



Universidad de las Ciencias Informáticas

Facultad 5

*Componente para el registro de elementos virtuales mediante
reconocimiento de características naturales.*

**Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas.**

Autores: Yanet Valiente Padrón.

Donic Pérez Guadarrama.

Tutores: Ing. Mileydi Moreno Mirabal.

Ing. Ernesto de la Cruz Guevara Ramírez.

Ciudad de La Habana - Julio 2010

“Año del 50 Aniversario del Triunfo de la Revolución.”

Todo lo que una persona puede imaginar, otras podrán hacerlo realidad.

Julio Verne

Declaración de Autoría

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de Julio del año _____.

Yanet Valiente Padrón

Firma del Autor

Donic Pérez Guardarrama

Firma del Autor

Mileydi Moreno Mirabal

Firma del Tutor

Ernesto de la Cruz Guevara Ramírez

Firma del Tutor

Datos de Contacto

Nombre y apellidos: Mileydi Moreno Mirabal

Institución: Universidad de las Ciencias Informáticas (UCI).

Título: Ingeniero en Ciencias Informáticas.

e-mail: mmirabal@uci.cu

Nombre y apellidos: Ernesto de la Cruz Guevara Ramírez

Institución: Universidad de las Ciencias Informáticas (UCI).

Título: Ingeniero en Ciencias Informáticas.

e-mail: elguevara@uci.cu

Agradecimientos

A la Revolución y a nuestro Comandante por darnos la oportunidad de superarnos y poner en nuestras manos un proyecto tan importante como la Universidad de las Ciencias informática.

A mis abuelitos por ser mis principales guías y en los momentos más difíciles apoyarme y ayudarme a seguir adelante, principalmente a mi abuelo que sin él hubiera sido imposible alcanzar mi meta.

A mi mamita linda por estar siempre a mi lado y ayudarme y apoyarme en todas mis decisiones, a mi papi del alma por ser el mejor papa del mundo y llenarme de amor y cariño cada momento de mi vida y enseñarme que hasta las cosas más difíciles con sacrificio y esfuerzo se logran.

A mi hermano que aunque en estos no se encuentra porque esta de misión me ayudo mucho en todos estos años de universidad.

A mi tía Rosita por ser mi ejemplo a seguir y estar siempre ahí para aconsejarme y guiarme en mis decisiones.

A Wilfredo mi ángel de la guarda por apoyarme mucho en estos años y estar a mi lado cuando más lo he necesitado.

A mi hermanita Yanay por ser la persona más especial que he conocido en mi vida en estos 5 años y estar a mi lado en las buenas y las malas y apoyarme y ayudarme cuando me he quedado casi sin fuerzas.

A Maydelin mi otra hermanita a la Kika a Oadis, por ser mis amigas en todo momento y compartir juntas en estos años momentos maravillosos.

A los chicos del aula a Ale, a Hugo, a Gustavo que me ha ayudado mucho en estos 5 años y hasta en el último momento.

A mi compañero de tesis y amigo Donic por aguantarme en todo este tiempo de estres y estar igual junto a mí desde los primeros años de universidad.

A mis tutores por brindarnos todos sus conocimientos en aras de lograr el mayor éxito.

En fin a todas las personas que día a día contribuyen a que yo sea mucho más feliz.

Yanet

Agradecimientos.

Llegar a la universidad fue un sueño hecho realidad, pero convivir y compartir con tantas buenas personas, a lo largo de estos cinco años en la UCI, fue lo mejor que me paso en la vida. Cada uno de mis logros y triunfos tienen impreso un sello de amor, comprensión y ayuda de mi familia, mis amigos, mis compañeros de aula y mis profesores.

A todos muchas gracias.

A Mami por ser lo más grande que tengo, por su amor y apoyo incondicional, por ser mi inspiración.

A Mamá por ser la diosa de mi corazón, por darme tanto amor y fuerza.

A mi novia por su amor y comprensión en todos estos años, sin ti nada tendría sentido, te amo Titi.

A la Nena y a Tata por apoyarme y complacerme en todo, sin Uds. no lo hubiera logrado.

A mi tío Adalberto y mi tía Ilda por siempre estar pendientes de mí.

A mi primo Alexei por su ayuda desde el primer día que entre en la universidad.

A mis suegros Belkys y Eduardo por abrirme las puertas de su corazón y hacerme sentir como su hijo.

A toda mi familia materna, que tanto me ha apoyado y me ha enseñando el verdadero valor de la familia.

Al tribunal de tesis, en especial a Dayren y Lidiexys por sus sugerencias para mejorar este trabajo.

A mis amigos que siempre estuvieron en las buenas y en las malas, en especial a:

Hector, Heriberto, Jose, Lisandro, Reinier, Rene, Yuniet y Yolier.

A Gustavo por ayudarnos cuando más falta nos hizo.

A mis amigas de la universidad, en especial a Dainerys, Lianet, Leixys, Maydelin, Misleidy y Yanay.

A mi CT Yanet por estar juntos desde el primer día de la universidad y aguantarme tanto en esta tesis.

A mis compañeros de aula y de cuarto, especialmente los del grupo 2 y 4.

Al piquete del WOW por compartir tan buenos momentos, For he Horde.

A Eliezer por ayudarme con la impresión y a Daniel porque sin conocerme me ayudo con los modelos 3D.

A mis profesores de la UCI, especialmente a Zaida y Villar por aconsejarme con tanto cariño.

A la Revolución por recompensar mi esfuerzo y brindarme la oportunidad de realizar esta linda carrera.

A mis tutores por su atención.

Donic

Dedicatoria

Este Trabajo de Diploma está dedicado a mis abuelitos que son la luz de mi vida y a mis padres por ser los mejores de este mundo y a mis hermanitos Carlin por ser mis tesoro y Dany mi ejemplo a seguir, para ellos con todo mi amor y cariño fue el sacrificio de estos 5 años de universidad.

Yanet

Todo el esfuerzo y las horas de empeño para lograr realizar esta tesis quiero dedicarlas a las personas que han sido y son la razón de mi ser, y que adoro desde lo más profundo de mi corazón.

A mi mamita del alma Dalis, este triunfo es todo tuyo.

A Mamá, porque me pediste este título y lo cumplí.

A mis hermanos Donel y Doniel, para que sigan mi ejemplo y logren lo que se propongan.

A mi novia Delia, porque es el amor de mi vida.

A mis primos María del Carmen e Idalberto, por ayudarme en toda mi vida.

A mis tíos Ilda y Adalberto por inspirarme tanto respeto.

A la memoria de Blanca Vázquez, por quererme tanto.

A mi papa Abilio para que se sienta orgulloso de mi.

A todos mis amigos.

Donic

Resumen

La Realidad Aumentada es una tecnología que está actualmente en constante desarrollo y evolución, su aplicación exitosa en diferentes áreas del conocimiento es cada día mayor, sobre todo en los países más desarrollados. Debido a la aceptación de las aplicaciones que utilizan esta tecnología, sobre todo en el campo del entretenimiento y la educación, la línea de investigación Interacción 3D del Centro de Desarrollo de Informática Industrial en la Universidad de las Ciencias Informáticas (UCI) está llevando a cabo un estudio sobre esta tecnología ya que podría formar parte de los procesos formativos de los estudiantes de la UCI. Esta investigación responde a la necesidad de registrar elementos virtuales, o sea, insertar información adicional (objetos en tres dimensiones) a una escena real mediante el reconocimiento de características naturales de imágenes planas ubicadas en dicha escena. En la presente tesis se hace un estudio de los conceptos y características de la Realidad Aumentada, se exponen los dispositivos de entrada y salida más utilizados y algunas técnicas de registros existentes, haciendo énfasis en las técnicas basadas en visión por computador. También se propone una solución para registrar los objetos virtuales, en una escena aumentada con respecto a la cámara, que involucra la utilización de BazAR como biblioteca de desarrollo de software que permite el reconocimiento de características naturales de una imagen plana. Finalmente, se describe la solución propuesta y se presenta un componente para demostrar su validez.

Palabras clave: características naturales, elementos virtuales, reconocimiento, registro.

Índice

Declaración de Autoría	I
Datos de Contacto	II
Agradecimientos	III
Dedicatoria	V
Resumen	VI
Índice	VII
Índice de Figuras	XI
Índice de Tablas	XIII
Introducción	1
Capítulo 1: Fundamentación Teórica.	5
Introducción del capítulo.	5
1.1 Realidad Aumentada	5
1.1.1 Continuo de Realidad Mezclada de Milgram.	6
1.1.2 Clasificación de los Sistemas de Realidad Aumentada (SRA)	6
1.2 Desarrollo y Aplicaciones de la Realidad Aumentada	8
1.2.1 Aplicaciones de Realidad Aumentada	8
1.3 Definición de registro (<i>tracking</i>)	12
1.3.1 Relaciones espaciales entre los entornos virtual y real	12
1.4 Técnicas de registro en los Sistemas de Realidad Aumentada	14
1.4.1 Técnicas de registro basadas en sensores (<i>sensor-based</i>)	14
1.4.2 Técnicas de registro basadas en visión (<i>vision-based</i>)	15
1.4.3 Técnicas de registro híbrido.	16
1.5 Estrategias del registro basado en visión	17

1.5.1	Sistemas de patrones de marcas planas (<i>planar pattern marker systems</i>).....	17
1.5.2	Sistemas de registro de características naturales (<i>natural feature tracking</i>).	18
1.6	Entradas de los Sistemas de Realidad Aumentada. Sensores.	19
1.6.1	Sensores ópticos.	20
1.7	Salidas de los Sistemas de Realidad Aumentada. Displays.....	20
1.7.1	Display Visuales.....	21
1.7.2	Display Espaciales.....	22
1.7.3	Display Espaciales de Video.	22
1.8	Bibliotecas para desarrollar aplicaciones de Realidad Aumentada.	23
1.9.1	Software basado en patrones de marcas planas (ARToolKit).....	24
1.9.2	Software de reconocimiento de rasgos naturales (BazAR).....	26
1.9.3	Principios básicos de BazAR.	26
	Conclusiones del capítulo.	27
	Capítulo 2: Descripción de la solución propuesta.	28
	Introducción del capítulo.	28
2.1	BazAR como biblioteca para desarrollar la solución.....	28
2.1.1	Ventajas y Desventajas de BazAR.....	28
2.1.2	Proceso de registro de BazAR.....	29
2.2	Propuesta de solución.	29
2.3	Soluciones técnicas.....	30
2.3.1	Metodología de desarrollo de software RUP.	31
2.3.2	Lenguaje de programación C++.....	31
2.3.3	OpenGL como biblioteca gráfica.	32

2.3.4	UML como lenguaje de modelado	32
2.3.5	Visual Paradigm para UML como herramienta CASE.	32
2.3.6	Microsoft Visual Studio 2008.	33
2.4	Entradas requeridas para el funcionamiento del componente.	33
2.4.1	Obtención de las entradas	34
2.5	Modelo de Domino.....	36
2.5.1	Descripción del modelo de dominio.....	37
2.6	Requerimientos del sistema.	37
2.6.1	Requisitos Funcionales (RF).....	38
2.6.2	Requisitos no funcionales (RNF).....	38
2.7	Modelación del sistema.	39
2.7.1	Actor del Sistema.....	39
2.7.2	Modelo de Casos de Uso del Sistema.....	40
2.7.3	Descripción de los Casos de Uso del Sistema.....	40
Conclusiones del capítulo.....		43
Capítulo 3: Diseño, Implementación y Resultados.		44
Introducción del capítulo.....		44
3.1	Diagrama de clases.....	44
3.2	Descripción de las clases.....	45
3.3	Diagrama de componentes.....	48
3.4	Diagramas de secuencia.....	48
3.5	Patrón de diseño utilizado.....	50
3.6	Análisis de los resultados obtenidos.....	50

Conclusiones del capítulo..... 52

Conclusiones Generales..... 53

Recomendaciones..... 54

Glosario de Términos..... 55

Bibliografía Citada..... 57

Bibliografía Consultada..... 58

Anexos..... 65

Índice de Figuras

Figura 1: Pirámide de las etapas de desarrollo de la tecnología de RA.	1
Figura 2: Continuo de Milgram.	6
Figura 3: Imágenes de Human Pacman.....	9
Figura 4: Usuarios interactuando con Construct3D.....	10
Figura 5: Sistema de RA adaptado a un escáner médico..	10
Figura 6: Tratamiento de fobias.....	11
Figura 7: Sistema de RA en procesos de manufactura.....	11
Figura 8: Sistemas de coordenadas real/virtual en aplicaciones basadas en registro por sensores para un sistema de referencia terrestre y un registro directo.....	13
Figura 9: Sistemas de coordenadas real/virtual en aplicaciones basadas en registro por cámaras para un sistema de referencia terrestre y un registro basado en visión.	14
Figura 10: Displays visuales.	21
Figura 11: ARToolkit facilita el desarrollo de Realidad Aumentada utilizando marcadores.	25
Figura 12: Bazar facilita el desarrollo de Realidad Aumentada basadas en la detección y correspondencia de rasgos naturales puntuales.	26
Figura 13: Adquisición del modelo.	34
Figura 14: Adquisición de los datos.....	35
Figura 15: Cómputo de la calibración geométrica..	36
Figura 16: Modelo de dominio.	37
Figura 17: Diagrama de Caso de Uso del Sistema.....	40
Figura 18: Diagrama de clases.....	44
Figura 19: Diagrama de Componentes.	48

Figura 20: Diagrama de secuencia del caso de uso: Inicializar Componente.	49
Figura 21: Diagrama de secuencia del caso de uso: Realizar Registro 3D.	49
Figura 22: Diagrama del Patrón Fachada.	50
Figura 23: Aplicación demostrativa.	51
Figura 24: Parcial oclusión de la imagen modelo.	66
Figura 25 : Máxima distancia con respecto a la cámara.	67
Figura 26 : Grundy Campeón	67

Índice de Tablas

Tabla 1: Software para desarrollar aplicaciones de RA.....	23
Tabla 2: Actor del Sistema.....	40
Tabla 3: Descripción del Caso de Uso Inicializar Componente.....	42
Tabla 4: Descripción del Caso de Uso Realizar Registro 3D.....	43
Tabla 5: Descripción de la clase IRA.....	46
Tabla 6: Descripción de la clase CamManager.....	46
Tabla 7: Descripción de la clase Cam.....	47
Tabla 8: Descripción de la clase Cam.....	¡Error! Marcador no definido.
Tabla 9: Descripción de la clase CalibModel.....	48

Introducción

A medida que las tecnologías progresan y nuevos dispositivos surgen, las posibilidades de desarrollo para el mundo son cada vez más amplias, y resulta indispensable estar cada vez más informados sobre los últimos avances para lograr un real valor diferencial frente a nuestros pares. Una tecnología que viene a revolucionar todo lo conocido sobre la Realidad Virtual (RV) y abriendo paso a un nuevo paradigma de la Informática es la denominada Realidad Aumentada (RA).

La Realidad Aumentada es un paradigma que surge a partir de la Realidad Virtual. Para ambas tecnologías existen una serie de fases, representadas en la pirámide elaborada a partir de [Bimber, 2005], que se interpreta de abajo a arriba (Figura 1). En el primer nivel de esta pirámide se encuentran los elementos básicos de la tecnología, como los **sistemas de registro** (*tracking*), **dispositivos de visualización** y **renderización**; en un segundo nivel están los elementos más complejos, como son **la interacción, presentación y autoría**; el tercer nivel lo forman las **aplicaciones** que constituyen la interfaz con el usuario; y en el último nivel están los **usuarios** o consumidores finales de dicha tecnología. Según [Bimber, 2005], “mientras que la comunidad científica se encuentra en el segundo nivel de desarrollo para el caso de la Realidad Virtual, la Realidad Aumentada se encuentra todavía en el primer nivel”, por lo que no es de extrañar que las aplicaciones actualmente desarrolladas no sean lo suficientemente robustas, que las interfaces no tengan un aspecto depurado, o que no se haya profundizado lo suficiente en temas referentes al usuario final.



Figura 1: Pirámide de las etapas de desarrollo de la tecnología de RA. [Bimber, 2005].

La parte más importante de este esquema es el usuario final, puesto que dependiendo de la utilidad real o beneficios que la tecnología de RA pueda aportar a la sociedad o a un grupo de usuarios se seguirá invirtiendo en ella y se mejorarán sus aspectos técnicos, para convertirla en una tecnología robusta, eficaz y asequible. La actual fragilidad de la RA impide que se elaboren aplicaciones lo suficientemente estables, complejas y/o atractivas, por lo que, según [Livingston, 2005], “muchas de ellas quedan como simples ensayos de laboratorio, sin posibilidad de comercializarse, si no se demuestran los beneficios que esta tecnología puede aportar con respecto a otras, difícilmente se fomentará su desarrollo.” Estas reflexiones datan del año 2005; en la actualidad se aprecia un interés creciente por el estudio de aspectos relacionados con el usuario final. Hoy en día los campos donde se aplica esta tecnología son inmensos y se augura que con la socialización de la RA se puede anticipar una verdadera revolución en nuestra forma de hacer las tareas diarias, con una mayor facilidad y una interacción más entretenida y cómoda para todos.

En Cuba cada día se están logrando mayores avances con respecto al tema de la RV, ya existen aplicaciones de esta tecnología en el marco social, en la educación, la cultura, el entretenimiento y la medicina. Se han desarrollado programas de RV para juegos, simuladores de conducción, simuladores de tiro; sin embargo las aplicaciones realizadas utilizando técnicas de RA aún son muy escasas ya que en el mundo los Sistemas de Realidad Aumentada (SRA) existentes son generalmente privados, y se requieren del uso de tecnologías avanzadas que no están al alcance del país y los dispositivos que se utilizan reportarían un costo muy alto.

Una de las instituciones encargadas de desarrollar aplicaciones de RV es la Universidad las Ciencias Informáticas (UCI) específicamente el Centro de Desarrollo de Informática Industrial (CEDIN), pero debido al auge que está teniendo a nivel mundial el desarrollo de programas aplicando exitosamente estas tecnologías de RA y la gran gama de beneficios que nos brinda en las diferentes áreas del conocimiento humano, la Facultad 5 de la UCI y específicamente la línea de investigación Interacción 3D que pertenece al CEDIN investiga sobre la tecnología de Realidad Aumentada. Se han logrado algunos avances en este campo, existen varios trabajos de diplomas que han aportado información y algunas aplicaciones sobre esta tecnología, un ejemplo de ello se puede encontrar en [Guevara, 2008], [Romero, 2008].

La línea de investigación Interacción 3D necesita una herramienta que registre elementos virtuales a partir del reconocimiento de las características naturales presentes en imágenes de una escena real. Se hace necesario lograr una mayor percepción de lo que representan estos objetos virtuales en función de lo que se está reconociendo en la realidad y así lograr un mayor realismo en las escenas aumentadas. Dicha línea de investigación cuenta actualmente con una herramienta para el registro de elementos virtuales mediante el reconocimiento de marcadores cuadrados utilizando la biblioteca ARToolKit pero esta biblioteca no brinda una vista clara de lo que se quiere aumentar en la escena y en ocasiones cuando hay una escena que involucra muchos marcadores de este tipo, entonces se está en presencia de una escena codificada, cuando realmente en algunos contextos ese no debería ser el sentido, además el registro 3D se pierde por completo cuando los marcadores son parcialmente ocluidos obligando a que estos siempre estén visibles, lo cual puede ser contraproducente cuando se utilizan marcadores de dimensiones mayores que la resolución de la cámara que capta la escena y la distancia que separa a la cámara del marcador es muy pequeña.

Lo expresado anteriormente conlleva a plantearse la siguiente interrogante ¿Cómo registrar características naturales, presentes en las imágenes de escenas reales, para insertar información aumentada? Este constituye el **Problema Científico**.

Por tanto el **Objeto de Estudio** es el Registro 3D para Realidad Aumentada y el **Campo de Acción** Técnicas de reconocimiento de características naturales para Sistemas de Realidad Aumentada. El **Objetivo de Investigación** de esta tesis es: Elaborar un componente reutilizable que permita registrar elementos virtuales tridimensionales a partir de las características naturales presentes en una imagen de una escena real, para utilizarlo en la construcción de aplicaciones de Realidad Aumentada. Para darle cumplimiento al objetivo en pos de resolver el problema científico, se plantearon las siguientes:

Tareas de Investigación:

1. Análisis del estado actual de las técnicas de registro en la Realidad Aumentada, para determinar las técnicas más recomendadas.

2. Análisis de las bibliotecas de código fuente existentes para el trabajo con Realidad Aumentada, para seleccionar la más adecuada para la investigación.
3. Selección del lenguaje, las metodologías y las herramientas de desarrollo para desarrollar la solución.
4. Implementación de un componente que sea una interfaz de funcionalidades para lograr el registro de elementos virtuales mediante reconocimiento de características naturales.
5. Implementación de una aplicación demostrativa donde se aplique el resultado de la investigación.

Entre los **métodos científicos de investigación** que apoyaron el desarrollo de esta investigación se encuentran:

Métodos Teóricos

- **Histórico-lógico:** Este método permitió ampliar los conocimientos acerca del estado actual en que se encuentra el fenómeno en cuestión, realizando un análisis de la bibliografía concerniente al tema, dándole así cumplimiento a la primera parte de la investigación sobre el análisis bibliográfico del tema.
- **Analítico-sintético:** Mediante este método se pudo analizar y estudiar el objeto de investigación, determinando los componentes significativos que forman parte de él. También brindó el cómo relacionar sus componentes de manera tal que se pueda ver el funcionamiento del objeto de investigación como un todo al integrar sus partes.
- **Modelación:** Este método permitió hacer un modelo de la propuesta de solución, creando abstracciones con el objetivo de lograr su mejor entendimiento.

Métodos Empíricos

- **Revisión de documentos:** Este método nos permitió determinar el estado del arte del objeto de investigación.

El presente trabajo de diploma está estructurado de la siguiente forma: Resumen, Introducción, tres capítulos de contenido, Conclusiones, Recomendaciones, Bibliografía Citada, Bibliografía Consultada, Glosario de Términos y los Anexos. A continuación se hace una breve descripción del contenido de cada uno de los capítulos:

- Capítulo 1 “Fundamentación Teórica”: Se hace un análisis de los principales conceptos que se encuentran involucrados con la Realidad Aumentada, se muestra el estado del arte relacionado con el tema de la investigación, las técnicas de registro en los Sistemas de Realidad Aumentada, los dispositivos de entrada y salida que más se utilizan y las bibliotecas utilizadas para el desarrollo de este tipo de aplicaciones.
- Capítulo 2 “Descripción de la Solución Propuesta”: En este capítulo se muestran las características de la solución elaborada. Se hace una conceptualización a través del modelo de dominio, un levantamiento de requisitos, se obtienen los casos de uso del sistema y una descripción de estos obteniendo así las entradas para el diseño e implementación del componente.
- Capítulo 3 “Diseño, Implementación y Resultados”: Muestra los diagramas de clases de diseño, los diagramas de secuencia, las descripciones de las clases de diseño, así como el diagrama de componentes y un análisis de los resultados observados.

Capítulo 1: Fundamentación Teórica.

Introducción del capítulo.

En éste capítulo se aborda algunos conceptos relacionados con la Realidad Aumentada, principalmente el de registro tridimensional, del cual se analizan algunas técnicas y estrategias, haciendo énfasis en la basada en visión y reconocimiento de características naturales. También se exponen algunos ejemplos de aplicaciones, dispositivos de entrada/salida y bibliotecas de desarrollo para esta tecnología.

1.1 Realidad Aumentada.

La Realidad Aumentada (RA) es una tecnología novedosa y aunque su aparición es bastante reciente, hoy en día su utilización es cada vez más extendida por los grandes beneficios que está brindando para el mundo. La RA incorpora información adicional al mundo real explotando las habilidades visuales y espaciales de los usuarios de forma que su foco de atención no esté en el ordenador, sino en el mundo real.

A partir del análisis de la bibliografía consultada se identificaron conceptos y definiciones dadas por muchos autores en sus artículos acerca de la RA [Janin,1993], [Milgram,1994], [Barfield,1995], [Billinghurst, 2002] y [Bimber, 2005], luego de hacer un análisis de estos se puede notar que la mayoría de los autores coinciden que la RA es el aumento de una escena real con objetos virtuales 3D de tal forma que éstos ofrezcan información complementaria a la realidad.

En lo adelante los autores de la investigación adoptarán por concepto de RA el enunciado por [Azuma 2007] por ser el más general y ajustado a este trabajo donde se define la Realidad Aumentada como los sistemas que contienen simultáneamente las tres características siguientes:

1. Mezcla de lo real y lo virtual.
2. Interactividad en el tiempo real.
3. Registro tridimensional.

1.1.1 Continuo de Realidad Mezclada de Milgram.

El concepto de Realidad Mezclada (RM) fue definido por primera vez en [Milgram, 1994] a partir del continuo Realidad-Virtualidad, comúnmente conocido como **Continuo de Milgram** (Figura 2) que aparece referenciado por múltiples autores, como en [Milgram, 1997], [Billinghurst, 2001], [Florins, 2005], [Jacobs, 2006], [Lam, 2006] Según este continuo, dependiendo de la cantidad de entorno generado por ordenador, se puede establecer una clasificación que va desde el entorno completamente real al completamente virtual, pasando por los estados intermedios de Realidad Aumentada (RA) y Virtualidad Aumentada (VA). La tecnología de RM engloba a la RA y la VA. La diferencia entre estas dos últimas tecnologías radica en la cantidad de entorno sintético (o contenido generado por ordenador) frente a la cantidad de entorno real, aunque en muchos casos la línea que separa ambas tecnologías es difícil de trazar y algunas aplicaciones no se pueden catalogar claramente en una de estas categorías. Cabe señalar que Milgram diferencia entre: Entorno Real – Realidad Aumentada – Virtualidad Aumentada – Entorno Virtual.

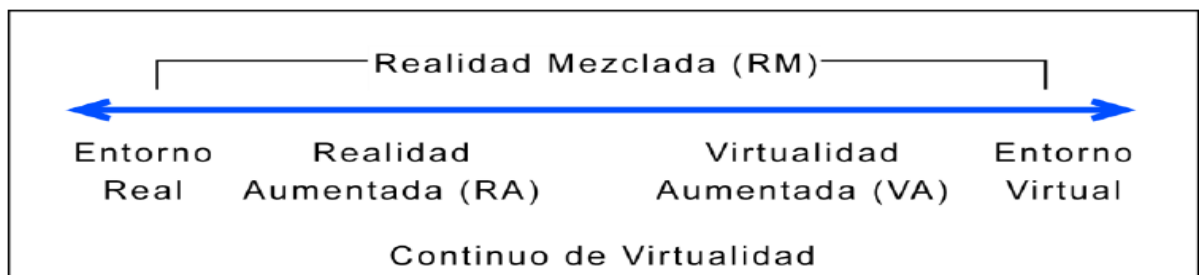


Figura 2: Continuo de Milgram. A partir de [Milgram, 1994].

1.1.2 Clasificación de los Sistemas de Realidad Aumentada (SRA).

Actualmente, no existe una única o clara clasificación de los Sistemas de Realidad Aumentada, sin embargo, existen algunos términos que establecen cierta categoría, atendiendo a distintos criterios; por cada una de las distinciones que mencionamos, referenciamos una aplicación que la ejemplifica.

- Según el **entorno físico** en el que se desarrolla la aplicación, podemos diferenciar entre **sistemas dentro de recintos o cerrados (*indoors*)** [Broll, 2004] o **sistemas al aire libre o abiertos**

(outdoors) [Avery, 2005]. La diferencia entre aplicaciones cerradas y las aplicaciones abiertas condiciona muchos aspectos de los SRA establecidos, principalmente el tipo de dispositivos de registro y displays utilizados. De hecho, en aplicaciones al aire libre, se suelen emplear receptores GPS y sensores inerciales para el registro, y PDAs o HMDs para los displays; mientras que en los recintos, los dispositivos de registro y displays son mucho más variados.

- Según la **extensión que abarquen**, podemos distinguir entre **sistemas locales** [Park, 2006] u **ubicuos** [Sandor, 2005]. Los sistemas locales se desarrollan en un ámbito acotado, bien en espacios abiertos o dentro de recintos. Las aplicaciones ubicuas (u omnipresentes) son aquellas en las que la extensión del entorno aumentado es tan amplio, que el usuario tiene la sensación de que vaya donde vaya seguirá estando inmerso en la aplicación. En este tipo de RA se suelen utilizar diversas tecnologías de registro que se complementen, para permitir que el usuario pueda entrar en recintos o salir a espacios abiertos. Además, los sistemas suelen ser móviles, ya que el usuario debe desplazarse libremente.
- Según la **movilidad de los dispositivos de registro y/o displays** [Lee, 2006] podemos distinguir entre **sistemas móviles** o **sistemas espaciales**. En los sistemas móviles generalmente el usuario lleva consigo los dispositivos de registro, el display e incluso, el ordenador o PDA que gestiona la aplicación. Estas aplicaciones son usuales, aunque no restrictivas, de espacios abiertos. Contrariamente, en aplicaciones espaciales, los dispositivos de registro y/o displays están fijos en el entorno, utilizándose en la mayoría casos sistemas de proyección sobre superficies tridimensionales. En [Bimber, 2005] se hace un estudio exhaustivo sobre la RA espacial.
- Según el **número de usuarios** que simultáneamente pueden interactuar con el sistema, podemos distinguir entre **sistemas individuales** [Schnädelbach, 2002] y los **colaborativos** [Cooper, 2004]. Así pues, el sistema es individual cuando existe la posibilidad de interacción de un solo usuario, y colaborativa si existe, como mínimo, la posibilidad de participación de dos usuarios de forma simultánea.

- Según el tipo de colaboración establecida (para sistemas colaborativos), podemos distinguir entre sistemas presenciales o cara a cara [Kaufmann, 2003] o remotos [Flintham, 2003]. En las aplicaciones colaborativas, se distingue si los usuarios colaboran cara a cara, estando físicamente presentes en el mismo entorno, o si la colaboración es remota, estableciéndose generalmente la comunicación a través de la red.

1.2 Desarrollo y Aplicaciones de la Realidad Aumentada.

A partir del año 1998 nacen los primeros congresos dedicados exclusivamente a la Realidad Aumentada. En [Zhou, 2004] se revisa los diez años del desarrollo de los trabajos presentados en la conferencia de ISMAR (*International Symposium on Mixed and Augmented Reality*) y sus predecesores con un enfoque particular en las técnicas de tracking, la interacción de estas en los SRA, display, entre otros temas. En el Anexo 1 se indican la fecha, lugar de celebración y nombre de estos simposios.

Es imposible obviar la labor del HITL (*Human Interface Technology Laboratory*) [HITL, 2010] en cuanto al desarrollo de la RA en los últimos tiempos. Este centro de investigación con sede en la Universidad de Canterbury en Nueva Zelanda posee entre sus temas de investigación la Realidad Virtual y Aumentada, Visualización Interactiva, entre otras temáticas para lograr revolucionar la forma en que las personas interactúan con las computadoras, mediante la creación de interfaces de vanguardia para mejorar las capacidades humanas, vencer las limitaciones humanas y aumentar la flexibilidad y la utilidad de los productos existentes de la industria.

1.2.1 Aplicaciones de Realidad Aumentada.

La RA se puede aplicar a cualquier área que nos imaginemos. En la actualidad se ha introducido en diversos campos, abarcando desde los ámbitos del entretenimiento, la medicina, la ingeniería, el arte, y la educación. A continuación se mencionan algunos ejemplos de las diferentes áreas donde se puede aplicar esta tecnología, para comprender la amplitud de esta tecnología.

Dentro del campo del entretenimiento existe una gran diversidad de SRA, basta con tener en cuenta la gran industria del videojuego que existe en la actualidad. Como ejemplo de aplicaciones en el área del

entretenimiento se encuentra Human Pacman (Figura 3) que es una aplicación de RA móvil que nació en el Mixed Reality Lab de la National University of Singapore, la misma está basada en el conocido video juego Pacman de Arcade de los años 80. El juego se desarrolla tanto fuera como dentro de un recinto, y los jugadores pueden interactuar de manera presencial con otros jugadores cuando físicamente están próximos, y de forma remota mediante una red inalámbrica [Cheok, 2003], [Cheok, 2004].

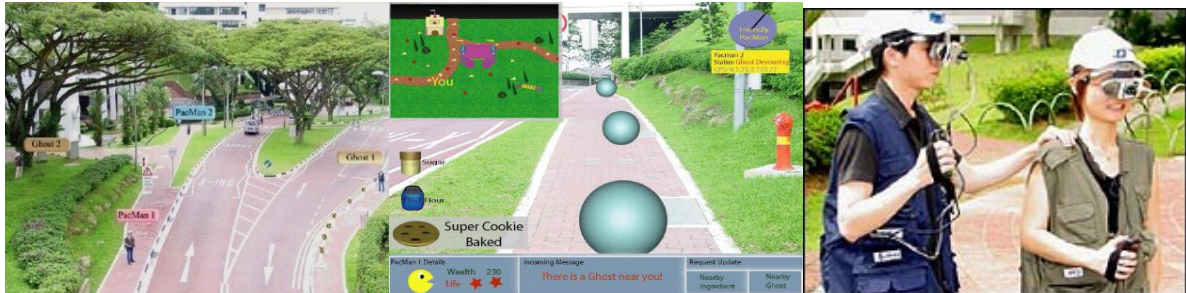


Figura 3: Imágenes de Human Pacman, en [Cheok, 2003].

Vinculadas al campo de la educación podemos encontrar también un creciente desarrollo de esta tecnología. Construct3D es una aplicación de RA específicamente diseñada para la educación en el campo de las matemáticas y la geometría como se muestra en la (Figura 4), combinando cuatro áreas de investigación: geometría, pedagogía, psicología y RA. “Es un sistema que trata de maximizar la transferencia de conocimientos y que permite varios escenarios de interacción entre alumnos y profesores, implementando métodos flexibles para el contexto y visualizaciones individualizadas de algunas partes.” [Kaufmann, 2002].

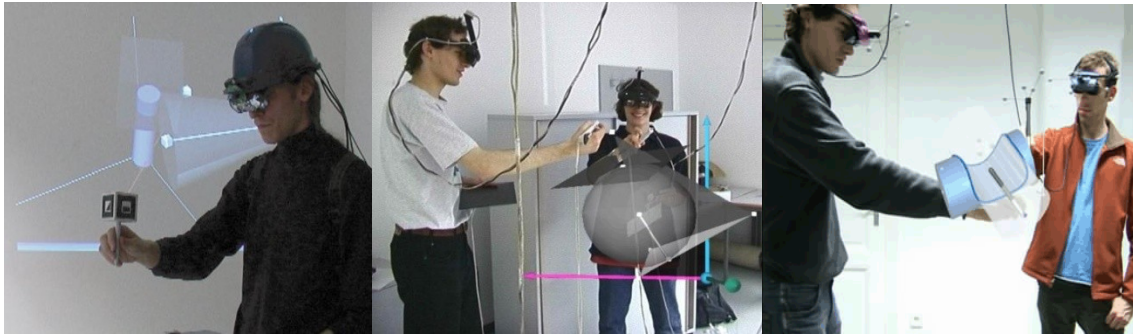


Figura 4: Usuarios interactuando con Construct3D .En [Kaufmann, 2002].

En el campo de la medicina es importante que los SRA tengan un alto grado de precisión y fiabilidad en el registro, sobre todo si estos sistemas se van a emplear en simulaciones de operaciones que puedan resultar críticas para los pacientes. A pesar de la poca madurez de la tecnología de RA, existen diversas actuaciones dentro del campo de la medicina, en [Wacker, 2005] se hace un estudio para adaptar un SRA a un escáner médico, y se prueba su viabilidad para utilizarlo en un tipo determinado de biopsias (Figura 5).

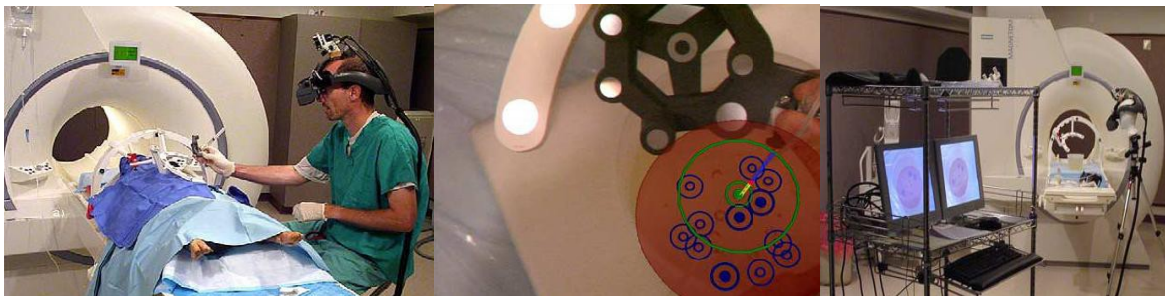


Figura 5: Sistema de RA adaptado a un escáner médico. En [Wacker, 2005].

En el área de la psicología también se logran grandes avances vale destacar el trabajo desarrollado en la Universidad Politécnica de Valencia por M. C. Juan que es pionero en el desarrollo de SRA para el tratamiento de distintos trastornos psicológicos. En [Juan, 2005] se presenta un SRA para el tratamiento de fobias a animales pequeños como cucarachas y arañas (Figura 6), Esta aplicación se probó con pacientes

reales que, después de alrededor de una hora de tratamiento con este sistema, fueron capaces de interactuar y matar animales reales.

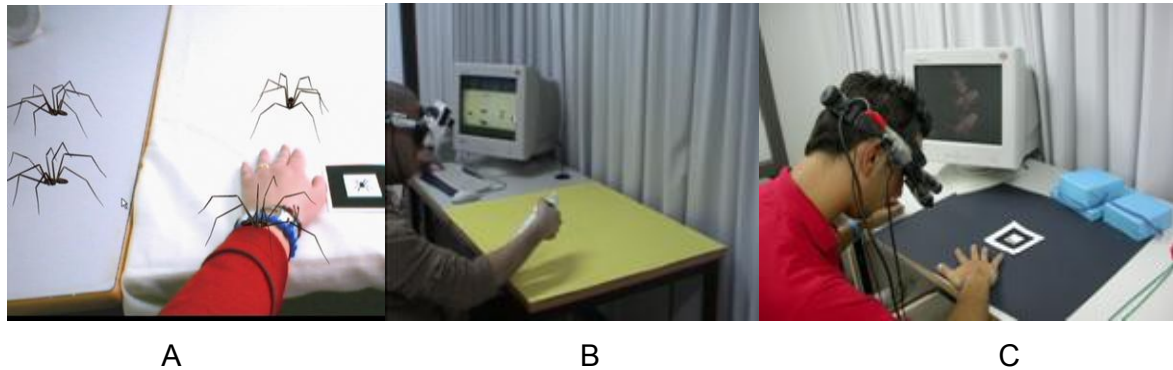


Figura 6: Tratamiento de fobias: A) Primer SRA para tratar fobias a insectos; B) SRA sin marcas; C) SRA mediante HMD óptico.

En el campo de la industria podemos encontrar diversos ejemplos, en [Dangelmaier, 2005] se describe la arquitectura de un SRA para dar soporte al proceso de planificación de sistemas de manufactura complejos (Figura 7). El sistema asiste al usuario en el modelado, la validación del modelo de simulación, y la consiguiente optimización de la producción del sistema.



Figura 7: Sistema de RA en procesos de manufactura. En [Dangelmaier, 2005].

Para las aplicaciones de RA un requerimiento es que las imágenes generadas por ordenador concuerden de manera precisa con el entorno del mundo real. En estas aplicaciones debe obtenerse información precisa del mundo real para poder llevar a cabo la calibración de los entornos real y virtual. Muchos autores coinciden que la parte más compleja de un SRA es el proceso de registro.

1.3 Definición de registro (*tracking*).

Según [Bimber, 2005], “el problema del registro es uno de los temas más desafiantes de la RA en nuestros días. El registro preciso, rápido y robusto de los usuarios, así como de los objetos reales y virtuales dentro de un entorno, es crítico para aplicaciones de RA”, entendiéndose por registro la captación del entorno y la transformación de esos datos en información útil que podrá ser utilizada para:

- Localizar al usuario (posición y orientación) dentro de un contexto determinado (campo de visión).
- Conocer la posición de un objeto que se encuentre dentro de ese contexto.
- Identificar un objeto y/o recuperar características únicas del mismo.

Según lo expresado por Borro, cuando se hace tracking “... se consigue disponer permanentemente de un sistema de referencia virtual alineado con el sistema de referencia del objeto real, y que por lo tanto realiza la misma transformación sólido rígida que este último (traslación y rotación)” [Borro, 2005].

1.3.1 Relaciones espaciales entre los entornos virtual y real

Se puede afirmar que sin el conocimiento preciso tanto de la geometría del mundo real como de la escena generada por ordenador, el registro exacto no es posible. Para alinear adecuadamente la imagen video y las imágenes generadas por ordenador, se deben tener en consideración varios sistemas de referencia [Barfield, 1995]. Como se puede ver en la (Figura 8), un punto situado sobre el área de trabajo se proyecta sobre la pantalla virtual. Los objetos del entorno real se pueden definir a partir de un sistema de coordenadas terrestre. El sistema de coordenadas del entorno virtual se define relacionando su localización

con respecto a la óptica del display físico. Los parámetros del entorno virtual incluyen tres rotaciones y tres traslaciones que determinan su localización respecto al sistema terrestre.

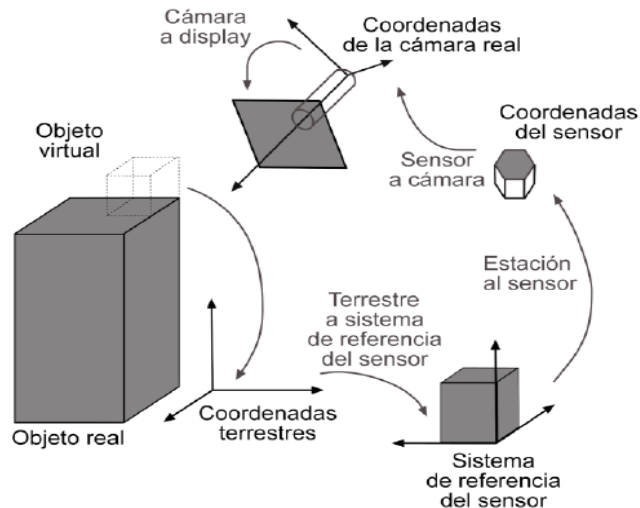


Figura 8: Sistemas de coordenadas real/virtual en aplicaciones basadas en registro por sensores para un sistema de referencia terrestre y un registro directo. A partir de [Azuma, 2004].

Además, puede haber otro par de sistemas de coordenadas que deben tenerse en consideración: el sistema de coordenadas propio del sensor que detecta la posición y rotación de la cabeza (ej. un sensor inercial combinado con un receptor GPS); y el sistema de coordenadas propio de los transmisores (ej. los satélites GPS) o dispositivos a partir de los cuales los sensores calculan su posición y rotación. Además, hay otros parámetros que cambian dependiendo de los usuarios, como la distancia interocular o la distancia vertical desde la posición del sensor, normalmente, situado en la parte superior de la cabeza, hasta el plano horizontal de los ojos. Así pues, para un registro adecuado de las imágenes, todos estos parámetros deben tenerse en cuenta en la calibración del sistema.

Los sistemas de coordenadas que emplean cámaras y marcas fiduciales difieren del esquema anterior. La diferencia esencial es que desaparece un sistema de coordenadas, el propio del sensor, por lo que el esquema se simplifica (Figura 9).

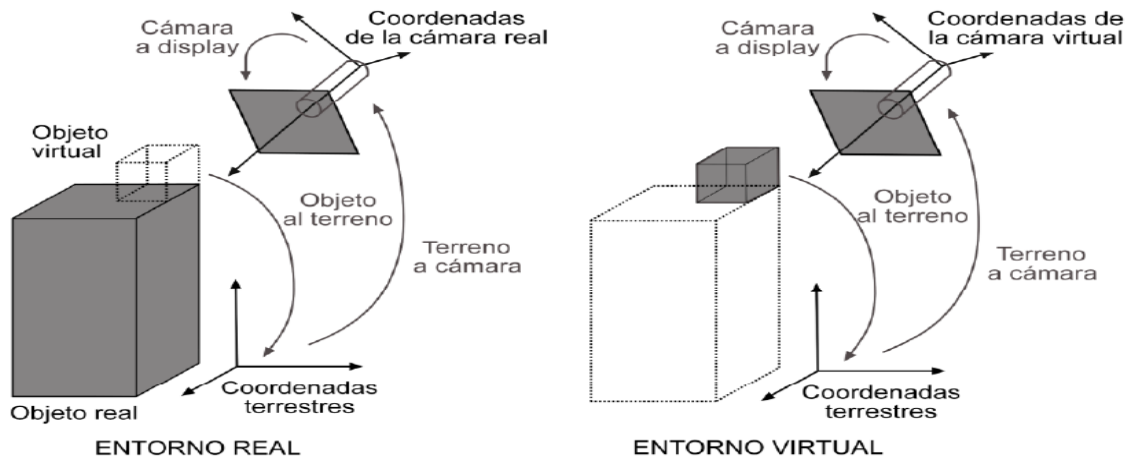


Figura 9: Sistemas de coordenadas real/virtual en aplicaciones basadas en registro por cámaras para un sistema de referencia terrestre y un registro basado en visión. A partir de [Azuma, 2004].

1.4 Técnicas de registro en los Sistemas de Realidad Aumentada.

Las técnicas de registro se pueden dividir en dos categorías: las basadas en sensores y las basadas en visión (visión por computador). A continuación se describen estas técnicas, abordando con mayor detalle las técnicas basadas en visión debido a que son las más importantes para lograr el objetivo de esta investigación.

1.4.1 Técnicas de registro basadas en sensores (*sensor-based*).

El registro basado en sensores se apoya en sensores tales como: magnéticos, acústicos, inerciales, medios ópticos y/o sensores mecánicos. Todos tienen sus respectivas ventajas y desventajas. Por ejemplo, los sensores magnéticos tienen una tasa de actualización alta y son ligeros, pero pueden ser distorsionados por cualquier sustancia metálica que altere las inmediaciones del campo magnético. En [Rolland, 2001] se aborda con más profundidad estas técnicas.

1.4.2 Técnicas de registro basadas en visión (*vision -based*).

Estas técnicas utilizan métodos de procesamiento de imágenes para calcular la orientación de la cámara en relación con los objetos del mundo real mediante técnicas y algoritmos matemáticos, a partir de los valores capturados por técnicas de visión por computador. En [Pressigout, 2006] se plantea que en la visión por ordenador, la mayoría de las técnicas de registro disponibles basadas en visión pueden dividirse en dos categorías: basada en características (*feature-based*) y basadas en modelos (*model-based*), en [Reitmayr, 2007], [Pascal, 2007], [Marchand, 2002] y [Marchand, 2006] concuerdan que estos métodos basados en modelos, son robustos a los cambios de luz y se abordan con más profundidad sus aplicaciones.

En los métodos de registro basados en visión, la obtención de la orientación externa de la cámara se realiza a partir del análisis de una o más imágenes. La correspondencia entre ambas imágenes se describe con la matriz de transformación, que incluye rotaciones, traslaciones y escalas o mediante vectores de desplazamiento. Se han desarrollado diversos sistemas de registro los cuales se pueden clasificar según [Neumann, 2001], atendiendo a los parámetros de entrada siguientes:

- Tres o más puntos con coordenadas 3D conocidas (que deben ser visibles en una imagen).
- Una secuencia de imágenes con correspondencias a partir de una cámara móvil, donde las coordenadas 3D de los puntos pueden ser conocidas o desconocidas.
- Un modelo 3D de la escena o plantillas/patronos de imagen (estando disponibles para relacionarlos a una imagen en concreto).

Los puntos necesarios para el registro pueden extraerse a partir de elementos naturales (esquinas, agujeros, etc.) o especialmente diseñados y ubicados en el entorno (puntos fiduciales, puntos luminosos, marcas de RA, etc.).

En el proceso de cálculo se suelen emplear diversos conceptos que son importantes distinguir:

- **Parámetros de orientación interna de la cámara:** O parámetros intrínsecos de la cámara, incluyen la distancia focal, las coordenadas del centro de la imagen y distorsiones radiales y tangenciales de la lente.
- **Calibración de la cámara:** Proceso mediante el cual se determinan los parámetros de orientación interna de la cámara. Algunos autores también incluyen en este término los parámetros de orientación externa [De la Escalera Hueso, 2001].
- **Parámetros de orientación externa de la cámara:** O parámetros extrínsecos de la cámara, incluyen la matriz de rotación y el vector 3D de translación. También se utiliza como sinónimo la matriz de transformación.
- **Orientación de la cámara:** Proceso mediante el cual se calculan los parámetros de orientación externa de la cámara [Jähne, 1999]. Cabe reseñar que en algunos métodos de orientación de la cámara (por ej., DLT), se obtienen simultáneamente los parámetros de orientación interna y los de orientación externa.

La orientación externa varía en aplicaciones en que la cámara se mueve y por tanto su cálculo deberá ser en tiempo real, mientras que la orientación interna suele permanecer constante y su cálculo será normalmente necesario una única vez, antes del cálculo de la orientación externa. También existen otros métodos por los que se calcula de manera simultánea tanto la orientación externa como la interna; estos métodos no suelen ser utilizados en aplicaciones de RA puesto que, al permanecer normalmente la orientación interna constante, sería un cálculo redundante.

1.4.3 Técnicas de registro híbrido.

Como se ha mencionado existen muchas formas de registro y aunque unas son más precisas que otras, ninguna está libre de errores. La distorsión óptica, la precisión o la necesidad de ahorrar algunos cálculos,

entre otros factores conllevan un error, que si bien se puede acotar y atenuar, no se puede eliminar por completo. En un sistema de tiempo real, un error pequeño puede convertirse en uno grande en muy poco tiempo.

Por cada fotograma procesado (de 15 a 60 frames por segundo dependiendo de la cámara) se calcula una nueva posición, lo que conlleva que el error se acumule en muy poco tiempo. “Para que un sistema de RA sea creíble y utilizable los objetos sintéticos deben estar alineados de forma precisa con la realidad” [Azuma, 1993]. Un cálculo impreciso o tolerante con los errores propiciará una representación errónea y confusa para el usuario. En algunos ámbitos de aplicación esto puede ser crítico, como por ejemplo en una cirugía apoyada por uno de estos SRA.

Algunos autores como Azuma en [Azuma, 2001] y [Azuma, 1993] defienden un registro híbrido como una posible solución de los problemas de registro, se dice que utilizar varias formas de tracking puede ser muy beneficioso para la minimización del error. Hay que remarcar que un mismo tipo de sensor puede implementar varias formas de registro. Según [Comport, 2006] ” con una cámara se podrían detectar marcas para obtener un posicionamiento global y se podrían detectar puntos de interés para el cálculo de posicionamiento relativo” Un registro híbrido se apoyaría en las partes más fuertes de las técnicas implementadas y de los sensores utilizados.

1.5 Estrategias del registro basado en visión.

Existen varias estrategias para realizar el proceso de registro basado en visión, tales como registro basado en marcadores (patrones de marcas planas) y el registro usando características naturales. En [Pascal, 2007] se debaten algunas de estas estrategias, sus fortalezas y debilidades, a continuación se describen brevemente estas estrategias de registro haciendo un análisis profundo del registro usando características naturales (natural feature tracking) ya que es el más importante para lograr los objetivos de la tesis.

1.5.1 Sistemas de patrones de marcas planas (*planar pattern marker systems*).

También se les conoce como sistemas de marcas fiduciales, resuelven el problema de la obtención de los parámetros de orientación externa de la cámara mediante la identificación de una sola de las marcas por

una cámara, aplicando para ello técnicas de visión por computador. La cámara debe ser calibrada previamente al registro, proceso en el cual se obtienen la distancia principal, las coordenadas planas del centro de proyección, y diversos parámetros de distorsión. Existen varios sistemas que permiten la detección de marcas, entre los que cabe destacar ARToolKit, ARTag, StudierStube y OSGART.

Generalmente, estas bibliotecas tienen una licencia de tipo GPL (*GNU General Public License*), lo que permite su uso libre con fines no comerciales. Este hecho, unido a los mínimos requerimientos de elementos físicos (una cámara Web y las marcas), a parte del ordenador, hace que estos sistemas sean unos de los más económicos, inmediatos y flexibles. En [Freeman, 2006], [Teichrieb, [2007] y [Guevara, 2008] podemos encontrar aplicaciones basadas en sistemas de marcas planas activas como también se les conoce.

1.5.2 Sistemas de registro de características naturales (*natural feature tracking*).

Hay que decir que el uso de los marcadores simplifica grandemente la tarea del registro, porque a veces es imposible llevar a cabo esta técnica sobre todo en entornos abiertos. Los sistemas de registro de rasgos naturales funcionan de manera similar a los de los patrones de marcas, mediante el análisis de una imagen capturada por técnicas de visión por computador. En primer lugar se procede a la identificación de una serie de puntos o entidades homólogas entre una imagen testigo y la imagen capturada en tiempo real, o entre dos imágenes contiguas capturadas por la cámara. La correspondencia entre ambas imágenes se describe con la matriz de transformación, que incluye rotaciones, translaciones y escalado, o mediante vectores de desplazamiento. Con estos elementos se establecen los parámetros de orientación externa de la cámara.

Según [Lowe, 1991] y [Jurie, 1998] los primeros métodos fueron basados en aristas, pero los métodos basados en la correspondencia de puntos característicos (*feature point matching*) se han vuelto populares, a partir de que las regiones invariantes locales en la imagen funcionan mejor para este propósito.

En [Guevara, 2008] se afirma que: “La técnica basada en la correspondencia de puntos característicos se acerca a ser la más robusta frente a los cambios de escalado, punto de visión e iluminación, así como a las oclusiones parciales. Operan bajo el siguiente principio: durante una etapa de entrenamiento, se construye una base de datos de puntos de interés (*keypoint*) pertenecientes al objeto y de los cuales se puede

calcular la posición que ocupan en su superficie. En tiempo de ejecución, los puntos de interés son primeramente extraídos de cada una de las imágenes de forma individual y luego se busca la correspondencia contra la base de datos. La posición del objeto puede ser estimada a partir de esta correspondencia. Tomando en cuenta lo expresado por [Pascal, 2007] para estimar la correspondencia, es muy conveniente utilizar los algoritmos de tipo RANdom SAmples Consensus (RANSAC) debido a que eliminan falsas correspondencias evadiendo problemas combinatorios.”

Por ejemplo, BazAR [CVLab, 2010] es una de las bibliotecas que realizan el análisis de la imagen para el cálculo de la orientación externa de la cámara a partir de la identificación de rasgos naturales puntuales (*natural feature point tracking*), permitiendo también el desarrollo de aplicaciones de RA. Algunas aplicaciones de RA que realizan este tipo de registro quedan descritas en [Cornelis, 2001] y [Prince, 2002] donde se presentan distintos sistemas de la estimación de los movimientos de la cámara, a partir de los vectores de desplazamiento entre puntos naturales homólogos en imágenes contiguas de una secuencia de vídeo. De otro modo, en [Lepetit, 2004a], [Lepetit, 2004b], [Lepetit, 2005] y [Pilet, 2006] se describe el funcionamiento de la biblioteca BazAR.

En [Guevara,2008] se mencionan dos familias de métodos de registro dependiendo de las características de las imágenes que se utilizan: métodos basados en aristas y métodos basados en texturas.

1.6 Entradas de los Sistemas de Realidad Aumentada. Sensores.

Los sensores realizan el registro del usuario y/o la cámara en tiempo real y proporcionan los datos de entrada (*inputs*) al ordenador en los SRA. Algunos autores distinguen entre registro de **fuera-a-dentro** y de **dentro-a-fuera**. El primer tipo, se refiere a los casos en que se aplican sensores fijos en el entorno que registran una serie de emisores sobre objetos móviles. El segundo tipo, utiliza sensores que están sujetos al objeto móvil. Existen muchos tipos de sensores utilizados en los SRA [NRC, 1994] entre los que se encuentran los sensores mecánicos, acústicos, sensores inerciales, sensores magnéticos y los sensores ópticos entre otros, estos últimos son los más importantes para la investigación.

1.6.1 Sensores ópticos.

Utilizando los sensores ópticos existen distintas maneras de realizar el registro de la posición y/o orientación. Según [NRC, 1994] la distancia se puede medir por triangulación (por ej. estéreo-visión), por tiempos (radar láser), o por interferometría. Se puede utilizar la luz pasiva del entorno (sistemas de estéreo-visión), la luz puede ser pulsada (radar láser), o se pueden poner sobre un cuerpo móvil una serie de marcas activas (diodos emisores de luz infrarroja – IREDs) o pasivas. Entre los sensores ópticos comúnmente utilizados en aplicaciones de RA se encuentran:

Sistemas de estéreo-visión pasivos.

Que utilizan un conjunto de dos o más cámaras para resolver el problema de correspondencia mediante la relación de una serie de puntos de muestra (*sample points*) en dos imágenes de un mismo entorno tomadas bajo distintos puntos de vista [Ribo, 2001].

Sistemas de marcas activas.

Estos se utilizan en la identificación de una serie de puntos fiduciales que son registrados sobre objetos móviles. El procedimiento más sencillo y preciso es utilizar una serie de IRED que crean unas marcas muy brillantes en las imágenes. Los IRED pueden ser pulsados en secuencia con la adquisición de la cámara para detectar inequívocamente cada marca. Para la detección, una técnica frecuentemente empleada es mediante PSDs (también llamados fotodiodos de efecto lateral). La luz incidente induce una corriente que se mide en cada esquina del cuadrado para el cálculo de las coordenadas [NRC, 1994].

1.7 Salidas de los Sistemas de Realidad Aumentada. Displays.

Los displays son aquellos dispositivos que muestran al usuario la información generada por ordenador, es decir, proporcionan los datos de salida (*outputs*) en los SRA. En aplicaciones de RA, este tipo de información es predominantemente de tipo gráfica o visual. Existen muchos tipos de displays, en esta investigación solo se mencionan brevemente los displays visuales y los displays espaciales.

1.7.1 Display Visuales.

“Los displays visuales o gráficos son dispositivos que permiten visualizar la imagen y están configurados por una serie de componentes ópticos, electrónicos y mecánicos para generar imágenes en algún lugar entre los ojos del usuario y el objeto físico que se aumenta” [Bimber, 2005]. Dependiendo de la óptica que se utilice, la imagen se puede formar en un plano o en una superficie no-plana más compleja. Los displays para la cabeza (*Head-Attached Displays*, HAD) requieren que el usuario lleve puesto el display en su cabeza, a la altura de sus ojos; dependiendo de la tecnología de generación de la imagen, existen tres tipos fundamentales: displays retinales, que emplean tecnología láser a baja potencia para proyectar imágenes directamente sobre la retina del ojo; displays sujetos a la cabeza (*Head Mounted Displays* – HMD) que utilizan pequeños displays a modo de gafas delante de los ojos; y proyectores sujetos a la cabeza (*Head-Mounted Projectors* – HMP) que utilizan mini-proyectores o mini-paneles LCD con luz negra y proyectan imágenes sobre superficies del entorno real.



A

B

C

Figura 10: Displays visuales. A) HMD de video con cámaras integradas ARvision-3D HMD. B) HMP y C) Display retinales.

1.7.2 Display Espaciales.

En contraste con los displays sujetos al cuerpo, los displays espaciales separan la mayoría de la tecnología del usuario y la integran en el entorno físico. Al igual que en los casos anteriores, existen tres técnicas que principalmente difieren en la manera en que aumentan el entorno: de vídeo, ópticos o proyectores.

Los displays espaciales ópticos (*Spatial Optical see – through Displays, SOD*) generan imágenes que están alineadas con el entorno físico. Los mezcladores espaciales ópticos, tales como espejos divisores del haz de luz, pantallas transparentes u hologramas ópticos, son componentes característicos de estos displays. Sus principales ventajas son una mayor facilidad de acomodación visual y convergencia, una resolución mayor y escalable, un mayor campo de visión, factores ergonómicos mejorados, una calibración más sencilla y más estable, y un entorno más controlable. Sin embargo, también existen una serie de desventajas que están descritas en [Bimber, 2005].

Los displays espaciales de proyectores (*Projector-based Spatial Displays, PSD*) aplican proyección frontal para proyectar imágenes directamente sobre superficies de objetos físicos. Se pueden utilizar tanto proyectores orientables como estáticos, o incluso varios proyectores para incrementar el área de display. Según [Bimber, 2005], si se muestran gráficos tridimensionales sobre la superficie de los objetos, se requiere una proyección estereoscópica, que a su vez depende del punto de vista del observador. Algunas ventajas de estos displays son que proporcionan un campo de visión teóricamente ilimitado, resolución escalable, y una mayor facilidad de acomodación visual.

1.7.3 Display Espaciales de Vídeo.

Los displays espaciales de vídeo (*Screen-Based Video See-through Displays, SBVD*) hacen uso de mezcla de vídeos y muestran las imágenes fusionadas en un monitor corriente. La técnica de aumentación basada en monitores de vídeo es bastante frecuente en aplicaciones de RA no móviles. Esto es debido a que representa probablemente el sistema más económico de RA, ya que sólo se requieren componentes de hardware comerciales y un PC estándar. Sin embargo, existen una serie de desventajas según [Bimber, 2005]:

- Al igual que sucede con algunos sistemas de RV basados en monitores, estos sistemas proveen de un bajo grado de inmersión, ya que el campo de visión se limita y restringe al tamaño del monitor, su alineación espacial con respecto al observador, y su distancia al observador.
- Campos de visión reducidos debido al tamaño relativamente pequeño de los monitores.
- La mayoría proveen una vista remota.
- Normalmente no soportan interacción directa con el entorno real y la aumentación gráfica.
- Solamente soportan técnicas de interacción indirecta y/o remota.

1.8 Bibliotecas para desarrollar aplicaciones de Realidad Aumentada.

Existen varias bibliotecas para desarrollar aplicaciones de RA, con vista a seleccionar la más adecuada para cumplir con el objetivo investigación de esta tesis, se analizaron algunas que utilicen técnicas de registro basadas en visión. En la siguiente tabla (Tabla 1) se hace una comparación entre estas bibliotecas en cuanto a algunos parámetros, como la versión actual del producto, portabilidad, el tipo de licencia y técnica de registro que lo caracteriza.

Bibliotecas	Versión	Licencia	Registro	Portabilidad
ARToolKit	2.x	GPL	Marcadores cuadrados (MC)	Multiplataforma
ARToolKit Profesional	4.3	Comercial	Marcadores cuadrados	Multiplataforma
ARToolKit NFT	2.3	Comercial	Características Naturales	Multiplataforma
SSTT	1.0	Dual	Logotipos gráficos	Multiplataforma
Windage	Desarrollo	GPL	Características Naturales	No es multiplataforma
StudierTube	4.4	Dual	MC con codificación binaria	No es multiplataforma
BazAR	1.3	GPL	Características Naturales	Multiplataforma

Tabla 1: Software para desarrollar aplicaciones de RA.

Como se puede apreciar en dicha tabla existen tres bibliotecas para desarrollar aplicaciones de RA que se basan en registro de características naturales y una que reconoce logotipos, esta última SSTT (*Simplified Spatial Target Tracker*) es multiplataforma y realiza el registro mediante el reconocimiento de logotipos gráficos y contiene una licencia comercial para su uso en dispositivos móviles [SSTT, 2009].

Entre las bibliotecas que utilizan técnicas de reconocimiento de características naturales analizadas, resalta como la más potente ARToolKit NFT (*ARToolKit Natural Feature Tracking*), esta cuenta con el algoritmo más potente y preciso para realizar este tipo de registro pero su licencia es comercial y reportaría un costo de 5000 dólares para fines educativos por cada doce personas, un lujo que no se puede dar la línea Interacción 3D del CEDIN.

También se investigó sobre Windage, una biblioteca que reconoce características naturales en imágenes planas y que se encuentra actualmente en desarrollo, como principal fortaleza esta biblioteca pretende resolver el problema del registro múltiple de características naturales.

En los siguientes epígrafes (1.9.1 y 1.9.2) se analizan con mayor profundidad ARToolKit 2.x y BazAR, la primera debido a que es la utilizada en la herramienta con que cuenta actualmente la línea de investigación, y la segunda porque es una biblioteca multiplataforma que permite la técnica de registro más importante para esta investigación, además de tener una versión libre y estable.

1.9.1 Software basado en patrones de marcas planas (ARToolKit).

ARToolKit es un software que consta de un conjunto de bibliotecas escritas en C y C++ que permite a los usuarios el desarrollo de aplicaciones de RA. Utiliza técnicas de visión por computador para el cálculo de la orientación externa de la cámara relativa a una serie de marcas con unas características específicas, permitiendo que se superpongan a ellas objetos virtuales. Como se explicó anteriormente ARToolKit utiliza patrones para determinar la posición en la que se desea insertar el objeto virtual. Pero este tipo de registro puede llevar a frustraciones cuando por ejemplo el marcador asociado se oscurece, existen muchas más limitaciones descritas a continuación:

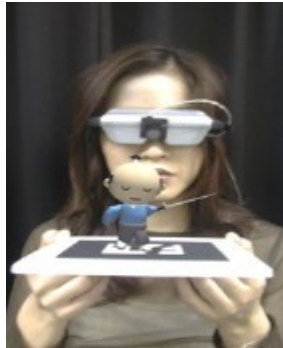


Figura 11: ARToolkit facilita el desarrollo de Realidad Aumentada utilizando marcadores.

Limitaciones:

- Los objetos virtuales solamente aparecen cuando las marcas físicas son visibles a la cámara. Este hecho limita el movimiento del usuario en el entorno y, además, implica que las marcas no deben ocultarse total, ni parcialmente (por ejemplo, por las manos del usuario o por otros objetos físicos).
- Existen también unas distancias críticas que deben ser consideradas para que la cámara reconozca correctamente las marcas. A mayor tamaño de las marcas físicas, más se podrá alejar la cámara de esta, además a mayor calidad de la imagen mejor y más rápido reconoce el patrón.
- Esta distancia también se ve afectada por la complejidad de los patrones dentro de la marca. A mayor simplicidad de los patrones, el reconocimiento será mejor. Los patrones con grandes regiones negras y blancas (patrones de baja frecuencia) son los más efectivos.
- El registro también se ve condicionado por la orientación relativa entre la marca y la cámara. A mayor grado de inclinación, menos fiable será el reconocimiento.
- Por último, el registro también depende de las condiciones de iluminación del ambiente. Las luces directas pueden producir reflejos y puntos brillantes sobre las marcas. Para evitar este problema se pueden utilizar superficies mate o entornos de iluminación controladas.

Mediante el desarrollo de esta investigación se pretende utilizar una herramienta que rompa con muchas de esas limitaciones excluyendo además el uso de marcadores para el registro.

1.9.2 Software de reconocimiento de rasgos naturales (BazAR).

BazAR [CVLab, 2007] es una biblioteca de visión por computador basada en la detección y correspondencia de rasgos naturales puntuales. Es capaz de detectar y registrar rápidamente objetos conocidos en imágenes planas, también contiene un sistema de calibración geométrica y fotométrica de la cámara bastante potente. Este software se ha desarrollado y se distribuye por el laboratorio CVLab de la EPFL (Ecole Polytechnique Federale Lausanne) entre sus características destacan que se basa en eficientes técnicas de visión por computador y trabaja en tiempo real con una o más cámaras, además de poseer un detector de texturas de objetos planos muy rápido. Con BazAR se puede utilizar la API OpenGL para aumentar el flujo de vídeo mediante la inserción artificial de objetos 3D, sombreando con el mapa de luz (*lightmap*) calibrado la superficie de dicho objeto.



Figura 12: Bazar facilita el desarrollo de Realidad Aumentada basadas en la detección y correspondencia de rasgos naturales puntuales.

1.9.3 Principios básicos de BazAR.

Las bibliotecas de BazAR dependen de OpenCV para su funcionamiento. BazAR está organizado en dos módulos:

Garfeild: Este módulo depende del segundo (Starter) y proporciona las herramientas para detectar puntos clave o rasgos puntuales, y establecer su correspondencia respecto de una serie de puntos almacenados (puntos modelos). Garfeild está diseñado para detectar rápidamente objetos planos en una imagen mediante dos fases de reconocimiento y correspondencias, o fase en tiempo real (on-line). Durante la fase de entrenamiento se detectan los rasgos puntuales de una imagen modelo. Se extraen sus elementos, se corrigen sus orientaciones y se aprende su apariencia. Durante la segunda fase, un algoritmo de reconocimiento de objetos planos detecta los rasgos puntuales de la imagen de entrada para establecer las correspondencias con los puntos modelo. A continuación, se activa un procedimiento para discernir correspondencias buenas y malas, y encontrar la homografía que relaciona los puntos modelo con los de entrada.

Starter: Contiene las estructuras básicas y las herramientas matemáticas.

Conclusiones del capítulo.

En este capítulo se puede apreciar que la Realidad Aumentada es una tecnología relativamente nueva y en constante desarrollo y evolución; que tiene como principal problemática lograr productos robustos apoyándose en técnicas eficientes de registro 3D de los elementos virtuales en el entorno real. La RA tiene un gran potencial en muchos campos de conocimiento y prueba de ello son los distintos SRA creados por distintos autores, a partir de las aplicaciones referenciadas se aprecia que esta tecnología avanza hacia los sistemas colaborativos, multimodales, altamente interactivos, y con la integración de dispositivos híbridos que realizan un registro 3D en tiempo real. Todos estos dispositivos son cada vez más ligeros, por lo que en las aplicaciones móviles de RA, el ordenador portátil se sustituye por PDAs o celulares con cámaras integradas y/o GPS, con capacidad suficiente para soportar SRA. Las aplicaciones no se limitan a dispositivos de visualización tipo HMD, sino que surge una gran variedad de displays, incluyendo proyecciones 3D. Al analizar algunas de las bibliotecas que registran elementos virtuales mediante técnicas de visión por computadora para desarrollar aplicaciones de RA, se pudo apreciar que las más potentes o que tiene un despliegue en tecnologías móviles reportaría un alto costo utilizarlas, por lo que fue seleccionada Bazar como biblioteca candidata para resolver la problemática de esta tesis. Resumiendo, se puede afirmar que con el desarrollo de este capítulo se trataron los aspectos teóricos necesarios para llevar a cabo la confección de la solución que propone esta investigación.

Capítulo 2: Descripción de la solución propuesta.

Introducción del capítulo.

En este capítulo se realiza la descripción de la solución propuesta a través de la metodología de desarrollo de software seleccionada. Se describen los conceptos y procesos fundamentales que intervinieron en la construcción del componente. Se especifican las condiciones y los pasos a tener en cuenta para el correcto funcionamiento del sistema a desarrollar. Se detallan las capacidades que cumple la solución a partir de los requisitos funcionales, así como las características de dicha solución expresadas por los requisitos no funcionales.

2.1 BazAR como biblioteca para desarrollar la solución.

Para elaborar aplicaciones de software existen dos alternativas, una es realizarlas implementando cada una de las herramientas que se necesitan para desarrollar la aplicación, y la otra variante es reutilizar herramientas existentes como bibliotecas del tema que agilizan el proceso de desarrollo. En esta investigación se asumió la segunda variante para obtener un resultado acorde con el objetivo de la misma. Existen muchas bibliotecas para el desarrollo de aplicaciones de RA, en esta investigación se escogió a BazAR ya que permite el registro mediante el reconocimiento de características naturales. Justifica esta elección el análisis de las características expresadas en el epígrafe 1.8.2 del Capítulo 1 además de las ventajas que a continuación se aprecian:

2.1.1 Ventajas y Desventajas de BazAR.

En este software se reducen algunas de las limitaciones encontradas en la biblioteca ARToolKit.

Ventajas:

- La imagen modelo se puede ocultar parcialmente sin impedir el registro, por lo que el usuario puede seguir viendo los objetos virtuales.
- Trabaja mejor en condiciones de luz menos favorables.

Capítulo 2: Descripción de la Solución Propuesta.

- No existe la necesidad de ubicar elementos artificiales en el entorno físico.

Sin embargo, existen una serie de limitaciones:

Desventajas:

- El software que se distribuye sólo contempla la posibilidad del reconocimiento de una sola imagen.
- La complejidad del algoritmo de registro hace que el software requiera de ordenadores potentes.

2.1.2 Proceso de registro de BazAR.

Según los autores de BazAR, en esta biblioteca se incluye un procedimiento novedoso para la correspondencia de rasgos puntuales bajo numerosos puntos de vista y cambios de iluminación, que es adecuado para la estimación precisa de la matriz de transformación con un coste computacional menor que en otros métodos anteriormente utilizados por otras soluciones de este tipo.

En [Lepetit, 2004a] se explica el procedimiento, que se basa en tratar las correspondencias de rasgos puntuales como un problema de clasificación, en el que cada clase se corresponde con un conjunto de todas las posibles vistas de dicho rasgo puntual. Durante la fase de entrenamiento, proporcionando como mínimo una imagen patrón, se sintetizan un gran número de vistas de puntos clave individuales (*keypoints*). Si se asume que el objeto es localmente plano, bastará con envolver los elementos de la imagen alrededor de los puntos bajo deformaciones afines u homografías. Por otra parte, dado un modelo 3D, se utilizan técnicas estándar de mapeado de texturas. En tiempo de ejecución, se utilizan técnicas potentes y rápidas de clasificación para decidir a qué conjunto, si lo hay, pertenece el elemento observado. Una vez que las correspondencias potenciales se han establecido entre los puntos de interés de la imagen de entrada y los del objeto (imagen modelo), se aplica un método basado en el procedimiento estándar denominado RANSAC para estimar la matriz de transformación.

2.2 Propuesta de solución.

Según Clemens Szyperski: “Un componente es una unidad de composición de aplicaciones software, que posee un conjunto de interfaces y un conjunto de requisitos, y que ha de poder ser desarrollado, adquirido,

Capítulo 2: Descripción de la Solución Propuesta.

incorporado al sistema y compuesto con otros componentes de forma independiente, en tiempo y espacio". Adquiriendo dicha definición se propone la construcción de un componente de software que sea capaz de brindar una interfaz de funcionalidades para realizar aplicaciones de Realidad Aumentada, mediante la utilización de la biblioteca BazAR que permite el registro de las características naturales de una imagen de superficie plana. Para referirse a dicho componente de software en los siguientes puntos se escogió el nombre de IRA que significa Interfaz de Realidad Aumentada.

La construcción de este componente de funcionalidades constituye el cumplimiento del objetivo de investigación planteado, y su realización consiste en brindar funcionalidades como cargar los datos del proceso de entrenamiento y realizar el registro 3D. La funcionalidad más importante que debe brindar IRA es realizar el registro tridimensional coherente para lo que es vital actualizar la matriz de proyección de la(s) cámara(s) y calcular con la mayor precisión la matriz de transformación del objeto virtual a aumentar, además se debe actualizar constantemente el mecanismo de detección de características naturales que provee BazAR.

Es necesario especificar que la solución brindada depende de un proceso de entrenamiento, por llamarlo de alguna manera, donde se realiza la configuración del mecanismo de detección de la imagen plana (modelo) y se realiza la calibración de la(s) cámara(s), la información de las actividades de este proceso se salvan en archivos que constituyen las entradas principales del componente a desarrollar.

2.3 Soluciones técnicas.

Para la elaboración de un programa se necesita utilizar uno o varios lenguajes de programación, así como apoyarse en alguna(s) herramienta(s) y metodología de desarrollo de software para lograr un producto con la calidad requerida. Para el desarrollo del componente IRA se utilizó una webcam conectada a la PC como dispositivo de captura del flujo de video proveniente del mundo y se eligió la variante de display espacial de vídeo (monitor de PC) como dispositivo de salida, ya que estos dispositivos resultaron ser los indicados para la investigación, además de constituir los recursos que posee actualmente la línea de investigación Interacción 3D. En los siguientes sub-epígrafes se justifica la selección de las herramientas, tecnologías y metodología para desarrollar la solución.

2.3.1 Metodología de desarrollo de software RUP.

Una metodología es el conjunto ordenado de pasos a seguir para cumplir un objetivo. Dicho objetivo, en la ingeniería de software, es el desarrollo con alta calidad de un producto que cumpla con las necesidades del cliente dentro de un plan y un presupuesto predecible. Para el desarrollo de la solución se usó la metodología del Proceso Unificado de Rational (RUP por sus siglas en inglés) ya que permite desarrollar aplicaciones sacando el máximo provecho de las tecnologías, mejorando la calidad, el rendimiento, la reutilización, la seguridad y el mantenimiento del software mediante una gestión sistemática.

2.3.2 Lenguaje de programación C++.

C++ se utiliza para desarrollar aplicaciones gráficas en 3D por ser uno de los lenguajes de programación que con más velocidad ejecutan el código (menor costo de ejecución del programa). Algunas de las características principales por las que se seleccionó este lenguaje para la implementación de la solución son:

- Es un lenguaje muy potente y rápido, solo superado por el ensamblador
- Es el lenguaje en el que se tiene una mayor experiencia por parte de los autores de la investigación
- Es un lenguaje de programación de propósito general, especialmente indicado para la programación de sistemas por su flexibilidad y potencia.
- Es uno de los lenguajes más utilizados por la comunidad de desarrollo de software, incluyendo la programación gráfica.
- Abarca tres paradigmas de la programación: la programación estructurada, la programación genérica y la programación orientada a objetos (OO).

2.3.3 OpenGL como biblioteca gráfica.

OpenGL es la biblioteca gráfica 3D por excelencia, puede utilizarse en casi todos los Sistemas Operativos. Desarrollada originalmente por Silicon Graphics Incorporated (SGI), es una biblioteca de gráficos que ofrece al programador una Interfaz de Programación de Aplicaciones (API). OpenGL se propone para el empleo de hardware informático diseñado y optimizado para la visualización y manipulación de gráficos 3D, soportando iluminación y sombreado, mapeado con texturas, animación y otros efectos especiales.

Existen algunas herramientas que facilitan la labor de unir OpenGL con un entorno de ventanas, ya que este no incorpora rutinas para el manejo de ventanas, este conjunto de rutinas es diferente para cada sistema operativo y no pertenece a OpenGL, por tanto, se utilizó GLUT (*OpenGL Utility Toolkit*) para la construcción de la aplicación demostrativa.

2.3.4 UML como lenguaje de modelado.

El Lenguaje Unificado de Modelado (UML por sus siglas en inglés) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Es un lenguaje gráfico, ya que cuenta con un grupo de diagramas, los cuales muestran diferentes aspectos de las entidades representadas y que se utilizan para visualizar, especificar, construir y documentar un sistema de software, en cada una de las etapas por las que tiene que pasar el mismo. Ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables.

2.3.5 Visual Paradigm para UML como herramienta CASE.

Visual Paradigm para UML es una herramienta profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue, incluye también actividades como la gestión de proyectos y la estimación. El software de modelado UML ayuda a una construcción de aplicaciones más rápida, de calidad, mejores y a un menor coste. Es fácil de usar y

Capítulo2: Descripción de la Solución Propuesta.

permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Incluye además importación desde Rational Rose, exportación/importación XML, generador de informes, editor de figuras, integración con MS Visio, plug-in, integración IDE con Visual Studio, IntelliJ IDEA, Eclipse, NetBeans y otros. Apoya un conjunto de lenguajes (entre ellos C++) tanto en la generación del código como en la Ingeniería Inversa.

2.3.6 Microsoft Visual Studio 2008.

Microsoft Visual Studio.NET 2008 es Ambiente Integrado de Desarrollo (IDE por sus siglas en inglés) de probada eficacia para construir aplicaciones de alto rendimiento permitiéndole a los desarrolladores crear rápidamente aplicaciones conectadas con sistemas ya existentes, ya que ofrece funcionalidades para generar componentes sólidos, utilizando el lenguaje de programación C++. También soporta las bibliotecas gráficas OpenGL, DirectX que son las utilizadas por excelencia en la industria de los gráficos por computadoras. Es válido destacar que no se trabajó con el framework de .NET, esencialmente se utilizó el Visual Studio como IDE y no como plataforma de desarrollo.

2.4 Entradas requeridas para el funcionamiento del componente.

En este epígrafe se describen las entradas requeridas para IRA, así como las características y modo de obtención de las mismas. Vale aclarar que el proceso de entrenamiento se asume como un proceso apartado (*off-line*) del trabajo de reconocimiento y aumento que maneja IRA. Dichas entradas son: el flujo de video que capturado por la webcam y ficheros de configuración de la fase de entrenamiento.

Los ficheros generados en el entrenamiento guardan información útil para realizar el reconocimiento de la superficie plana que contiene la imagen modelo, actualizar las matrices de proyección de la(s) cámara(s) y la matriz de transformación necesaria para dibujar de forma coherente el objeto tridimensional que se desee aumentar. Los parámetros de la cámara se obtienen a partir de un proceso de calibración el cual permite conocer los valores intrínsecos principales y necesarios para la representación de la cámara.

Los parámetros salvados en los ficheros son:

- Alto y ancho de resolución de la(s) cámara(s) en píxel.

Capítulo 2: Descripción de la Solución Propuesta.

- Coordenadas de la región de interés en la imagen plana (patrón de calibración o modelo).
- Matrices de rotación traslación de la cámara con respecto al patrón de calibración (Matriz de Proyección).
- Imagen de la superficie plana que contiene dicho modelo.

2.4.1 Obtención de las entradas.

BazAR provee una aplicación llamada FullCalib que permite realizar el proceso de calibración de varias cámaras simultáneamente (el modelo de cámara utilizado es el Pin-hole) [Isern, 2003], donde se calcula la geometría de la(s) cámara(s) y un mapa de iluminación ambiental. La única intervención manual requerida es el movimiento arbitrario de la imagen plana dentro del campo de visión de la(s) cámara(s) y los datos resultantes de este proceso son guardados en ficheros que constituyen las entradas anteriormente mencionadas. En los siguientes puntos (y figuras) se explican los pasos para la calibración y puesta en marcha de esta sencilla aplicación:

a. Adquisición del modelo: En primer lugar, hay que presentar la imagen plana a la cámara, a continuación presionar la barra espaciadora para crear una imagen fija, y finalmente arrastrar un marco verde que aparece sobre la imagen para seleccionar el modelo (imagen de referencia o patrón de calibración) deseado dentro de la imagen.

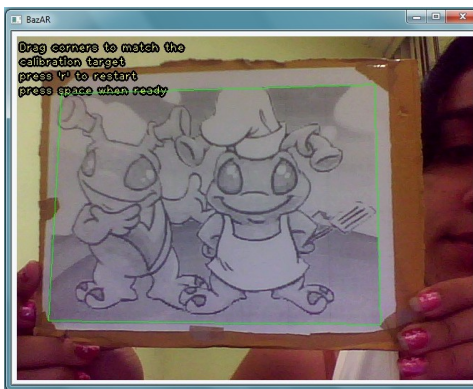


Figura 13: Adquisición del modelo.

Capítulo 2: Descripción de la Solución Propuesta.

b. Adquisición de datos para la calibración de la cámara: Hay que mover el patrón de calibración dentro del campo de visión de la cámara. Unos puntos verdes muestran los rasgos puntuales correctamente detectados.

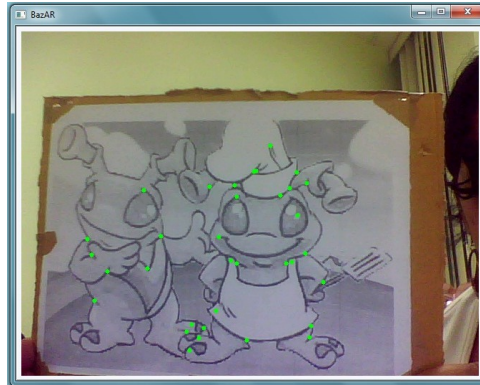


Figura 14: Adquisición de los datos.

c. Cómputo de la calibración geométrica: El sistema calcula de forma automática la calibración de la(s) cámara(s). Primeramente, el sistema calcula las homografías entre el plano geométrico de la imagen capturada y la imagen modelo. A continuación se retienen las homografías más fiables y sus *frames* correspondientes para calcular los parámetros intrínsecos de la cámara y la matriz de transformación respecto al objeto plano en cada uno de los *frames* correspondientes. Finalmente, se calcula la matriz de transformación entre las diversas cámaras. Durante el proceso aparece un histograma que muestra los errores de la re-proyección en unidades de píxel. La parte negra muestra el límite de un píxel. Este paso es completamente automático.

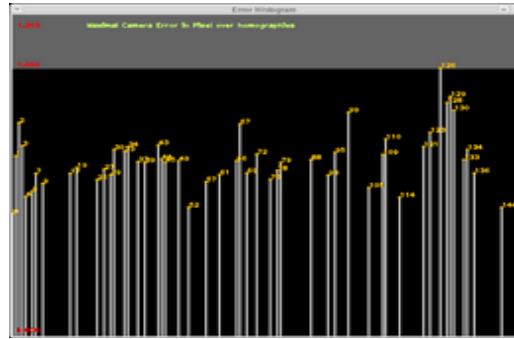


Figura 15: Cómputo de la calibración geométrica. Tomado de [CVLab, 2010].

d. Calibración fotométrica: Después de la calibración geométrica, es posible aumentar los objetos mediante elementos virtuales tridimensionales. Sin embargo, todavía no existe ningún mapa de iluminación para proporcionar una visualización más idónea de los objetos virtuales. Es por ello que al inicio de la fase de aumentación el sistema recoge datos de iluminación para calcular el mapa de irradiación. Para la calibración fotométrica se utiliza la misma imagen modelo que en la calibración geométrica. Para cada una de las cámaras, siempre y cuando la iluminación y los parámetros de las cámaras no cambien, las intensidades de los píxeles dentro del patrón de calibración dependen únicamente de su normal. Es decir, cada imagen de entrada en la que se detecta el patrón proporciona un número de ejemplos correspondientes a la orientación de una superficie individual. Debido a que se registran automáticamente muchos ejemplos, se puede simultáneamente recuperar la ganancia (*gain*) y la desviación (*bias*) de cada una de las cámaras y crear un modelo de iluminación ambiental que se puede utilizar con fines de re-iluminación.

2.5 Modelo de Domino.

Los procesos de negocio que dan lugar a la solución a implementar no se encuentran bien definidos, es decir, no son visibles y las fronteras no están bien establecidas. Por lo antes afirmado, se realizó este modelo (Figura 16), que permitió de manera visual mostrar los principales conceptos que se manejan en el dominio del sistema a desarrollar y un mejor entendimiento de sus relaciones.

Capítulo 2: Descripción de la Solución Propuesta.

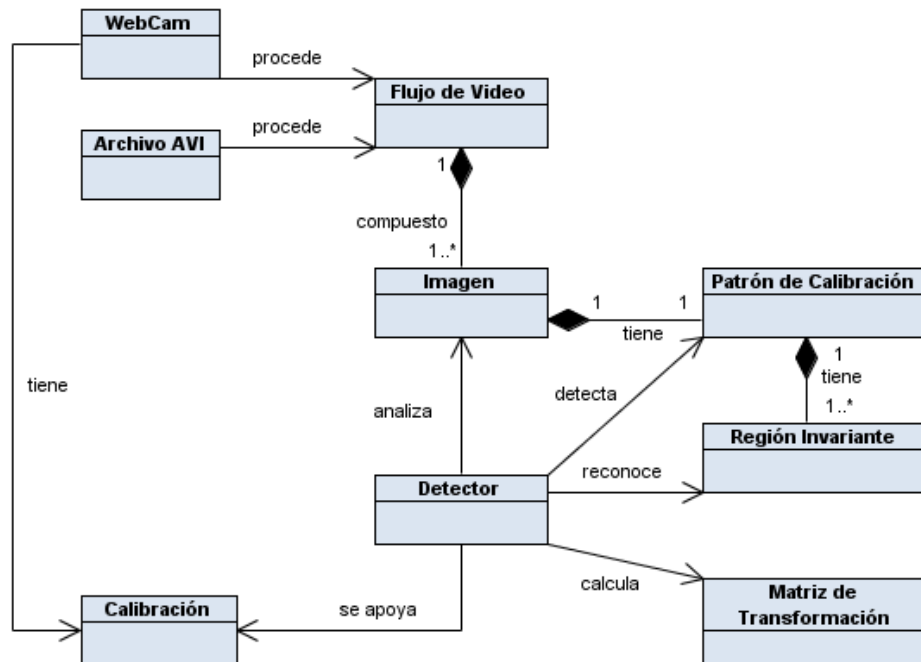


Figura 16: Modelo de dominio.

2.5.1 Descripción del modelo de dominio.

El **flujo de video** analizado puede proceder de un **archivo de video** o de la captura de una **cámara** (webcam), el mismo está compuesto una secuencia de **imágenes** (*frames*) que representan escenas en movimiento. El **detector** representa el mecanismo capacitado para analizar cada frame de dicho flujo de video y reconocer las **regiones invariantes** en el **patrón de calibración** seleccionado en la fase de entrenamiento de la imagen modelo y **calibración** de la cámara. Luego el detector puede calcular y proveer la **matriz de transformación** aplicable al objeto para aumentar sobre la superficie plana (que contenga el patrón de calibración) en la escena filmada.

2.6 Requerimientos del sistema.

Los requisitos constituyen capacidades o condiciones que el sistema debe cumplir. Estos facilitan el entendimiento entre usuarios y desarrolladores del sistema a elaborar. A continuación se exponen los

Capítulo 2: Descripción de la Solución Propuesta.

requisitos funcionales por los que se regirá el sistema y los no funcionales que exponen las características de IRA.

2.6.1 Requisitos Funcionales (RF).

Los siguientes requisitos establecen las funcionalidades e instrucciones que el sistema debe cumplir en su implementación.

1. Cargar los ficheros de configuración del proceso de entrenamiento.
 - 1.1 Cargar los ficheros de calibración de la(s) cámara(s).
 - 1.2 Cargar los ficheros del mecanismo de detección de características naturales.
2. Realizar el registro de elementos virtuales en tres dimensiones en la escena real.
 - 2.1 Devolver la matriz de transformación del objeto virtual con respecto a la cámara.
 - 2.2 Devolver la matriz de proyección de la(s) cámara(s).
 - 2.3 Actualizar en tiempo real el mecanismo de detección de la imagen modelo.

2.6.2 Requisitos no funcionales (RNF).

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener. Son importantes para que clientes y usuarios puedan valorar las características no funcionales del producto, como cuán usable, seguro, conveniente y agradable es el sistema, estas son cualidades que pueden marcar la diferencia entre un producto bien aceptado y uno con poca aceptación. Existen varios tipos de requisitos no funcionales. A continuación se presentan los correspondientes a la solución propuesta.

- Usabilidad:** Brindar una interfaz de funcionalidades bien definida, de fácil comprensión y uso, la cual se le pueda añadir como complemento a cualquier SRA.

Capítulo 2: Descripción de la Solución Propuesta.

- ❑ **Rendimiento:** La aplicación debe ejecutarse y mostrar los elementos virtuales en tiempo real, debe tener alta velocidad de procesamiento o cálculo.
- ❑ **Diseño e Implementación:** Se regirá por el paradigma de Programación Orientada a Objetos (POO) y se usará el lenguaje de programación C/C++.
- ❑ **Soporte:** En su versión inicial deberá ser compatible con la plataforma Windows, pero debe estar preparado para migrar a GNU/Linux haciéndole pocas o ninguna modificación.
- ❑ **Hardware:** Debe funcionar sobre computadoras que tengan un buen procesamiento de cálculos, se recomienda que tengan como mínimo 512 GB de RAM y microprocesadores superiores a Pentium 4.
- ❑ **Software:** Las computadoras que utilizarán el software deben tener instalado: Windows 2000 NT, Windows XP Professional Service Pack 2.

2.7 Modelación del sistema.

En este epígrafe se identifica y describe el actor del sistema, también se modelan en términos de Casos de Uso del Sistema (CUS) los RF identificados anteriormente.

2.7.1 Actor del Sistema.

Un actor del sistema es un agente externo, es decir, es una persona o sistema que interactúan con el primero intercambiando información. En la siguiente tabla se describe el actor de la solución desarrollada.

Actor	Descripción
-------	-------------

Capítulo 2: Descripción de la Solución Propuesta.

Aplicación	La aplicación que utilice este componente se beneficiará con las funcionalidades que el mismo brinda para realizar el registro de características naturales con técnicas de RA.
------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tabla 2: Actor del Sistema.

2.7.2 Modelo de Casos de Uso del Sistema.

Los casos de uso son funcionalidades, o fragmentos de ellas, correspondientes a los procesos automatizados. En ellos se describe la secuencia determinada de eventos que realiza un actor en interacción con el sistema. En la (Figura 17) se muestra la representación UML de los CUS de la solución desarrollada.

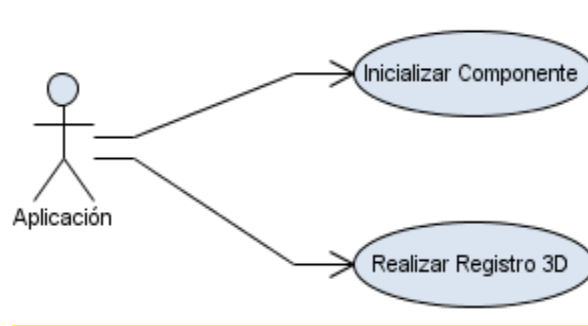


Figura 17: Diagrama de Caso de Uso del Sistema.

2.7.3 Descripción de los Casos de Uso del Sistema.

A continuación se proporciona una descripción resumida de los flujos de operaciones que poseen los CUS representados anteriormente, con el propósito de mostrar la interacción del actor con el sistema.

Nombre del Caso de Uso	Inicializar Componente
Actor	Aplicación.
	Leer los ficheros del proceso de entrenamiento e inicializar la captura del flujo de

Capítulo2: Descripción de la Solución Propuesta.

Propósito	video.
Resumen:	El caso de uso se inicia cuando la aplicación necesita cargar los datos de calibración de la(s) cámara(s), inicializar el mecanismo de detección, cargar el mapa de luminosidad e inicializar la captura del flujo de video de la(s) cámara(s).
Precondiciones	Los ficheros del proceso de entrenamiento deben estar creados. La(s) Cámara(s) deben estar conectadas(s) a la PC. El componente debe conocer la dirección de los ficheros.
Referencias	RF1, RF1.1 y RF1.2
Prioridad	Crítico.
Curso normal de eventos:	
Acción del Actor	Respuesta del Sistema
1 Solicita cargar los ficheros e inicializar el sistema.	1. Detecta la(s) cámara(s) conectada(s). 1.1 Cargar ficheros del proceso de calibración. 1.2 Cargar los archivos de configuración del mecanismo de detección de características naturales. 1.3 Cargar el mapa de luminosidad. 1.4 Retorna que fue exitosa la inicialización.
Curso alternativo de los eventos:	
	1.1 En caso de que los ficheros del proceso de calibración no existan, informa de que no puede inicializar el componente y retorna que fue fallido el proceso de lectura de los mismos. 1.2 En caso de que los ficheros de configuración del proceso de detección no existan, informa de que no puede inicializar el componente y retorna que fue fallido el proceso de lectura de los mismos.
Resultados:	1 Se inicializaron los parámetros de la(s) cámara(s). 2 Se inicializó la captura del flujo del video.

Capítulo 2: Descripción de la Solución Propuesta.

	3 Se inicializó el mecanismo de detección.
--	--------------------------------------------

Tabla 3: Descripción del Caso de Uso Inicializar Componente.

Nombre del Caso de Uso		Realizar Registro 3D
Actores	Aplicación.	
Propósito	Hacer el registro 3D en una escena de realidad aumentada.	
Resumen: El caso de uso de inicia cuando se necesita detectar de las características naturales en una imagen plana filmada por una cámara, actualizar la matriz de proyección de dicha cámara y devolver la matriz de transformación del objeto virtual a aumentar.		
Precondiciones	El proceso de inicialización del componente debe haberse ejecutado satisfactoriamente.	
Referencias	RF2, RF2.1, RF2.2 y RF2.3	
Prioridad	Crítico.	
Curso normal de eventos:		
Acción del Actor	Respuesta del Sistema	
1. Solicita detectar la imagen modelo en el flujo de video.	1. Ejecuta el mecanismo de detección. 1.1 Actualiza el mecanismo de detección.	
2. Solicita la matriz de proyección de la(s) cámara(s).	2. Actualiza la matriz de proyección de la(s) cámara(s). 2.1 Devuelve la matriz de proyección de la(s) cámara(s).	
3. Solicita la matriz de transformación del objeto a aumentar.	3. Calcula la matriz de transformación del objeto. 3.1 Devuelve la matriz de transformación del objeto.	
Curso alternativo de los eventos:		

Capítulo 2: Descripción de la Solución Propuesta.

	1.1 No ocurre el aumento de la escena 2.1 No ocurre el aumento de la escena 3.1 No ocurre el aumento de la escena
Resultados:	Aumento del objeto virtual en la escena real

Tabla 4: Descripción del Caso de Uso Realizar Registro 3D.

Conclusiones del capítulo.

Al realizar la descripción de la solución propuesta y exponer las herramientas, tecnologías y metodología seleccionadas para desarrollar dicha solución se dio el primer paso para construir un componente que permita realizar el registro 3D mediante el reconocimiento de características naturales usando la biblioteca de desarrollo BazAR. Al explicar el proceso de obtención de las entradas del componente y mencionar en qué consisten estas, se brinda la información necesaria para que dicho proceso se realice correctamente y así garantizar un registro 3D de mayor calidad. Identificados los requerimientos del sistema, tanto funcionales como no funcionales, se pudo obtener una modelación del componente, la cual contiene los aspectos principales en el diseño e implementación de IRA. Se mostraron las principales clases del dominio para lograr una mayor comprensión de los conceptos relacionados con la solución, se describió el actor que interactúa con el sistema, que en este caso es la aplicación. Se identificaron los requisitos funcionales y se modelaron en términos de CUS. Con la realización de este capítulo es posible comenzar a realizar el diseño e implementación del componente.

Capítulo 3: Diseño, Implementación y Resultados.

Introducción del capítulo.

En este capítulo se abordan los temas de la implementación del componente basados en el análisis de los casos de uso y los requisitos determinados en el capítulo anterior, también se describen las principales clases y componentes utilizados para construir la solución, además se expone el patrón de diseño aplicado en la solución. Se hace un análisis de los resultados del sistema, en cuanto al cumplimiento de los objetivos propuestos y resultados observados.

3.1 Diagrama de clases.

Los diagramas de clases muestran las diferentes entidades que componen un sistema y cómo se relacionan unas con otras. A continuación se muestra el diagrama que recoge las clases utilizadas para la implementación de IRA.

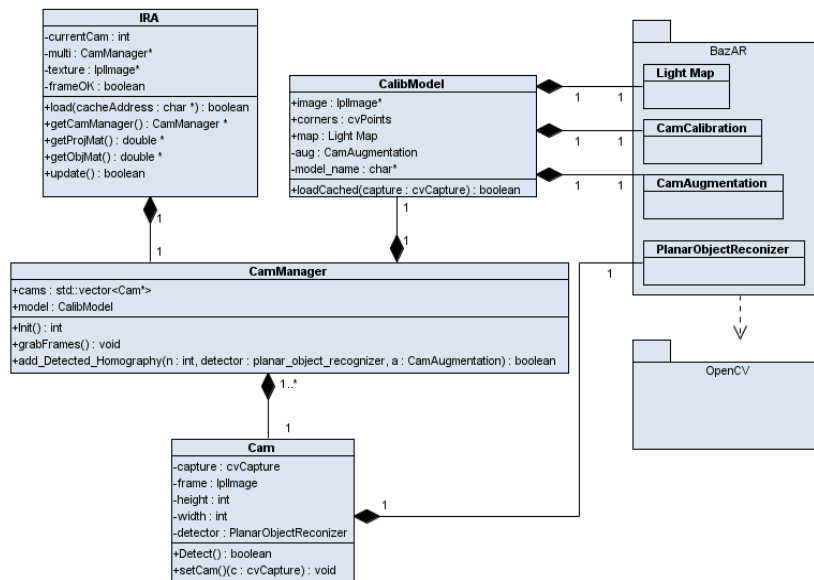


Figura 18: Diagrama de clases.

3.2 Descripción de las clases.

Una vez visto el diagrama de clases, a continuación se proporcionará una descripción de los atributos y métodos más significativos de las clases que se consideran importantes para la elaboración de la solución.

Nombre:	IRA
Tipo de clase:	Controladora
Resumen:	Representa la clase controladora principal.
Atributo:	Tipo:
multi	CamManager
texture	IpTexture
currentCam	int
frameOK	bool
Funciones	
Nombre:	IRA()
Descripción:	Constructor por defecto de la clase.
Nombre:	Load(char* cacheAddress)
Descripción:	Permite inicializar el componente cargando los ficheros.
Nombre:	Update()
Descripción:	Realiza el proceso de detección del patrón de calibración.
Nombre:	getTexture()
Descripción:	Devuelve el frame de video a visualizar en el flujo de video aumentado.
Nombre:	getProjectionMat()
Descripción:	Actualiza, convierte y devuelve la matriz de proyección de la cámara como una matriz de proyección de OpenGL.
Nombre:	getObjMat()
Descripción:	Calcula y devuelve la matriz de transformación del objeto virtual a aumentar.
Nombre:	getFrameOK()

Capítulo 3: Diseño, Implementación y Resultados.

Descripción	Devuelve el valor de la variable frameOK la cual representa con un valor positivo si en un frame se detectó el patrón de calibración.
Nombre:	getModelMat()
Descripción	Calcula y devuelve la matriz de transformación del modelo o patrón de calibración.
Nombre:	getCurrentCam()
Descripción	Devuelve el número de la cámara actual con que se está trabajando.

Tabla 5: Descripción de la clase IRA.

Nombre:	CamManager	
Tipo de clase:	Controladora	
Resumen:	Representa la clase controladora de las cámaras.	
Atributo:	Tipo:	
model	CalibModel	
cams	std::vector<Cam *>	
Funciones		
Nombre:	CamManager()	
Descripción:	Constructor por defecto de la clase.	
Nombre:	init()	
Descripción	Detecta las cámaras conectadas e inicializa el detector.	
Nombre:	grabFrames()	
Descripción	Captura los frames de cada cámara utilizada.	

Tabla 6: Descripción de la clase CamManager.

Nombre:	Cam
Tipo de clase:	Entidad
Resumen:	Contiene los atributos y métodos de una cámara.
Atributo:	Tipo:

Capítulo 3: Diseño, Implementación y Resultados.

Capture*	cvCapture
frame*, gray*	IplImage
width, height	int
detector	PlanarObjectRecognizer
Funciones	
Nombre:	Cam()
Descripción:	Constructor por defecto de la clase.
Nombre:	~Cam()
Descripción	Destructor de la clase.
Nombre:	setCam(cvCapture* pc)
Descripción	Cambia la cámara que captura el flujo de video.
Nombre:	Detect()
Descripción	Activa el mecanismo de detección.

Tabla 7: Descripción de la clase Cam.

Nombre:	CalibModel
Tipo de clase:	Entidad
Resumen:	Representa el patrón de calibración.
Atributo:	Tipo:
model_name	char*
Image*	IplImage
corners[4]	cvPoint
map	LightMap
aug	CamAugmentation
Funciones	
Nombre:	CalibModel(char* pmodel_file)
Descripción:	Constructor por defecto de la clase.
Nombre:	~CalibModel()

Descripción	Destructor de la clase.
Nombre:	LoadCached()
Descripción	Cambia la cámara que captura el flujo de video.

Tabla 8: Descripción de la clase CalibModel.

3.3 Diagrama de componentes.

Las clases utilizadas para la construcción del componente se materializan en los componentes que se muestran en el siguiente diagrama.

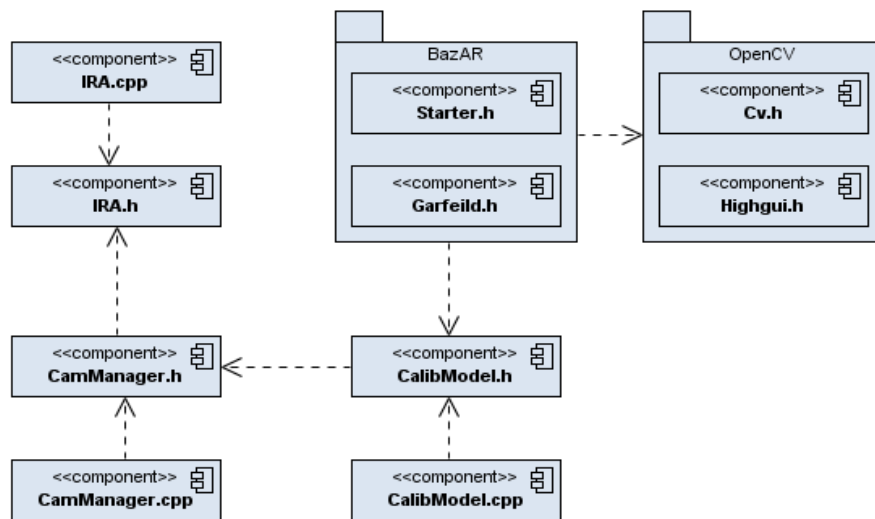


Figura 19: Diagrama de Componentes.

3.4 Diagramas de secuencia.

En este epígrafe se muestran los diagramas de secuencia correspondientes a los casos de uso descritos anteriormente en el Capítulo 2. Estos diagramas representan los eventos generados por el actor, su orden y los eventos internos del sistema. Para lograr claridad en los diagramas se detallaron los eventos de mayor importancia.

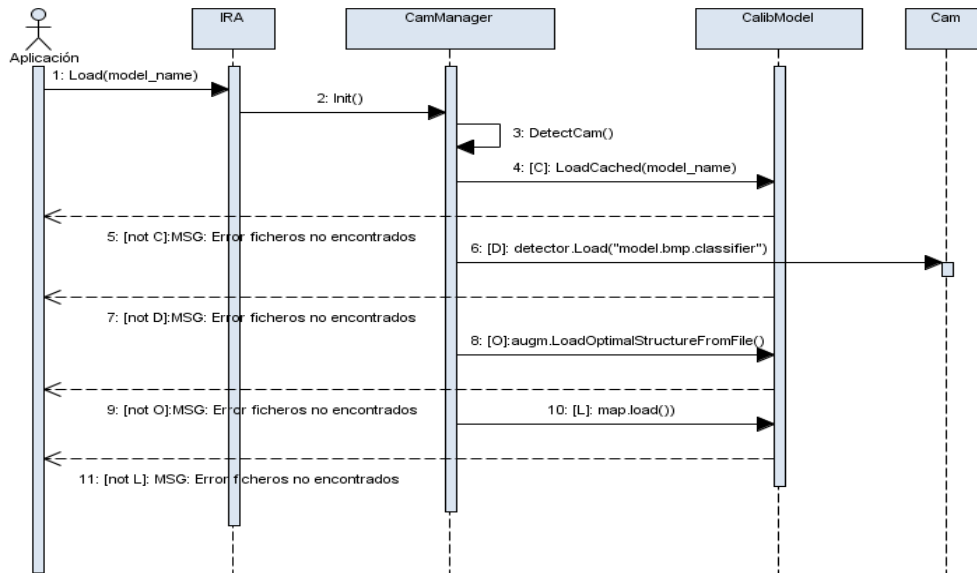


Figura 20: Diagrama de secuencia del caso de uso: Inicializar Componente.

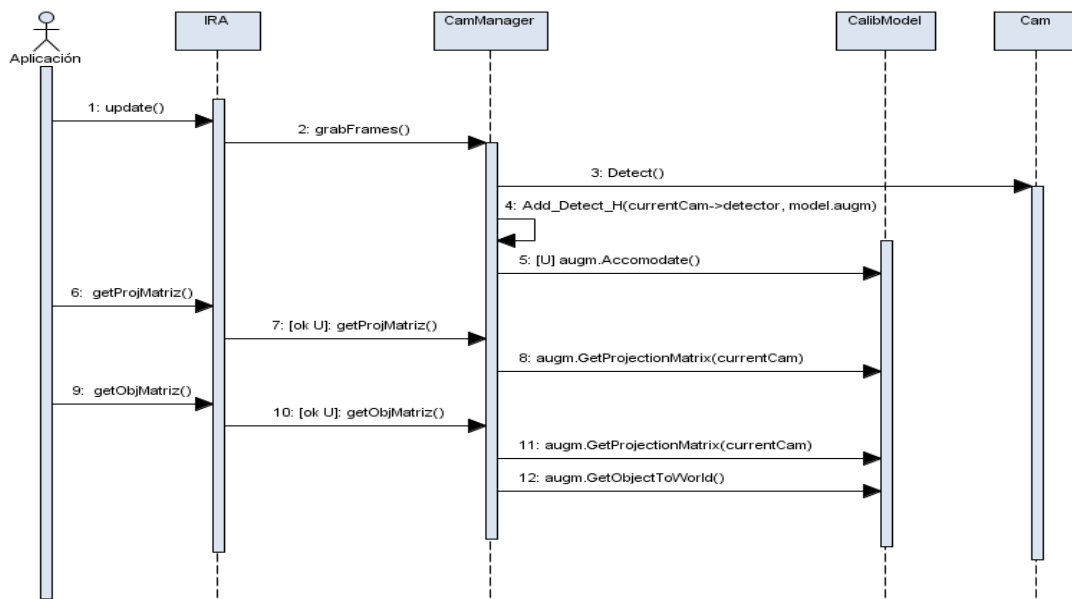


Figura 21: Diagrama de secuencia del caso de uso: Realizar Registro 3D.

3.5 Patrón de diseño utilizado.

El patrón estructural **Fachada** (*Facade*) define una interfaz que permite acceder solamente a las funcionalidades que esta posee de forma sencilla y clara por parte de los clientes ocultando la complejidad del sistema. Favorece un acoplamiento débil entre el sistema y sus clientes, ayuda a dividir un sistema en capas y reduce dependencias de compilación. Por las características anteriormente mencionadas se decidió aplicarlo en la solución.

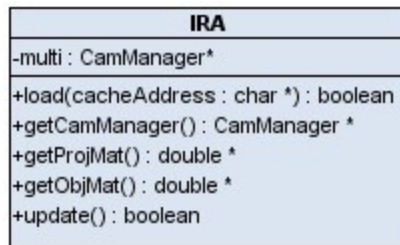


Figura 22: Diagrama del Patrón Fachada.

3.6 Análisis de los resultados obtenidos.

Se realizó una aplicación demostrativa (Figura 23) para validar la correcta funcionalidad del componente desarrollado. Los resultados obtenidos fueron los esperados, se observó que las propiedades idóneas que debe tener una imagen modelo para que sea reconocida satisfactoriamente por el mecanismo de detección de características naturales son:

- Líneas bien definidas, preferentemente de color negro.
- Variaciones de colores bien delimitadas.

También registro se ve condicionado por la orientación relativa entre la imagen modelo y la cámara. A mayor grado de inclinación, menos fiable será el reconocimiento.

Otra cuestión que influye con peso en el reconocimiento de la imagen modelo es la iluminación de la escena en que esta se encuentra, si anteriormente se exponía que Bazar trabajaba bien condiciones de

Capítulo 3: Diseño, Implementación y Resultados.

luz menos favorables, es válido aclarar que cuando la imagen plana se expone a cambios bruscos de iluminación la detección del mismo tiende a ser inestable.

Los dispositivos de captura utilizados para probar el funcionamiento de este componente lo constituyeron dos webcam que presentan poca resolución, una de estas cámaras cuenta con una resolución de 640x480 y la otra cuenta con 1024x768, se pudo observar que la resolución de las cámaras es un parámetro que interactúa de forma exponencial con respecto a la calidad del proceso de registro.

También se pudo comprobar que aproximadamente más de 2 metros de distancia entre la cámara y la imagen modelo el reconocimiento se torna casi nulo, en este resultado negativo influyen la resolución de la cámara, el tamaño del modelo y la distancia entre dicha cámara y la imagen modelo.

La aplicación demostrativa se probó en varias computadoras con distintas propiedades de hardware donde se comprobó que a mayor capacidad de procesamiento se realiza más preciso el registro 3D, también se pudo observar que si se cuenta con una tarjeta de video, esta aumenta considerablemente la rapidez del proceso de registro 3D. Aunque no se tiene el dato exacto de las propiedades mínimas que debe tener una computadora para ejecutar el sistema desarrollado se considera que con 512 Mega Byte de memoria RAM cualquier PC puede utilizar a IRA.



Figura 23: Aplicación demostrativa.

Conclusiones del capítulo.

En este capítulo se realizó la descripción de la implementación y el análisis de los resultados del componente. Se obtuvo el diagrama de clases del diseño, las cuales fueron descritas posteriormente con sus atributos y responsabilidades principales para lograr un mayor entendimiento. Se obtuvieron los diagramas de secuencia y los de componente correspondiente al flujo de trabajo de implementación, lo cual facilitó la codificación de la solución. Del análisis de los resultados obtenidos se llegó a la conclusión de que con los recursos que cuenta actualmente la línea de investigación Interacción 3D, el registro de marcadores cuadrados utilizando ARToolKit es de mejor calidad. Las observaciones al proceso de reconocimiento que provee IRA evidenciaron que se necesitan mejores recursos de captura para que este funcione con mayor precisión.

Conclusiones Generales.

Para la realización de esta tesis se hizo un estudio acerca del desarrollo y evolución de la Realidad Aumentada lo permitió a los autores conocer el estado de arte de esta fascinante tecnología. Se investigaron los dispositivos de entrada y salida, las técnicas de registro y bibliotecas de desarrollo más utilizadas en los Sistemas de Realidad Aumentada; lo cual permitió seleccionar las tecnologías más adecuadas para elaborar la solución.

Fue proveído, a la línea de investigación Interacción 3D del CEDIN, un componente de software libre, capaz de realizar reconocimiento de características naturales en una imagen plana de una escena real para aplicaciones de RA, cumpliendo así con el objetivo principal de la investigación. La solución permite lograr una mayor percepción de lo que representan los objetos virtuales en función de lo que se está reconociendo en la realidad y así lograr un mayor realismo en la escena aumentada. El componente provee una interfaz asequible y que pueda ser utilizada por cualquier aplicación de RA, ya que está centrado en un buen diseño Orientado a Objetos y brinda un bajo acoplamiento de las principales funcionalidades de la biblioteca BazAR.

Este componente logro romper con las limitaciones encontradas en la herramienta desarrollada en la línea de investigación Interacciones 3D utilizando ARToolKit, ya que permite una independencia de los marcadores cuadrados y resuelve el problema de la oclusión parcial de dichos marcadores.

Recomendaciones.

Una vez concluida la investigación, se recomienda que este resultado se utilice para la elaboración de un Sistema de Realidad Aumentada, debido a que cualquier aplicación de RA podría utilizarlo para realizar el registro 3D de características naturales. También se recomienda:

- Adaptar la solución a la plataforma libre GNU/Linux.
- Incorporar la funcionalidad de reconocer más de una imagen modelo al mismo tiempo en la escena real.
- Añadir la funcionalidad de representar los elementos virtuales utilizando la API DirectX o empleando algún motor de render como Ogre u OpenSceneGraph.
- Integrar a la herramienta de autor que se desarrolla en la línea de investigación.

Glosario de Términos.

- **2D:** Bidimensional.
- **3D:** Tridimensional.
- **Elemento virtual:** Cualquier imagen generada por computador que se genere en el dispositivo de visualización y se presente como parte de la Realidad Aumentada, esto incluye texto, figuras, imágenes 2D o modelos 3D.
- **Escena Aumentada:** Escena percibida por el usuario a través de un dispositivo de visualización donde se combina la escena real y el elemento virtual generado por el ordenador mediante técnicas de RA.
- **Imagen Modelo:** Es la imagen plana con la cual se entrena el mecanismo de detección de características naturales que posee BazAR para su posterior registro.
- **Escena Real:** Escena percibida por el usuario a través de un dispositivo de visualización.
- **Reconocimiento:** Acción en el cual se detecta la imagen modelo en el flujo de video en el proceso de registro.
- **Herramienta Case:** Ingeniería de sistemas asistida por ordenador (Computer-Aided Systems Engineering - CASE) es la aplicación de tecnología informática a las actividades, las técnicas y las metodologías propias de desarrollo de sistemas. Su objetivo es automatizar o apoyar una o más fases del ciclo de vida del desarrollo de sistemas.

- **Homografía:** Es un concepto en matemático que la ciencia de geometría. Se define como relación entre dos figuras, tales que cualquier punto dado en una figura corresponde a un y solamente un punto en el otro, y viceversa.
- **Visión por Computador:** La visión por computador es una rama de la inteligencia artificial que desarrolla tanto la teoría como la tecnología necesaria para emular la percepción visual humana. Sus objetivos típicos incluyen: la detección, segmentación, localización y reconocimiento de ciertos objetos en imágenes. También se encarga del registro de diferentes imágenes en una misma escena, seguimiento de un objeto en una secuencia de imágenes, mapeo de una escena para generar un modelo tridimensional de la escena, estimación de las posturas tridimensionales de humanos, búsqueda de imágenes digitales por su contenido, entre otros.

Bibliografía Citada.

- **[Azuma, 1993]** R. Azuma. (1993). Tracking requirements for augmented reality. **[En línea] [Citado el: 18 de Enero de 2010.]**<http://portal.acm.org/citation.cfm?doid=159544.159581>
- **[Azuma, 1997]** Azuma, Ronald. (1997). A Survey of Augmented Reality. (Vol. 6). **[En línea] [Citado el: 18 de Enero de 2010.]** <http://portal.acm.org/citation.cfm?id=159544.159581>
- **[Bimber, 2005]** Bimber, O. y R. Raskar (2005d). Spatial Augmented Reality: Merging Real and Virtual Worlds. **En línea] [Citado el: 8 de febrero de 2010.]** <http://www.merl.com/reports/docs/TR2006-105.pdf>
- **[Borro, 2005]** Borro, Diego, Manuel Lardizábal (2005). Introducción a la Realidad Aumentada: Aplicaciones en la Industria.
- **[Clemens, 1998]** Clemens Szyperski, “Component Software Component Software Beyond Object – Oriented Programming”. P 20 -22. 1998.
- **[Kaufmann, 2003]** Kaufmann, H. y D. Schmalstieg (2003). Mathematics and geometry education with collaborative augmented reality. **[En línea] [Citado el: 18 de febrero de 2010.]** <http://doi.acm.org/10.1145/1242073.1242086>
- **[Lerma, 2002]** Lerma, J. L. (2002).Fotogrametría Moderna: Analítica y Digital. Universidad Politécnica de Valencia. (2002). págs. 53-55. **[En línea] [Citado el: 28 de febrero de 2010.]** <http://jllerma.webs.upv.es/fotomo.htm>
- **[Livingston, 2005]** Livingston, M. A. (2005). Evaluating Human Factors in Augmented Reality Systems. **[En línea] [Citado el: 1 de Marzo de 2010.]** <http://doi.ieeecomputersociety.org/10.1109/MCG.2005.130>

Bibliografía Consultada.

- **[Azuma, 1993]** R. Azuma. (1993). Tracking requirements for augmented reality. **[En línea] [Citado el: 18 de Enero de 2010.]**<http://portal.acm.org/citation.cfm?doid=159544.159581>
- **[Azuma, 1997]** Azuma, Ronald. (1997). A Survey of Augmented Reality. (Vol. 6). **[En línea] [Citado el: 18 de Enero de 2010.]** <http://portal.acm.org/citation.cfm?id=159544.159581>
- **[Azuma, 2001]** R. Azuma, Y. B. (2001). Recent Advances in Augmented Reality.
- **[Avery, 2005]** Avery, B., B. H. Thomas, J. Velikovsky y W. Piekarski (2005). Outdoor augmented reality gaming on five dollars a day. Australian Computer Society, Inc.
- **(Barfield, 1995)** Barfield, W., C. Rosenberg y W. A. Lotens (1995). Augmented Reality Displays. Virtual Environments and Advanced Interface Design. Oxford University Press, USA.
- **[Barron, 1995]** Barron, S. B. (1995).The computation of optical flow. s.l. : ACM Computing Surveys(CSUR)., (1995). Vol. (Vol. 27).
- **[Billinghurst, 2001]** Billinghurst, M., H. Kato y I. Poupyrev (2001a).The MagicBook - Moving Seamlessly between Reality and Virtuality.
- **[Billinghurst, 2002]** Billinghurst, M. y H. Kato (2002). Collaborative Augmented Reality. Communications of the ACM.
- **[Bimber, 2002]** O. Bimber and B. Fröhlich(2002). Occlusion shadows: using projected light to generate realistic occlusion effects for view-dependent optical see-through displays. SMAR '02.

- **[Bimber, 2005]** Bimber, O. y R. Raskar (2005d). Spatial Augmented Reality: Merging Real and Virtual Worlds.
- **[Borro, 2005]** Borro, Diego, Manuel Lardizábal (2005). Introducción a la Realidad Aumentada: Aplicaciones en la Industria.
- **[Broll, 2004]** Broll, W., I. Lindt, J. Ohlenburg, M. Wittkämper, C. Yuan, T. Novotny, A. F. g. Schiecky, C. Mottramyy A. Strothmann (2004). ARTHUR: A Collaborative Augmented Environment for Architectural Design and Urban Planning.
- **[Cheok, 2003]** Cheok, A. D., y otros. (2003). Human Pacman: a sensing-based mobile entertainment system with ubiquitous computing and tangible interaction. Network and System Support for Games. Redwood City, California, ACM New: s.n., (2003).
- **[Cheok, 2004]** Cheok, A. D., y otros. (2004). "Human Pacman: a mobile, wide-area entertainment system based on physical, social, and ubiquitous computing." Personal and Ubiquitous Computing. (2004). págs. 71 - 81. Vol. 8.
- **[Comport, 2006]** A.I. Comport, E. M. (2006). Real-time markerless tracking for augmented reality: the virtual visual servoing framework.
- **[Cornelis,2001]** Cornelis, K. M. (2001). Tracking Based Structure and Motion Recovery for Augmented Video Productions. Banff, Canada, ACM. : ACM symposium on virtual reality software and technology (VRST'01), (2001).
- **[Cooper, 2004]** Cooper, N., A. Keatley, M. Dahlquist, S. Mann, H. Slay, J. Zucco, R. Smith y B. H. Thomas (2004). Augmented Reality Chinese Checkers.
- **[CVLab, 2010]** CVLAB - Computer Vision Laboratory. (2010). bazAR Documentation. <http://cvlab.epfl.ch/software/bazar/html/index.html>.

- **[Dangelmaier, 2005]** Dangelmaier, W., M. Fischer, J. Gausemeier, M. Grafe, C. Matysczok y B. Mueck (2005). Virtual and augmented reality support for discrete manufacturing system simulation.
- **[De la Escalera Hueso, A ,2001]** De la Escalera Hueso, A. (2001). Visión por computador. Fundamentos y métodos. Madrid: s.n., (2001).
- **[Flintham, 2003]** Flintham, M., R. Anastasi, S. Benford, T. Hemmings, A. Crabtree, C. Greenhalgh, T. Rodden, N. Tandavanitj, M. Adams y J. Row-Farr (2003). Where On-Line Meets On-The-Streets:
- **[Florins, 2005]** Florins, M., D. G. Trevisan y J. Vanderdonckt (2005). The continuity property in mixed reality and multiplatform systems: a comparative study.
- **[Freeman, 2006]** Freeman, R. y A. Steed (2006). Interactive Modelling and Tracking for Mixed and Augmented Reality.
- **[Garcia,2004]** Garcia, Oscar y A. G. (2004). Introducción a la Programación Gráfica con OpenGL. Escuela Técnica Superior de Ingeniería Electrónica e Informática La Salle. (2004).
- **[Guevara ,2008]** Mileydi Moreno y Ernesto de la Cruz Guevara (2008) Localización de objetos virtuales en el mundo real con técnicas de Realidad Aumentada.
- **[HITL, 2007]** Human. (2007). Human Interface Technology Laboratory. "ARToolKit". **[En línea]** (2007). [Citado el: 19 de 4 de 2010.] <http://www.hitl.washington.edu/artoolkit/>.
- **[HITLab, 2010]** Human Interface Technology Laboratory (2010). **[En línea]** <http://www.hitl.washington.edu/home/> Cede en Washington
<http://www.hitlabnz.org/wiki/Home> Cede en Nueva Zelanda
<http://www.hitlab.utas.edu.au/wiki/Home> Cede Australia.
- **[Isern,2003]** Isern González, Josep. 2003. Estudio experimental de métodos de calibración y autocalibración de cámaras. 2003.

- **[Jähne, 1999]** Jähne, B., H. Haussecker y P. Geissler, Eds. (1999). Handbook of Computer Vision and Applications. Sensors and Imaging.
- **[Janin, 1993]** Janin, A. L., D. W. Mizell y T. P. Caudell (1993). Calibration of head-mounted displays for augmented reality applications. IEEE Virtual Reality Annual Int. Symp., Seattle, WA.
- **[Jacobs, 2006]** Jacobs, K. y C. Loscos (2006). Classification of Illumination Methods for Mixed Reality.
- **[Juan, 2005]** Juan, M. C., M. Alcañiz, C. Monserrat, C. Botella, R. M. Baños y B. Guerrero (2005). Using augmented reality to treat phobias. IEEE Computer Graphics and Applications.
- **[Jurie,1998]** Jurie, F. (1998).Tracking objects with a recognition algorithm. (1998). Vols. (Vols. 3-4).
- **[Kaufmann, 2002]** Kaufmann, H. (2002). Construct3D: An Augmented Reality Application for Mathematics and Geometry Education. ACE'02.
- **[Lam, 2006]** Lam, A. H. T., K. C. H. Chow, E. H. H. Yau y M. R. Lyu (2006). ART: Augmented Reality Table for Interactive Trading Card Game.
- **[Lee, 2006]** Lee, S. P., A. D. Cheok, T. K. S. James, G. P. L. Debra, C. W. Jie, W. Chuang y F. Farbiz (2006b). A mobile pet wearable computer and mixed reality system for human–poultry interaction through the internet.
- **[Lepetit, 2004a]** Lepetit, V. y P. Fua (2004a). Towards Recognizing Feature Points using Classification Trees. Lausanne, Switzerland, Swiss Federal Institute of Technology (EPFL).
- **[Lepetit, 2004b]** Lepetit, V., J. Pilet y P. Fua (2004b). Point Matching as a Classification Problem for Fast and Robust Object Pose Estimation. Computer Vision and Pattern Recognition. Lausanne, Switzerland, Swiss Federal Institute of Technology (EPFL).

- **[Lepetit, 2005]** Lepetit, V., P. Lagger y P. Fua (2005). Randomized Trees for Real-Time Keypoint Recognition. Computer Vision and Pattern Recognition. Lausanne, Switzerland, Swiss Federal Institute of Technology (EPFL). <http://cvlab.epfl.ch/publications/publications/2005/LepetitLF05.pdf>
- **[Lepetit, 2007]** Fua, Pascal and Vincent Lepetit (2007). Vision Based 3D Tracking and Pose Estimation for Mixed Reality.
- **[Livingston, 2005]** Livingston, M. A. (2005). Evaluating Human Factors in Augmented Reality Systems.
- **[Lowe,1991]** Lowe, D. (1991).Fitting parameterized three-dimensional models to images. (1991). Vol. (Vol. 13).
- **[Marchand, 2002]** Marchand, r. y Chaumette, F. (2002).Virtual Visual Servoing: a framework for real-time augmented reality. (2002). Vol. 21.
- **[Marchand, 2006]** M. Pressigout and É. Marchand (2006). Hybrid tracking algorithms for planar and non-planar structures subject to illumination changes.
- **[Milgram ,1994]**Milgram, P. y F. Kishino (1994). A taxonomy of mixed reality visual displays. IEICE Transactions on Networked Reality
- **[Milgram, 1997]** Milgram, P. y D. Drascic (1997). Perceptual effects in aligning virtual and real objects in augmented reality displays.
- **[NRC, 1994]** National Research Council Staff (1994). Virtual Reality: Scientific and Technological Challenges, Washington, DC, USA: National Academies Press.
- **[Park, 2006]** Park, J.-S., T. Kim y J.-H. Yoon (2006). AR Table Tennis: A Video-Based Augmented Reality Sports Game.

- **[Pascal,2007]** Pascal, Fua y Lepetit, Vincent. (2007). Vision Based 3D Tracking and Pose Estimation for Mixed Reality. en: Emerging Technologies of Augmented Reality, Interfaces and Design. s.l. : HALLER, M.et al, Idea Group Publishing., (2007).
- **[Pilet, 2006]** Pilet, J., A. Geiger, P. Lagger, V. Lepetit y P. Fua (2006). An All-In-One Solution to Geometric and Photometric Calibration. International Symposium on Mixed and Augmented Reality.
- **[Pressigout, 2006]** M. Pressigout and É. Marchand(2006). Hybrid tracking algorithms for planar and non-planar structures subject to illumination changes.
- **[Prince,2002]** Prince, S. J. (2002)."Augmented Reality Camera Tracking with Homographies." s.l.: IEEE Computer Graphics and Applications: (2002). Págs. 39 - 45.
- **[Reitmayr, 2007]** Reitmayr, G. y Drummond, T. (2007).Initialisation for Visual Tracking in Urban Environments. s.l.: In ISMAR '07, (2007). Págs. 161-172.
- **[Ribo, 2001]** Ribo, M., A. Pinz y A. L. Fuhrmann (2001). A new Optical Tracking System for Virtual and Augmented Reality Applications.
- **[Romero,2008]** Romero, Maydelis y Galván, Julio A. (2008).Aplicación de Shaders de Relieve a objetos 3D en entornos de Realidad Aumentada. (2008).
- **[Rolland, 2001]** J. P. Rolland, L. Davis and Y. Baillot. (2001), A survey of tracking technology for virtual environments.
- **[Sandor, 2005]** Sandor, C. y G. Klinker (2005). A rapid prototyping software infrastructure for user interfaces in ubiquitous augmented reality.
- **[Schnädelbach, 2002]** Schnädelbach, H., B. Koleva, M. Fintham, M. Fraser, S. Izadi, P. Chandler, M. Foster, S. Benford, C. Greenhalgh y T. Rodden (2002). The Auguroscope: a mixed reality interface for outdoors.

- **[SSTT, 2009]** Simplified Spatial Target Tracker (2009) .Tecnoture **[En línea]** **[Citado el: 14 de 4 de 2010.]** <http://technotecture.com/>
- **[Teichrieb, 2007]** Teichrieb, V., S. G. Neto, T. Farias, J. M. Teixeir, J. P. Lima, G. Almeida y J. Kelner (2007).Augmented Ambient: An Interactive Mobility Scenario.
- **[Wacker,2005]** Wacker, F. K., y otros. (2005)."MR image-guided needle biopsies with a combination of augmented reality and MRI: A pilot study in phantoms and animals. s.l. : " International Congress Series, (2005).
- **[Zhou, 2008]** Feng Zhou, Henry Been-Lirn Duh, Mark Billinghurst (2008).Trends in Augmented Experiences with Mobile Mixed Reality Games.

Anexos.

ANEXO 1: Citas de todos los simposios de Realidad Aumentada ISMAR.

- 1ro Noviembre, 1998, San Francisco, CA (USA). IEEE International Workshop on Augmented Reality (IWAR).
- 9-11 Marzo, 1999, Yokohama (Japón). International Symposium on Mixed Reality (ISMR).
- 5-6 Octubre, 2000, Munich (Alemania). IEEE, ACM, and Eurographics International Symposium on Augmented Reality (ISAR).
- 29-30 Octubre, 2001, New York (USA). IEEE, International Symposium on Augmented Reality (ISAR 2001).
- 30 Septiembre – 1 Octubre, 2002, Darmstadt (Alemania). International Symposium on Mixed and Augmented Reality (ISMAR). Este congreso unió los dos congresos de ISMR (International Symposium on Mixed Reality) e ISAR.
- 7-10 Octubre 2003, Tokyo (Japan). IEEE / ACM International Symposium on Mixed and Augmented Reality (ISMAR 2003).
- 2-5 Noviembre 2004, Arlington (serca de Washington, DC, USA). IEEE / ACM International Symposium on Mixed and Augmented Reality (ISMAR 2004).
- 5-8 Octubre 2005, Vienna (Austria) IEEE / ACM International Symposium on Mixed and Augmented Reality (ISMAR 2005).
- 22-25 Octubre 2006, University of California (Santa Barbara, CA, USA). IEEE / ACM International Symposium on Mixed and Augmented Reality (ISMAR 2006).

- 13 -16 Noviembre 2007, Nara-Ken New Public Hall (Nara, Japan) IEEE / ACM International Symposium on Mixed and Augmented Reality (ISMAR 2007).
- 15 -18 Septiembre 2008, Cambridge (Inglaterra). IEEE / ACM International Symposium on Mixed and Augmented Reality (ISMAR 2008).

19 – 22 Octubre 2009, Orlando (Florida, USA). IEEE / ACM International Symposium on Mixed and Augmented Reality (ISMAR 2009).

ANEXO 2: Imágenes de la aplicación demostrativa.

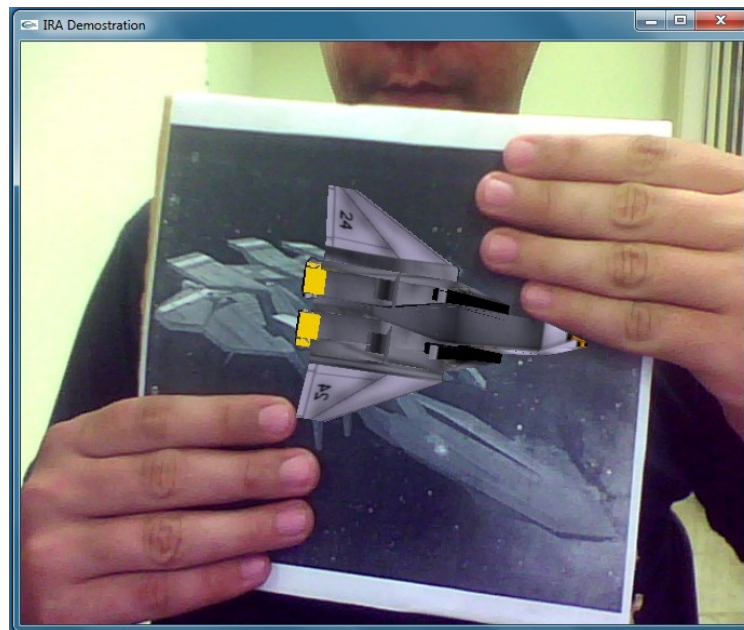


Figura 24: Parcial oclusión de la imagen modelo.

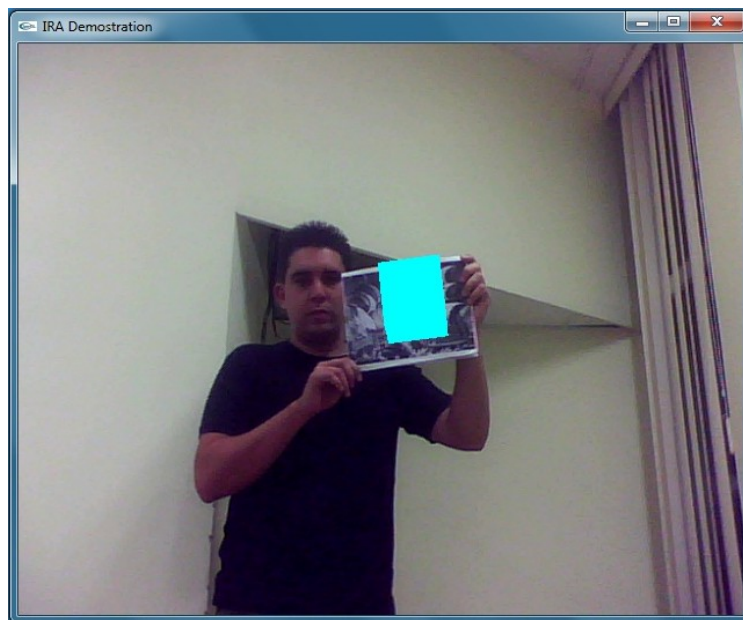


Figura 25 : Máxima distancia con respecto a la cámara para realizar el registro: 2m aproximadamente.



Figura 26 : Grundy Campeón