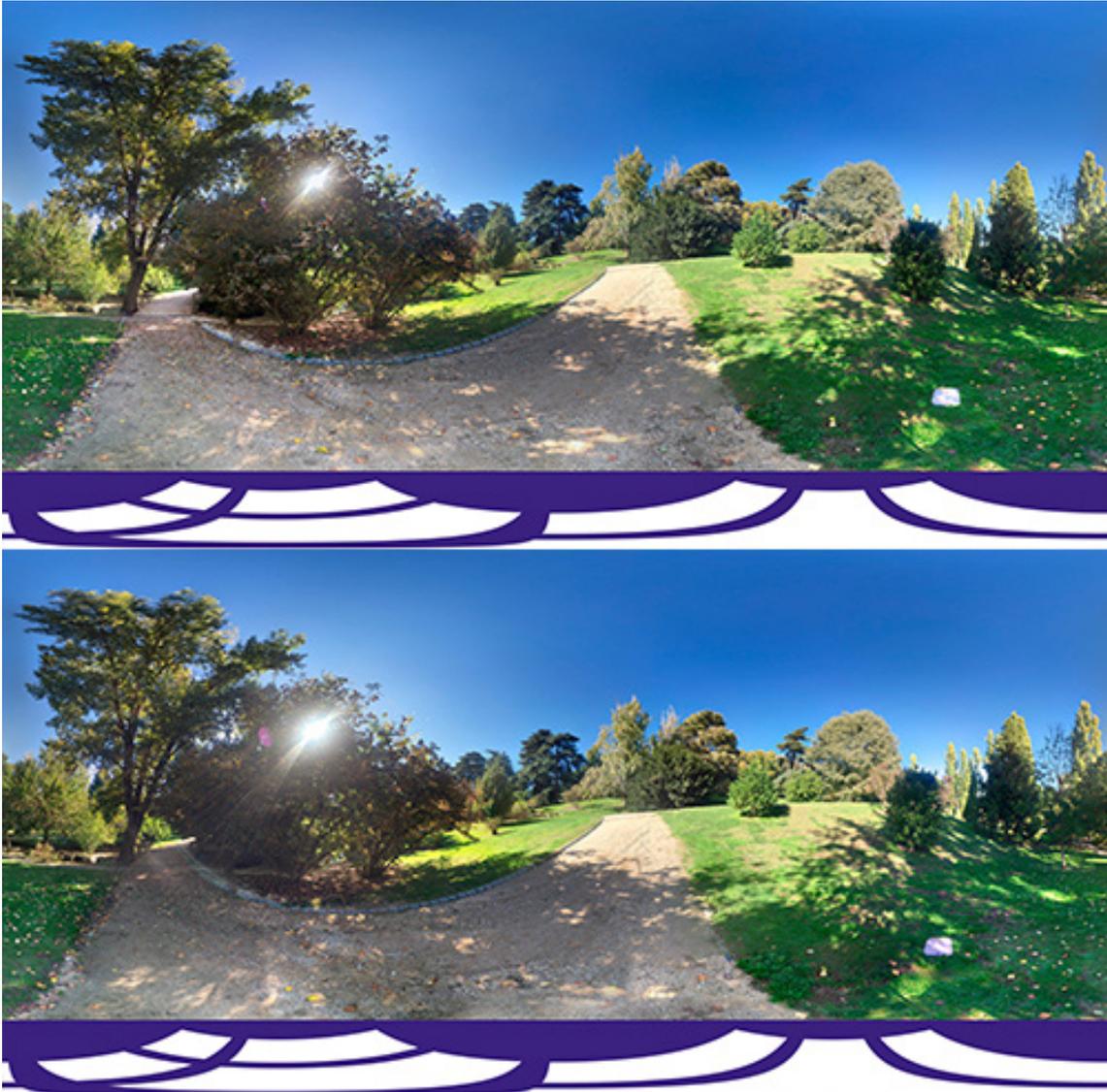


Figura 105. Par estereoscópico esférico del Parque del Oeste (Madrid) (elaboración propia)

Este par ha sido tomado utilizando un solo objetivo. Por tratarse de una escena orgánica, sin ningún detalle demasiado alejado y con muy buena iluminación, se expresan al máximo las cualidades del teléfono. El resultado es una escena extraordinaria, muy ergonómica y totalmente inmersiva.

La parte morada de abajo, es el logo de la Escuela de Ingenierías Industriales de Valladolid. Que se verá en verdadera magnitud una vez proyectada la imagen. Esta es otra forma más rápida y muy extendida de eliminar el trípode.

Además, esta imagen se comparará más adelante con otras imágenes de la misma escena, para analizar diversos métodos de generación de pares estereoscópicos.

### 7.2.1.3- Cámara 360°

Para obtener las imágenes esféricas se ha utilizado la cámara Ricoh Theta SC. Este periférico es capaz de capturar fotografías esféricas de 24 megapíxeles con un solo clic.

Para crear un panorama estereoscópico con esta cámara, tendremos que tomar dos capturas, separadas 65 mm entre sí. Para ello nos podemos ayudar de la rótula estereoscópica.

Estas capturas por sí solas no conforman un par estereoscópico, necesitan de postproducción para lograr tales fines. Y es que, la fotografía tomada con el objetivo delantero de la fotografía 1, ha de ser unida con el objetivo trasero de la fotografía 2. Para ello, nos ayudamos de una herramienta de retoque fotográfico como Photoshop.

- **Peso de la aplicación:** las imágenes generadas por esta cámara no son demasiado pesadas, lo cual producirá aplicaciones algo más livianas. El peso de la aplicación para HTC es de unos 45 MB.

Para smartphone Android, observamos una notable reducción respecto de las aplicaciones vistas hasta ahora, el peso de la aplicación se queda en 25 MB.

- **Calidad de imagen:** teniendo en cuenta la gran superficie que abarca la imagen, nos encontramos ante panoramas que no proporcionan demasiada nitidez en los detalles. A pesar de su escasa resolución, 3584x1792 píxeles, las cualidades cromáticas de la imagen son bastante buenas.

- **Inmersividad:** las aplicaciones que utilizan este tipo de imágenes no crean una sensación de inmersividad demasiado buena. Por un lado, la poca resolución de la imagen hace que no podamos apreciar la escena con la suficiente nitidez como para sentirnos inmersos. Además, la disparidad está muy mal distribuida a lo largo de la escena, por lo que nos encontramos ante una aplicación que no posee muy buenas propiedades tridimensionales.

Probablemente, si nos encontrásemos en una escena en la que ningún objeto se alejase demasiado de la cámara, como un interior, la inmersividad de la aplicación mejoraría.

- **Esquema de disparidad:** debido a las escasas fotografías que componen la escena, dos por panorama, la disparidad pasará de ser nula en algunas zonas a completa en otras. Por lo cual nos encontramos ante un esquema muy extremo.

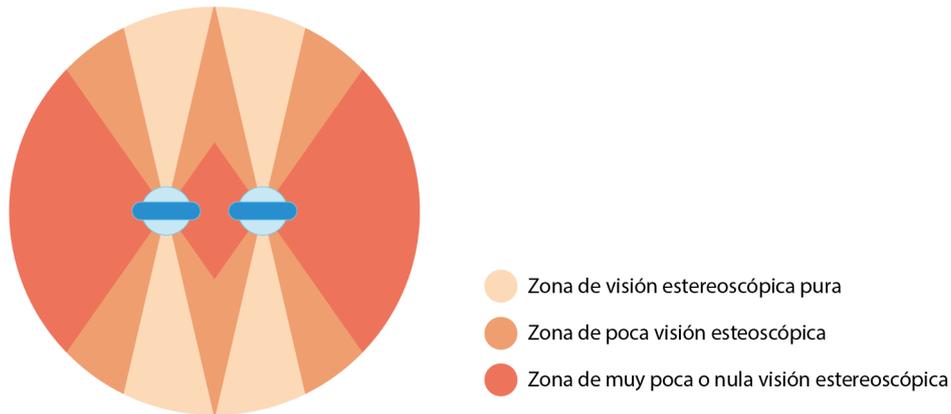


Figura 106. Esquema de la calidad estereoscópica del panorama esférico utilizando una cámara 360°

· **Tiempo de generación de las imágenes:** el tiempo de toma de fotos es bastante reducido, solo tenemos que montar el trípode, colocar la cámara en la rótula y sacar dos fotografías. Pero como comentábamos, a esto hay que añadirle un proceso de recorte de las imágenes y eliminación del trípode que, aunque no es muy complejo, puede durar entre 10 y 30 minutos.

· **Recursos utilizados:** para la obtención de las imágenes necesitamos una cámara 360°, una rótula estereoscópica y un trípode que permita situar a la cámara a una altura de 1,70 metros.

El trípode y la rótula pueden costar alrededor de 20€ cada uno, lo más caro es la cámara, en este caso, la Ricoh Theta SC se puede adquirir por unos 240€. Por lo que por 260€ podríamos llevar a cabo este proceso.

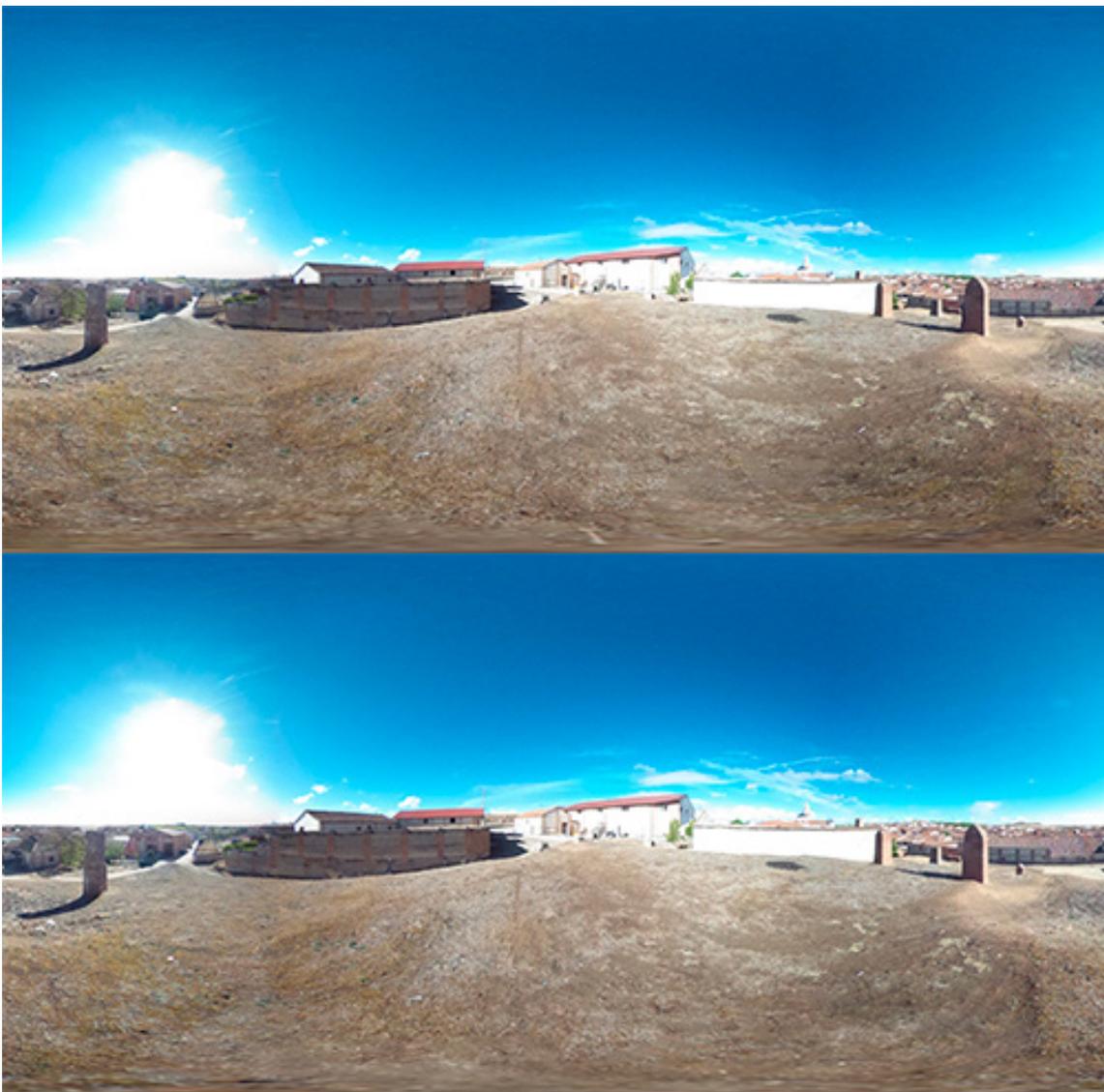
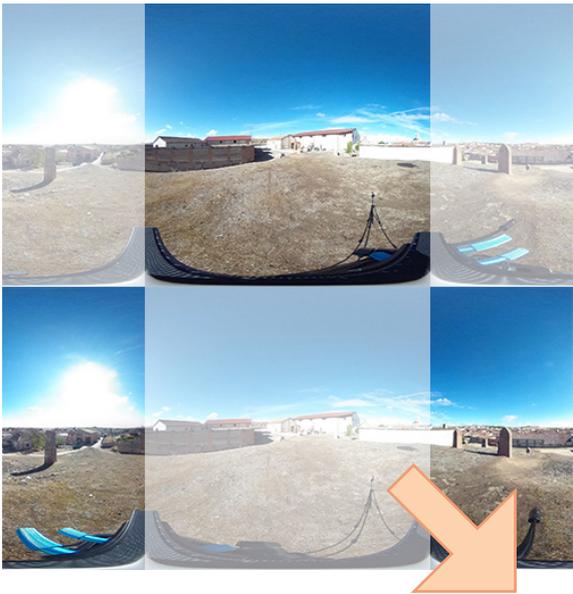


Figura 107. Par estereoscópico esférico de Rueda (Valladolid) (elaboración propia)

En las imágenes de la página anterior observamos las fotografías originales realizadas con la cámara Ricoh Theta, arriba en pequeño y el par estereoscópico resultante.

Para conseguir el par estereoscópico, es necesario dividir cada una de las imágenes iniciales. Como se puede ver, las zonas de las imágenes con filtro blanco formarán el panorama correspondiente a un ojo, mientras que las zonas sin filtro formarán el otro panorama.

Para mejorar la calidad de las imágenes, se ha utilizado el mismo cielo para las dos imágenes, de esta forma el cosido del cielo será óptimo, y no aparecerán movidas las nubes. El resultado es un cielo que no se presenta de forma tridimensional, pero por su lejanía el efecto estereoscópico que podrían ofrecer las nubes es depreciable.

En cuanto a la calidad de la escena, llama la atención la escasa homogeneidad que presenta la disparidad de la escena, mientras que algunos objetos se observan tridimensionalmente, otros carecen en absoluto de este efecto, lo cual resulta chocante. Además, la lejanía a la que se encuentran la mayoría de elementos que integran la imagen, hace que muy pocas zonas aparezcan con la nitidez suficiente como para que se consiga una buena sensación de inmersividad.

#### 7.2.1.4- Vuze Camera

La cámara de Vuze es el único dispositivo que permite capturar en un solo disparo un par estereoscópico esférico. Esto la convierte en el dispositivo más rápido para generar contenido de realidad virtual 360°. Dado que para este proyecto no ha sido posible disponer de este dispositivo, vamos a analizar la calidad de la aplicación a partir de fotogramas extraídos de vídeos de YouTube o cedidos por la propia compañía de la cámara, Humaneyes Technologies LTD.

- **Peso de la aplicación:** el tamaño de las imágenes no es demasiado grande, lo cual supone una reducción del tamaño. La aplicación de HTC Vive, pesa 40 MB, lo cual es despreciable teniendo en cuenta las características del ordenador al que conectaremos las gafas.

Para smartphone Android, notamos una reducción del 50% en el peso, lo cual es un dato bastante notable. Siendo el peso de la aplicación unos 22 MB de media.

- **Calidad de imagen:** la calidad de las imágenes no es demasiado elevada, no obtenemos demasiada nitidez en objetos lejanos. En cuanto a exposición, brillo y contraste, tampoco nos encontramos ante una gran cámara. La resolución de cada imagen es de 3840x1080 píxeles, para las imágenes extraídas de YouTube y de 3840x1920 píxeles para las cedidas por la compañía. Esta resolución no es suficiente para representar pequeños detalles con nitidez.

Además, debido a que los pares de cámaras no se encuentran girados respecto de su punto nodal, el cosido de las imágenes no va a poder efectuarse correctamente en la mayoría de los casos. Y es que, cada uno de los pares, al no rotar sobre su punto nodal, no puede efectuar el cosido de forma correcta con la imagen contigua, en todos los niveles de profundidad de campo. Este es un efecto que vimos en la definición de punto nodal y sucede porque la posición relativa de los objetos situados en diferentes planos varía cuando rotamos la cámara. En esta cámara la rotación se produce bastante lejos del punto nodal, de 90° en 90°, por lo cual la zona de costura entre imágenes tendrá puntos críticos cuando haya mucha variación en la distancia a la que se encuentran los objetos de la cámara.

Este problema es inevitable en este tipo de dispositivos, ya que, si las cámaras rotasen en torno a su punto nodal, unas cámaras se interpondrían en la visión de las otras.

- **Inmersividad:** al utilizar solamente 8 cámaras para capturar toda la escena, la falta de concordancia espacial, que era un problema en las imágenes obtenidas con la cámara réflex, se soluciona de forma extraordinaria. Con lo que las imágenes se adaptan fácilmente a los ojos y se reduce la posibilidad de mareo.

La contra, es la poca definición que ofrecen las imágenes, algo más escasa que las obtenidas con smartphone, que, unida a su poca resolución, puede no ser suficiente para representar con absoluto realismo algunas escenas con muchos detalles. Aunque la escena cedida, con un 77% más de píxeles, ofrece resultados superiores.

- **Esquema de disparidad:** con tan solo 4 fotos realizadas en torno al plano horizontal, la disparidad no se distribuye de forma uniforme por la escena. No es algo que se aprecie en exceso, pero si agudizamos los sentidos observamos que las zonas de cosido tienen algo menos de disparidad.

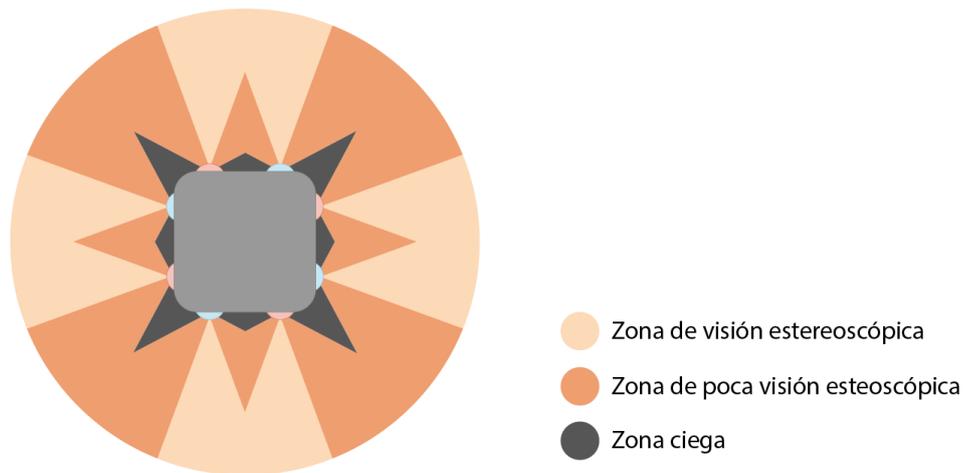


Figura 108. Esquema de la calidad estereoscópica del panorama esférico utilizando la Vuze Camera

- **Tiempo de generación de las imágenes:** la Vuze camera no tiene rival en cuanto a rapidez. Con un solo clic obtenemos una representación espacial 360° de toda la escena, generando una imagen para cada ojo.

El cosido de la imagen se realiza de forma muy rápida con un software desarrollado por Vuze. Posteriormente habría que editarla en Photoshop para eliminar el trípode.

- **Recursos utilizados:** para obtener nuestra fotografía con la Vuze camera, tan solo necesitamos un trípode y la propia cámara. El trípode no requiere desempeñar ninguna función más que la de sostener la cámara, por lo que por 20€ podemos encontrar trípodes sencillos que realicen la función. La mayor pega es el precio de la cámara, 999€.



Figura 109. Par estereoscópico del Times Square (New York City) (elaboración propia)

En el siguiente par, que recoge el Times Square de Nueva York, apreciamos varias cualidades relativas a la cámara. En línea discontinua verde están marcadas las zonas de costura de las 4 imágenes que componen cada panorama.

Tener presentes estas líneas es importante para comprender la naturaleza de la imagen. Empezaremos a comentar lo que ocurre en cada línea de costura de izquierda a derecha.

En la primera zona, observamos que la barandilla que se encuentra en un primer plano, presenta muy mala costura, mientras que la costura del edificio amarillo, en un segundo plano se ha realizado de forma correcta.

Lo mismo ocurre en la segunda zona, donde se ha cuidado la costura de los objetos situados en un segundo plano.

En la tercera zona ocurre lo contrario, el chico del polo blanco se encuentra sorprendentemente bien cosido, mientras que los carteles que se encuentran en un segundo plano tienen zonas dobles.

En la cuarta zona, se ha dado prioridad al segundo plano, encontrando solamente defectos perceptibles de costura en la barandilla localizada en el primer plano.

En cuanto a la nitidez, se echa en falta algo de resolución para apreciar los carteles más lejanos.

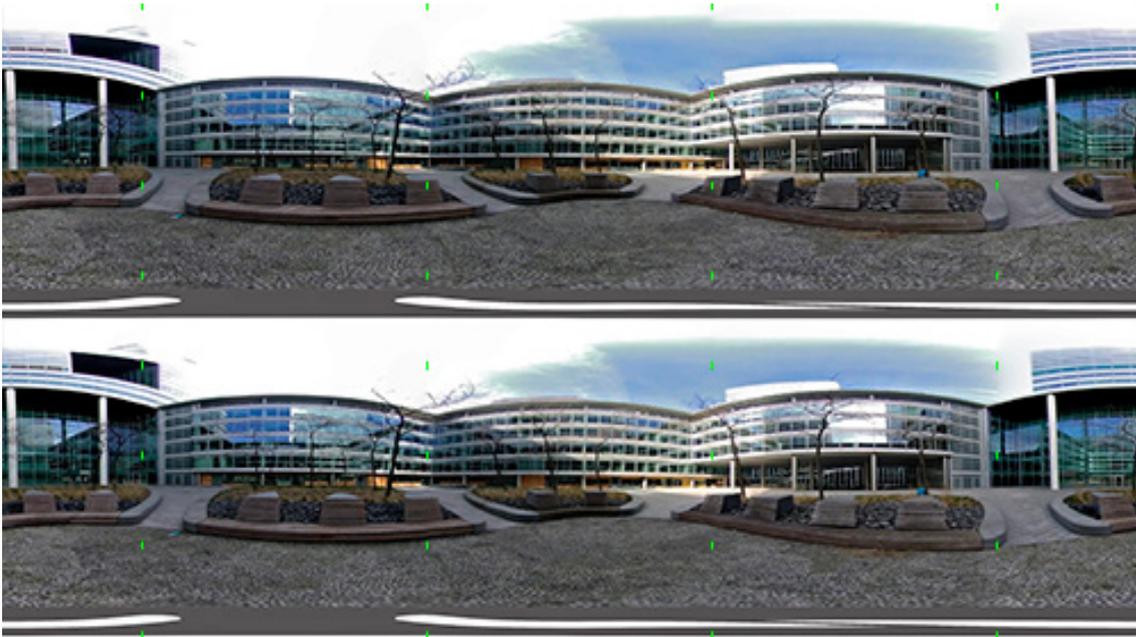


Figura 110. Par estereoscópico de una plaza en el distrito financiero de Zuidas (Amsterdam) (elaboración propia)

El siguiente par, muestra muy buen cosido, ya que no tiene demasiados objetos en diferentes planos. Los edificios muestran muy buen cosido, por lo que podemos deducir que la escena ha sido optimizada para representarlos de forma correcta. Prueba de ello es el mal cosido que presentan los bancos, aunque por ser elementos secundarios no resalta demasiado este defecto.

Por lo demás, al no haber elementos muy alejados, toda la escena aparece con la suficiente nitidez como para que la sensación de inmersividad sea satisfactoria.



Figura 111. Par estereoscópico de un lago (Amir Weiss) (elaboración propia)

Vemos en esta página un par estereoscópico realizado por Amir Weiss. La imagen ha sido cedida por Humaneyes Technologies LTD.

Se aprecia un cosido muy bueno, en parte porque se trata de un entorno orgánico. En este tipo de entornos el cosido no necesita ser tan preciso para disimularse, es por esto que este tipo de entornos son ideales para esta cámara.

Por otro lado, este par, con una resolución de 3840x1920 píxeles, supone un aumento del 77% en el número de píxeles. Lo cual aumenta considerablemente la sensación de inmersión en la escena.

### 7.2.1.5- Renders

Podemos obtener pares estereoscópicos esféricos a partir de procesos de renderizado, con el motor de renders V-Ray. Lo primero que necesitamos para conseguir estas imágenes es un modelo 3D previamente texturizado. El proceso consistirá en abrir en modelo en algún programa de modelado 3D como 3ds Max, que permita la obtención de renders, para posteriormente iluminar la escena y tomar las panorámicas con una cámara.

En este caso, analizaremos el proceso para obtener un par estereoscópico a partir del modelo 3D de una bodega utilizando 3ds Max, con V-Ray como motor de render. Probablemente se puedan utilizar otros motores de renderizado, pero hay que asegurarse que permitan crear cámaras estereoscópicas y obtener panoramas esféricos.

Una vez tenemos el modelo dentro del programa, tenemos que asegurarnos de que las texturas se proyectan de forma correcta a lo largo de toda la maya. Si trabajamos con varias texturas, lo recomendable es utilizar un material Multi/Sub-Object, integrado por diferentes materiales V-RayMtl, uno por textura.



Figura 112. Modelo 3D de la bodega en 3ds Max

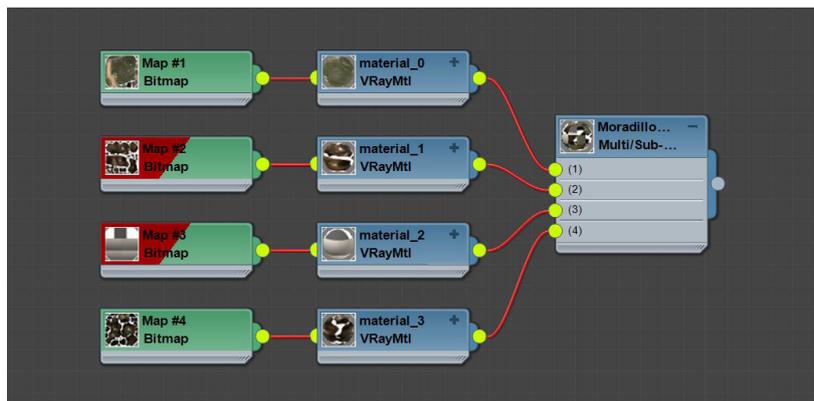


Figura 113. Esquema multimaterial 3ds Max

Para configurar la iluminación y dado que estamos trabajando con texturas, la forma más sencilla y efectiva de iluminar correctamente la escena es aplicando al apartado de iluminación propia (Self-ilumin) el mapa correspondiente a la textura.

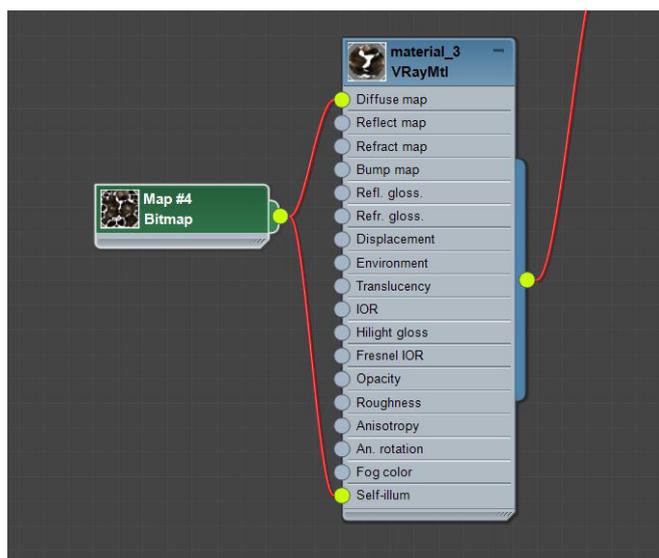


Figura 114. Esquema self-ilumin material

Solo falta por añadir a la escena la cámara estereoscópica. Este tipo de cámara se consigue creando una PhysicalCamera, y añadiendo a la escena un helper VRayStereoscopic. Llegados a este punto podemos empezar a configurar el motor de renderizado. Dentro de esta configuración, elegimos el aspecto de imagen 2:1 y dentro del apartado de VRay, seleccionamos el tipo de cámara como esférica con un override FOV de 360°.

De esta forma, conseguimos pares esféricos con un efecto de estereoscopia total, homogéneamente repartido por toda la escena.

· **Peso de la aplicación:** el peso de la aplicación dependerá en gran medida del tamaño de las imágenes que obtengamos. Para la escena representada, con imágenes de 4096x2048 píxeles es suficiente para no perder detalle. En este caso el peso de la aplicación será de 46 MB para las gafas HTC.

Para smartphone Android, el peso sería de 30 MB.

· **Calidad de imagen:** la calidad de la imagen dependerá principalmente de la calidad de las texturas del modelo 3D y de la configuración de la cámara. En este aspecto el motor de renderizado V-Ray nos permite modificar una gran cantidad de parámetros para obtener el resultado deseado.

· **Inmersividad:** el hecho de poder adaptar las características de la imagen a las necesidades del proyecto nos hace posible crear escenas muy inmersivas donde el límite lo pondrá la calidad de nuestro modelo 3D. Además, al no depender de métodos físicos, el par estereoscópico es capturado de forma ideal, por lo cual el efecto tridimensional no variará en toda la escena.

Estas características hacen que la sensación de inmersividad esté muy conseguida.

· **Esquema de disparidad:** como ya se ha comentado, al capturarse el par estereoscópico por medio de cálculos matemáticos, el efecto estereoscópico estará repartido de forma homogénea por toda la imagen.

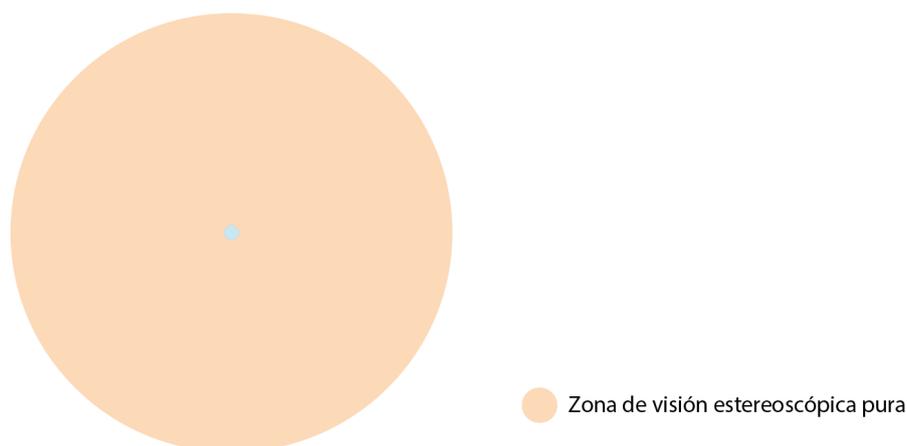


Figura 115. Esquema de la calidad estereoscópica del panorama esférico utilizando renders

· **Tiempo de generación de las imágenes:** el tiempo de generación de las imágenes corresponderá a la duración del renderizado de las mismas. Hay muchos factores que pueden afectar a este tiempo, como lo son la resolución de las imágenes, la potencia computacional de nuestro equipo o la configuración del motor de renderizado.

Por lo cual, el tiempo no es fácil de predecir con exactitud, pero raramente sobrepasará los 5 minutos para la obtención de un par esférico con una resolución de 4096x2048 píxeles.

· **Recursos utilizados:** para obtener estas imágenes será necesario disponer de un ordenador con el software oportuno instalado. El precio de un portátil básico para esta tarea parte de los 500€.

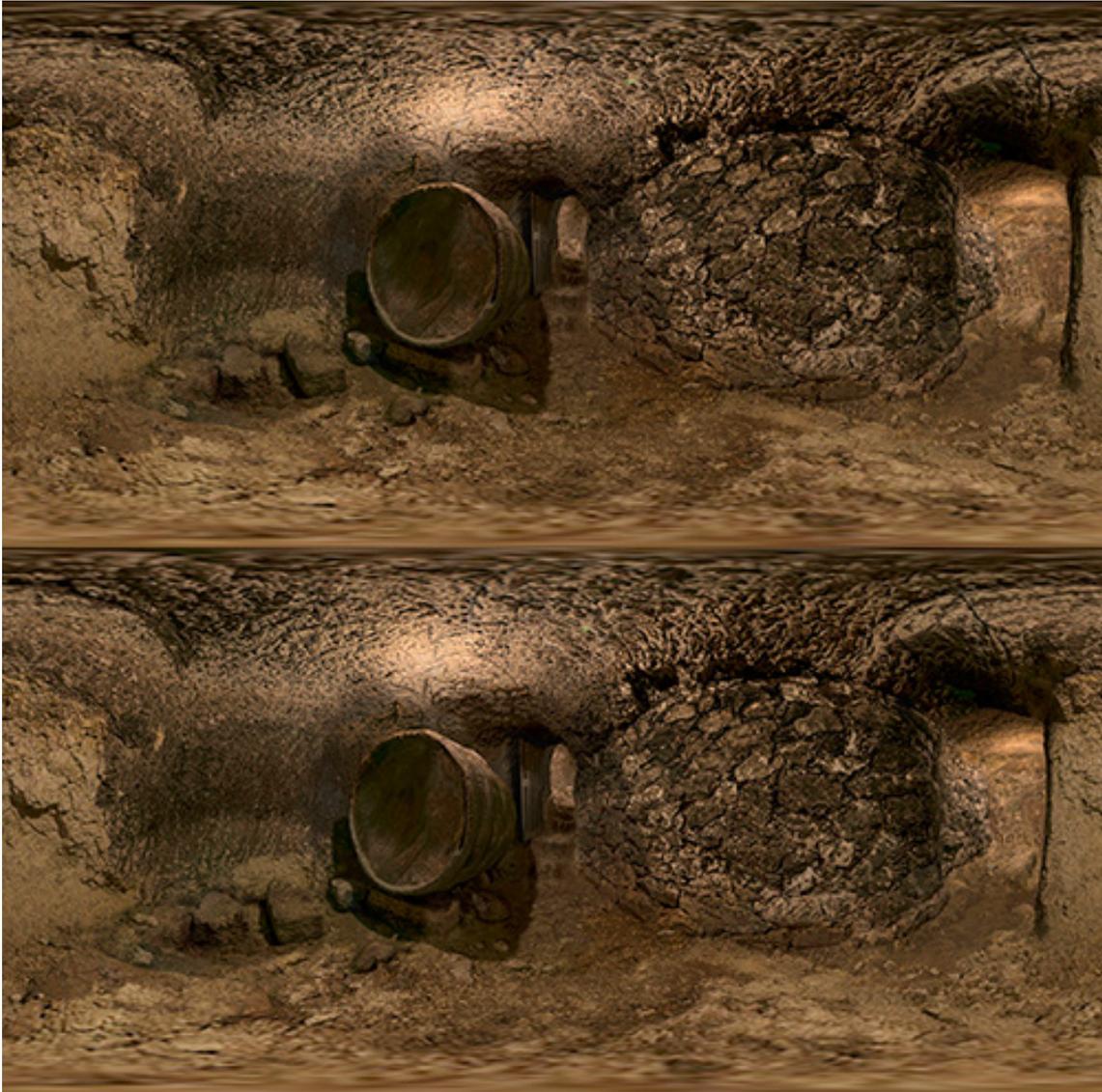


Figura 116. Par estereoscópico esférico del modelo 3D de una bodega (elaboración propia)

En esta imagen, podemos apreciar la calidad de un par estereoscópico obtenido por este método. Como se puede observar las cualidades de la imagen son muy buenas.

Los pequeños detalles se aprecian con la suficiente nitidez y las cualidades cromáticas las podemos personalizar a nuestro gusto. En este caso al ser la bodega un espacio bastante oscuro, podemos configurar la cámara para hacer más luminosa la imagen y hacer más vistosa la escena.

## 7.2.2- Métodos basados en fotografía semiesférica

Podemos obtener fotografías semiesféricas a partir de los mismos métodos utilizados para obtener fotografías esféricas basados en el cosido de imágenes, y es que realizando el cosido de las imágenes correspondientes a los 180° de la escena obtenemos este tipo de panoramas. También podemos utilizar las herramientas de renderizado para los mismos fines.

Además, hay un tipo de cámaras con una sola lente que obtiene directamente fotografías semiesféricas, muy interesantes para la obtención directa de panoramas semiesféricos.

### 7.2.2.1- Cámara réflex

La cámara utilizada para realizar las fotografías esféricas es la Canon EOS 750D, montada con un objetivo de 18mm. Con dicha cámara podemos crear un panorama esférico con 40 fotografías, por lo cual solo necesitaremos 20 para crear un panorama semiesférico.

Al importar las imágenes a nuestro software de cosido, obtendremos un resultado no muy esperado, el producto será un panorama esférico con zonas en negro, sobre las cuales no se proyectan imágenes.

Por lo cual lo que tendremos que hacer a continuación será recortar la imagen de tal forma que nos quedemos con el trozo correspondiente a una imagen semiesférica.

· **Peso de la aplicación:** respecto de sus aplicaciones análogas esféricas, deberíamos de ver una reducción en el tamaño de la aplicación, ya que el tamaño de las texturas se reduce a la mitad. La aplicación de HTC Vive, apenas experimenta reducción, el peso de la aplicación se reduce 1 MB para dar aplicaciones de 56 MB.

Para smartphone Android, la aplicación pesa unos 27 MB, esto supone una reducción del peso de la aplicación del 30%.

· **Calidad de imagen:** este método, que necesita la mitad de fotografías que su análogo, da como resultado imágenes con cierta distorsión espacial en algunas zonas de la imagen. Esto puede deberse a que el programa de cosido no puede cerrar la escena, quedando los bordes de algunas imágenes sin ninguna referencia. Por lo demás la imagen conserva muy buena calidad tanto en nitidez como en resolución.

- **Inmersividad:** este método ofrece unos resultados sorprendentes. Las deformaciones en la imagen no se aprecian desde nuestras gafas de realidad virtual y se corrigen los defectos de falta de concordancia espacial que estaban presentes en las imágenes esféricas. Estos efectos, aunque no son demasiado evidentes, facilitan la acomodación de la vista y disminuyen la sensación de mareo.
- **Esquema de disparidad:** la disparidad de las imágenes sigue la misma distribución que su análoga esférica, solo que abarcando 180°.

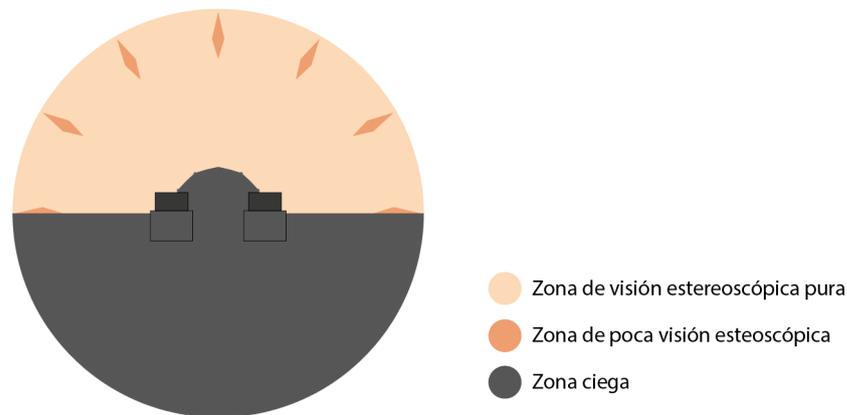


Figura 117. Esquema de la calidad estereoscópica del panorama semiesférico utilizando una cámara réflex

- **Tiempo de generación de las imágenes:** este método es algo más rápido que su análogo esférico. Al tener que sacar fotografías para los dos ojos, nos ahorramos un total de 40 fotografías por escena. Por lo demás, los procedimientos de cosido y edición llevan prácticamente el mismo tiempo.
- **Recursos utilizados:** los recursos utilizados serán los mismos que los utilizados para generar el par esférico.



Figura 118. Par estereoscópico semiesférico de la Catedral de Zamora (elaboración propia)

En la siguiente imagen, se pueden ver arriba los panoramas semiesféricos, resultado directo del cosido de las fotografías de media escena, que realmente son imágenes esféricas que tienen en negro las zonas que no han sido cubiertas.

Las imágenes resultantes son fruto de recortes de proporción 1:1, de esta forma el campo de visión recogido es de 180°. En este par, observamos la ligera deformación en los barrotes de la puerta central, que están ligeramente inclinados hacia la derecha, pero también se mejora la concordancia espacial entre los elementos de las dos imágenes.

El trípode de esta escena no ha sido eliminado, ya que no es necesario para analizar la calidad del par.

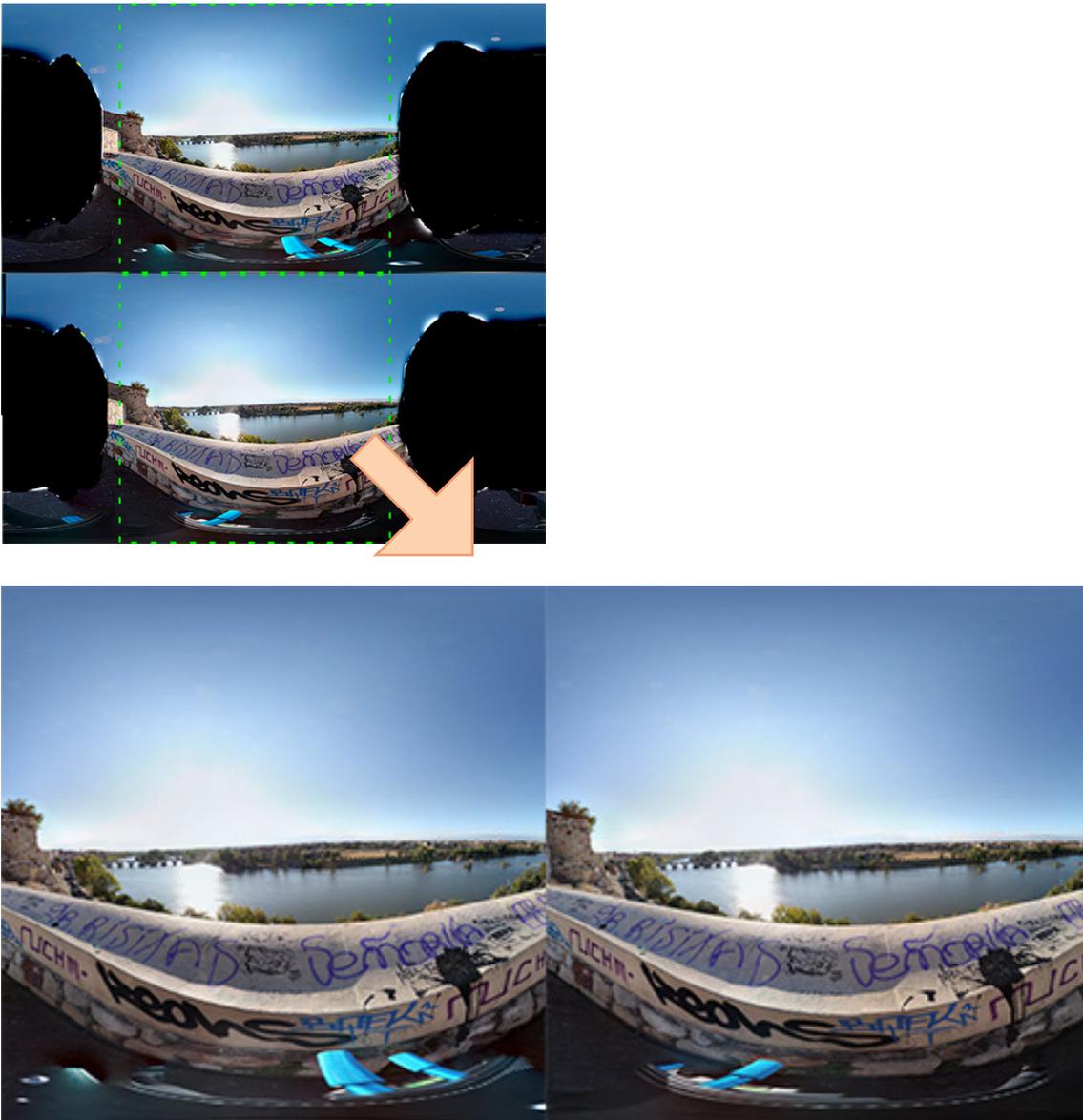


Figura 119. Par estereoscópico semiesférico del Mirador del Troncoso (Zamora) (elaboración propia)

El siguiente par ha sido obtenido a partir de los recortes de los panoramas que vemos en la parte superior.

Se observa una ligera deformación de las imágenes en el horizonte, que aparece inclinado hacia la izquierda. También se atenúan los efectos de la falta de concordancia espacial entre las imágenes del par, resultando su visualización mucho más ergonómica.

En este par tampoco ha sido eliminado el trípode ya que tampoco es necesario para analizar su calidad y ha sido eliminado en panoramas anteriores.

### 7.2.2.2- Smartphone

Se ha utilizado el iPhone 6 para generar, gracias a la aplicación Google Street View estos pares.

El método consiste en realizar solo las capturas necesarias con la aplicación de Google. Frente a las 45 fotografías que se necesitan para obtener un panorama esférico, con solo 26 podemos capturar la escena semiesférica.

La imagen resultante recogerá más de 180°, por lo que posteriormente tendremos que recortarla. Tendremos que realizar un recorte en formato 1:1 para asegurarnos que el producto del recorte es una panorámica semiesférica.

· **Peso de la aplicación:** al utilizar fotografías más pequeñas, es de esperar que también se reduzca el peso de la aplicación. Aunque para las HTC, la reducción es de solo 1 MB, lo que supone un peso total de la aplicación de 56 MB.

Para smartphone Android, la aplicación pesa unos 32 MB, que supone una reducción del peso de la aplicación del 30% aproximadamente.

· **Calidad de imagen:** utilizando este método, observamos una ligera deformación en los laterales de la imagen. Por lo demás la calidad y resolución de la imagen están al nivel del panorama esférico.

· **Inmersividad:** en la escena resultante, al estar formada por imágenes deformadas, aparecen incongruencias espaciales entre los elementos de la escena. Parece ser que la aplicación de Google, no realiza del todo bien el cosido si las imágenes no se interconectan en algún punto. Por lo cual esta aplicación, a pesar de tener buena calidad de imagen no será ergonómica, pudiendo producirse mareos.

· **Esquema de disparidad:** la disparidad de las imágenes sigue la misma distribución que su análoga esférica, solo que abarcando 180°.

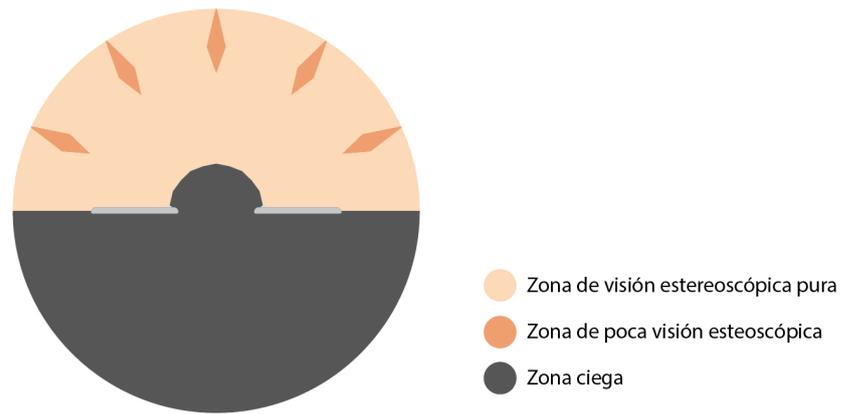


Figura 120. Esquema de la calidad estereoscópica del panorama semiesférico utilizando un smartphone

- **Tiempo de generación de las imágenes:** este método supone un ahorro de tiempo respecto al esférico, ya que necesitamos sacar 26 fotografías por panorama, frente a las 45 necesarias para construir uno esférico. Lo cual supone un ahorro de 38 capturas por par.
- **Recursos utilizados:** los recursos utilizados serán los mismos que los utilizados para generar el par esférico.



Figura 121. Par estereoscópico semiesférico del Parque del Oeste (Madrid) (elaboración propia)

En la siguiente imagen, observamos una ligera deformación en los extremos de las fotografías. Esta deformación, que es un tanto arbitraria, tiene su origen en el propio software de cosido que emplea la aplicación de Google Street View. Al no interconectarse las imágenes que van a ser solapadas, el software no puede establecer ninguna referencia en los bordes, siendo el producto del cosido un tanto aleatorio.

De esta forma, el siguiente par estereoscópico es menos ergonómico que su análogo esférico, ya que los laterales de ambas imágenes pueden presentar falta de afinidad espacial.

### 7.2.2.3- Cámara 180°

Hay cámaras, como la Kodak PixPro, que permiten grabar en 180°. Para conseguir una escena estereoscópica, tan solo hay que disponer 2 cámaras en paralelo con un soporte, como puede verse en la Figura 60. No ha sido posible disponer este tipo de cámara para este proyecto, por lo cual las imágenes utilizadas han sido descargadas de Internet, en concreto son fotogramas de vídeos de YouTube.

Debido al poco material que se encuentra de este tipo de cámaras, solo se ha analizado un par estereoscópico.

- **Peso de la aplicación:** las imágenes descargadas no son de gran tamaño. La aplicación de HTC tiene un peso de 34 MB, aunque como repetimos, para esta plataforma un peso de este orden de magnitud es insignificante.

Mientras que la aplicación de Android, tiene un peso bastante aceptable para un smartphone, unos 24 MB.

- **Calidad de imagen:** la imagen analizada no es un buen ejemplo para estudiar con detalle la calidad de esta cámara, ya que está realizada bajo unas condiciones lumínicas pésimas. Aun así, podemos apreciar que los objetos lejanos no aparecen con buena nitidez. Problema que puede estar relacionado con la falta de resolución de cada imagen, tan solo 1920x1920 píxeles. O lo que es lo mismo, menos de la cuarta parte de la densidad de píxeles de la misma imagen obtenida con la cámara réflex.

Por lo demás, las zonas de imagen que no están quemadas y no se encuentran demasiado alejadas, tienen buena calidad.

- **Inmersividad:** la sensación de inmersión, se ve altamente condicionada por la calidad de la foto. Debido a que nos encontramos ante una foto de tan mala calidad es difícil ser objetivo. Por lo demás, observamos que la percepción estereoscópica en el centro de la imagen es muy buena. Si nos desplazamos hacia los bordes, se va perdiendo esta sensación de tridimensionalidad, pero por tratarse de los bordes de la imagen, no es una zona en la que se demande esta sensación.

Además, al estar tomado el fotograma entero en una única captura, la concordancia espacial entre los elementos de ambas imágenes es perfecta. Por lo cual nos encontramos antes un par muy ergonómico para la vista, al que se le pide más resolución para lograr una sensación de inmersividad mayor.

- **Esquema de disparidad:** la disparidad es el mayor hándicap de utilizar esta cámara, a medida que nos alejamos del centro de la imagen vamos perdiendo progresivamente tridimensionalidad.

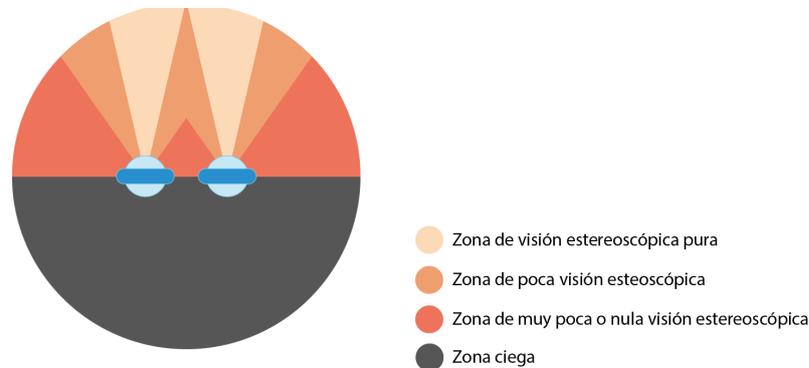


Figura 122. Esquema de la calidad estereoscópica del panorama semiesférico utilizando una cámara 180°

- **Tiempo de generación de las imágenes:** podemos utilizar tanto dos cámaras de forma simultánea tanto una sola para la obtención de este tipo de fotografía. El tiempo de generación entre uno y otro método es parecido, la diferencia puede ser de unos segundos, ya que con una sola cámara necesitamos realizar una fotografía extra. Por lo cual, estamos ante un método muy rápido.

A esto hay que sumarle el tiempo necesario para eliminar el trípode, que dependerá de la complejidad de la escena.

- **Recursos utilizados:** si utilizamos dos cámaras de forma simultánea, nos vemos obligados a construir un soporte para ambas cámaras que debe de poder anclarse al trípode. Este soporte podría fabricarse de forma barata mediante impresión 3D y su impresión rondaría los 20€.

Si por el contrario, utilizamos una sola cámara, necesitaremos anclarla a la rótula estereoscópica, que nos permitirá desplazar la cámara 65 mm entre una y otra captura.

El precio del material utilizado es de 370€ para la cámara, unos 20€ para el trípode y otros 20€ para la rótula estereoscópica. Si decidimos utilizar dos cámaras de forma simultánea, el presupuesto ascendería a los 770€.



Figura 123. Par estereoscópico semiesférico de un bosque (elaboración propia)

El siguiente par muestra una imagen de un bosque, extraída de un vídeo de YouTube. La escena ha sido obtenida mediante el uso de dos cámaras de forma simultánea, de hecho, en el centro de las imágenes podemos ver la parte de lente que cada cámara observa de la otra. Como se ha comentado, la disparidad central del par es muy buena, los objetos cercanos cobran mucha tridimensionalidad. Desgraciadamente, este efecto se pierde hasta anularse en los bordes de la imagen.

En cuanto a calidad, el espectro de color se asemeja a la realidad, fallando, como ya se ha comentado antes la nitidez de los objetos lejanos, para lo cual sería necesario que la cámara ofreciera más resolución.

#### 7.2.2.4- Renders

El procedimiento seguido para obtener un par estereoscópico semiesférico mediante técnicas de renderizado es muy similar al seguido para obtener un par esférico. La única diferencia es que tendremos que modificar una opción del RenderSetup. Anteriormente habíamos escogido una cámara esférica con un Override FOV de  $360^\circ$ , pues bien, para obtener fotografías semiesféricas, simplemente tenemos que modificar ese valor a  $180^\circ$ .

- **Peso de la aplicación:** notaremos una pequeña reducción del peso de las aplicaciones como consecuencia del menor tamaño de las imágenes. Para las gafas de HTC la aplicación tendrá un peso de unos 40 MB. Mientras que para smartphone Android, el peso de la aplicación será de 25 MB.
- **Calidad de imagen:** al tratarse de un render, que se obtiene a partir de cálculos matemáticos, este par es como si se tratase de un recorte del esférico. Por lo cual sus propiedades son las mismas.
- **Inmersividad:** la inmersividad es comparable a la de la escena esférica, salvo por el hecho de que al observar los laterales de la escena veremos el borde de la imagen. De esta forma la sensación de inmersividad irá decreciendo a medida que observemos los bordes de la escena.
- **Esquema de disparidad:** la disparidad se reparte de forma homogénea en los  $180^\circ$  de la escena.

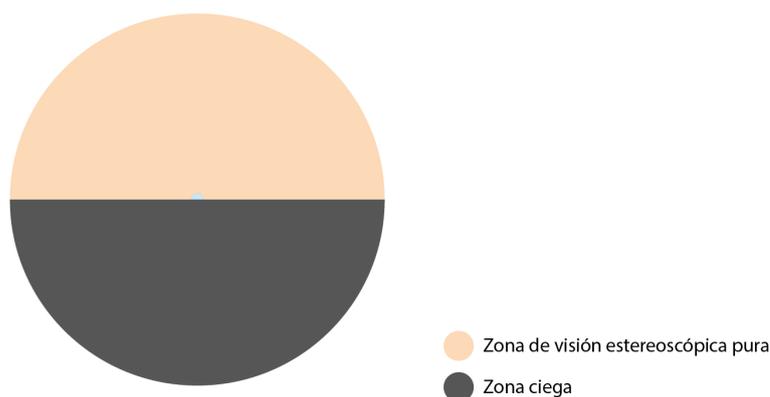


Figura 124. Esquema de la calidad estereoscópica del panorama semiesférico utilizando renders

- **Tiempo de generación de las imágenes:** debido a que las imágenes cubren la mitad de área, el ordenador tendrá que realizar la mitad de cálculos para obtener las imágenes. Por lo cual el tiempo de renderizado se reduce a la mitad.
- **Recursos utilizados:** los recursos utilizados serán los mismos que para realizar los renders esféricos.



Figura 125. Par estereoscópico semiesférico del modelo 3D de una bodega (elaboración propia)

En esta página vemos el resultado directo del proceso de renderizado en 3ds Max. Como se aprecia las características de las imágenes son idénticas a las esféricas afines.

---

### 7.2.3- Métodos basados en fotografía cilíndrica

Podemos obtener fotografías cilíndricas a través de métodos físicos basados en el cosido de imágenes, o bien a través de procesos de renderizado.

En los métodos físicos este tipo de fotografía presenta una ventaja respecto de la esférica y semiesférica, y es que el trípode no sale en la foto, por lo que supone un ahorro de tiempo.

Vamos a ver los diferentes procesos para obtener fotografías cilíndricas y a comentar sus resultados.

#### 7.2.3.1- Cámara réflex

La cámara utilizada para realizar las fotografías cilíndricas es la Canon EOS 750D, montada con un objetivo de 18mm. El procedimiento consiste en sacar fotos en el plano horizontal hasta completar la escena, se necesitan un total de 12 fotografías con esta configuración de cámara. Es importante que el objetivo no sea un gran angular, ya que si no en ese barrido, estaríamos recogiendo mucha información vertical de la escena.

A este proceso hay que sumarle un pequeño recorte de los bordes superior e inferior de la imagen, que presentan aberraciones.

· **Peso de la aplicación:** el tamaño de estas imágenes genera aplicaciones de un tamaño de 53 MB aproximadamente, para las gafas HTC Vive.

Para smartphone Android, la aplicación pesa unos 33 MB de media.

· **Calidad de imagen:** las imágenes obtenidas tienen muy buenas propiedades en cuanto a nitidez y cualidades cromáticas. Además, presentan una diferencia respecto de sus análogas esféricas, y es que las partes superior e inferior de la imagen aparecen ligeramente estiradas.

En cuanto a resolución, cada imagen es un poco diferente en función del recorte que hagamos, pero suele estar en torno a 9300x1800 píxeles. Siendo esta resolución más que suficiente para exprimir al máximo las posibilidades de Unity, que como máximo, admite texturas de 8192 píxeles.

· **Inmersividad:** de por sí, una aplicación cilíndrica es el método menos inmersivo visto hasta ahora. Es difícil sumergirse en la escena si constantemente estamos viendo los bordes de la imagen.

Por otro lado, la concordancia espacial entre los elementos de las imágenes mejora respecto de la fotografía esférica. De esta forma la aplicación es algo más ergonómica.

· **Esquema de disparidad:** la disparidad de las imágenes sigue la misma distribución que su análoga esférica.

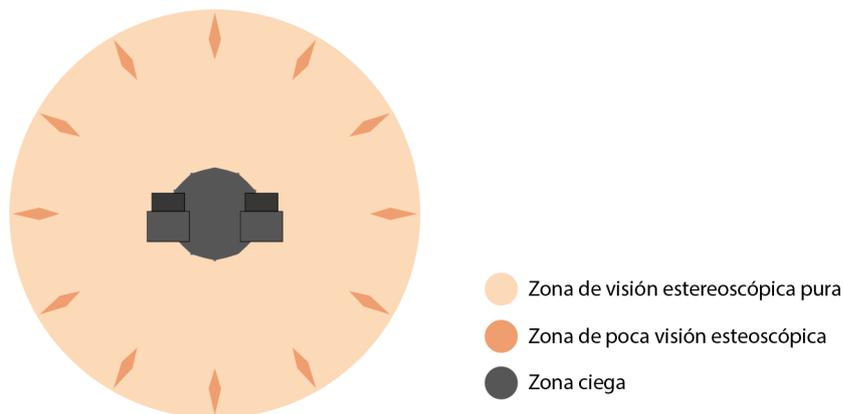


Figura 126. Esquema de la calidad estereoscópica del panorama cilíndrico utilizando una cámara réflex

· **Tiempo de generación de las imágenes:** este método ofrece la ventaja de que solo necesitamos sacar 12 fotos para cada imagen del par. Si lo comparamos con la obtención de fotogramas esféricos, donde necesitábamos 40 fotografías para cada imagen. Estamos ante un ahorro de 60 instantáneas, o lo que es lo mismo, entre 5 y 10 minutos.

La ventaja principal que presenta en este aspecto este método respecto de los anteriores, es que el trípode no aparece en la imagen, por lo cual también nos ahorramos su proceso de eliminación.

· **Recursos utilizados:** los recursos utilizados serán los mismos que los utilizados para generar el par esférico.

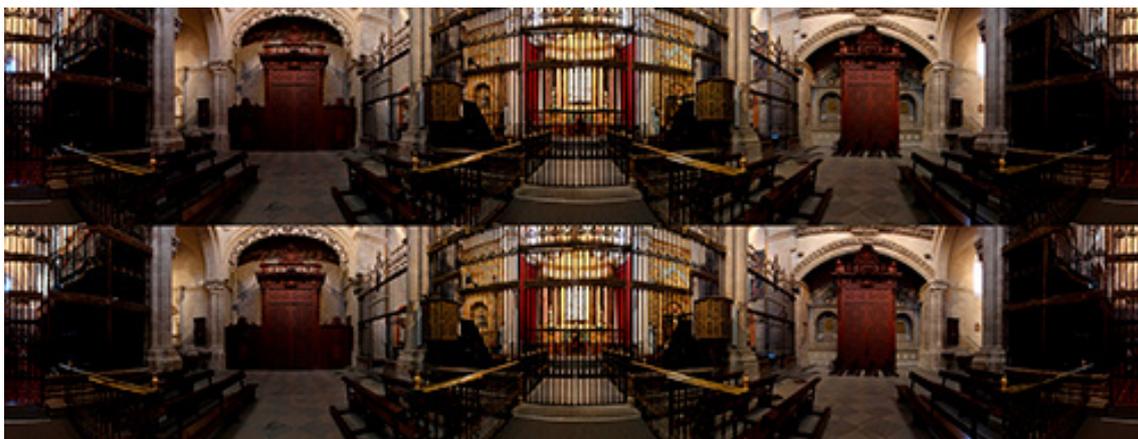


Figura 127. Par estereoscópico cilíndrico de la Catedral de Zamora (elaboración propia)



Figura 128. Par estereoscópico cilíndrico del Mirador del Troncoso (Zamora) (elaboración propia)

Vemos en esta página los pares estereoscópicos esféricos obtenidos con la cámara réflex. Si los comparamos con los panoramas esféricos no observaremos muchas diferencias a simple vista, pero si nos fijamos bien los bordes superior e inferior de estas imágenes, aparecen ligeramente más estirados.

Esto provoca que cuando visualicemos las imágenes proyectadas sobre cilindros, veamos la escena con menos deformación.

### 7.2.3.2- Smartphone

Para obtener un panorama cilíndrico utilizando la aplicación Google Street View, solo tenemos que sacar las fotografías necesarias para dicha tarea. El móvil, al encontrarse en posición apaisada, no captura mucho campo vertical, por lo que será necesario capturar fotografías en tres niveles.

De esta forma, con un total de 29 imágenes, la app de Google nos proporcionará un panorama cilíndrico. Vamos a analizar las características de la aplicación.

- **Peso de la aplicación:** las imágenes obtenidas son de gran tamaño, pero como exceden la resolución máxima que admite Unity, al final las imágenes son comprimidas. Para HTC el peso de la aplicación ronda los 53 MB.

Para smartphone Android, el peso de la aplicación suele andar por los 35 MB.

- **Calidad de imagen:** en este aspecto, podemos afirmar que las imágenes obtenidas por este método son idénticas a los panoramas esféricos de la misma escena. No hay diferencias cromáticas ni deformaciones de la escena, de hecho si superponemos las imágenes cilíndrica y esférica, es difícil apreciar las diferencias.

- **Inmersividad:** por las razones que hemos comentado, el método de obtención de las imágenes, no influirá en la calidad inmersiva de la aplicación.

Al contrario de lo que sucedía con la aplicación obtenida con la cámara réflex, la imagen cilíndrica del smartphone es algo más alta. Por lo cual, si nos esforzamos podemos evitar ver los bordes de la imagen, llegando a sentirnos inmersos en la escena.

- **Esquema de disparidad:** la disparidad de las imágenes sigue la misma distribución que su análoga esférica.

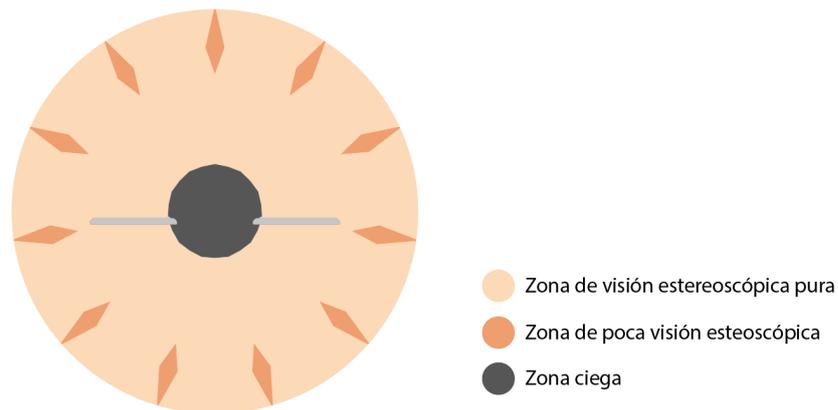


Figura 129. Esquema de la calidad estereoscópica del panorama cilíndrico utilizando un smartphone

· **Tiempo de generación de las imágenes:** si comparamos este proceso con el seguido para obtener una imagen esférica, notamos un ahorro en el tiempo de toma de fotos. El número de imágenes necesarias para obtener un panorama cilíndrico es de 29, mientras que para uno esférico de 45. Al final, el ahorro es de 32 imágenes por par.

Por lo cual este método es más eficiente. A esto hay que sumarle el tiempo de recorte de imágenes, ya que es difícil que ambas imágenes del par coincidan en altura, debido a que la inclinación del smartphone puede variar ligeramente entre el panorama correspondiente a uno y otro ojo.

· **Recursos utilizados:** los recursos utilizados serán los mismos que los utilizados para generar el par esférico.



Figura 130. Par estereoscópico cilíndrico del Parque del Oeste (Madrid) (elaboración propia)

Si comparamos el siguiente par con el esférico de la misma escena, apenas distinguiremos alguna diferencia. Las pequeñas diferencias que pueda haber estarán relacionadas con los cambios en la luz producidos durante la toma de fotos.

Como se puede apreciar ambas imágenes son de muy buena calidad y conservan la concordancia espacial de los elementos por toda la escena.

### 7.2.3.3- Renders

El proceso de renderizado para obtener el par cilíndrico, es muy similar a los procesos de renderizado vistos hasta ahora, ya que el motor de render sigue siendo el mismo.

Para configurar la vista cilíndrica de nuestra cámara, elegimos la cámara Warped Spherical dentro del menú de RenderSetup. El motivo por el que no elegimos la cámara cilíndrica es porque este tipo de cámara no nos da información relativa a las proporciones de la imagen en relación al volumen de escena representado.

La cámara Warped Spherical se comporta como una fotografía esférica, que tiene formato 2:1. Si escogemos unas proporciones de imagen 5:1, como es nuestro caso, la imagen resultante será una esférica con los polos recortados, que hace la función de imagen cilíndrica. Aunque no es un recorte exacto de la esférica anteriormente calculada, ya que este tipo de cámara que también se conoce como Warped Spherical (old- style), no realiza la proyección de la misma manera.

Una vez hemos conseguido la imagen, solo necesitamos proyectarla en Unity sobre un cilindro de proporciones afines al formato de la imagen.

Vamos a analizar las propiedades de este tipo de escena:

- **Peso de la aplicación:** los pares utilizados no son de gran formato, por lo que obtendremos aplicaciones relativamente ligeras, para HTC, obtendremos pesos de unos 40 MB.

Mientras que para smartphone Android el peso aproximado será de unos 24 MB.

- **Calidad de imagen:** la calidad de la imagen, al igual que su análoga esférica es muy buena. Su resolución de 4096x819 píxeles es suficiente para ofrecer una buena nitidez en todos los elementos.

- **Inmersividad:** como ya hemos comentado, este tipo de escenas al no abarcar mucho área, no consiguen encerrar todo nuestro área de visión. Por lo tanto, veremos la escena de forma tridimensional, pero sin experimentar una gran sensación inmersiva.

- **Esquema de disparidad:** la disparidad de las imágenes sigue la misma distribución que su análoga esférica, es decir, la disparidad será total en toda la escena.



Figura 131. Esquema de la calidad estereoscópica del panorama cilíndrico utilizando renders

- **Tiempo de generación de las imágenes:** como siempre, este tiempo depende de la potencia computacional y del tamaño de las imágenes. Al ser algo más pequeñas que las esféricas análogas, notaremos una reducción del tiempo de renderizado.

Con un ordenador medio, el tiempo difícilmente excederá los 4 minutos, para una resolución de 4096x819 píxeles.

- **Recursos utilizados:** para obtener estas imágenes será necesario disponer de un ordenador con el software oportuno instalado. El precio de un portátil básico para esta tarea parte de los 500€.



Figura 132. Par estereoscópico cilíndrico del modelo 3D de una bodega (elaboración propia)

Cada imagen del par tiene unas proporciones 5:1, aunque con otra proporciones podemos representar más porción de escena, cuanto más amplia sea la misma, más deformación notaremos en los bordes superior e inferior al visualizarla en nuestro dispositivo HMD.

## 7.2.4- Métodos basados en fotografía plana

La fotografía plana, es un método de interesante estudio, por su simpleza para la obtención de pares estereoscópicos. Ya que cualquier persona sin apenas formación puede capturar de forma relativamente sencilla un par de estas características. Con este método nació la visión estereoscópica en la fotografía, y hoy en día tenemos formas muy sencillas de implementarlo.

Vamos a hacer un repaso de los métodos para generar pares estereoscópicos planos.

### 7.2.4.1- Cámara réflex

Emplearemos la cámara utilizada durante todo el proyecto, la Canon EOS 750D, montada con su objetivo de 18 mm. El método utilizado para realizar las capturas es el más sencillo, tan solo tenemos que tomar una instantánea y mover la cámara 65 mm en el eje transversal, para sacar en esta posición la segunda instantánea. Nos podemos ayudar de la rótula estereoscópica para dotar de precisión a esta tarea.

Vamos a ver las propiedades de este tipo de entornos:

- **Peso de la aplicación:** las imágenes utilizadas recogen muy poca área de la escena, por lo cual no necesitaremos demasiada definición para poder ver la escena con nitidez. Para HTC la aplicación tiene un tamaño de unos 37 MB.

Para smartphone Android, la aplicación pesa unos 21 MB.

- **Calidad de imagen:** al no ser el resultado de ningún proceso de cosido, las imágenes tienen muy buenas propiedades. Además al ser tan rápido el proceso de toma de fotos, es más fácil que las imágenes de uno y otro ojo tengan la misma iluminación y no presenten objetos movidos.

- **Inmersividad:** debido al poco campo de visión que recoge esta escena, no se consigue que el usuario se sumerja en ella. Pero es un método interesante de ver imágenes estereoscópicas, que nos recuerda a los primeros estereoscopios.

- **Esquema de disparidad:** aunque la proyección de las imágenes es cónica y no ortogonal, podemos suponer que la disparidad se distribuye de forma homogénea por toda la imagen. Los laterales de la imagen tendrán ligeramente mermada su disparidad, aunque el efecto es despreciable.

- **Tiempo de generación de las imágenes:** las ventajas de este método es que se invierte muy poco tiempo en la toma de imágenes y no se necesita postproducción. Contando con el tiempo que se tarda en colocar el trípode y ajustar la cámara a la rótula, con 5 minutos será más que suficiente.
- **Recursos utilizados:** los recursos utilizados serán los mismos que los utilizados para generar el par esférico.



Figura 133. Par estereoscópico plano de la estatua de Viriato (Zamora) (elaboración propia)

Estas dos imágenes forman el par estereoscópico. Como se puede apreciar la calidad del par es muy buena.

### 7.2.4.2- Smartphone

Para esta ocasión utilizaremos el smartphone Xiaomi Mi4C. No se ha hecho uso de él hasta ahora, ya que no funciona de forma correcta con la aplicación GoogleStreetView. Pero para este tipo de foto no necesitaremos de ninguna función de nuestro teléfono más que la de la cámara.

Para realizar con precisión las fotografías necesitaremos un trípode y el soporte estereoscópico. El proceso consistirá en sacar una foto con el móvil a cada lado del soporte. De esta manera, tendremos 2 fotografías planas que guardan la disparidad entre sí.

- **Peso de la aplicación:** las imágenes utilizadas recogen muy poca área de la escena, por lo cual no necesitaremos demasiada definición para poder ver la escena con nitidez. Para HTC la aplicación tiene un tamaño de unos 40 MB.

Para smartphone Android, la aplicación pesa unos 23 MB.

- **Calidad de imagen:** las imágenes al obtenerse directamente presentan muy buenas cualidades, además, se eliminan los problemas relacionados con la falta de concordancia espacial vistos hasta ahora.

- **Inmersividad:** no se consigue una buena sensación de inmersividad con este método. Para lograr dicho propósito es necesario representar más escena. Por lo demás, la representación es muy buena y los objetos se aprecian tridimensionalmente.

- **Esquema de disparidad:** aunque la proyección de las imágenes es cónica y no ortogonal, podemos suponer que la disparidad se distribuye de forma homogénea por toda la imagen. Los laterales de la imagen tendrán ligeramente mermada su disparidad, aunque el efecto es despreciable.

- **Tiempo de generación de las imágenes:** las ventajas de este método son que se invierte muy poco tiempo en la toma de imágenes y no se necesita postproducción. Contando con el tiempo que se tarda en colocar el trípode y ajustar la cámara a la rótula, con 5 minutos será más que suficiente para tomar las fotografías.

- **Recursos utilizados:** se necesitará de un telefono Xiaomi Mi4C, con un precio de mercado de 150€. Además de un trípode y un soporte estereoscópico fabricado mediante impresión 3D. En total por unos 190€ podemos realizar este tipo de fotografía.



Figura 134. Par estereoscópico plano de la estatua de Viriato (Zamora) (elaboración propia)

Estas dos fotografías componen el par estereoscópico plano obtenido con el smartphone. Como se puede apreciar la cualidades de la imagen son muy buenas, aunque en cuanto a nitidez, la cámara réflex ofrece más definición, sobre todo cuando enfocamos objetos con mucha profundidad de campo.

### 7.2.4.3- Renders

Para obtener un par estereoscópico plano utilizando el motor de render V-Ray, el procedimiento es similar a los vistos anteriormente.

En primer lugar, tenemos que elegir el tipo de cámara, dentro del menú de RenderSetup, Default. De esta manera la imagen se obtendrá en base a la configuración que establezcamos de nuestra cámara.

A continuación, podemos modificar los parámetros de la cámara a nuestro antojo, como si de una cámara réflex se tratase. Se nos permite modificar el tamaño del sensor, la distancia focal del objetivo, el tiempo de exposición, la apertura del diafragma o el punto de enfoque, entre un gran número de parámetros.

Tras elegir la configuración deseada, podemos realizar el render del par estereoscópico.

- **Peso de la aplicación:** este tipo de aplicación no necesita de texturas de gran tamaño, por lo que repercutirá en el peso de las aplicaciones. El peso de la aplicación para HTC es de 37 MB.

Mientras que para smartphone Android el peso es de unos 22 MB.

- **Calidad de imagen:** la resolución de cada textura es de 2000x1333 píxeles, más que suficiente teniendo en cuenta la pequeña porción de escena que vamos a representar. Las cualidades de la imagen son las mismas que las de la fotografía esférica, toda la escena se ve con muy buena nitidez.

- **Inmersividad:** como ya hemos comentado este tipo de aplicaciones no produce mucha sensación de inmersividad. Pero ver una fotografía de forma tridimensional es siempre más espectacular que verla simplemente en 2D.

- **Esquema de disparidad:** aunque la proyección de las imágenes es cónica y no ortogonal, podemos suponer que la disparidad se distribuye de forma homogénea por toda la imagen. Los laterales de la imagen tendrán ligeramente mermada su disparidad, aunque el efecto es despreciable.

- **Tiempo de generación de las imágenes:** al ser las imágenes de un tamaño bastante reducido, el tiempo de renderizado no será muy alto.

- **Recursos utilizados:** los recursos utilizados serán los mismos que los utilizados para generar el par esférico.



Figura 135. Par estereoscópico plano del modelo 3D de una bodega (elaboración propia)

El siguiente par también presenta muy buena calidad de imagen, aunque la disparidad no es tan buena como la vista hasta hora en los pares obtenidos por procesos de renderizado. Los pares obtenidos por procesos de renderizado suelen tener mejor disparidad que los obtenidos por métodos físicos. Lo cual es fruto de la configuración del motor de renderizado, que hace prácticamente imposible obtener un fotografía con las mismas características en la realidad.

Sin embargo, este tipo de fotografía presenta la disparidad repartida de la misma forma que las obtenidas por métodos físicos.

### 7.2.5- Métodos basados en modelos 3D

Utilizar un modelo 3D directamente en Unity es una forma muy sencilla de obtener aplicaciones de realidad virtual. Tan solo tenemos que importar el modelo a nuestra biblioteca de Unity, introducirlo en la escena, e insertar una cámara estereoscópica para poder verla tridimensionalmente.

La única pega es, que dependiendo del modelo 3D que utilicemos, el peso de la aplicación puede ser excesivo para una escena de estas características.

Vamos a ver las características de una aplicación de este tipo:

- **Peso de la aplicación:** el peso de la aplicación dependerá sobre todo de la complejidad del modelo 3D utilizado. En este caso, la aplicación resultante es de unos 46 MB para HTC.

Para smartphone Android el peso es de unos 27 MB.

- **Calidad de imagen:** la definición con la que visualizamos la escena no es excesiva. La calidad de la escena en este caso, no solo depende de la calidad del modelo 3D, sino que también influye el motor de renderizado que emplea Unity. Dado que las imágenes que llegan a nuestros ojos son renderizadas a tiempo real por la aplicación, la calidad no puede ser excesiva.

- **Inmersividad:** la calidad inmersiva de la escena es excelente, la disparidad se reparte homogéneamente por todo el entorno. Por lo cual, la aplicación se asemejará a nuestra experiencia cotidiana de visión.

- **Esquema de disparidad:** la disparidad de la escena es perfecta. Al igual que los métodos de renderizado, esta aplicación ofrece unos resultados que son muy difíciles de imitar mediante procedimientos físicos.

- **Tiempo de generación de las imágenes:** las imágenes que visualizamos se renderizan a tiempo real, por lo cual no tenemos que invertir tiempo en esta tarea. Simplemente necesitamos de un modelo 3D que introducir en la escena.

- **Recursos utilizados:** para generar este tipo de escena, una vez conseguido el modelo 3D, solo hay que utilizar un ordenador para montar el escenario en Unity.

---

### 7.3- Recreación de entornos dinámicos en realidad virtual

Vamos a analizar las diferentes formas de representar entornos dinámicos con Unity, en función de los elementos de partida, bien sean vídeos estereoscópicos o un modelo 3D.

#### 7.3.1- Métodos basados en imágenes

Para conseguir representar el movimiento a través de imágenes, tenemos que proporcionar a cada ojo los fotogramas correspondientes de un vídeo estereoscópico. El proceso de recreación es similar al utilizado en entornos estáticos, pero utilizando vídeo en vez de imágenes.

Las diferencias principales residen en la forma en la que Unity trata a los vídeos, al importarlos tendremos que convertirlos en MovieClip. Esta función solo está disponible en las últimas versiones de Unity (la utilizada en este caso es la versión Unity 5.6.1p2). Una vez convertido el vídeo a MovieClip, podremos trabajar con él y aplicárselo a objetos como si de una textura se tratase, aunque esta vez no lo haremos a través de un material.

Para aplicar el vídeo, tenemos que preparar primero el objeto. Este, ha de tener como componentes un Material, un Video Player y un Audio Source, este último solo es necesario si queremos reproducir audio con el vídeo. El shader del material determinará el aspecto de la textura sobre este, el resto de opciones del material no tendrán ningún efecto sobre la proyección de la textura. El componente Video Player lo tendremos que añadir manualmente y sobre él podemos insertar el MovieClip, así como configurar algunas opciones relativas a su reproducción.

Si queremos reproducir audio, tenemos que crear el componente Audio Source y, a continuación, introducirle el AudioClip que queramos reproducir.

Vamos a analizar los diferentes tipos de entornos que podemos representar en función del campo de visión de sus fotogramas.

### 7.3.1.1- Vídeo esférico

El procedimiento a seguir es muy similar al utilizado en su aplicación análoga estática. En cuanto a la forma de ordenar las esferas y las cámaras, el proceso no difiere en absoluto. Como hacíamos hasta ahora, establecemos cada esfera en una capa, que serán observadas alternamente por cada cámara. Por último, activamos un solo ojo dentro de cada cámara, de tal forma que ambas cámaras se comporten como una sola, dirigiendo una imagen a cada ojo.

Para introducir vídeo, tenemos que preparar las esferas, añadimos los componentes Video Player en ambas, y si queremos introducir audio, añadimos en una sola el componente Audio Source. A continuación, introducimos los vídeos izquierdo y derecho. Se recomienda separar los vídeos, en vez de introducir uno solo que contenga en sus fotogramas las imágenes izquierda y derecha, de esta forma la reproducción de cada vídeo consumirá menos recursos a nivel de software. Estos vídeos, los tendremos que convertir en MovieTexture, para que Unity pueda trabajar con ellos. A continuación, insertamos dentro del Vídeo Source de cada esfera la correspondiente MovieTexture.

Con este procedimiento, conseguimos que se reproduzcan los fotogramas para cada uno de los ojos, pero nos falta el audio. Para ello, en la esfera que tiene el Audio Source, e introducimos el AudioClip dentro de este componente. Aunque, si el propio audio está contenido en el vídeo, también se puede activar su reproducción. En este caso, es necesario crear el componente Audio Source, pero solo tendremos que modificar el componente Video Player, seleccionando el objeto que contenga el audio en la opción Audio Source.

Vamos a ver los componentes de Unity que integran esta aplicación:

•**Objects:** necesitaremos 2 esferas sobre las que aplicaremos cada una de las MovieTexture, los fotogramas de cada vídeo serán vistos en verdadera magnitud desde el centro de la esfera.

•**Meshers:** la esfera que Unity trae por defecto está compuesta por muy pocos polígonos, lo que se traduce en una textura que se proyecta de forma extraña, sobre todo en los polos de la esfera. Para solucionar este problema, hay que utilizar un Mesher que modifica la esfera, dando como resultado una esfera con suficientes polígonos como para que no se distorsione la textura del vídeo. Podemos crear una esfera en 3ds Max, importarla a la biblioteca y aprovechar su mesher (se recomienda que la esfera esté integrada por un número de polígonos en torno a 2000, ya que con muchos polígonos los fotogramas se solapan mal en los polos de la esfera).

---

·**Materials:** cada esfera traerá un material por defecto, simplemente seleccionaremos el Shader que proyecte los fotogramas de la manera deseada.

·**Shaders:** aplicaremos al material un shader que invierta las normales de la superficie de la esfera para poder ver el material desde el interior.

·**MovieClip:** el vídeo estereoscópico que vamos a proyectar ha de ser importado a la biblioteca. Una vez dentro, convertimos cada vídeo a MovieClip, insertando cada uno en el Video Player de su correspondiente esfera.

·**Cameras HTC Vive:** configurar la pareja de cámaras para las gafas HTC es un proceso ligeramente más complejo. Una vez hayamos importado el paquete de assets SteamVR Plugin, podemos empezar a crear la escena, ya que necesitaremos algunos de sus componentes.

En primer lugar, arrastramos a la escena un CameraRig. A continuación, creamos 2 GameObjects vacíos que más adelante harán la función de cámaras. Añadimos a estos 3 elementos el script de SteamVR\_Camera y expandimos el script dentro de cada elemento. Introducimos los GameObjects, ahora convertidos en cámaras dentro del CameraRig y posteriormente seleccionamos una capa de visualización y el ojo de destino de cada cámara. De esta forma, conseguimos que cada cámara tenga un solo ojo activo, que apuntará a una esfera diferente.

Pero en este tipo de gafas surge otro problema: el Tracking que caracteriza a las HTC, que permite que nos desplazemos, por la escena. Como en este tipo de aplicación nos interesa permanecer estáticos en el centro de la esfera, para observar la imagen en verdadera magnitud, podemos bloquear el tracking añadiendo a cada cámara el script que utilizamos en las aplicaciones estáticas.

La función del script es anular el tracking, añadiendo a cada cámara una función inversa a su efecto, con lo que este se anula, permaneciendo en todo momento la cámara estática en el centro de la esfera.

·**Cameras Android:** configurar las cámaras para la aplicación de Android es un proceso bastante sencillo. Solamente tenemos que crear 2 cámaras y arrastrarle a cada una de ellas el prefab GvrViewerMain. De esta forma obtendremos 2 cámaras estereoscópicas. A continuación, solo tendremos que establecer en cada cámara el ojo activo y la máscara de visualización, para que cada cámara observe una imagen diferente.

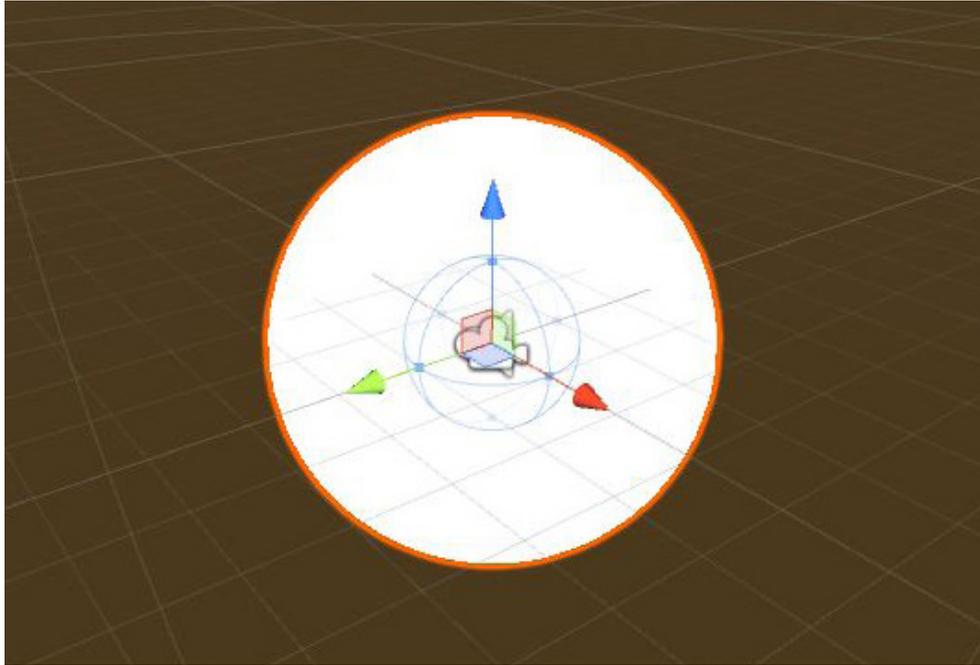


Figura 136. Elementos que integran la escena dinámica esférica

### 7.3.1.2- Vídeo semiesférico

El procedimiento es el idéntico al que hemos descrito para representar escenas dinámicas 360°, pero sustituyendo las esferas por semiesferas. Las cámaras las colocaremos en el centro de las esferas coincidentes con las semiesferas y encerraremos todos los objetos en 2 cubos negros, uno por capa, con el objetivo de no visualizar el entorno de Unity en la parte que no cubre la semiesfera.

Vamos a ver los componentes de Unity que integran esta aplicación:

- **Objects:** necesitaremos 2 semiesferas sobre las que aplicaremos cada una de las MovieTexture.

Además, necesitaremos dos cubos negros para no visualizar en entorno de Unity.

- **Meshers:** Unity no tiene ningún objeto semiesférico en su biblioteca, por lo que tendremos que apoyarnos en 3ds Max para crear una semiesfera que importaremos a la biblioteca de Unity. Para evitar la distorsión de la imagen, hay que intentar que la semiesfera esté integrada por unos 1000 polígonos.

· **Materials:** cada esfera traerá un material por defecto, simplemente seleccionaremos el Shader del material que mejor proyecte los fotogramas.

Además, para los cubos utilizaremos un material con el shader que invierte las normales de los objetos, modificando el código del shader para que sea negro.

· **Shaders:** aplicaremos a los materiales un shader que invierta las normales de la superficie de los objetos para poder ver el material desde el interior.

· **MovieClip:** el vídeo estereoscópico que vamos a proyectar, ha de ser importado a la biblioteca. Una vez dentro, convertimos cada vídeo a MovieClip, insertando cada uno en el Video Player de su correspondiente semiesfera.

· **Cameras:** el procedimiento de creación de las cámaras es el mismo que el utilizado para crear la aplicación de fotografías esféricas.

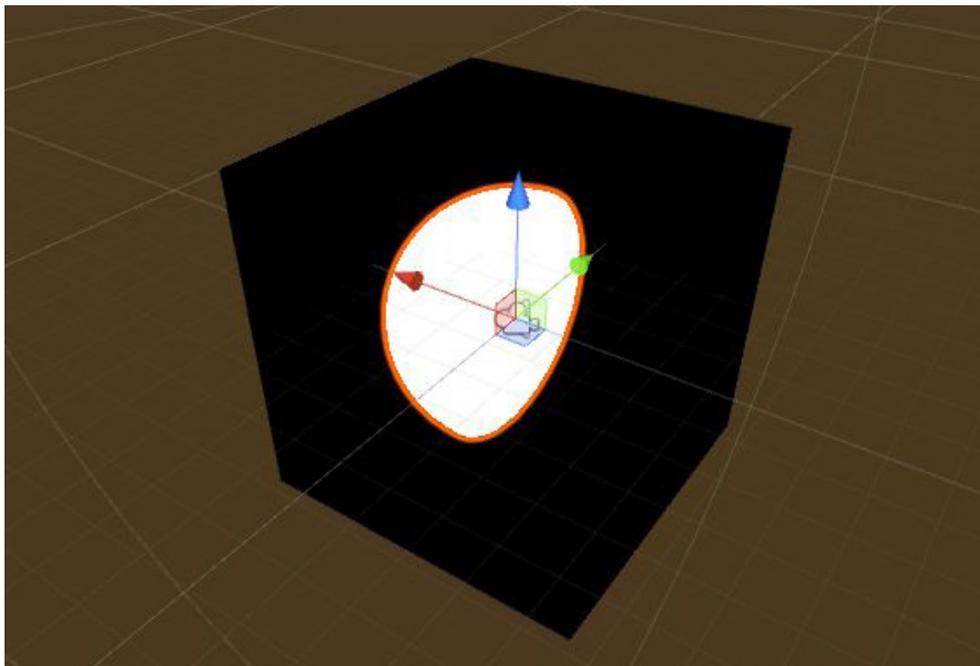


Figura 137. Elementos que integran la escena dinámica semiesférica

### 7.3.1.3- Vídeo cilíndrico

El procedimiento es idéntico al que hemos descrito para representar escenas dinámicas 360°, pero sustituyendo las esferas por cilindros. Las cámaras las colocaremos en el centro de los cilindros y encerraremos todos los objetos en 2 cubos negros, uno por capa, con el objetivo de no visualizar el entorno de Unity en la parte que no cubre la semiesfera.

Vamos a ver los componentes de Unity que integran esta aplicación:

- **Objects:** necesitaremos 2 cilindros sobre los que aplicaremos cada una de las MovieTexture, los fotogramas de cada vídeo serán vistos en verdadera magnitud desde el centro del cilindro.

Además, encerraremos todos los elementos de la escena en dos cilindros negros para no visualizar el entorno de Unity.

- **Meshers:** Unity no tiene ningún cilindro con los extremos abiertos en su biblioteca, por lo que tendremos que apoyarnos en 3ds Max para crear dicho cilindro e importarlo a Unity. Para visualizar de forma suavizada los bordes de la imagen, crearemos un cilindro de 200 lados, que es el mayor número que permite 3ds Max. De esta forma, las aristas del cilindro serán menos angulosas.

- **Materials:** cada esfera traerá un material por defecto, simplemente seleccionaremos el Shader que proyecte los fotogramas de la manera deseada.

Además, para los cubos utilizaremos un material con el shader que invierte las normales de los objetos, modificando el código del shader para que sea negro.

- **Shaders:** aplicaremos a los materiales un shader que invierta las normales de la superficie de los objetos para poder ver el material desde el interior.

- **MovieClip:** el vídeo estereoscópico que vamos a proyectar, ha de ser importado a la biblioteca. Una vez dentro, convertimos cada vídeo a MovieClip, insertando cada uno en el Video Player de su correspondiente cilindro.

- **Cameras:** el procedimiento de creación de las cámaras es el mismo que el utilizado para crear la aplicación de fotografías esféricas.

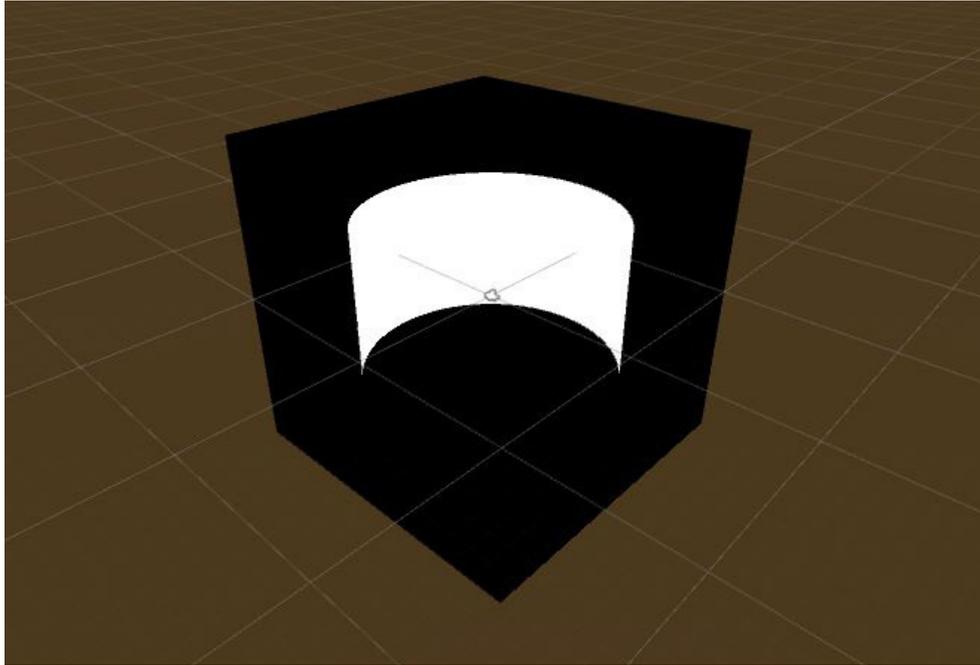


Figura 138. Elementos que integran la escena dinámica cilíndrica

#### 7.3.1.4- Vídeo plano

Para generar vídeo plano debemos seguir un procedimiento similar a los vistos hasta ahora. Una vez tengamos los vídeos convertidos a MovieTexture, los proyectamos en planos que coincidan con las proporciones de sus fotogramas.

Las cámaras las colaremos en la perpendicular al centro a una distancia óptima para su visualización. Y por último, encerraremos todos los componentes de la escena en dos cubos negros, de esta forma no visualizaremos el entorno de Unity.

Vamos a ver qué componentes de Unity integran este tipo de aplicación:

- **Objects:** necesitaremos 2 planos, de las proporciones correspondientes a los fotogramas, sobre las que aplicaremos cada una de las MovieTexture. Los fotogramas de cada vídeo serán vistos en verdadera magnitud desde la perpendicular al centro del plano.

Además, se necesitarán dos cubos, uno por capa, para encerrar todos los componentes de la escena.

- **Meshers:** utilizaremos el Mesher que los planos de Unity traen por defecto. De esta forma es muy fácil ajustar las dimensiones del plano, sin tener que recurrir a software externo.

- **Materials:** cada esfera traerá un material por defecto, simplemente seleccionaremos el Shader que proyecte los fotogramas de la manera deseada.

Además, para los cubos, utilizaremos un material con el shader que invierte las normales de los objetos, modificando el código del shader para que sea negro.

- **Shaders:** aplicaremos a los materiales un shader que invierta las normales de la superficie de los objetos para poder ver el material desde el interior.

- **MovieClip:** el vídeo estereoscópico que vamos a proyectar, ha de ser importado a la biblioteca. Una vez dentro, convertimos cada vídeo a MovieClip, insertando cada uno en el Video Player de su correspondiente plano.

- **Cameras:** el procedimiento de creación de las cámaras es el mismo que el utilizado para crear la aplicación de fotografías esféricas.

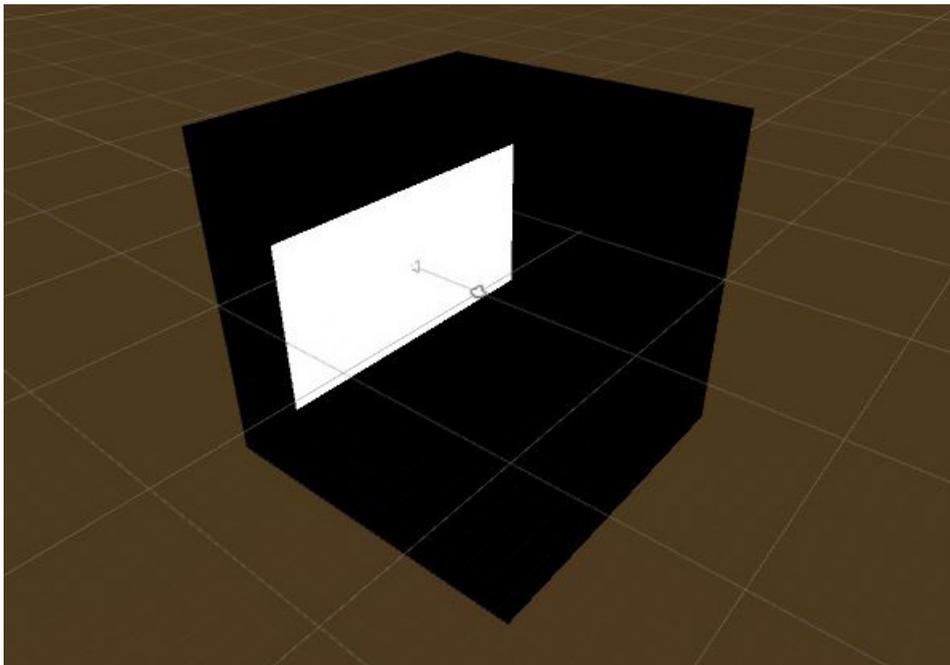


Figura 139. Elementos que integran la escena dinámica plana

### 7.3.2- Métodos basados en modelos 3D

Podemos utilizar un modelo 3D en Unity para crear una aplicación con movimiento. Es una forma muy eficiente de crear una aplicación dinámica, ya que no hace falta importar un vídeo que tiene mayor o menor peso en función de su duración. De esta forma, los fotogramas se renderizan a tiempo real, por lo cual se reduce notablemente el peso de la aplicación.

Para generar este tipo de aplicaciones, se necesita importar a Unity un modelo 3D con sus texturas. Para hacerlo, no hay más que arrastrar a la carpeta de Assets, dentro del archivo de Unity, la carpeta que incluya el modelo 3D y sus texturas. A continuación, creamos nuestra cámara de realidad virtual para HTC o Android, que ya podemos animar. La cámara de HTC se crea importando a la escena el CameraRig que encontramos en Prefabs. Para Android, tenemos que introducir dentro de una cámara de Unity el GvRViewerMain que se encuentra en la carpeta de Prefabs. Es imprescindible introducir la cámara dentro de un GameObject vacío, de lo contrario Unity entrará en conflicto al realizar la animación. A continuación, fijamos en el mismo punto del espacio la cámara y el GameObject y comenzamos el proceso de animación.

Unity tiene un motor de animación que se despliega en el menú de Window. A continuación, señalamos el objeto que queremos animar, en este caso el GameObject y creamos una animación que se guardará en la carpeta de Assets. Una vez creada la animación, aparecerá una línea de tiempo, en la que podemos editar la posición de nuestro objeto en cada instante. Es muy importante que la rotación de la cámara permanezca intacta, ya que, de lo contrario, al visualizar la escena a través de la cámara, se producirán giros involuntarios, que pueden provocar una sensación de mareo en el usuario.

Cuando hayamos ajustado el trayecto de la cámara, podemos exportar la escena a nuestra plataforma correspondiente.

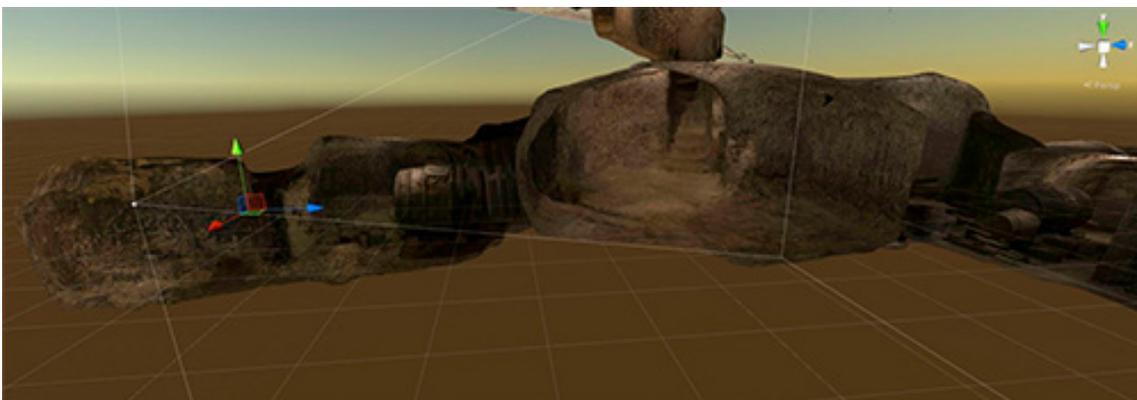


Figura 140. Aspecto del objeto 3D en la escena de Unity

## 7.4- Análisis de métodos de recreación de entornos dinámicos

Vamos a proceder al análisis de las aplicaciones creadas siguiendo estos métodos. Se clasificarán según su origen, bien sea vídeo estereoscópico o un modelo 3D.

Vamos a analizar diferentes factores dentro de cada aplicación, como se puede ver, serán los mismos que en las aplicaciones de entornos estáticos:

- **Peso de la aplicación:** puede ser un factor limitante dependiendo del dispositivo que utilicemos para abrirla.
- **Calidad de imagen:** está relacionada con la nitidez, exposición y color de la imagen. Dependerá del dispositivo que utilicemos para fotografiar la escena.
- **Inmersividad:** lo fiel que la aplicación es a la realidad. Cuanto más fiel sea la representación, la experiencia de usuario será más real. En este punto intervienen factores como la distorsión de las imágenes o la disparidad del par. Ya que, si hay distorsión, nuestro cerebro no podrá mezclar las imágenes de forma correcta y sin disparidad no veremos tridimensionalmente.
- **Esquema de disparidad:** no todos los puntos del par estereoscópico conservan la misma disparidad. Esto ocurre porque la escena está formada por diferentes fotografías cuya perspectiva es cónica, por lo cual, solamente la vertical central de estas parejas de fotografías conservará la disparidad deseada. A medida que nos alejemos a los extremos de la foto, la disparidad irá disminuyendo. Por consiguiente, cuantas más fotos integren el panorama, mayor será la disparidad general del par estereoscópico. Estudiaremos mediante un esquema, como se reparte la disparidad a lo largo de la imagen.
- **Tiempo de generación del vídeo:** salvo cuando utilicemos directamente un modelo 3D para representar un entorno, necesitaremos vídeo estereoscópico para representar el entorno deseado. Como hemos visto, hay multitud de dispositivos con los que podemos capturar un par estereoscópico. Para generar este tipo de vídeo, los dispositivos utilizados no hacen uso de demasiadas cámaras, por lo que la tarea de cosido no será muy compleja.
- **Recursos utilizados:** cada método de obtención de imágenes utiliza un dispositivo diferente, cuyo costo ha de tenerse en cuenta a la hora de realizar un proyecto.

---

## 7.4.1- Métodos basados en vídeo esférico

No hay muchos dispositivos que permitan obtener vídeo estereoscópico. Vamos a estudiar de qué formas lo podemos obtener, tanto utilizando métodos físicos como digitales, analizando las cualidades de cada dispositivo.

### 7.4.1.1- Vuze camera

La Vuze camera es el único dispositivo comercializado que por sí mismo permite capturar vídeo estereoscópico 3D. Dado que no se ha podido disponer de la cámara, procederemos a descargar estos vídeos de YouTube para su estudio.

Analizaremos tres tipos de escenas, aquella en la que la cámara permanece estática, en la que la cámara se traslada sin rotación y en la que la cámara avanza y rota indistintamente.

Es importante conocer la ergonomía de cada uno de estos tres supuestos, ya que las sensaciones, como veremos, serán muy diferentes en cada uno de los casos.

- **Peso de la aplicación:** la duración del vídeo marcará el peso de la aplicación. Para un vídeo de una duración de 48 segundos, obtenemos aplicaciones de 158 MB para HTC.

Para Android el peso de la aplicación es de unos 145 MB, lo cual excede el tamaño medio de aplicaciones y es un aspecto limitante para dispositivos con poca memoria.

- **Calidad de imagen:** dado que los vídeos han sido descargados de YouTube, la resolución sigue siendo la misma que la de los fotogramas anteriormente extraídos, es decir de 3840x1080 píxeles para cada ojo. Como ya hemos comentado no es una resolución suficiente para representar los detalles más pequeños.

- **Inmersividad:** a pesar de que la calidad ofrecida no proporciona excelentes sensaciones, el hecho de sumarle movimiento a la escena nos sumerge más en ella. Aunque no se puede generalizar, ya que los vídeos que han sido tomados rotando la cámara, no producen buenos resultados.

- **Esquema de disparidad:** con tan solo cuatro puntos de vista en torno al plano horizontal, la disparidad no se distribuye de forma uniforme por la escena. No es algo que se aprecie en exceso, pero si agudizamos los sentidos observamos que las zonas de cosido tienen algo menos de disparidad.

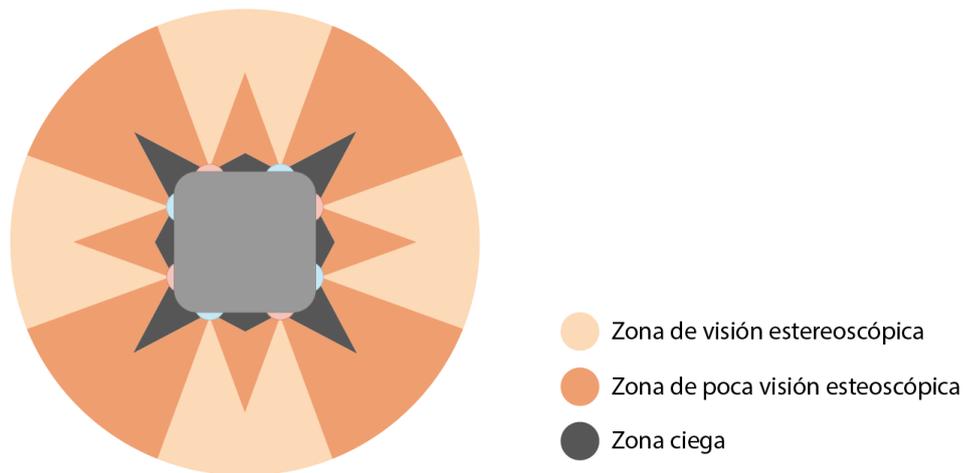


Figura 141. Esquema de la calidad estereoscópica del vídeo esférico utilizando la Vuze Camera

· **Tiempo de generación del vídeo:** la Vuze camera no tiene rival en cuanto a rapidez. Con un solo clic se inicia el proceso de grabación de vídeo estereoscópico esférico.

El cosido del vídeo se realiza de forma muy rápida con un software desarrollado por Vuze. Además, este software permite coser la zona del trípode con un parche, para eliminarlo de forma automática. Es muy interesante en vídeos con la cámara en movimiento, ya de que de la otra forma tendríamos que eliminar el trípode fotograma por fotograma.

· **Recursos utilizados:** para obtener nuestra fotografía con la Vuze camera, tan solo necesitamos un trípode y la propia cámara. El trípode no requiere desempeñar ninguna función más que la de sostener la cámara, por lo que por 20€ podemos encontrar trípodes sencillos que realicen la función. La mayor pega es el precio de la cámara, 999€.

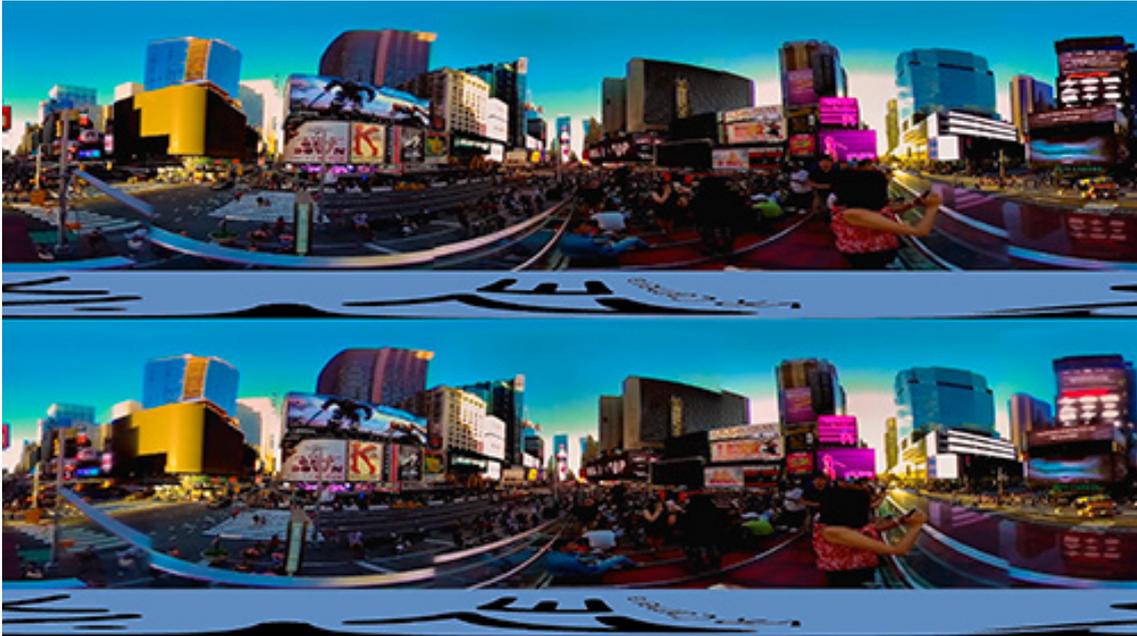


Figura 142. Fotograma de la escena con la cámara estática (elaboración propia)

En la siguiente escena la cámara permanece estática en todo momento. Este tipo de aplicación es muy ergonómica y nos permite observar la escena como si estuviéramos allí. Es más inmersiva que la misma escena estática, pero se echa de menos poder desplazarse para ver todos los objetos desde otra perspectiva.



Figura 143. Fotograma de la escena en la que la cámara sigue una trayectoria lineal (elaboración propia)

En esta ocasión, la cámara se desplaza a lo largo de una línea recta sin experimentar rotación. Esta aplicación es muy ergonómica y al desplazarse la cámara, es más fácil para el usuario sumergirse en la escena. Es por ello que las sensaciones que producen este tipo de aplicaciones son muy positivas, haciéndonos sentir como si viajáramos por la escena encima de una cinta transportadora.



Figura 144. Fotograma de la escena en la que la cámara sigue una trayectoria libre (elaboración propia)

Esta aplicación es la que peores resultados produce. Al producirse giros que el usuario no controla, aparece la sensación de mareo. De hecho, el vídeo, que dura poco más de un minuto se hace difícil de ver hasta el final. Por lo cual las sensaciones que se experimentan no son para nada satisfactorias y no se recomienda utilizar este tipo de capturas.

### 7.4.1.2- Cámara 360°

Como ya vimos en el caso de la fotografía esférica, con dos cámaras 360° se puede construir un par estereoscópico. Por lo tanto, también se puede capturar vídeo estereoscópico.

Como no ha sido posible disponer de dos cámaras 360°, tenemos que recurrir a vídeos de YouTube, grabados siguiendo esta metodología. El vídeo descargado ha sido capturado utilizando cuatro cámaras 180°, aunque el resultado es el mismo que si se hubieran utilizado solamente dos cámaras 360°.

Vamos a analizar las características de esta aplicación:

- **Peso de la aplicación:** en comparación con la Vuze, los fotogramas son más grandes, por lo que el tamaño de la aplicación será aún más elevado. El vídeo utilizado tiene una duración de 1 minuto y 36 segundos, que origina una aplicación de 256 MB para HTC.

Mientras que para Android es de 253 MB, lo cual es excesivo para este tipo de plataforma.

- **Calidad de imagen:** la calidad es bastante buena para la escena que se representa. Al tratarse de un interior sin objetos demasiado alejados, la escena es bastante fiel a nuestra experiencia cotidiana. Habría que analizar otro tipo de escena para poder realizar comentarios más objetivos.

- **Inmersividad:** a pesar de que la calidad de imagen es relativamente buena, las cualidades de este tipo de aplicación no son demasiado ergonómicas.

Por un lado, el esquema de disparidad nos muestra el desequilibrio de esta aplicación en su efecto tridimensional. Por otro, se ven las lentes de las cámaras adyacentes en las zonas de cosido. Esto provoca que mirar estos puntos produzca una sensación cuanto menos desagradable.

Por lo demás, las zonas frontal y trasera de la imagen presentan buenas propiedades y se presencia el efecto estereoscópico.

- **Esquema de disparidad:** este tipo de cámaras presenta un problema, y es que los escasos objetivos implicados en la captura de la escena, dos por panorama, la disparidad pasará de ser nula en algunas zonas a completa en otras. Por lo cual, nos encontramos ante un esquema que representa una repartición muy desigual.

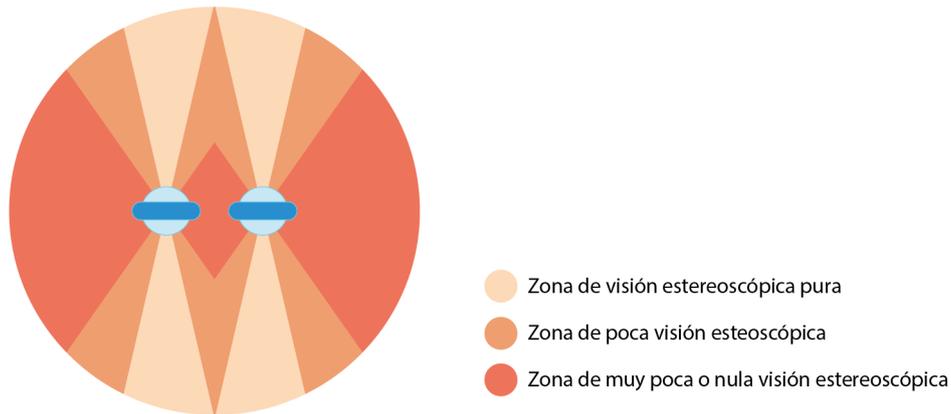


Figura 145. Esquema de la calidad estereoscópica del vídeo esférico utilizando una cámara 360°

· **Tiempo de generación del vídeo:** al tratarse de cuatro cámaras independientes, tendremos que sincronizar los cuatro vídeos de forma manual. El solapamiento lo podemos realizar en un software específico de la PixPro o en otro software de edición de vídeo. Este método es algo más laborioso que utilizar la Vuze camera, pero aun así no es complejo y su tiempo de ejecución es escaso. En menos de 10 minutos podemos tener listo un vídeo.

Si por el contrario utilizamos dos cámaras 360°, el proceso es algo diferente, aunque el tiempo de ejecución sería prácticamente el mismo. En primer lugar, tenemos que sincronizar los dos vídeos, recortarlos y posteriormente coserlos con sus partes correspondientes.

· **Recursos utilizados:** en este caso en concreto, se han utilizado para capturar el vídeo 4 cámaras Kodak PixPro, con un precio por unidad de 370€. Estas cámaras han de estar dispuestas de la forma correcta, para ello, podemos utilizar un soporte impreso en 3D que podría rondar los 20€. Además, para dotarlas de estabilidad, necesitamos un trípode que se podrá adquirir por 20€.

Esto hace un total de 1520€, un precio desorbitado teniendo en cuenta que hay propuestas en el mercado que ofrecen mejor rendimiento, como la Vuze camera. La cual, es más sencilla de utilizar, es más rápida y es más asequible.

También podemos utilizar dos cámaras 360° ancladas a un soporte. Teniendo en cuenta que una cámara Ricoh Theta ronda los 240€, por unos 520€ de presupuesto podemos realizar prácticamente la misma aplicación.

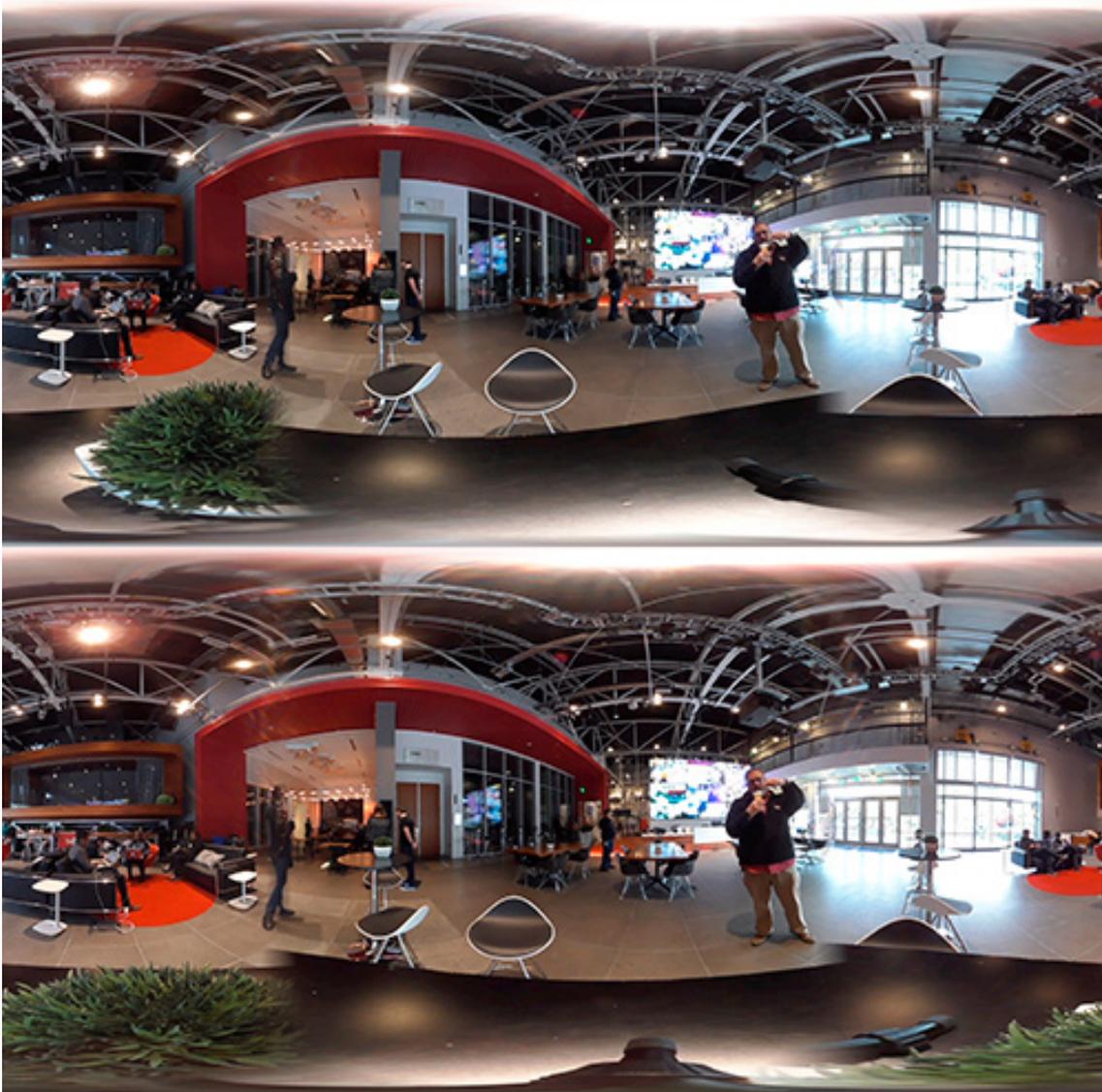


Figura 146. Fotograma del vídeo esférico utilizando una cámara 360° estática (elaboración propia)

En la siguiente escena, exceptuando las zonas de cosido de imagen, aparecen bien representados los elementos. Además, en estos puntos de cosido podemos apreciar las lentes de las cámaras adyacentes, aunque no son muy visibles en estas imágenes. Al visualizar esta zona con las gafas se produce una sensación de mareo, ya que estas lentes se encuentran en primer plano y no guardan una simetría entre los dos ojos.

Por otro lado, la mala repartición del efecto estereoscópico provoca que no sea muy ergonómico visualizar la escena, sobre todo si realizamos giros bruscos.

Sin embargo, la resolución del par es suficiente para representar con nitidez la totalidad del entorno.

### 7.4.1.3- Renders

El uso de renders para la generación de vídeo esférico estereoscópico nos expande las posibilidades frente a los procedimientos físicos. Por un lado, la calidad estereoscópica alcanzada por este procedimiento digital es muy superior a la de los métodos físicos y por otro, la precisión de la trayectoria de la cámara en el entorno virtual no se puede lograr con la misma facilidad de forma práctica.

Estas cualidades convierten a los renders en una herramienta muy sencilla y precisa para la creación de vídeo estereoscópico. La mayor desventaja que presenta el uso de esta tecnología, es el tiempo que se tarda en renderizar cada uno de los fotogramas del vídeo. Por ello es conveniente optimizar todos los parámetros de la escena, con el objetivo de reducir al mínimo el tiempo de renderizado de cada fotograma. Hay que tener en cuenta que un ahorro de 2 segundos por fotograma en un vídeo de 1 minuto, supone un ahorro de 1 hora en el proceso global.

Para conseguir una reducción significativa en el tiempo de cada fotograma, utilizaremos una configuración del motor de renderizado distinta. En primer lugar, reduciremos el tamaño de los fotogramas y en segundo lugar, cambiaremos la forma de iluminar la escena. La escena la iluminaremos con una AmbientLight, en vez de utilizar materiales con iluminación propia y reduciremos a la mitad el tamaño de los fotogramas.

Vamos a analizar la aplicación obtenida, en la que la cámara sigue una trayectoria con desniveles. Para mejorar la ergonomía, la cámara no rota en ningún momento.

- **Peso de la aplicación:** los fotogramas del vídeo analizado son de un tamaño bastante reducido para tratarse de vídeo esférico. Por lo cual, el tamaño de la aplicación no será excesivamente grande, unos 138 MB para un vídeo de 36 segundos HTC.

Para Android, el peso es de 125 MB.

- **Calidad de imagen:** la calidad de la imagen dependerá de la ambición de nuestro proyecto y del tiempo que estemos dispuestos a invertir en el proceso de renderizado. En este aspecto la calidad está directamente relacionada con el tiempo necesario para obtener cada fotograma.

- **Inmersividad:** gracias a la precisión que ofrece el motor de renderizado, obtenemos una escena con unas cualidades estereoscópicas excepcionales. Además el trayecto seguido por la cámara también puede ser ajustado con absoluta precisión. De esta forma, podemos adecuar al máximo la ergonomía de la escena.

- **Esquema de disparidad:** el proceso de renderizado, por su perfección, nos permite obtener escenas con disparidad total en cada fotograma.



Figura 147. Esquema de la calidad estereoscópica del vídeo esférico utilizando renders

- **Tiempo de generación del vídeo:** la duración del renderizado dependerá de la configuración del motor de render, de la resolución del vídeo y de las características de la escena. El tiempo de renderizado es muy importante, ya que necesitamos 1800 fotogramas para producir 1 minuto de vídeo. Por lo cual, moverse en torno a tiempos de 1 minuto por fotograma es demasiado, no obstante, es una tarea que se puede fraccionar y que no requiere nuestra presencia.

- **Recursos utilizados:** aunque podemos utilizar un ordenador básico que ronde los 500€, para esta tarea se recomienda el uso de máquinas con mayor potencia gráfica, que pueden rondar los 1200€. Aunque no es un requisito, simplemente una herramienta que nos permitirá ahorrar mucho tiempo.



Figura 148. Fotograma esférico del recorrido en la bodega (elaboración propia)

En la siguiente imagen, observamos los fotogramas izquierdo y derecho del vídeo esférico obtenido mediante renderizado. Respecto de la escena estática análoga, observamos ligeras diferencias en la tonalidad del entorno, ya que la configuración del renderizado ha sido diferente para ahorrar tiempo.

Por lo demás la escena es muy ergonómica, ya que la cámara no rota y posee la perfección que se le atribuye al motor de renderizado.

## 7.4.2- Métodos basados en vídeo semiesférico

Para la obtención de vídeo semiesférico no tenemos muchas opciones, solo un tipo de cámara puede proporcionarlo de forma física. Y las herramientas de renderizado, por su versatilidad, también nos permiten crear una secuencia de fotografías con estas propiedades.

Este tipo de escenas tiene una peculiaridad: la semiesfera proyecta el vídeo como si de una esfera se tratase, es decir, la mitad del vídeo se pierde. Este problema en el caso de la fotografía es fácil de solucionar, ya que modificando el tiling del material conseguimos visualizar el total de la textura.

El problema viene con que no se puede hacer lo mismo con los MovieClip, las opciones relativas a Tiling y Offset del material no tienen efecto alguno sobre la película. Es por ello que necesitamos un shader especial para poder realizar estas modificaciones:

```

1 Shader "Unlit/Tiling"
2 {
3   Properties
4   {
5     _MainTex ("Texture", 2D) = "white" {}
6     _Vector ("myVector", Vector) = (1,1,0,0)
7   }
8   SubShader
9   {
10    Tags { "RenderType"="Opaque" }
11    LOD 100
12
13    Pass
14    {
15
16      Cull Front
17
18      CGPROGRAM
19      #pragma vertex vert
20      #pragma fragment frag
21      // make fog work
22      #pragma multi_compile_fog
23
24      #include "UnityCG.cginc"
25
26      sampler2D _MainTex;
27      float4 _MainTex_ST;
28      float4 _Vector;
29
30      struct appdata
31      {
32        float4 vertex : POSITION;
33        float2 uv : TEXCOORD0;
34      };
35
36      struct v2f
37      {
38        float2 uv : TEXCOORD0;
39        UNITY_FOG_COORDS(1)
40        float4 vertex : SV_POSITION;
41      };
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Figura 149. Shader que permite aplicar tiling y offset a nuestro MovieClip

El siguiente shader añade a nuestro material un componente extra, un vector (X,Y,Z,W), donde la X e Y corresponden al tiling y la Z y W offset. Ahora los cambios que apliquemos en el siguiente vector sí tendrán efecto sobre el VídeoClip.

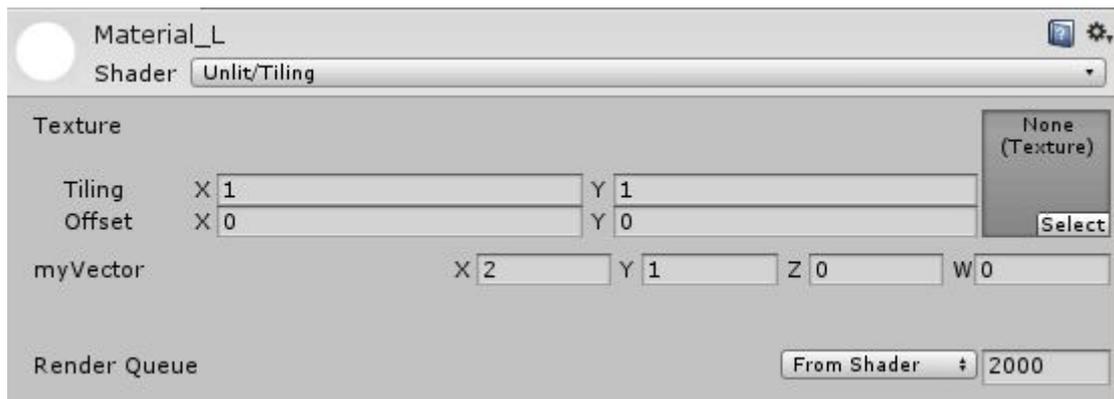


Figura 150. Opciones añadidas por el Shader a nuestro MovieClip

### 7.4.2.1- Cámara 180°

Ya hemos analizado la cámara de Kodak PixPro, que con una sola lente es capaz de realizar capturas semiesféricas de la escena. Por lo cual, si disponemos dos cámaras de forma paralela, podremos grabar vídeo estereoscópico.

Debido a que no hemos podido disponer de esta cámara para su análisis, vamos a utilizar contenido procedente de YouTube para realizar el estudio.

Se analizarán dos aplicaciones, una en la que la cámara permanece estática y otra en la que se encuentra en movimiento.

Vamos a analizar las propiedades de estas aplicaciones:

- **Peso de la aplicación:** la duración del vídeo marcará el peso de la aplicación. Para un vídeo de una duración de 8 segundos, obtendremos aplicaciones de 54 MB para HTC.

Para Android el peso de la aplicación es de unos 43 MB.

- **Calidad de imagen:** la resolución de los fotogramas, de 1920x1920 píxeles, debería ser más que suficiente para mostrar de forma correcta la escena. Aunque la realidad es que la escena no presenta buenas propiedades en parámetros como brillo, contraste o saturación. Parece ser que para una sola lente es difícil presentar en una zona tan amplia que presenta tanto zonas iluminadas como en sombra. Es por ello que aparecen zonas quemadas en la imagen que parecen deslumbrarnos.

· **Inmersividad:** la aplicación estática es la que mejores sensaciones produce en este aspecto, aunque las propiedades de imagen del vídeo, no consiguen transportarnos definitivamente a la escena.

El vídeo con la cámara en movimiento, no consigue generar una buena experiencia, en esta aplicación la cámara además de avanzar, también rota ligeramente, por lo cual si nos fijamos en un objeto, la cámara al girar cambiará la disparidad con la que representa ese punto. De esta manera nos costará ver tridimensionalmente la aplicación. Probablemente si la cámara sigue un trayecto suave, sin rotar la experiencia sea positiva.

· **Esquema de disparidad:** la disparidad en el vídeo estará repartida de tal forma que en los laterales de la escena la disparidad es nula aumentando progresivamente hasta hacerse total en el centro.

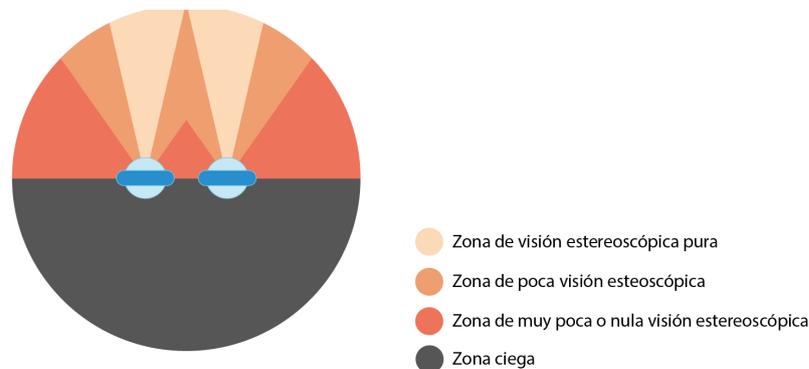


Figura 151. Esquema de la calidad estereoscópica del vídeo semiesférico utilizando una cámara 180°

· **Tiempo de generación del vídeo:** el tiempo de generación de la escena se resumiría al tiempo invertido en sincronizar los dos vídeos. Por lo cual en menos de 5 minutos deberíamos tener el material listo para representar la escena.

· **Recursos utilizados:** en este caso en concreto, se han utilizado para capturar el vídeo dos cámaras Kodak PixPro, con un precio por unidad de 370€. Estas cámaras han de estar dispuestas de la forma correcta, para ello, podemos utilizar un soporte impreso en 3D que podría rondar los 20€. Además, para dotarlas de estabilidad, necesitamos un trípode que se podrá adquirir por 20€.

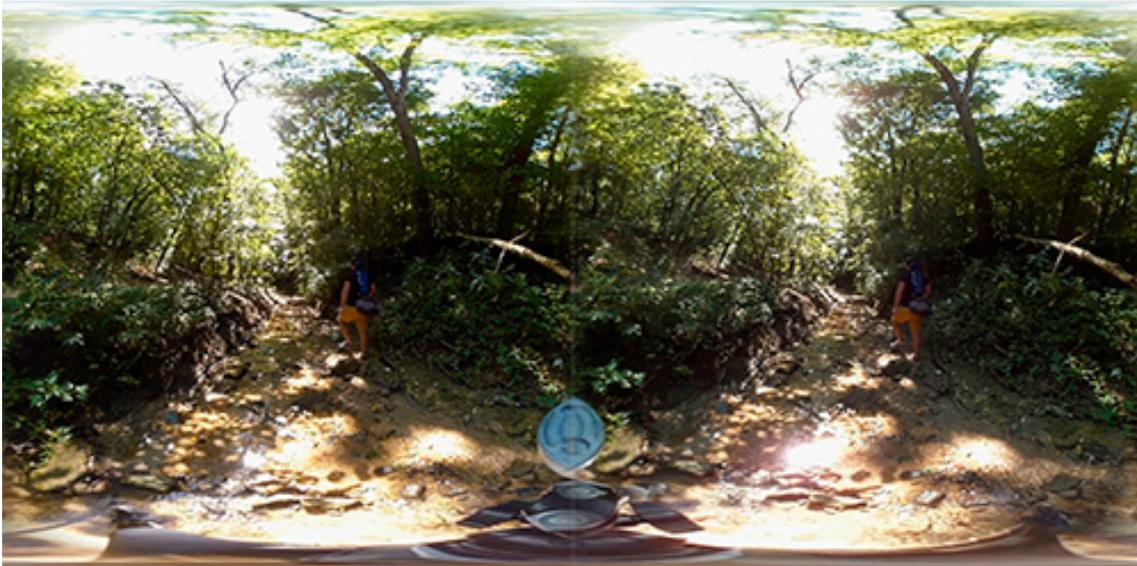


Figura 152. Fotograma de la escena en la que la cámara permanece estática (elaboración propia)

La siguiente escena ha sido grabada con la cámara estática. La escena no presenta muy buenas calidades de imagen, lo cual puede ser debido a las condiciones lumínicas de la escena. Podemos apreciar falta de nitidez general y algunas zonas quemadas, además de falta de concordancia entre las imágenes izquierda y derecha. Por lo cual la fusión de las imágenes en el cerebro va a generar zonas borrosas y la experiencia no va a ser inmersiva.



Figura 153. Fotograma de la escena en la que la cámara sigue una trayectoria lineal (elaboración propia)

Los fallos comentados en la escena anterior van a seguir presentes en esta, con la diferencia de que, al encontrarse todos los elementos de la escena en movimiento, va a ser más difícil todavía mezclar las imágenes. Los objetos cercanos se ven dobles y el fondo no logra definirse con nitidez.

Por lo cual podemos deducir, que este tipo de cámara no es adecuada para entornos con una iluminación tan irregular, ya que, al incidir la luz sobre ellas de una forma tan arbitraria, los pares de fotogramas no casan bien entre sí.

### 7.4.2.2- Renders

Vamos a utilizar la misma trayectoria utilizada con la cámara esférica para representar la escena de vídeo semiesférico. Esta es prácticamente recta por lo que se adecúa bastante bien a este tipo de cámara.

Si por el contrario la trayectoria sufriese un giro de  $180^\circ$ , la cámara, que no debe rotar en ningún momento para no marear al usuario, dejaría de enfocar al frente. En este momento la cámara estaría enfocando marcha atrás, por consiguiente, la aplicación resultaría un poco caótica.

Por lo cual, este tipo de escenas será interesante solo cuando la trayectoria sea recta. Vamos a analizar las propiedades de la aplicación:

- **Peso de la aplicación:** el vídeo será analogo al esférico, por tanto, el peso de la aplicación tampoco será muy elevado. Un vídeo de 36 segundos da como resultado aplicaciones de unos 126 MB para HTC.

Mientras que para Android, el tamaño de la aplicación se encuentra en torno a los 114 MB.

- **Calidad de imagen:** por tratarse de una trayectoria igual a la seguida en caso del vídeo esférico, con idéntica configuración de renderizado, la calidad de imagen será la misma.

- **Inmersividad:** siempre y cuando mantengamos la atención del espectador en la zona central de la imagen, la calidad inmersiva de esta aplicación será muy buena. De esta manera evitaremos ver los bordes de las imágenes, lo cual, sumado a la precisión que ofrece el motor de renderizado, da como resultado una aplicación muy ergonómica.

- **Esquema de disparidad:** el proceso de renderizado, por su perfección, nos permite obtener escenas con disparidad total en cada fotograma.

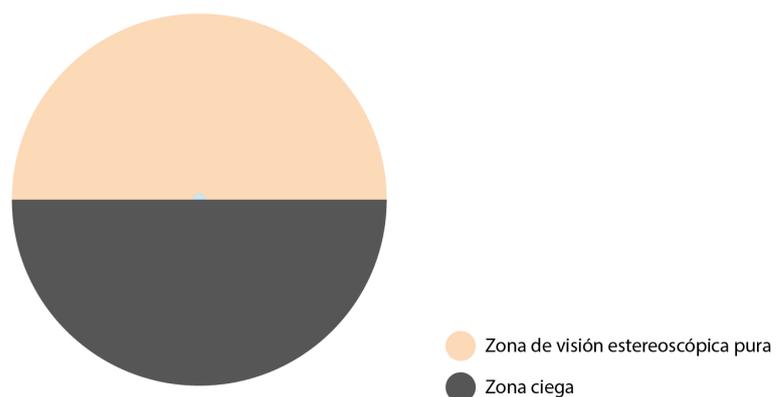


Figura 154. Esquema de la calidad estereoscópica del vídeo semiesférico utilizando renders

- **Tiempo de generación del vídeo:** por tratarse de fotogramas con la mitad de resolución que los esféricos, el tiempo de renderizado también será la mitad. Aunque el tiempo global de renderizado seguirá siendo muy alto.
- **Recursos utilizados:** igual que en el vídeo esférico, se recomienda utilizar un ordenador con alta potencia gráfica para renderizar de una forma más fluida todos fotogramas. Aunque uno básico de 500€, puede realizarlo igual.



Figura 155. Fotograma semiesférico del recorrido en la bodega (elaboración propia)

Los fotogramas son idénticos a los esféricos, aunque con la mitad de área de enfoque. Por esta razón, conservan las mismas propiedades, como si de un recorte se tratara.

Además, debido a que el sentido de avance de la cámara corresponde con la dirección de enfoque, la experiencia de usuario es muy buena.

### 7.4.3- Métodos basados en vídeo cilíndrico

Este tema no ha sido abordado por ninguna compañía. Hasta la fecha no se ha creado un dispositivo para la generación de pares de vídeo cilíndrico estereoscópico. No sería complicado disponer una serie de cámaras, de forma similar a la que utiliza la Vuze camera, pero con menor ángulo de visión, de tal forma que las zonas superior e inferior de la escena no sean capturadas. De esta manera podríamos representar vídeo cilíndrico con métodos físicos, pero debido a que no es un tipo de aplicación demasiado interesante por su poca inmersividad, solo vamos a generarlo mediante procesos de renderizado.

#### 9.4.3.1- Renders

Crear un vídeo cilíndrico con herramientas de renderizado, requiere seguir la mayoría de pasos necesarios para crear el resto de vídeos utilizando estas técnica. Para ello, animaremos la cámara de tal forma que siga el recorrido deseado, sin que llegue a rotar en ningún momento. Una vez tengamos definida la animación, configuramos la cámara de la misma forma que la configuramos para obtener los renders cilíndricos.

A continuación, solo tenemos que renderizar la escena y obtendremos un par de vídeo cilíndrico listo para importar a Unity. Vamos a ver las características de este tipo de aplicación:

- **Peso de la aplicación:** la duración del vídeo marcará el peso de la aplicación. Para un vídeo de una duración de 36 segundos, obtendremos aplicaciones de 129 MB para HTC.

Para Android, el peso de la aplicación es de unos 116 MB, lo cual excede el tamaño medio de aplicaciones y es un aspecto limitante para dispositivos con poca memoria.

- **Calidad de imagen:** la calidad de la imagen dependerá de la ambición de nuestro proyecto y del tiempo que estemos dispuestos a invertir en el proceso de renderizado. En este aspecto la calidad está directamente relacionada con el tiempo necesario para obtener cada fotograma.

- **Inmersividad:** aunque este tipo de proyección no es demasiado apropiada para generar aplicaciones inmersivas, gracias a la precisión del motor de renderizado, obtendremos una escena muy ergonómica. Si bien, el hecho de observar constantemente los bordes de la imagen, difícilmente logrará sumergirnos en el entorno.

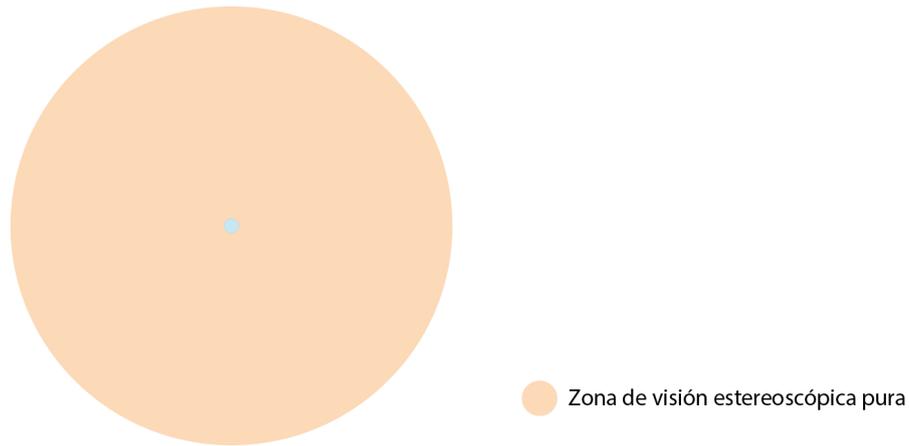


Figura 156. Esquema de la calidad estereoscópica del vídeo cilíndrico utilizando renders

- **Tiempo de generación del vídeo:** por tratarse de fotogramas con la mitad de resolución que los esféricos, el tiempo de renderizado también será la mitad. Aunque el tiempo global de renderizado seguirá siendo muy alto.
- **Recursos utilizados:** igual que para el resto de vídeos, se recomienda utilizar un ordenador con alta potencia gráfica para ahorrar tiempo. De cualquier manera, podemos utilizar un ordenador básico de 500€ para obtener vídeo cilíndrico.



Figura 157. Fotograma cilíndrico del recorrido en la bodega (elaboración propia)

Esta pareja de fotogramas cilíndricos guarda una estrecha relación con los fotogramas del vídeo esférico. En ambas tomas se realiza el mismo recorrido, solamente cambia el ángulo de visión de la cámara. Por lo cual, la aplicación resultante ofrecerá unas cualidades similares.

---

#### 7.4.4- Métodos basados en vídeo plano

El vídeo plano es el más fácil de conseguir por métodos físicos, para ello solo tenemos que colocar dos cámaras con objetivos comunes de forma paralela.

Lo complejo es conseguir una estructura que fije a cada cámara en la posición adecuada. Además, hay cámaras que por el tamaño de su carcasa no se pueden disponer lo suficientemente juntas como para que la disparidad entre ambas sea de 65 mm. Pero gracias al soporte estereoscópico para smartphone, conseguimos colocar dos teléfonos móviles a la distancia adecuada para capturar este vídeo.

Por supuesto también podremos usar las herramientas de renderizado para capturarlo.

##### 7.4.4.1- Smartphone

Crear un vídeo estereoscópico plano es un proceso relativamente sencillo. Con tan solo disponer de dos smartphones y un soporte, podemos convertir ambos dispositivos en una cámara estereoscópica. En esta ocasión se han utilizado dos smartphones Xiaomi Mi4C.

Lo más complejo es sincronizar ambos vídeos, por ello se aconseja realizar todos los clips en la misma grabación, de esta forma sincronizaremos todos los vídeos de una vez. Vamos a analizar tres escenas diferentes para estudiar la ergonomía de estas aplicaciones.

- **Peso de la aplicación:** como es de esperar, la duración del vídeo marcará el peso de la aplicación. Para un vídeo de 54 segundos de duración obtenemos un peso de aplicación para HTC de 166 MB.

Para Android la aplicación resultante ronda los 153 MB.

- **Calidad de imagen:** las cámaras de los smartphones de hoy en día tienen muy buenas prestaciones, tal es así, que los vídeos realizados con smartphones de gama media ofrecen una calidad de imagen más que suficiente como para representar la escena con realismo.

- **Inmersividad:** dependerá del tipo de escena, la aplicación con la cámara estática es la que proporciona mejores sensaciones. Por otro lado, las aplicaciones en las que la cámara rota, no generan una experiencia comfortable para el usuario. De todas formas, por las características del vídeo, que abarca muy poco campo de visión, la aplicación difícilmente logrará sumergirnos en el entorno.

- **Esquema de disparidad:** la disparidad de esta escena puede analizarse como total por el poco campo de visión que recoge, aunque realmente va disminuyendo hacia los laterales según nos alejamos del centro de la imagen.

- **Tiempo de generación del vídeo:** el tiempo de generación de la escena se resumiría al tiempo invertido en sincronizar los dos vídeos. Por lo cual, en menos de 5 minutos deberíamos tener el material listo para representar la escena.

- **Recursos utilizados:** para realizar dicho vídeo, necesitamos dos smartphones y un soporte estereoscópico para los móviles. Además, podemos emplear un trípode para estabilizar la escena.

En esta ocasión, se han utilizado dos móviles Xiaomi Mi4C. Dado que el precio de estos terminales es de 150€, el coste total del material será de aproximadamente unos 340€.



Figura 158. Fotograma de la escena en la que la cámara permanece estática (elaboración propia)

En el siguiente vídeo, donde la cámara permanece estática, la experiencia de usuario es buena. Los dos vídeos se acomodan fácilmente a la vista y se percibe el movimiento de forma correcta.

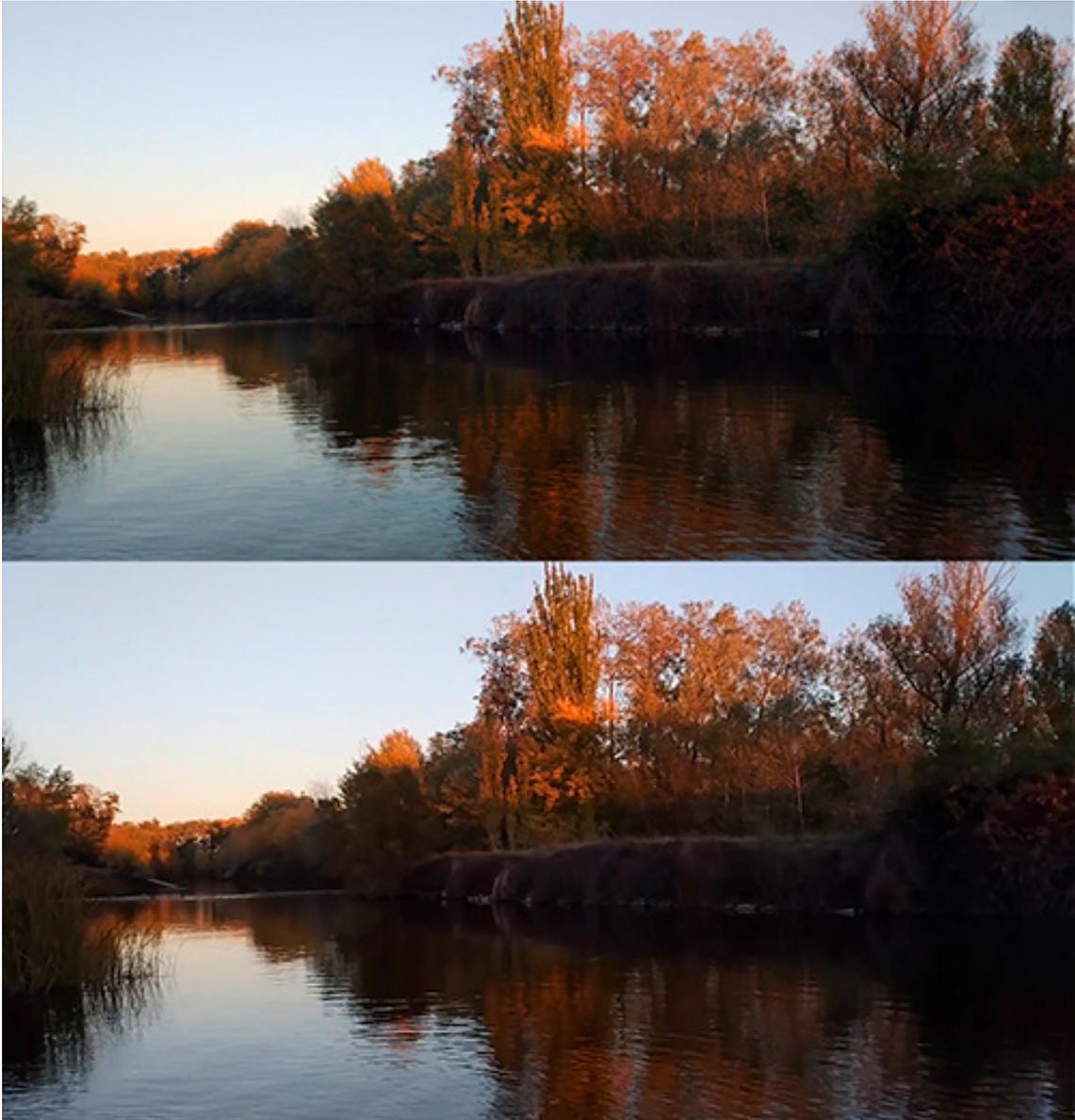


Figura 159. Fotograma de la escena en la que la cámara gira (elaboración propia)

En la siguiente escena, la cámara únicamente rota. A pesar de que lo hace lentamente, las sensaciones percibidas no son para nada satisfactorias. Por un lado el usuario puede elegir hacia donde mirar con solo mover la cabeza, sintiendo el control. Y por otro lado, el vídeo rota libremente, sintiendo que todo se mueve de forma inesperada.



Figura 160. Fotograma de la escena en la que la cámara sigue una trayectoria libre (elaboración propia)

En la siguiente escena, la cámara realiza tanto movimientos de rotación como de translación. La experiencia de usuario es todavía peor, ya que no podemos predecir de ninguna manera los movimientos de la cámara, generando una sensación intensa de mareo.

#### 7.4.4.2- Renders

Crear un vídeo estereoscópico plano utilizando herramientas de renderizado es un proceso similar al seguido para otros tipos de vídeo.

En este caso, se ha optado por que la cámara siguiese a modo de travelling, una trayectoria perpendicular a la dirección de enfoque de la cámara.

El resultado como veremos, es una aplicación ergonómica que nos muestra la escena de una forma nunca vista hasta ahora:

- **Peso de la aplicación:** la duración del vídeo marcará el peso de la aplicación. Para un vídeo de una duración de 36 segundos, obtendremos aplicaciones de 123 MB para HTC.

Para Android, el peso de la aplicación es de unos 110 MB, lo cual excede el tamaño medio de aplicaciones, y es un aspecto limitante para dispositivos con poca memoria.

- **Calidad de imagen:** la calidad de la imagen dependerá de la ambición de nuestro proyecto y del tiempo que estemos dispuestos a invertir en el proceso de renderizado. En este aspecto la calidad está directamente relacionada con el tiempo necesario para obtener cada fotograma.

- **Inmersividad:** aunque el escaso campo de visión de este tipo de vídeo no logra sumergirnos en la escena el recorrido que realiza la cámara ofrece unos resultados muy interesantes.

- **Esquema de disparidad:** la disparidad de esta escena puede analizarse como total por el poco campo de visión que recoge, aunque realmente va disminuyendo hacia los laterales según nos alejamos del centro de la imagen.

- **Tiempo de generación del vídeo:** de la misma manera que en todos los procesos de renderizado, obtener por este proceso todos los fotogramas de una secuencia, supone una inversión de tiempo bastante alta.

- **Recursos utilizados:** al igual que para renderizar otros tipos de vídeo se recomienda un ordenador con mucha potencia gráfica, aunque uno básico de 500€ puede realizar igual de bien la tarea.



Figura 161. Fotograma plano del recorrido en la bodega (elaboración propia)

En esta escena, la cámara realiza un travelling perpendicular a su dirección de enfoque. Además, el recorrido se realiza desde el exterior de la bodega, por lo cual, sería imposible conseguir esta aplicación a través de métodos físicos.

El resultado es una aplicación que, a pesar de su reducido campo de visión, ofrece una visualización de la bodega bastante interesante.

#### 7.4.4- Métodos basados en modelos 3D

Cuando se trata de representar escenas dinámicas, el uso de un modelo 3D para generar directamente el entorno supone una gran ventaja respecto de los métodos basados en imágenes.

En este caso, obtenida se genera de forma rápida y es muy ligera. En esta aplicación, la duración de la animación en la escena prácticamente no influye en su peso, lo cual supone una gran ventaja respecto de los métodos basados en vídeo. Además la aplicación resultante será de 360°, ofreciendo una experiencia más inmersiva.

- **Peso de la aplicación:** en este caso la duración de la aplicación no influye en su peso, el peso de la aplicación lo determinará la calidad del modelo 3D empleado. Con el modelo utilizado, para HTC, obtenemos aplicaciones de 46 MB.

Para Android el peso es algo más reducido, de unos 27 MB.

- **Calidad de imagen:** la calidad de la imagen no es excesiva, aunque sirve para mostrar la escena adecuadamente. En este aspecto notamos una ligera reducción en la calidad respecto de los vídeos obtenidos por renderizado, aunque la diferencia entre el peso de estas aplicaciones justifica el uso de modelos 3D.

- **Inmersividad:** en cuanto a inmersividad, esta aplicación ofrece una experiencia de calidad. El hecho de que nuestra zona de enfoque se renderice a tiempo real, no supone ningún handicap, incluso en plataformas más vulnerables como Android. Además, con esta aplicación, disfrutaremos de una experiencia ergonómica.

- **Esquema de disparidad:** la disparidad, por la naturaleza de la aplicación, estará homogéneamente repartida por toda la escena.

- **Tiempo de generación del vídeo:** dado que las imágenes que visualizamos en esta aplicación se renderizan a tiempo real, no será necesario invertir tiempo en esta tarea. Nada más que el necesario para animar la escena en Unity.

- **Recursos utilizados:** para generar este tipo de escena, una vez conseguido el modelo 3D, solo hay que utilizar un ordenador para montar el escenario en Unity.

---

## 8- CONCLUSIONES

### 8.1- Conclusiones sobre aplicaciones de entornos estáticos

Podemos representar entornos estáticos con aplicaciones basadas en la fotografía o en modelos 3D. Estas se caracterizarán por el campo de visión que recojen, pudiendo distinguir entre aplicaciones esféricas, semiesféricas, cilíndricas o planas.

Indudablemente las aplicaciones esféricas son las que mejores sensaciones producen. Gracias a que cubren todos los puntos de la escena, la experiencia del usuario es muy cercana a la de la realidad. Además, se pueden obtener por gran multitud de métodos, incluyendo los modelos 3D. Tienen como contra que son las aplicaciones más pesadas y tardan más tiempo en producirse.

Las aplicaciones semiesféricas no logran producir una experiencia tan inmersiva, ya que limitan nuestro campo de visión. Solo pueden obtenerse por métodos basados en la fotografía y su uso se recomienda para entornos en los que una zona de la escena no sea de interés.

Las aplicaciones cilíndricas solo se recomiendan para casos funcionales, en los que importe más visualizar el contenido que la calidad de visionado del mismo. Como en el caso anterior, este tipo de aplicación solo puede obtenerse a partir de imágenes. Ofrecen una visión de 360° en el plano horizontal, en el vertical el ángulo de visión varía en función del método, aunque suele rondar los 90°. El hecho de que varíe el campo de visión vertical, repercute en las proporciones de la imagen y por tanto en el cilindro sobre el cual se va a proyectar, siendo necesario crear un cilindro a medida para cada aplicación. Además, la escena no se aprecia en verdadera magnitud en los extremos superior e inferior, para ello, la imagen debería ser proyectada sobre una esfera con los polos recortados, pero es una figura mucho más compleja de obtener. Tienen como ventaja: la sencilla generación de los panoramas y la desaparición del trípode en la imagen.

Por último, las aplicaciones planas son las más sencillas de obtener. También es otro tipo de representación que solo se logra a partir de imágenes, pero que ofrece los resultados más pobres en cuanto a inmersividad. Al igual que la cilíndrica, este tipo de aplicación no muestra los bordes de la escena en verdadera magnitud, aunque no se aprecia a simple vista. La ventaja es que las aplicaciones son muy rápidas de obtener y el peso es reducido.

### 8.1.1- Aplicaciones esféricas

Es el único tipo de aplicación que se puede obtener tanto con fotografía esférica como con modelos 3D. Entre las diversas herramientas de fotografía que encontramos en el mercado, vamos a analizar como se comportan en la obtención de panoramas esféricos estereoscópicos una cámara réflex, un smartphone, una cámara 360°, la Vuze camera y un motor de renderizado como VRay.

De todos estos dispositivos, la cámara 360° ofrece los peores resultados, aunque la obtención de las imágenes se realiza de forma muy rápida. Estas no poseen la suficiente resolución como para mostrar la escena con nitidez. Además, la distribución de la disparidad en el par es muy mala, por lo que solo veremos tridimensionalmente determinadas zonas.

La Vuze camera tiene la cualidad de ser muy rápida, ya que con un solo disparo obtenemos el par estereoscópico que necesitamos. En cuanto a calidad de imagen, la nitidez es mejor que la ofrecida por la cámara 360°, aunque se echa en falta más calidad. Sus mayores desventajas son su excesivo precio, la visibilidad de las zonas de costura en algunos panoramas y su mala distribución de la disparidad.

Tanto la cámara réflex como el smartphone compiten en la misma categoría. Ambos necesitan de la realización de múltiples fotografías para la generación del par estereoscópico por cosido, por lo que ambos métodos son muy lentos. La diferencia es que el cosido de estas imágenes se realiza de forma diferente, lo cual hace que la aplicación obtenida con el smartphone sea más ergonómica. Ya que los elementos entre las imágenes de este par son más afines espacialmente que los del par de la réflex (que por su método de costura pueden presentar distorsión). Por otro lado, los panoramas ofrecidos por la réflex ofrecen mejor resolución, lo cual solo se aprovecha en escenas con elementos muy lejanos.

Resumiendo, ambas herramientas representan la escena de forma muy realista, ofreciendo imágenes con mucha resolución y la disparidad homogéneamente repartida. Aunque el smartphone destaca sobre la cámara réflex por ergonomía y ser más asequible.

El motor de renderizado, es la herramienta más versátil, ya que podemos configurar la resolución la calidad y la apariencia de la imagen. Además, la distribución de la disparidad es perfecta y la ergonomía ofrecida inigualable por los métodos físicos, en los que siempre hay posibilidad de fallo.

Por último, el uso directo de un modelo 3D en Unity, también nos ofrece una escena tridimensional en todos los sentidos. El aspecto negativo es la relación entre la calidad y el peso de la aplicación, que se ve superada por el empleo de renders y, aparte de ser más liviana, ofrece unas cualidades de imagen muy superiores.

Por lo cual, siempre y cuando sea viable obtener un modelo 3D de la escena en cuestión, será el método que mejores prestaciones ofrezca. En caso contrario, el smartphone es la herramienta por excelencia, aunque se puede justificar el uso de la Vuze camera en proyectos muy extensos o en aquellos en los cuales la rapidez sea una prioridad.

### **8.1.2- Aplicaciones semiesféricas**

Este tipo de escenas solo se puede representar mediante panoramas semiesféricos estereoscópicos. En este caso hay menos dispositivos que permitan obtener este tipo de fotografía. En cuanto a métodos físicos, tanto la cámara réflex como el smartphone permiten obtener por cosido panoramas semiesféricos, y por último las cámaras 180°, permiten obtener de forma directa estos panoramas. Por supuesto, modificando los parámetros de nuestro motor de renderizado VRay también podremos obtener de forma directa estas imágenes.

Al igual que en la escena esférica, siempre y cuando sea posible, es recomendable recurrir a los procesos de renderizado, ya que las imágenes resultantes poseen la máxima perfección. Realmente poseen las mismas características de las esféricas, como si de un recorte suyo se tratase.

Por el contrario, tanto la cámara réflex como el smartphone, presentan comportamientos un tanto diferentes. En los dos casos se produce una ligera distorsión, un tanto arbitraria, de los elementos en los laterales de la imagen. Aunque en ambos casos se mantiene la alta resolución y la homogeneidad de esta disparidad que ofrecen estos panoramas. Este tipo de escena supone una ventaja en estos métodos, y es que se necesitan de menos imágenes para realizar los panoramas, lo que se traduce en un ahorro de tiempo y del tamaño de las fotografías.

La cámara 180° podríamos compararla con la cámara 360°, tanto por su escasa calidad y disparidad como por su rapidez. En este tipo de fotografías, la disparidad disminuye progresivamente desde el centro del panorama hasta hacerse nula en los extremos.

En este tipo de escenas, por encontrarse los elementos de interés en torno a una zona en concreto, la distribución homogénea de la disparidad no tendrá tanta importancia como la tenía en la escena esférica. Por lo cual el uso de una cámara 180°, que ofrece una gran rapidez, puede estar más justificado. Por otro lado, si no queremos perder en calidad de imagen se recomienda el uso de smartphone.

### 8.1.3- Aplicaciones cilíndricas

Solamente podremos hacer uso de la fotografía cilíndrica para crear este tipo de escenas. Como ya hemos visto no ofrecen una experiencia inmersiva de calidad, aunque permiten representar de una forma rápida los 360° de la escena. En cuanto a métodos físicos, solo podremos hacer uso de la cámara réflex y del smartphone para obtener este tipo de panoramas. Además, gracias a su versatilidad, también podremos configurar el motor de renderizado para este tipo de escenas.

En este caso, los procesos de cosido de imágenes de la cámara réflex y del smartphone no tienen nada que ver. En cuanto al smartphone, las imágenes experimentan una proyección esférica parcial, las de la cámara réflex experimentan una proyección puramente cilíndrica. Estas diferencias se observan a simple vista. Las imágenes generadas por el smartphone, por ser un recorte de una imagen esférica, presentan una contracción vertical en los extremos superior e inferior. A pesar de esta deformación, que no es excesiva, es preferible el uso del smartphone, ya que, por desgracia, los problemas de afinidad espacial entre ambas fotografías del par, aunque se atenúan, siguen presentes en el caso de la réflex. Por otro lado, el tiempo de generación de estas imágenes es notablemente inferior en ambos casos respecto de la obtención de panoramas esféricos.

La herramienta de renderizado proporciona imágenes con el mismo problema que nuestro smartphone. Por lo cual, aunque la aplicación es muy ergonómica, observaremos comprimidos los extremos superior e inferior de la escena.

El uso de este tipo de aplicación, por tanto, no es muy recomendable. Aunque en caso de utilizarla, se recomienda obtener las imágenes por procesos de renderizado. En caso de que no sea posible o no compense generar un modelo 3D del entorno, tanto la cámara réflex como el smartphone están indicados. Por un lado, el smartphone ofrece una escena con mayor calidad de imagen, menor deformación y algo más rápida de obtener. Mientras que el smartphone ofrece una aplicación más ergonómica.

### 8.1.4- Aplicaciones planas

Este tipo de escena supone una forma sencilla de representar tridimensionalmente una parte del entorno. Al igual que en la cilíndrica, la podemos obtener con fotografía réflex, con un smartphone o mediante renders (aunque en esta ocasión no será necesario realizar un cosido de imágenes).

Tanto con la cámara réflex como el smartphone, solo tenemos que disponer el dispositivo en dos posiciones, de tal forma que se conserve la disparidad entre ambas capturas. Así pues, el tiempo de ejecución no cambiará, mientras que la réflex ofrece imágenes con más nitidez. Por lo cual, aunque tampoco haya demasiada diferencia entre ambos dispositivos, preferi-

blemente escogeremos la cámara réflex por su alta resolución.

Si disponemos de un modelo 3D, podemos obtener un par estereoscópico plano a nuestra medida, con un motor de render. Sin embargo, la diferencia de precisión respecto de los métodos físicos será prácticamente inapreciable. Por ello, no es tan interesante el empleo de renders en este caso como en los anteriores.

### 8.1.5- Comparativa general de aplicaciones

Esta tabla se ha creado con la intención de poder visualizar de forma gráfica las cualidades más relevantes de las aplicaciones de entornos estáticos hasta ahora analizadas:

Cualidades Aplicaciones		 Peso	 Calidad	 Inmersividad	 Estereoscopía	 Tiempo	 Presupuesto
		 Esférica	Cámara réflex	Dark Blue	Light Blue	Medium Blue	Medium Blue
Smartphone	Dark Blue		Medium Blue	Light Blue	Medium Blue	Dark Blue	Medium Blue
Cámara 360°	Medium Blue		Dark Blue	Dark Blue	Dark Blue	Medium Blue	Light Blue
Vuze camera	Light Blue		Dark Blue	Medium Blue	Dark Blue	Light Blue	Dark Blue
Renders	Dark Blue		Light Blue	Light Blue	Light Blue	Light Blue	Medium Blue
 Semi-esférica	Cámara réflex	Medium Blue	Light Blue	Dark Blue	Medium Blue	Dark Blue	Dark Blue
	Smartphone	Dark Blue	Medium Blue	Dark Blue	Medium Blue	Dark Blue	Medium Blue
	Cámara 180°	Medium Blue	Dark Blue	Dark Blue	Dark Blue	Medium Blue	Medium Blue
	Renders	Medium Blue	Light Blue	Dark Blue	Light Blue	Light Blue	Medium Blue
 Cilíndrica	Cámara réflex	Dark Blue	Light Blue	Dark Blue	Medium Blue	Light Blue	Dark Blue
	Smartphone	Dark Blue	Medium Blue	Dark Blue	Medium Blue	Light Blue	Medium Blue
	Renders	Dark Blue	Light Blue	Dark Blue	Light Blue	Light Blue	Medium Blue
 Plana	Cámara réflex	Light Blue	Light Blue	Dark Blue	Medium Blue	Light Blue	Dark Blue
	Smartphone	Light Blue	Medium Blue	Dark Blue	Medium Blue	Light Blue	Medium Blue
	Renders	Light Blue	Light Blue	Dark Blue	Medium Blue	Light Blue	Medium Blue
 Modelo 3D	Modelo 3D	Dark Blue	Medium Blue	Medium Blue	Light Blue	Light Blue	Medium Blue



Se entiende por una aplicación con cualidades positivas aquella que pese poco, tenga buena calidad de imagen, sea inmersiva, conserve la disparidad, se obtenga rápidamente y necesite poco presupuesto.

Tabla 1. Comparativa de aplicaciones de entornos estáticos

## 8.2- Conclusiones sobre aplicaciones de entornos dinámicos

Para recrear entornos dinámicos podemos utilizar, tanto un par de vídeo estereoscópico, como un modelo 3D directamente. Aunque en este último, respecto de los entornos estáticos, vamos a ver mermadas las opciones que permiten el uso de procedimientos físicos para la obtención de vídeo estereoscópico. Esto es así porque se necesita que todas las cámaras que van a capturar la escena funcionen a la vez, lo cual es físicamente imposible en muchos casos. Podemos clasificar estas aplicaciones en función del campo de visión que representen, distinguiendo entre aplicaciones esféricas, semiesféricas, cilíndricas y planas.

Hay una cuestión relativa a la ergonomía que afecta a todas estas aplicaciones, y es que, la cámara, en el caso de desplazarse durante el transcurso de la aplicación, no puede rotar. El usuario es dueño de la rotación de la escena, esto en parte aviva la sensación de estar inmerso en ella. Si además de él, la aplicación también controla estos movimientos, puede producirse sensación de mareo.

Las aplicaciones esféricas, producen una experiencia inmersiva de lo más realista, además de permitir observar al usuario en todas las dimensiones, generan movimiento, haciendo que nuestros sentidos se involucren más en la escena. Este tipo de aplicaciones se puede generar a partir tanto de vídeo como de un modelo 3D. La desventaja de este tipo de aplicaciones, es que los fotogramas del vídeo al necesitar mucha resolución para representar toda la escena, dan como resultado aplicaciones muy pesadas. La solución como ya veremos, siempre y cuando sea viable, es utilizar un modelo 3D.

En cuanto a las aplicaciones semiesféricas, que solo se pueden obtener mediante vídeo, podemos destacar que son algo más livianas que las aplicaciones esféricas, ya que la resolución de los fotogramas es la mitad. Están indicadas tanto para escenas en las que la cámara permanezca estática, como en las que siga un recorrido rectilíneo. Este tipo de escenas solo admite este recorrido, pues la cámara al no poder rotar, no consigue adaptarse a las curvas del recorrido. La experiencia es bastante inmersiva, siempre y cuando la atención del usuario esté focalizada en la parte visible del entorno y no se desvíe a los extremos de los fotogramas.

Por otro lado, las aplicaciones cilíndricas recogen los 360° en el plano horizontal, despreciando las partes superior e inferior de la escena. Por ello, pueden ser una opción interesante cuando la cámara siga un trayecto curvo y el peso de la aplicación sea un factor decisivo.

Finalmente, las aplicaciones planas, pese a que no son tan interesantes por la pequeña porción de escena que representan, son muy fáciles de conseguir.

### 8.2.1- Aplicaciones esféricas

Este tipo de entornos es el único que se puede representar con modelos 3D directamente, aunque por supuesto también puede hacerse a partir de vídeo. Solo dos dispositivos físicos permiten capturar vídeo esférico estereoscópico. Y, si disponemos de un modelo 3D, podemos hacer uso de los procesos de renderizado para generar vídeo esférico estereoscópico o utilizarlo en Unity para generar directamente una escena dinámica.

Para generar aplicaciones con la cámara 360° necesitamos dos cámaras de forma simultánea, por lo ello crece el presupuesto de esta aplicación. Por otro lado, la calidad de esta aplicación es pésima y la estereoscopia está muy mal repartida. Por si no fuera suficiente, en las zonas donde el efecto estereoscópico se hace nulo se aprecia el reflejo de la lente de la otra cámara. Observar estos reflejos, que están a muy poca distancia del objetivo de la cámara, produce una sensación bastante molesta. Debido a ello, se desaconseja el uso de este tipo de cámaras para representar aplicaciones esféricas.

La mejor opción en lo que a métodos físicos se refiere, es la Vuze camera, que ha sido especialmente diseñada para la captura de foto y vídeo estereoscópicos 360°. Aunque se echa en falta una mejor distribución de la disparidad por toda la escena y mayor resolución. Un handicap considerable de esta cámara es su elevado precio, por lo demás supone la mejor herramienta física en el mercado para capturar este tipo de vídeo.

Por último, si disponemos de un modelo 3D tenemos dos opciones: crear un vídeo con herramientas de renderizado, que dará como resultado una aplicación muy pesada, o introducir el modelo en Unity y animar directamente la escena. Estos dos procedimientos se diferencian en que el primero renderiza y guarda toda la información relativa al movimiento de la escena en los fotogramas del vídeo, y el segundo, sin embargo, renderiza a tiempo real la escena en función de la posición de la cámara. Sin embargo, ambos tienen en común la precisión de las técnicas virtuales, que ofrece aplicaciones muy ergonómicas y con la disparidad homogéneamente repartida.

El uso de un modelo 3D da como resultado aplicaciones muy ligeras, aunque no con excesiva calidad, pero sí más que suficiente como para ofrecer una experiencia inmersiva.

El vídeo ofrece la posibilidad de personalizar la calidad de la aplicación. No obstante, demasiada calidad (que normalmente se traduce en fotogramas más grandes), origina que las aplicaciones resultantes tengan pesos desorbitados o no se puedan reproducir con fluidez en las plataformas de destino. Además, renderizar una secuencia de fotogramas de estas características, es un proceso muy largo.

---

### 8.2.2- Aplicaciones semiesféricas

Las aplicaciones semiesféricas solo podrán ser representadas a partir de vídeo. Tan solo un dispositivo permitirá grabar vídeos estereoscópicos de estas características, y por supuesto, también los podremos obtener gracias a las herramientas de renderizado.

El dispositivo que permite grabar vídeo estereoscópico es la cámara 180°, que gracias a su lente de gran angular, permite capturar panoramas semiesféricos. Para grabar vídeo necesitaremos dos cámaras simultáneamente, con lo que se incrementará el presupuesto del proyecto y la lente de cada cámara será visible para la otra. La aplicación resultante será muy pesada y no ofrecerá una calidad de vídeo demasiado buena ni una distribución de la disparidad homogénea. Aunque, si se logra que la atención del observador permanezca en torno al centro de los fotogramas, las deficiencias de la falta de la disparidad no son tan evidentes y se evita que se observen los límites de la escena.

Por último, el motor de renderizado ofrecerá aplicaciones en las que podemos personalizar la calidad del vídeo. Además de presentar la estereoscopia homogéneamente repartida por toda la escena. Por lo tanto, esta herramienta proporciona aplicaciones más ergonómicas, aunque es la opción menos inmediata, puesto que el tiempo de renderizado de un vídeo de estas características suele ser bastante elevado.

### 8.2.3- Aplicaciones cilíndricas

Este tipo de aplicación solo puede obtenerse mediante vídeo cilíndrico estereoscópico. La gran pega de este tipo de vídeo es que solo puede obtenerse mediante procedimientos de renderizado, por lo cual, se necesita de un modelo 3D previo para su generación.

Este tipo de aplicaciones son algo más livianas que las esféricas, aunque el ahorro en peso no compensa para nada la pérdida en calidad. Ya que las sensaciones que se producen al observar este tipo de escenas, en las que constantemente estamos viendo los bordes de los fotogramas, no son demasiado inmersivas.

Por esta razón, no se aconseja el uso de este tipo de aplicación, ya que además no sirve para representar entornos físicos que no se encuentren digitalizados.

### 8.2.4- Aplicaciones planas

Al igual que en las aplicaciones estáticas, este es el formato menos interesante para representar aplicaciones dinámicas. Solo se puede generar a partir de pares de vídeo estereoscópico planos. Para grabar este tipo de vídeo necesitamos dos cámaras de vídeo dispuestas en paralelo y separadas 65mm. Por disponibilidad vamos a utilizar el smartphone, aunque hay multitud de dispositivos que se podrían colocar en esta disposición. Además, también podemos recurrir a los motores de renderizado para generar este tipo de vídeo.

El uso de smartphones es la forma más inmediata de obtener este tipo de vídeo. Se recomienda emplear dos terminales idénticos para que los parámetros de vídeo coincidan entre ambas cámaras. La calidad del vídeo obtenido es muy buena y solo con sincronizar manualmente los vídeos en el ordenador, tendremos el material listo para generar este tipo de aplicación.

Para finalizar, podemos emplear renders para la generación de estos vídeos. No obstante, por presentar generalmente peor calidad que el vídeo real, y poseer prácticamente las mismas cualidades en cuanto a ergonomía, se aconseja utilizar dos smartphones siempre y cuando sea posible. De esta manera nos ahorramos la necesidad de disponer de un modelo 3D y la espera del largo proceso de renderizado.

Tabla 2. Comparativa de aplicaciones de entornos dinámicos

### 8.1.5- Comparativa general de aplicaciones

Esta tabla se ha creado con la intención de poder visualizar de forma gráfica las cualidades más relevantes de las aplicaciones de entornos dinámicos hasta ahora analizadas:

Cualidades Aplicaciones							
		Peso	Calidad	Inmersividad	Estereoscopia	Tiempo	Presupuesto
	Vuze camera	Dark	Dark	Light	Dark	Light	Dark
	Cámara 360°	Dark	Dark	Dark	Dark	Light	Dark
	Renders	Dark	Light	Light	Light	Dark	Light
Semi-esférica	Cámara 180°	Dark	Dark	Dark	Dark	Light	Dark
	Renders	Dark	Light	Dark	Light	Dark	Light
Cilíndrica	Renders	Dark	Light	Dark	Light	Light	Dark
Plana	Smartphone	Dark	Light	Dark	Light	Light	Dark
	Renders	Dark	Light	Dark	Light	Dark	Light
Modelo 3D	Modelo 3D	Light	Light	Light	Light	Light	Light



Se entiende por una aplicación con cualidades positivas aquella que pese poco, tenga buena calidad de imagen, sea inmersiva, conserve la disparidad, se obtenga rápidamente y necesite poco presupuesto.

## 9- BIBLIOGRAFÍA

STEVEN M. LA VALLE (University of Illinois), A. 2017 Virtual Reality. [Consulta 20/10/2017]. Disponible en: <http://vr.cs.uiuc.edu/vrbook.pdf>

UNIVERSIDAD NACIONAL DE SAN JUAN (ARGENTINA). Visión estereoscópica. [Consulta 20/10/2017]. Disponible en: <ftp://ftp.unsj.edu.ar/agrimensura/Fotogrametria/Unidad4/VISION-ESTEREO.pdf>

OCULARIS, A. 2006. La visión tridimensional. [Consulta 20/10/2017]. Disponible en: <https://ocularis.es/la-vision-tridimensional/>

RESEARCHGATE, A.2014. Estereoscopia de dos colores, anaglifos. [Consulta 22/10/2017]. Disponible en: [https://www.researchgate.net/profile/Rolando\\_Serra\\_Toledo/publication/261985555\\_Estereoscopia\\_de\\_dos\\_colores\\_anaglifos/links/0a85e53611c9e5cfd5000000.pdf](https://www.researchgate.net/profile/Rolando_Serra_Toledo/publication/261985555_Estereoscopia_de_dos_colores_anaglifos/links/0a85e53611c9e5cfd5000000.pdf)

PARALLAX3D BLOGSPOT, A. 2013. Sistemas de proyección y visionado-POLARIZACIÓN. [Consulta 22/10/2017]. Disponible en: <http://parallax3d.blogspot.com.es/2013/09/3d-8-sistemas-de-proyeccion-y-visionado.html>

NAUKAS, A. 2013. Viendo la televisión 3-D con tu mascota. [Consulta 22/10/2017]. Disponible en: <http://naukas.com/2013/10/31/viendo-la-television-3-d-con-tu-mascota/>

VRS. History of Virtual Reality. [Consulta 25/10/2017]. Disponible en: <https://www.vrs.org.uk/virtual-reality/history.html>

SCIENCEFOCUS, A.2016. The History of Virtual Reality. [Consulta 25/10/2017]. Disponible en: <http://www.sciencefocus.com/article/history-of-virtual-reality>

SCIENCE & SOCIETY PICTURE LIBRARY. Reflecting stereoscope originally used by Charles Wheatstone, 19th century. [Consulta 25/10/2017]. Disponible en: <http://www.ssplprints.com/image/129529/reflecting-stereoscope-originally-used-by-charles-wheatstone-19th-century>

COLLECTTOYS. View-Master. [Consulta 25/10/2017]. Disponible en: <http://m.collecttoys.net/View-Master/viewmaster.php>

TECNOLOGÍA OBSOLETA, A. 2009. Sensorama. [Consulta 25/10/2017]. Disponible en: <http://alpoma.net/tecob/?p=1141>

FLASHBAK, A. 2014. Jaron Lanier's EyePhone: Head And Glove Virtual Reality In The 1980s. [Consulta 25/10/2017]. Disponible en: <https://flashbak.com/jaron-laniers-eyephone-head-and-glove-virtual-reality-in-the-1980s-26180/>

DRA, M<sup>a</sup> CINTA PUELL MARÍN (Universidad Complutense de Madrid). El Sistema Óptico del Ojo y la Visión Binocular. [Consulta 30/10/2017]. Disponible en: [http://eprints.ucm.es/14823/1/Puell\\_%C3%93ptica\\_Fisiol%C3%B3gica.pdf](http://eprints.ucm.es/14823/1/Puell_%C3%93ptica_Fisiol%C3%B3gica.pdf)

WEARABLE, A. 2017. We've tried out every VR headset to help you find the best for you. [Consulta 3/11/2017]. Disponible en: <https://www.wearable.com/vr/best-vr-headsets-2017>

XATAKA, A. 2017. HTC Vive, análisis: esto sí que es realidad virtual interactiva. [Consulta 3/11/2017]. Disponible en: <https://www.xataka.com/analisis/htc-vive-analisis-esto-si-que-es-realidad-virtual-interactiva>

BLOG DEL FOTOGRAFO, A. 2015. La Distancia Hiperfocal Explicada. [Consulta 4/11/2017]. Disponible en: <http://www.blogdelfotografo.com/distancia-hiperfocal/>

MARIO CARVAJAL, A. 2013. Fotografía esférica 360 usando un 18-55 mm si es posible. [Consulta 4/11/2017]. Disponible en: <http://www.mariocarvajal.com/fotografia-esferica-360-usando-un-18-55-mm-si-es-posible/>

XATAKAPHOTO, A. 2012. Cómo encontrar el punto de no paralelaje: Truco exprés. [Consulta 4/11/2017]. Disponible en: <https://www.xatakafoto.com/trucos-y-consejos/como-encontrar-el-punto-de-no-paralelaje-truco-expres>

BERNIE ROEHL, A. 2014. Full 360 stereoscopic vídeo playback in Unity. [Consulta 10/11/2017]. Disponible en: <http://bernieroehl.com/360stereoinunity/>

DOCS UNITY. Configuración del SDK de Android. [Consulta 10/11/2017]. Disponible en: <https://docs.unity3d.com/es/current/Manual/android-sdksetup.html>

DOCS UNITY. OpenVR. [Consulta 10/11/2017]. Disponible en: <https://docs.unity3d.com/Manual/VRDevices-OpenVR.html>

DOCS UNITY. Movie Texture. [Consulta 12/11/2017]. Disponible en: <https://docs.unity3d.com/es/current/Manual/class-MovieTexture.html>

ANSWERS UNITY. Unable to list target platforms. Please make sure the android sdk path is correct. See the Console for more details. [Consulta 10/11/2017]. Disponible en: <https://answers.unity.com/questions/1323731/unable-to-list-target-platforms-please-make-sure-t.html>

ANSWERS UNITY. Video Player as texture, can't set tiling. [Consulta 13/11/2017]. Disponible en: <https://answers.unity.com/questions/1366010/video-player-as-texture-cant-set-tiling.html>

## 9.1- Bibliografía referenciada

[1] <ftp://ftp.unsj.edu.ar/agrimensura/Fotogrametria/Unidad4/VISION-ESTEREO.pdf> [Consulta 6/11/2017]

[2] <http://www.easypano.com/es/products.html> [Consulta 6/11/2017]

[3] <https://www.adobe.com/es/products/photoshop.html> [Consulta 6/11/2017]

[4] <https://www.autodesk.es/products/3ds-max/overview> [Consulta 6/11/2017]

## 9.2- Bibliografía gráfica

Figura 3 - <https://www.bostonglobe.com/>

Figura 13 - <https://www.microsoft.com>

Figura 14 - <https://group.sciencemuseum.org.uk/>

Figura 15 - <http://ojs.viewjournal.eu>

Figura 16 - Universidad de Huelva

Figura 17 - <http://m.collecttoys.net/>

Figura 18 - <https://www.researchgate.net/>

Figura 19 - <http://www.mortonheilig.com>

Figura 20 - <http://www.newmediabusinessblog.org>

Figura 21 - <http://www.sciencefocus.com>

Figura 22 - <http://flashbak.com>

Figura 23 - <http://www.sciencefocus.com>

Figura 24 - <http://www.sciencefocus.com>

Figura 25 - <https://www.oculus.com>

Figura 27 - <https://vr.google.com>

Figura 28 - <https://uploadvr.com/>

Figura 29 - <https://vr.google.com>

Figura 30 - <https://innoactive.de/>

Figura 31 - <https://www.xda-developers.com/>

Figura 32 - <https://www.oculus.com/>

Figura 33 - <https://www.playstation.com/>

Figura 41 - <http://casual-effects.blogspot.com.es/>

Figura 58 - <https://www.panono.com/>

Figura 60 - <https://theta360.com/es/>

Figura 62 - <https://assemblyl.com/>

Figura 63 - <https://www.sony.co.uk>

Figura 66 - <https://vuze.camera/>

Figura 67 - <https://phandroid.com>

Figura 68 - <https://facebook360.fb.com>

### 10.3- Vídeos extraídos para crear aplicaciones

Vuze (NY) - <https://www.youtube.com/watch?v=H6jyOIE7ygU>

Vuze (Zuidas) - <https://www.youtube.com/watch?v=kcfoOsE8qhA>

Vuze (Lineal) - <https://www.youtube.com/watch?v=7Le-fp48whk>

Vuze (Free) - [https://www.youtube.com/watch?v=vaZO\\_BB7rfg&t=92s](https://www.youtube.com/watch?v=vaZO_BB7rfg&t=92s)

PixPro 180 - <https://www.youtube.com/watch?v=QmUxssm8das&t=16s>

PixPro 360 - <https://www.youtube.com/watch?v=EsfRpXwjQes&t=112s>