



Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Escuela Técnica Superior de Ingeniería Informática

PROYECTO FINAL DE CARRERA

Realizado por:

Domingo Sánchez Blázquez

Dirigido por:

M. Carmen Juan Lizandra

RESUMEN.....	5
CAPITULO 1. MOTIVACIÓN Y OBJETIVOS DEL PROYECTO.....	7
1.1. Introducción.....	7
1.2 Objetivos.....	9
1.3 Estructura del proyecto.....	9
CAPÍTULO 2. INTRODUCCIÓN A REALIDAD AUMENTADA Y APLICACIONES PARA APRENDIZAJE, EN ESPECIAL DE QUÍMICA.....	11
2.1 Realidad Aumentada, conceptos.....	11
2.2 Tipos de Realidad Aumentada.....	12
2.3 Campos de aplicación.....	14
2.3.1 Medicina.....	14
2.3.2 Entretenimiento.....	15
2.3.3 Psicología.....	17
2.3.4 Área Militar.....	18
2.3.5 Diseño industrial.....	20
2.3.6 Robótica y telerrobótica.....	21
2.3.7 Fabricación, mantenimiento y reparación.....	22
2.3.8 Arquitectura y construcción.....	24
2.3.9 Diseño de consumo.....	25
2.3.10 Enseñanza.....	27
CAPÍTULO 3: Programas de Modelado Molecular.....	31
3.1 Introducción.....	31
3.2 Exposición de las aplicaciones.....	31
Abalone.....	31
AutoDock.....	34
Amira.....	37
Ascalaph Graphics.....	42
Ascalaph Designer.....	44
Amber / Amber Tools.....	47
Avogadro.....	49
BOSS.....	52
Maestro.....	55
BALLView.....	58
CAPITULO 4: Uso de ARToolKit e inclusión de moléculas.....	60
4.1 Introducción.....	60
4.2 Análisis de las herramientas a utilizar.....	60
4.2.1 ARToolKitVRML.....	60
4.2.1 Microsoft Visual Studio 2005.....	61
4.2.3 BALLView.....	62
4.2.4 DirectX, DirectShow.....	62
4.3 Trabajando con Microsoft Visual Studio 2005, BALLView y ARToolKit.....	64
4.3.1 La cámara.....	64
4.3.2 Creación de Moléculas.....	64
4.3.3 Búsqueda de marcas: Cálculo de la matriz de transformación.....	65
4.3.4 Calibración de la cámara.....	68
4.3.4.1 Calibrando por ARToolKit.....	68
4.3.4.2 Calibrando mediante programa externo.....	69
4.3.5 Creación de patrones.....	70
4.3.6 Creación de la Aplicación de Realidad Aumentada.....	72
4.3.6.1 Main.....	73
4.3.6.2 Init.....	73
4.3.6.3 Mainloop.....	74

2.3.6.4 Draw.....	76
2.3.6.5 Cleanup.....	77
4.3.7 Requisitos mínimos.....	79
CAPITULO 5 Conclusiones y trabajos futuros.....	81
5.1 Conclusiones.....	81
5.2 Trabajos futuros.....	81
Capitulo 6 Bibliografía.....	82
Glosario.....	86

RESUMEN

El objetivo principal de este proyecto es desarrollar y evaluar un sistema de RA (Realidad Aumentada) enfocado a la química, bioquímica, farmacia, y ámbito molecular para facilitar el aprendizaje de las distintas proteínas o elementos químicos y orgánicos por parte de estudiantes tanto de educación media como educación superior.

Lo que se pretende es facilitar el estudio de la composición, estructura y propiedades de la materia, como los cambios que esta experimenta durante las reacciones químicas y su relación con la energía.

Las disciplinas de la química han sido agrupadas por la clase de materia bajo estudio o el tipo de estudio realizado. Entre estas se tienen la química inorgánica que estudia la materia inorgánica; la química orgánica que trata con la materia orgánica; la bioquímica el estudio de sustancias en organismos biológicos; la físico-química comprende los aspectos energéticos de sistemas químicos a escalas macroscópicas, moleculares y atómicas; la química analítica que analiza muestras de materia tratando de entender su composición y estructura. Otras ramas de la química han emergido en tiempos recientes, por ejemplo, la neuroquímica que estudia los aspectos químicos del cerebro.

La ubicuidad de la química en las ciencias naturales hace que sea considerada como una de las ciencias básicas. La química es de gran importancia en muchos campos del conocimiento, como la ciencia de materiales, la biología, la farmacia, la medicina, la geología, la ingeniería y la astronomía, entre otros.

Los procesos naturales estudiados por la química involucran partículas fundamentales (electrones, protones y neutrones), partículas compuestas (núcleos atómicos, átomos y moléculas) o estructuras microscópicas como cristales y superficies.

Desde el punto de vista microscópico, las partículas involucradas en una reacción química pueden considerarse como un sistema cerrado que intercambia energía con su entorno. En procesos exotérmicos, el sistema libera energía a su entorno, mientras que un proceso endotérmico solamente puede ocurrir cuando el entorno aporta energía al sistema que reacciona. En la gran mayoría de las reacciones químicas hay flujo de energía entre el sistema y su campo de influencia, por lo cual podemos extender la definición de reacción química e involucrar la energía cinética (calor) como un reactivo o producto.

Es común que entre las comunidades académicas de químicos la química analítica no sea considerada entre las subdisciplinas principales de la química y sea vista más como parte de la tecnología química. Otro aspecto notable en esta clasificación es que la química inorgánica sea definida como "química no orgánica".

La gran importancia de los sistemas biológicos hace que en nuestros días gran parte del trabajo en química sea de naturaleza bioquímica. Entre los problemas más interesantes se encuentran, por ejemplo, el estudio del desdoblamiento de las proteínas y la relación entre secuencia, estructura y función de proteínas.

Si hay una partícula importante y representativa en la química es el electrón.

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Uno de los mayores logros de la química es haber llegado al entendimiento de la relación entre reactividad química y distribución electrónica de átomos, moléculas o sólidos. Los químicos han tomado los principios de la mecánica cuántica y sus soluciones fundamentales para sistemas de pocos electrones y han hecho aproximaciones matemáticas para sistemas más complejos. La idea de orbital atómico y molecular es una forma sistemática en la cual la formación de enlaces es entendible y es la sofisticación de los modelos iniciales de puntos de Lewis. La naturaleza cuántica del electrón hace que la formación de enlaces sea entendible físicamente y no se recurra a creencias como las que los químicos utilizaron antes de la aparición de la mecánica cuántica. Aun así, se obtuvo gran entendimiento a partir de la idea de puntos de Lewis.

La RA consiste en fusionar imagen real con objetos virtuales. Es decir, el usuario está viendo el mundo real y superpuesto visualiza imágenes virtuales. Permitiendo así añadir información a la escena que el usuario no podría observar con sus propios sentidos.

En este proyecto los dispositivos que se pretende utilizar son dispositivos de uso y adquisición común, accesibles a cualquier entidad académica, ordenador portátil o de sobremesa y una webcam que frecuentemente ya vienen integradas. Para desarrollar aplicaciones de RA en PCs existen herramientas libres como por ejemplo: ARToolKit [Kato99], OsgART, ImageTclAR, MXRToolkit, etc. Como resultado, con estas herramientas se pueden obtener aplicaciones de alto nivel para el desarrollo rápido sin necesidad de conocimientos dedicados a programación y dichas herramientas. En este caso, para desarrollar el presente proyecto se utilizará una versión ampliada de ARToolKit, ARToolKit con soporte VRML (ARToolKitVRML 2.65).

Para la creación de moléculas, proteínas o lo que se quiera mostrar utilizaremos un programa de edición molecular. Actualmente hay un gran abanico de aplicaciones de ámbito químico/molecular dedicados a la creación de fármacos, proteínas o estudio de moléculas orgánicas, inorgánicas, viendo su comportamiento e interacción con un medio acuoso o intermolecular. Como se puede imaginar el ámbito de estas aplicaciones va desde el simple visor a el más específico desarrollo para su estudio energético. Por ese motivo se dedica un capítulo al análisis y comparación de un conjunto de los programas más completos y genéricos que se pueden encontrar.

Este proyecto desarrollará una aplicación de RA. Este tipo de aplicaciones se puede utilizar para enseñar estructuras básicas, explicar las Fuerzas de Campo, de Amber o Dock. También sirve como un método eficaz de examen, entrenamiento ante el estudio de un conjunto de moléculas; el usuario final no tiene dudas de cómo es la molécula que a de identificar, porque la vé en 3D, el examinador tiene la seguridad de que el alumno no sabe que se le va a plantear, pues el dispondría las plantillas.

CAPITULO 1. MOTIVACIÓN Y OBJETIVOS DEL PROYECTO

1.1. Introducción

«La química implica estudiar las propiedades y el comportamiento de la materia. La materia es el material físico del universo; es cualquier cosa que tiene masa y ocupa espacio. Incontables experimentos han demostrado que la enorme variedad de la materia en nuestro mundo se debe a combinaciones de apenas poco más de un ciento de sustancias muy básicas o elementales, llamadas elementos. La química también proporciona antecedentes para entender las propiedades de la materia en términos de átomos, los bloques de construcción casi infinitesimalmente pequeños de la materia. Los átomos se pueden combinar para formar moléculas, en las que dos o más átomos se unen en estructuras específicas.»[Theodore03]

Al ser la química la ciencia que trata de la constitución, no hay ninguna rama de la ciencia que posea una extensión tan amplia e incluso, un examen cuidadoso del contenido de todas ellas revela que la química tiene una relación estrecha con cualquier ciencia particular. Así, la biología incluye el estudio de las transformaciones químicas que ocurren en las minúsculas células integrantes del organismo vivo durante los procesos de digestión y crecimiento; la geología constituye un prodigioso lienzo natural en el que se muestran los cambios químicos verificados en las rocas durante las diversas eras geológicas; la física se ocupa de las distintas formas de energía, pero los efectos de la energía sobre la materia son casi siempre de naturaleza química; y la astronomía se apoya en la química al buscar información acerca de la estructura y constitución de los astros. En la figura 1 se puede ver el alcance de la química en nuestro mundo.

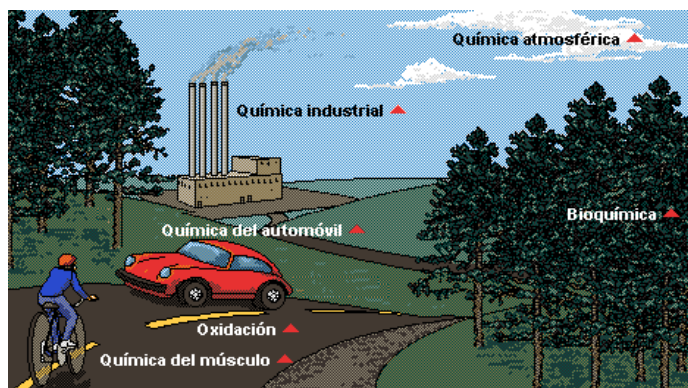


Figura 1: Visión global de la química en nuestro entorno

La primera finalidad de la química es averiguar cómo los materiales pueden identificarse o distinguirse, por las sustancias de que están formados los objetos. La química no está interesada en las *propiedades accidentales* o atributos tales como el tamaño y la forma sino en las **propiedades específicas** de la clase de materia que pueden reconocerse en cualquier cuerpo formado por ella. Así, por ejemplo, una moneda de plata, una copa de plata y un electrodo de plata difieren en tamaño, forma, utilidad e incluso aspecto estético, pero desde el punto de vista químico son

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

esencialmente lo mismo al estar formados por el mismo metal, la plata.

Al contemplar nuestro mundo material se observan incesantes cambios en los que unos cuerpos desaparecen y se transforman en otros distintos. Estos cambios o transformaciones se conocen como **reacciones químicas**. Muchas veces no nos interesamos en las nuevas sustancias que se originan en las reacciones químicas sino en la energía producida en las mismas. Así, quemamos carbón en el hogar, no para obtener las cenizas, ni para utilizar el dióxido de carbono, sino para aprovechar el calor desprendido en su combustión. El estudio de las reacciones químicas, es la segunda finalidad de la química.

El objeto final de la química es el de la constitución de la materia, puesto que este conocimiento permite identificar y diferenciar las sustancias, comprender sus propiedades y establecer su comportamiento frente a otras clases de sustancias o bajo la acción de cualquier forma de energía.

La química no se limita al estudio de las sustancias, se extiende su finalidad al descubrir incesantemente nuevas sustancias que no se encuentran en la naturaleza y cuyas propiedades y aplicaciones las hacen en muchísimos casos de incalculable valor. [SanFrutos03]

La química permite obtener un entendimiento importante de nuestro mundo y su funcionamiento. Se trata de una ciencia eminentemente práctica que tiene una influencia enorme sobre nuestra vida diaria. De hecho, la química está en el centro de muchas cuestiones que preocupan a casi todo mundo: el mejoramiento de la atención médica, la conservación de los recursos naturales, la protección del entorno, la satisfacción de nuestras necesidades diarias en cuanto a alimento, vestido y albergue. Con la ayuda de la química, hemos descubierto sustancias farmacéuticas que fortalecen nuestra salud y prolongan nuestra vida. Hemos aumentado la producción de alimentos mediante el desarrollo de fertilizantes y plaguicidas. Hemos creado plásticos y otros materiales que se usan en casi todas las facetas de nuestra vida. Desafortunadamente, algunos productos químicos también pueden dañar nuestra salud o el entorno. Nos conviene, como ciudadanos educados y consumidores, entender los profundos efectos, tanto positivos como negativos, que las sustancias químicas tienen sobre nuestra vida, y encontrar un equilibrio sobre su uso.

No obstante la química es un componente obligatorio en muchas carreras como biología, ingeniería, agrónomos, geología, o muchas otras. [Theodore03]

Por otra parte, la enseñanza es un campo que siempre ha preocupado a la sociedad, intentando mejorar su nivel educativo y facilitando la misma para que sea accesible a todo el mundo. Una manera de mejorar la enseñanza es la integración de nuevas tecnologías, facilitando la comprensión de las materias. Dentro de estas tecnologías encontramos la RA. Una ventaja educativa, es la disponer de libros de texto con plantillas para RA integradas, permitiendo al alumno ampliar el aprendizaje de la estructura estudiada. Además se puede mostrar animaciones o interactuar con el objeto tridimensional. Quizá una de las aplicaciones más conocidas de RA en la educación sea el proyecto Magic Book del grupo activo HIT de Nueva Zelanda. El alumno lee un libro real a través de un visualizador de mano y ve sobre las páginas contenidos virtuales. Así el alumno ve una escena de RA y le ayuda a introducirse dentro de la escena y experimentarla en un entorno virtual inmersivo [Woods02]. La figura 2 muestra el

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Magic Book y sus aplicaciones en la enseñanza de materias como los volcanes y el sistema solar.

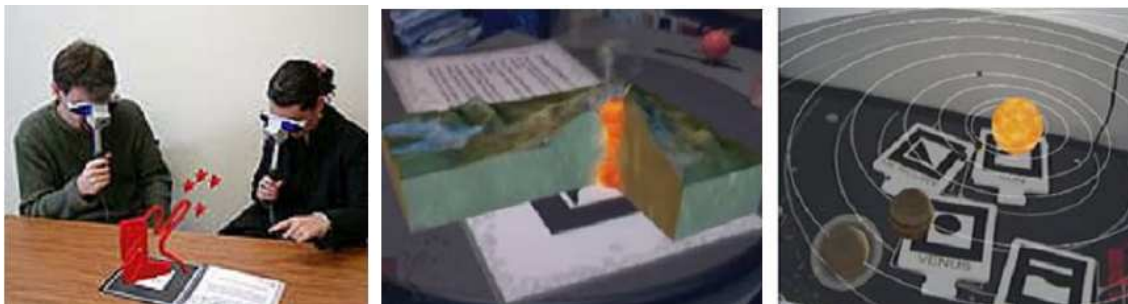


Figura 2 El «Magic Book» y sus aplicaciones de educación.

Universidades como Universidad Politécnica de Valencia (UPV), Massachusetts Institute of Technology (MIT) o Harvard están desarrollando en sus programas y grupos de educación aplicaciones de RA en formato de juegos; buscan involucrar a los estudiantes en situaciones que combinan experiencias del mundo real con información adicional que se les presenta en sus dispositivos móviles. Se han creado juegos como «Environmental Detectives» que utiliza PDA con sistema GPS y el juegos de interior «Mystery @ The Museum» que se basa en la tecnología Wi-Fi [Klopfer08].

No sólo se han creado juegos, sino que ya hay aplicaciones y libros comerciales como LearAR o LarnGear, enfocados a la enseñanza y la química.



Aplicación de LarnGear



Aplicación de LearnAR

1.2 Objetivos

Dirigido a la educación, el objetivo principal de este proyecto es el desarrollo de un sistema de RA que permita realizar una simulación de moléculas. Con dicha aplicación de RA se facilitará la comprensión y la enseñanza de nuevos compuestos químicos, de reacciones moleculares, y síntesis de medicinas.

1.3 Estructura del proyecto

Este proyecto se ha dividido en seis capítulos más la bibliografía y glosario

- El primero de ellos, motivación y objetivos del proyecto, nos muestra las posibilidades de la RA, de la necesidad de la química, de los tipos de química existentes, y de su uso en el mundo real, los objetivos que se han marcado y dónde se enmarca el sistema desarrollado.

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

- En el segundo capítulo, introducción a RA y aplicaciones para aprendizaje, se explican las soluciones que se están adoptando, explicamos qué es la RA y se describen los usos actuales sobre la RA en la educación.
- El tercer capítulo, descripción de programas moleculares y prueba de algunos de ellos, se describen las aplicaciones más utilizadas, con más capacidad y las que tienen más años de experiencia en el sector. Se describen que usos principales tienen, sus capacidades, la facilidad de uso, la capacidad de ser extensible, licencias de usos, y la experiencia de instalación y puesta a prueba.
- El cuarto capítulo, uso de ARToolKit e inclusión de moléculas, se expone cómo se realiza una aplicación de RA con ARToolKit, cómo se integra un conjunto de moléculas y el resultado final.
- En el quinto capítulo, conclusiones, se comentan las conclusiones obtenidas, así como los posibles trabajos futuros.
- Por último, se encuentra la bibliografía y el glosario, dónde se encuentran las definiciones de los términos importantes e interesantes.

CAPÍTULO 2. INTRODUCCIÓN A REALIDAD AUMENTADA Y APLICACIONES PARA APRENDIZAJE, EN ESPECIAL DE QUÍMICA.

2.1 Realidad Aumentada, conceptos

Un sistema de RA es aquel que enriquece el mundo real con elementos virtuales, generados mediante ordenador, que coexisten en un mismo espacio con los objetos reales. En Realidad Virtual, el usuario se sumerge en un mundo totalmente virtual, sin tener contacto con el mundo real que le rodea. Sin embargo, la RA permite que el usuario vea el mundo real aumentado con información adicional generada por ordenador. Este aumento consiste en objetos virtuales que se añaden al entorno o en información no geométrica sobre los objetos reales existentes. Existe una definición genérica que define RA como sistemas que deben tener las siguientes tres características [Azuma01]:

- Combinación de imagen real y virtual
- Interacción en tiempo real
- Localización 3-D

Así pues, con esta definición, se pueden incluir aquellos sistemas que utilizan como dispositivo de visualización HMD's (Head-Mounted Display o casco) u otro tipo de tecnologías tales como dispositivos basados en monitor, sistemas monoculares, etc., siempre y cuando estas mantengan las características esenciales de la RA. Por ejemplo, eso no incluye películas o superposiciones 2-D en la imagen. Películas como «Jurassic Park» que poseen objetos virtuales fotorrealistas (dinosaurios), o «Minority Report» con imágenes holografías que interactúan con los personajes, aunque han sido perfectamente fusionados con el entorno 3-D no pueden ser consideradas como RA porque no existe una interacción con ellos. De igual modo que no lo son los objetos virtuales superpuestos a vídeos, ya que no están combinados con el mundo real en 3-D. Ejemplos de RA son en la actualidad, por ejemplo, los que se usan en los dispositivos móviles como SpyGlass, una aplicación para iPhone que integra una aplicación RA, con la cámara, el GPS del móvil, la brújula del móvil, y (mediante conexión a red) la base de datos de Google indicando dónde estás y qué sitios de interés hay alrededor tuyo, con información superpuesta sobre la imagen capturada.

Así, con la RA, lo que se consigue es complementar la realidad en lugar de reemplazarla, ofreciendo de este modo un mayor realismo en todas nuestras acciones. En la figura 4 (a) se muestra una imagen real, mientras que en la 4 (b) se ve el modelo en alámbrico de otra totalmente virtual. Y por último, en la figura 4 (c), vemos un sencillo ejemplo de lo que podría considerarse como RA, combinando las dos anteriores. En ella aparece un conjunto de personas en una calle real y junto a ellos hay un coche generado por ordenador que parece formar parte de la misma escena. Tanto es así que el coche tapa parcialmente a dos de las personas que allí aparecen y proyecta su sombra como cualquier otro objeto de los allí presentes.

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas



Fig. 4 (a): Imagen real

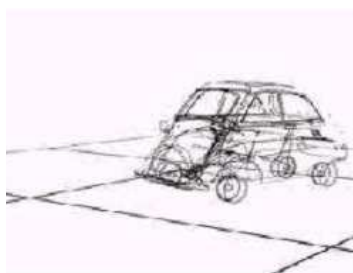


Fig. 4 (b): Modo alámbrico



Fig. 4 (c): Imagen aumentada

Otro ejemplo, comercial en el que se ve un cliente que va a la peluquería, antes de hacerse el cambio de estilo se puede ver con el cambio de pelo realizado, decidiendo si es de su agrado. En la Figura 5 (a) se ve al cliente posicionándose ante el dispositivo para que reconozca sus facciones, en la 5 (b) muestra en una malla las características de reconocimiento donde pueden ajustarse a mano y hacer coincidir los puntos claves, y en la 5 (c) se puede ver que el programa le superpone un pelo distinto. El cliente puede interactuar, girando e inclinándose para verse mejor, o cambiar el estilo y el pelo.



Fig. 5 (a):
Im. Real



Figura 5 (b):
Im. ajustando parámetros



Figura 5 (c):
Im. aumentada

El objetivo último de la RA es crear un sistema en el que no se distinga el mundo real de los objetos virtuales incluidos en él, logrando así una fusión total de ambos entornos.

2.2 Tipos de Realidad Aumentada

Existen distintas clasificaciones de RA y no establecidas como definitivas, se puede clasificar por cómo son las interfaces de interacción con la aplicación:

→ Interfaces tangibles basadas en el uso de marcadores.

Ishii et al. definen las interfaces tangibles como aquellas interfaces en las que aumentan el mundo físico real emparejando información digital con entornos y objetos físicos del día a día. En ellas, el usuario manipula un elemento real al que se le ha colocado un marcador, y los resultados son reflejados en los movimientos del correspondiente objeto virtual asociado.[Hiroshi97]

→ Interacción basada en movimiento corporal.

Una forma natural de interacción en sistemas de RA consiste en la detección y

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

seguimiento del movimiento de algún miembro del cuerpo. Existen distintas tecnologías para realizar el seguimiento de la posición y orientación de las manos, entre otras, tracking magnético, tracking inercial, sistemas de reconocimiento basados en visión, etc.

→ Interacción basada en dispositivos de bajo coste.

Diversos sistemas de RA hacen uso de dispositivos fácilmente accesibles en el mercado, a los que incorporan sistemas de tracking, y los utilizan como dispositivos de interacción. Ejemplos de éstos son: cubo con pegatinas reflectantes de infrarrojos [Martens04], una pieza de material médico accesible en mercado [Straier05], usando como métodos de entrada y de salida una pizarra y una PDA diseñados de tal forma, que el sistema de tracking es capaz de detectar su posición y orientación sobre un mapa usando visión por computador [Reitmayr05], espejo virtual (dispositivo de medicina comercial) con infrarrojos [Feuerstein07], usando el Wiimote de la consola Wii de Nintendo [Looser07], o usando teléfonos móviles [Rohs06] y [Henrysson06].

→ Interacción multimodal

Se define los sistemas multimodales como aquellos que procesan métodos de entrada naturales combinados (como voz, lápiz, táctil, gestos de mano, guiños, y movimientos corporales) de forma coordinada con la salida multimedia del sistema. [Oviatt99]

También se puede clasificar por como se muestran las imágenes [Bimber05]:

- Visores sobre la cabeza: Se muestran directamente las imágenes en los ojos mediante un dispositivo que se pone sobre la cabeza, Head Mounted Display (HMD).
- Visualizador de mano: El usuario lleva una pantalla pequeña que cabe en la mano y en el que va viendo
- Proyector espacial: En lugar de que el usuario use o lleve consigo la pantalla; Realidad Aumentada espacial (SAR) hace uso de proyectores digitales para mostrar información gráfica sobre los objetos físicos

Otra catalogación posible, atendiendo a distintos criterios [Portalés08]:

- Según el entorno físico en el que se desarrolla la aplicación, podemos diferenciar entre sistemas dentro de recintos o **cerrados** («*indoors*») vs. sistemas al aire libre o **abiertos** («*outdoors*»). La diferencia entre aplicaciones dentro de recintos y las aplicaciones al aire libre condiciona muchos aspectos de los sistemas de RA establecidos, principalmente el tipo de dispositivos de registro y displays utilizados.
- Según la extensión que abarquen, podemos distinguir entre sistemas **locales** («*local*») vs. **ubicuos** («*ubiquitous*»). Los sistemas locales se desarrollan en un ámbito acotado, bien en espacios abiertos o dentro de recintos.

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

- Según la movilidad de los dispositivos de registro y/o displays podemos distinguir entre **sistemas móviles** («*mobile*») vs. **sistemas espaciales** («*spatial*»). 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.
- Según el número de usuarios que simultáneamente pueden interactuar con el sistema, podemos distinguir entre sistemas **individuales** («*individual*») vs. **colaborativos** («*collaborative*»). 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** («*face-to-face*») vs. **remotos** («*remote*»). 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.

2.3 Campos de aplicación

En los últimos tiempos, la RA ha generado un gran interés en investigadores, diseñadores de software y usuarios, convirtiéndose así en un área actualmente en expansión. Así pues, se muestra cómo una nueva e interesante forma de interacción entre las personas y los dispositivos. El trabajo de los desarrolladores de RA ha conseguido crear soluciones en muchos y diversos campos, tales como el militar, la publicidad y otros muchos sectores que se detallarán en los apartados siguientes. Se citan sectores en los que se ha aplicado RA, pero cabe destacar que la RA es aplicable a cualquier campo en el que añadir información que no existe en el mundo real ayuda al usuario en su tarea. En cada uno de los campos se van a citar algunos ejemplos. Lo que se pretende es mostrar ejemplos en distintas áreas con el objetivo de ver la aplicabilidad que ha tenido hasta el momento la RA.

2.3.1 Medicina:

esta es una de las áreas más importantes por la mejora en facilidad y precisión para los especialistas, aportando en menos tratamientos a el paciente y mejorando el postoperatorio.

La mayoría de las aplicaciones médicas están relacionadas con la visualización y entrenamiento para la cirugía. El estudio de las imágenes preoperatorias del paciente, que proporcionan al cirujano la vista necesaria para hacerse una idea de su anatomía interna. Sin embargo, estas son imágenes planas.

Cuando son cirugías que requieran incisiones mínimamente invasivas, uno de los principales problemas consiste en que se reduce la visión del doctor en el interior del paciente, haciendo la cirugía mucho más dificultosa. Se ha desarrollado una aplicación del MIT en colaboración con hospitales, con la que puede crearse un modelo 3D a partir

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

de las múltiples vistas y cortes del estudio preoperatorio, permitiendo a los cirujanos hacerse una idea mucho más clara y realista de la anatomía del paciente. Superpone en tiempo real la reconstrucción 3D de las estructuras internas del paciente sobre la imagen del mismo. En la figura 6 se ve a un paciente en la mesa de quirófano al que se le va a realizar un corte y el cirujano se ayuda de la aplicación para mejorar los resultados. Esta aplicación supone reducción de costes y reducción en el tiempo de intervención. Esta técnica ya ha sido utilizada en el Hospital “Brighan and Women’s” para más de 200 casos neuroquirúrgicos, y se está utilizando rutinariamente para 1-2 casos por semana [Grimson96].



Fig. 6: Paciente en operación, con tecnología de RA

2.3.2 Entretenimiento

Este campo es el que está más en auge, ya que cada vez más creadores de juegos intentan crear nuevos métodos de interacción con el jugador que lo envuelva más en el juego. Desde juegos de karaoke en los que puedes verte en un escenario mientras cantas y te van superponiendo los tonos y letras, a dispositivos más complejos como los que va a sacar microsoft para su consola Xbox 360: Proyecto Natal [PNM09]. En la figura 7 se ve a una usuaria de la consola seleccionando ropa con la que probarse en RA.



Fig. 7: Aplicación de RA de Xbox 360

Otro ejemplo más conocido y referenciado es el ARQuake (96). Dos estudiantes de la universidad de Australia del Sur, han adaptado el juego Quake a un sistema de RA. Consiste en llevar un equipamiento consistente en unas gafas/pantalla semitransparentes, un dispositivo de captura de posición (GPS) para ubicar al jugador, un ordenador portátil y una pistola para videojuegos. Con todo el equipamiento el

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

jugador llevaba en sus primeras versiones unos 14 kilogramos de peso, y cuesta unos 20000 euros. A partir de un plano del sitio donde se va a jugar, se ha creado un escenario para poder jugarlo con el juego. Después se han trasladado los inputs del sistema RA al juego. El resultado final es poder jugar al Quake con una inmersión mayor y la sensación de que ocurre en la realidad [Thomas00]. En la figura 8 (a) se puede observar como un jugador de ARQuake utiliza el sistema HMD y una pistola de juego, en la figura 8 (b) se ve la aplicación en acción, y en la figura 8 (c) se ve a los creadores de ARQuake con el sistema de display y mochila.



Fig. 8 (a): Jugador



Fig. 8 (b): Vista subjetiva



Fig. 8 (c): Los creadores

Un último ejemplo es «ARhrrr» en el que se sobrevuela una ciudad infestada de zombis. Se ha de liquidar zombis intentando que no mueran los pocos supervivientes que quedan, los zombis atacarán a los civiles, y también lanzarán objetos con los que intentan matar al jugador. Es un juego de RA que necesita un teléfono móvil con un mínimo de requisitos y un mapa donde se creará la acción [AEL09]. En la figura 9 (a) se puede observar la perspectiva que aprecia el jugador en el móvil, y una imagen del jugador utilizando Arhrrr, en la figura 9 (b) se ve una escena del juego donde se aprecian zombis, escenario y parámetros del juego como puntos vida,...

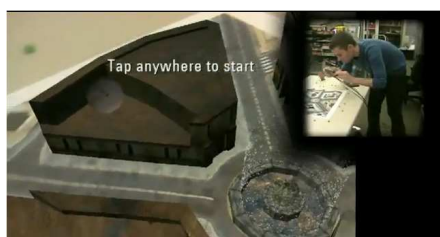


Fig. 9 (a): PJ y perspectiva subjetiva



Fig. 9 (b): Escena del juego

Por último se muestra otro juego de RA en el que se va interactuando con cubos con marcas en sus lados, lo que muestra el sistema es una habitación dentro del cubo con las mismas dimensiones, puede tener escaleras, puertas y diferentes niveles cada marca de un cubo muestra una habitación interconectada, además se puede interactuar con más cubos. Por último aparece en la primera habitación una persona e inclinándolo el

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

cubo se desplaza en un sentido u otro, al cruzar una puerta cambia de habitación y algunas habitaciones conectan con otros cubos. Así pues, hay que llevar al personaje de la salida a la llegada. En la figura 10 (a) se ven tres cubos juntos con la visualización de 3 habitaciones interconectadas. En la figura 10 (b) se ve como inclinando el cubo el personaje se dirige a un extremo de la habitación y así conseguir cambiar de habitación [Oliver09].



Fig. 10 (a): Tres habitaciones juntas



Fig. 10 (b): Interacción jugador juego

2.3.3 Psicología

En el campo de la psicología cabe destacar el trabajo desarrollado en la Universidad Politécnica de Valencia por Juan, Alcañiz, Monserrat, Botella, Baños, y Guerrero, que es pionero en el desarrollo de sistemas de RA para el tratamiento de distintos trastornos psicológicos. Se presenta un sistema de RA para el tratamiento de fobias a animales pequeños como cucarachas y arañas (Figura 11), iniciado con un proyecto final de carrera en la facultad de informática de la UPV, realizado por el alumno César Carrión Villaplana en el año 2004. 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. Se realizaron otras dos versiones de este sistema, uno con marcas invisibles y otro mediante un HMD óptico. Ambos sistemas se evaluaron para presencia, ansiedad y juicio de la realidad con usuarios sin fobia [Juan05; Juan06a; Juan07; Pérez07].

También se desarrollaron dos sistemas de RA para el tratamiento del miedo a las alturas (Figura 12), evaluados de igual forma que los trabajos anteriores; el primero de ellos incluye, como elementos virtuales, fotografías inmersivas; en el segundo los elementos virtuales están mezclados con el entorno real de una habitación, simulando que el suelo se cae y las paredes se elevan. El sistema de fotos navegables se comparó con un entorno real, mientras que el segundo se comparó con un sistema de realidad virtual con las mismas características (habitación real modelada) [Juan06b; Pérez07].

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas



Fig. 11 a) Primer sistema de RA para tratar fobias a insectos



Fig. 11 b) Sistema de RA sin marcas



Fig. 11 c) Sistema de RA mediante HMD óptico



Fig. 12:a) Usuario inmerso en un sistema CAVE para tratar el vertigo



Fig. 12 b) Sistema de RA con tratamiento de vertigo

2.3.4 Área Militar

El área por antonomasia de la RA ha sido siempre el ámbito militar, ya que es dónde más se usa, y desde hace más tiempo, se puede decir que los albores se fraguaron en el ejército. La información en sus inicios era para facilitar datos de navegación, se reflejaba en cristales para que el piloto tuviera toda lo que necesitaba mientras pilotaba, como latitud, longitud, inclinación, horizonte, brújula, presión, ...

Una de las primeras aplicaciones que se diseñó fue «Head Up Displays» para los aviones de combate, sobre una pieza de vidrio transparente se proyecta toda la información necesaria para el piloto, que se superpone sobre la vista hacia el exterior. Ello permite seguir simultáneamente lo que ocurre en el exterior del avión y tener a mano todos los datos de la aviónica sin desviar la mirada. En la figura 13 (a) se puede ver el visor que se coloca en el cristal delantero del avión. Uno de los primeros aviones con este dispositivo fue el TSR-2 (1965), figura 13 (b).

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas



Fig. 13(a): Visor de RA para aviones



Fig. 13 (b): Avión TSR.2

Otra aplicación más moderna y en continuo desarrollo es el Body Wearable Computer (BWC). Puede tener controlada en todo momento la posición de sus compañeros, consultar mapas del terreno, calcular y adjudicarse objetivos, recibir órdenes precisas, ... No ocupa espacio ya que va incorporado a la ropa o el cuerpo. En la figura 14 se puede ver a un soldado equipado con un sistema BWC, tiene sensores en la ropa, visor en el casco para mostrarle información, y una mochila con lo necesario para el sistema, como un equipo de comunicaciones, un GPS, brújula.



Fig. 14: Soldado equipado con BWC

Esta aplicación está integrada dentro del conjunto llamado Land Warrior que tuvo tres objetivos principales: la mejora de la letalidad de un soldado individual, el aumento de la supervivencia de un soldado, proporcionar órdenes, comunicaciones y control a un soldado.

Land Warrior tuvo siete subsistemas principales: arma, conjunto integrado del casco, ropa de protección y equipo, ordenador, navegación, radio, software del sistema.

Actualmente las características del sistema Land Warrior incluyen:

Facilitar la identificación de combate del soldado en ruta y la recarga de suministro para reducir los incidentes de 'fuego amigo', y Commander's Digital Assistant herramienta de planificación [Army09]

Se han desarrollado muchas herramientas para los aviones de combate con RA, una de las más destacadas por su utilidad es la de visión nocturna, ya que permite sobre la imagen real realizar mejoras en tiempo real, que permite ver claramente el entorno, además se le añade información relevante a la misión como pueden ser objetivos o

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

líneas de tiro, poner a los aviones enemigos pintados de rojo o a los aliados pintados de verde. En la figura 15 se muestra la vista que obtiene el piloto desde un avión F35 con tecnología (Northrop grumman) de visión nocturna y localización de objetivos sobre un campo de entrenamiento al atardecer.



Fig. 15: Visión desde F35

Otro gran proyecto ha sido el desarrollado por la Armada de Estados Unidos, el cual consistía en un sistema de RA para la navegación de sus barcos durante situaciones climatológicamente adversas. Este sistema permitía a la persona bajo los mandos de la nave tener un rápido conocimiento de la situación bajo este tipo de condiciones (Figuras 16 (a) y (b)). Algunos estudios realizados indicaban que la habilidad de los usuarios para llevar a cabo complicadas maniobras de navegación bajo este tipo de situaciones desfavorables mejoraba considerablemente mediante la utilización de este sistema. Actualmente este tipo de tecnología está intentando ser trasladada a los barcos no militares. De momento sólo se ha experimentado en yates, pero se cree que estos sistemas revolucionarán por completo la navegación y que llegarán a convertirse en elementos tan habituales en los barcos como lo son los radares o las cartas de navegación [Benton04].



Figura 16 (a) - Navegación durante un día soleado

Fig. 16 (b) - Navegación durante un día con niebla.

2.3.5 Diseño industrial

Generalmente la utilidad de la RA en este sector, ha estado más relacionada a facilitar el entendimiento de los proyectos a los clientes. Permite ver un modelo

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

completo de lo que se le presentará al cliente aun sin poderle mostrar nada más que una parte de forma real. De esta manera, el modelo físico diseñado sería proyectado en 3D al cliente, permitiéndole contemplarlo de una manera mucho más realista. Y, además, se permitiría la interacción en tiempo real con elementos del diseño para que se pudieran hacer los ajustes y cambios necesarios en la estructura del mismo. Como ejemplo podemos tomar cualquiera de las imágenes que aparecen a continuación y que tratan sobre el diseño de automóviles. En la figura 17, podemos ver el despiece de un coche en todas sus partes, y cómo sería montado. En la figura 18 apreciamos el modelo de una furgoneta con distintos tipos de frontales [Matyszczok02].



Figura 17 despiece del modelo iQ de Toyota



Figura 18: Diseño de furgoneta consultando distintos frontales

2.3.6 Robótica y telerrobótica

El manejo de un robot es a menudo complicado, especialmente cuando el robot se encuentra muy lejos y además existen retrasos en la llegada de la comunicación. En lugar de manejar el robot directamente, es mejor manejar una versión virtual, en primer lugar se especifican las acciones manipulando la versión local virtual en tiempo real. Una vez se decide cuáles son las acciones, el usuario puede llevar a cabo el mismo procedimiento con el robot real. Las versiones virtuales pueden predecir los efectos de manipular el entorno, ayudando así al operador a comprobar si sus acciones llevarán a cabo la tarea deseada. La figura 19 muestra ejemplo de una tarea simulada para un brazo mecánico de un robot, la figura 20 muestra el recorrido de un robot a lo largo de una nave, comprobando antes de su activación si se bloqueará con alguna parte de la nave

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

[Azuma97].

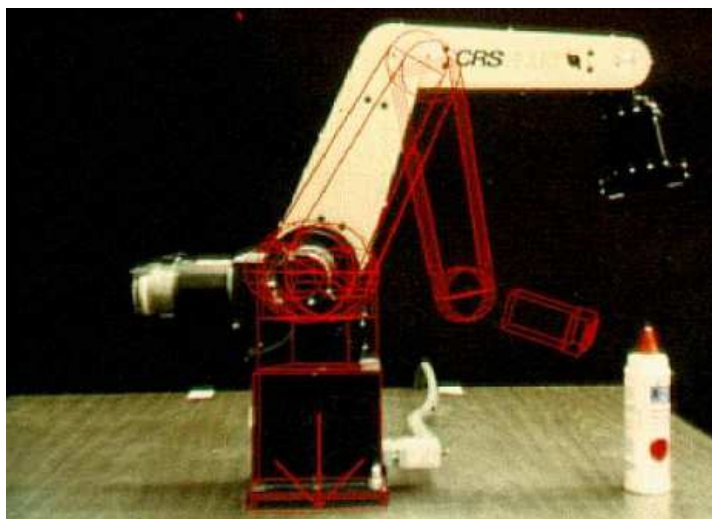


Fig. 19 (a): Las líneas rojas (virtuales) muestran el posible movimiento del brazo del robot



Fig. 20 (b): Se ve la operatividad del robot antes de poner el sistema en la nave.

2.3.7 Fabricación, mantenimiento y reparación

Cuando los técnicos de mantenimiento tengan que reparar o cambiar una determinada pieza de un equipo, en lugar de ojear distintos manuales de reparación podrían utilizar un dispositivo de visión de RA. En este dispositivo la imagen sería aumentada con anotaciones y la información pertinente para llevar a cabo la reparación o el cambio de pieza. También para el cambio de una pieza, se podría iluminar o marcar de alguna manera dicha pieza, facilitando de este modo su labor a los técnicos de mantenimiento. Actualmente se está trabajando para conseguir que todos estos procesos

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

que se han explicado se realicen mediante animaciones de manera que sea mucho más sencilla su comprensión. En la figura 21 [Fruend02] puede apreciarse cómo aparecen una serie de cruces amarillas que nos marcan la posición exacta donde debe situarse el motor. Además en la parte superior izquierda aparecen en rojo una serie de indicaciones que nos dicen cómo llevar a cabo este procedimiento. También se puede ver en la figura 22 el uso de RA por la empresa BMW mediante un proyecto de investigación que llevan a cabo [BMWAR08]. En la Figura 23 el uso de la tecnología de RA en una cadena de montaje, ayuda al montaje de fusibles dependiendo del modelo sin necesidad de buscar y consultar manual. En la Figura 24 se puede apreciar un uso esencial en la que se ve como mediante RA se puede ir guiando en una estación espacial para que se realice el cambio de filtro [Regenbrecht05].



Fig. 21: Ejemplo de una serie de indicaciones para situar un motor

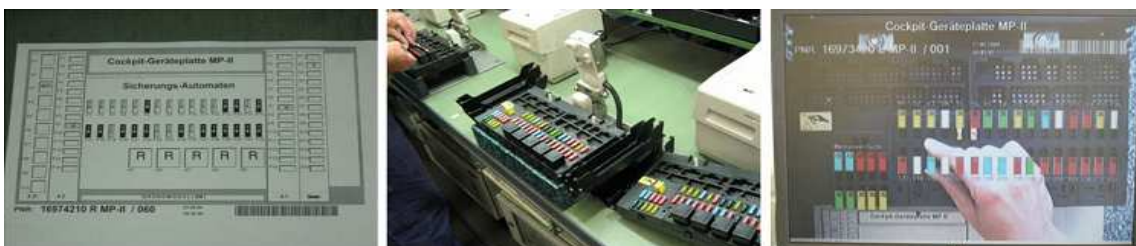


Fig. 23 uso de AR para la ayuda del montaje/ reparación en un sistema de producción.

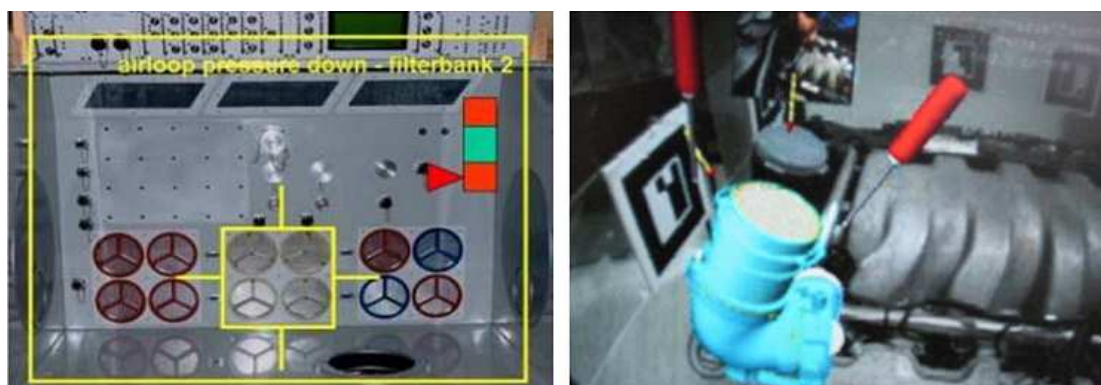


Fig. 24: proyectos de Prestación de servicios de izquierda a derecha: el cambio del filtro de la Estación Espacial y el mantenimiento del motor

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas



Fig. 22 (a): gafas, cascos, micro y sistema de comunicación integrado en técnico

Fig. 22(b): Animación para quitar pieza

Fig.22 (c): Enseñando destornillador a usar.

2.3.8 Arquitectura y construcción

Un campo donde la RA puede tener una gran aplicación es en el mundo de la arquitectura. Para la representación de modelos arquitectónicos con los sistemas de realidad virtual se generan completamente tanto la estructura a construir como el entorno, y en ocasiones es muy complicado o muy costoso a nivel computacional, reproducir fielmente las características (especialmente lumínicas) de un entorno, principalmente si se trata de un entorno natural. En cambio los sistemas de RA facilitan esta tarea, ya que se pueden generar diseños y modelos de edificios, puentes y otras estructuras y ver en tiempo real cómo quedarían estos sobre el lugar donde se desean construir (Figura 25), sin necesidad de generar el entorno también mediante el ordenador [Klinker99].



Fig. 25 (a): Simulación de resultado en la construcción y colocación de un puente

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas



Fig. 25 (b): Edificio posicionado en tiempo real obtenido de la comunidad de google earth

También podemos emplear los sistemas de RA para ayudar en la visualización de la estructura de los edificios o distintas construcciones. Por ejemplo, para dar a los arquitectos una visión de rayos-X dentro de un edificio. Así el arquitecto tendría la posibilidad de conocer dónde se encuentra el sistema de tuberías, el sistema eléctrico o en el caso de la figura 26, los pilares de la estructura del edificio. Estas aplicaciones pueden ser de utilidad tanto al arquitecto como a los técnicos de mantenimiento de estas instalaciones, etc [Tripathi00].



Fig. 26: Vista de la estructura de un edificio mediante RA

2.3.9 Diseño de consumo

Existen aplicaciones orientadas al diseño de interiores, así como al remodelamiento de las casas. De esta forma cualquier persona podría ver su casa con diferentes decoraciones, mobiliario, etc., y hacerse una mejor idea de cómo sería el resultado final de los cambios que desean realizar sobre ella. Por ejemplo la empresa ARQ se dedica a este propósito. Por lo tanto, permite a los usuarios de este tipo de aplicaciones una mayor facilidad en la toma de decisiones puesto que no tienen que imaginar cómo quedarían decoradas, sino que solamente tienen que ojearlas. Es algo similar a lo que podemos observar en la figura 27, donde una persona está viendo cómo queda decorado su salón con un par de sillones y una mesita virtual [Sirna 09].

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas



Fig. 27: Ejemplo de decoración de interiores

En este mismo sentido, existen aplicaciones que nos permiten colocarnos vestimentas virtuales que nos evitarían el tener que ir probándonos los distintos modelos existentes, como se ve en la figura 28.

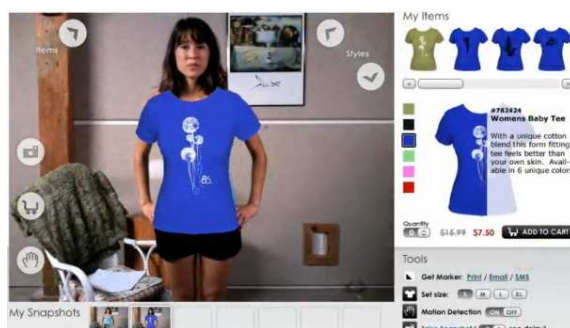


Fig. 28: Tienda virtual Azuraga

Pero sobre todo, en lo que más se utiliza la RA es para el sector de la publicidad, pudiéndose citar muchas soluciones realizadas ya para el mercado, como puede ser los cereales de cuetara, promoción de coches como Nissan, Ford, Toyota, BMW (figuras 29, 30)... revistas de actualidad como Esquire (figura 32), o marcas deportivas como Adidas (figura 31).



Fig. 29: Toyota iQ en movimiento dependiendo de tus acciones



Fig. 30: Mini, la aplicación reconoce la contraportada de una revista

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas



Fig. 31: Zapatillas adidas con marca de RA



Fig. 32: Revista Esquire con RA

En los últimos tiempos la RA se está implantando como un recurso innovador, integrándolo en aplicaciones para dispositivos con GPS. Estas aplicaciones lo que hacen es dar indicaciones de navegadores de GPS directamente en lo que captura el dispositivo. Otras aplicaciones, teniendo GPS y brújula digital, hacen reconocimiento de dónde estas y te van dando información de los edificios, esculturas o museos que vas capturando con la cámara, es decir hacen de guía turístico. Como ejemplo Wikitude Drive (figura 33) o Travel (figura 34).



Fig 33:Wikitude Drive



Fig. 34: Wikitude Trave

2.3.10 Enseñanza

En el campo de la enseñanza puede llegar a ser de gran utilidad mostrar gráficos en 3D, y animaciones para que sea más fácil, intuitivo y didáctico el tema, pero lo cómodo de RA en comparación de un entorno virtual, es que el estudiante puede ir estudiando en el libro para llegado el momento de consultar un recurso animado o en 3D mirar el monitor que puede tener enfrente, como el sistema de RA reconoce la plantilla por la que va el estudiante, le mostrará directamente la animación correspondiente, sin necesidad de buscarla. En la figura 35 se muestra el libro didáctico de LamGear Technology [Dabpetch07].

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas



Fig. 35: Libro didáctico, cámara al lado del estudiante enfocando al libro y monitor presentando animaciones.

Otro buen uso para la RA en la enseñanza es la resolución de ejercicios o problemas, pudiéndose actualizar, independientemente de cuando adquirieses las plantillas, al acceder al sistema (por conexión a Internet) se accede al conjunto didáctico y de problemas más nuevo. Este es el caso de la empresa learnAR, que imparte desde ciencias hasta idiomas, así al entrar en tu asignatura, le has de presentar la plantilla para que te muestre el objeto, animación o pregunta, y para la respuesta también has de presentarle la plantilla adecuada (figura 36) [SCAT10].



Fig. 36: Aplicación LearnAR examinando a un alumno de Francés.

En la UPV se están desarrollando varias aplicaciones de RA enfocadas a la educación, como es el caso del estudio del cuerpo humano para niños [Juan08b]. Es una aplicación de RA que permite visualizar el interior del cuerpo humano. Para ser exactos, el usuario es capaz de abrir el abdomen de un cuerpo humano virtual, usando sus propias manos. El usuario verá, dentro del cuerpo, las áreas donde se encuentran el estómago y los intestinos. En este proyecto, se usaron interfaces tangibles para interactuar con el sistema. Una tela blanca tendida sobre una estructura cuadrada de madera, que define el contorno. Tiene dos cortes en las zonas donde se localizaban el intestino (parte baja del abdomen) y el estómago (parte alta del abdomen) y se puede abrir con cremalleras. En los agujeros hay dos marcas para poder detectarlos. La marca necesaria para mostrar todos los objetos virtuales está situada en una esquina de la plataforma de madera, fijada sobre una estructura cuadrada. El dispositivo se muestra en la figura 37.

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas



Figura 37. Niño usando el sistema de RA del cuerpo humano.

Otra aplicación de enseñanza de la UPV es sobre el cambio climático, es un juego para niños [Juan08a]. Con el juego se pretende concienciar al niño para el reciclado. Con esto se consigue que se interese en reciclar y aprenda a hacerlo de forma correcta. El niño tiene que aprender a separar los residuos y a clasificarlos correctamente. El niño a de colocar el residuo en su cubo de reciclado. Este juego funciona en un móvil con cámara, al principio se le muestra un vídeo explicando la situación climática y como se recicla. Cuando empieza a jugar, el niño va recorriendo con el móvil un terreno, en el que hay unas marcas que el móvil reconoce y pone un residuo en ella. En la parte baja de la pantalla están los cubos y el niño ha de llevar el residuo a su cubo para que le den puntos. En la figura 38 se ve como el niño intenta colocar el residuo en su cubo.



Fig. 38: Niño jugando a reciclar

La UPV también ha desarrollado otros juegos para la enseñanza a los niños, enfocados al medio ambiente, en específico, para la concienciación de los animales en extinción. Dos juegos enfocados al mismo temas son el de cubos magnéticos y el de encontrar parejas. En los dos juegos se le muestra un vídeo al alumno, así le conciencia, se le muestra la lista de animales en peligro de extinción y como va el juego.

En el de cubos, se utilizan 3 cubos con marcas. Los de los extremos muestran 8 marcas diferentes, y el del medio 4. El sistema le pregunta por un animal y el jugador, girando el de la derecha y el del medio va viendo distintos animales, cuando encuentra el correcto, ha de girar el tercero hasta que muestre un asterisco. El juego entonces te dice si has acertado y si quieres ver un vídeo relacionado [Juan10a]. En la figura 39 se ve el juego en uso por una alumna del colegio de verano de la UPV

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas



Fig. 39: Juego de animales en extinción

El de parejas tiene el mismo sistema, la diferencia es cómo se interactúa con el jugador. El juego ve la imagen en un visor que hay tras un cristal, y enfrente tiene un conjunto de plantillas, todas bocabajo. Cuando el juego pregunta por un animal, el jugador ha de ir girando plantillas hasta encontrar la que cree que es correcta. En la figura 40 se ve como un jugador intenta encontrar el animal correcto.



Fig. 40 Niño frente a monitor jugando a las parejas en RA

En la UPV también se han hecho juegos de RA para niños de educación primaria, enfocado al aprendizaje de las letras, y las palabras [Juan10b]. Mediante marcas se van mostrando letras asociadas a un dibujo de algo que empieza por esa letra. Este juego tiene opciones de juego como, con que empieza..., con que acaba..., completa la palabra, o encuentra el intruso. Son juegos sencillos en los que el niño practica las letras y aprende palabras, se puede ver un ejemplo en la figura 41, en la que el niño ha de encontrar la letra que falta para completar la palabra.

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

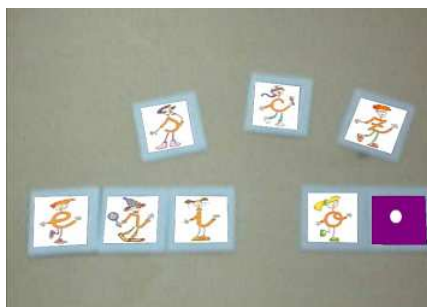


Fig. 41: El niño a de encontrar la letra que falta para completar «erizo»

Centrando el interés de la aplicación de la Realidad Aumentada en la educación superior se pueden indicar diferentes grupos activos y aplicaciones; destacamos entre ellos las aplicaciones realizadas para distintas disciplinas académicas como la enseñanza de conceptos de ingeniería mecánica [Mourkoussis04] en combinación de Web3D, la enseñanza de matemáticas [Esteban07] y la enseñanza de geometría [Kaufmann10] como se ilustra en las figura 42 y 43.



Figura 42 y 43. Sistema de enseñanza de geometría basado en la plataforma Studierstube

CAPÍTULO 3: Programas de Modelado Molecular.

3.1 Introducción

En este capítulo se va a realizar un análisis y estudio de los programas de edición y modelización molecular más importantes o más extendidos. A continuación se irá exponiendo uno a uno los programas intentando ser lo más objetivo posible, se detallará si hubo problemas en la instalación o si realizaban lo esperado.

3.2 Exposición de las aplicaciones

Abalone

Nombre Abalone

Creador/es - Propietarios Agile Molecule

Contacto

E-mail: agilemolecule@gmail.com

Momsreg./VAT No: SE661226355501

Stockholm

Sweden

Socios-Patrocinadores

Division of Physical Chemistry, Arrhenius Laboratory of Stockholm University.

MDynaMix project.

Laboratory of Chemical Cybernetics at Moscow State University.

PC GAMESS/Firefly project.

Sistemas Operativos en los que está implementado Windows

Versión 1.6.12 → estable

Fecha inicio del proyecto/programa 2006

Fecha última modificación actualización del proyecto 2010

Idiomas Inglés

Licencia Copyright © 2006-2010 Agile Molecule.

Precio Gratuito para uso personal.

Formatos de entrada

MolMeccano (*.mlm)

HyperChem (*.hin)

SYBYL mol2 (*.mol2, *.ml2)

Protein Data Bank (*.pdb, *.ent)

XYZ (*.xyz, *.xmol)

Formatos de salida / Tipos creables

MolMeccano (*.mlm)

HyperChem (*.hin)

Protein Data Bank (*.pdb, *.ent)

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

XYZ (*.xyz, *.xmol)

Dependencias ninguna.

Requisitos

Pedidos por el creador no especificados

Espacio en disco 18 MB

Memoria 55 MB

Lenguaje No descrito

Características:

Representación en alámbrico	✓
Representación CPK en alámbrico	✓
Representación en palo y bola	✓
Representación en palo	✓
Representación CPK	✓
Calculo distancia entre átomos	✓
Calculo de ángulo entre átomos	✓
Calculo de ángulo dihedral	✓
Calculo de energía intermolecular	✓
Calculo de energía	✓
Calculo del gradiente relativo	✓

Diseñado principalmente para:

Simular el plegamiento de las proteínas y las complejas ligaduras de ADN en el campo de fuerza AMBER (anacronismo en inglés de «construcción de modelo asistido con perfeccionamiento de energía»).

Resumen

Pros Uso gratuito personal, simple y rápido

Contras Pocas opciones de creación, y no muy intuitivas.

Comentarios / Problemas

Después de incluir los ejemplos facilitados en la web, crear unas moléculas de prueba (amino ácido), añadir una molécula pre-hecha al diseño y cerrar sin guardar, falló el programa debiendo de forzar el cierre. Después de varios intentos más, siempre se repite el error antes obtenido al hacer una molécula, añadir otra y cerrar sin guardar.

AutoDock

Nombre AutoDock

Creador/es - Propietarios

Michel F.Sanner, Ruth Huey, Sargis Dallakyan, Sowjanya Karnati, William /Lindy) Lindstrom, Garrett M. Morris, Brian Norledge, Anna Omelchenko, Daniel Stoffer, Guillaume Vareille

The Scripps Research Institute; California: 10550 North Torrey Pines Road, La Jolla, CA 92037 - (858) 784-1000, Florida: 130 Scripps Way, Jupiter, FL 33458 - (561) 228-2000

Software de terceros:

Fast Isocontouring, Volume Rendering – Chandrait Bajaj, tu Austin

GL extrusion Library (GLE) – Linas Vepstas

Secondary Structure Assignment (Stride) – Patrick Argos EMBL

Mesh Decimation (Qslim 2.0) – Micheal Garland, University of Illinois

Tiled Rendering (TR 1.3) – Brian Paul

Sistemas Operativos en los que está implementado

Unix/Linux i86/i86_64/IA64, AIX5, Mac OS X, Windows, Sun SPARC Solaris.

Posibilidad de compilar en otras plataformas.

Versión

MGLTools 1.5.4

Fecha inicio del proyecto/programa

4/11/2006; primera versión estable: MGLTools 1.4.3 22/09/2008

Fecha última modificación actualización del proyecto 13/03/2010

Idiomas Inglés

Licencia

AutoDock GNU/GPL

«AutoDockTool» se distribuye dentro del paquete de MGLTools. A menos que se especifique lo contrario, MGLTools se licencia por Michel F. SANNER y The Scripps Research Institute. Algunas partes del software de este paquete proporcionan sus propias licencias.

Todos los componentes que no especifican sus propias licencias están bajo la licencia de código abierto de Mozilla.

© 1999-2009 Molecular Graphics Laboratory, The Scripps Research Institute

Coste AutoDock → Gratuito

ADT → Gratuito versión académica.

→ Poner en contacto con el creador para comercialización o uso profesional.

Formatos de entrada

MolMeccano (*.mlm)

HyperChem (*.hin)

XYZ (*.xyz, *.xmol)

Protein Data Bank (*.pdb, *.ent)

MEAD Files (*.pqr)

SYBYL mol2 (*.mol2, *.ml2)

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

AutoDock Files (*.pdbq, *.pdbqs, *.pdbqt)
Gromacs Files (*.gro)
MMCIF Files (*.cif)
Phyton Scripts (*.py) (ojo es para cargar programaciones sobre moléculas hechas para este programa)
Resource Files (*.rc)

Formatos de salida / Tipos creables

Protein Data Bank (*.pdb, *.ent)
AutoDock Files (*.pdbq, *.pdbqs, *.pdbqt)
MMCIF Files (*.cif)
MEAD Files (*.pqr)
VMRL 2.0
STL

Dependencias

PyThon 2.5.2 → 59MB HDD

Opcionales y sólo en versión académica:

MSLIB → Paquete para calcular la superficie molecular.

STRIDE → Paquete usado para calcular estructuras secundarias

UTPackages → Paquete/s usado para renderizar isocontornos y volúmenes.

Requisitos

Pedidos por el creador:

mínimos y recomendados:

facilitados en <http://secondlife.com/support/system-requirements/> y en <http://www.adobe.com/products/aftereffects/opengl.html>.

comprobados en sistema:

Espacio en disco: 120MB
Memoria: 97MB al arranque – 300MB en uso.
CPU

Lenguaje PyThon

Características:

Representación en alámbrico	✓
Representación CPK en alámbrico	✓
Representación en palo y bola	✓
Representación en palo	✓
Representación CPK	✓
Posicionamiento intermolecular	✓
Movimiento moleculares	✓
Calculo distancia entre átomos	✓
Calculo de ángulo entre átomos	✓
Calculo de ángulo dihedral	✓
Calculo de energía intermolecular	✓
Calculo de energía	✓
Calculo del gradiente relativo	✓
Visionado de estructuras en 3D (con gafas para ver en 3D)	✓

Diseñado principalmente para

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Autodock es una suite de herramientas de generación automática de acoplamiento. Está diseñado para predecir cómo moléculas pequeñas, como substratos o candidatos de drogas, se unen a un receptor conocido, en estructuras 3D.

Autodock en realidad consta de dos programas principales: Autodock realiza el acoplamiento del ligando a un conjunto de enlaces para describir la proteína objetivo, AutoGrid pre-calcula estos enlaces.

Además de su uso para el acoplamiento, pueden ser visualizadas los enlaces de afinidad atómica. Esto puede ayudar, por ejemplo, para guiar a los químicos orgánicos sintéticos con un mejor diseño de los enlaces.

También desarrolla una interfaz de usuario gráfica llamada AutoDockTools (ADT), que entre otras cosas ayuda a establecer vínculos, girar las moléculas con los enlaces y el análisis a escala.

Autodock tiene usos en:

- Cristalografía de rayos X;
- Estructura basada en el diseño de fármacos;
- La optimización de plomo;
- High-throughput screening (HTS);
- Combinación de bibliotecas de diseño;
- Anclaje proteína-proteína;
- Estudios mecanismo químico.

Resumen

Pros: Programa muy especializado en el análisis y comportamiento de moléculas, e interacción entre ellas, además del movimiento de estas y cómo se posicionarán entre un conjunto molecular. Contiene muchas aplicaciones y extensiones para cubrir la mayor necesidad de un químico que se encuentre en necesidad de este tipo de funciones. Tiene un gran abanico para el análisis y las interacciones moleculares.

Contras: Al ser muy completo y disponer de muchas opciones de simulación, no es sencillo de manejar sin un documento al que poder consultar. Aunque está completamente documentado, hay que realizar una tutorización al programa o estar realmente muy introducido en el campo químico de las moléculas.

No es sencillo ni inmediato la realización de nuevas moléculas, está concebido para las interacciones moleculares, sus posicionamientos, reacciones,...

Comentarios / Problemas:

Este programa es sencillo de instalar con el binario preparado para las distribuciones, tiene muchísimos métodos de análisis, pero no es sencillo de manejar, aunque lo más directo (la visualización de moléculas) es muy simple de entender.

Amira

Nombre Amira (Molecular Option)

Creador/es - Propietarios Visage Imaging©, que a su vez es un subsidiario de Mercury Computer Systems, Inc.

Sistemas Operativos en los que está implementado

Windows® 2000/XP/Vista, 32-bit y 64-bit

Mac OS® X 10.5 (Leopard)

Red Hat Enterprise Linux 5.0 para x86_64 o compatible

Versión 5.2.2 (estable)

Fecha inicio del proyecto/programa no especificados

Fecha última modificación actualización del proyecto 2010

Idiomas Inglés

Licencia

Copyright © 2010 Visage Imaging GmbH. All rights reserved.

Coste

Existen diferentes licencias con diferentes costes:

- Licencia de prueba: 15 días con uso completo gratuito.
- Licencia permanente: Puedes pedir por módulos, te envían individualmente la licencia después de consultar tu petición.
- Licencia de actualización: Para cuando hay un módulo nuevo o una versión de Amira nueva que quieres utilizar teniendo ya una activa.
- Licencia para agregar/cambiar equipos.

Después de una semana responden proporcionando los siguientes precios:

Product Option	License Price*	Annual Maintenance*
Amira (base package)	€ 3,600.00	€ 720.00
Amira Developer Option	€ 2,700.00	€ 540.00
Amira Microscopy Option	€ 1,800.00	€ 360.00
Amira Molecular Option	€ 990.00	€ 198.00
Amira Mesh Option	€ 1,800.00	€ 360.00
Amira Very Large Data Option	€ 4,500.00	€ 900.00
Amira Dicom Reader	free	Free
Amira Virtual Reality Option	on request	on request
other options	on request	on request

*Teniendo un 30% de descuento en el precio si es de uso académico.

Formatos de entrada

AMBER (*.top/crd/vel)

AMF (*.amf)

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

DX APBS DX (*.dx/apbs)
GROMACS (*.gro/top/trr)
MDL (*.sdf)
PDB (*.pdb)
PHI (*.phi)
PSF/DCD (CHARMM) (*.psf/dcd)
Mol2 (*.mol2)
UniChem (*.uni)
ZIB (*.zmf)

Formatos de salida / Tipos creables

AMF (*.amf)
GROMACS (*.gro/top/trr)
PDB (*.pdb)
Mol2 (*.mol2)
UniChem (*.uni)
ZIB (*.zmf)

Además de formatos para modelos 3D (VRML, TIFF, RAW, PSI, Open Inventor, Analyze 7.5,...), imágenes (jpeg, png, pnm, SGI-RGB,...), modelos médicos, ...

Requisitos

Pedidos por el creador

Un procesador con soporte a conjunto de instrucciones SSE2

Tarjeta gráfica con capacidad de aceleración hardware al mapeo de texturas y soporte a OpenGL. Algunos módulos de visualización, es decir, los de representación de más alta calidad de renderizado de volúmenes, requieren hardware de gráficos moderno con soporte de sombreado de fragmentos y vértices, así como la instalación de los controladores gráficos más recientes.

Mínimos:

CPU con soporte al conjunto de instrucciones SSE2
512 MB RAM
Tarjetas gráficas con soporte OpenGL

Recomendados:

CPU - Multi-core con 2 GHz
2 GB RAM
Una tarjeta de «juegos» moderna de cualquiera de los principales vendedores con 512 MB de RAM

Espacio en disco 700 MB

Memoria ~190 MB

Lenguaje C++, OpenGL

Características *paquete básico*: (facilitadas por creador)

- Análisis
- Estadísticas de Materiales
 - Mediciones de volúmenes y superficies
 - Medidas lineales y anotaciones 2D/3D
 - Análisis de co-localización
 - Operaciones aritméticas

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

- Densimetría
- Análisis de superficie
- Integración en MATLAB (Windows y Linux)
- Estadísticas de red

Procesamiento

- Filtrado de imágenes
- Generación de superficies
- Segmentación de imágenes
- Alineamiento en rebanadas
- Registro automático y manual
- Muestreo de volumen e imagen
- Recorte selectivo de volumen
- Edición directa de volumen
- Edición de superficie
- Conversor de tipos de datos
- Automatización
- Fusión con visores en 3D MPR
- Registro manual y automatizado en visores 3D MPR

Visualización

- Cortes ortogonales, oblicuos y curvos
- Renderizado de volúmenes sombreados multi-resolución
- Renderizado de superficies
- Isolíneas e isosuperficies
- Visualización de imágenes multi-canal
- Fusión de imágenes
- Visualización de campo en altura y protuberancias de corte
- proyecciones ortogonales
- Visualización de filamentos de trazado
- Selección 3D ROI
- Recorte y corte
- Visualización de datos con tiempo de caducidad
- Visor 3D + MPR para explorar conjuntos de datos complejos

Presentación

- Herramientas de animación
- Demos de alta calidad
- Generación de película
- Varios modos de visualización en 3D Stereo
- Soporte automático de visualización estereoscópica en anotaciones 2D y 3D.

Editores

- Herramientas de edición de segmentación interactivas y semi-automáticas para la segmentación de imágenes 3D
- Visor multi-planar para la visualización simultánea de más de dos conjuntos de datos en un visor 3+1 MPR
- Editor CameraPath para la creación de una ruta de cámara usando fotogramas clave para animaciones y películas
- Interfaz gráfica del dialogo de color para definir valores de color
- Dos editores mapa de color para modificar los valores RGBA de un mapa de

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

color discreto

- Un editor de curvas para crear y editar curvas
- Filtros de imagen digital para aplicar filtros de procesamiento de imagen estandar
- Editor de cortado de imágenes para cortar imágenes 3D, cambiar el cuadro delimitador, y el tamaño de vóxel (pixel 3D)
- Editor de referencia para añadir, mover o eliminar marcadores en un punto de referencia establecidos
- Editor LineSet para seleccionar, crear, modificar y eliminar polilíneas
- Editor de parámetros para añadir, cambiar o borrar atributos de los datos de un objeto
- Editor simplificador de superficie para reducir el número de triángulos de una superficie
- Editor de superficie para modificar triángulos, borrar intersecciones, asignar identificadores frontera
- Editor de transformaciones para trasladar, rotar, o escalar cualquier objeto
- Interfaz gráfica de las herramientas de impresión para mostrar impresiones en 2D
- Interfaz gráfica de gestor de demostraciones para gestionar y controlar demostraciones

Características *paquete molecular*:

La extracción y visualización de la estructura secundaria

Cálculo y visualización de la trayectoria Molecular

Cálculo de campo potencial

Pseudo-cálculo de la densidad de electrones

Etiquetado de molécula

Cálculo de superficie de molécula

Alinear molécula y las secuencias

Calculo del conjunto Pareto-optimio

Cálculo y visualización de enlaces H

Visualización en palo, alámbrico, cinta y CPK

Visualización dinámica

Evaluación de la configuración de densidad

Calculo de molécula-promedio

Editores:

Molecule Attribute Editor: permite la edición de grupos de atributos contenidos en una molecula

Molecule Editor: permite cambiar geometría y topología de una molécula

Diseñado principalmente para

El paquete básico es obligatorio para todas las aplicaciones de Amira. Se dispone de módulos avanzados de análisis y visualización 3D/4D general. Cuando los datos multidimensionales deben ser procesados y explorados, este paquete ofrece un amplio conjunto de poderosas herramientas con intuitivas interfaces gráficas de usuario para obtener una visión detallada de complejos conjuntos de datos. Además de los requisitos de alta visualización, este paquete responde a las necesidades de análisis avanzado y proceso de datos de la imagen clínica o preclínica, datos nucleares, las imágenes de microscopía óptica o electrónica, y todo tipo de imagen multidimensional.

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Varios módulos también se proporcionan para preparar presentaciones, demostraciones y películas. El paquete básico incluye el potente editor de segmentación y el Multi-Viewer plana.

La **opción molecular** combina sólidas capacidades de Amira para la visualización 3D de datos, como la restitución de volumen por aceleración hardware, con herramientas específicas para la visualización y el análisis molecular, tales como superficies moleculares, alineamiento de secuencias, cálculo de la densidad de la configuración, las trayectorias de la molécula, y más. La opción Molecular Amira viene con demos auto-ejecutables y tutoriales paso a paso para las tareas más comunes de visualización molecular.

Resumen

Programa sencillo con grandes capacidades de representación, permite editar y componer moléculas pero como con otros programas no está pensado para la Creación, sino para edición de moléculas ya creadas. Tiene muchos recursos y una documentación sencilla y con ejemplos. Puedes realizar objetos 3D, editar imágenes, cargar proyectos de CAD, manipular scripts de MATLAB, trabajar con radiografías,... Está pensado para cubrir las necesidades médicas de cualquier investigador.

Pros Fácil instalación y desarrollo no muy complicado. Acepta y escribe muchos formatos. Permite editar moléculas. Permite desarrollar herramientas con integración en este programa mediante un módulo de desarrollo (para C++). Maneja muy bien las moléculas, aprovecha muy bien la memoria y maneja los renderizados. Buena documentación.

Contras Es de pago, permite editar moléculas de manera limitada.

Comentarios / Problemas

Carga rápidamente y va de forma sencilla todo el manejo. Probando moléculas que ya tenía disponibles, al cargar ADN no pudo mostrar correctamente las uniones entre átomos (no aparecen unidos), los demás visualizados sí.

Ascalaph Graphics

Nombre Ascalaph Graphics

Creador/es - Propietarios *Agile Molecule, MdynaMix* (Alexander Lyubartsev, Aatto Laaksonen, Department of Physical Chemistry, Stockholm University), CP2K, Ascalaphus pictures (Vladimir Eskin, Sergey Nikolaev)

Sistemas Operativos en los que está implementado Windows, Linux

Versión 1.6.15

Fecha inicio del proyecto/programa 1993

Fecha última modificación actualización del proyecto 2010

Idiomas Inglés

Licencia Copyright © 2006-2010 Agile Molecule.

MDynaMix Copyright © 1996, 1999, 2005, 2007, 2008 by: Alexander Lyubartsev, Aatto Laaksonen, Department of Physical Chemistry, Stockholm University.

CP2K. Copyright © 2000-2009 CP2K developers group

Ascalaphus pictures. Copyright © 2007 Vladimir Eskin. Copyright © 2005-2007 Sergey Nikolaev. All rights reserved.

Coste Gratuito

Formatos de entrada

MolMeccano (*.mlm)

m.DynaMix (*.mmol)

HyperChem (*.hin)

SYBYL mol2 (mol2, *.ml2)

Protein Data Bank (*.pdb, *.ent)

XYZ (*.xyz, *.xmol)

Formatos de salida / Tipos creables

MolMeccano (*.mlm)

HyperChem (*.hin)

Protein Data Bank (*.pdb, *.ent)

XYZ (*.xyz, *.xmol)

Requisitos

Pedidos por el creador No especificados

Espacio en disco 44 MB

Memoria 93 MB (reservados) (máximo en uso 65 MB)

Subprocesos 5

Lenguaje C++

Características:

Representación en alámbrico ✓

Representación CPK en alámbrico ✓

Representación en palo y bola ✓

Representación en palo ✓

Representación CPK ✓

Calculo distancia entre átomos ✓

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Calculo de ángulo entre átomos	✓
Calculo de ángulo dihedral	✓
Calculo de energía intermolecular	✓
Calculo de energía	✓
Calculo del gradiente relativo	✓

Diseñado principalmente para

Representación de gráficos moleculares
Construcción de modelos moleculares
Optimización geométrica molecular
Modelización dinámica molecular con integración simpléctica de cuatro orden
Modelización cuántica usando NWChem, CP2K y PC GAMESS/Firefly
Integración con paquetes dinámicos moleculares MDynaMix
Simulaciones aceleradas por Hardware tecnología NVIDIA CUDA

Resumen

Aunque este programa también tiene la capacidad de construcción de modelos moleculares no está concebido para ese uso principal. Es básicamente el programa de Agile: Abalone con el añadido de dinámica molecular (MYDynaMix).

Pros Es totalmente gratuito, sin restricción, al ser GPL, pero sólo este paquete. Está más documentado que Abalone.

Contras Pocas opciones de creación y no muy intuitivas.

Comentarios / Problemas

En definitiva es el mismo programa que Abalone, son dos programas que se unieron, la diferencia es que incluye el paquete de dinámica molecular, con el que se puede añadir más opciones. Al ser este paquete GPL, también lo es el programa entero, aunque Abalone solo no es GPL (si gratuito para uso personal). Para el diseño molecular ver programa AscalaphDesigner.

Ascalaph Designer

Nombre Ascalaph, paquete Designer

Creador/es - Propietarios *Agile Molecule, MdynaMix* (Alexander Lyubartsev, Aatto Laaksonen, Department of Physical Chemistry, Stockholm University), CP2K, Ascalaphus pictures (Vladimir Eskin, Sergey Nikolaev)

Sistemas Operativos en los que está implementado Windows

Versión 1.6.15

Fecha inicio del proyecto/programa 1993

Fecha última modificación actualización del proyecto 2010

Idiomas Inglés

Licencia Copyright © 2006-2010 Agile Molecule.

MDynaMix Copyright © 1996, 1999, 2005, 2007, 2008 by: Alexander Lyubartsev, Aatto Laaksonen, Department of Physical Chemistry, Stockholm University.

CP2K. Copyright © 2000-2009 CP2K developers group

Ascalaphus pictures. Copyright © 2007 Vladimir Eskin. Copyright © 2005-2007 Sergey Nikolaev. All rights reserved.

Coste 80€

Formatos de entrada

MolMeccano (*.mlm)

m.DynaMix (*.mmol)

HyperChem (*.hin)

SYBYL mol2 (mol2, *.ml2)

Protein Data Bank (*.pdb, *.ent)

XYZ (*.xyz, *.xmol)

Formatos de salida / Tipos creables

MolMeccano (*.mlm)

HyperChem (*.hin)

Protein Data Bank (*.pdb, *.ent)

XYZ (*.xyz, *.xmol)

Requisitos

Pedidos por el creador *No especificados*

Espacio en disco 96 MB

Memoria 98 MB

Subprocesos 7

Lenguaje C++

Características:

Representación en alámbrico ✓

Representación CPK en alámbrico ✓

Representación en palo y bola ✓

Representación en palo ✓

Representación CPK ✓

Dos cámaras por modelo ✓

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Múltiples ventanas	✓
Constructor de cristales	✓
Constructor de cadenas	✓
Dibujar con el ratón	✓
Edición geométrica	✓
Optimización preliminar (Ad hoc)	✓
Optimización cuántica	✓
Optimización geométrica con métodos de conjugación de gradiente	✓
Características de mecánica cuántica	✓
Manejo de dinámica molecular	✓
Manejo de química cuántica con ayuda a CP2K y PC GAMESS/Firefly	✓
Calculo distancia entre átomos	✓
Calculo de ángulo entre átomos	✓
Calculo de ángulo dihedral	✓
Calculo de energía intermolecular	✓
Calculo de energía	✓
Calculo del gradiente relativo	✓

Diseñado principalmente para

Representación de gráficos moleculares
Construcción de modelos moleculares
Optimización geométrica molecular
Modelización dinámica molecular con integración simpléctica de cuarto orden
Modelización cuántica usando NWChem, CP2K y PC GAMESS/Firefly
Integración con paquetes dinámicos moleculares MDynaMix
Simulaciones aceleradas por Hardware tecnología NVIDIA CUDA

Resumen

Ascalaph Designer proporciona métodos para construir y editar modelos moleculares, optimización de geometría, simulaciones de dinámica molecular. Es básicamente el programa de Agile: Abalone con el añadido de dinámica molecular (MYDynaMix).

Pros Además de lo señalado para Ascalaph, se ha de destacar el diseño explícito de interfaz exclusiva para la creación de moléculas, en líquidos, DNA con agua implícita, o cálculos cuánticos. También la capacidad de múltiples vistas o ventanas y la aceleración por tecnología CUDA

Contras Además de lo señalado para el programa Ascalaph, lo más relevante es que no es gratuito. No está en Linux, MacOS X.

Comentarios / Problemas

Son programas independientes tanto Abalone como todos los programas de Ascalaph, los están desarrollando para fines distintos y colaboran. Cada uno lo desarrolla una compañía distinta aunque todas tienen una en común Agile. Este paquete no tiene dependencias del programa genérico Ascalaph Graphics. Aunque no es gratuito, se puede obtener un paquete/programa totalmente libre y equivalente llamado ***Ascalaph Liquid GPU*** con la diferencia de que está siendo desarrollado para la explotación de

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

GPU con CUDA 2.0, lo que le permite manejar fácilmente más de 5000 átomos.¹

¹ *Ascalaph Liquid GPU* es una versión idéntica de *Ascalaph Designer* con aceleración hardware con todas las características excepto la apariencia. Esta versión requiere un hardware y software especiales. El programa usa una tarjeta gráfica NVIDIA aumentando la velocidad de cómputo.

Amber / Amber Tools²

Nombre Amber 11 / Amber Tools 1.4

Creador/es - Propietarios D.A. Case, T.A. Darden, T.E. Cheatham, III, C.L. Simmerling, J. Wang, R.E. Duke, R. Luo, R.C. Walker, W. Zhang, K.M. Merz, B.P. Roberts, B. Wang, S. Hayik, A. Roitberg, G. Seabra, I. Kolossváry, K.F. Wong, F. Paesani, J. Vanicek, X. Wu, S.R. Brozell, T. Steinbrecher, H. Gohlke, Q. Cai, X. Ye, J. Wang, M.-J. Hsieh, G. Cui, D.R. Roe, D.H. Mathews, M.G. Seetin, C. Sagui, V. Babin, T. Luchko, S. Gusarov, A. Kovalenko y P.A. Kollman.

Sistemas Operativos en los que está implementado

Máquinas NSF TeraGrid
Mac OSX (Power PC e Intel)
Ubuntu Linux
PMEMD en IBM BlueGene/L
Sun OS
Microsoft Windows
MD-GRAPE hardware en RIKEN
Fujitsu Primepower y sistemas VPP
PMEMD con tarjetas GPU NVIDIA

Versión 11 (1.4 Amber Tools)

Fecha inicio del proyecto/programa 1984

Fecha última modificación actualización del proyecto 2010

Idiomas Inglés

Licencia Copyright © 2010

Coste Académica/no lucrativa/gubernamental: \$400. Industrial (para explotación económica): \$20,000 para nuevas licencias, \$15,000 para licencias de Amber 10.

Formatos de entrada *no especificados*

Formatos de salida / Tipos creables *no especificados*

Requisitos *no los especifico la compañía*

Lenguaje C, C++, Fortran90

Características: *no se pudieron comprobar*

Constructor de ácido nucleico DNA	✓
Constructor de Peptidos	✓
Control de cargas parciales y de Amber	✓
Optimizado de estructuras	✓
Control de mecanismos moleculares	✓
Soporte para el desarrollo de campo de fuerzas	✓

Diseñado principalmente para

Simulaciones de dinámica molecular.

Resumen

Programa de modelado molecular clásico.

2 De este programa sólo se pudo probar el AmberTool debido a que para probar Amber se ha de pagar previamente la licencia de 400\$ mínimo. AmberTool está pensado para que funcione con Amber. Así, aunque se instaló AmberTool, las pruebas que se hicieron no son más que test de verificación de la instalación.

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Pros Es un programa de referencia, aun teniendo mucho tiempo el proyecto desde que se inicio, sigue desarrollándose en la actualidad.

Contras Programa de pago, tiene un coste de licencias elevado para uso particular y no se proporciona ningún programa de prueba, test o demo.

Comentarios / Problemas

Aunque se puede ejecutar en cualquier máquina, esto es debido a que se a de implementar el usuario el programa. Ellos proporcionan el código fuente con lo que además de tener conocimientos de compilación, has de tener también de Linux, pues se ha de compilar bajo esta estructura, si se está en otro sistema distinto de Unix te ves obligado a instalar un emulador de UNIX/LINUX como MinGW o Cygwin. Para ser de pago, el soporte que proporcionan es limitado a su programa. Está documentado y la copia cuesta 11,46€, aunque se puede obtener el pdf gratuitamente de: <http://ambermd.org/doc11/>

Avogadro

Nombre Avogadro

Creador/es - Propietarios

Avogadro, como muchos proyectos de código abierto está ayudado por un gran elenco. Son fundamentalmente un proyecto comunitario.

Código:

Ali Shahzad, Michael Banck, Fox Naomi, Marcus Hanwell, Geoff Hutchison, Benoît Jacob, David Lonie, Jordan Mantha, Thomas Margraf, Carsten Niehaus, Simon Ochsenreither, Tim Vandermeersch

Todos los traductores de Avogadro

También:

Grupo Obelisco Azul, Boost.Python, Estructuras químicas de base de datos, Cmake, Eigen, GL2PS, Jason Harris para el KPlotWidget KDE, Open Babel, Qt Software

Sistemas Operativos

Unix/Linux, Windows

Posibilidad de implementación en otras plataformas.

Versión 1.0.0

Fecha inicio del proyecto/programa inicio proyecto: 2006-04-14

Primera versión: Febrero 2007

Fecha última modificación actualización del proyecto 2010

Idiomas Chino, francés, alemán, italiano, ruso, español y más de 12 lenguas en desarrollo.

Licencia GNU General Public License (GPL)



Coste Gratuito.

Formatos de entrada

CML (*.cml), Salida química computacional (*.out, *.log, *.dat, *.output), Intercambio Cristalográfico CIF (*.cif), Salida de Gaussian 98/03 (*.g98, *.g03), Fichero Checkpoint de Gaussian formateado (*.fchk), HyperChem (*.hin), Mol de MDL (*.mdl, *.mol, *.sd, *.sdf), Salida de NWChem (*.nwo), PDB (*.pdb, *.ent), Mol2 (*.mol2), XYZ (*.xyz), formatos ACR, ADF, ADF TAPE41, Alchemy, Amber, MSI BGF, Dock 3.5, Cacao Cartesian, Accelrys/MSI Biosym/Insight II CAR, Fractional libre, GAMESS, Gaussian, Ghemical, HiperChem HIN, InChI, CCC, ChemDraw, Chem3D Cartesian 1 y 2, Crystallographic, Chemical Markup Language, Chemical Resource Kit, Dmol3, FastSearching, Jaguar, MCDL/SDF, Molden, Molpro, MOPAC, MPQC, OpenDX, APBS, PCModel, PNG, PQR, PubChem, Q-Chem, ShelX, SMILES, Thermo, TurboMole, VASP, ViewMol, XML, YOB, *.Pam7, Además permite la carga desde: directorio local, directamente de base de datos química PDB, o desde la red por la URL.

Formatos de salida / Tipos creables

PNG, BMP, JPEG, PDF, SVG, EPS, POV-RAY, CML, XYZ, Entrada para GAMESS (*.gamin), Entrada para Gaussian en cartesianas (*.gau) y en matriz-Z (*.gzmat), Archivo Mdfile de MDL (*.mol), PDB, Entrada para NWChem (*.nw), Mol2 de Sybyl (*.mol2), *.ent, *.alc, *.chm, *.cdx, *.cdxml, *.c3d1, *.c3d2, *.gpr, *.mdl, *.sdf, *.sd, *.crk3d, *.cht, *.dmol, *.bgf, *.gam, *.inp, *.gamout,

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

*.tmol, *.fract, *.mpd, *.mwo

Dependencias

Requisitos

Pedidos por el creador *No especificados*

Espacio en disco 48,4 MB

Memoria 65MB

Lenguaje Python, C++ (Qt)

Características:

Facilitadas por el proyecto:

- Libre, de código abierto.
- Intuitivo, hecho para ser fácil de manejar tanto para estudiantes como para investigadores avanzados.
- Rápido: Soporte a renderizado de multi-hilos y computación.
- Extensible: Arquitectura de plug-in para el desarrollo, incluyendo renderizado, herramientas interactivas, comandos, y scripts en Python.
- Flexible: Las características incluyen la importación Open Babel de archivos químicos, generación de entrada para paquetes químicos multi computación, cristalografía, y biomoléculas.
- Construcción de mecánica molecular (incluido MMFF94 y UFF)
- Generación de entrada para Gaussian, GAMESS-US, Molpro, NWChem, Q-Chem, MOPAC, con muchos más paquetes por incluir pronto.
- Soporte para unidad celulares cristalográficas.
- Constructor Oligopeptido
- Visualización de isosuperficies y orbitales, incluyendo cubos Gaussianos, OpenDX, y ficheros fchk Gaussian.
- Animación de vibraciones moleculares computarizadas y trayectorias de reacción.
- Visor avanzado espectral con ampliación y ensanche.
- Exportación de modelos a POV-Ray.

También, se ofrece las librerías de Avogadro (libavogadro), el cual puede ser usado para el desarrollo de software.

Representación en alámbrico	✓
Representación CPK en alámbrico	✓
Representación en palo y bola	✓
Representación en palo	✓
Representación CPK	✓
Calculo distancia entre átomos	✓
Calculo de ángulo entre átomos	✓
Calculo de ángulo dihedral	✓
Calculo de energía intermolecular	✓
Calculo de energía	✓
Calculo del gradiente relativo	✓

Diseñado principalmente para

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Avogadro es un editor molecular avanzado diseñado para un uso multiplataforma en química computacional, modelado molecular bioinformática, ciencias materiales y áreas relacionadas. Ofrece un marco de representación flexible y una arquitectura de plug-in de gran alcance.

Resumen

Pros Fácil de instalar, intuitivo, rápido, flexible y extensible. Tutorial y foro actualizados. Multi-plataforma, multi-idioma, y multi-formatos.

Contras No especifica el uso directo de GPU como procesador, al menos sin extensión.

Comentarios / Problemas Un programa liviano y muy destacable.

BOSS

Nombre BOSS (Biochemical and Organic Simulation System)

Creador/es - Propietarios William L. Jorgensen Research Group, Departamento de Química, YALE University

Sistemas Operativos en los que está implementado UNIX/Linux, Windows, SGI

Versión 4.6

Fecha inicio del proyecto/programa 1970

Fecha última modificación actualización del proyecto 2010

Idiomas Español (parcial), Inglés

Licencia Derechos de autor y uso.

Coste para empresas con animo de lucro

UNIX, O Windows → 12,5 \$

UNIX Y Windows → 15 \$

Código de predicción de código (PROP) → 1 \$

Directorio de estructuras de drogas → 1 \$

Actualizaciones → 2 \$

para escuelas y agencias gubernamentales

UNIX O Windows → 1,2 \$

UNIX Y Windows → 1,5 \$

PROP → 200 \$

Actualizaciones → 250 \$

Formatos de entrada *no comprobados*

PDB, PDB2, PDBB → (ChemEdit, RasMol, Sybyl, MacroModel, Midas)

MDLMO (o MOL), MDLSD → (Molfile)

MIND → (MindTool)

Formatos de salida / Tipos creables *no comprobados*

PDB, PDB2, PDBB → (ChemEdit, RasMol, Sybyl, MacroModel, Midas)

MDLMO (o MOL), MDLSD → (Molfile)

MIND → (MindTool)

Dependencias FORTRAN

Requisitos

Pedidos por el creador

mínimos CPU 200MHz

recomendados CPU 2.4GHz, RedHat 8 o superior (kernel 2.0 o superior); Windows 9x, 2000, XP

Espacio en disco *no especificados/comprobados*

Memoria *no especificados/comprobados*

Lenguaje C/C++

Características: *no comprobados*

- Simulaciones Monte Carlo, se emplea el conjunto del TNP, aunque simulaciones NVT también se puede realizar al no permitir los cambios de volumen.
- Preferencia de muestreo, se utiliza según el cual los solutos y moléculas

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

cercanas del disolvente se mueven con más frecuencia que las moléculas del disolvente distantes. La polarización se basa en $1 / (r^2 + WKC)$ donde WKC es una constante.

- Las coordenadas de soluto pueden ser proporcionados por entrada Z-matriz o PDB y coordenadas internas se utilizan para tomar muestras de la geometría del soluto.
- Cambios de energía libre se calcula a partir de la teoría de perturbación estadística (FEP).
- El unida-átomo OPLS y funciones de potencial de todos los átomos se utilizan con referencias que figuran en los archivos de parámetros. La cobertura incluye numerosos grupos funcionales orgánicos, todos los residuos comunes péptidos y nucleótidos. Más parámetros se pueden agregar fácilmente en el archivo de parámetros, parámetros inéditos incluso para heterociclos, sulfamidas y otros grupos importantes farmacológicamente también se proporcionan a los usuarios de BOSS.
- Por otra parte, la energética del soluto (s) puede ser descrito con el AM1, PM3, PDDG/PM3 o PDDG / MNDO métodos semiempíricos de orbitales moleculares. Cualquier interacción solvente-solvente se describen con los cálculos potenciales de OPLS mixta QM / MM. Las interacciones soluto-solvente se describen con el campo de fuerza usando parámetros de OPLS norma-AA Lennard-Jones para el soluto (s) y cargos obtenidos a partir de los procedimientos CM1 o CM3 o como cargas de Mulliken.
- Ángulos diedros, los ángulos de enlace y la longitud de enlace en los solutos pueden ser detallados durante las simulaciones. Las moléculas del disolvente también puede ser totalmente flexible.
- Restricciones de armónicos pueden ser incluidas entre los pares de átomos de soluto o entre un átomo y un punto fijo. Esto, no sólo facilita algunos cálculos de energía libre para acoger los sistemas de evaluación, sino que también se puede utilizar para optimizaciones y refinado de estructuras RMN.
- Los archivos de salida puede ser en formato PDB o MDL mol para la visualización fácil con los programas estándar de gráficos moleculares.

Diseñado principalmente para

El programa BOSS realiza (a) simulaciones mecánica estadística Monte Carlo (MC) para las soluciones de cero a 25 moléculas de soluto en una caja de solvente periódico, en un grupo solvente, o en un dieléctrico continuo, incluida la fase de gas, y (b) minimización de la energía estándar, análisis de modo normal, y la conformación de la búsqueda. Las energías se representan con los campos de fuerza OPLS o el AM1, PM3, PDDG/PM3 y PDDG / MNDO métodos semiempíricos de orbitales moleculares. Además de los resultados cuantitativos estructurales y energéticas, otras cantidades se calculan incluyendo los momentos dipolares, cargas atómicas CM1 y CM3, solvente las superficies accesibles, y cuenta con enlaces de hidrógeno, que son valiosos para muchas aplicaciones, como las actividades de relaciones de estructura.

Resumen

Pros Muy especializado en cargas, optimización, mecanismos moleculares, mecanismos cuánticos, soporte al desarrollo de campo de fuerzas, para diversos sistemas operativos.

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Contras Aun siendo de los que menos cuesta, no es gratuito, no tiene una gran extensión para compatibilidad con otros programas. El manejo no es intuitivo, no está muy documentado, no tiene interfaz gráfica (por lo que no se puede editar con ratón).

Comentarios / Problemas

Al no ser gratuito y no facilitar ninguna versión para pruebas ni testeó aun solicitándolo, no se instaló y probó en un computador.

Maestro

Nombre Maestro

Creador/es - Propietarios Schrödinger, L.L.C.

Sistemas Operativos en los que está implementado Linux (32 o 64 bits);
Windows (2000, XP, Vista (32 64 bits); IRIX

Versión v9.0.211

Fecha inicio del proyecto/programa 2006

Fecha última modificación actualización del proyecto junio 2009

Idiomas Inglés

Licencia Copyright (c) Schrödinger, LLC. All rights reserved.

Coste licencia académica → gratuita

licencia normal → No disponible en su web, hay que pedirla por mail.

Formatos de entrada PDB (.pdb,ent), Maestro (.mae, maegz, mae.gz), MOL2 (.mol2), MDL SD (.sd, .sdf, .mol, .sdfgz, .sdf.gz), Jaguar Input/Output (.in, .out), ReagentPrep (.bld), MacroModel (.dat, .out), MCPRO Input (.zmat), Desmond (.cms, cmsgz, cms.gz), BioGraf (.bgf), GAMESS Output (.gamin), Gaussian Input/ 92 Output/ 94 Output (.com, .log), Mopac Cartesian/Internal/Output (.mopcart, .mopint, .out), Spartan (.scar), Spartan Mol. Mechanics/ Semi-Empirical, XYZ (.xyz)

Formatos de salida / Tipos creables Proyecto Maestro (.prj), Imagen (.tiff, .jpg), Maestro (.mae, .maegz, .mae.gz), PDB (.pdb, .ent); SDF (.sdf, .sd, .mol, .sdfgz, .sdf.gz); Mol2 (.mol2), MacroModel (.dat, .out), GAMES Output (.gamin), Gaussian Cartesian (.cart), Gaussian Z-Matrix (.g), Mopac Cartesian (.mopcart), Mopact Internal (.mopiny), Spartan (.spar), XYZ (.xyz)

Dependencias Para Linux: Perl, gunzip, Python 3, nfs-utils, glibc 2.5, nss_ldap, servidor X, GLX OpenGL, OpenGL, fontconfig 2.2.1-13
Para Windows: Visual C++ 2005 SP1

Requisitos

Pedidos por el creador

mínimos 256 MB de memoria principal; 4 GB de disco duro dedicado a 10000 RPM (si sólo se instala Maestro son 109 MB); suficiente memoria cache; monitor de 16-bits colores en 1280 x 1024; OpenGL.
Para Linux: Perl no inferior a versión 5.004; gunzip; Python 3.0 (32-bit); tarjeta de red.

Para Linux x86 32 bits: nfs-utils; kernel 2.4.21 en series 2.4; 2.6.9 en series 2.6; procesador x86-compatible

Para Linux x86-64 bits: nfs-utils; glibc 2.5; nss_ldap (en RHEL 5.2 y entOS 5.2); procesador compatible x86 64-bit; servidor X X11R6; GLX OpenGL y OpenGL; fontconfig 2.2.1-13; Firefox; lector de PDF

Para Linux ia64: procesador Itanium 2

Para Windows: Visual C++ 2005 SP1; Internet Explorer; lector de PDF

recomendados 1 GB de memoria principal; 60 GB de disco duro dedicado a 10000 RPM mínimo; monitor con frecuencia de refresco 100 Mhz; para visualización en estéreo: tarjeta gráfica con soporte quad-

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

buffered stereo, monitor con más de 100 MHz de refresco.

Para Linux: Perl no inferior a versión 5.004; gunzip; Python 3.0 (32-bit); tarjeta de red; servidor X X11R6; GLX OpenGL y OpenGL; fontconfig 2.2.1-13; Firefox; lector de PDF

Para Linux x86 32 bits: glibc; nfs-utils; kernel 2.4.21 en series 2.4; 2.6.12 en series 2.6; procesador x86-compatible

Para Linux x86-64 bits: nfs-utils; glibc 2.5 o superior; nss_ldap (en RHEL 5.2 y entOS 5.2); procesador compatible x86 64-bit

Para Linux ia64: procesador Itanium 2

Para Windows: Visual C++ 2005 SP1; Internet Explorer; lector de PDF

Espacio en disco 833 MB

Memoria 457 MB

Subprocesos 21

Lenguaje Python, C, Java.

Características:

El formato es Maestro de libre formato de texto ASCII y es sustancialmente autodescriptivos.

La versión actual proporciona la capacidad para importar, exportar, construir y modificar las estructuras moleculares, para mostrar las estructuras en diversas formas; crear, iniciar y supervisar CombiGlide, Epik, Glide, Jaguar, de impacto, de enlace, LigPrep, Macromodel, MCPRO +, Fase, el primer, Primex, QikProp, QSite, Mapa del Sitio, y los cálculos huelga y llevar a cabo diversos análisis.

Las características de Maestro incluyen:

Lectura / escritura de archivos de estructuras

Construcción de estructuras extensible

Capacidad de añadir / borrar hidrógeno (basado en la noción de un tratamiento de hidrógeno)

Ejecución de secuencias/guiones de comandos de Python y el lenguaje de comandos Maestro

Comando de aliasing (similares a los alias de shell UNIX)

Edición de guiones de comandos (le permite leer, escribir, modificar y ejecutar guiones de comandos de Maestro)

Macros (asignar una secuencia de comandos a una macro con nombre o con las teclas F1 ... F12)

Deshacer (un nivel)

Etiquetado y coloreado de átomos

Elección de representación molecular (CPK, Ball & Stick, etc)

Alta calidad de salida PostScript

Varios paneles para crear, iniciar y supervisar el trabajo

Medición - distancia, ángulo, diedro

La superposición de estructuras

Fuerza del campo de visión (para examinar todas las interacciones de campos de fuerza presentes en un cálculo determinado de una estructura)

Trazado 1D/2D (este último para trazado Ramachandran, ambos para uso con cálculos dihedrales)

Ayuda online (que dispone de diversos mecanismos para localizar el tema de ayuda adecuado - buscar por nombre, por categorías, temas relacionados, etc)

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Ayuda automática (similar a los globos de ayuda en el Mac o herramientas bajo Windows)

Ayuda contexto-sensitiva (que proporciona consejos para la realización de tareas a el usuario)

Diseñado principalmente para

Maestro es la interfaz gráfica de usuario para casi todos los productos que Schrödinger distribuye: CombiGlide, ConfGen, Desmond, Epik, Glide, Impact, Jaguar, Liaison, LigPrep, Macro-Model, MCPRO+, Phase, Prime, PrimeX, QikProp, QSite, SiteMap, Strike, y WaterMap.

Contiene herramientas para construir, mostrar y manipular las estructuras químicas, para la organización, carga y almacenamiento de estas estructuras y los datos asociados, y para la creación, seguimiento, y visualización de los resultados de los cálculos de estas estructuras.

Maestro es una interfaz gráfica de usuario de propósito general (GUI) para modelado molecular 3D. Se proporcionan herramientas para generar y mostrar los modelos moleculares, a partir de los cálculos y el seguimiento de estos modelos, y el análisis y la manipulación de estos modelos.

El trabajo de control de Maestro fácilmente gestiona trabajos presentados desde Maestro y desde la línea de comandos para tanto máquinas locales como remotas. Maestro se puede ejecutar localmente y enviar los trabajos a cualquier host que tenga acceso. La interfaz de Maestro utiliza las herramientas de gráficos OpenGL, y puede tomar ventaja de las capacidades de hardware de gráficos, incluyendo la capacidad de visualización estéreo.

Resumen

Pros Es uno de los más completos, y abarca todo el campo de especialidades, drogas, química, bioquímica. Dispone de: Gráficos moleculares, capacidad de dibujar moléculas con el ratón, construcción de polímeros, construcción de ácido nucleico, construcción de peptidos, construcción de cristales, Adicción de solventes, manejo de cargas parciales, docking, optimización, mecánica molecular, mecanismos cuánticos,.soporte al desarrollo de campo de fuerzas.

Tiene mucha documentación. Está para diversos sistemas operativos. Aunque la versión de Maestro es de licencia gratuita, es sólo de uso académico, para cualquier otro programa del paquete o otro uso de este, hay que solicitar una licencia por correo.

Contras Debería tener más extensiones de entrada y salida. No especifica el uso directo de GPU para aceleración gráfica.

Comentarios / Problemas

Cuando me registré para obtener la versión académica, lo hice sin especificar el tipo de uso, con lo que a la hora de la descarga sólo me daba acceso a la versión standard, mandando un correo a help@schrodiner.com me respondieron la misma semana y me facilitaron una licencia académica explicando detalladamente los pasos que debía realizar para que funcionara la versión que descargue. Son distintas, la versión normal ocupa 20 MB más que la académica.

BALLView

Nombre BALLView

Creador/es - Propietarios El desarrollo de BALL empezó en el Max Planck Institute for Computer Science en el grupo de Kurt Mehlhorn bajo la supervisión de Hans-Peter Lenhof

Actualmente está siendo mantenido en el Center for Bioinformatics Saar y el Zentrum für Bioinformatik Tübingen por el grupo de Oliver Kohlbacher, Andreas Hildebrandt, y Hans-Peter Lenhof.

A continuación la lista completa de codificadores actuales y anteriores: Andreas Hildebrandt, Anna Katharina Dehof, Stefan Nickels, Andreas Moli, Oliver Kohlbacher, Hans-Peter Lenhof, Tim Aubertin, Christian Bender, Andreas Bertsch, Lisa Katharina Blaß, Nicolas Pascal Boghossian, Andreas Burchardt, Andreas Crauser, Anna Dehof, Holger Franken, Jan Fuhrmann, Carla Haid, Silke Jansen, Heiko Klein, Andreas Kerzmann, Bettina Leonhardt, Andreas Moll, Peter Müller, Alexander Rurainski, Stefan Schuh, Daniel Stöckel, Stefan Strobel, Marc Sturm, Nora Toussaint, Hongbo Zhu,

Sistemas Operativos en los que está implementado BALL está disponible para descargar bajo varias plataformas. Disponen para la descarga de los códigos fuentes, o de los binarios para BALLView (Windows y MacOS X).³

Versión 1.3.2

Fecha inicio del proyecto/programa 1996

Fecha última modificación actualización del proyecto 11 Febrero 2010

Idiomas Inglés

Licencia Copyright (C) Andreas Hildebrandt, Oliver Kohlbacher, Hans-Peter Lenhof, and others 1996-2010 (BALL está bajo LGPL, ViewBALL con GPL)

Coste gratuito

Formatos de entrada .pdb, .ent, .brk, .hin, .mol, .mol2, .sdf, .xyz, proyecto (.bvp)

Formatos de salida / Tipos creables .pdb, .ent, .brk, .hin, .mol, .mol2, .sdf, .xyz, imágenes (.png)

Dependencias No requiere instalaciones ni configuraciones previas.

Requisitos

Pedidos por el creador *no especificados*

Espacio en disco 190.9 MB

Memoria 53 MB

Subprocesos 6

Lenguaje C++, Qt 4.5.1, Phyton

Características:

- Gráficos moleculares (3D, visión estereoscópica)
- Dibujo interactivo molecular y la edición conformacional

³Probado en: Linux/x86 2.x usando g++ 4.1.2 , Linux/x86_64 2.x usando g++ 4.1.2 , Solaris/SPARC 8 usando g++ 3.2.1 , Solaris/SPARC 8 usando g++ 2.95.3 , Solaris/SPARC 8 usando SUN Forte Developer 7 C++ 5.4 2002/03/09 , IRIX 6.5 usando CC 7.3.1.1m (32 y 64 bit) , Microsoft Windows XP usando Microsoft Visual Studio .NET (MSVC 7.0)

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

- Lectura y escritura de formatos de archivo molecular (AP, MOL2, MOL, HIN, XYZ, KCF, SD)
- Fuentes de datos secundarios, por ejemplo, (DCD, DSN6, GAMESS JCAMP, SCWRL, TRR)
- SMILES / SMARTS
- Optimización de geometría
- Minimizador y clases de dinámica moleculares
- Apoyo a la fuerza de campos, Force Fields, (MMFF94, AMBER, CHARM) para la minimización de la puntuación y la energía
- Funcionalidad Scripting
- Plug-in
 - Representación en alámbrico ✓
 - Representación CPK en alámbrico ✓
 - Representación en palo y bola ✓
 - Representación en palo ✓
 - Representación CPK ✓
 - Calculo distancia entre átomos ✓
 - Calculo de ángulo entre átomos ✓
 - Calculo de ángulo dihedral ✓
 - Calculo de energía intermolecular ✓
 - Calculo de energía ✓
 - Calculo del gradiente relativo ✓

Diseñado principalmente para

BALL ha sido diseñado para un uso sencillo, robusto, de código abierto, y que proporcionara una amplia funcionalidad.

BALL ha sido diseñado cómo una herramienta para el desarrollo rápido de prototipos de software en el modelado molecular y diseño de fármacos. Se empezó a desarrollar en 1996, inicialmente como un conjunto de herramientas para el acoplamiento de proteína-proteína. Rápidamente se convirtió en un gran conjunto que cubre una amplia gama de aplicaciones

Resumen

Pros LGPL, gratuito, simple, código abierto, salida a vmrl, ampliable por plug-in, posibilidad de creación de scripts por Python

Contras podía tener mayor número de formatos químicos.

Comentarios / Problemas

Rápido y fácil de instalar y usar.

CAPITULO 4: Uso de ARToolKit e inclusión de moléculas

4.1 Introducción

A continuación, vamos a explicar cómo obtener una aplicación de RA con moléculas. Para ello expondremos de forma detallada tanto el funcionamiento de la aplicación como la creación de moléculas y la integración en la aplicación. Para ello, presentaremos todas las herramientas y dispositivos utilizados para la creación y ejecución, así como los requisitos técnicos mínimos para que todo funcione correctamente.

4.2 Análisis de las herramientas a utilizar

Para realizar una aplicación de RA vamos a usar:

- ARToolKitVRML una modificación de ARToolKit versión 2.65
- Microsoft Visual Studio 2005
- Un programa de edición molecular, en este caso BALLView
- DirectX anterior a 2006 o WinSDK con las cabeceras y herramientas de DirecShow (DirecShowLib y DirecShowSamples).

Explicaremos para qué se utiliza cada herramienta.

4.2.1 ARToolKitVRML

ARToolKit es una librería que permite la creación de aplicaciones de RA, en las que se sobrepone imágenes virtuales al mundo real. Para ello, utiliza las capacidades de seguimiento de vídeo, con el fin de calcular, en tiempo real, la posición de la cámara y la orientación relativa a la posición de los marcadores físicos. Una vez que la posición de la cámara real se conoce, la cámara virtual se puede colocar en el mismo punto y modelos 3D son superpuestos exactamente sobre el marcador real. Así ARToolKit resuelve dos de los principales problemas en la RA, el seguimiento de punto de vista y la interacción objeto virtual

ARToolKit fue desarrollado originalmente por Kato en 1999 en el HIT Lab de la Universidad de Washington. Actualmente se mantiene como un proyecto de código abierto alojado en SourceForge con licencias comerciales disponibles en ARToolWorks. ARToolKit es una herramienta para desarrollar aplicaciones de RA muy utilizada, con más de 160.000 descargas desde el año 2004.

ARToolKitVRML es una modificación de ARToolKit para poder usar directamente objetos VRML (sigla del inglés Virtual Reality Modeling Language). VRML es un formato de archivo normalizado que tiene como objetivo la representación de escenas u objetos interactivos tridimensionales.

Al igual que ARToolKit, proporciona una serie de funciones para inicializar el Tracker que se encargará de buscar las marcas que capte la cámara. Los pasos que hay

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

que seguir son los siguientes:

- Cargar los parámetros de la cámara. En nuestro caso, hemos usado el archivo por defecto que incluye ARToolKit: camera_para.dat.
- Actualizar los parámetros del Tracker respecto al tamaño de la imagen que debe analizar.
- Inicializar los parámetros de la cámara.
- Obtener la matriz de proyección que usaremos para dibujar las imágenes.
- Cargar los patrones o marcas a buscar.

4.2.1 Microsoft Visual Studio 2005

Microsoft Visual Studio es un IDE (entorno de desarrollo integrado) para sistemas Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET, y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

Visual Studio permite a los desarrolladores crear aplicaciones, sitios y aplicaciones Web, así como servicios Web en cualquier entorno que soporte la plataforma .NET (a partir de la versión NET 2002). Así se pueden crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas Web y dispositivos móviles.

Visual Studio 2005 se empezó a comercializar a través de Internet a partir del 4 de Octubre de 2005 y llegó a los comercios a finales del mes de Octubre en inglés. En castellano no salió hasta el 4 de Febrero de 2006. Microsoft eliminó .NET, pero eso no indica que se alejara de la plataforma .NET, de la cual se incluyó la versión 2.0 de la máquina virtual Java.

La actualización más importante que recibieron los lenguajes de programación fue la inclusión de tipos genéricos, similares en muchos aspectos a las plantillas de C#. Con esto se consigue encontrar muchos más errores en la compilación en vez de en tiempo de ejecución, incitando a usar comprobaciones estrictas en áreas donde antes no era posible. C++ tiene una actualización similar con la adición de C++/CLI como sustituto de C# manejado.

Se incluye un diseñador de implantación, que permite que el diseño de la aplicación sea validado antes de su implantación. También se incluye un entorno para publicación Web y pruebas de carga para comprobar el rendimiento de los programas bajo varias condiciones de carga.

Visual Studio 2005 también añade soporte de 64-bit. Aunque el entorno de desarrollo sigue siendo una aplicación de 32 bits Visual C++ 2005 soporta compilación para x86-64. El SDK incluye compiladores de 64 bits así como versiones de 64 bits de las bibliotecas.

Visual Studio 2005 tiene varias ediciones radicalmente distintas entre sí:

«Express», «Standard», «Professional», «Tools for Office», y «5 ediciones

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Visual Studio Team System». estas últimas se proporcionaban conjuntamente con suscripciones a MSDN cubriendo los 4 principales roles de la programación: «Architects», «Software Developers», «Testers», y «Database Professionals». La funcionalidad combinada de las 4 ediciones «Team System» se ofrecía como la edición «Team Suite».

«Tools for the Microsoft Office System» está diseñada para extender la funcionalidad a Microsoft Office.

Las ediciones Express se han diseñado para principiantes, aficionados y pequeños negocios, todas disponibles gratuitamente a través de la página de Microsoft se incluye una edición independiente para cada lenguaje: Visual Basic, Visual C++, Visual C#, Visual J# para programación .NET en Windows, y Visual Web Developer para la creación de sitios Web ASP.NET. Las ediciones Express carecen de algunas herramientas avanzadas de programación así como de opciones de extensibilidad.

Se lanzó el «Service Pack 1» para Visual Studio 2005 el 14 de Diciembre de 2006.

La versión interna de Visual Studio 2005 es la 8.0, mientras que el formato del archivo es la 9.0.

4.2.3 BALLView

BALL (Biochemical Algorithms Library) es un marco de desarrollo de algoritmos y estructuras de datos extensible y de código abierto en C++ para el modelado molecular y bioinformática estructural computacional. La biblioteca también ofrece una interfaz de secuencias de comandos Python. Entre los sistemas soportados son Linux, Solaris, Microsoft Windows y MacOS X. La biblioteca se complementa con las utilidades de línea de comandos y admite una visualización con Qt y OpenGL también.

Hay un visor molecular BALLView desarrollados por el mismo equipo, lo que permite ver y editar varios formatos de archivo molecular, por ejemplo, AP, HIN, MOL2 y muchos más. Es el componente de visualización de BALL. Ambos, BALL y BALLView, están disponibles bajo licencias LGPL y GPL. Los programas son desarrollados y mantenidos por los grupos de Hans-Peter, Kohlbacher e Hildebrandt. BALLView es una aplicación escrita en C++ que utiliza BALL para el modelado molecular y visualización de modelos moleculares. Está disponible bajo la licencia GPL para Linux, Windows y Mac OS.

4.2.4 DirectX, DirectShow

DirectX es una colección de API creadas y recreadas para facilitar las complejas tareas relacionadas con multimedia, especialmente programación de juegos y vídeo en la plataforma Microsoft Windows.

DirectX consta de las siguientes APIs:

- Direct3D: utilizado para el procesamiento y la programación de gráficos en tres dimensiones (una de las características más usadas de DirectX).

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

- Direct Graphics: para dibujar imágenes en dos dimensiones (planas), y para representar imágenes en tres dimensiones.
- DirectInput: para procesar datos del teclado, mouse, joystick y otros controles para juegos.
- DirectPlay: para comunicaciones en red.
- DirectSound: para la reproducción y grabación de sonidos de ondas.
- DirectMusic: para la reproducción de pistas musicales compuestas con DirectMusic Producer.
- DirectShow: para reproducir audio y vídeo con transparencia de red.
- DirectSetup: para la instalación de componentes DirectX.
- DirectCompute: lenguaje e instrucciones especiales para el manejo de cientos o miles de hilos de procesamiento, especial para procesadores de núcleos masivos.

A pesar de ser desarrollado exclusivamente para la plataforma Windows, se está desarrollando una implementación de código abierto de su API para sistemas Unix (en particular Linux) y X Window System por el proyecto WineHQ, del cual existe fork propietario, Cedega, desarrollada por la empresa de software Transgaming y orientada a la ejecución de juegos desarrollados para Windows bajo sistemas Unix.

DirectX es necesario para que al compilar ArtoolKit, este pueda utilizar las librerías de DirectShow para Windows.

A partir de la versión DirectX 8.0 (actualmente va por la 11) DirectShow dejó de incluirse como parte del paquete, sólo se obtenía en la SDK. En posteriores versiones, no se incluye en DirectX, paso a formar parte de Windows SDK.

DirectShow (a veces abreviado como DS o DShow), nombre en clave de Quartz, es un framework multimedia API y producido por Microsoft para los desarrolladores de software para realizar varias operaciones con archivos multimedia o stream. Es el reemplazo para la tecnología de vídeo anterior de Microsoft Windows. Sobre la base de Microsoft Windows Component Object Model (COM) Marco, DirectShow proporciona un interfaz común para los medios de comunicación a través de muchos lenguajes de programación, y es extensible, un marco basado en filtros, que puede hacer o grabar archivos de medios en demanda, a petición del usuario o desarrollador. Las herramientas de desarrollo de DirectShow y la documentación se distribuyeron originalmente como parte del SDK de DirectX. Actualmente, se distribuyen como parte del SDK de Windows (anteriormente conocido como plataforma SDK).

Las contrapartes de DirectShow en otras plataformas constituyen un marco de QuickTime de Apple y los diversos marcos multimedia GStreamer como Linux o Xine. Microsoft planea reemplazar por completo DirectShow gradualmente con la Fundación de Medios de Windows en versiones futuras. Las aplicaciones de Windows Vista y Windows 7 utilizan los medios de comunicación de la Fundación en lugar de DirectShow para varios medios de comunicación y tareas relacionadas.

Si se quiere o se tiene instalada la última versión de DirectX, la alternativa que

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

nos queda es instalar además Windows SDK antes conocido como Plataforma SDK

El Microsoft Windows SDK (Software Development Kit) es un kit de desarrollo de software de Microsoft que contiene archivos de cabecera, bibliotecas, muestras, documentación y herramientas que utilizan las API necesarias para desarrollar aplicaciones para Microsoft Windows y .NET Framework 3.0/.NET Framework 3.5. El actual Windows SDK se puede utilizar para escribir aplicaciones que se dirigen a Windows XP y Windows Server 2003, así como Windows Vista y Windows Server 2008, mientras que el más antiguo Platform SDK puede ser utilizado para desarrollar aplicaciones para Windows 2000, Windows XP y Windows Server 2003. El Windows SDK sustituye al Platform SDK, al WinFX SDK y al .NET Framework 2.0 SDK. El paquete contiene amplia documentación y cerca de 1000 ejemplos. A pesar de que el propio SDK está disponible de forma gratuita, la obtención de los SDK requiere que la copia de Windows esté validada.

4.3 Trabajando con Microsoft Visual Studio 2005, BALLView y ARToolKit

En este apartado, nos centraremos en aspectos de implementación del proyecto.

4.3.1 La cámara

Uno de los problemas más importantes a los que nos enfrentamos a la hora de realizar este proyecto era hacer funcionar la cámara, ya no tanto para ARToolKit en concreto, sino para la integración que se hizo con otros entornos de RA como eran NyARToolKit, que necesitaba la integración y el acceso a cámara desde aplicaciones independientes a Windows, como en Java o Python.

Para ello se instaló «JRE System Library», haciendo previamente una desinstalación de «Systems Library de JMF». Con ello conseguimos instalar las librerías para Java, y se comprobó su correcto uso y configuración (resolución, tamaño, profundidad, brillo y contraste) con la aplicación JMStudio (conjunto de API's y herramientas dedicadas al multimedia stream).

Para poder usar correctamente la cámara con aplicaciones de Windows simplemente se instalaron drivers propietarios.

Para el uso de la cámara por parte de ARToolKit fue necesaria la actualización de DirectX que había en el sistema y la instalación de DirectX SDK y Windows SDK, seleccionando en el instalador las librerías, herramientas y cabeceras para DirectShow.

4.3.2 Creación de Moléculas

Para la creación de moléculas podemos usar programas tan extensos como Maestro o programas tan simples como Abalone, pero lo más eficaz es utilizar aplicaciones como BALLView, que permiten usar directamente la API o integrar el programa mediante plug-ins. Además permite la salida directa de molecular en formato VRML.

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Para la elección y creación de una molécula nos basamos en una molécula compleja y de fácil identificación como es un trozo de secuencia de ADN. Para crearla se optó por buscar su formulación en una base de datos www.rcsb.org en su forma de PDB.

Una vez creamos la composición molecular la recuperamos para ARToolKit y seguimos trabajando desde Microsoft Visual Studio 2005 para crear la aplicación.

4.3.3 Búsqueda de marcas: Cálculo de la matriz de transformación

En este punto describiremos el proceso mediante el cual la biblioteca ARToolKit es capaz de obtener el punto de vista del usuario mediante el cálculo de la matriz de transformación.

Una de las labores más complicadas en el desarrollo de una aplicación de RA es, precisamente, el cálculo del punto de vista del usuario en tiempo real a la vez que las imágenes virtuales aparecen perfectamente alineadas con los objetos reales del mundo. En la figura 44 se puede ver una representación de como se transforma un punto que se capta por la cámara (en 3D) a un punto del monitor (en 2D).

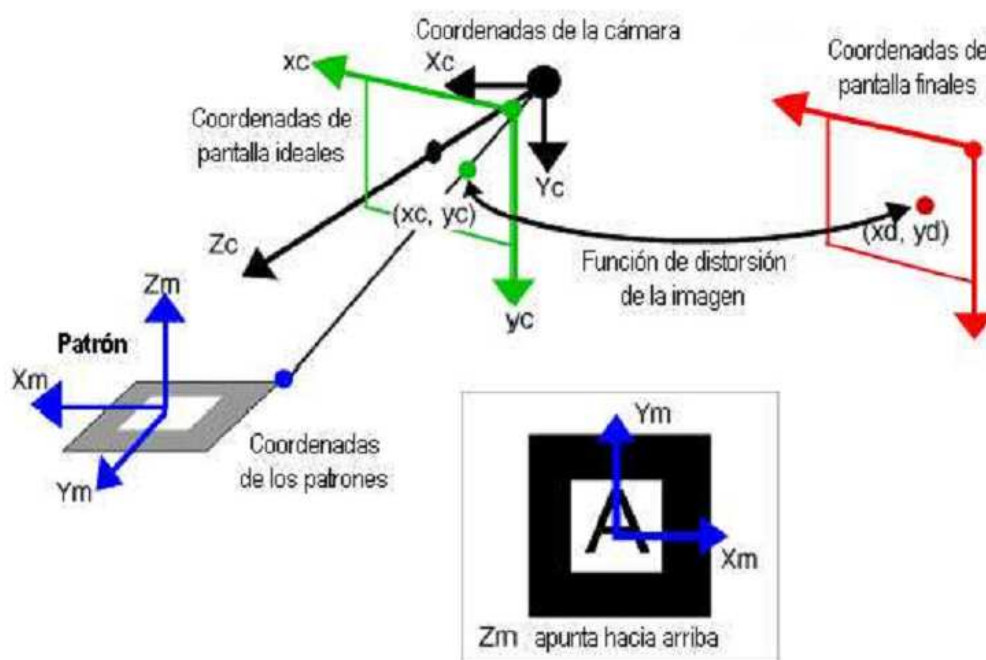


Fig. 44: Cálculo de la matriz de coordenadas de la cámara a partir de las coordenadas del patrón

Al estar utilizando una cámara, se debe tener en cuenta el factor de distorsión producido por la lente de esta y corregirlo (Figura 45). ARToolKit ofrece, por defecto, un fichero de calibración de la cámara o bien un programa para obtener nuestra propia calibración. Mediante esta calibración realiza las operaciones para la corrección de dicha distorsión y obtener las coordenadas de la matriz para la cámara virtual, consiguiendo así dibujar los objetos correctamente.

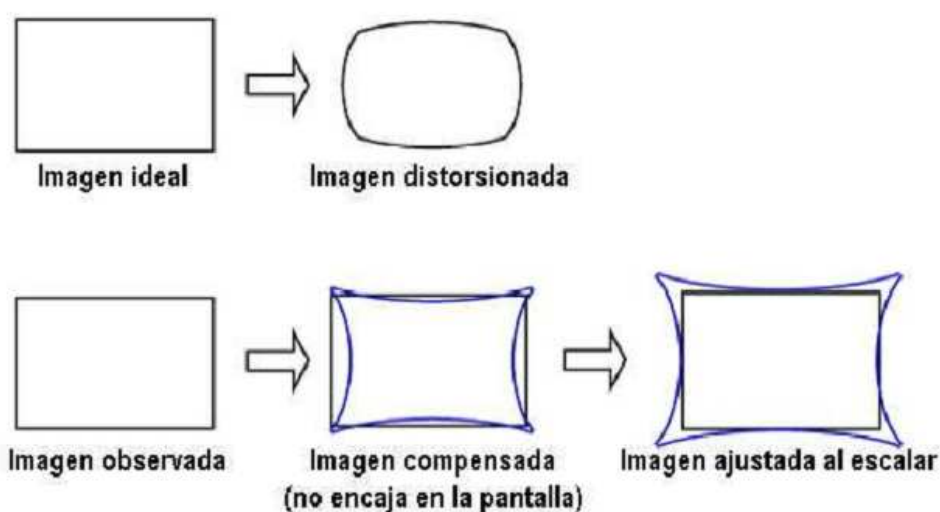


Fig. 45: Distorsión producida por la lente de la cámara y corrección de esta distorsión.

A continuación, se explica el proceso que sigue la herramienta para la estimación de la matriz de transformación. Utiliza el tamaño que se introduce mediante el fichero de los patrones previamente almacenados junto con la orientación y posición en la que se encuentran en la imagen y calcula la matriz de transformación adecuada.

Mediante el análisis de la imagen (gráfico en figura 46) se obtiene la matriz de transformación representada en (eq.1) para pasar de las coordenadas del patrón a las coordenadas de la cámara (T_{CM}).

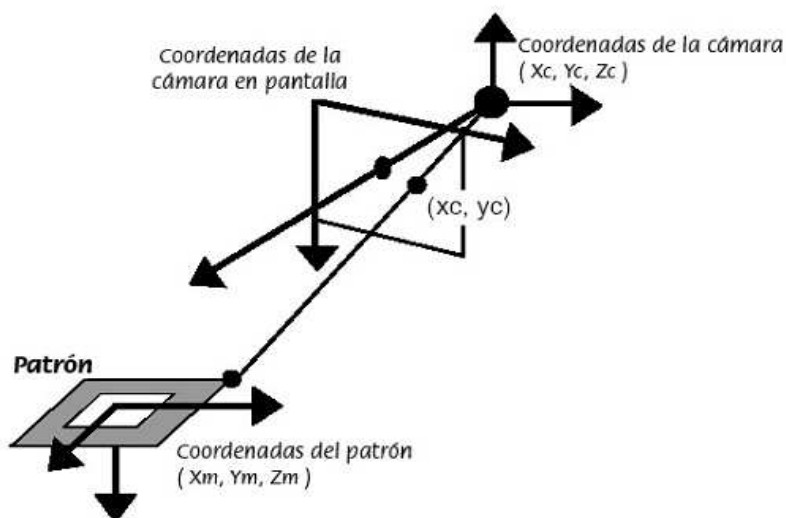


Fig. 46: Relación existente entre las coordenadas del patrón y las coordenadas de la cámara

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} V_{11} & V_{12} & V_{13} & W_x \\ V_{21} & V_{22} & V_{23} & W_y \\ V_{31} & V_{32} & V_{33} & W_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{V}_{3 \times 3} & \mathbf{W}_{3 \times 1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} = \mathbf{T}_{cm} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix}$$

eq. 1

Una vez realizado el proceso de umbralización, se extraen las regiones cuyo contorno concuerda con cuatro segmentos. Las regiones se normalizan para ser comparadas con las plantillas almacenadas. Para este proceso de normalización se usa la transformación perspectiva (eq.2). Todas las variables de la matriz de transformación se determinan por sustitución de las coordenadas de la pantalla y las coordenadas de los cuatro vértices del patrón encontrado por (x_c, y_c) y (X_m, Y_m) respectivamente. Después, ya se puede realizar el proceso de normalización mediante esta matriz de transformación.

$$\begin{bmatrix} hx_c \\ hy_c \\ h \end{bmatrix} = \begin{bmatrix} N_{11} & N_{12} & N_{13} \\ N_{21} & N_{22} & N_{23} \\ N_{31} & N_{32} & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ 1 \end{bmatrix}$$

eq. 2

Cuando dos lados paralelos del cuadrado del patrón son proyectados en la imagen, se obtienen las siguientes ecuaciones para esos segmentos en las coordenadas de la cámara en pantalla:

$$a_1x + b_1y + c_1 = 0, \quad a_2x + b_2y + c_2 = 0$$

eq. 3

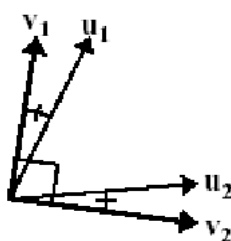
Para cada uno de los patrones, se obtienen estos valores. Dados estos segmentos y la matriz de proyección perspectiva P (eq.4) que se obtiene a partir de la calibración de la cámara, obtenemos mediante sustitución los planos que incluyen estos dos lados de la marca (eq.5).

$$\mathbf{P} = \begin{bmatrix} P_{11} & P_{12} & P_{13} & 0 \\ 0 & P_{22} & P_{23} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} hx_c \\ hy_c \\ h \\ 1 \end{bmatrix} = \mathbf{P} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}$$

eq. 4

$$\begin{aligned} a_1 P_{11} X_c + (a_1 P_{12} + b_1 P_{22}) Y_c + (a_1 P_{13} + b_1 P_{23} + c_1) Z_c &= 0 \\ a_2 P_{11} X_c + (a_2 P_{12} + b_2 P_{22}) Y_c + (a_2 P_{13} + b_2 P_{23} + c_2) Z_c &= 0 \end{aligned}$$

eq. 5



Dados estos dos planos, se obtiene la dirección del vector intersección formado por ambos mediante el producto de sus normales. Una vez calculados los dos vectores unitarios de dirección, se obtiene la componente de rotación $V_{3 \times 3}$ en la matriz de transformación que quedaría especificado como $[V_1^t \ V_2^t \ V_3^t]$.

Una vez obtenidos estos valores, existen en realidad 8 ecuaciones, de las cuales se obtiene el valor que falta para la componente de traslación $W_{3 \times 1}$ $[W_x \ W_y \ W_z]$. Y por tanto, ARToolKit ya es capaz de obtener la matriz de transformación.

4.3.4 Calibración de la cámara

4.3.4.1 Calibrando por ARToolKit

Las lentes de las cámaras introducen en la imagen una distorsión que provoca una deformación en la misma, de manera que la imagen que se ve a través de ellas es distinta a la imagen real que están capturando. Por este motivo se debe llevar a cabo un proceso de calibración de la cámara. Mediante este proceso se intenta corregir la distorsión introducida por la cámara. Para ello, ARToolKit proporciona un fichero denominado «camera_para.dat» que contiene los parámetros por defecto de la calibración. Estos parámetros son válidos para un amplio rango de cámaras. Sin embargo, si no se desea utilizar este fichero se puede generar nuestro propio fichero de calibración, mediante el programa «calib_camera2.exe». Este programa viene junto con ARToolKit y su manejo es bien sencillo. Se necesita un patrón consistente en 24 puntos, dispuestos en 6 columnas y 4 filas, separados 40 milímetros entre sí (también se encuentra dentro de ARToolKit) (Figura 47 (a)). Una vez que se dispone de este patrón lo que se trata es de ir marcando los puntos con el ratón de izquierda a derecha y de

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

arriba abajo, siguiendo siempre este orden (Figura 47 (b)). Se repite esta operación desde distintos ángulos (al menos 10 capturas) para obtener así un mejor resultado, y, por supuesto, es aconsejable que las condiciones lumínicas sean las mismas o parecidas a las que se usarán con la aplicación.

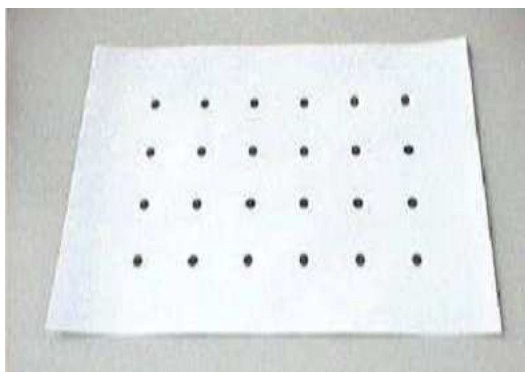


Figura 47 (a) - Patrón de calibración

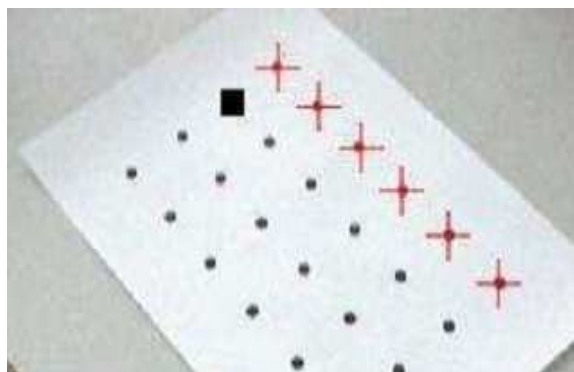


Fig. 47 (b) - Proceso de calibración

4.3.4.2 Calibrando mediante programa externo

No fue el caso de este equipo, y en general no hay problemas al calibrar con ARToolKit, pero en caso de que no nos detecte la cámara correctamente (aunque para eso ya hemos hecho la instalación de drivers, JMStudio y System Librari) deberemos calibrarla manualmente si no queremos usar el fichero de calibración proporcionado por defecto «camera_para.dat».

En esta página Web se nos proporciona un programa y unas instrucciones para calibrar nuestra cámara: http://www.vision.caltech.edu/bouguetj/calib_doc/ Para ejecutar dicho programa necesitaremos una de las siguientes versiones de Matlab: Matlab 5.X, Matlab6.X, o Matlab 7.X. Y uno de los siguientes sistemas operativos: Windows, Unix o Linux.

A continuación, explicaremos los pasos básicos que hay que seguir para usar el programa.

- Descargamos el programa y lo descomprimos en una carpeta. Por defecto, en «TOOLBOX_calib».
- Ejecutamos Matlab y añadimos la ubicación del directorio al «path» principal de este.
- Ejecuta la función de calibración principal «calib_gui». Entonces, aparecerá una ventana de selección en la pantalla que te permitirá escoger entre dos modos de operar: Estándar o eficiente. o Estándar: Todas las imágenes que se usan en la calibración son cargadas en memoria una vez y nunca se vuelve a leer del disco. Se pueden producir fallos de memoria si las imágenes son muy grandes. o Eficiente: Las imágenes se cargan una a una y nunca se guardan de forma permanente en la memoria.

La forma de calibrar la cámara es bastante parecida a cómo la realiza ARToolKit. Se necesita un patrón, proporcionado por el programa. Una vez que se

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

dispone de este patrón lo que se trata es de ir marcando los puntos con el ratón de izquierda a derecha y de arriba abajo, siguiendo siempre este orden. Se repite esta operación desde distintos ángulos (al menos 24 capturas) para obtener así un mejor resultado y, por supuesto, es aconsejable que las condiciones lumínicas sean las mismas o parecidas a las que se usarán con la aplicación.

En la siguiente dirección tenemos un ejemplo de calibración manual: http://www.vision.caltech.edu/bouguetj/calib_doc/htmls/example.html

Para el desarrollo del presente proyecto se crearon varios ficheros de calibración manual. Sin embargo, debido a la escasez de tiempo y la dificultad que entraña una calibración correcta, finalmente se decidió utilizar el fichero que venía por defecto puesto que se comprobó que con este la distorsión era menor que con los generados.

4.3.5 Creación de patrones

Los patrones constituyen una parte fundamental en el desarrollo de la presente aplicación de RA. Son marcas de seguimiento que permiten situar un objeto u otro en una determinada posición de la escena. Los hay de muchos tipos y todos ellos han de poder ser reconocidos por la aplicación. Para ello, primero se deben crear para posteriormente realizar un entrenamiento con los mismos. Para crear una nueva plantilla de reconocimiento basta con usar la marca en blanco (Figura 22) que proporciona ARToolKit, situar en su interior los símbolos que se deseen e imprimirlo para pasar al posterior entrenamiento mediante el programa denominado «mk_patt.exe».

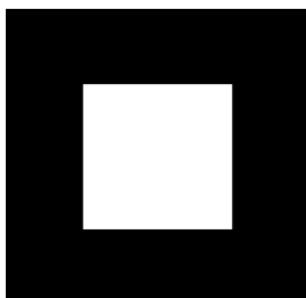


Figura 22 - Marca en blanco para la creación de patrones

A la hora de elegir los mejores patrones que dibujar, éstos son aquellos que son asimétricos y que no tienen muchos detalles. De esta manera la aplicación podrá distinguir fácilmente la orientación de la marca y lo hará lo más rápido posible. Estos son algunos ejemplos propuestos en Hit Lab de posibles patrones de entrenamiento.



Fig. 48: Posibles patrones de entrenamiento

El funcionamiento del programa «mk_patt.exe» es el siguiente. Inicialmente solicita la introducción por teclado del nombre del fichero que contiene los valores de los parámetros de la cámara. Si no se escribe ninguno toma por defecto el fichero «camera_para.dat».



Figura 49: Patrón durante el entrenamiento de la aplicación

Seguidamente, se abrirá una ventana que muestra la imagen que captura la cámara. Para realizar el entrenamiento se debe colocar el patrón sobre una superficie plana e intentar mantener unas condiciones de iluminación similares a las que existirán cuando la aplicación se esté ejecutando. Entonces se debe coger la cámara, colocarla sobre el patrón apuntando directamente sobre él y girarla hasta que aparezca un cuadrado rojo y verde a su alrededor tal y como se muestra en la figura 49. Esto indica que el programa de reconocimiento ha encontrado el cuadrado alrededor de la marca. Una vez que el cuadrado ha sido encontrado y orientado correctamente se debe pulsar el botón izquierdo del ratón. Será entonces cuando se introduzca por teclado el nombre del fichero donde se almacenará el patrón. Una vez introducido el nombre del fichero, un bitmap del patrón es creado y copiado en su interior. Este será el que utilizará ARToolKit para identificar los patrones en tiempo de ejecución. Tras haber entrenado uno de ellos, otros pueden ser entrenados apuntándolos con la cámara y repitiendo el proceso antes descrito, o pulsar el botón derecho del ratón y abandonar la aplicación.

Este procedimiento que se acaba de explicar de manera genérica es el que se lleva a cabo para la creación de todos los patrones presentes en la aplicación. En este caso las marcas elegidas han sido un conjunto de letras y palabras, intentando que sean lo más asimétricas posibles. En la figura 50 se vé un ejemplo de patrón creado.

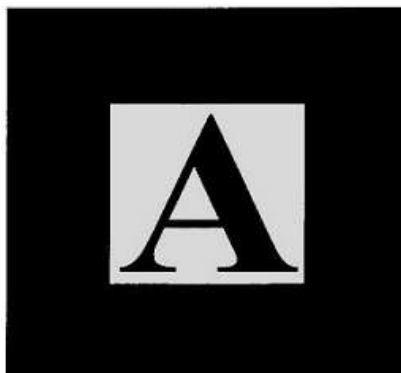


Fig. 50: Patrones de letras y palabras

A la hora de utilizar estos patrones hay que tener en cuenta una serie de pequeñas consideraciones. Por ejemplo, que los objetos virtuales solamente aparecerán cuando las marcas de seguimiento estén completamente visibles. En el momento en el que cualquier parte de la marca quede tapada (ya sea por la mano o por cualquier otro elemento), el objeto representado desaparecerá inmediatamente de la escena, por lo que se hará necesario el diseño de la «multimarca», formada por un conjunto de marcas y que funciona como una sola, con la ventaja de si una marca queda oculta, mediante cualquiera de las otras se puede conseguir la orientación y posición de los objetos.

Otro hecho a tener en cuenta es el tamaño de los patrones. No es lo mismo reconocer un patrón de 4 centímetros que uno de 8 a una distancia fija de 30 centímetros. Esto es debido a que existen unas distancias máximas para que ARToolKit pueda reconocer el patrón en función de su tamaño. Por lo tanto, cuánto más grande sea la marca mayor podrá ser la distancia a la que se podrá situar la cámara para poder identificar el patrón. En este mismo sentido también influye la complejidad del patrón, puesto que cuánto más complejo sea más difícil será de reconocer.

Así pues, los mejores patrones son aquellos que son simples y tienen grandes regiones de color blanco y negro.

4.3.6 Creación de la Aplicación de Realidad Aumentada

Una vez tenemos todo el sistema configurado, tenemos los objetos a mostrar y las plantillas, cargamos el proyecto ARToolKit en Visual Studio, para ello hemos de configurarle al Visual Studio las rutas para que coja las librerías necesarias.

Para el desarrollo de la aplicación, vamos paso a paso a través de la estructura de un programa existente, que modificaremos: SimpleTest.c

Este programa consiste simplemente en una rutina gráfica principal y varias rutinas de dibujo.

Las funciones que corresponden a la aplicación constan de seis pasos:

- | | | |
|----|---------------------------------------|-----------------|
| 1. | Iniciar la aplicación | init |
| 2. | Coge un fotograma de entrada de vídeo | arVideoGetImage |
| 3. | Detectar los marcadores | arDetectMarker |

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

4. Calcula la transformación `arGetTransMat`
5. Dibuja los objetos virtuales `draw`
6. Cierra la captura de vídeo `cleanup`

Los pasos de 2 a 5 se repiten indefinidamente en `mainloop`

Las funciones más importantes son `main`, `mainloop`, `init`, `draw` y `cleanup`. Seguidamente veremos estas llamadas con un poco más de detalle.

4.3.6.1 Main

La rutina de llamada es la siguiente en `Simple.c`:

```
main(int argc, char *argv[])
{
    init();
    arVideoCapStart();
    argMainLoop(NULL, keyEvent, mainLoop );
}
```

La rutina de inicialización de inicio contiene código para iniciar la captura de vídeo, la lectura de la marca y parámetros de la cámara y la configuración de la ventana gráfica. Esto corresponde al paso 1. A continuación, entramos en el estado en tiempo real, con una llamada a la función de vídeo `arVideoCapStart`. Después, la función llama `argMainLoop` que comienza el bucle principal del programa y asocia la función `keyEvent` con los eventos de teclado y `mainloop` con la representación gráfica principal del bucle. La definición de `argMainLoop` está contenida en el archivo `gsub.c`.

4.3.6.2 Init

La rutina de inicio se llama desde la rutina principal y se utiliza para inicializar la captura de vídeo y leer en los parámetros de aplicación inicial `ARToolKit`.

Comienza abriendo la ruta de video y estableciendo el tamaño de la imagen.

```
/* open the video path */
if(arVideoOpen(vconf ) < 0 ) exit(0);

/* find the size of the window */
if(arVideoInqSize(&xsize, &ysize) < 0 ) exit(0);
printf("Image size (x,y) = (%d,%d)\n", xsize, ysize);
```

La variable `vconf` contiene la configuración de vídeo inicial y se define en la parte superior de `simple.c`. Para cada plataforma se define una cadena por defecto, que generalmente abre la primera secuencia de vídeo disponibles.

Después, tenemos que inicializar los parámetros de aplicación. Los parámetros clave son:

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

- Los patrones que se utilizarán para las plantillas de reconocimiento y los objetos virtuales correspondientes a estos patrones.
- Las características de la cámara de vídeo que se utiliza.

Se trata de leer en los nombres de archivo que puede ser especificado en la línea de comandos o por defecto en el código los nombres de archivo.

Así, los parámetros de la cámara son leídos en el archivo de parámetros por defecto para la cámara `Data/camera_para.dat`:

```
/* set the initial camera parameters */
if(arParamLoad(cparaname, 1, &wparam) < 0 ) {
    printf("Camera parameter load error !!\n");
    exit(0);
}
```

A continuación, los parámetros se transforman para el tamaño de imagen actual, ya que los parámetros de la cámara cambian en función del tamaño de la imagen, incluso si es la misma cámara.

```
arParamChangeSize(&wparam, xsize, ysize, &cparam );
```

Los parámetros de cámara se configuran por los leídos, y se muestran por pantalla:

```
arInitCparam(&cparam );
printf("*** Camera Parameter ***\n");
arParamDisp(&cparam );
```

Después de leer la definición del patrón por defecto de `Data/patt.hiro`:

```
if((patt_id=arLoadPatt(patt_name)) < 0 ) {
    printf("pattern load error !!\n");
    exit(0);}
}
```

`patt_id` es una identificación de patrón identificado.

Por último se abre la ventana de gráficos:

```
/* open the graphics window */
argInit(&cparam, 1.0, 0, 0, 0, 0 );
```

El segundo argumento de `argInit` define una función de zoom, se ajusta a 1,0 para que coincida con el formato de imagen del video, a 2,0 para el doble del tamaño.

4.3.6.3 Mainloop

esta es la rutina dónde se hacen la mayor cantidad de llamadas a funciones de `ARToolKit` y que contiene el código correspondiente a los pasos 2 a 5.

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

En primer lugar se captura un cuadro de vídeo usando la función `arVideoGetImage`:

```
/* grab a video frame */
if((dataPtr = (ARUint8 *)arVideoGetImage()) == NULL ) {
    arUtilSleep(2);
    return;
}
```

La imagen de vídeo se muestra a continuación en la pantalla. Esto puede ser una imagen sin corregir, o una imagen deformada y corregidas las distorsiones de la cámara. Corregir la imagen produce una imagen más normal, pero puede dar lugar a una reducción significativa en la tasa de fotogramas de vídeo. En esta aplicación la imagen se corrige:

```
argDrawMode2D();
argDispImage(dataPtr, 0,0 );
```

Entonces la función `arDetectMarker` se utiliza para buscar la imagen de vídeo en busca de cuadrados que concuerden con los patrones marcador correctos:

```
if(arDetectMarker(dataPtr, thresh, &marker_info,
&marker_num) < 0 ) {
    cleanup();
    exit(0);
});
```

El número de marcadores encontrados se encuentran en la variable `marker_num`, mientras que `marker_info` es un puntero a una lista de las estructuras de marcas que contiene la información de coordinación y valores de los objetos de reconocimiento y números de identificación para cada uno de los marcadores.

En este momento, la imagen de vídeo se ha expuesto y analizado. Así que no es necesario que se use más: podemos pedir al captador iniciar una nueva captura de imagen al mismo tiempo que operamos más funciones. Para ello se llama a `arVideoCapNext`:

```
arVideoCapNext();
```

A continuación, todos los valores de confianza de los marcadores detectados se comparan para asociarles los correctos números de identificación de marca con el valor más alto de confianza:

```
/* check for object visibility */
k = -1;

for(j = 0; j < marker_num; j++ ) {
    if(patt_id == marker_info[j].id ) {
        if(k == -1 ) k = j;
    }
}
```

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

```
        else if(marker_info[k].cf < marker_info[j].cf ) k
= j;
    }
}

if(k == -1 ) {
    argSwapBuffers();
    return;
}
```

La transformación entre las tarjetas de marcador y luego la cámara se puede encontrar utilizando la función `arGetTransMat` :

```
/* get the transformation between the marker and the real
camera */
arGetTransMat(&marker_info[k], patt_center, patt_width,
patt_trans);
```

La posición real de la cámara y orientación en relación con la marca *i* están en la matriz 3x4, `patt_trans` .

Por último, los objetos virtuales se puede dibujar en la marca utilizando la función `draw`:

```
draw();
argSwapBuffers();
```

Nota: un simple paso de optimización se realiza si no se encuentra ningún patrón (`k == -1`) , se cambia directamente el búfer sin llamar a `draw` y vuelve:

```
if(k == -1 ) {
    argSwapBuffers();
    return;
}
```

2.3.6.4 Draw

Las funciones se dividen en el renderizado, ajustar la matriz, renderizar el objeto. Se inicializa una representación 3D por demanda de `ARToolKit` para hacer renderización objetos en 3D, y la mínima configuración de estado de `OpenGL`:

```
argDrawMode3D();
argDraw3dCamera(0, 0 );
glClearDepth(1.0 );
glClear(GL_DEPTH_BUFFER_BIT);
glEnable(GL_DEPTH_TEST);
```

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

```
glDepthFunc (GL_LEQUAL) ;
```

Después se necesita convertir la transformación computarizada (3x4 matriz) a un formato OpenGL (serie de 16 valores), utilizando la llamada a la función `argConvGlp` para. Estos dieciséis valores son la posición y la orientación real de la cámara, usándolos para fijar la posición de la cámara virtual hace que cualquier objeto gráfico se dibuje y aparezca exactamente alineada con la marca física correspondiente.

```
/* load the camera transformation matrix */  
argConvGlp (patt_trans, gl_para) ;  
glMatrixMode (GL_MODELVIEW) ;  
glLoadMatrixd (gl_para ) ;
```

La posición de la cámara virtual se establece mediante la función de OpenGL `glLoadMatrixd (gl_para)`. La última parte del código es la representación de objetos en 3D.

```
glEnable (GL_LIGHTING) ;  
glEnable (GL_LIGHT0) ;  
glLightfv (GL_LIGHT0, GL_POSITION, light_position) ;  
glLightfv (GL_LIGHT0, GL_AMBIENT, ambi) ;  
glLightfv (GL_LIGHT0, GL_DIFFUSE, lightZeroColor) ;  
glMaterialfv (GL_FRONT, GL_SPECULAR, mat_flash) ;  
glMaterialfv (GL_FRONT, GL_SHININESS, mat_flash_shiny) ;  
glMaterialfv (GL_FRONT, GL_AMBIENT, mat_ambient) ;  
glMatrixMode (GL_MODELVIEW) ;  
glTranslatef (0.0, 0.0, 25.0 ) ;
```

Y para terminar, se reiniciará por defecto algunas variables OpenGL:

```
glDisable (GL_LIGHTING ) ;  
glDisable (GL_DEPTH_TEST ) ;
```

Los pasos mencionados anteriormente se producen cada vez a través del bucle de representación principal. Mientras el programa se está ejecutando, se manejan eventos de ratón por la función `MouseEvent` y eventos de teclado por la función `KeyEvent`.

2.3.6.5 Cleanup

La función de limpieza está llamado a detener el procesamiento de vídeo y cerrar la ruta de vídeo para liberar la cámara a otras aplicaciones:

```
arVideoCapStop () ;  
arVideoClose () ;  
argCleanup () ;
```

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

Con esto se compila y tenemos la aplicación a nuestra medida.

Se hicieron modificaciones como agregar otra plantilla o cambiar la plantilla por otra personalizada, para eso hay que cambiar el valor de la variable `patt_name` y poner la ruta a nuestra plantilla. Para agregar más plantillas me base en `simpleVRML`.

Cuando se compila la aplicación, se nos queda un ejecutable que podemos cargar desde consola de comandos, incluir en un `bat` para que se cargue sin necesidad de poner comandos o incluir el ejecutable en otro proyecto. Aunque si se va a incluir en otro proyecto, lo mejor es usar directamente el objeto que podemos crear al compilar. Si vamos a usarlo como un ejecutable tipo `bat`, tenemos la ventaja de que editándolo podemos cambiarle los parámetros o dejarle ordenes condicionales. En este caso se utilizó el método de crear un `bat`, aunque se cargaba directamente desde Visual Studio, también se probó por los métodos descritos, excepto incluyéndolo en otro proyecto.

Se hizo una carga de la molécula que se diseñó desde cero mediante el `BALLView`, y se arrancó la aplicación. En la figura 51 se puede ver la ejecución de la aplicación en la que crea una RA añadiendo ADN. Se testeó intercambiando plantillas que se crearon y se pudo probar que cada plantilla cargaba su objeto correspondiente. Se cargaron 6 plantillas diferentes y 4 moléculas simples (H_2O , $NaOH$, HCl , O_3) y 2 complejas (ADN creada desde `BALLView`, y Polietilenglicol obtenida de base de datos de proteínas⁴). En la figura 52 se ve como cargan en tiempo real dos plantillas con los objetos 3D más complejos.



Fig. 51: Molécula de ADN creada por patrón `pdb` con `BALLView` e integrada en una aplicación de RA.

4 «Protein Data Bank» ubicado en: <http://www.rcsb.org/>

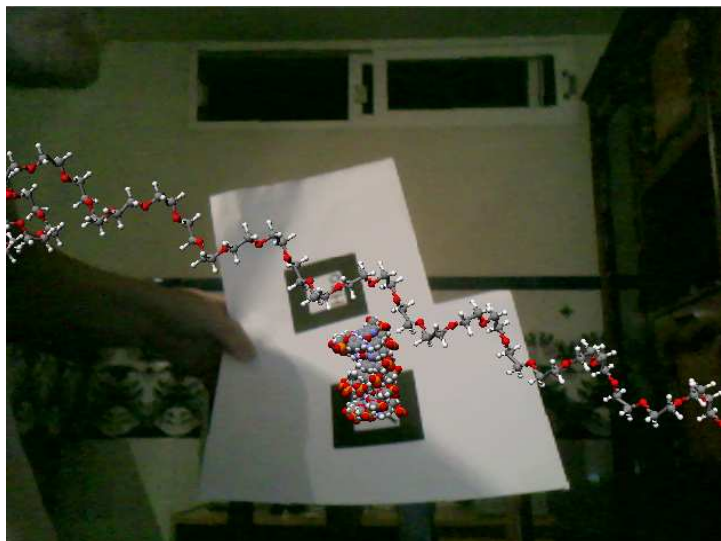


Fig. 52: Dos plantillas mostrando a la vez dos molecular realizadas con BALLView, la de arriba polipropileno (POLYETHYLENE GLYCOL (N=34)) uno de los plásticos más comunes, la de abajo, otra vez DNA.

4.3.7 Requisitos mínimos

Dependiendo del editor molecular, los requisitos varían, pueden ir desde tener una tarjeta gráfica de última generación con capacidad con monitor de visión en 3D, a no necesitar nada más que un computador convencional medio – bajo porque no tiene entorno gráfico para la creación de moléculas. En general si se utilizará un editor de licencia GPL, se necesitará hacer una compilación en la máquina, con lo que conlleva el requisito muchas veces de obtener librerías o ejecutar la compilación bajo cygwin. Bien para este caso en concreto que se utilizó BALLView no siendo necesaria una compilación (ya que además del fuente, tienen accesible los binarios para Windows y MacOS). Los requisitos específicos se detallaron convenientemente en su correspondiente apartado.

Con respecto a ARToolKit es necesario obtener las librerías de DirectX 8 o el ejecutable de DirectX y Windows SDK último con todas las librerías, cabeceras, y herramientas para DirectShow. Además se necesita las librerías DirectShowLib, DirectShowSamples-2010-February, openvrml-0.18.6, OpenGL (1.X, la 2 no es recomendable, pues aunque da soporte retrospectivo a versiones anteriores pesa más y puede crear conflictos), y es recomendable también librerías GLUT (se instaló la 3.7). Sistema Operativo UNIX (Windows, Linux, MacOS, ...)

Las especificaciones hardware son pocas para ARToolKit:

- Tarjeta gráfica con OpenGL.
- Memoria mínima de 128 MB o menos.
- Cámara.

Pero estas especificaciones son las mínimas, ya que si se cargan varios objetos complejos a la vez, se vería reducido el sistema hasta tal punto que dejaría de ser RA

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

por no estar en tiempo real. Se comprobó con la aplicación que se creó, al cargar los dos objetos complejos al unisono el sistema iba a golpes (Pentium Intel Core 2 Duo, Nvidia 8400, 4 GB de memoria)

CAPITULO 5 Conclusiones y trabajos futuros

5.1 Conclusiones

Concluyendo con los editores moleculares, cada aplicación tiene unas características distintivas, pudiendo elegir según las necesidades. Hay editores que están más enfocados a hospitales, mientras que otros se ven claramente didácticos, muy simples y quizás escuetos.

Dentro de que cada editor es independiente, se puede resaltar que todos tienen más o menos las mismas capacidades representativas de las moléculas, entre las que estaban la representación en bola, en palo o combinación. Además se vio, que por contra de la representación de estructura, la representación de colores no tiene una representación unánime.

Respecto a la aplicación en ARToolKit e integración de moléculas a esta, se ha de destacar que al estar las representaciones en VRML, dan una gran visión de objeto integrado con movimiento ágiles. Pero en el momento que se carga a la aplicación con un gran número de moléculas y varias estructuras a la vez, le cuesta un poco reaccionar y cargar los objetos, haciendo un poco difícil integrar muchas estructuras complejas a la vez en una aplicación comercial directamente (en la que puede estar limitada de recursos).

Se ha desarrollado este proyecto para estudiar las posibilidades que ofrece el mercado en el desarrollo de aplicaciones de edición moleculares, también se ha pretendido ofrecer una solución didáctica a la enseñanza, en específico en el campo de la química, haciendo una aplicación de RA. Este tipo de aplicación pensamos que se puede utilizar de manera útil en sectores tales como:

Educación y entretenimiento: Con el fin de facilitar la comprensión de los elementos químicos, moléculas y reacciones. U observar reactivos y compuestos ante diversas situaciones.

Evaluación: Ya que se representa una reproducción fidedigna de la molécula por la que se puede estar entrenando.

5.2 Trabajos futuros

El mundo de la química tiene descubrimientos a diario y con cada descubrimiento se dedican más recursos, lo que conlleva a conseguir sistemas que nos faciliten más las interacciones moleculares, la visión espacial, y el cálculo de sus propiedades, por ello es seguro que cada vez se le va a exigir al mercado aplicaciones más rápidas, intuitivas y con más capacidades de síntesis y análisis. Así pues los programas aquí analizados en un tiempo van a ser sustituidos por sus sucesores que les superaran en varias características. Se tendrá que analizar, estudiar, y probar nuevas aplicaciones.

Además también se pueden realizar aplicaciones de RA que intenten integrar directamente los archivos químicos, transformándolos directamente bajo demanda en un objeto de RA, sin necesidad de aplicación alguna de edición molecular.

Capítulo 6 Bibliografía

- [AEL09] Augmented Environments Lab; *ARhrrr!*, news 15 May 2009
<http://www.augmentedenvironments.org/lab/research/handheld-ar/arhrrrr/>
- [Army09] Henderson, S.J.; Feiner, S.; (2009) *Evaluating the benefits of augmented reality for task localization in maintenance of an armored personnel carrier turret* .
<http://www.computer.org/portal/web/csdl/doi/10.1109/ISMAR.2009.5336486>
- [Azuma01] Azuma, R.; Bailiot, Y.; Behringer, R.; Feiner, S.; Julier, S.; and MacIntyre, B.; (2001) Recent advances in augmented reality. *IEEE Computer Graphics and Applications*, 21 (6):34–47.
- [Azuma97] Azuma, R.T. (1997) A survey of Augmented Reality, Presence: Teleoperators and Virtual Environments, Vol. 6, N. 4, pp. 355 – 385. HYPERLINK
«<http://www.cs.unc.edu/~azuma/ARpresence.pdf>»
<http://www.cs.unc.edu/~azuma/ARpresence.pdf>
- [Benton04] Benton C. (2004) Augmented Reality for Maritime Navigation: The Next Generation of Electronic Navigational Aids. HYPERLINK "http://www.l-3klein.com/navigation/looksea/projec_usa_augmented_reality_paper.pdf" http://www.l-3klein.com/navigation/looksea/projec_usa_augmented_reality_paper.pdf
- [Bimber05] Bimber O., Raskar R.; (2005) *Spatial augmented reality: merging real and virtual worlds*. Pages 71-92
- [BMWAR08] BMW Augmented Reality. BMW research projects - virtual world meets reality. Obtenido de HYPERLINK
«http://www.bmw.com/com/en/owners/service/augmented_reality_introduction_1.html»
http://www.bmw.com/com/en/owners/service/augmented_reality_introduction_1.html
- [Dabpetch07] Dabpetch, A.; Viratanapanu, A.; Sukuntho, S.; Choochaisri, S.; Kunarittipol, W.; (2007) Larngear Technology. <http://www.larngeartech.com/>
- [Esteban07] Esteban, P.; Restrepo, J.; Trefftz, H.; Jaramillo, J.E.; Alvarez, N.; (2007). La realidad aumentada: un espacio para la comprensión de conceptos del cálculo en varias variables. Universidad Eafit, Medellín, Colombia
<http://www.eafit.edu.co/EafitCn/Investigacion/Grupos/Ingenieria/RealidadVirtual/Realidad+Virtual.htm>
- [Feuerstein07] Feuerstein, M.; Navab, N.; and Bichlmeier, C.; (2007) Laparoscopic virtual mirror: New interaction paradigm for monitor based augmented reality. In *Virtual Reality Conf*, page 43–50. IEEE Press
- [Fruend02] Fruend, J., Grafe, M., Matyszczok, C., Vienenkoetter, A (2002) AR-based Training and Support of Assembly Workers in Automobile Industry, The first IEEE International Workshop on Augmented Reality ToolKit
- [Grimson96] Grimson, W.E.L.; Ettinger, G.J.; Kapur, T.; Leventon, M.E.; Wells III, W.M.; Kikinis R.; (1996) "Utilizing Segmented MRI Data in Image-Guided Surgery." In *IJPRAI*. http://www.ai.mit.edu/projects/medical-vision/surgery/surgical_navigation.html

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

[Henrysson06] Henrysson, A.; Billingham, M.; and Ollila M.; (2006) Ar tennis. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Sketches*, page 13, New York, NY, USA. ACM

[Hiroshi97] Hiroshi, I.; Brygg, U.; Tangible bits: towards seamless interfaces between people, bits and atoms. In *CHI '97: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 234–241, New York, NY, USA, 1997. ACM.

[Juan05] Juan, M.C., Alcañiz, M., Monserrat, Botella, C., Baños, R.M. and Guerrero, B. (2005) Using augmented reality to treat phobias., *IEEE Computer Graphics and Applications*. 25 (6): 31-37

[Juan06a] Juan, M.C., Joele, D. Baños, R., Botella, C., Alcañiz, M., Van Der Mast, C. (2006) *A markerless augmented reality system for the treatment of phobia to small animals*. The 9th international workshop on Presence – Presence 06. pp. 71-74

[Juan06b] Juan, M.C.; Baños, R.; Botella, C.; Pérez, D.; Alcañiz, M.; Monserrat, C. (2006) An Augmented Reality System for acrophobia: The sense of presence using immersive photography, *Presence: Teleoperators & Virtual Environments*, 15: 393-402

[Juan07] Juan, M.C., Alcañiz, M., Calatrava, J., Zaragoza, I., Baños, R., Botella, C. (2007) An Optical see-through Augmented Reality system for the treatment of phobias to small animals. *LECTURE NOTES IN COMPUTER SCIENCE: ICVR 2007*. Vol. 4563. pp. 651-659

[Juan08a] Juan, M.C.; Furió, D.; Mollá, R.; Vicent, M.J.; Vivó, R.; Giménez, M.; (2008) Edutainment games included as activities in the Summer School of the Technical University of Valencia, *Rev. Game-On*, pag. 147-151, Valencia

[Juan08b] Juan, M.C.; Beatrice, F.; Cano, J.; Gimenez, M.; Augmented Reality for learning the interior of the Human Body (2008) *IEEE International Conference on Advanced Learning Technologies Learning technologies in the Information Body*, *IEEE International Conference on Advanced Learning Technologies Learning technologies in the Information society (ICALT'08)*, pp. 186-188

[Juan10a] Juan, M. C.; Toffetti, G.; Abad, F.; Cano, J.; Tangible Cubes used as the user interface in an Augmented Reality game for edutainment (2010); *IEEE International Conference on Advanced Learning Technologies*, pp. 599-603

[Juan10b] Juan, M. C.; Llop, E.; Abad, F.; Lluch, J.; Learning words using Augmented Reality (2010); *IEEE International Conference on Advanced Learning Technologies*; pp. 422-426

[Kato99] Kato, H., Billingham, M. "Marker tracking and hmd calibration for a video-based augmented reality conferencing system.", *In Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR 99)*, October 1999. pp 85-95. Hiperlink: <http://www.hitl.washington.edu/artoolkit/Papers/IWAR99.kato.pdf>

[Kaufmann10] Kaufmann, H.; (2010) Construct3D. An Application and Framework for using Augmented Reality in Mathematics and Geometry Education. Interactive Media Systems Group, Vienna University of Technology. <http://www.ims.tuwien.ac.at/research/construct3d/>

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

- [Klinker99] Klinker G., Stricker D., Reiners D. Augmented Reality: A Balance Act between High Quality and Real-Time Constraints (1999) HYPERLINK
"http://www.bruegge.in.tum.de/publications/includes/pub/%20klinker1999ismr/"
<http://www.bruegge.in.tum.de/publications/includes/pub/klinker1999ismr/klinker1999ismr.pdf>
- [Klopfer08] Klopfer, E.; (2008) Augmented Learning Research and Design of Mobile Educational Games MIT Handheld Augmented Reality Simulations. Hiperlink:
<http://education.mit.edu/ar/>
- [Looser07] Looser, J.; Billingham, M.; Grasset, R.; and Cockburn, A.; An evaluation of virtual lenses for object selection in augmented reality. In *GRAPHITE '07: Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*, pages 203–210, New York, NY, USA, 2007. ACM.
- [Martens04] Martens, J.B; Qi, W.; Aliakseyeu, D.; J. F. Kok, A.; and van Liere, R.; (2004) Experiencing 3d interactions in virtual reality and augmented reality. In *EUSAI '04: Proceedings of the 2nd European Union symposium on Ambient intelligence*, pages 25–28, New York, NY, USA. ACM.
- [Matyszczok02] Matyszczok, C., Freund, J., Radkowski, R. (2002) AR-based Product Design in Automobile Industry, The First IEEE International Augmented Reality Toolkit Workshop
- [Mourkoussis04] Mourkoussis, F.; White, N.; Darcy, M.; Sifniotis, J.; Petridis, M.; Basu, P.; Lister P.F., A.; (2004); Web3D and Augmented Reality to support Engineering Education. *Liarokapis, World Transactions on Engineering and Technology Education, UICEE*, 3(1): 11-14.
- [Oliver09] Oliver J.; (2009) *LevelHead*. International Symposium on Mixed and Augmented Reality. Orlando, Florida, USA. <http://www.ismar09.org/>
- [Oviatt99] Oviatt S.; (1999) Ten myths of multimodal interaction. *Commun. ACM*, 42 (11):74–81.
- [Pérez07] Pérez, D., Juan, M.C., Alcañiz, M., Baños, R., Botella, C. (2007) Two similar systems for the treatment of acrophobia. An Augmented Reality system and a Virtual Reality system. *LECTURE NOTES IN COMPUTER SCIENCE: ICVR 2007*. pp. 850-854.
- [PNM09] Presentacion *Project Natal* de Microsoft en E3, los angeles, USA,2009.
<http://www.xbox.com/es-ES/news-features/news/Project-Natal-in-detail-050609.htm>
- [Potalés08] Portalés, C.; (2008). Tesis Doctoral: *Entornos Multimedia de Realidad Aumentada en el Campo del Arte*. Doctorado: Artes Visuales e intermedia. Páginas:67-69. Sección 1.2.2.
- [Regenbrecht05] Regenbrecht H.; Dunedin, Baratoff G.; Regensburg; Wilke W, Ulm; IEEE COMPUTER GRAPHICS AND APPLICATIONS FINAL MANUSCRIPT University of Otago, Siemens VDO Automotive AG and DaimlerChrysler AG; November / December 2005

Desarrollo de una aplicación de Realidad Aumentada para simulación de moléculas

- [Reitmayr05] Reitmayr, G.; Eade, E.; and Drummond T.; (2005) Localisation and interaction for augmented maps. In *ISMAR '05: Proceedings of the Fourth IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 120–129, Washington, DC, USA,. IEEE Computer Society.
- [Rohs06] Rohs M.; (2006) Marker-based embodied interaction for handheld augmented reality games. *Journal of Virtual Reality and Broadcasting*, 4 (5).
- [SanFrutos03] San Frutos, J.; (2003) Internet para la enseñanza de la Química. Cap. ¿Qué es la química? Dep. CCTT Físico químicas. UNED. Hiperlink: <http://encina.pntic.mec.es/~jsaf0002/p1.htm>
- [SCAT10] *Specialist Schools and Academies Trust*. LearnAR. (2010). Hiperlink: <https://www.ssatrust.org.uk/achievement/future/Pages/AugmentedReality.aspx>
- [Sirna09] Sirna G.; (2009) *ARQ, realidad aumentada para Diseñadores de interiores y Arquitectos*. Innovar 2009; <http://www.inventionary.com.ar/2009/10/24/arq-realidad-aumentada-para-disenadores-y-arquitectos-innovar-2009/>
- [Straier05] Straier W Fischer J, Bartz D. Intuitive and lightweight user interaction for medical augmented reality. In *Proceedings of vision, modeling, and visualization*, pages 375–82, 2005.
- [Theodore03] Theodore L. Brown, Bruce E. Bursten, Julia R. Burdge; (2003) Química: La Ciencia Central. Capitulo 1-2. pp. 1-74
- [Thomas00] Thomas, B., Close, B., Donoghue, J., Squires, J., De Bondi, P., Morris, M., and Piekarski, W.; (2000) *ARQuake: An Outdoor/Indoor Augmented Reality First Person Application*. In 4th Int'l Symposium on Wearable Computers, pp 139-146, Atlanta, Ga, USA. <http://wearables.unisa.edu.au/projects/arquake/>
- [Tripathi00] Tripathi A.; (2000) *Augmented reality : an application for architecture*. A Thesis Presented to the Faculty of the school of architecture university of southern California; In Partial Fulfillment of the Requirements for the Degree Master of building science (Building Science). Se puede encontrar en: http://www.usc.edu/dept/architecture/mbs/thesis/anish/thesis_report.htm
- [Woods02] Woods, E.; Binder, G.; Stangl, M.; Billinghamurst, M.; (2002) Magic Book . The Human Interface Technology Laboratory New Zealand (HIT Lab NZ), University of Canterbury, Christchurch, New Zealand. Hiperlink: http://www.hitlabnz.org/wiki/Black_Magic_Book

Glosario

Alámbrico: Tipo de visualización de entornos 3D en los que sólo se ven los puntos, rectas, superficies, etc.

API: Interfaz de programación de aplicaciones. Es un conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

HMD: Dispositivo de visualización que utilizan pequeñas pantallas enfrente de los ojos.

IDE: Es un entorno de programación integrado en una aplicación software que proporciona una interfaz gráfica, un editor de texto, un compilador y un depurador.

Plataforma: Principio sobre el cual se constituye un hardware, sobre el que un software puede ejecutarse y desarrollarse.

Plug-in: Aplicación que se relaciona con otra para aportarle funciones nuevas y generalmente muy específicas.

Realidad Aumentada: Sistema que enriquece el mundo real con elementos virtuales, generados mediante ordenador, que coexisten en un mismo espacio con los objetos reales, además existe una interacción en tiempo real y una localización 3D.

Realidad Virtual: un entorno tridimensional, interactivo y generado por ordenador en el cual se sumerge una persona.

SDK: Conjunto de herramientas de desarrollo que le permite a un programador crear aplicaciones para un sistema bastante concreto.

See-through HMD: Tecnología para mostrar gráficos sobre la vista del mundo real que utilizan mezcla de vídeo y muestran las imágenes mezcladas en un HMD sin visión directa del mundo real.

Umbralización: Técnica que permite determinar qué puntos corresponderían a la marca y cuáles no.