

# Interacción basada en la Mirada en Entornos Virtuales

Jorge Jiménez, Diego Gutierrez y Pedro Latorre

Grupo de Informática Gráfica Avanzada (GIGA)  
Laboratorio Aragonés de Usabilidad  
Departamento de Informática e Ingeniería de Sistemas  
Centro Politécnico Superior  
50018 Zaragoza  
{487232, diegog.latorre}@unizar.es

**Resumen** En este artículo presentamos un interfaz alternativo que permite a los usuarios percibir nuevas sensaciones en entornos virtuales. Interactuar en entornos tridimensionales mediante la mirada crea la falsa sensación de controlar objetos con la mente, consiguiendo una inmersión más profunda que la obtenida con otras técnicas. La incorporación de un motor de física en tiempo real aumenta en gran medida la percepción de estar moviendo algo *real*.

La exposición de este trabajo se centra en el estudio de las técnicas existentes, en la descripción detallada del interfaz ideado, así como en la evaluación objetiva y subjetiva del interfaz.

El filtrado de los datos provenientes del *eye tracker* es de fundamental importancia para el uso de interfaces oculares en entornos virtuales, y por ello se ha hecho especial hincapié en este aspecto.

## 1 Introducción

### 1.1. Motivaciones

La interfaz habitual de los ordenadores actuales, basada en entornos WIMP<sup>27</sup>, permite una inmersión relativamente limitada. Recientemente, se han introducido en el mercado nuevos dispositivos que pretenden superar las limitaciones de los interfaces convencionales. Un ejemplo de innovación en este campo es el mando introducido por la consola Nintendo Wii, capaz de reconocer de forma limitada los movimientos realizados por los jugadores, o *EyeToy*, un videojuego que a través de una cámara de video captura los movimientos de los usuarios.

En este artículo exploramos el uso de los dispositivos de rastreo ocular mediante un sistema que permite al usuario interactuar con la mirada en un entorno virtual de una forma innovadora, y que consigue una mayor inmersión de los usuarios en los entornos virtuales.

---

<sup>27</sup> *Windows, Icons, Menus, Pointer*

IX Congreso Internacional Interacción, Albacete 9-11 de Junio de 2008  
Grupo LoUISE-Universidad de Castilla-La Mancha

## 1.2. Estado del arte

En la bibliografía previa se han descrito y analizado interfaces de usuario cuya única entrada es la mirada del usuario [6,3,1]. En dichos sistemas es posible realizar acciones como seleccionar objetos, moverlos o mostrar información sobre los mismos, bajo el contexto de un interfaz de usuario convencional.

Algunos estudios realizados nos indican que los interfaces basados en la mirada pueden ser más eficientes que los tradicionales basados en ratón [5].

En el trabajo de Tanriverdi y Jacob [11] se introduce la selección de objetos mediante la mirada en entornos virtuales y se compara con la selección usando un dispositivo de rastreo de la posición de la mano del usuario. Aunque los resultados de las pruebas indican que la mirada es una forma de interacción más eficiente, estudios posteriores desvelan que el rendimiento conseguido no es tan importante si se usa el mismo algoritmo de trazado de rayos en ambos tipos de dispositivos [7].

Smith y Graham han estudiado la integración de interfaces basados en la mirada en videojuegos ya existentes pertenecientes a diversos géneros [4]. De los resultados obtenidos se deduce que la eficiencia es peor con respecto a un sistema más tradicional como puede ser el ratón. No obstante, se observa que la satisfacción de los usuarios es mayor con dispositivos de detección de la mirada cuando ésta se usa como cursor en el control de avatares.

El uso de la mirada para apuntar en juegos en primera persona también ha sido analizado previamente [8]. El resultado nos dice que la eficiencia de dichos sistemas es peor que la del ratón. Por otro lado, también indica que el rendimiento usando un rastreador ocular mejora con el entrenamiento, y que sería necesario equiparar el entrenamiento en ambos dispositivos para poder hacer una comparación justa.

## 2 Entorno de trabajo y recursos necesarios

### 2.1 El Laboratorio Aragonés de Usabilidad

El Laboratorio Aragonés de Usabilidad es un espacio de investigación y servicios dedicado al desarrollo, análisis y optimización de nuevas técnicas de mejora de las interfaces persona-ordenador. El Laboratorio pretende resolver inquietudes más allá de las técnicas propias de funcionalidad, enfocándose en analizar de forma integral la experiencia del usuario.

El Laboratorio ha proporcionado el hardware, software y personal necesario para la implementación y evaluación del sistema propuesto en este artículo.

### 2.2. Hardware

Para la obtención de la posición observada por el usuario es necesario el uso de hardware específico que sea capaz de detectar la orientación de los ojos en tiempo real. Para ello se ha utilizado un *eye tracker* comercializado por Tobii, en concreto el modelo T60. Se trata de un monitor convencional que tiene acoplado en su parte inferior una serie de sensores cuya labor es rastrear la mirada del usuario.

Uno de los usos habituales de estos dispositivos es la recogida de datos estadísticos en interfaces de usuario, posibilitando la evaluación objetiva de los mismos. Un ejemplo es el estudio de una página web, donde la información recogida serían las áreas observadas por el usuario, el tiempo que ha permanecido en ellas y el orden de observación. Con estos datos se puede evaluar la visibilidad de la información más importante de la web, como podría ser el menú de navegación, así como el impacto real que tiene la publicidad incorporada en la página.

Existen otros usos de este hardware en campos como la medicina o la psicología. Concretamente, en este artículo se pretende explorar el uso de dichos dispositivos como interfaces hombre-máquina dentro del contexto de mundos virtuales.

### 2.3. Software

El esquema funcional del sistema se puede dividir en tres bloques: captura, actualización y visualización (Fig. 1).

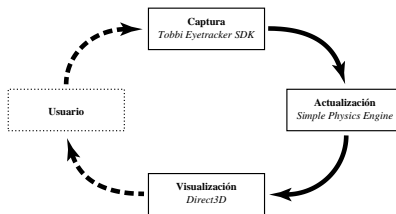


Fig. 1. Esquema general del sistema

Para la captura se ha utilizado el software proporcionado por Tobii, *Tobii Eyetracker SDK*, que consiste en una serie de librerías que permiten la adquisición de los datos capturados por el *eye tracker*.

Por lo que respecta a la actualización del mundo virtual en respuesta a los eventos generados por el usuario, es necesario una librería de física que simule el comportamiento de sólidos rígidos bajo la influencia de fuerzas como pueden ser la gravedad o la propia entrada del usuario. Para ello, se ha utilizado la librería *Simple Physics Engine* [10] que permite simular de forma bastante realista las interacciones entre los objetos del entorno virtual, a través de un sencillo interfaz de programación.

Finalmente, para la visualización de la escena se utilizó *Direct3D*.

## 3 Nuevas formas de interacción

El objetivo del sistema que proponemos es la incorporación de acciones que permitan la alteración del entorno virtual. En el sistema propuesto por Jacob [6] estas acciones existen, pero no en el ámbito de los entornos virtuales. En otros entornos virtuales [11] se contempla la observación del mundo pero no su modificación, por tanto la interacción se limita a la selección de objetos. A diferencia de otros usos similares [4], se ha diseñado un sistema desde cero tratando de adaptar el entorno al

interfaz y no el recíproco. Un elemento clave que ayuda a establecer un vínculo más fuerte entre lo que se ve en la escena y la mirada del usuario es la incorporación de un motor de física [12] en el sistema. Este elemento diferenciador hace que el usuario crea realmente que mueve algo *real* con los ojos y no representaciones icónicas de objetos.

El sistema desarrollado, bautizado con el nombre *EyeMotion*, consiste en un plano fijo donde caen una serie de dados (Fig. 2). Los objetos cambian a un color verdoso al ser observados por el usuario, de forma que éste sabe en todo momento cual es el objeto sobre el que actuará la herramienta antes de usarla. Si mientras está observando un objeto pulsa el botón de acción (el botón derecho del ratón), se inicia la acción correspondiente, que puede ser coger, empujar o disparar el objeto observado. El usuario puede cambiar de herramientas moviendo el cursor del ratón hasta la opción correspondiente de un menú mostrado en pantalla. Finalmente, para cambiar el punto de vista se utiliza el ratón como suele ser habitual en los programas de visualización de escenas.

En *EyeMotion* es posible mostrar la posición donde se mira en forma de puntero láser. Incluimos esta opción porque conocer la posición de la mirada ayuda en aquellos casos en los cuales la captura del *eye tracker* no es tan precisa como sería deseable. De esta forma el usuario puede hacer pequeñas correcciones en base a la posición mostrada por el cursor ocular.

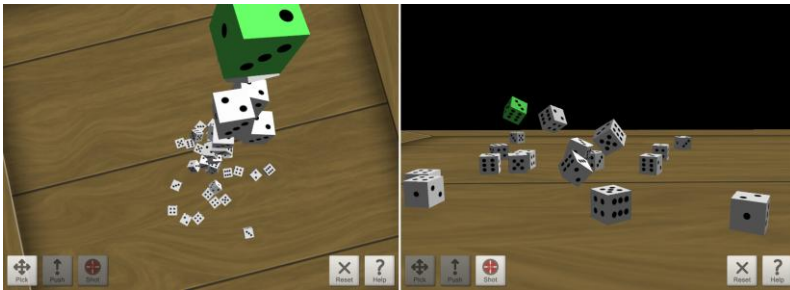


Fig. 2. EyeMotion

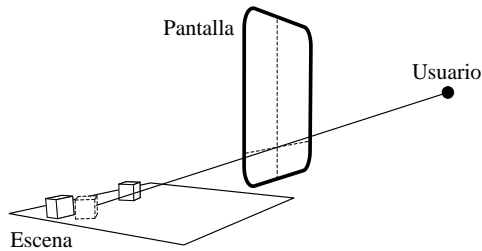
### 3.1. Selección

El problema de la selección en interfaces de seguimiento ocular es conocido como el problema del *toque de Midas*. Según la mitología griega, Midas era capaz de transformar todo lo que tocaba en oro. Algo parecido ocurre en este caso, ya que si no se toman medidas, se actuaría sobre todo lo que se mirase. Para solucionar este problema existen tres posibilidades:

- Usar guiños.
- Usar un tiempo de espera (*dwell time* [5]), que permite el inicio de una acción tras mantener la mirada sobre un objeto durante un tiempo determinado.
- Usar hardware, es decir un botón que comience y termine las acciones.

Descartamos usar guiños porque creemos que es una forma poco natural de activar herramientas y además puede ocasionar fatiga al usuario del sistema. Por otro lado, usar un tiempo de espera no se adapta a todas las herramientas que deseábamos implementar. En concreto, no es compatible con la herramienta de arrastre porque ésta necesita una forma de iniciar y terminar las acciones. La solución más simple es usar un botón que active la herramienta mientras esté presionado, y la desactive en su liberación.

Para la selección se utiliza la técnica de trazado de rayos (Figura 3) la que nos permite seleccionar el objeto que se encuentre más cercano a la cámara en el punto observado.



**Fig. 3.** Selección de objetos usando trazado de rayos

### 3.2. Arrastre

El arrastre permite al usuario coger objetos, moverlos y soltarlos en el lugar que se desee. El procedimiento consiste en mirar el objeto a mover, pulsar el botón de acción (con la herramienta arrastre seleccionada), mirar el lugar donde se desea dejar el objeto y soltar el botón. La física hará que el objeto caiga libremente hasta alcanzar una posición de reposo. Al ser posible girar la cámara mientras se tiene cogido un objeto, éste puede ser colocado en cualquier punto del espacio.

### 3.3. Empuje

La herramienta de empuje permite mover los objetos sin necesidad de observarlos previamente para seleccionarlos. Una vez activada la herramienta, los objetos se van apartando de los lugares observados siguiendo la dirección indicada por el movimiento de la mirada. La razón de ser de esta forma de interacción es que permite mover objetos de una forma bastante cómoda ya que no requiere preseleccionarlos previamente.

En general, para que los ojos puedan moverse suavemente es necesario que persigan a un objeto móvil [6]. A priori puede parecer que no tiene sentido una herramienta de empuje –que necesita movimientos suaves– en el contexto de un *eye tracker*, sin embargo esto no es así ya que una vez que un objeto está en movimiento sí que es posible continuar y modificar su movimiento siguiéndolo con los ojos. Además, como

veremos más adelante, es posible filtrar los movimientos oculares para dotarles de una mayor suavidad.

### 3.4. Disparo

La última forma de interacción consiste en disparar los objetos en la dirección en la que se mira la escena. Para disparar el objeto basta con mirarlo y pulsar el botón de acción. Tras pulsarlo, el objeto será lanzado en la dirección indicada por la normal al plano de visualización.

## 4 Filtrado de los datos

Para ver claramente un objeto es necesario mover el ojo de tal manera que la fovea quede centrada en el mismo. La fovea es la parte central de la retina, que gracias a su alta concentración de células fotorreceptoras sensibles al color, llamadas conos, permite observar el área de interés con el máximo detalle permitido por el sistema visual humano.

El movimiento que permite al ojo moverse de forma que un objeto quede en la fovea recibe el nombre de sacada. Después de una sacada se dice que el ojo se encuentra en estado de fijación. Durante dicho estado se producen una serie de movimientos involuntarios, llamados microsacadas, que desvían la mirada alrededor de la zona de interés, normalmente en un ángulo inferior a un grado [6].

El movimiento involuntario del ojo alrededor del punto observado hace que la mirada pueda caer fuera del objeto seleccionado, sin que el usuario sea consciente de que lo que está sucediendo realmente. El usuario piensa que está mirando dentro del objeto, y sin embargo el sistema selecciona y deselecciona el objeto observado, dificultando la interacción y restando control al usuario.

Para solucionar estos problemas, Jacob [6] propone detectar las fijaciones y considerar únicamente la posición de la mirada en el momento en el cual la fijación fue detectada. Tiene la desventaja de que se pierde la continuidad del cursor ocular ya que sólo considera la posición el momento de la fijación y no en las sacadas.

En [2] se da un paso más allá. Tras reconocer una fijación dentro del área de un objeto, se selecciona y se inicia un contador de tiempo. Una vez seleccionado, pequeños movimientos no deseleccionan el objeto, aunque la mirada caiga fuera del mismo. Cada vez que la mirada cae sobre el objeto el contador se reinicia, permitiendo de esta forma una selección más estable.

Para la realización del sistema propuesto en este trabajo se ha optado por extender la funcionalidad del método explicado en el párrafo anterior mediante el uso de un filtro que estabiliza la posición del cursor. Como las desviaciones producidas durante las fijaciones poseen las características de un paseo aleatorio [9], en media tienden a anularse unas con otras. Por tanto, mediar los datos para suavizarlos es una forma razonable de filtrar los datos. Para cada muestra  $k$ , se propone hacer la media ponderada con las  $n - 1$  muestras anteriores:

$$p'_k = \sum_{i=1}^{\min(k,n)} w_i p_{k-i+1}. \quad (6)$$

La Ecuación 1 nos permite incorporar unos pesos que den más importancia a unas muestras que a otras. Usando una función gaussiana para generar los pesos conseguimos dar más importancia a las muestras más recientes:

$$f(x) = e^{-\frac{x^2}{2\sigma^2}}. \quad (7)$$

Para conservar la escala de las posiciones es necesario que los pesos sumen 1. Por tanto, es necesario normalizarlos:

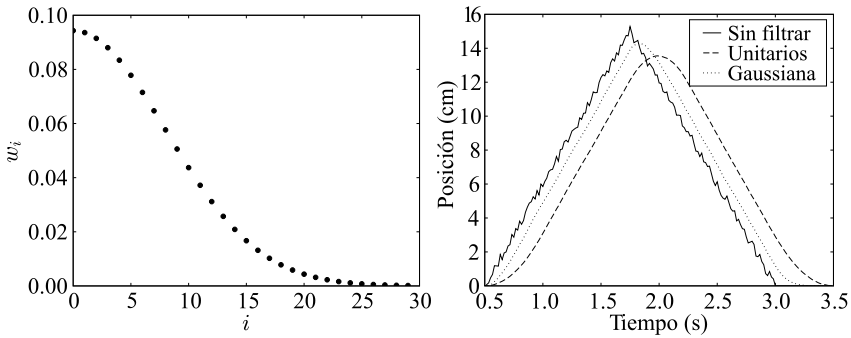
$$w_i = \frac{f(i)}{\sum_{i=1}^n f(i)}. \quad (8)$$

Como se puede apreciar en la Fig. 4 (izquierda), la curva dibujada por los pesos consta de tres partes bien diferenciadas: aceleración, velocidad constante y deceleración.

En la Figura 4, (derecha) puede verse el efecto de filtrar una serie de posiciones con ruido blanco usando pesos unitarios (media aritmética), y pesos generados por una gaussiana. Puede observarse que la gaussiana transforma las posiciones dotándolas de una aceleración más rápida, y que por tanto introduce un menor retraso entre la posición original y la posición filtrada. Por otro lado, la deceleración aportada por la gaussiana es más suave y aún con todo más rápida que en el caso de los pesos unitarios.

Mediante el uso de la técnica del contador [2] se consigue estabilizar la selección, y se facilita la interacción en aquellos casos en los que la precisión del *eye tracker* sea insuficiente, ya que no es necesario mirar dentro de un objeto para seleccionarlo, basta con pasar por encima de él. A través del filtrado presentado en esta sección se suavizan los movimientos bruscos de la mirada, consiguiendo de esta forma un cursor más estable.

Aunque sería necesario realizar un estudio objetivo, el filtrado presentado en esta sección ha demostrado ser una solución simple y lo suficientemente potente como para permitir una interacción satisfactoria.



**Fig. 4.** Izquierda: pesos usados en el cálculo de la media ( $\sigma^2 = 65$ ). Derecha: posiciones filtradas usando pesos unitarios y pesos determinados por una gaussiana

## 5 Análisis

### 5.1. Participantes

Seis empleados del Laboratorio de Usabilidad se ofrecieron a participar en el experimento. Todos ellos estaban familiarizados con el *eye tracker* y lo habían usado previamente para analizar interfaces mediante el software proporcionado por el fabricante. No obstante, como el software mencionado es de índole totalmente diferente al sistema que proponemos, este uso no supone un aprendizaje previo al experimento. Uno de los sujetos fue rechazado por problemas de calibración. El resto de los participantes (cuatro hombres y una mujer) pudieron completar el experimento satisfactoriamente.

### 5.2. Procedimiento

Los participantes dispusieron de tiempo ilimitado para entrenarse con las tres herramientas, en ambos tipos de interfaz (ratón y *eye tracker*). La calibración se realizó usando 5 puntos de referencia. Como la navegación en el entorno requiere práctica, se les aconsejó sobre la forma óptima de realizar las pruebas. Después del entrenamiento inicial, cada participante tuvo un descanso de aproximadamente 2 horas hasta el inicio del experimento. En el experimento, cada participante realizó tres veces cada una de las pruebas.

Las pruebas se realizaron en un tablero especial que consta de diferentes zonas numeradas del uno al tres (Figura 5). Se han dispuesto varias zonas para obligar al usuario a navegar por el entorno para completar su tarea. Para probar acciones de diferente precisión se han utilizado zonas de diferentes tamaños. Por otro lado, las zonas se han colocado en las esquinas para que el usuario tenga que tener cuidado de



no tirar los dados fuera del tablero. Al comenzar cada prueba, la cámara se coloca en una posición por defecto y comienzan a caer 25 dados. Las pruebas son las siguientes:

- Arrastre: el sujeto debe mover 6 dados a las zonas marcadas. El número de dados que debe llevar a cada zona se corresponde con su numeración.
- Empuje: el sujeto debe empujar 3 dados a la zona roja.
- Disparo: el sujeto debe sacar los 25 dados del tablero.

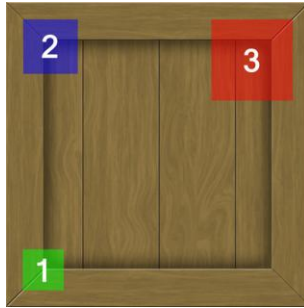


Fig. 5. Zona de pruebas

### 5.3. Resultados

Los resultados de nuestro estudio pueden dividirse en dos categorías:

- Medidas objetivas, las cuales nos permiten comparar la eficiencia y el aprendizaje del interfaz tradicional basado en ratón con el interfaz basado en la mirada (Figura 6).
- Medidas subjetivas, las cuales nos informan de la satisfacción de los usuarios en ambos tipos de interfaz, para cada herramienta (Cuadro 1) y en general (Cuadro 2).

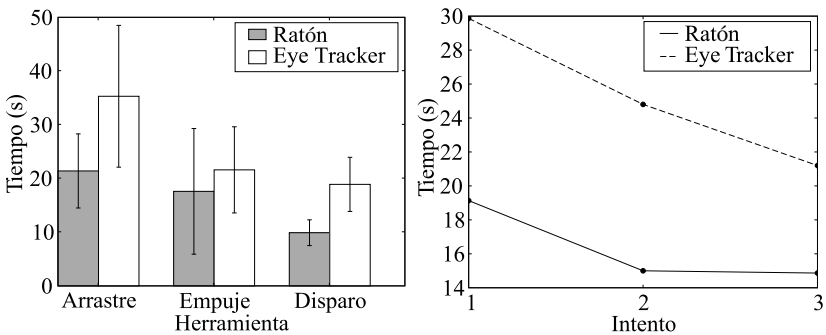


Fig. 6. Resultados del análisis objetivo

**Cuadro 4.** Resultados específicos del análisis subjetivo. ET (*eye tracker*). R (ratón)

Pregunta	Arrastre (%)		Empuje (%)		Disparo (%)	
	ET	R	ET	R	ET	R
¿Qué método te ha gustado más?	80	20	40	60	40	60
¿Cuál te ha parecido más fácil de aprender?	20	80	0	100	20	80
¿Cuál te ha parecido más fácil de usar?	60	40	20	80	60	40
¿En cuál te has sentido más inmerso?	100	0	100	0	80	20

**Cuadro 5.** Resultados generales del análisis subjetivo. ET (*eye tracker*). R (ratón)

Pregunta	ET (%)	R (%)
¿Has sentido fatiga al usarlo?	80	20
¿Crees que es suficientemente preciso?	40	60
¿Crees que podría ser útil en entornos virtuales?	0	100

## 6. Conclusiones

Los resultados objetivos nos dicen que la tecnología de rastreo ocular todavía necesita madurar. No es lo suficientemente precisa como para permitir seleccionar objetos distantes, obligando al usuario a acercarse a él con la cámara virtual. De hecho, en algunas personas el error producido puede ser de hasta un centímetro, impidiendo la selección incluso a distancias cortas.

Por otro lado, se puede ver que el aprendizaje es considerablemente mejor en el caso del *eye tracker*, si se tiene en cuenta el número de intentos versus el tiempo de ejecución. Esto puede ser debido a que se suele trabajar a diario con el ratón, de forma que ya se tiene una destreza en su uso, y esto influye en los resultados porque existe un aprendizaje previo a la realización del experimento. Sin embargo, los usuarios se encuentran interactuando mediante la mirada por primera vez, por lo que todavía están en el inicio de la curva de aprendizaje.

Desde el punto de vista subjetivo, se puede ver que no existen preferencias por ninguno de los dos interfaces. No obstante, se ha observado que las personas que mejores calibraciones realizaban, tendían a preferir el interfaz ocular. Desde el punto de vista de la inmersión, se han obtenido resultados que reafirman los obtenidos por Smith y Graham [4].

Como trabajo pendiente, resaltar que para conseguir la máxima satisfacción por parte de los usuarios es necesario enfrentarse al problema de la precisión. Éste podrá ser abarcado de dos formas diferentes, o bien mejorando la tecnología de rastreo, o bien desarrollando algoritmos que predigan lo que el usuario está observando realmente.

## Agradecimientos

Este trabajo ha sido financiado por el proyecto UZ2007-TEC06 (Universidad de Zaragoza) y TIN2007-63025 (Ministerio de Educación y Ciencia). Jorge Jiménez ha sido financiado por una beca de investigación del Instituto Tecnológico de Aragón.

Nos gustaría agradecer al Laboratorio de Usabilidad los medios materiales y personales que nos han proporcionado para el desarrollo de este artículo, especialmente a Elena Lafuente y David Eguizábal. De igual forma, a John Michael Guerrero por la textura utilizada en el suelo virtual.

## Referencias

1. Ware C. y Mikaelian H. H. An evaluation of an eye tracker as a device for computer input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems and Graphics Interfaces*, 1987.
2. Miniotas D. y Špakov O. An algorithm to counteract eye jitter in gaze-controlled interfaces. *Information Technology and Control*, 2004.
3. Salvucci D. D. y Anderson J. R. Intelligent gaze-added interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2000.
4. Smith J. D. y Graham T. C. Use of eye movements for video game control. In *Advances in Computer Entertainment Technology*, 2006.
5. Sibert L. E. y Jacob R. J. K. Evaluation of eye gaze interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2000.
6. Jacob R. J. K. What you look at is what you get: Eye movement based interaction techniques. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Empowering people*, 1990.
7. Cournia N., Smith J. D., y Duchowski A. T. Gaze vs. hand based pointing in virtual environments. In *CHI Extended Abstracts*, 2003.
8. Isokoski P. y Martin B. Eye tracker input in first person shooter games. In *Proceedings of COGAIN 2006: Gazing into the Future*, 2006.
9. Engbert R. y Kliegl R. Microsaccades keep the eyes balance during fixation. *Psychological Science*, 15, 2004.
10. Simple physics engine, <http://www.spehome.com>.
11. Tanriverdi V. y Jacob R. J. K. Interacting with eye movements in virtual environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2000.
12. Yeh T. Y., Faloutsos P., y Reinman G. Enabling real-time physics simulation in future interactive entertainment. In *Proceedings of the 2006 ACM SIGGRAPH symposium on Videogames*, 2006.