

Autoestereoscopia en dispositivos móviles para facilitar el aprendizaje

Universidad Politécnica de Valencia
Departamento de Sistemas Informáticos y
Computación



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

Trabajo fin de Máster
Inteligencia Artificial, Reconocimiento de Formas e Imagen Digital

Autor:
Carlos Carrasco Torres

Supervisor:
Dra. M. Carmen Juan Lizandra

Valencia, Julio 2014

Agradecimientos

Terminar este trabajo ha sido un gran reto para mí. Quiero agradecer a todas esas personas que de una forma o de otra han hecho que este viaje sea mucho más fácil. Sois muchos, por lo que no podré mencionaros a todos en este trabajo. Espero poder hacerlo algún día.

Para empezar me gustaría agradecer a mi profesora M. Carmen Juan Lizandra, tutora de este trabajo final de máster. Muchas gracias por tu paciencia, por tus ganas de mejorar, por tu continua revisión y por tus ingeniosas ideas. Todo este proceso ha sido mucho más fácil e instructivo gracias a ti.

También darles las gracias a David y Juan Fernando, grandes investigadores, siempre dispuestos a ayudarme cuando las cosas no salían como yo quería. Vuestra ayuda ha sido un oasis en medio de un desierto.

Por supuesto me gustaría agradecer a toda mi familia la ayuda que me han brindado todos estos años. A mi hermana agradecerle su humor inconfundible que me ha hecho reír en los malos momentos, a mi padre agradecerle el afán de superación que me ha inculcado desde que era pequeño y que ha hecho que siempre quiera mejorar en todo lo que hago y como no darle las gracias a mi madre, por su paciencia y su constancia, por sus grandes consejos y por haberme dado ese impulso constante que me ha hecho conseguir todas las metas que me he propuesto.

Por último darle las gracias a mi novia, la cual me ha apoyado en mis peores momentos y ha sabido darme el tiempo y los ánimos necesarios para poder acabar este trabajo cuando las cosas se ponían cuesta arriba. Gracias por tu apoyo incondicional, por saber aguantarme cuando ni siquiera yo lo hacía y sobre todo por hacerme ver que los problemas, cuando se comparten, siempre son más fáciles.

Esta tesis está dedicada a todos vosotros, pero sobre todo a ti.

Índice

1. Introducción.....	10
1.1 Motivación	10
1.2 Objetivos.....	11
1.3 Estructura de la tesina	13
2. Estado del arte	14
2.1 Introducción.....	14
2.2 Aprendizaje dental.....	14
2.2.1 Simuladores dentales basados en maniquís.....	15
2.2.2 Simuladores dentales basados en dispositivo hápticos.....	17
2.2.3 RA y aprendizaje dental.....	19
2.3 Sistemas autoestereoscópicos en procedimientos dentales	20
3. Estudio de las tecnologías empleadas.....	24
3.1 Open Scene Graph (OSG)	24
3.2 Wrapper OSG – Android	26
3.3 Realidad Aumentada.....	28
3.3.1 Toolkits de realidad aumentada	33
3.4 Tecnología 3D	33
3.4.1 Estereoscopía	33
3.4.2 Autoestereoscopía.....	37
3.4.3 LG Real3D SDK	40
4. Desarrollo.....	41
4.1 Arquitectura del sistema	41
4.2 Hardware	41
4.3 Software.....	42
4.3.1 3D Max.....	42
4.3.2 NDK de Android.....	44
4.3.3 Dependencias entre Proyectos	45
4.3.4 Uso de OSG y Real 3D.....	46
4.4 Detalles de la implementación	48

4.4.1 Sistema 1: OSG + Real 3D.....	48
4.4.2 Sistema 2: OSG + Real 3D + RA.....	50
4.5 Problemas encontrados.....	53
5. Validación.....	54
5.1 Introducción.....	54
5.2 Diseño.....	54
5.2.1 Participantes.....	54
5.2.2 Cuestionarios.....	55
5.3 Procedimiento.....	55
5.4 Resultados.....	57
5.4.1 Resultados del aprendizaje.....	57
5.5 Preferencias.....	60
5.6 Resultados de satisfacción.....	62
6. Conclusiones y trabajo futuro.....	66
6.1 Conclusión.....	66
6.2 Trabajo futuro.....	67
Bibliografía.....	68
Cuestionarios.....	71

LISTA DE ILUSTRACIONES

ILUSTRACIÓN 1 - SIMULADOR DENTAL DSEPLUS	15
ILUSTRACIÓN 2 - SIMULADOR DENTAL DENTSIM.....	16
ILUSTRACIÓN 3 - DESPLIEGUE DE SIMULADORES DENTSIM EN LA UNIVERSIDAD DE CARDIFF, GALES, REINO UNIDO	16
ILUSTRACIÓN 4 - SIMULADOR DENTAL VRDTS	17
ILUSTRACIÓN 5 - COMPARACIÓN ENTRE MANEJAR UN ELEMENTO REAL Y UN ELEMENTO HÁPTICO.....	18
ILUSTRACIÓN 6 - EJEMPLO DE USO DEL SISTEMA HVRPTS	18
ILUSTRACIÓN 7 - ENTORNO DE TRABAJO INMERSIVO DE LA COMPAÑÍA DEVINSENSE	19
ILUSTRACIÓN 8 - EJEMPLO DE FUNCIONAMIENTO DE ARDENTAL.....	20
ILUSTRACIÓN 9 - ARQUITECTURA DEL SISTEMA AUGMENTED REALITY NAVIGATION WITH AUTOMATIC MARKER	21
ILUSTRACIÓN 10 - RESULTADOS OBTENIDO POR EL SISTEMA ARNWAM	22
ILUSTRACIÓN 11 - FONDO NEUTRO	23
ILUSTRACIÓN 12 - FONDO REAL	23
ILUSTRACIÓN 13 - DESCRIPCIÓN GENERAL DE LA ARQUITECTURA EMPLEADA POR OSG.....	25
ILUSTRACIÓN 14 - ARQUITECTURA PROPUESTA PARA EL WRAPPER REALIZADO	27
ILUSTRACIÓN 15 - ARQUITECTURA PROPUESTA PARA EL WRAPPER REALIZADO	28
ILUSTRACIÓN 16 - ESQUEMA DEL CONTINUO DE MILGRAM	29
ILUSTRACIÓN 17 - EJEMPLO DE UN OBJETO EN LOS DIFERENTES ESTADOS.....	29
ILUSTRACIÓN 18 - ESQUEMA DE FUNCIONAMIENTO DE UN SISTEMA DE REALIDAD AUMENTADA TÍPICO EN FUNCIÓN DEL ÁNGULO DE CAPTACIÓN.....	30
ILUSTRACIÓN 19 - MARCADOR PERSONALIZADO, MARCADOR VUFORIA Y MARCADOR EN UN LIBRO ...	31
ILUSTRACIÓN 20 - EJEMPLO DE RA UTILIZANDO UN MARCADOR.....	32
ILUSTRACIÓN 21 - EJEMPLO DE RA UTILIZANDO POSICIONAMIENTO GPS	32
ILUSTRACIÓN 22 - PERCEPCIÓN ESTEREOSCÓPICA.....	34
ILUSTRACIÓN 23 - ANAGLIFOS	35
ILUSTRACIÓN 24 - GAFAS POLARIZADAS.....	36
ILUSTRACIÓN 25 - 3D ACTIVO	36
ILUSTRACIÓN 26 - BARRERAS DE PARALAJE UTILIZANDO DOS IMÁGENES	38
ILUSTRACIÓN 27 - BARRERAS DE PARALAJE MULTIVISTA	38
ILUSTRACIÓN 28 - HOJAS LENTICULARES.....	39
ILUSTRACIÓN 29 - PANTALLAS AUTOMULTIESCÓPICAS	39
ILUSTRACIÓN 30 - FUNCIONAMIENTO DEL MÓVIL ESTEREOSCÓPICO LG.....	40
ILUSTRACIÓN 31 - LG OPTIMUS 3D P920	42
ILUSTRACIÓN 32 - MOLAR 47 EN 3D MAX DESDE DIFERENTES ÁNGULOS.....	43
ILUSTRACIÓN 33 - CREACIÓN DE LOS MODELOS Y LEYENDA DE LOS ELEMENTOS MORFOLÓGICOS.....	43
ILUSTRACIÓN 34 - CARPETA JNI DE NUESTRO PROYECTO	46
ILUSTRACIÓN 35 - COMPILACIÓN DE LA LIBRERÍA OSG	46
ILUSTRACIÓN 36 - DIAGRAMA DE LA CLASE OSGSTEREOACTIVITY	47
ILUSTRACIÓN 37 - INTERFAZ PROPUESTA PARA LA APLICACIÓN.....	48
ILUSTRACIÓN 38 - ARQUITECTURA DEL SISTEMA OSG - REAL 3D.....	49
ILUSTRACIÓN 39 - RESULTADO FINAL OBTENIDO POR EL PRIMER SISTEMA. PARTE 1.....	50
ILUSTRACIÓN 40 - RESULTADO FINAL OBTENIDO POR EL PRIMER SISTEMA. PARTE 2	50
ILUSTRACIÓN 41 - ARQUITECTURA DEL SISTEMA OSG - REAL 3D - RA.....	51
ILUSTRACIÓN 42 - RESULTADO FINAL OBTENIDO POR EL SEGUNDO SISTEMA. PARTE 1	52
ILUSTRACIÓN 43 - RESULTADO FINAL OBTENIDO POR EL SEGUNDO SISTEMA. PARTE 2	53
ILUSTRACIÓN 44 - PROCEDIMIENTO SEGUIDO POR EL GRUPO A.....	56
ILUSTRACIÓN 45 - PROCEDIMIENTO SEGUIDO POR EL GRUPO B	56

ILUSTRACIÓN 46 – PUNTUACIÓN OBTENIDA PARA LA VARIABLE CONOCIMIENTO EN EL PRE Y POST TEST USANDO AMBOS SISTEMAS.....	58
ILUSTRACIÓN 47 - CONOCIMIENTO INICIAL Y FINAL, EN BASE AL SISTEMA UTILIZADO	59
ILUSTRACIÓN 48 - CONOCIMIENTO INICIAL Y FINAL EN BASE A LA ESPECIALIDAD DEL ALUMNO.....	60
ILUSTRACIÓN 49 - PREFERENCIA DE SISTEMA POR ESPECIALIDAD.....	61
ILUSTRACIÓN 50 - PREFERENCIA DE SISTEMA POR SEXO.....	62
ILUSTRACIÓN 51 - ALUMNA REALIZANDO EL TEST ESTEREOSCÓPICO.....	64
ILUSTRACIÓN 52 - ALUMNA UTILIZANDO EL SISTEMA 2D.....	64
ILUSTRACIÓN 53 - ALUMNO COMPLETANDO EL TEST Q3 Y Q4	65

Lista de Tablas

TABLA 1 DISTRIBUCIÓN DE LOS ALUMNOS.....	55
TABLA 2 RESULTADO DEL TEST ESTEREOSCÓPICO	57
TABLA 3 RESULTADOS OBTENIDOS CON EL TEST DE KRUSKAL-WALLIS PARA LOS DIFERENTES FACTORES UTILIZADOS	59
TABLA 4 MEDIANAS PARA LAS PREGUNTAS DE PREFERENCIAS, EL TEST MANN-WHITNEY U Y R, TAMAÑO DEL EFECTO	60
TABLA 5 - PREFERENCIA DE SISTEMA POR ESPECIALIDAD.....	61
TABLA 6 - PREFERENCIA DE SISTEMA POR SEXO	61
TABLA 7 MEDIANAS PARA LAS PREGUNTAS DE SATISFACCIÓN, EL TEST MANN-WHITNEY U Y R, TAMAÑO DEL EFECTO	62

1. Introducción

1.

1.1	Motivación.....	10
1.2	Objetivos	11
1.3	Estructura de la tesina.....	13

1.1 Motivación

En la actualidad el uso de dispositivos móviles se ha extendido de forma apabullante en nuestra sociedad. En uno de los últimos estudios realizados se estima que en el año 2014 habrá alrededor de 7.3 billones de dispositivos móviles en el mundo, más que toda la población existente en la tierra [PRAMIS13]. En España se estima que para este año tengamos 55,740,000 de dispositivos, lo que equivale a 1.18 móviles por persona [WIKI14].

Barajando estos números es normal que exista una gran expectativa en torno a este tipo de tecnologías móviles. Esto ha hecho que un gran número de compañías y desarrolladores se hayan interesado en este mercado.

Existen infinidad de sectores en los que podemos utilizar los dispositivos móviles para generar un beneficio, en nuestro caso hemos querido aplicarlo al sector de la enseñanza, donde la incorporación de las nuevas tecnologías puede ser un punto de inflexión en el concepto que tenemos de “aprender”.

Para poder realizar este proyecto hemos querido responder a cuatro preguntas:

- ¿Cómo vamos a crear una aplicación atractiva y efectiva con las limitaciones que tiene un móvil?
- ¿Qué novedad le vamos a ofrecer al alumno para que se implique más en el proceso de aprendizaje?
- ¿Qué hemos planteado para resolver este problema?
- ¿Realmente esta tecnología mejorará el aprendizaje adquirido por el alumno?

Las personas vivimos en un mundo donde el diseño y lo visual marcan las modas y las tendencias, por lo tanto, la capacidad para crear representaciones tridimensionales y un contenido exclusivo es fundamental para poder llamar la atención del usuario.

El renderizado de escenas tridimensionales y de grafos complejos es un problema importante que se complica cuando hablamos de dispositivos móviles. Estos dispositivos disponen de una memoria y una capacidad de procesamiento limitada.

En esta tesina se ha utilizado la librería OpenSceneGraph (OSG), el estándar OpenGL para grafos de escena, una librería de código libre y multiplataforma escrita en C++. ¿Por qué OSG? Lo primero, porque es de código libre y está respaldada por una gran comunidad que mejora la librería cada día. Además es muy versátil y portable, ya que actualmente permite realizar aplicaciones sobre Windows, Linux, Unix y los sistemas Apple y Android.

Otra cuestión es: ¿De qué forma podemos utilizar un dispositivo móvil para que un alumno pueda aprender de una forma novedosa? Es aquí donde entra la tecnología 3D, la cual intenta dar un paso más allá a la simple visualización “plana” a la que estamos acostumbrados.

Esta tecnología existe desde hace mucho tiempo, pero en los últimos años ha adquirido una gran relevancia. La podemos ver en las películas de los cines o en las televisiones que tenemos en casa.

Además es una tecnología en la cual se sigue investigando, ya que sigue produciendo beneficios. En 2013 generó casi 2000 millones de dólares en la industria del cine. En los últimos años se ha estado intentando mejorar esta tecnología y una de las mejoras propuestas ha sido la denominada autoestereoscopia, el “3D sin gafas” como se lo conoce popularmente. Gracias a esta tecnología, eliminamos el inconveniente de tener que utilizar unas gafas para poder observar el efecto 3D.

Por lo tanto, tenemos una forma de crear representaciones tridimensionales complejas utilizando un móvil, además podemos utilizar la tecnología 3D para crear un sistema diferente, y aportar algo nuevo al alumno. Solo nos queda ver qué sistema hemos planteado para resolver este problema.

El sistema que hemos desarrollado en este trabajo fin de máster ha sido el siguiente. Un móvil dotado de una pantalla autoestereoscópica mostrará modelos 3D con los cuales el alumno podrá interactuar para poder adquirir conocimiento. Gracias a este sistema y a las pruebas realizadas podremos comprobar si puede llegar a ser una herramienta de aprendizaje recomendable en las nuevas generaciones. Además se ha querido añadir un componente de Realidad Aumentada (RA) al sistema para poder observar el efecto que produce el reproducir modelos 3D mientras se enfoca a un marcador de RA.

1.2 Objetivos

En este trabajo queremos cubrir un objetivo principal. Desarrollar y validar un sistema que permita cargar modelos OSG en un dispositivo con pantalla autoestereoscópica para poder así crear una herramienta de aprendizaje basada en dispositivos móviles.

Para poder cumplir este objetivo, nos hemos fijados una serie de metas que queremos cumplir:

- Diseñar una aplicación que permita cargar modelos dentales complejos en un dispositivo con pantalla autoestereoscópica.
- Desarrollar una segunda aplicación que nos permita representar estos mismos modelos pero utilizando además la tecnología de RA.
- Comparar ambos sistemas y estudiar cuál ofrece mayores prestaciones de cara al aprendizaje.
- Comprobar si hay alguna diferencia estadística significativa entre los conocimientos adquiridos utilizando un sistema 2D y uno 3D.
- Analizar el nivel de satisfacción y aceptación de los estudiantes después de usar la aplicación.

Para completar todos estos objetivos hemos implementado dos aplicaciones, como ya hemos mencionado anteriormente.

En la primera aplicación hemos utilizado la librería OSG para cargar modelos tridimensionales en un móvil dotado de una pantalla autoestereoscópica. Para ello hemos implementado un Wrapper que nos permitirá utilizar la librería nativa OpenSceneGraph (OSG) en dispositivos Android. Gracias a este Wrapper podremos utilizar las funciones y los objetos que ofrece OSG, implementados en C++, en un sistema Android cuyo lenguaje de programación es Java.

En la segunda aplicación hemos seguido la línea que llevamos en la primera, pero además le hemos añadido la tecnología de RA. El objetivo será cargar los modelos anteriores pero ahora deberemos enfocar la cámara hacia el marcador de RA para que aparezca la pieza dental en 3D en la pantalla autoestereoscópica del móvil.

Las librerías que se han utilizado para realizar la implementación de estas dos aplicaciones son:

Librerías gráficas:

- OpenSceneGraph: Librería basada en OpenGL/OpenGL ES la cual añade estructuras y métodos para el manejo de objetos 3D.
- LG Real3D: librería para hacer uso de la parte autoestereoscópica del móvil LG Optimus 3D.

Librerías de RA:

- AndAR: librería basada en ARToolKit. Es una solución que hace uso de una implementación específica de la librería ARToolKit para Android.
- NyARToolKit: Implementación en Java de la librería ARToolKit.
- Vuforia/QCar: librería de la empresa Qualcomm basada en detección por características. Al igual que AndAR es una implementación nativa.

1.3 Estructura de la tesina

La tesina se estructurará en 5 partes:

1 Introducción:

En el primer capítulo explicaremos brevemente cuál ha sido la motivación principal para desarrollar este trabajo. Explicaremos el porque hemos querido utilizar esta tecnología, qué objetivos queremos cumplir y cómo se ha estructurado la tesis.

2 Estado del arte:

En este segundo capítulo analizaremos las novedades que existen en los sistemas de aprendizaje dental. Para ello comentaremos algunos simuladores dentales que utilizan RA y algunas aplicaciones estereoscópicas [DENX14], [LUCIANO06], [WANG14]. Veremos cómo funcionan y su arquitectura. Además comprobaremos si alguno de ellos ya está en el mercado y cuál ha sido su aceptación.

3 Estudio de las tecnologías empleadas:

En este capítulo estudiaremos las principales características de las tecnologías utilizadas. Veremos cómo funciona la librería OSG. Mostraremos las partes principales del Wrapper que se ha utilizado para usar la librería OSG en un sistema Android. Explicaremos cómo funciona la tecnología de RA y cuáles son las plataformas más utilizadas. Hablaremos de la tecnología 3D y profundizaremos en el 3D estereoscópico. Por último detallaremos el framework que utiliza el móvil que hemos empleado.

4 Desarrollo:

En el cuarto capítulo expondremos los dos sistemas de aprendizaje que hemos desarrollado. Explicaremos cómo funcionan, de qué partes están compuestos y cuáles son sus ventajas e inconvenientes. Por último haremos una comparativa entre ellos respecto a la respuesta que pueden ofrecer en materia de educación.

5 Validación y análisis de los resultados:

En esta sección mostraremos las pruebas que se han realizado. Enseñaremos las fases que hemos realizado en el estudio, analizaremos los datos obtenidos y evaluaremos los resultados para poder ver si hemos conseguido el objetivo principal de esta tesina.

6 Conclusiones y trabajo futuro:

En el último capítulo resumiremos todo lo aprendido en este trabajo y veremos las posibles mejoras que se podrían aplicar en un futuro.

7 Cuestionarios:

Enseñaremos los diversos cuestionarios que hemos utilizado en las pruebas realizadas.

2. Estado del arte

2.0

2.1	Introducción.....	14
2.2	Aprendizaje dental	14
2.3	Sistemas autoestereoscópicos en procedimientos dentales	20

2.1 Introducción

Este capítulo incluye información sobre las diferentes tecnologías que existen hoy en día entorno a la materia que nos ocupa en este proyecto. En una primera instancia hablaremos sobre algunos simuladores dentales que utilicen RA y posteriormente mencionaremos algunas aplicaciones que están usando la tecnología estereoscópica y autoestereoscópica.

2.2 Aprendizaje dental

A pesar de que los simuladores médicos son cada vez más comunes en la educación médica, el uso de simuladores en el campo de la odontología aún no ha sido explotado por completo. Los simuladores dentales son una forma eficaz de enseñar a los estudiantes de odontología los procedimientos dentales rápidamente, mientras que aumenta las habilidades manuales considerablemente.

Hay dos tipos de simuladores dentales disponibles en la actualidad: los simuladores basados en maniquís, los cuales proporcionan un modelo físico de la cabeza y la boca y los simuladores hápticos. En el primer tipo de sistemas los procedimientos dentales se realizan utilizando instrumentos reales. En el segundo tipo se emplean dispositivos hápticos PHANTOM™ [PHANTOM14] y modelos virtuales de un diente o de una boca humana, que facilitan la práctica dental al usuario. En lugar de utilizar los instrumentos dentales reales, el alumno tiene el lápiz electrónico para manipular un conjunto de instrumentos virtuales que se muestran en una pantalla. La retroalimentación táctil reproduce sensaciones reales en la mano del operador como si estuviera utilizando instrumentos dentales reales. A continuación mostraremos algunos ejemplos de cada sistema.

2.2.1 Simuladores dentales basados en maniqués

Hay dos compañías que comercializan simuladores dentales basados en maniquí en todo el mundo: KaVo y DenX.

- Unidad Dental de simulación (DSEplus), comercializado por KaVo [KAVO14], es una unidad de formación dental completa con chasis neumático, espacio de almacenamiento, unidad de alimentación, módulo de agua a presión, módulo de aspiración, y elementos odontológicos como por ejemplo, tubos de instrumentación 3-F, 2 tubos de aspiración, control de pie, simulador del paciente, cráneo con mandíbulas, modelo dental, máscara facial y un sistema de posicionamiento neumático para el paciente. Este sistema deberá estar asistido por un profesor que vaya guiando al alumno en el proceso de aprendizaje. En la Ilustración 1 podemos ver un ejemplo de este simulador.



Ilustración 1 - Simulador dental DSEplus

- DentSim™, desarrollado por DenX Ltd [DENX14], consiste en un maniquí con la cabeza, los dientes de plástico, una herramienta de mano dental, una cámara de infrarrojos y dos ordenadores. La cabeza de maniquí y la pieza de mano contienen emisores de infrarrojos que son captados por la cámara de infrarrojos. El uso de este sistema de rastreo óptico sirve para detectar los movimientos que hace el usuario con la herramienta de mano. El funcionamiento está basado en los principios que sustentan la tecnología GPS. Cabe destacar que el sistema es muy preciso a la hora de utilizarlo y cuenta con unos gráficos muy completos. A diferencia del sistema anterior, en éste no es necesaria la asistencia de un experto, ya que el propio ordenador te va corrigiendo, haciendo posible un aprendizaje autónomo. En la Ilustración 2 podemos observar a un estudiante utilizando el sistema DentSim.



Ilustración 2 - Simulador dental DentSim

El uso de simuladores dentales basados en maniquís, en comparación con los métodos tradicionales preclínicos, aumentó los resultados educativos [BUCHANAN01] [BUCHANAN04] [POHLENZ10].

Jasinevicius et al. [JASINEVICUIS04] descubrieron que el tiempo que necesitaban dedicar los profesores se reducía a la mitad al utilizar simuladores como DentSim™. Además, un único profesor puede enseñar a más de un alumno al mismo tiempo, ya que no se requiere una relación uno a uno entre el estudiante y el instructor [CHAN00]. En la Ilustración 3 podemos ver un despliegue de simuladores dentales en la Universidad de Cardiff.



Ilustración 3 - Despliegue de simuladores DentSim en la Universidad de Cardiff, Gales, Reino Unido

Sin embargo, los simuladores presentan las siguientes desventajas:

- El modelo físico o el maniquí tiene que ser reemplazado con frecuencia, lo que aumenta los costos de mantenimiento considerablemente.
- No proporcionan ninguna indicación si el tratamiento realizado por el estudiante sería potencialmente agresivo para un paciente real.

2.2.2 Simuladores dentales basados en dispositivo hápticos

A diferencia de los simuladores basados en maniquís, éstos son mucho más rentables, ya que no hay un modelo físico que necesite ser reemplazado. Además, puesto que el dispositivo háptico mide las fuerzas aplicadas por el alumno al tocar la boca del paciente virtual, es posible detectar cuándo el tratamiento del aprendiz es demasiado agresivo.

Los siguientes son algunos de los simuladores dentales hápticos más conocidos:

- Virtual Reality Dental Training System (VRDTS).

VRDTS fue desarrollado por Novint Technologies en colaboración con la Escuela de Medicina Dental de Harvard [PATIL09] [YAU06]. La finalidad principal de este sistema es dotar al usuario de un entorno virtual donde poder trabajar, para ello el software simula un conjunto de instrumentos dentales (taladro de baja velocidad, un explorador, dos escultores y un empacador), además de incluir varios materiales virtuales como esmalte, dentina, materiales de celulosa o empastes. En la Ilustración 4 vemos un ejemplo de uso del sistema VRDTS.



Ilustración 4 - Simulador dental VRDTS

- Haptics-based virtual reality periodontal training simulator (HVRPTS)

[LUCIANO06] En este sistema el usuario interactúa con el simulador a través de una pantalla estereoscópica y un dispositivo háptico. La pantalla estereoscópica consiste en un monitor CRT, capaz de renderizar imágenes a 100 Hz, sincronizado con unas gafas 3D, que funcionan de forma activa. El simulador permite al aprendiz manipular la posición y la orientación del instrumento periodontal virtual mediante el movimiento de un lápiz táctil, que tiene dimensiones similares a los de los verdaderos instrumentos.

La punta del instrumento virtual está unida al brazo háptico. El sistema mantiene una relación de movimiento de 1:10 entre el instrumento virtual y el lápiz óptico, permitiendo al alumno obtener una gran destreza manual, muy necesaria cuando se trabaja con pacientes reales. En la Ilustración 5 se muestran las similitudes en la postura de la mano del usuario cuando sostiene un instrumento real y un lápiz táctil y en la Ilustración 6 vemos el sistema al completo.



Ilustración 5 – Comparación entre manejar un elemento real y un elemento háptico



Ilustración 6 – Ejemplo de uso del sistema HVRPTS

Existen otros sistemas que podemos combinar con el sistema háptico. La empresa DevinSense [DEVINSENSE14] ha creado un entorno de trabajo inmersivo bastante realista que combina los sensores hápticos PHANTOM™ con un proyector estéreo 3D. En la Ilustración 7 podemos ver el sistema completo y un ejemplo de uso.



Ilustración 7 – Entorno de trabajo inmersivo de la compañía DevinSense

2.2.3 RA y aprendizaje dental

ARDental es una aplicación para el aprendizaje de la morfología dental en la cual se utiliza la tecnología de RA [ARDENTAL13].

El funcionamiento es el siguiente. Una vez se ejecuta la aplicación y se detecta el marcador, aparece flotando sobre la imagen el maxilar inferior. En ese momento se puede acercar, alejar, girar, subir, bajar y mover la Tablet a voluntad para observar el modelo desde cualquier posición. Otra posibilidad es mover el marcador para que la imagen 3D virtual se mueva. Y finalmente también se puede mover el modelo con dos dedos directamente sobre la pantalla y girando el modelo.

Se pueden seleccionar los dientes 31, 32, 33, 34, 35, 36, 37 y 41, 42, 43, 44, 45, 46, 47. Una vez seleccionado un diente, el maxilar inferior desaparece, y se queda en su sitio el modelo 3D del diente seleccionado. En la parte inferior de la pantalla aparece un menú con botones para poder interactuar con el diente.

Los botones son de tipo toggle (ON/OFF) y facilitan la visualización de las partes morfológicas que queremos aprender. En cualquier momento se puede interactuar con el diente y elegir el botón que se desee. Para mostrar dichas partes morfológicas se han creado unos modelos 3D en Autodesk 3ds Max que se van añadiendo al modelo 3D del diente. En la Ilustración 8 se puede observar un molar en el que se muestran distintos elementos morfológicos con distintos colores, como por ejemplo el contorno oclusal o las cúspides céntricas. La aplicación se ha desarrollado sobre la plataforma creada para el desarrollo de videojuegos Unity3D con la integración de Vuforia – Unity3D Extensión.

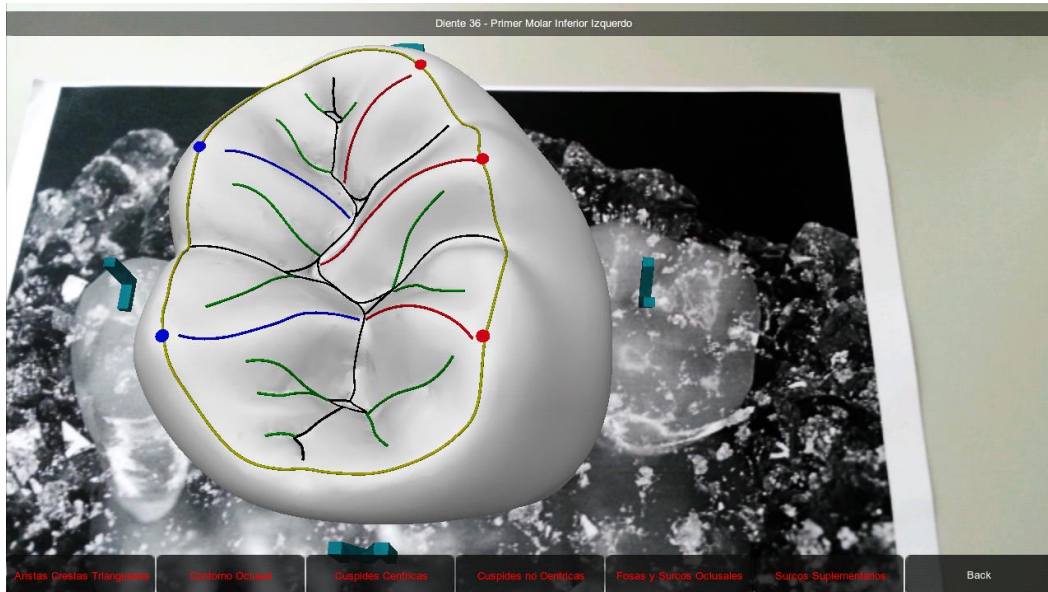


Ilustración 8 – Ejemplo de funcionamiento de ARDental

2.3 Sistemas autoestereoscópicos en procedimientos dentales

Augmented Reality Navigation With Automatic Marker-Free Image Registration Using 3-D Image Overlay for Dental Surgery [WANG13] [WANG14]

Éste es un sistema parecido a los anteriores, con la diferencia de que en este caso se combina la Realidad Aumentada con la tecnología autoestereoscópica. El sistema está compuesto por los siguientes elementos:

Un Dispositivo de visualización 3D, el cual está compuesto por una pantalla de cristal líquido (LCD) de alta resolución y una lente convexa con una disposición hexagonal colocada frente la pantalla para que nos sirva de emisor 3D autoestereoscópico. Un espejo semi-plateado que nos servirá para la superposición de las imágenes. Una unidad de procesamiento de gráficos (GPU) que será la encargada de superponer todas las imágenes y crear el modelo correspondiente.

Una cámara estéreo que se utilizará para el seguimiento del paciente y del instrumental. Por último tenemos la cabeza de un maniquí, en la cual se superpondrán la mandíbula virtual generada por el ordenador.

En la Ilustración 9 podemos ver la arquitectura empleada para este sistema:

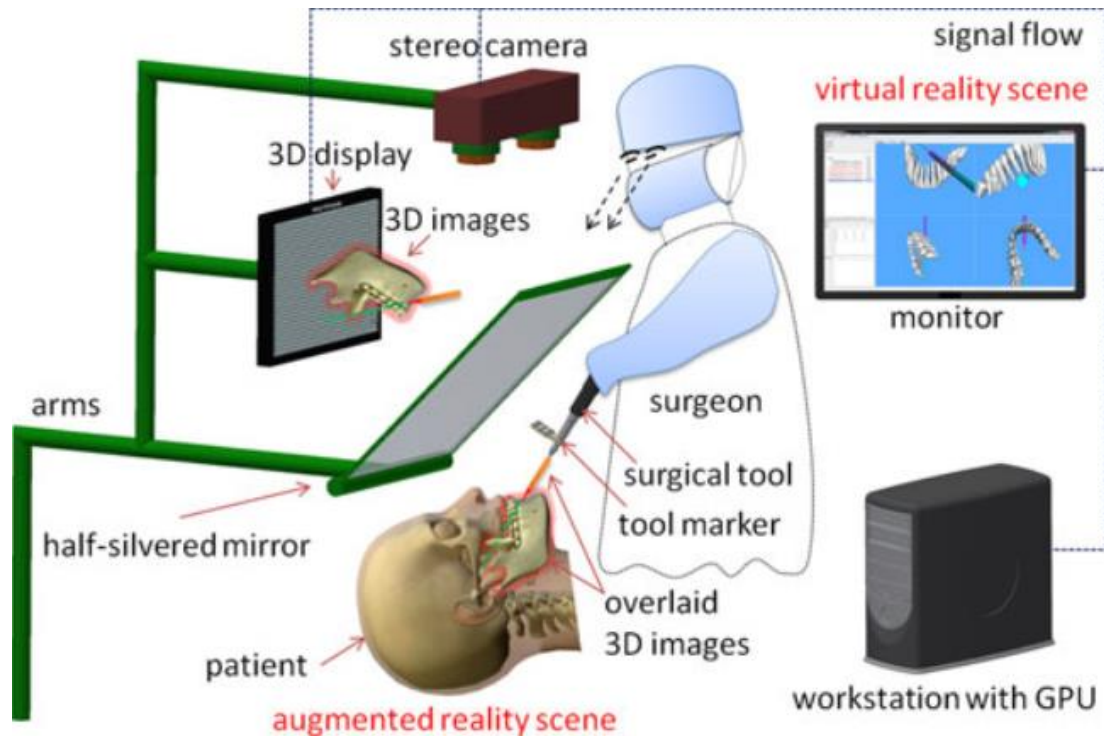


Ilustración 9 - Arquitectura del sistema Augmented Reality Navigation With Automatic Marker

El proceso de funcionamiento es el siguiente. La cámara va captando los movimientos que realiza el odontólogo con la herramienta. Al mismo tiempo va captando la posición que tiene la cabeza. Con esta información el ordenador compone la mandíbula que va mostrar y la adapta a la posición del maniquí. Además coloca un determinado modelo (herramienta dental) en el marcador registrado por el instrumental [LEE07]. Todas estas imágenes se mandan al display 3D que emitirá la señal hacia el espejo. En esta superficie se superpondrán las imágenes generadas por el ordenador con la imagen real del maniquí [KATO99]. Esta superficie será la que mire el experto cuando realice las pruebas. En la Ilustración 10 podemos observar el resultado obtenido por este sistema.

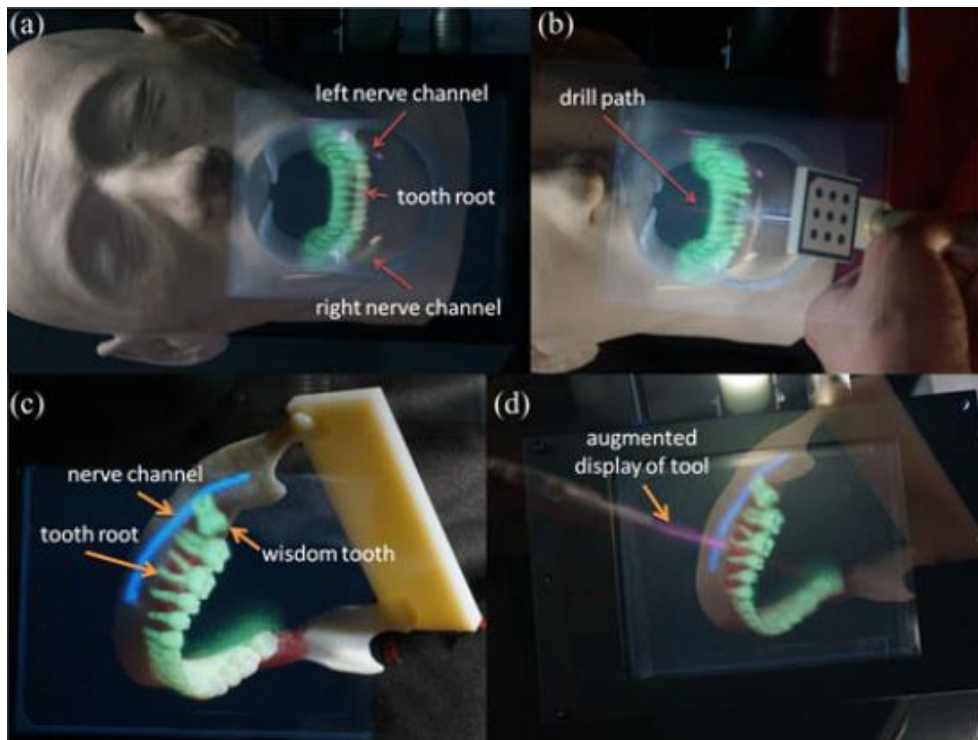


Ilustración 10 - Resultados obtenido por el sistema ARNWAM

Natural user interfaces and autostereoscopy for learning in dentistry
[AUTOESTEREO NUI14]

AutoestereoNUI es una aplicación que integra el uso de una pantalla autoestereoscópica con una interfaz de usuario natural. El objetivo es desarrollar un “juego” que facilite el aprendizaje de la morfología dental del maxilar inferior.

El juego está dividido en dos fases:

- Fase 1 - Aprendizaje: En esta fase el usuario ve los 14 dientes del maxilar inferior en 3D acompañados cada uno con su nombre.
- Fase 2 - Interacción: En esta fase se muestran uno a uno los dientes del maxilar inferior y el usuario debe colocarlos en sus respectivos lugares. Si el usuario falla escuchará una señal de fallo y tendrá que volver a intentar colocar el diente. Si el usuario acierta el diente se colocará en su sitio y se le facilitará el siguiente diente a colocar. El usuario continuará colocando dientes hasta que los haya colocado todos.

Durante la fase de interacción el usuario puede acceder a una vista más detallada del diente que tiene que colocar pulsando la tecla “zoom”. En esta opción el diente se muestra ocupando casi la totalidad de la pantalla y aparecen unas flechas que permiten al usuario girar el modelo tridimensional del diente hacia un lado o hacia el otro para poder observar mejor su morfología. El usuario puede salir en cualquier momento de esta vista pulsando el botón “Salir” situado en la esquina superior derecha de la pantalla.

La Ilustración 11 muestra una escena con el fondo neutro y la Ilustración 12 muestra una escena con el fondo real. Todo el proceso de navegación e interacción lo realizará el usuario usando sus manos.



Ilustración 11 – Fondo neutro



Ilustración 12 – Fondo real

La aplicación usa como motor principal de gráficos OSG, e integra tres librerías adicionales:

- Mirage SDK: Esta librería se emplea para lograr la visualización autoestereoscópica. Este SDK se utiliza para calcular la visualización autoestereoscópica de una escena OSG, realizando los cálculos necesarios para generar las 8 vistas requeridas y entrelazarlas.
- Open NI: Esta librería se emplea para poder usar Kinect. Con esta librería se calibra a los usuarios, se identifica qué usuario va a jugar y detecta durante la actividad la posición de las manos del usuario que está jugando.
- Fmod: Con esta librería se reproducen los distintos sonidos que la aplicación emitirá en cada momento del juego.

3. Estudio de las tecnologías empleadas

3.

3.1 Open Scene Graph (OSG)	24
3.2 Wrapper OSG – Android	26
3.3 Realidad Aumentada.....	28
3.4 Tecnología 3D	33

3.1 Open Scene Graph (OSG)

OpenSceneGraph (OSG) es una librería de alto rendimiento para trabajar con gráficos tridimensionales, está publicada como código libre y ha sido integrada en aplicaciones libres y comerciales de todo tipo, [OSG14]. Es capaz de manejar entornos tridimensionales complejos y representar gráficos de última generación sin ningún tipo de limitación. A diferencia de otras alternativas, OSG únicamente ofrece las funciones de un grafo de escena donde se pueden incluir nodos propios para extender sus funcionalidades básicas de representación. Debido a la no especialización de la librería, puede ser empleada para la creación de todo tipo de aplicaciones científicas o comerciales que necesiten representar información gráfica. Ha sido empleada para crear aplicaciones de visualización, RA, simuladores de vuelo o juegos.

El núcleo de OSG es el empleo de una metodología de grafos de escena, son grafos acíclicos que forman una representación jerárquica de la escena. Esta ordenación permite realizar optimizaciones espaciales y de representación para mejorar el rendimiento de la visualización en escenas complejas. Esto se realiza mediante inspecciones del grafo que permiten visualizar, o no, ramas enteras dependiendo de nuestro criterio de visibilidad y, a partir de la selección generar un orden de visualización que sea óptimo para la renderización en la API. Esto permite al programador abstraerse del uso de los comandos de bajo nivel de las API gráficas y concentrarse a nivel de objetos de escena sin preocuparse del código necesario para generar la visualización

Una de las características de OSG es su arquitectura modular que permite añadir elementos y generar nuevos módulos y plugins para ser empleados en el grafo de escena

permitiendo al programador extender la librería según sus necesidades.

Si estudiamos su estructura, OSG está formada por una serie de pequeñas librerías y plugins.

Ambos extienden la funcionalidad, bien añadiendo efectos y patrones gráficos que se pueden incluir directamente en el grafo de escena, o bien añadiendo la posibilidad de trabajar con tipos de archivos.

Generalmente estos plugins requieren de librerías externas independientes de OSG que se enlazan dinámicamente con el programa en ejecución.

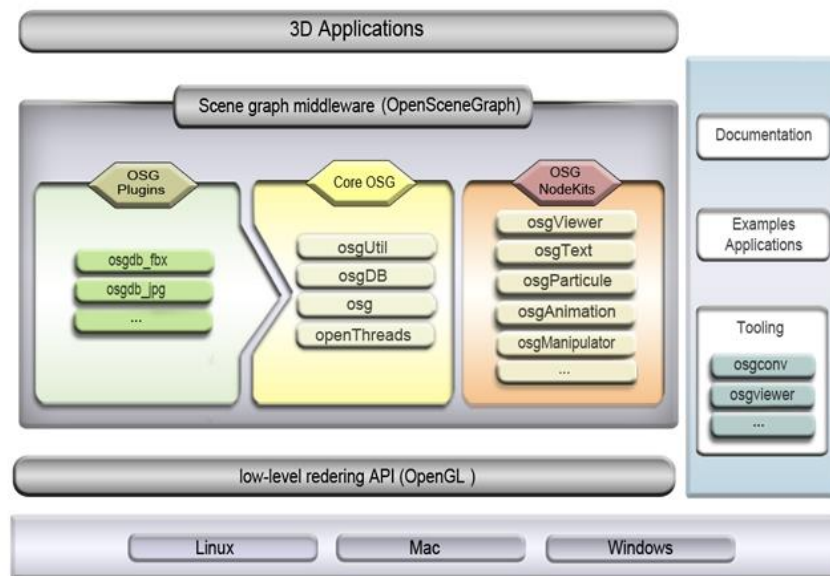


Ilustración 13 - Descripción general de la arquitectura empleada por OSG

La estructura de OSG que podemos ver en la Ilustración 13, está formada por dos partes fundamentales, el núcleo de la librería OSG (core) y un set de NodeKits.

El núcleo o “core” está compuesto por cuatro módulos:

1. `osg`. Este módulo es el núcleo principal de `OpenSceneGraph`, ofrece clases y métodos para la creación del grafo de escena. Contiene clases que representan los distintos tipos de nodos, geometría de la escena, transformaciones geométricas, así como las clases necesarias para las operaciones vectoriales y matriciales 2D y 3D. También incluye un gestor de memoria.
2. `OpenThreads (OT)`. La librería OSG únicamente tiene una dependencia directa, la librería OT. Para simplificar la cantidad de código dependiente de sistemas operativos, OSG encapsula todas las operaciones de hilos en la librería OT. Esta librería ha sido publicada como un proyecto independiente de OSG, aunque se encuentra incorporada en la estructura de la distribución de la librería. De forma opcional, OSG depende de una larga cantidad de librerías para el uso de diferentes formatos de archivos.

3. `osgUtil`. Contiene las funcionalidades necesarias para el renderizado de la escena gráfica. Además optimiza la representación y transforma las llamadas a OpenGL. También proporciona medios para la interacción con la escena.
4. `osgDB`. OpenSceneGraph también viene con una gran colección de cargadores de bases de datos y muchos controladores para manipular formatos 2D y 3D.

El NodeKits es un conjunto de módulos que vienen incorporados en la librería y que intentan solucionar problemas comunes en el uso de grafos 3D. El más destacado es el `osgViewer`.

- `osgViewer`. El módulo `osgViewer` proporciona una forma rápida y fácil de visualizar la escena gráfica. También es una abstracción independiente de la plataforma para diversas interfaces y sistemas operativos.

OSG además se ha diseñado para cargar las librerías y plugins bajo demanda. Esto supone que un programa que emplee OSG, únicamente cargará las librerías básicas y, dependiendo de las necesidades, el resto de librerías y plugins serán enlazados dinámicamente cuando el usuario lo requiera. Esta característica permite ahorrar memoria durante la ejecución de un programa al no tener que cargar los módulos que no se utilicen.

Esta metodología funciona bien en plataformas de sobremesa. Por el contrario en dispositivos embebidos o smartphones resulta, muchas veces, imposible de usar, por ello, OSG incluye la posibilidad de una compilación estática de todas las librerías y plugins. Esta compilación requiere que el usuario registre una serie de macros y así realizar una serie de definiciones internas en la base de datos de OSG que permitan emplear los plugins y librerías sin necesidad de enlazarlos dinámicamente.

Por otro lado, OSG tiene implementadas diversas estrategias para la reducción de la complejidad de una escena tridimensional y la carga de elementos bajo demanda con lo que permite realizar inspecciones completas de los grafos de escena. Con estas posibilidades, se pueden crear unas técnicas de optimización más complejas como el uso de quadrees, octrees u otros modos de ordenación del espacio geométrico tridimensional.

3.2 Wrapper OSG – Android

Para poder hacer uso de la librería OSG en Android hemos tenido que desarrollar un Wrapper para facilitar el uso de las características más importantes de una forma sencilla.

Este Wrapper tiene una parte nativa la cual nos sirve de enlace con la librería OSG haciendo uso de JNI (Java Native Interface) y otra parte Java que es la API que se puede

usar desde aplicaciones Java. En la Ilustración 14 podemos ver la arquitectura empleada.

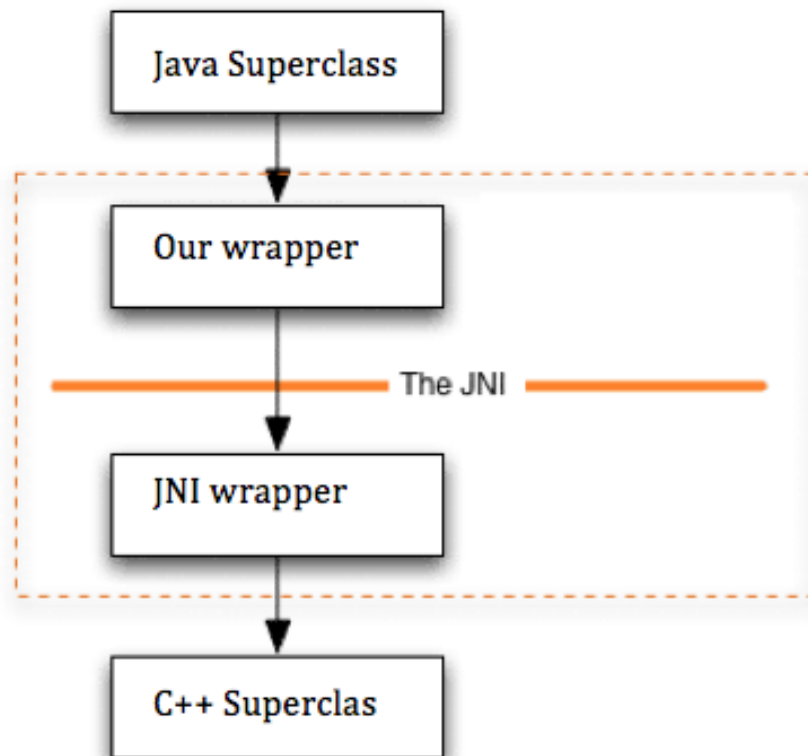


Ilustración 14 - Arquitectura propuesta para el Wrapper realizado

El JNI es un framework de programación que permite que un programa escrito en Java ejecutado en la máquina virtual (JVM) pueda interactuar con programas escritos en otros lenguajes como C, C++ y ensamblador.

Se usa para escribir métodos nativos que permitan solventar situaciones en las que una aplicación no puede ser enteramente escrita en Java, como por ejemplo en el caso de que la biblioteca estándar de clases no proporcione soporte para funcionalidades dependientes de la plataforma. Además puede modificar programas existentes escritos en algún otro lenguaje, permitiéndoles ser accesibles desde aplicaciones Java.

El Wrapper está estructurado en una serie de paquetes que se muestran en la Ilustración 15 y que se vienen a equiparar con la distribución en la que se nos presenta OSG y que ya hemos visto anteriormente.

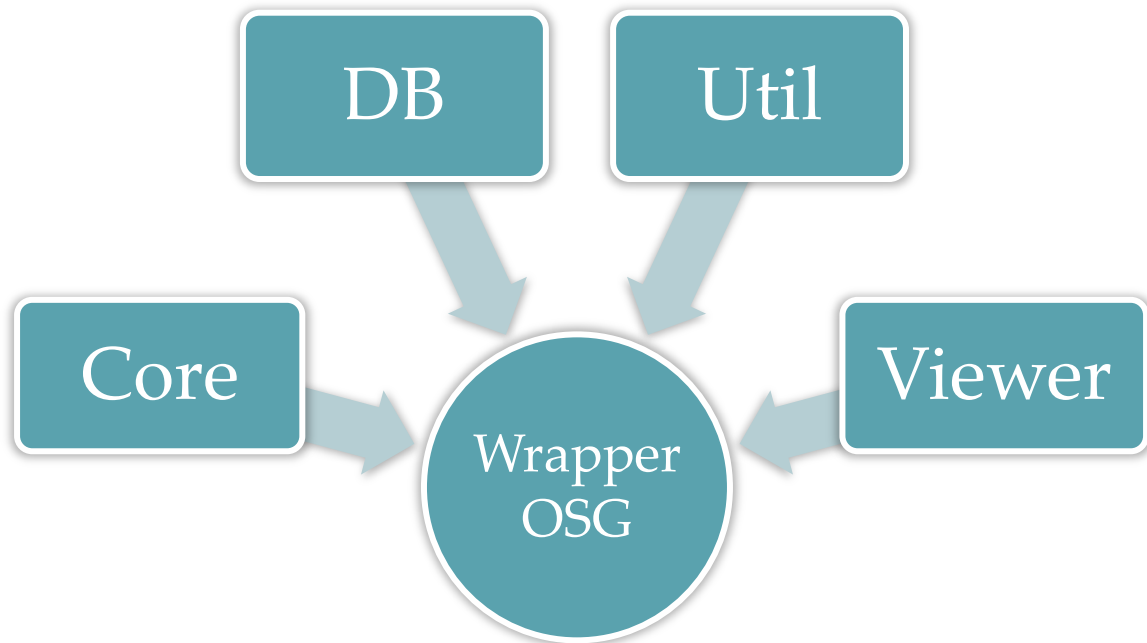


Ilustración 15 - Arquitectura propuesta para el Wrapper realizado

1. core. En el paquete core podemos encontrar los Wrappers correspondientes a los elementos de más uso de la librería Core de OSG. Los nodos más comunes de osg son la clase Node, Group y MatrixTransform. Clases para el manejo de vectores y matrices como Vec3, Vec4 y Matrix. La clase para el manejo de la cámara como es Camera. Y la clase para usar imágenes como es Image.
2. db. En el paquete db podemos encontrar las clases de más uso de la librería osgDB de OSG. En este caso solo hay una clase que es la encargada de cargar los modelos, la clase ReadFile.
3. util. En el paquete util podemos encontrar una serie de clases que nos sirven de apoyo en el desarrollo. Como es la clase MultiviewNode, que genera un nodo con varias cámaras, permitiendo generar el estéreo necesario para su uso en el móvil LG Optimus 3D.
4. viewer. En el paquete viewer podemos encontrar los wrappers correspondientes a los elementos de más uso de la librería osgViewer de OSG. Hay clases como Viewer que sirve como clase de partida para OSG. Contiene todo lo referente a la escena.

3.3 Realidad Aumentada

Existen varias definiciones de lo que es la realidad aumentada. Una de las definiciones más comúnmente aceptadas es la propuesta por Azuma [AZUMA97]:

- Combina elementos reales y virtuales.
- Es interactiva en tiempo real.
- Está registrada en 3D.

En otras palabras, la RA es el término que se usa para definir una visión directa o indirecta de un entorno físico del mundo real, cuyos elementos se combinan con elementos virtuales para la creación de una realidad mixta en tiempo real.

En la Ilustración 16 podemos ver la posición que ocupa la RA respecto al mundo real [MILGRAM94].

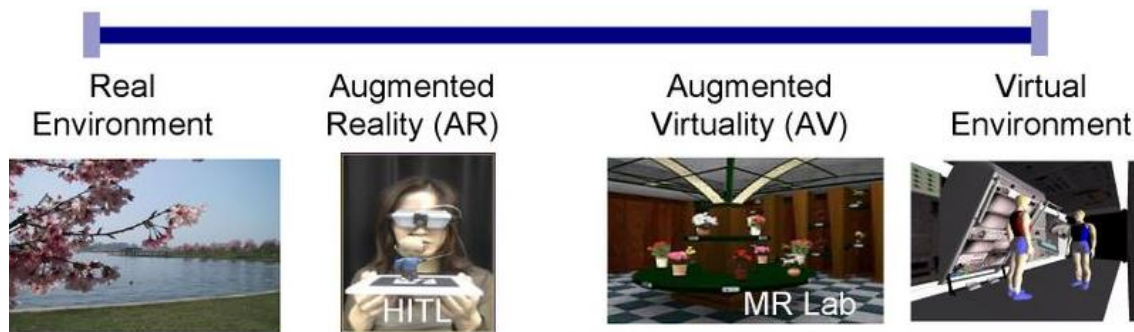


Ilustración 16 – Esquema del Continuo de Milgram

Como podemos observar, a la izquierda de la imagen está el mundo real, es el mundo en el que vivimos, donde los objetos son tangibles. La siguiente imagen corresponde a la RA, en ella existe una gran cantidad de objetos reales pero algunos de ellos son generados por ordenador. La tercera imagen pertenece a la Virtualidad Aumentada. En este caso se combina realidad y virtualidad, siendo la parte de virtualidad mayoritaria. La última imagen representa un entorno totalmente virtual. Aquí el usuario se sumerge por completo en un mundo virtual, en el cual no existe ningún objeto real.

En la Ilustración 17 podemos comparar el mismo elemento en los diferentes estados del Continuo de Milgram



Ilustración 17 – Ejemplo de un objeto en los diferentes estados

En este trabajo profundizaremos en el mundo de la RA. Explicaremos cómo funciona esta tecnología, cuáles son las herramientas principales para implementarla y cuál es el estado del arte de esta tecnología en los dispositivos móviles.

¿Cómo funciona la Realidad Aumentada?

El proceso de RA se puede dividir en dos fases principales: La fase de Reconocimiento & Tracking y la fase de Rendering.

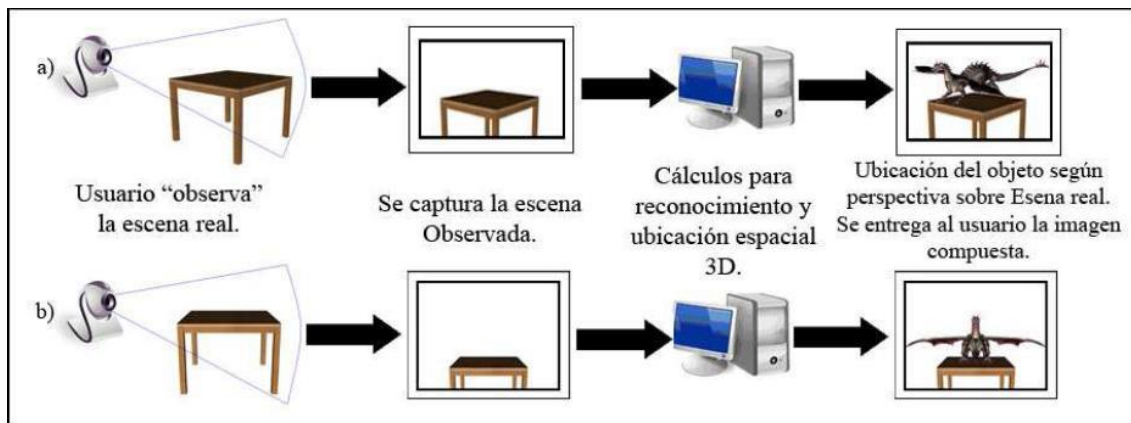


Ilustración 18 – Esquema de funcionamiento de un sistema de Realidad Aumentada típico en función del ángulo de captación.

En la Ilustración 18 podemos observar el proceso completo de generación de RA. La primera parte del proceso consiste en capturar la escena del mundo real. La escena es procesada y es cuando actúa la fase de Reconocimiento & Tracking (o seguimiento) de la pieza.

Esta fase consiste en procesar la escena capturada por la cámara con el fin de solucionar uno de los principales problemas de la RA: el Viewpoint tracking (o Seguimiento del punto de vista). Esto es un problema clave porque la RA debe saber dónde y cómo posicionar los objetos virtuales para su correcta visualización por parte del usuario,

Por lo tanto tenemos dos fases principales, la captura de la escena real y el posicionamiento del objeto virtual en la escena (Viewpoint tracking)

Captación de la escena:

Para capturar el mundo real es indispensable contar con algún mecanismo que permita recoger la escena para que pueda ser posteriormente procesada. Los dispositivos de captura de imágenes son dispositivos físicos que recogen la realidad que deberá ser ampliada. A grandes rasgos, estos dispositivos se pueden agrupar, principalmente, en dos conjuntos:

- Dispositivos video-through: dentro de este grupo se encuentran aquellos dispositivos que realizan la captura de imágenes o video de forma separada a los dispositivos que se encargan de la visualización de la escena. En este conjunto

se encontrarían las cámaras de video o los terminales móviles (siempre y cuando tengan una cámara).

- Dispositivos see-through: son los dispositivos que realizan tanto la tarea de capturar la escena real como de mostrarla con información aumentada al usuario. Estos dispositivos acostumbran a trabajar en tiempo real, haciéndolos no sólo más costosos en presupuesto sino también en complejidad. Cabe remarcar que estos dispositivos llevan años siendo usados, por ejemplo, en los Head Up Displays utilizados por los aviones de combate para mostrar información al piloto sobre altura, velocidad, identificación de blancos, y otros sin necesidad de apartar la vista de la zona frontal de la cúpula de su cabina.

Posicionamiento del objeto virtual (viewpoint tracking):

Existen básicamente dos métodos totalmente diferentes para solucionar el problema del viewpoint tracking, utilizando marcadores o sin utilizarlos.

Reconocimiento por marcadores:

Este método se basa en la detección de lo que se conoce como “markers”. Un marcador es un objeto cuya imagen es conocida por el sistema.

El modo en que el sistema conoce el marcador se pueden agrupar en tres conjuntos, mediante su geometría, su color o mediante ambas características. En la Ilustración 19 podemos ver algunos ejemplos.

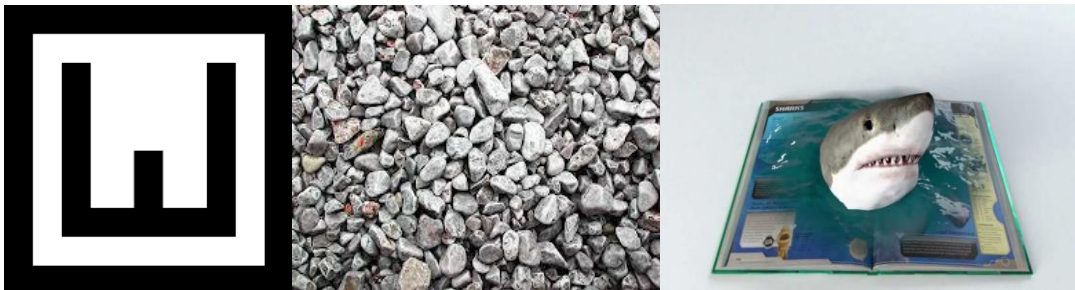


Ilustración 19 - Marcador personalizado, marcador Vuforia y marcador en un libro

El reconocimiento del marcador se suele conseguir utilizando librerías de visión artificial (visión tracking libraries), con este proceso se consigue establecer la posición y orientación de la cámara respecto al centro del marcador. El origen de coordenadas suele ser el centro del marcador. Sistema de coordenadas que se utiliza para mostrar los objetos 3D virtuales. Se muestra un ejemplo en la Ilustración 20.



Ilustración 20 - Ejemplo de RA utilizando un marcador

Reconocimiento sin marcadores:

Existen otras técnicas para la RA que no necesitan un marcador para posicionar el objeto. Un ejemplo de ellas es la localización espacial utilizando GPS.

En base a las coordenadas obtenidas por el GPS, el software calcula la posición de los objetos virtuales y los añade a la escena.

Es sabido que la precisión GPS no es milimétrica sino que conlleva varios metros de margen de error debido a la triangulación de los satélites. Este método por tanto, no puede utilizarse en sistemas de RA donde la precisión sea crítica. Por el contrario, éste es el método más rápido debido a que sólo necesita conocer la posición del GPS, sin tener que pasar por fase de reconocimiento que suele tener un coste computacional más alto. Podemos ver un ejemplo de este método en la Ilustración 21.

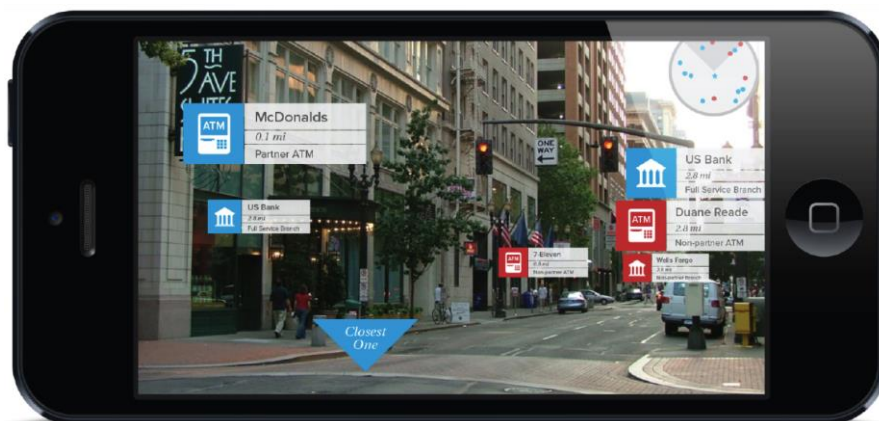


Ilustración 21 - Ejemplo de RA utilizando posicionamiento GPS

En esta tesina nos centraremos únicamente en los sistemas que utilizan marcadores específicos para generar la escena virtual.

3.3.1 Toolkits de realidad aumentada

Existen herramientas que permiten implementar un sistema de realidad aumentada. A continuación se realiza una revisión de los toolkits más conocidos y utilizados:

- AndAR [ANDAR14] es un SDK de código abierto para el desarrollo de aplicaciones de RA para Android basadas en el reconocimiento de marcadores. Utiliza marcadores del tipo ARToolkit. Permite la carga de objetos 3D con formato .obj.
- NyARToolkit [NYARTOOLKIT14] es un SDK de código abierto para el desarrollo de aplicaciones de RA basadas en el reconocimiento de marcadores. Se trata de un framework multiplataforma disponible para Android, Java, C#, AS3, C++ y Processing. Utiliza marcadores del tipo ARToolkit, y dispone de soporte para diferentes formatos 3D (.mqo, .md2, .obj) mediante el uso de la librería min3D.
- Vuforia [VUFORIA14] es una plataforma de desarrollo de aplicaciones de RA para Android e iOS desarrollada por el departamento de I+D de la empresa Qualcomm en Austria. Esta plataforma fue publicada en 2010 y por ejemplo en el último año se han desarrollado más de 1.000 aplicaciones con ella. Grandes marcas comerciales han utilizado esta plataforma para las campañas publicitarias de sus productos. Una de las principales ventajas de esta plataforma es que se basa en el reconocimiento de características naturales, incluyendo objetos 3D, y que existe una extensión para Unity 3D que permite crear escenas virtuales con animaciones muy completas. La plataforma se presenta en código abierto aunque su uso con Unity 3D requiere de la adquisición de la licencia de ésta.

3.4 Tecnología 3D

En este último punto vamos a ver cómo funciona la tecnología 3D. Cuáles son sus principios esenciales y qué elementos necesitamos para reproducirla. Posteriormente veremos una vertiente del 3D convencional, la autoestereoscopia, de la cual haremos uso en este trabajo. Por último estudiaremos el LG Real3D SDK el cual trabaja sobre la plataforma Android y proporciona una API para implementar aplicaciones y contenidos 3D.

3.4.1 Estereoscopia

La estereoscopia es cualquier técnica capaz de recoger información visual tridimensional y/o crear la ilusión de profundidad mediante una imagen estereográfica, un estereograma, o una imagen 3D (tridimensional). La ilusión de la profundidad en una fotografía, película, u otra imagen bidimensional se crea presentando una imagen ligeramente diferente para cada ojo, como ocurre en nuestra forma habitual de ver.

Muchas pantallas 3D usan este método para transmitir imágenes.

El principio biológico en el que está basada la estereoscopia es que los dos ojos, al estar situados en posiciones diferentes, recogen cada uno en sus retinas una imagen ligeramente distinta de la realidad que tienen delante. Esas pequeñas diferencias se procesan en el cerebro para calcular la distancia a la que se encuentran los objetos mediante la técnica del paralaje. El cálculo de las distancias sitúa los objetos que estamos viendo en el espacio tridimensional, obteniendo una sensación de profundidad o volumen. Por lo que si tomamos o creamos dos imágenes con un ángulo ligeramente distinto y se las mostramos a cada ojo por separado, el cerebro podrá reconstruir la distancia y por lo tanto la sensación de tridimensionalidad. En la Ilustración 22 podemos observar este principio biológico.

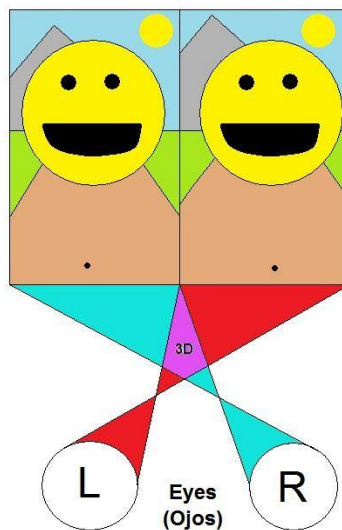


Ilustración 22 - Percepción estereoscópica

Resumiendo, el efecto 3D se debe al ligero distanciamiento entre los dos ojos (entre 50 y 70 mm), de modo que captan las imágenes desde ángulos distintos. La combinación de imágenes individuales vistas por cada ojo consigue una sensación de profundidad y dimensión en la mente. Es lo que se conoce como paralaje.

Pero, ¿cómo capturamos una imagen para luego poder visionarla en 3D?

La solución se encuentra en las cámaras estereoscópicas. Una cámara estereoscópica o cámara 3D, es una cámara capaz de capturar imágenes (fotografías) en tres dimensiones. Estas cámaras, intentan imitar la visión binocular humana, utilizando dos objetivos (o dos cámaras separadas estratégicamente) captando la fotografía en el mismo instante, pero con diferente ángulo, dando como resultado las imágenes 3D.

Una vez grabado el contenido, ¿cómo podemos reproducir este efecto en una pantalla que es plana? ¿Cómo podemos convertir las imágenes en dos dimensiones que estamos acostumbrados a ver en la tele en tres dimensiones?

La forma más empleada y más efectiva para resolver este problema es utilizar un reproductor 3D. Estos reproductores generan las imágenes del ojo izquierdo y derecho intercaladamente. Debido a la velocidad con que se intercalan las imágenes y las lentes utilizadas, cada uno de los ojos del espectador recibe una imagen, con puntos de vista

distintos, haciendo que el cerebro interprete la profundidad mediante la fusión de las imágenes.

Para poder hacer distinción de las dos imágenes que nos llegan intercaladas existen dos tecnologías que nos ayudarán en este proceso. Las dos generan el mismo efecto 3D pero cada uno utiliza una forma.

3D Pasivo:

En esta tecnología el proyector irá mostrando imágenes donde la imagen para el ojo izquierdo se fusiona con la del ojo derecho gracias a unas gafas polarizadas.

En esta tecnología el proyector emite 2 imágenes simultáneas, una para el ojo derecho y otra para el ojo izquierdo. Utilizando unas gafas determinadas, logramos que cada ojo vea solo su imagen. Esa fusión se puede hacer tanto horizontalmente como verticalmente.

Tipo de gafas para el 3D Pasivo:

Gafas de anaglifos:

Las imágenes de anaglifo se componen de dos capas de color, superpuestas pero movidas ligeramente una respecto a la otra para producir el efecto de profundidad. La imagen estará compuesta por dos imágenes filtradas por color, rojo - azul (cian) normalmente.

Para producir imágenes de anaglifo seguiremos estos pasos:

- Tomar la foto derecha con el filtro que no deje pasar el rojo
- Tomar la foto izquierda con el filtro que no deje pasar ni el verde ni el azul
- Montar las imágenes

Gracias a las gafas de anaglifo (Ilustración 20), las cuales incorporan un cristal de cada color (rojo – azul), cada ojo percibirá una imagen diferente, ya que la longitud de onda de cada color es distinta. Finalmente el cerebro superpondrá las dos imágenes y creará el efecto 3D. Fue una de las primeras técnicas empleadas, pero dejó de utilizarse ya que es un sistema poco efectivo ya que limita la gama de colores considerablemente.



Ilustración 23 – Anaglifos

Gafas polarizadas:

Las gafas polarizadas (Ilustración 23) son la evolución de las gafas anáglifos para solventar el problema de la limitación de colores. Estas gafas siguen el mismo principio que las anteriores, pero en este caso cada lente tiene una polarización diferente (suele ser horizontal y vertical) con lo que filtran las ondas de luz y las proyecta a ángulos distintos para cada ojo. Así conseguimos que cada ojo vea su determinada imagen y se cree el efecto 3D. En la Ilustración 24 podemos ver un ejemplo.



Ilustración 24 – Gafas polarizadas

3D Activo

En el 3D activo las gafas incorporan su propia electrónica mediante la cual sincronizamos la imagen que vamos a ver. El televisor irá mostrando a una gran velocidad fotogramas para el ojo izquierdo y derecho. Las gafas evitarán que el ojo derecho vea las imágenes para el izquierdo y viceversa. Todo a una velocidad tan elevada que nuestro cerebro no es capaz de notar el cambio. Así se interpreta como una sola imagen y se crea el efecto tridimensional como podemos ver en la Ilustración 25.

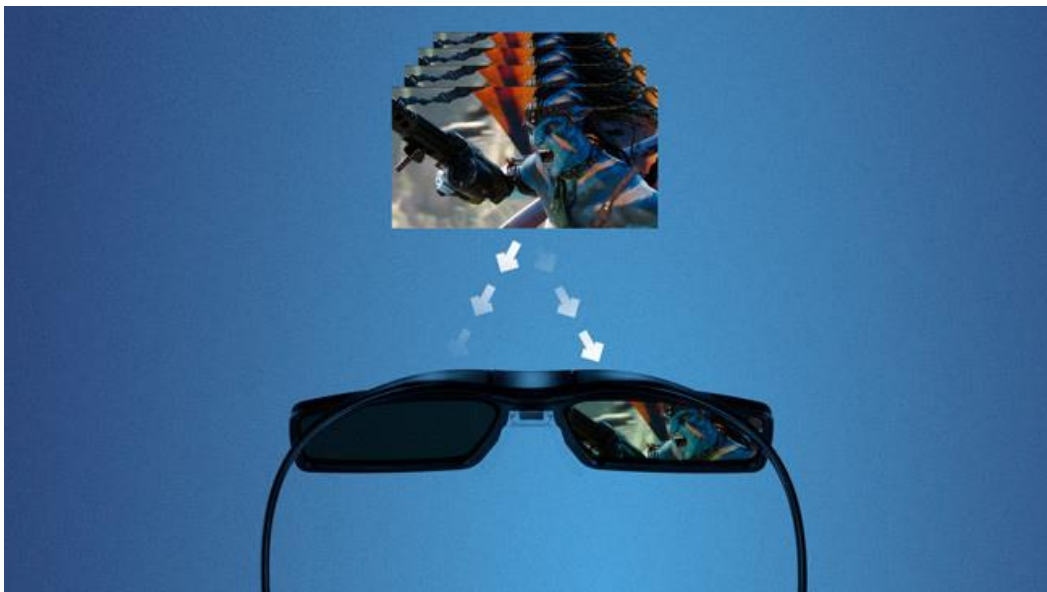


Ilustración 25 – 3D Activo

Tipo de gafas para el 3D Activo:

Gafas activas:

En estas gafas cada lente es independiente, cada una de ellas contiene un obturador de cristal líquido (LCS), el cual está sincronizado con la pantalla. Las gafas hacen que las lentes sean transparentes u opacas, en función de la imagen que está proyectando.

3.4.2 Autoestereoscopia

La autoestereoscopia es el método para reproducir imágenes tridimensionales que puedan ser visualizadas sin que el usuario tenga que utilizar ningún dispositivo especial (como gafas o cascos especiales) ni necesite condiciones especiales de luz. Esta tecnología se conoce popularmente como 3D sin gafas.

Se calcula que el 8-10% de la población no puede percibir 3D a través del efecto estereoscópico, por tener menor visión en uno de los dos ojos, aunque frecuentemente no son conscientes de este hecho hasta que no prueban uno de estos sistemas.

El objetivo principal de esta tecnología es ofrecer un sistema para representar objetos tridimensionales, pero en este caso sin necesidad de utilizar ningún objeto especial para conseguirlo, como sucedía en los casos anteriores.

Existen varios métodos que explican el funcionamiento de esta tecnología:

Barreras de paralaje

Los métodos de la barrera de paralaje ya se utilizaban desde principios del Siglo XX. El estereograma de paralaje consiste en una rejilla vertical fina puesta delante de una imagen especialmente diseñada. La rejilla se hace normalmente de un material opaco con hendiduras verticales transparentes y finas con un espaciamiento regular. Cada hendidura transparente actúa como ventana a un trozo vertical de la imagen puesta detrás de ella. El trozo que vemos depende de la posición del ojo [FLACK07].

La imagen del estereograma de paralaje está hecha interpolando las columnas a partir de dos imágenes. Esta imagen y la rejilla vertical están alineadas de modo que el ojo izquierdo pueda ver solamente las tiras de la imagen para el ojo izquierdo y el ojo derecho pueda ver solamente las tiras de la imagen para el ojo derecho. La Ilustración 26 muestra el funcionamiento de una barrera de paralaje.

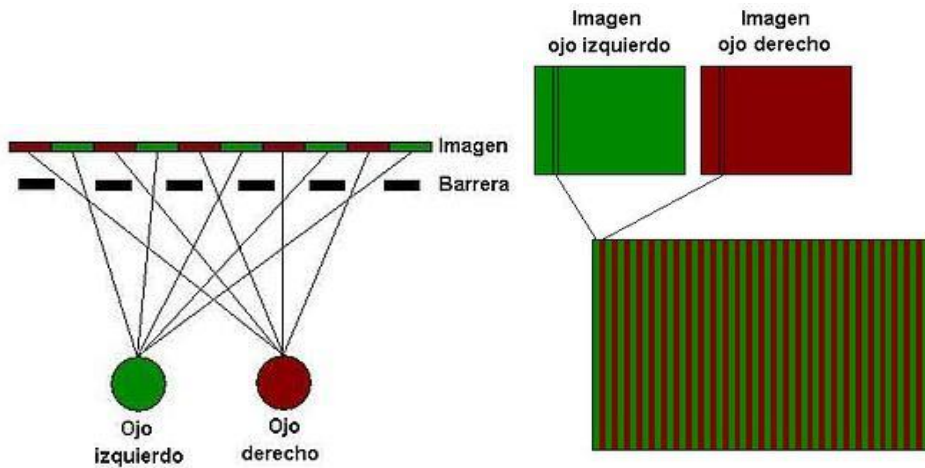


Ilustración 26 - Barreras de paralaje utilizando dos imágenes

Originalmente, estas pantallas trabajaban solo con 2 vistas, enviando la imagen del ojo izquierdo hacia la izquierda, y del ojo derecho hacia la derecha, como en el caso de la figura anterior. Este sistema permitía percibir el volumen cuando cada ojo se situaba en el centro de la pantalla, pero el efecto no podía percibirse desde otras posiciones.

Actualmente casi todos los sistemas autoestereoscópicos son multivista: trabajan con varias imágenes (de 5 a 9 es lo habitual) visibles a diferentes ángulos. Esto permite percibir el efecto desde diferentes posiciones. La Ilustración 27 muestra el funcionamiento de una barrera de paralaje multivista.

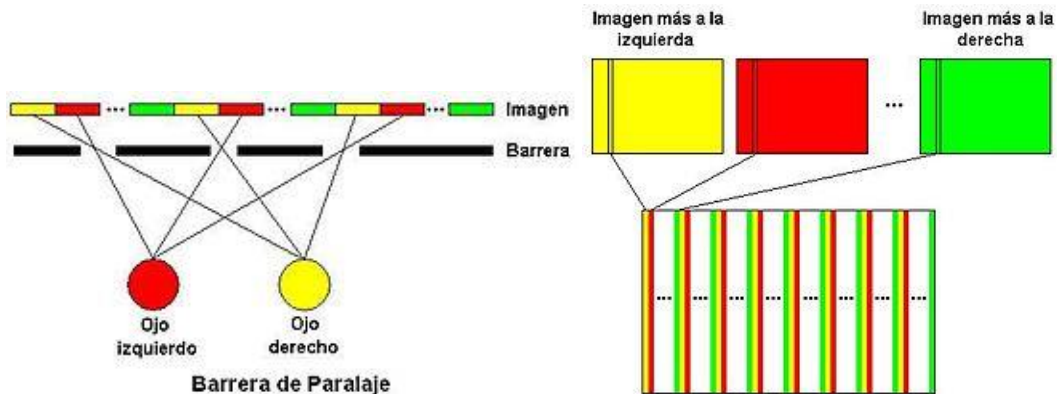


Ilustración 27 - Barreras de paralaje multivista

El espectador puede mover su cabeza de lado a lado y ver diversos aspectos de la escena 3D, excepto en una cierta posición donde los ojos ven los pares incorrectos. Esto ocurre en la transición de una tira de la imagen más a la izquierda a una tira de la imagen más a la derecha.

Este efecto es reducido al mínimo usando una gran cantidad de sub-imágenes con un ángulo muy pequeño entre ellas o manteniendo una profundidad pequeña de la visión. Desafortunadamente el número de imágenes está definido por la resolución de pantalla y de la barrera.

Hoja lenticular

En vez de utilizar una barrera, puede utilizarse una hoja lenticular, la cual es una hoja de lentes finas y largas. Esta hoja lenticular contiene una serie de lentes cilíndricas moldeadas en un sustrato de plástico. Se diseña la imagen trasera para enfocar la línea de la vista de cada ojo sobre diversas tiras y se forma esencialmente de la misma manera que para un estereograma de paralaje.

La clave para la creación acertada de las imágenes autoestereoscópicas basadas sobre estas lentes es la calidad y la uniformidad de la lente. A diferencia que con el método de barrera, la superficie entera de la lente irradia luz, por lo que no hay zonas opacas. La Ilustración 28 muestra el funcionamiento de una hoja lenticular.

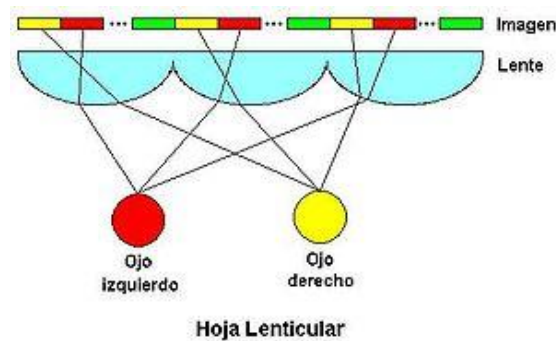


Ilustración 28 - Hojas lenticulares

La función principal de estas lentes es la creación de pantallas automultiescópicas. Estas pantallas están provistas de hojas lenticulares las cuales irradian las imágenes en varios ángulos debido a la difracción que sufre la luz al atravesar la lente. Esto permite que varios usuarios puedan visualizar un contenido 3D utilizando una única pantalla [CHEN13].

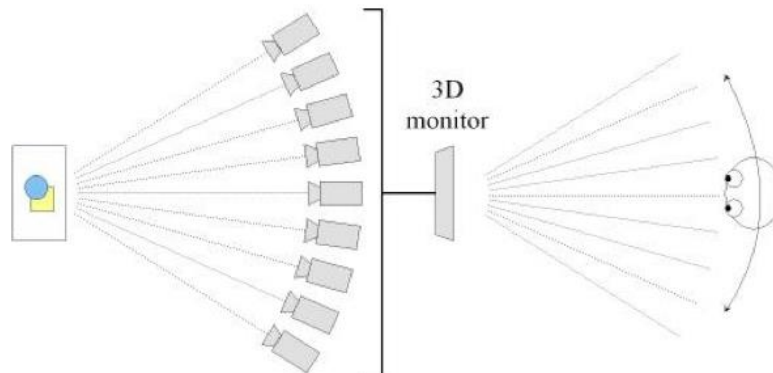


Ilustración 29 - Pantallas automultiescópicas

La hoja lenticular permite la visualización de nueve vistas 3D (Ilustración 29). Está compuesta por lentes semicilíndricas que utilizan la difracción de la luz para dirigir las vistas en una cierta dirección.

3.4.3 LG Real3D SDK

El LG Real3D SDK trabaja sobre la plataforma Android y provee una API para implementar aplicaciones y contenidos 3D. Está disponible de forma gratuita en el SDK Manager de Android [REAL3D].

Para ver imágenes 3D sin utilizar gafas, la pantalla debe tener una barrera de paralaje puesta delante. Gracias a esta barrera, las imágenes se emiten en ángulos diferentes. El API Real3D nos ayuda a controlar el hardware de entrelazado que incorpora la pantalla de nuestro móvil LG.

La Ilustración 30 muestra cómo funciona el LG Real3D.

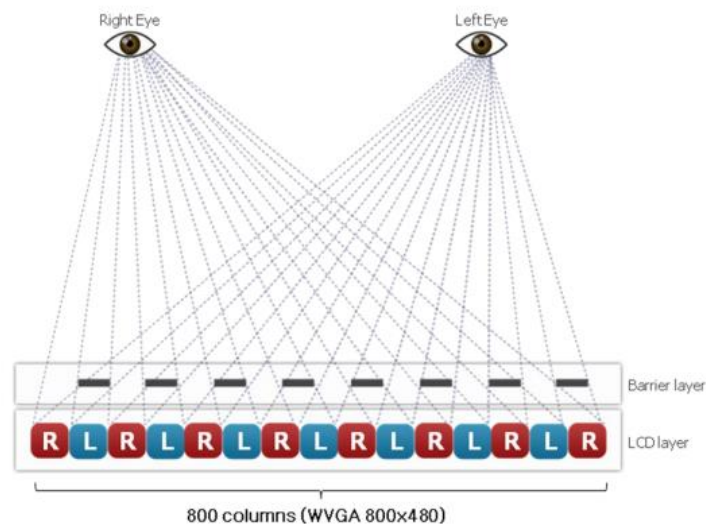


Ilustración 30 – Funcionamiento del móvil estereoscópico LG

Como se aprecia en la imagen, esta barrera de paralaje lo que hace es mostrarle al ojo izquierdo la imagen izquierda y al ojo derecho la imagen derecha. Consiguiendo así el efecto 3D deseado.

Las imágenes de las que se alimenta el API para realizar el entrelazado son en formato “side by side”. Una imagen al lado de la otra (ojo izquierdo, ojo derecho) en el mismo frame. Una vez se obtiene la imagen que queremos mostrar, el API se encarga de entrelazarla de manera correcta para generar la imagen 3D final.

Los distintos formatos de entrada que admite el API son los siguientes:

- Horizontal split: imágenes del mismo frame 3D posicionadas en posición horizontal. Por lo tanto queda la imagen del ojo izquierdo en la parte izquierda y la del ojo derecho en la derecha.
- Vertical split: Igual que el caso anterior pero en vez de estar partidas de forma horizontal, están distribuidas de forma vertical.
- Entrelazado: Imágenes donde el ojo izquierdo se coloca en las líneas o columnas impares y las del ojo derecho en las líneas o columnas pares. Es el modo más parecido a la imagen que el API genera al final y es el que utilizaremos.

4. Desarrollo

4

4.1	Arquitectura del sistema	41
4.2	Hardware.....	41
4.3	Software	42
4.4	Detalles de la implementación	48
4.5	Problemas encontrados	53

4.1 Arquitectura del sistema

En este cuarto capítulo explicaremos con detalle cuál es la infraestructura empleada en nuestro sistema, tanto la parte hardware como la software. En la primera parte hablaremos del móvil que hemos utilizado, cuáles son sus características principales y explicaremos por qué hemos elegido este dispositivo. En la segunda parte hablaremos de los programas que hemos utilizado para el desarrollo, cómo funcionan y de qué forma los hemos empleado para alcanzar nuestro objetivo. Por último explicaremos en profundidad de qué forma hemos implementado ambos sistemas. El primero de ellos tendrá dos vertientes, una parte que funcionará de una forma autoestereoscópica y otra parte que funcionará sin autoestereoscopia. En ambas aplicaciones podremos representar un molar mediante un modelo OSG. En el segundo sistema haremos el mismo procedimiento pero añadiéndole un componente de Realidad Aumentada. Veremos qué dificultades hemos tenido a la hora de realizar el trabajo y qué ventajas e inconvenientes plantea cada sistema.

4.2 Hardware

En este trabajo el Hardware utilizado para hacer el estudio comparativo entre los sistemas autoestereoscópicos y no autoestereoscópicos ha sido el dispositivo móvil LG Optimus 3D P920. Las características principales son, pantalla LCD capacitiva de 4.3 pulgadas a 480 x 800 pixels de resolución; Procesador TI OMAP4 doble núcleo 1GHz, GPU PowerVR SGX540; 8 GB memoria interna, 512 MB RAM. El sistema operativo es Android OS, versión 4.0.4. En la Ilustración 31 podemos ver qué forma tiene el dispositivo.



Ilustración 31 - LG Optimus 3D P920

Es uno de los pocos móviles capaces de emitir imágenes y videos de forma autoestereoscópica. Además es posible realizar fotos y grabar videos en 3D gracias a la doble cámara que incorpora.

4.3 Software

4.3.1 3D Max

Autodesk 3Ds Max es un programa de creación de gráficos en 3D. Es una de las aplicaciones más utilizadas tanto para el modelado 3D como para la creación de videojuegos, entre otras muchas funcionalidades.

Los modelos que hemos empleado en este trabajo han sido creados utilizando las herramientas proporcionadas por 3D Max. Las mallas nos las han proporcionado el Centro Folguera-Vicent. Estas mallas tienen un formato STL (estereolitografía), el cual contiene un gran número de polígonos. Debido a que el número de polígonos era excesivo para que el móvil pudiera renderizar bien la escena, se han reducido con la función ProOptimize. Una vez conseguido un número de polígonos aceptables tanto para el móvil como para que la calidad de la imagen sea buena, hemos suavizado los bordes y las superficies utilizando la función Smooth. En la Ilustración 32 podemos ver el resultado final del molar 47.

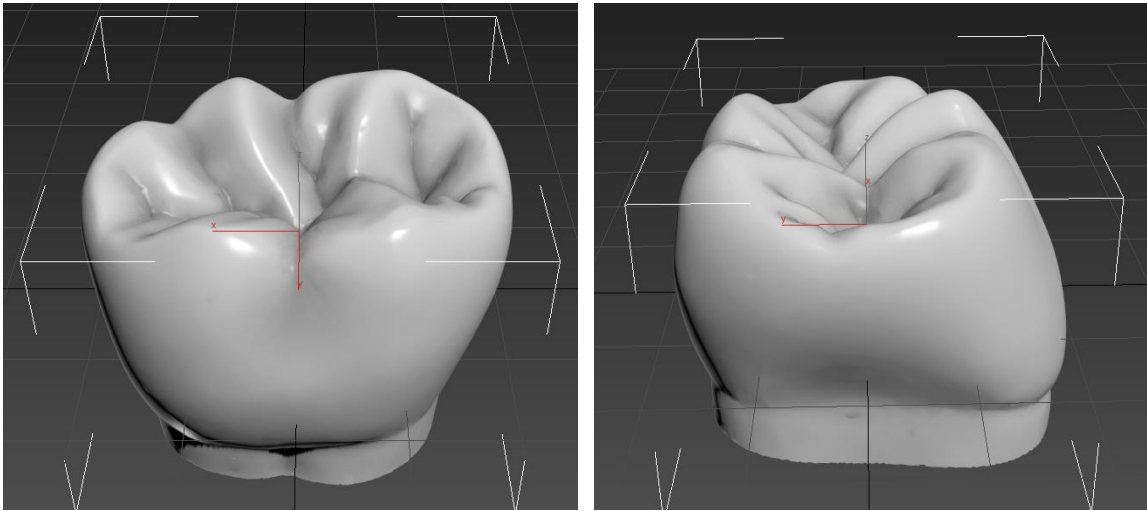
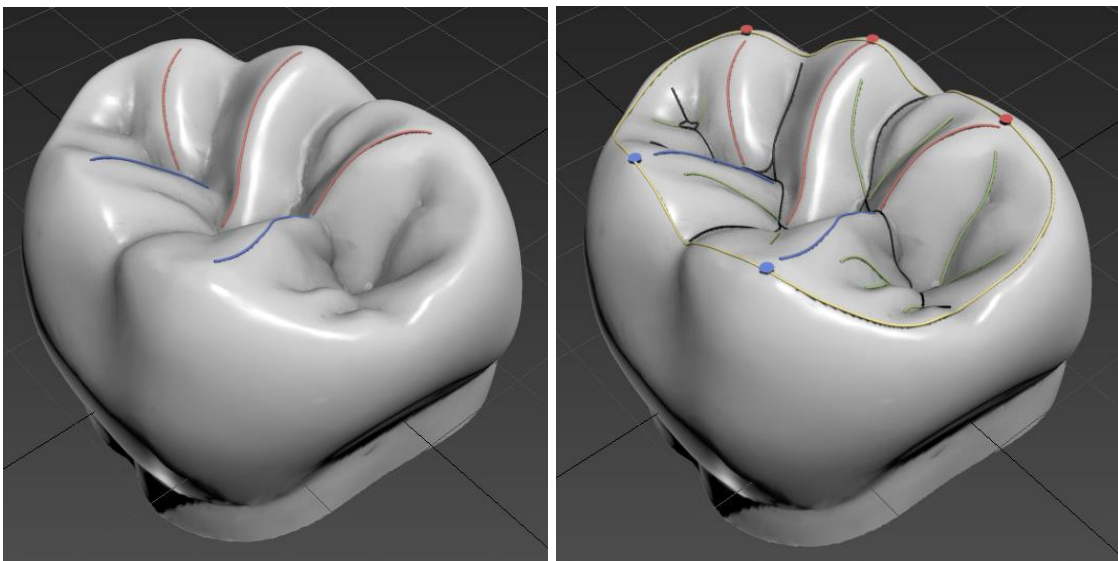


Ilustración 32 – Molar 47 en 3D Max desde diferentes ángulos

Para la creación de los modelos que representan la morfología del molar (líneas de colores) hemos utilizado la técnica “Conform Mesh”, que lo que hace es proyectar el volumen de una malla en otra. Con esto conseguiremos recrear las líneas de las partes más significativas adaptándolas al contorno del molar. En la Ilustración 33 podemos ver la leyenda de todos los elementos morfológicos que hemos incluido.










	Contorno Oclusal		Aristas Triangulares
	Crestas Triangulares		Surcos Suplementarios
	Fosas y Surcos Oclusales		Cúspides no Céntricas
	Cúspides Céntricas		

Ilustración 33 – Creación de los modelos y leyenda de los elementos morfológicos

Todo el trabajo realizado ha sido supervisado por profesores expertos del Centro Folguera-Vicent para validar la correcta representación del molar y de sus partes fundamentales.

4.3.2 NDK de Android

El NDK de Android es un conjunto de herramientas que permiten embeber código máquina nativo compilado en lenguajes C y/o C++. Los desarrolladores pueden reutilizar código legado, hacer programación de bajo nivel hacia el hardware, o diferenciar sus aplicaciones aprovechando características que no es posible o práctico acceder con otros mecanismos. En nuestro caso la elección de la NDK viene dada por la necesidad de cargar librerías escritas en C y que no podemos utilizar directamente en Java.

El funcionamiento general de un sistema que hace uso del NDK es el siguiente. La Máquina Virtual de Android (VM) permite que el código de la aplicación (escrito en Java) llame a métodos implementados en código nativo a través de JNI. El JNI es una interfaz que nos permite hacer llamadas a código escrito en C desde Java mediante un sencillo sistema de nombrado de los métodos.

En nuestro caso, tenemos una librería, OSG, escrita en C. Nosotros necesitamos utilizarla en nuestra aplicación Android. Gracias a las herramientas del NDK, la máquina virtual del sistema Android, será capaz de llamar a los métodos nativos de nuestra librería, utilizando las instrucciones realizadas por el JNI.

A continuación veremos con más detalle qué estructura de carpetas hemos utilizado en nuestro sistema y de qué forma hemos hecho uso de las herramientas del NDK.

La estructura de nuestro trabajo consta de cinco carpetas principales:

- AR : Librerías de Realidad Aumentada
 - AndAR: Librerías y archivos necesarios para compilar y usar esta librería.
 - NyArtookit-and: Librerías y archivos necesarios para usar esta librería.
 - Vuforia-1-5-9: Librerías y archivos necesarios para usar esta librería.
- org.openscenegraph.osgandroid: Wrapper de la librería OpenSceneGraph para Android.
- org.openscenegraph.osg2D: Aquí albergaremos la aplicación que únicamente utiliza la librería OSG para renderizar el molar.
- org.openscenegraph.osg3D: Será igual que la anterior, pero en este caso combinaremos la librería OSG con las funcionalidades que nos proporciona el SDK del Real3D.
- org.openscenegraph.osgRA: En este paquete tendremos la aplicación que combina OSG + RA + Real 3D haciendo uso del paquete osgAndroid e integrando la funciones del Real3D y las librerías de RA incorporadas en el SDK.

4.3.3 Dependencias entre Proyectos

A continuación se explica la lista de proyectos y sus dependencias entre ellos.

- AndAR. Esta librería no depende de nadie, teniendo una parte nativa donde está incluida la librería ARToolKit.
- NyARToolKit. Esta librería tampoco tiene dependencias. A diferencia de AndAR, ésta no tiene dependencias nativas. Ya que el código se ha implementado totalmente en Java.
- Vuforia. Esta librería no tiene dependencias. Está implementada de forma nativa, siendo obligatorio hacer un Wrapper JNI de la misma para su uso. Además el código es completamente cerrado, lo que impide hacer modificaciones a la misma.
- org.openscenegraph.osgAndroid. Wrapper de la librería OpenSceneGraph. Al igual que Vuforia, la librería está íntegramente desarrollada en nativo. Siendo necesario generar una serie de wrappers para realizar su manejo desde Java.
- org.openscenegraph.osg2D y org.openscenegraph.osg3D. Nuestras dos aplicaciones hacen uso de la librería org.openscenegraph.osgAndroid.
- org.openscenegraph.osgRA. Este paquete contiene una serie de actividades que muestran el funcionamiento de cada una de las distintas librerías de realidad aumentada que se han explicado anteriormente. Esta librería depende tanto de cada una de las librerías de RA, como también de osgAndroid y de la librería Real3D (la encargada de manejar el estéreo).

Como se ha explicado anteriormente la librería org.openscenegraph.android es un Wrapper de la librería nativa OpenSceneGraph (OSG). Por lo tanto tiene una parte nativa la cual nos sirve de enlace con la librería OSG haciendo uso de JNI y otra parte Java que es la API que se puede usar desde aplicaciones Java.

El Wrapper está estructurado en una serie de paquetes que se vienen a equiparar con la distribución en la que se nos presenta OSG. Todos estos paquetes se encuentran dentro del paquete org.openscenegraph.osg. La estructura queda de la siguiente forma.

- org.openscenegraph.osg.core
- org.openscenegraph.osg.db
- org.openscenegraph.osg.util
- org.openscenegraph.osg.viewer

Para hacer uso de todas estas herramientas el primer paso es crear en la carpeta raíz del proyecto una carpeta llamada JNI, que incluirá dos ficheros. El primero de ellos es Android.mk, en el cual se especifican todos los archivos de código C/C++ nativos a ser

compilados, el encabezado y el tipo de compilación. El segundo fichero es Application.mk, cuya finalidad principal es describir qué módulos o librerías son necesarias en nuestra aplicación. Por último se añaden los archivos .h y .cpp, los cuales contendrán las llamadas y las implementaciones de las funciones nativas que necesitamos. En la Ilustración 34 vemos el resultado final de la carpeta JNI.

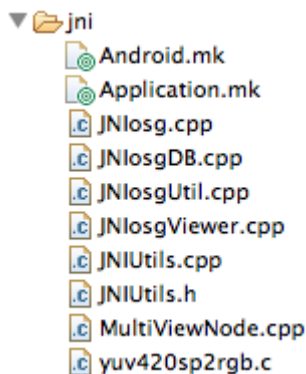


Ilustración 34 - Carpeta JNI de nuestro proyecto

Una vez tengamos esta estructura, compilaremos los ficheros para construir la librería que será cargada desde Java. Para ello se utiliza la herramienta ndk-build, ejecutando este comando:

```
./ndk-build -C /ruta/a/nuestro/proyecto
```

Esta sentencia genera la librería, copiándola al directorio correcto dentro de nuestro proyecto. El resultado lo vemos en la Ilustración 35.

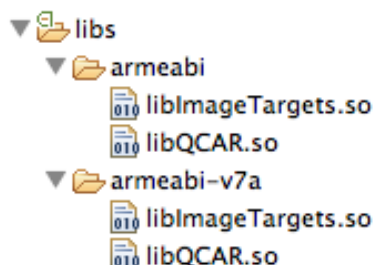


Ilustración 35 – Compilación de la librería OSG

Una vez que tenemos la librería compilada y generada, podemos llamarla desde Java. En nuestro caso necesitaremos incluir el paquete org.openscenegraph.osg en todos los proyectos que queramos hacer uso de la librería de OSG.

4.3.4 Uso de OSG y Real 3D

En este apartado veremos cómo integrar la librería OSG a nuestro proyecto y cómo utilizar el SDK que proporciona LG, el Real 3D.

Lo primero será incorporar el paquete que contiene la librería OSG a nuestro proyecto. Una vez incluido, nos descargaremos desde el repositorio oficial de Android el Real 3D SDK. Una vez instalado estaremos listos para crear nuestra aplicación.

Para facilitar el trabajo de implementación se ha creado una clase principal (OSGStereoActivity) de la cual heredan todas las actividades que van a hacer uso de OSG y de la API de LG Real3D. En esta clase encontramos una serie de métodos que inicializan y configuran OSG y Real3D. Gracias a esta clase las funciones principales se implementarán en un único sitio y se pueden utilizar en los hijos de una forma sencilla y clara. En la Ilustración 36 se muestra el diagrama de la clase OSGStereoActivity.

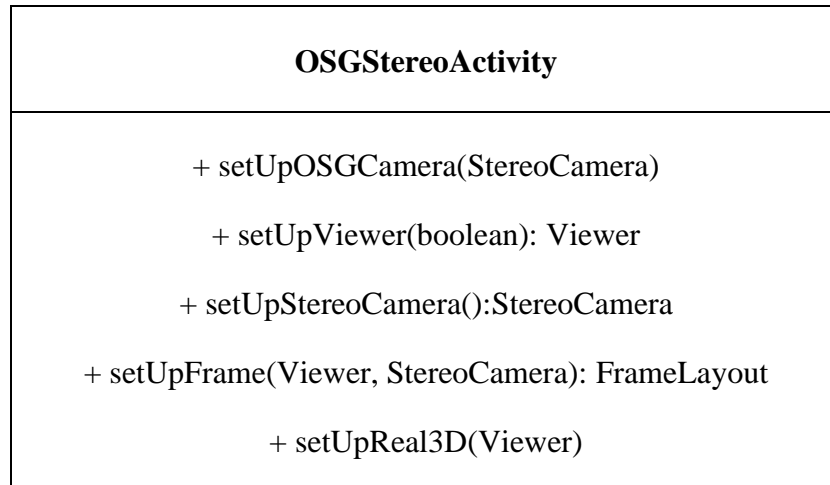


Ilustración 36 – Diagrama de la clase OSGStereoActivity

Los métodos más destacados son:

- setUpOSGCamera(StereoCamera camera): método que se encarga de configurar la cámara osg, ya que todos los sistemas de realidad aumentada están pensados para ser usados con OpenGL. Es necesario configurar la cámara de OSG igual que la de OpenGL.
- Viewer setUpViewer(boolean isStereo): Método que se encarga de inicializar el viewer y dependiendo del valor de la variable isStereo lo pone en estéreo o no. Devuelve el Viewer creado.
- StereoCamera setUpStereoCamera(): método que crea una instancia de la clase StereoCamera y se la devuelve. Esta clase usa la librería Real3D para abrir la cámara. Según los parámetros de entrada se abren las dos cámaras o solo la del ojo izquierdo. En caso de estéreo realiza el entrelazado adecuadamente para que funcione en los móviles autoestereoscópicos LG.
- FrameLayout setUpFrame(Viewer viewer, StereoCamera camera): Método que configura el frame que se va a añadir a la vista de Android. Recibe como parámetro el viewer y la cámara que se ha creado. De esta manera devuelve un FrameLayout que se puede añadir al content View de Android.
- setUpReal3D(Viewer viewer): Método que pone a la librería Real3D entre el viewer y la salida de pantalla. De esta forma la salida de OSG es capturada por la librería y se realiza el entrelazado para el móvil.

4.4 Detalles de la implementación

En este apartado veremos cómo se han utilizado las funciones anteriores y en qué orden las hemos implementado para cada uno de los sistemas.

4.4.1 Sistema 1: OSG + Real 3D

Esta aplicación representará el molar 47 de una forma dual, es decir, se podrá visualizar tanto en 2D como en 3D. Para facilitar el manejo a los alumnos y las pruebas que debemos realizar, se han separado en dos aplicaciones diferentes. La primera renderizará el molar y las partes morfológicas fundamentales en 2D y la segunda lo hará en 3D. La interfaz está compuesta por una vista y siete botones. En la Ilustración 37 podemos ver cómo se han distribuido los diferentes elementos.

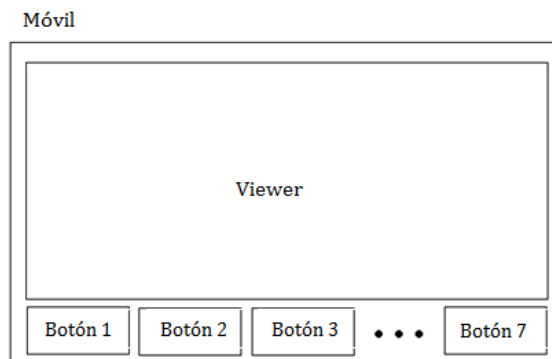


Ilustración 37 – Interfaz propuesta para la aplicación

Para conseguir esta interfaz hemos hecho uso de las funcionalidades principales de nuestro Wrapper OSG. Para ello hemos creado un Viewer que extiende de la clase Viewer de la librería OSG. Este Viewer permite asignarle una matriz transformada como si fuera nuestra escena principal. Gracias a esto tendremos la posibilidad de ir añadiendo o quitando modelos (OSG Nodes) a la escena. En la Ilustración 38 podemos ver un grafo en el cual se explica con detalle el funcionamiento de la aplicación.

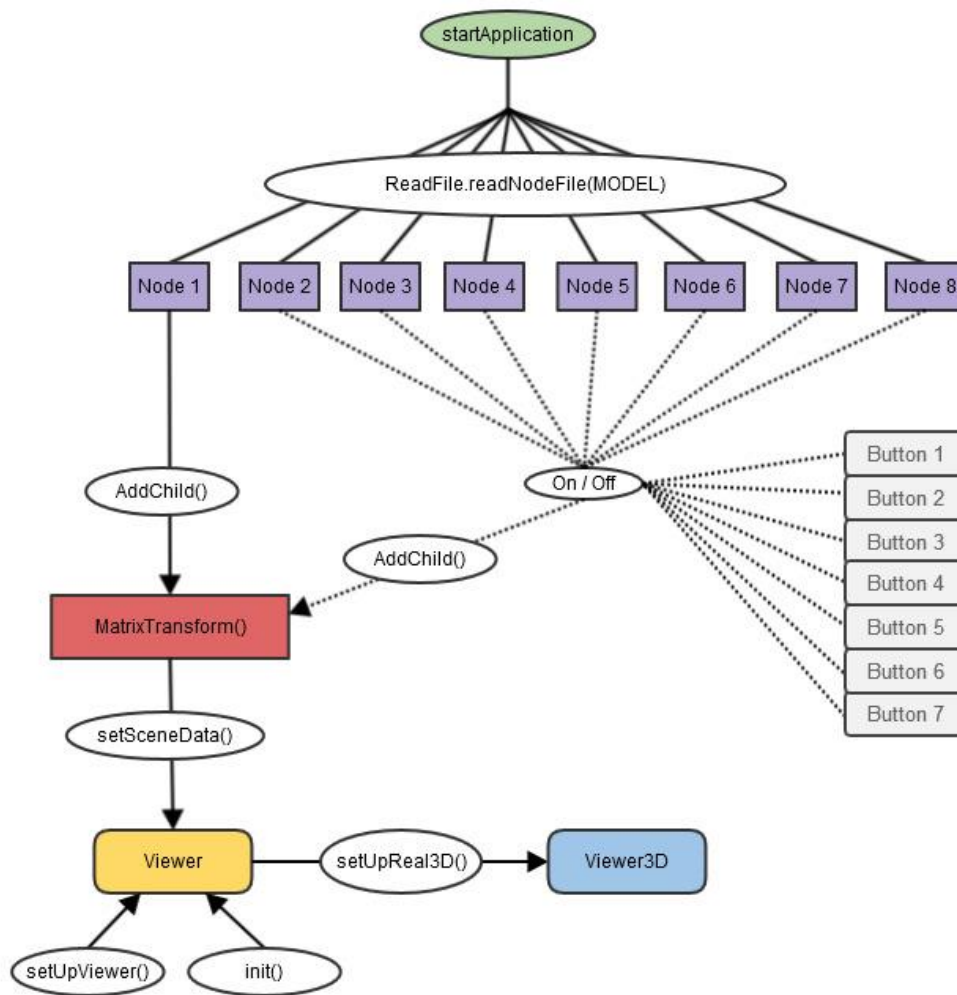


Ilustración 38 – Arquitectura del sistema OSG – Real 3D

Cuando se inicia la aplicación, se leen todos los modelos utilizando la función `ReadFile.readNodeFile(MODEL)`, donde `MODEL` será la ruta de la SD donde están almacenados nuestros modelos. El Node 1 contendrá el molar vacío sobre el cual se irán añadiendo los otros modelos los cuales contendrán las partes morfológicas más importantes. Se añadirá el primer Nodo a la matriz transformada que hemos creado y añadiremos la matriz al Viewer. Antes de añadir el Nodo habremos configurado nuestro Viewer a través de la función `setUpViewer()` y `setUpOSGCamera()`. Esta función asignará los parámetros más óptimos a nuestra vista y configurará el modo estéreo que queramos. Para finalizar llamaremos a la función `setUpReal3D()` la cual activa o desactiva el modo 3D en nuestro dispositivo.

Para ir añadiendo o quitando el conjunto de nodos que tenemos, iremos activando y desactivando los botones que tenemos en nuestra interfaz.

A continuación podemos observar el resultado final que hemos obtenido en nuestro sistema. Las imágenes que mostramos a continuación son las imágenes correspondientes al sistema 2D, ya que no es posible representar imágenes 3D en este documento. En la Ilustración 39 y 40 vemos los resultados obtenidos.

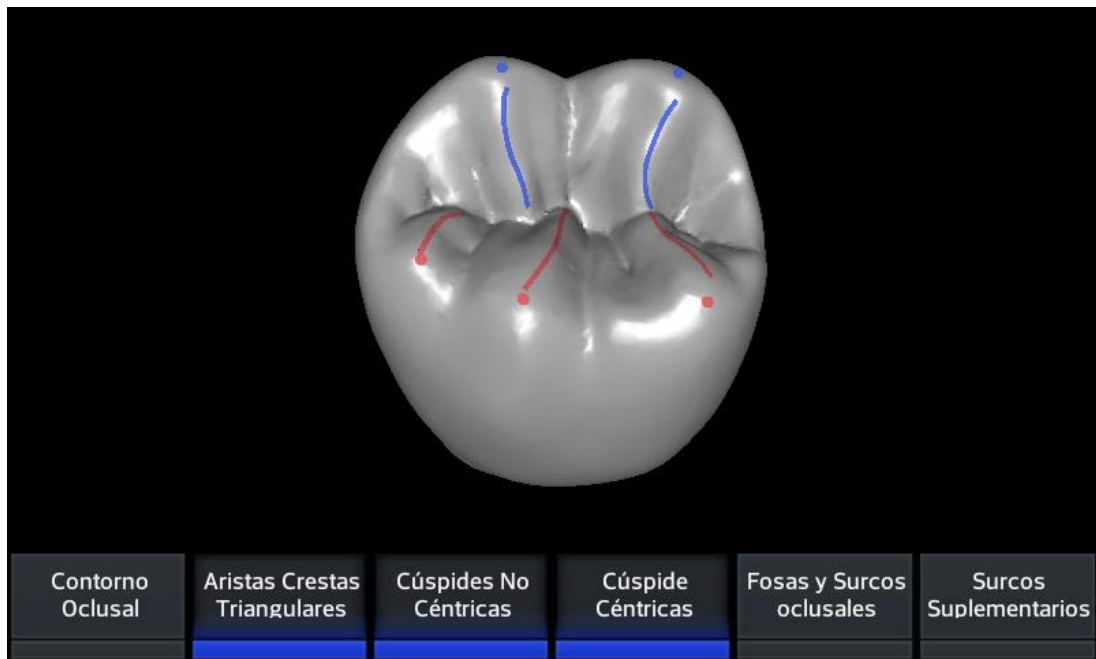


Ilustración 39 – Resultado final obtenido por el primer sistema. Parte 1

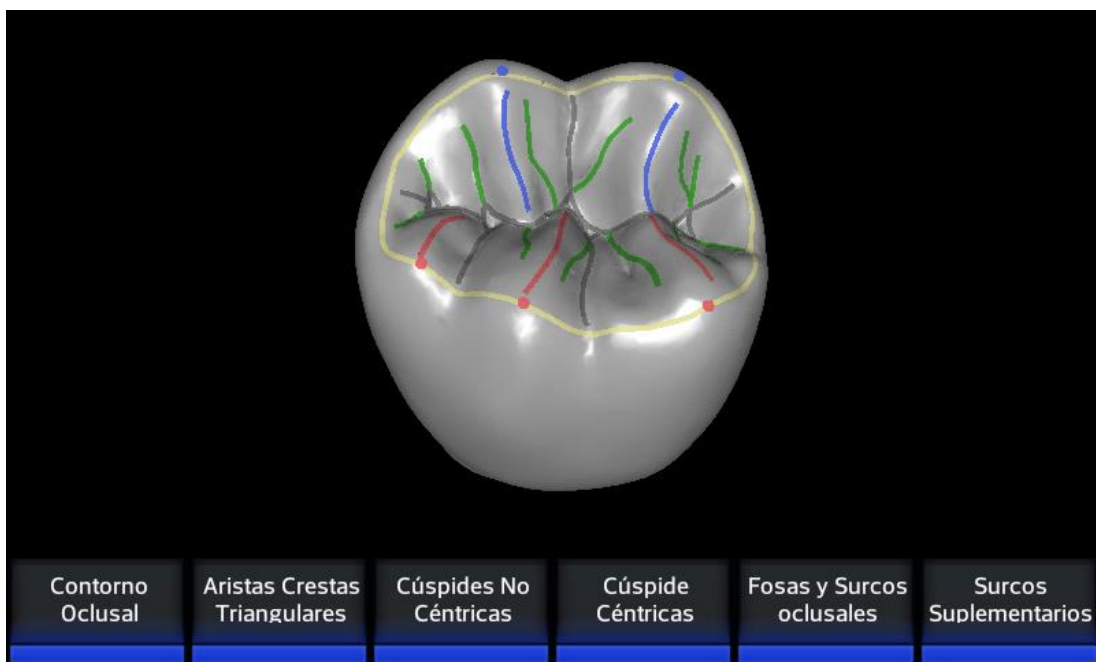


Ilustración 40 - Resultado final obtenido por el primer sistema. Parte 2

4.4.2 Sistema 2: OSG + Real 3D + RA

En este segundo sistema hemos combinado la RA con las tecnologías que hemos empleado anteriormente. Para este proyecto disponemos de varios toolkits de RA para poder comprobar los resultados que ofrece cada uno.

Debido a que AndAR y NyARToolKit han ofrecido un resultado poco satisfactorio, hemos decidido desestimar su uso, por lo que en este apartado nos basaremos únicamente en la implementación y funcionamiento del toolkit Vuforia.

Al igual que con la librería OSG, la librería Vuforia está escrita en C++ por lo que deberemos utilizar el NDK para poder compilar la librería como ya hicimos en capítulos anteriores. Además habrá que añadir la ruta del proyecto como una variable de clase dentro de Eclipse para poder hacer uso de la librería en nuestro proyecto.

Al igual que hicimos en el sistema anterior, se ha implementado un Wrapper para poder llamar a las funciones principales de Vuforia y así facilitar el proceso de implementación. En la Ilustración 41 se ha creado un grafo donde podremos ver con más claridad el funcionamiento de la aplicación.

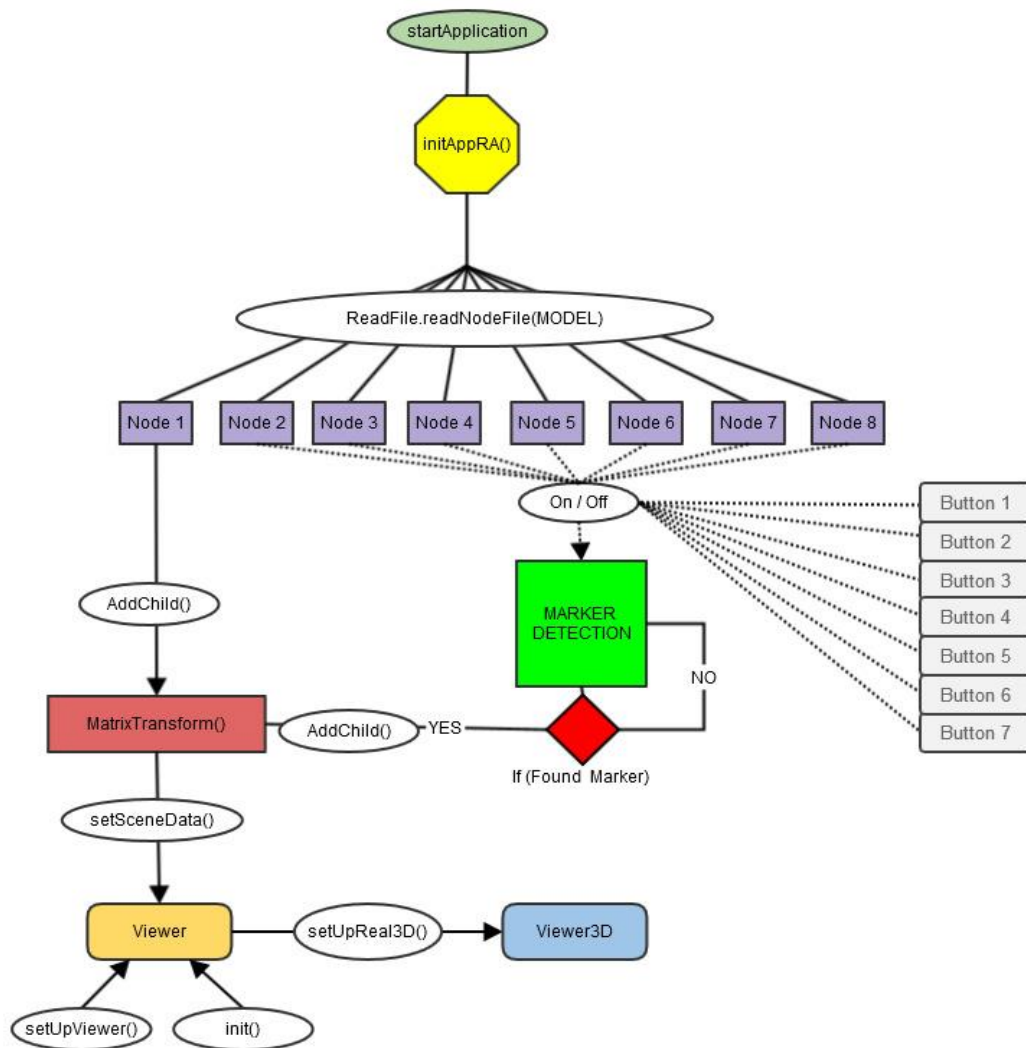


Ilustración 41 - Arquitectura del sistema OSG – Real 3D – RA

Como podemos observar, el funcionamiento es similar al sistema anterior, con la excepción de que aquí lo primero que haremos será inicializar la librería de Vuforia. La inicialización de esta librería difiere un poco de las anteriores.

En la clase ImageTargets (una de las clases del Wrapper implementado) existe un método llamado synchronized void updateApplicationStatus(int appStatus). Este método es el encargado de inicializar todos los estados necesarios al inicio de la aplicación.

Los distintos estados por los que va pasando la aplicación son los siguientes:

- APPSTATUS_INIT_APP Inicia la aplicación.
- APPSTATUS_INIT_QCAR: Inicia el SDK de QCAR.
- APPSTATUS_INIT_TRACKER: Inicia los trackers.
- APPSTATUS_INIT_APP_AR: Inicia la aplicación de RA.
- APPSTATUS_LOAD_TRACKER: Carga los trackers que se van a utilizar. Es el equivalente a los marcadores en el resto de aplicaciones.
- APPSTATUS_INITED: Lanza la aplicación de RA. En este estado es en el que se añaden al surface los distintos contenidos. Se puede observar como se añade el Viewer de OSG, justo después de añadir el surface de vuforia. A diferencia de las otras librerías, Vuforia controla y renderiza la cámara de forma automática.
- APPSTATUS_CAMERA_RUNNING: Inicializa la captura de la cámara del dispositivo y configura la cámara 3D.

Además se ha añadido un estado general en el cual se comprobará que el marcador está detectado y es visible para el usuario. Esta detección estará activa durante todo el ciclo de la aplicación y nos servirá para poder añadir o quitar Nodos en el caso que la detección del marcador sea correcta. En la Ilustración 42 y 43 vemos los resultados obtenidos.



Ilustración 42 - Resultado final obtenido por el segundo sistema. Parte 1



Ilustración 43 - Resultado final obtenido por el segundo sistema. Parte 2

4.5 Problemas encontrados

Como vimos en anteriores capítulos, el dispositivo utilizado permite representaciones en 3D de forma autoestereoscópica gracias a las barreras de paralaje que contiene la pantalla. Esto significa que en una única imagen se concatenan las dos imágenes que forman el 3D, haciendo que la barrera de paralaje separe cada imagen hacia su correspondiente ojo. Esto no tiene porque acarrear ningún problema, siempre y cuando el usuario que está viendo el contenido centre el foco de visión en el objeto principal y mantenga siempre el mismo ángulo de visión.

En nuestro primer sistema no nos afectará el problema anterior, ya que el usuario dejará el móvil en una superficie fija y siempre mantendrá el mismo ángulo de visión. En cambio en nuestro segundo sistema el usuario deberá enfocar el móvil hacia el marcador de RA para que el objeto aparezca, y deberá ir desplazando el móvil para ir captando las diferentes partes del objeto. Esto hará que el ángulo de visión respecto al móvil vaya variando haciendo que la imagen sea borrosa y poco nítida.

Debido a este problema, hemos desestimado este segundo sistema como herramienta de aprendizaje, ya que en este caso, la experiencia de usuario no es la más óptima.

Por lo tanto, la validación se realizará sobre el primer sistema, utilizando sus dos vertientes, autoestereoscópica y no autoestereoscópica.

5. Validación

5.

5.1 Introducción.....	54
5.2 Diseño	54
5.3 Procedimiento.....	55
5.4 Resultados	57
5.5 Preferencias	60
5.6 Resultados de satisfacción	62

5.1 Introducción

El objetivo de este estudio es determinar si hay una diferencia significativa en el proceso de aprendizaje al utilizar un sistema autoestereoscópico y no autoestereoscópico. Para ello analizaremos y compararemos nuestros dos sistemas, evaluando a más de treinta alumnos del Centro Folguera-Vicent.

En este capítulo se explica con detalle qué tipo de alumnos han participado, qué mediciones se han realizado durante la evaluación, los pasos que hemos seguido para realizar el procedimiento y el análisis de los resultados obtenidos en base a las observaciones y comentarios realizados por los participantes.

5.2 Diseño

5.2.1 Participantes

Un total de 34 alumnos del Centro Folguera-Vicent participaron en nuestro estudio. El estudio contó con 13 hombres (38.24%) y 21 mujeres (61.76%). El rango de edades oscila entre 18 y 40 años. La media de edad es de 25.12 ± 5.98 años. En este estudio han participado dos especialidades, 25 estudiantes que cursan la especialidad de Prótesis (73.53%) y 9 estudiantes que cursan la especialidad de Higiene (26.47%).

5.2.2 Cuestionarios

Para recabar toda la información del procedimiento realizado, hemos empleado cinco cuestionarios. Los hemos llamado de la siguiente manera:

- Pre-Test (Q1) – Éste será el primer cuestionario que rellenarán todos los alumnos que van a participar en la prueba. En él se encuentra dibujada la superficie de un molar en el cual deberán rellenar las partes morfológicas más importantes. El objetivo de este test es determinar el conocimiento inicial que tiene el alumno sobre la materia. También deberá rellenar unos campos con sus datos personales: nombre, edad, sexo, especialidad y grupo.
- Test de estereoscopía (Q2) – En este test determinaremos si el alumno está dotado de una visión estereoscópica. El test elegido para este fin se llama Test de Lang. Consiste en una placa de plástico, tamaño postal, en la que la persona ha de ver unas imágenes en relieve a una distancia de 40 cm. La lámina contiene cuatro imágenes (elefante, estrella, luna y camión), situadas a diferentes niveles. Estas imágenes solo se pueden reconocer si se tiene una visión binocular, a excepción de la estrella, la cual se puede ver de forma normal. Por lo tanto los alumnos que no sean capaces de distinguir al menos una de las figuras estereoscópicas, deberán abandonar el procedimiento.
- Post Test (Q3) – Es el mismo test que el anterior. Se lo entregaremos al alumno una vez haya utilizado uno de los dos sistemas. Nos servirá para evaluar si ha habido un aumento en el conocimiento de la materia.
- Test de usabilidad (Q4) – Este test cuenta con 12 preguntas en las cuales se le pide al usuario que valore de 1 a 5 cuál ha sido su experiencia en el uso de la aplicación.
- Test comparativo entre sistemas (Q5) – Éste será el último test que realice el alumno. Aquí el usuario valorará que sistema le ha gustado más y con cuál se ha sentido más identificado. Además contendrá preguntas de respuesta libre, donde los participantes pueden darnos sus críticas y sugerencias.

5.3 Procedimiento

Hemos distribuido a los estudiantes en los siguientes dos grupos:

Grupo	Prótesis	Higiene	Grupo	Prótesis	Higiene	Total
A1	7	3	A2	6	2	18
B1	6	2	B2	6	2	16
Total	13	5		12	4	34

Tabla 1 Distribución de los alumnos

Lo primero que deberán hacer todos los alumnos (Grupo A y Grupo B) será completar el Pre-Test. Se le proporcionarán a cada alumno los materiales necesarios para que puedan situar las partes morfológicas más importantes en el molar dibujado. Una vez terminen este paso, cada alumno realizará de forma individual el test de estereoscopia.

Los alumnos que no sean capaces de ver ningún objeto en la lámina, no podrán continuar con el procedimiento, ya que esta persona no será capaz de valorar el funcionamiento del sistema 3D ya que su visión no le permite captar el efecto. Una de las causas más frecuentes para que cualquier persona no consiga percibir el efecto 3D es el estrabismo.

Grupo A: Una vez acabado el Pre-Test (Q1) y el test autoestereoscópico (Q2), se le proporcionará a los alumnos toda la información necesaria sobre la aplicación que van a utilizar. En este grupo el primer sistema proporcionado será el 2D. Una vez entienda cómo funciona la aplicación, el alumno podrá utilizarla el tiempo que necesite. Una vez acabe, se le dará el Post-Test (Q3) y un cuestionario de usabilidad (Q4). Cuando termine le dejaremos el sistema 3D autoestereoscópico para que pueda probarlo, y finalmente rellena el cuestionario Q5, comparando ambos sistemas. La Ilustración 44 muestra el procedimiento del grupo A.

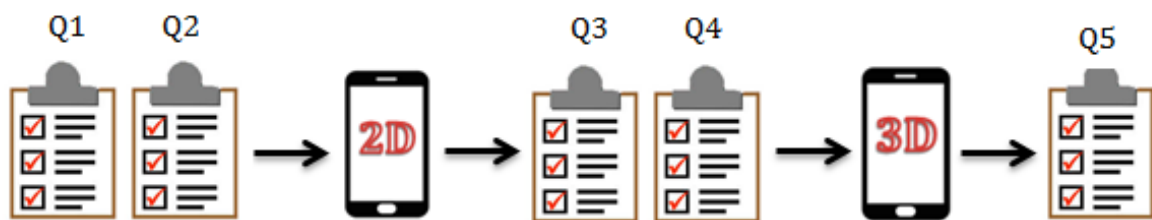


Ilustración 44 – Procedimiento seguido por el grupo A

Grupo B: El procedimiento será el mismo que el del grupo A, pero en este caso el primer sistema que les proporcionemos será el 3D autoestereoscópico y posteriormente se le dejará el sistema 2D como se aprecia en la Ilustración 45.

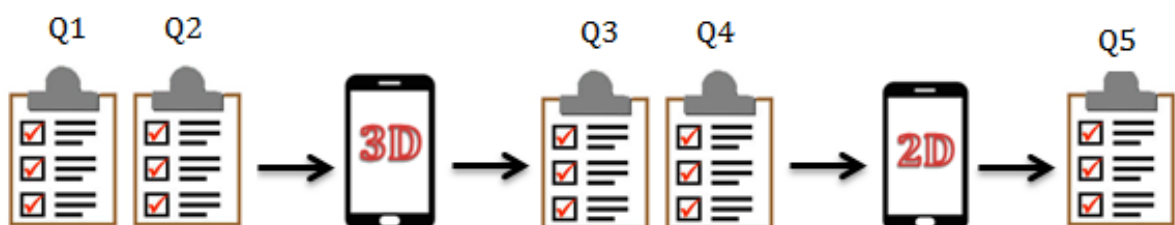


Ilustración 45 - Procedimiento seguido por el grupo B

El objetivo principal de este protocolo es poder determinar qué sistema es el más adecuado y el que mejor se adapta a los alumnos.

5.4 Resultados

Para evaluar los datos obtenidos hemos utilizado un toolkit estadístico open source R usando RStudio [R14] [RSTUDIO14].

5.4.1 Resultados del aprendizaje

En primer lugar comentar los resultados obtenidos en el Test de estereoscopia (Q2), ya que será el que determine cuántos alumnos en total participarán en la prueba. Los que no superaren este test no podrán participar. En la Tabla 2 podemos ver un resumen del número de objetos que fueron capaces de distinguir los estudiantes.

Objetos Identificados	Número de alumnos	Puede participar en el procedimiento
3	29 (85.4%)	Sí
2	3 (8.8%)	Sí
1	1 (2.9%)	Sí
0	1 (2.9%)	No

Tabla 2 Resultado del Test estereoscópico

Solo una persona no fue capaz de distinguir ningún objeto, el 2.9% de nuestro conjunto de estudiantes. Se estima que entre un 5% y un 8% de la población no dispone de visión binocular, lo que en cierta manera se acerca a nuestro porcentaje. Por lo tanto, 33 personas serán las que finalmente participen.

El objetivo principal de este análisis es determinar cuánto han aprendido los alumnos después de utilizar nuestros dos sistemas. Para ello hemos analizado la variable conocimiento, comparando los test Q1 y Q3.

Para comprobar la normalidad de los datos se han utilizado los métodos Shapiro-Wilk y Anderson-Darling. Por los resultados de estos tests no podemos asumir que los datos sigan una distribución normal. Por ello, se han utilizado métodos no paramétricos como el test Mann-Whitney, el test Wilcoxon Signed-rank y el test de Kruskal-Wallis.

Los descriptores de cada grupo se presentan en el formato (*mediana; rango intercuartílico*), y el resultado de los tests se presenta en el formato (*estadístico U/W, aproximación a la normal Z, p-valor, tamaño del efecto r*); y ** indica la existencia de diferencias estadísticas significativas a nivel $\alpha=0.05$.

Comparativa PreTest – PreTest

Pre3D (5; 3) – Pre2D (5; 1): ($U = 151.5, Z = 0.591, p = 0.556, r = 0.103$)

No hay diferencias estadísticas significativas entre ambos tests, los dos grupos tenían un conocimiento similar antes de utilizar el sistema.

Comparativa PreTest – PostTest

Pre3D (5; 3) – Post3D (6; 1): ($W = 6$, $Z = -2.640$, $p = 0.007^{**}$, $r = 0.453$)

En esta comparativa sí que hemos encontrado diferencias estadísticas significativas. Los que han utilizado primero el 3D han adquirido más conocimiento.

Pre2D (5; 1) – Post2D (5.5; 1): ($W = 4$, $Z = -2.741$, $p = 0.008^{**}$, $r = 0.484$)

Al igual que en el caso anterior, existen diferencias estadísticas significativas. Los que han utilizado primero el sistema 2D también han adquirido conocimiento.

Estos resultados indican que los dos sistemas facilitan la adquisición de conocimiento.

Comparativa PostTest – PostTest

Post3D (6; 1) – Post2D (5.5; 1): ($U = 150$, $Z = 0.567$, $p = 0.649$, $r = 0.099$)

En esta comparativa no hemos encontrado diferencias estadísticas significativas. Los dos grupos tenían un conocimiento similar después de utilizar los dos sistemas.

Los resultados de estos tests se corroboran en la Ilustración 46.

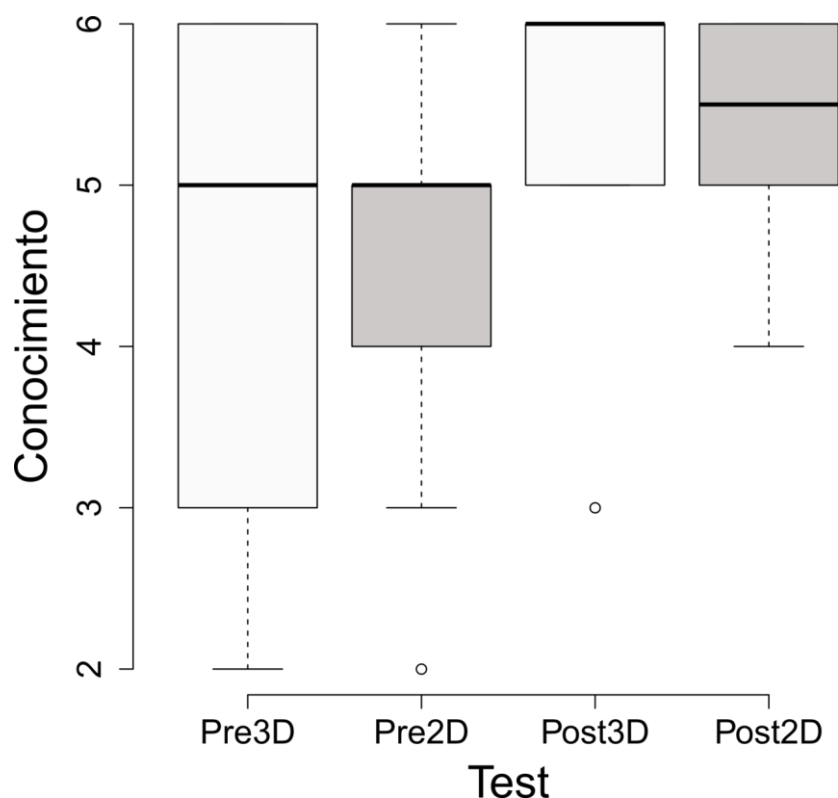


Ilustración 46 – Puntuación obtenida para la variable conocimiento en el Pre y Post Test usando ambos sistemas

Realizamos un test de Kruskal-Wallis para comprobar si hay diferencias entre los distintos factores:

Factors	Kruskal-Wallis χ^2	d.f.	p-value
Género	0.096	1	0.755
Edad	11.961	17	0.802
Sistema	0.321	1	0.570
Máster	2.512	1	0.112
Repetidor	0	1	1.000
Especialidad	1.081	1	0.298

Tabla 3 Resultados obtenidos con el Test de Kruskal-Wallis para los diferentes factores utilizados

No hay diferencias estadísticas significativas entre los distintos factores. Con esto podemos concluir que el conocimiento adquirido no está relacionado con las características individuales del alumno (genero, edad,..) sino que viene determinado por el sistema que se ha utilizado en el proceso. Ambos sistemas han facilitado la adquisición de conocimiento.

Basados en los resultados anteriores hemos representado la interacción entre conocimiento – sistema y conocimiento – especialidad para ambos sistemas.

La Ilustración 47 muestra la interacción entre el conocimiento adquirido y el sistema utilizado, antes y después de su uso. Así como la diferenciación entre sexos.

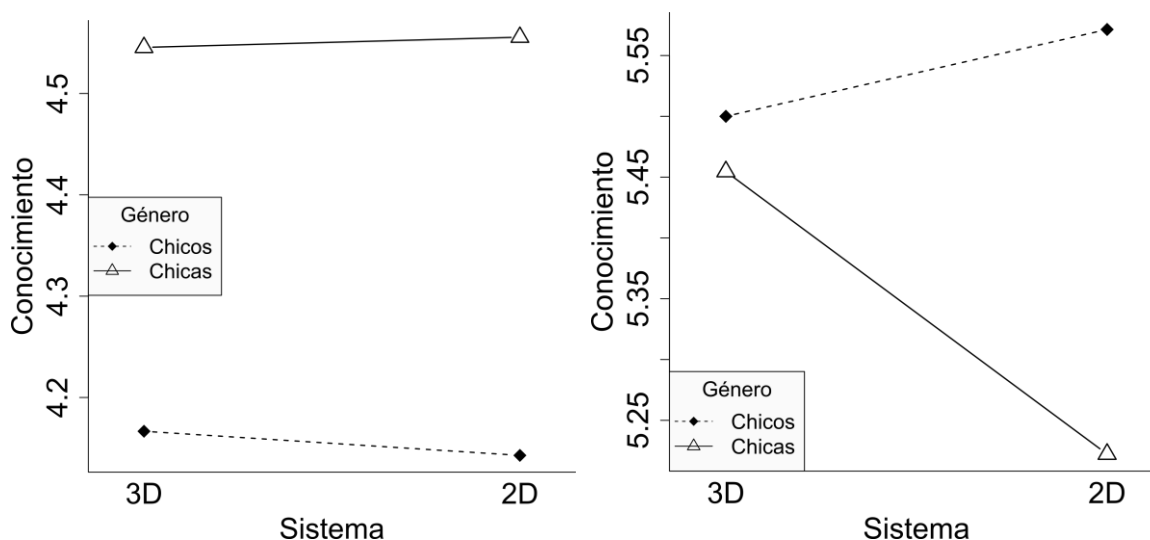


Ilustración 47 - Conocimiento inicial y final, en base al sistema utilizado

Analizando los datos de la Ilustración 43, podemos determinar que los hombres parten con un conocimiento inicial inferior al de las mujeres. En cambio, el conocimiento adquirido después de utilizar alguno de los sistemas es mayor para hombres que para mujeres.

También podemos destacar que los hombres han aprendido más con el sistema 2D. En cambio, las mujeres han aprendido más usando el sistema 3D.

La Ilustración 48 muestra la interacción entre el conocimiento adquirido y la especialidad del estudiante, antes y después de su uso. Así como la diferenciación entre sexos.

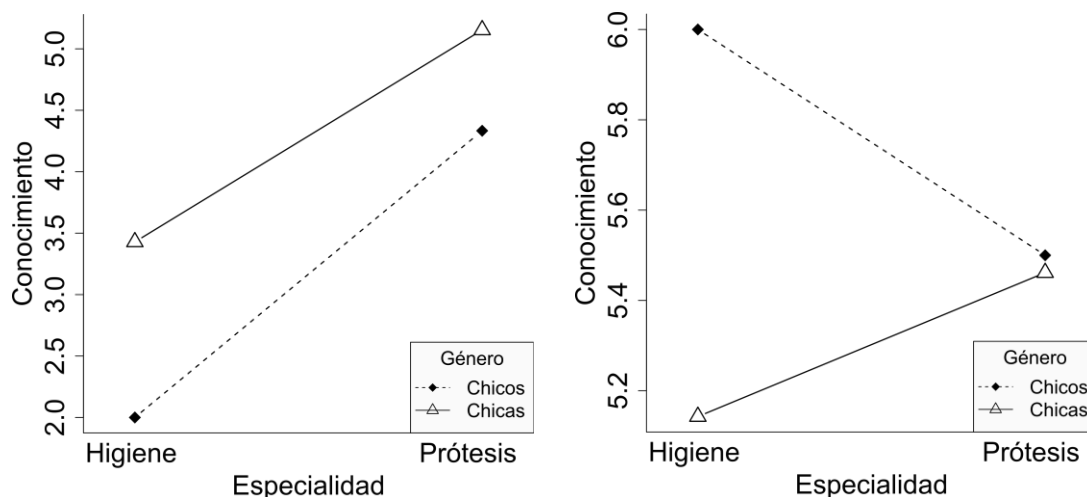


Ilustración 48 - Conocimiento inicial y final en base a la especialidad del alumno

En este grafico podemos observar que los chicos de higiene partían con un conocimiento inferior a los de prótesis, pero después del proceso, su conocimiento ha aumentado mucho más, quedando finalmente por encima de ellos. Para las mujeres el resultado ha sido diferente, ya que el conocimiento inicial en las alumnas de higiene era menor que en las de prótesis, y una vez utilizado uno de los sistemas, su conocimiento se sigue manteniendo por debajo de las chicas de prótesis, aunque en ambas especialidades los resultados han mejorado notablemente.

5.5 Preferencias

El test Q5 contiene cinco preguntas donde el usuario debe marcar qué sistema prefiere. La opción 1 es el sistema 3D y la opción 2 es el 2D.

En la Tabla 4 se representan los datos obtenidos.

#	3D	2D	U	Z	p	r
Q1	1	1	125	-0.485	0.721	0.084
Q2	1	1.5	108	-1.191	0.296	0.207
Q3	1	1	125	-0.485	0.721	0.084
Q4	1	1	113	-1.030	0.327	0.179
Q5	1	1	130	-0.289	0.755	0.050

Tabla 4 Medianas para las preguntas de preferencias, el test Mann-Whitney U y r, tamaño del efecto

Como se puede observar en la Tabla 4 no existen diferencias estadísticas significativas para las preguntas de satisfacción entre los dos grupos, ya que ambos prefieren el sistema 3D.

También hemos querido responder a dos preguntas. La primera de ellas es cuál ha sido el sistema favorito para cada especialidad. En la Tabla 5 y en la Ilustración 49 podemos observar los resultados obtenidos.

	Higiene	Prótesis
3D	62.5	68.0
2D	37.5	32.0

Tabla 5 - Preferencia de sistema por especialidad

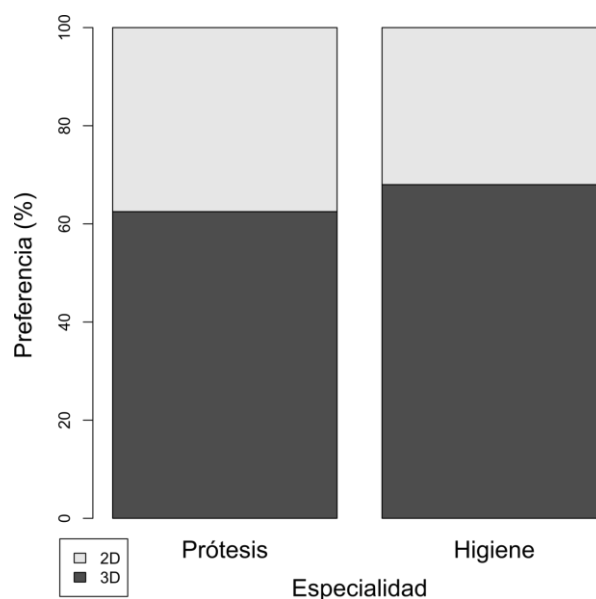


Ilustración 49 - Preferencia de sistema por especialidad

La segunda pregunta ha sido qué sistema ha sido el preferido por hombres y mujeres. En la Tabla 6 y en la Ilustración 50 vemos los resultados logrados.

	Chicos	Chicas
3D	69.23077	65.00000
2D	30.76923	35.00000

Tabla 6 - Preferencia de sistema por sexo

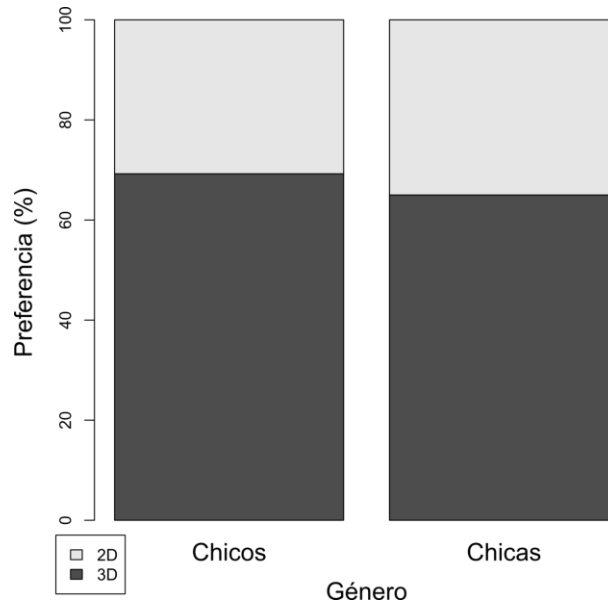


Ilustración 50 - Preferencia de sistema por sexo

5.6 Resultados de satisfacción

Para analizar los resultados de usabilidad y satisfacción del alumno nos basaremos en el test Q4. Este cuestionario consta de 12 preguntas, se utiliza una escala Likert de 5 puntos, donde 1 es nada y 5 mucho.

#	3D	2D	<i>U</i>	<i>Z</i>	<i>p</i>	<i>r</i>
Q3	5	4	193.5	2.281	0.023**	0.397
Q4	4	3	199	2.353	0.019**	0.410
Q5	5	5	137	0.064	1.000	0.011
Q6	5	5	121	-0.808	0.656	0.141
Q7	5	4.5	155.5	0.805	0.407	0.140
Q8	5	5	127	-0.401	0.724	0.070
Q9	5	4	160.5	0.965	0.347	0.168
Q10	5	5	151	0.624	0.541	0.109
Q11	4	4	166	1.146	0.255	0.199
Q12	5	3	226.5	3.400	< 0.001**	0.592
Q13	4	3	199.5	2.511	0.009**	0.437
Q14	5	5	136.5	0.022	1.000	0.004

Tabla 7 Medianas para las preguntas de satisfacción, el test Mann-Whitney U y *r*, tamaño del efecto

Hay diferencias estadísticas significativas en las preguntas Q3 (Mi experiencia con el uso de teléfonos móviles es), Q4 (Mi experiencia en jugar con juegos en teléfonos móviles es), Q12 (Hubo momentos en los que me parecía que el diente se salía de la pantalla) y Q13 (Hubo momentos en los que parecía que podía tocar el diente).

Las diferencias estadísticas significativas en las preguntas Q3 y Q4 demuestran que el grupo de estudiantes que han participado en la prueba es un grupo heterogéneo. Este resultado lo podemos combinar con el resultado del capítulo anterior en el cuál decíamos que ambos sistemas habían mejorado el aprendizaje del alumno. Combinando ambas ideas podemos deducir que nuestro sistema mejora el proceso de aprendizaje, tanto a usuarios expertos como a usuarios principiantes.

En la pregunta Q12 y Q13 podemos apreciar que los alumnos que utilizaron el sistema 3D antes de realizar el test, captaron perfectamente el efecto de profundidad. En cambio, los alumnos que utilizaron el sistema 2D no pudieron apreciar con tanta claridad este efecto. Con esto podemos decir que la autoestereoscopía en un dispositivo móvil es capaz de generar un efecto de profundidad bastante realista.

Respecto a la pregunta Q14 observamos que no existen diferencias significativas entre ambos sistemas. Aun así hemos querido resaltar esta pregunta ya que los alumnos de ambos sistemas han puntuado con la nota máxima su experiencia de uso. Creemos que es una pregunta muy importante, ya que sin la aceptación del alumnado y del profesorado, el objetivo de nuestro proyecto queda incompleto.

En base a los resultados obtenidos, podemos concluir que nuestro sistema ha mejorado el proceso de aprendizaje de los alumnos y que además ha tenido una gran aceptación entre todos los participantes.

Por último, decir que todos los estudiantes fueron muy participativos en las respuestas libres ya que propusieron varias ideas de cómo se podría aplicar esta aplicación a otros campos. Algunos de ellos plantearon posibles mejoras y en general tanto los alumnos como los profesores quedaron bastante satisfechos. A continuación se citan algunas de las respuestas más originales.

Q5 – Pregunta 6

¿Qué es lo que más te ha gustado de la experiencia?

- *“Que parecía que el molar se podía coger y porque se interpretaba muy bien”*

Q5 – Pregunta 7

¿Qué es lo que menos te ha gustado?

- *“Añadiría el aumento de los dientes para verlos de cerca y el giro más rápido”*

Q5 – Pregunta 8

¿Para qué se podría utilizar este sistema?

- *“Para enseñar a los alumnos una vista diferente del diente, ya que es más cómodo que el modelo en yeso.”*

Para terminar hemos querido incluir algunas imágenes que nos muestran las fases que realizaron los alumnos de nuestro estudio. En la Ilustración 51 vemos a una alumna realizando el test Estereoscópico. En la Ilustración 52 vemos a otra alumna utilizando el sistema 2D. Y por último, en la Ilustración 53 vemos a un alumno realizando el test Q3 y Q4.

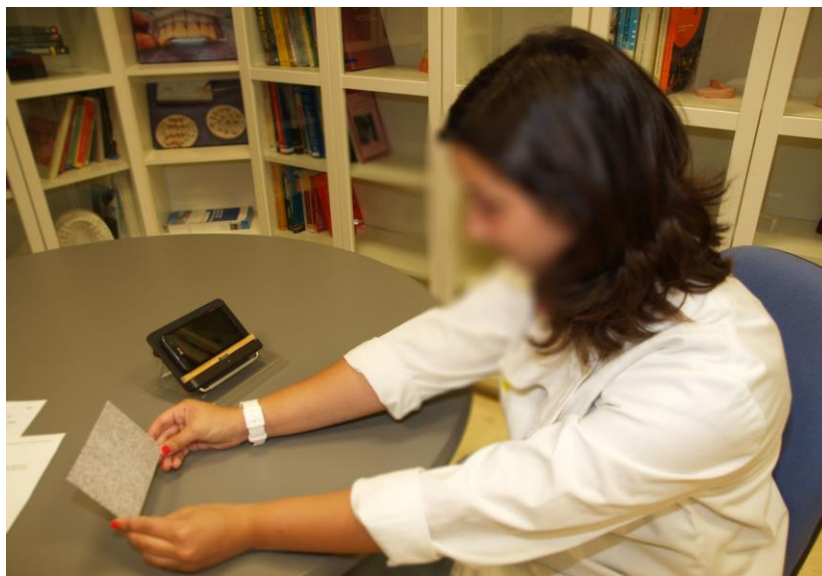


Ilustración 51 - Alumna realizando el Test Estereoscópico

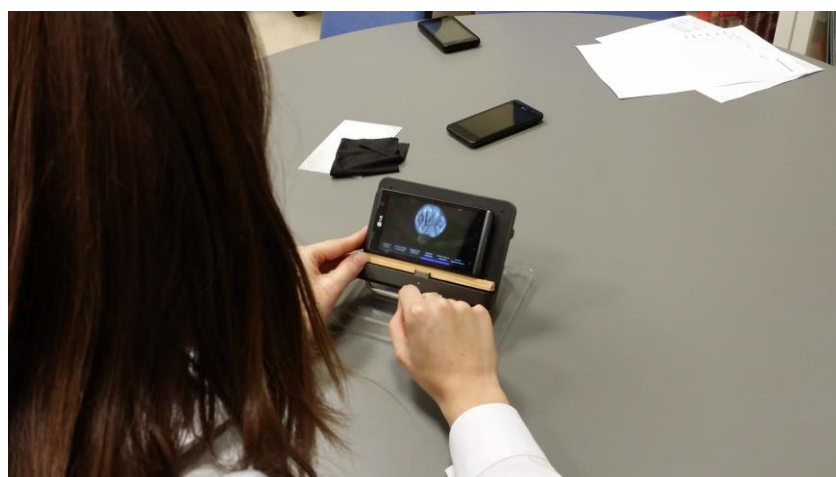


Ilustración 52 - Alumna utilizando el sistema 2D



Ilustración 53 - Alumno completando el test Q3 y Q4

6. Conclusiones y trabajo futuro

6

6.1 Conclusión.....	66
6.2 Trabajo futuro	67

6.1 Conclusión

En este trabajo final de máster hemos querido determinar el impacto que tienen las nuevas tecnologías en el campo de la educación. Para ello hemos desarrollado dos sistemas que intentan mejorar el proceso de aprendizaje de un alumno. El primero, un sistema dual, que puede representar modelos tanto en 2D como 3D. El segundo, un sistema que utiliza la Realidad Aumentada para mostrar objetos en 3D.

El funcionamiento de la autoestereoscopia acarrea ciertos problemas intrínsecos a su forma de representar los objetos, ya que el usuario debe mantener su ángulo de visión fijo respecto al objeto 3D. Debido a que en la RA debemos mover el móvil para poder captar todos los ángulos de la imagen, hace que ambos sistemas no se complementen bien. Ésta es la razón fundamental por la que elegimos nuestro primer sistema para realizar la validación con usuarios.

Al analizar los datos estadísticamente, determinamos el incremento del conocimiento obtenido por los usuarios que utilizaron el primer sistema. En él comparamos cuál de las dos aplicaciones, 2D o 3D, ha ofrecido un incremento mayor. El resultado ha sido que ambos sistemas han facilitado el incremento del conocimiento sobre la morfología dental. Además podemos destacar que ha habido una gran aceptación del sistema 3D por parte del alumnado. Un 66.7% de los estudiantes han preferido el sistema 3D al 2D.

Este sistema abre nuevas expectativas para el aprendizaje, ya que el alumno podrá estudiar, dónde, cómo y cuándo quiera. La única necesidad es tener un móvil, lo que hoy en día es bastante usual. Debemos recordar que no todos los móviles permiten visualizaciones 3D. Hoy en día existen pocos modelos que incorporen esta tecnología.

Por último, destacar que es un sistema fácil e intuitivo de manejar y que podría ser utilizado en otros ámbitos, como por ejemplo la medicina.

6.2 Trabajo futuro

Este trabajo, puede ser ampliado y mejorado de varias formas, a continuación mencionaremos algunas de ellas.

- Crear una nueva funcionalidad que nos permita realizar un zoom en las partes que nos interesen, para poder apreciar con mejor precisión los detalles del modelo.
- Aumentar la velocidad de rotación.
- Crear más modelos dentales para poder verlos utilizando el sistema.

Además se podría portar a otros ámbitos como por ejemplo la medicina. Una de las ramas de la medicina en la que se podría emplear es por ejemplo en la materia de anatomía del cuerpo humano. Se podrían crear modelos de los órganos principales o quizás de los huesos más importantes. El abanico de posibilidades es muy amplio.

Además podría emplearse desde colegios hasta universidades, ya que la generación de hoy en día está mucho más familiarizada con las nuevas tecnologías.

Para acabar esta tesina queremos mencionar el artículo que se está redactando sobre este trabajo, el cual será enviado a la revista *The Scientific World Journal* (revista indexada en JCR).

Bibliografía

[ANDAR14] Proyecto AndAR - *Android Augmented Reality* - <http://code.google.com/p/andar/> - (Accedido el 21 - 6 -2014)

[AUTOESTEREONUI14] Rodríguez D. - *Natural user interfaces and autostereoscopy for learning in dentistry* - tesina del máster de IARFID, 2014

[ARDENTAL13] Folguera, F., Juan, M. C., Herrero, A., Alexandrescu, L. - *Introducción a una nueva dimensión en la morfología dentaria: ARDental (Realidad Aumentada Dental)*. Gaceta dental: Industria y profesiones, (252), 200-209, 2013

[AZUMA97] Azuma - *A survey of augmented reality* - Presence: Teleoperators and Virtual Environments, Vol. 6, No 4, pp. 355–385, 1997

[BUCHANAN01] Buchanan, J. - *Use of Simulation Technology in Dental Education* - Journal of Dental Education; Vol. 65, No. 11, pp. 1225-30, 2001

[BUCHANAN04] Buchanan, J. *Experience with Virtual Reality-Based Technology in Teaching Restorative Dental Procedures* - Journal of Dental Education; Vol. 68, pp. 1258-65, 2004

[CHAN00] Chan, D. , Pruzler, K. , Caughman, W. - *Simulation with Preclinical Operative Dentistry Courses, 3 year Retrospective Study* - Journal of Dental Education (Abstract), Vol. 64, No. 3, pp. 190. 2000

[CHEN13] Chen, C. Y., Cheng, S. C., Wu, C. S., Yang, S. W., Tsai, C. H., Chen, C. Y. *Auto-stereoscopic Display with Multi-Projectors and Image Guiding Screen* - Proceedings of 3DSA2013, Vol 5, pp. 7, 2013

[DenX14] - DentSim - <http://www.dentsimlab.com/> (Accedido el 21 - 6 -2014)

[DEVINSENSE14] DevinSense (<http://www.devinsense.com>) - (Accedido el 21 - 6 - 2014)

[FLACK07] Flack, J., Harrold, J., & Woodgate, G. J. - *A prototype 3D mobile phone equipped with a next-generation autostereoscopic display*. In Electronic Imaging, Vol. 6490 , pp. 22, 2007

[JASINEVICIUS04] Jasinevicius T., Landers M., Nelson S., Urbankova A.- *An Evaluation of Two Dental Simulation Systems: Virtual Reality versus Contemporary Non-Computer-Assisted*, Journal of Dental Education, Vol. 68, No. 11, pp. 1151-1162, 2004

[KATO99] Kato, H., Billinghurst - *MMarker tracking and HMD calibration for a video-based augmented reality conferencing system* - Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR 99), pp: 85-94, 1999

[KAVO14] - DESplus - <http://www.kavo.com/Products/Dental-Training/Dental-simulation-units/DSEplus.aspx> (Accedido el 21 - 6 -2014)

[LEE07] Lee, T., Höllerer, T. Handy AR - *Markerless Inspection of Augmented Reality Objects Using Fingertip Tracking* - IEEE International Symposium on Wearable Computers (ISWC '07), pp. 83-90, 2007

[LUCIANO06] - Luciano C., Banerjee P., DeFanti T. - *Haptics-based virtual reality periodontal training simulator* - University of California at San Diego, San Diego, USA - Springer-Verlag, 2006 disponible para descargar en: <http://www.evl.uic.edu/files/pdf/PeriodontalSimulator.pdf> (Accedido el 21 - 6 -2014)

[MILGRAM94]Milgram, P., Takemura, H., Utsumi, A., Kishino, F. *Augmented Reality - A class of displays on the reality-virtuality continuum* - Proceedings of Telemanipulator and Telepresence Technologies, pp. 2351–34, 1994

[NYARTOOLKIT14] NyARToolkit NyARToolkit - http://nyatla.jp/nyartoolkit/wp/?page_id=198 - (Accedido el 21 - 6 -2014)

[OSG14] OSG Community - <http://www.openscenegraph.org> - Open Source high performance 3D graphics toolkit, 2005. (Accedido el 21 - 6 -2014)

[PATIL09] Patil J. , Saran N. , Shetty R. - *Haptic-based virtual reality dental simulator as an educational tool* disponible para descargar en: <http://medind.nic.in/eaat/t11/i2/eaat11i2p112.pdf> (Accedido el 21 - 6 -2014)

[PHANTOM14] Phantom Haptic Device - <http://www.sensegraphics.com/> - (Accedido el 21 - 6 -2014)

[POHLENZ10] Pohlenz P., Grobe A, Petersik A., Sternberg N., Pflesser B., Pommert A., K.-H. Hhne, Tiede U, Springer I, Heiland M. - *Virtual dental surgery as a new educational tool in dental school* - J. Cranio Maxill., Vol. 38, No. 8, pp. 560–564, 2010.

[PRAMIS13] Pramis J.- *Number of mobile phones to exceed world population by 2014* - <http://www.digitaltrends.com/mobile/mobile-phone-world-population-2014/#!6AZfh> (Accedido el 21 - 6 -2014)

[R14] Introduction to R - <http://www.r-project.org> - (Accedido el 21 - 6 -2014)

[REAL3D14] LG Real 3D SDK - <http://developer.lge.com/common/search/search.dev> - (Accedido el 21 - 6 -2014)

[RSTUDIO14] R Studio - <http://www.rstudio.com> - (Accedido el 21 - 6 -2014)

[VUFORIA14] SDK de Vuforia para Android, disponible para descarga en: <http://ar.qualcomm.at/qdevnet/sdk/android>, (Accedido el 21 - 6 -2014)

[WANG13] Wang, J., Suenaga, H., Yang, L., Liao, H., Kobayashi, E., Takato, T., & Sakuma, I. - *Real-Time Marker-Free Patient Registration and Image-Based Navigation Using Stereovision for Dental Surgery* - In *Augmented Reality Environments for Medical Imaging and Computer-Assisted Interventions*, pp. 9-18, 2013

[WANG14] Wang J. , Suenaga H., Hoshi K., Yang L., Kobayashi E., Sakuma I., Liao H. - *Augmented Reality Navigation With Automatic Marker-Free Image Registration Using 3-D Image Overlay for Dental Surgery* - *IEEE Transactions On Biomedical Engineering*, Vol. 61, No. 4, pp. 1-1, 2014

[WIKI14] - *List of countries by number of mobile phones in use* - http://en.wikipedia.org/wiki/List_of_countries_by_number_of_mobile_phones_in_use< (Accedido el 21 - 6 -2014)

[YAU06] Yau H. , Tsou L. , Tsai M. - *Octree based virtual dental training system with a haptic device* - *Computer/Aided Design & Applications*, vol. 3, pp. 415–424, 2006.

Questionarios

Q.

Q.1 Pre- Test.....	73
Q.2 Test Estereoscopia.....	74
Q.3 Post-Test.....	75
Q.4 Test Usabilidad.....	76
Q.5 Test comparativo entre sistemas.....	77

Q1

Código:

Nombre:

Edad:

Chico

Chica

Prótesis

Higiene

Grupo: A1

A2

B1

B2

1. En el diente siguiente, indica mediante distintos colores:

1. Fosas y surcos oclusales (boli negro)
(boli rojo)

2. Cúspides céntricas (boli azul)

3. Cúspides no céntricas

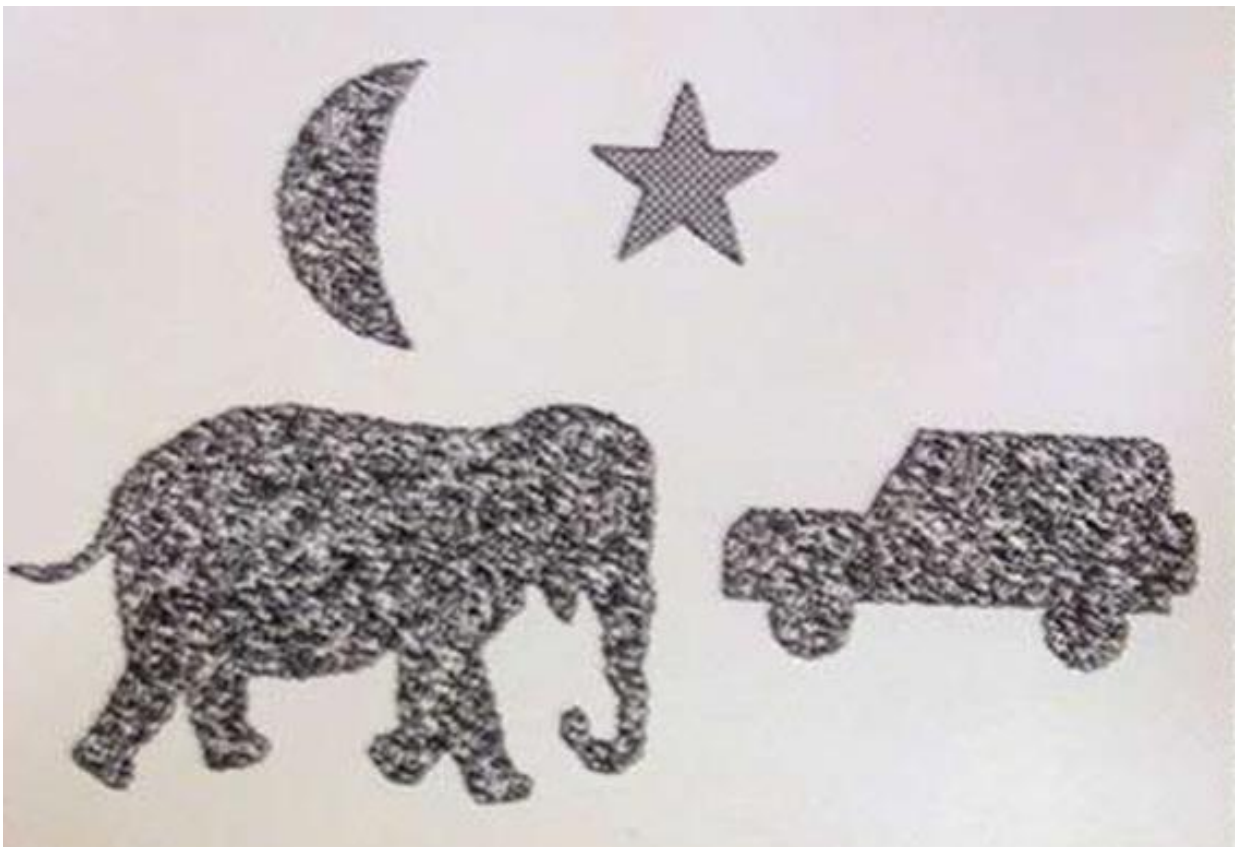
4. Contorno oclusal (boli amarillo)
(boli verde)

5. Aristas de cúspides triangulares (boli rojo y azul)

6. Surcos suplementarios



Q2



Q3

Código:

Nombre:

Edad:

Chico

Chica

Prótesis

Higiene

Grupo: A1

A2

B1

B2

2. En el diente siguiente, indica mediante distintos colores:

2. Fosas y surcos oclusales (boli negro)
(boli rojo)

2. Cúspides céntricas (boli azul)

3. Cúspides no céntricas

4. Contorno oclusal (boli amarillo)
(boli verde)

5. Aristas de cúspides triangulares (boli rojo y azul)

6. Surcos suplementarios



Q4

Código: Sexo:

Con autoestereoscopia

Edad:

Sin autoestereoscopia

Te vamos hacer unas preguntas para conocer que te ha parecido el sistema que has utilizado.

Señala con una cruz tu respuesta:

	Nada	Poco	Regular	Bastante	Mucho
1. ¿Cuánta experiencia tienes con el uso de teléfonos móviles/Tablets?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2. ¿Cuánta experiencia tienes en jugar con juegos en teléfonos móviles/Tablets?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3. El sistema me ha parecido fácil de usar	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4. Ver el diente desde distintas posiciones tocando la pantalla me ha parecido fácil	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5. ¿Te ha gustado la pieza dental que has visto?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6. ¿Crees que este sistema es útil para aprender la morfología dental?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
7. ¿Crees que has aprendido con este sistema?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
8. ¿Recomendarías este sistema a tus compañeros de clase?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
9. Hubo momentos en los que creíste que el diente era un modelo de escayola real	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10. Hubo momentos en los me parecía que el diente se salía de la pantalla	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
11. Hubo momentos en los me parecía que podía tocar el diente	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
12. Puntúa la experiencia de 1 a 5	<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5

Q5

Código:

Ahora me gustaría que compararas los dos sistemas que has utilizado

1. ¿Qué sistema te parece mejor para el aprendizaje dental?

Con autoestereoscopia Sin autoestereoscopia

¿Por qué?

2. ¿Cuál te ha resultado más fácil de manejar?

Con autoestereoscopia Sin autoestereoscopia

3. ¿En cuál se veía mejor la pieza dental?

Con autoestereoscopia Sin autoestereoscopia

4. ¿Cuál recomendarías a tus compañeros de clase?

Con autoestereoscopia Sin autoestereoscopia Ninguno de los dos

5. ¿Te gustaría que alguno lo utilizara tu profesor en clase?

Con autoestereoscopia Sin autoestereoscopia Ninguno de los dos

6. ¿Qué es lo que más te ha gustado de toda la experiencia? ¿Por qué?

7. ¿Qué es lo que menos te ha gustado de toda la experiencia? ¿Por qué?

8. ¿Para qué crees que se podrían utilizar este sistema?

9. Añade los comentarios que quieras