

Interacción humano-computador en escenarios educativos y artísticos. Kinect como propuesta viable

Julián Rodríguez*

Institución Universitaria Politécnico Grancolombiano. Bogotá - Colombia

FECHA DE ENTREGA: 29 DE JUNIO DE 2012
FECHA DE APROBACIÓN: 29 DE JULIO DE 2012

Resumen Las posibilidades y herramientas que ofrece un sistema computacional son cada vez más avanzadas, amplias y completas. Con el avance, en este sentido, las posibilidades de interacción con los sistemas computacionales modernos han venido ampliándose y tomando rutas y formas que hace apenas unos años no se consideraban plausibles. Desde los esquemas iniciales de entrada-salida de datos, pasando por el uso tarjetas perforadas o interruptores y alertas luminosas o sistemas de impresión, hasta el desarrollo del teclado y el *mouse*, y las interacciones que utilizan esquemas multitoque e interacciones naturales; la interacción humano-computador es un campo de crecimiento constante. Existe sin embargo un vacío en términos de esquemas de interacción para escenarios de creación artística y de enseñanza. En este trabajo se presenta una breve reseña de los requerimientos, características y necesidades de una solución a este vacío, y se propone la utilización de Kinect como dispositivo de interacción humano-máquina para los escenarios descritos, teniendo en cuenta su accesibilidad, robustez, facilidad de uso y modificación, así como la amplísima gama de posibilidades que ofrece.

Abstract The possibilities and tools offered by a computer system are becoming more advanced, thorough and comprehensive. With these advancements, the potential for interaction with modern computer systems have been expanding and taking routes and ways that a few years ago were not considered plausible. Since the initial input-output data outlines, through the use punch cards, switches, light alerts or printing systems, to the development of the keyboard and mouse, interactions using multi-touch and natural interactions diagrams, human-computer interaction is a field of constant growth. However, there is a gap in terms of interaction schemes for artistic creation and teaching scenarios. This paper provides a brief overview of the requirements, characteristics and needs of a solution to this gap, and proposes the use of Kinect as a device of human-machine interaction for the scenarios described, taking into account accessibility,

* El proyecto de investigación del cual es producto este documento, ha sido financiado por la Fundación Politécnico Grancolombiano Institución Universitaria, mediante el contrato de investigación No. 2011-FICB-SC-BC-62, del 1 de Junio de 2011. jerodrig@poli.edu.co

robustness, ease of use and modification, as well as the wide range of possibilities.

Palabras Clave: interfaz de usuario, interfaz hombre-máquina. software educativo, software artístico, Kinect.

Keywords: user interface, human-machine interface, educational software, artistic software, Kinect

1. Introducción

Actualmente no existe una definición del término “Interacción hombre - computador” con la que todos los estudiosos del campo estén de acuerdo. Sin embargo, y en aras de ofrecer una definición que permita establecer un piso común sobre el cuál basar el trabajo, a continuación presentado, se sugiere la siguiente definición: *La interacción hombre-computador es un espacio disciplinar dentro del que se estudian el diseño, la implementación y la validación de sistemas computacionales interactivos, orientados al uso por parte de seres humanos.*

En el escenario tradicional, el foco de estudio de la disciplina se ve orientado al trabajo alrededor de aplicaciones de *software* utilizando computadores y sus periféricos como intermediarios. Es evidente sin embargo, que con el crecimiento y ampliación de las posibilidades que en términos de *hardware* y *software* existen hoy en día, esta definición puede y de hecho debe, ampliarse.

Por otro lado, si se toman en consideración también las posibles variaciones en la definición de “seres humanos”, la situación se vuelve aún más compleja. El incluir por ejemplo grupos de personas que interactúen simultáneamente con un sistema de cómputo, dentro de la categoría de usuarios, permite entrar a definir un universo nuevo y amplísimo de posibilidades de interacción, con todas las implicaciones disciplinares que esto trae.

Es claro que independientemente de la definición puntual que se ofrezca del término, existen múltiples áreas particulares dentro del estudio de la interacción hombre-computador, que deben ser consideradas si se espera abordar dicho estudio de una manera más o menos comprehensiva. Por tanto, se encuentra, de manera evidente y esencial la comunicación entre el usuario y la máquina. Adicionalmente se puede nombrar la capacidad del usuario de utilizar de manera efectiva la máquina (punto estrechamente vinculado con la definición de usabilidad), y los procesos de *software* asociados al diseño y construcción de las interfaces que permiten la comunicación entre el usuario y la máquina. No obstante están también allí las preocupaciones asociadas con la representación visual, sonora y, en general, física de los resultados de los procesos de diseño de *software*. Y varios otros aspectos, enfoques y preocupaciones que atañen a la disciplina. Todos estos elementos construyen los resultados esperados de la labor conjunta entre el usuario y el sistema computacional, y describen los procesos para permitir y optimizar dicha labor.

Es evidente entonces con lo dicho, que el problema de la interacción hombre-computador es eminentemente multidisciplinar. Con sólo el atisbo que se ha dado

a sus posibles implicaciones y raíces, resulta claro que le atañen preocupaciones de índole sociológica, psicológica, estética, de diseño industrial, y prácticamente cualquier otro campo y disciplina que se requiera de acuerdo con la situación particular de estudio. Cuando se habla del campo de la interacción hombre-computador, la teoría de la comunicación, la lingüística, el diseño gráfico e industrial, se encuentran lado a lado con tópicos relacionados con los gráficos por computador, los sistemas operativos, las disciplinas de diseño, desarrollo y pruebas de *software*.

2. Estado actual, alternativas y consideraciones

El estado actual de las alternativas de interacción humano-computador es de dinámico crecimiento. Además de las alternativas que ya se consideran tradicionales, como el uso de teclados, ratones y otros dispositivos similares de entrada de información, las herramientas tecnológicas actuales han abierto la puerta a otras múltiples posibilidades. Es de notar que de la mano con estas posibilidades, viene la responsabilidad de valorar las alternativas que aparecen cuando se está llevando a cabo el proceso de diseño de una aplicación y sus herramientas de interacción. Para reforzar este punto, y a manera de ejemplo, a continuación se presentan algunas de las áreas y posibilidades que actualmente se encuentran en el centro de las posibilidades.

Interacción por voz. A pesar de que existen diferentes enfoques para la interacción a través de la voz, y a pesar de que una de las más conocidas sea la interpretación de comandos de voz, existen otras posibilidades que pueden representar alternativas interesantes. Existe por ejemplo, el concepto de *VoiceSites*. Pensado específicamente para países en vías de desarrollo y poblaciones rurales, los *VoiceSites* son equivalentes auditivos de *web sites*. El acceso a los mismos se da a través de una llamada telefónica, y los usuarios pueden escuchar la información disponible en el *VoiceSite*, y efectuar opciones de navegación a través de comandos de voz o la utilización del teclado del teléfono [22].

Interacción con usuarios con limitaciones. Las posibilidades que actualmente ofrecen las interfaces humano-computador, potenciadas por las facultades del *hardware* y el *software* sobre el que se apoyan, presentan un sinnúmero de escenarios en los que se pueden desplegar interacciones y procesos cognitivos de una sofisticación elevada. Sin embargo, en la mayor parte de las ocasiones no se tiene en cuenta a los usuarios que presentan algún tipo de dificultad física, ya sea de tipo visual o motora. Existen, no obstante, iniciativas que pretenden establecer precedentes en términos de lo que es necesario para ofrecer experiencias inmersivas y productivas a usuarios con limitaciones. Herramientas y facilidades para la navegación, retroalimentación y personalización del ambiente son fundamentales en situaciones como las descritas. Existen iniciativas de considerable peso y con trabajo constante que buscan determinar las principales necesidades y las mejores alternativas de solución en escenarios como los descritos [8,17,18].

Interacciones “naturales”. La simplificación de conceptos y procesos de interacción, adaptándolos de otros que se hacen de manera usual cuando no se está

interactuando con un sistema computacional, presenta fortalezas que se exploran con mucho interés. Procesos tan naturales —como el bocetar representaciones simplificadas de las ideas en una discusión o proceso de planeación o diseño— son portados a escenarios digitales buscando “naturalizar” la interacción de los usuarios con el computador. Alrededor de esta iniciativa particular, y en gran medida gracias a la ubicuidad de dispositivos portátiles como *tablets* (iPad favorita entre todas), existen desarrollos promisorios [6]. De la misma manera, tecnologías emergentes, apoyadas en paradigmas —como por ejemplo la interacción con *multitouch*, y la *surface computing*— ofrecen avenidas de investigación, renovación y exploración en los escenarios asociados al diseño y definición de interfaces humano-computador [25,27].

Desde luego, no se pretende que las áreas aquí enumeradas sean una representación completa de las posibilidades existentes. Sólo se usan como una muestra de las posibilidades que presenta el escenario que ocupa este escrito, pues se busca evidenciar la necesidad de evaluar que al emprenderse un proceso de conceptualización y diseño de un esquema de interacción humano-computador, es imprescindible considerar las características puntuales de los usuarios finales, tanto como las del ambiente donde se utilizará el *software* resultado del proceso de desarrollo. Este tipo de cuidado durante el proceso de construcción ofrecerá resultados mucho más cercanos a lo realmente deseable.

3. Aplicaciones en escenarios de interés

Dentro del contexto de lo que se busca en este documento, resulta importante entender que los cambios en los paradigmas de interacción han generado un despertar en torno a las posibilidades de aplicación de nuevas estrategias, especialmente en ambientes que previamente se veían limitados debido a sus requerimientos particulares. Dos de los que en particular resultan de interés, se presentan a continuación.

Aplicaciones en educación. La aplicación de nuevos paradigmas y tecnologías a los campos relacionados con la educación es una tendencia que demuestra hoy más que nunca ser productiva. Gracias a las posibilidades de acceso a tecnología por parte de las instituciones educativas, y de los estudiantes mismos, la posibilidad de integrar los avances técnicos presentes a los escenarios de formación es cada vez más grande. Si a esto se suma la posibilidad de utilizar otras herramientas que ya poseen los estudiantes, como por ejemplo la familiaridad con procesos y esquemas mentales asociados con escenarios lúdicos, se vislumbran posibilidades de generación y fijación de conocimiento que eran imposibles hace apenas unos años. Actualmente es posible utilizar redes sociales como marco conceptual y estructural para la educación, así como de manera más puntual se presenta la posibilidad de utilizar herramientas tecnológicas y conceptuales asociadas con juegos y consolas de juego a procesos cognitivos [4].

Aplicaciones en escenarios artísticos. Las nuevas tecnologías encuentran también terreno fértil en las exploraciones de tipo artístico. Gracias a la naturaleza misma de dichas exploraciones, es posible encontrar cabida a un sinnúmero de

posibilidades tecnológicas, que generan resonancias y expresiones antes imposibles, al utilizarse la tecnología. En espacios como la música [14], las artes visuales, incluso artes tan tradicionalmente “físicas” como la escultura, es posible encontrar por hoy aplicaciones de tecnologías emergentes. La penetración tecnológica es tal, que espacios previamente reservados para interacciones “serias” y formales se han abierto a las posibilidades de integración disponibles para los artistas que están dispuestos a experimentar con nuevas formas de hacer arte. Es así como es posible encontrar artistas que utilizan combinaciones de *hardware* y *software* para construir y/o ejecutar música tanto popular [13], como aquella que es considerada “cultura” y reservada para escenarios y formas más tradicionales [24]. Desde luego, expresiones artísticas más físicas, tales como la danza no se quedan atrás, y actualmente resulta cada vez más común encontrar experiencias que combinan las capacidades y habilidades tradicionales, con las oportunidades que ofrece la tecnología [15].

4. Consideraciones para encontrar la mejor alternativa

En un escenario como el planteado como marco de este proyecto, es evidente que las condiciones buscadas para la solución ideal son varias, y a pesar de que en su mayoría coinciden con las características de una solución de *software* común, existen elementos que resultan un poco diferentes a los que usualmente se consideran idóneos. A continuación se presentan algunas de las consideraciones que se tuvieron en cuenta al definir qué se busca.

Costo. En un espacio de educación o de creación artística, y más en un país en vía de desarrollo, resulta importante considerar el costo de la solución cuando se evalúan alternativas. Tanto si se busca contar con múltiples instancias de la solución para utilizarla dentro de un escenario de enseñanza-aprendizaje, como si se busca integrarla dentro de un proceso artístico (la mayor parte de los cuales, cubre todo los procesos de tipo experimental y tecnológico que no cuentan con financiación sólida), lo ideal es encontrar una alternativa que ofrezca soluciones sin que su costo resulte prohibitivo.

Facilidad de uso. A pesar de que el proceso mismo de aprendizaje e interiorización del uso de la solución puede considerarse parte y recompensa del proceso educativo y de creación, es evidente también que en la mayor parte de los casos y los escenarios se busca contar con una alternativa que no dificulte la obtención de resultados, sino que más bien facilite a los usuarios la adquisición de habilidades y conocimiento.

Familiaridad por parte de los usuarios. Estrechamente relacionada con el punto anterior, esta característica busca priorizar soluciones que permitan integrar en su uso o concepto elementos o paradigmas que sean familiares para los potenciales usuarios, de tal manera que la curva de aprendizaje resulte mucho menos empinada. La utilización o modificación de elementos o estrategias que sean utilizadas en otros escenarios y resulten familiares a los usuarios, es entonces un objetivo al que apuntar.

Facilidad de desarrollo de software y de soporte del mismo. Actualmente, en un escenario de desarrollo tecnológico, resulta prácticamente imposible ignorar la necesidad de contar con un componente de *software* que articule la funcionalidad de la solución propuesta. Bajo esta idea, la facilidad de llevar a cabo el desarrollo inicial de dicho componente, así como las oportunidades de contar con soporte correctivo y de actualización, resultan de suma importancia. Es este uno de los motivos por los cuales las comunidades de desarrollo de *software open source* son ampliamente favorecidas para participar en este tipo de iniciativas.

Facilidad de adquisición-construcción del hardware. De la mano con lo mencionado en el punto anterior, la posibilidad de utilizar componentes físicos (de ser necesario contar con una parte física para la solución) de fácil construcción o adquisición redondea una experiencia que satisfaga las necesidades planteadas, sin que el proceso de construcción sea más costoso, dispendioso y delicado para obtener los resultados esperados.

5. Propuestas y soluciones

Teniendo en cuenta lo presentado hasta ahora, el siguiente paso natural es tratar de decantar las necesidades y características mismas de los procesos cognitivos apoyen la solución de interacción propuesta. Es decir, —caracterizar en términos de lenguaje y las definiciones dadas hasta ahora— los procesos de educación y creación artística, encontrando —hasta donde sea posible— un terreno común entre los dos.

El resultado de este ejercicio es una lista de características que se encuentran de manera común en ambos espacios y que rigen, hasta cierto punto, la búsqueda de nuevas propuestas de interacción humano-computador:

- Los dos son procesos en los que se busca que las herramientas utilizadas se conviertan en una extensión del usuario, facilitando su accionar sin convertirse en muletas evidentes.
- Los dos son escenarios en los que se construye conocimiento (si se acepta la idea de la creación de una obra de índole artística como la creación o reestructuración de un conjunto de ideas).
- Apoyándose en lo anterior, es posible afirmar que, en los dos casos, se busca materializar, de alguna manera, el resultado del proceso cognitivo del usuario, de manera que sea visible para espectadores, estudiantes o docentes.
- Las herramientas utilizadas, en ambos casos, deben ser lo suficientemente flexibles como para adaptarse a diferentes estructuras de pensamiento y creación, pero de uso natural.
- Dadas las características generales del usuario objetivo, la interacción debería darse de la manera menos invasiva y técnica posible. El tránsito hacia el uso efectivo de las herramientas debería ser sencillo.

A la luz de lo expuesto hasta el momento, la solución buscada presenta un conjunto de características que la ponen fuera del contexto de lo que hasta el momento se encuentra disponible. La ruta que se presenta como más promisoría

es un conjunto de desarrollos de *hardware* y *software* que una varias alternativas de reconocimiento de patrones y movimientos, con un sofisticado sistema de integración con *software*, ofreciendo a la vez una API (*Application Programming Interface*) que facilite los desarrollos posteriores que utilicen la solución creada. Todo esto sustentado con una interfaz que permita una fácil apropiación de su funcionamiento por parte de usuarios sin un perfil eminentemente técnico. Si se añade a lo expuesto un costo razonable y una comunidad amplia de usuarios y programadores que soporten desarrollos futuros, la situación se antoja difícil. Afortunadamente, la dificultad es solo aparente, pues una solución con estas características existe en el mercado, y ha existido desde el 2010. Kinect de Microsoft ofrece un paquete que incluye las características planteadas, a un precio muy razonable, y con una comunidad de personas interesadas en su uso y modificación constante. Estas características lo hacen altamente promisorio en el contexto que interesa a este artículo.

6. Kinect como alternativa

Kinect es un dispositivo sensor de movimiento desarrollado por Microsoft para ser utilizado de manera inicial como complemento para su consola de juegos Xbox 360. Introducido en 2010 representó un punto de quiebre en las estrategias de interacción con consolas de juego, pues eliminó en gran medida la necesidad de utilizar dispositivos físicos para controlar los juegos, planteando en su lugar la idea de utilizar el cuerpo del jugador y sus movimientos como elementos de control. Más aún, componentes de reconocimiento facial y de control por voz añaden otra capa de posibilidades a las interacciones posibles.

Kinect fue el fruto de años de trabajo al interior de Microsoft y algunas de sus compañías subsidiarias. Además de ser un éxito en términos de ventas en los primeros 60 días, Kinect impulsó un cambio de paradigma en el escenario de los juegos en ambiente doméstico. A pesar de que el término oficial para las funciones que realiza es “detección de cuerpo completo”, un término que aproxima mejor su funcionalidad es “Interacción natural humano-computador”. Sin la necesidad de aprender el manejo y características de un control físico, la curva de aprendizaje para los juegos y demás actividades se reduce considerablemente [7].

Debido a esta facilidad, y los espacios que abre, Kinect se presenta como un ejercicio revolucionario en términos de las posibilidades de interactuar con un sistema de cómputo. A pesar de que inicialmente se pensó en el dispositivo como una herramienta de control exclusiva para consolas de juego, pronto fue evidente que el interés de utilizarlo en otros escenarios existía en los usuarios.

Al reconocer las posibilidades que este espacio ofrecía, Microsoft elaboró un kit de desarrollo de *software* para Kinect, facilitando que desarrolladores integraran el dispositivo en aplicaciones Windows. El éxito de esta iniciativa ha sido enorme, y ha conducido a la creación de experimentos revolucionarios en términos de la integración de interacciones naturales en diversos escenarios. Hasta ahora la mayor parte de dichos espacios han sido experimentales y simples, pero futuras interacciones en escenarios más formales y organizados definitivamente

se encuentran en el reino de las posibilidades y los intereses de Microsoft, así como de un gran número de compañías e individuos. A partir del momento en que los *drivers* para Windows se hicieron disponibles, e incluso desde antes, los experimentos abundaron. Existen iniciativas para integrar Kinect con dispositivos robóticos iniciados por miembros del MIT [2], aplicaciones en el terreno del *body scanning* que utiliza múltiples Kinects [1,23], aplicaciones fuertemente centradas en lo pragmático, como iniciativas para controlar Windows a través de Kinect [12], e incluso para labores tan prosaicas como controlar un carrito de supermercado utilizando Kinect [26]. También existen avances en torno al desarrollo de aplicaciones médicas como el diagnóstico de casos de autismo, o desorden obsesivo-compulsivo [11], así como el uso del sensor en aplicaciones de radiología e imágenes médicas [21].

Aplicaciones en escenarios educativos son frecuentes también, yendo desde aplicaciones en la enseñanza de vocabulario y reconocimiento de objetos en niños, [16], hasta iniciativas orientadas a la enseñanza de programación [20], generando tanto interés, que existen comunidades dedicadas a la exploración de posibilidades y el intercambio de recursos orientados a la utilización de Kinect en el aula de clase [3].

Desde luego, aplicaciones en el escenario de las artes no se hicieron esperar, y también aparecieron proyectos como títeres virtuales controlados a través de Kinect [9], así como aplicaciones de dibujo en 3D controladas a través de gestos [5].

Bien sea visto desde una perspectiva creativa o educativa, o como un avance pragmático y prosaico, es evidente que Kinect y sus posibilidades están aquí para quedarse, al menos en el futuro cercano. Dadas las múltiples posibilidades que ofrece, su accesibilidad en términos económicos (el dispositivo físico en su versión para Windows puede conseguirse en la fecha de este documento por US \$230) y tecnológicos (los *drivers* oficiales para implementar desarrollos sobre ambiente Windows pueden descargarse de manera gratuita desde el sitio web <http://www.microsoft.com/en-us/kinectforwindows/>), y teniendo en cuenta además la comunidad que ha crecido a su alrededor (para mencionar un sitio y un grupo de desarrolladores, ver <http://www.kinecthacks.com/>), sería un error no aprovechar al máximo este artefacto que ha revolucionado en muy poco tiempo la manera en la que es posible interactuar con un sistema de cómputo, y cuya influencia parece ser un fenómeno que se extenderá en el futuro.

7. Presentación del trabajo realizado

Con base en lo expuesto hasta el momento, y pretendiendo poner en práctica lo predicado, se emprendió un proceso de desarrollo con miras a generar herramientas que faciliten la utilización de Kinect y las alternativas que ofrece en los campos que atañen a este documento.

A continuación se presentan las consideraciones a lugar, y las decisiones tomadas.

7.1. Consideraciones técnicas

Una vez decidida la ruta en términos de *hardware*, resultó necesario definir qué complemento de *software* sería el que mejor resultado ofrecería. A pesar de que existen alternativas *open source* que ofrecen funcionalidades comparables —y en algunos escenarios incluso mejores que los *drivers* oficiales, como por ejemplo OpenNI [10]— las ventajas que ofrece la compatibilidad asegurada y las actualizaciones estables que solamente son posibles al conocer por completo el *hardware* sobre el que se ejecutará el *software*, hicieron que la decisión final fuera utilizar los *drivers* oficiales ofrecidos por Microsoft.

La última versión al momento de escribir este artículo es la 1.0.3.191. Esta versión, publicada el 2 de mayo de 2012 y que se encuentra disponible en <http://www.microsoft.com/en-us/kinectforwindows/develop/overview.aspx>, y ofrece compatibilidad completa con la versión del *hardware* para Windows disponible en el mercado.

En términos de las consideraciones acerca del entorno de desarrollo, con el fin de aprovechar las posibilidades ofrecidas por los *drivers* oficiales, se hizo uso de Visual Studio® 2010 Express, utilizando el .NET Framework 4.0.

Con el fin de facilitar aún más el desarrollo de los componentes visuales de la aplicación de ejemplo desarrollada, se utilizó OpenFrameworks [19], un muy popular *toolkit open source* que ofrece funcionalidades gráficas y de procesamiento, encapsuladas para ofrecer facilidades de uso a públicos con perfil no técnico.

7.2. Resultados del desarrollo

Una de las propuestas iniciales para este proyecto de investigación era la implementación de la interfaz de interacción para una aplicación previamente existente utilizando lo encontrado a lo largo del proceso descrito en este documento. Sin embargo, y teniendo en cuenta la amplitud de las posibilidades que ofrece Kinect y la gran flexibilidad de los desarrollos de *software* posibles con éste, se decidió implementar una manera de integrar de forma rápida y sencilla las funcionalidades ofrecidas por Kinect en una aplicación escrita en C++. De esta forma, las posibilidades de utilizar el dispositivo se abren para nuevos desarrollos, si así se desea. Se consideró que es este un resultado mucho más útil y flexible que la simple demostración de las capacidades de Kinect. Para conseguir lo propuesto, se desarrolló un *wrapper* que, apoyándose en los *drivers* oficiales, encapsula las llamadas más frecuentes al dispositivo y permite el desarrollo de aplicaciones que integren Kinect a su flujo de acción. El centro de dicho *wrapper* es la clase `pgKinect`, que con métodos como `getRGBBufferBits()`, `getDepthBufferBits()` y `getSkeletonFrame()` da acceso a la información recolectada por Kinect de una manera sencilla y rápida. Una parte del diagrama de clases de la aplicación, mostrando la clase `pgKinect`, se presenta en la figura 1.

Para mostrar el *wrapper* desarrollado, y con el fin de demostrar la facilidad de su utilización, se implementó una aplicación de prueba que, utilizando Kinect permite controlar el volumen y el *pitch* de dos archivos de audio, según la posición



Figura 1. Parte del diagrama de clases, que muestra la clase `pgKinect`.

de las manos del usuario. En la figura 2 se presenta una captura de pantalla de la aplicación en acción.

El código fuente de la aplicación y el *wrapper* pueden ser solicitados al departamento de investigación del Politécnico Grancolombiano Institución Universitaria, para su utilización en proyectos de desarrollo de *software*.

8. Trabajo futuro

Dentro de los trabajos recomendados a futuro existen dos rutas que se consideran de importancia: el desarrollo de aplicaciones que aprovechen el *near mode* de



Figura 2. Captura de pantalla de la aplicación de ejemplo

Kinect, y la integración y utilización de *drivers* de terceras partes. El trabajo en el primer punto permitirá contar con aplicaciones que no funcionen mediante la detección de todo el cuerpo del usuario a distancias mayores a un metro, sino que funcionen con base en la detección de los dedos de la mano del usuario, a cortas distancias. Este tipo de desarrollos abre posibilidades muy interesantes en torno a la interacción fina y de detalle elevado que no resulta tan fácil cuando se utiliza todo el cuerpo del usuario.

El segundo grupo de desarrollos ofrecerá posibilidades que, a pesar de no ser sancionadas de manera directa por Microsoft, pueden abrir espacios de experimentación apoyados por la comunidad *open source*, que exploten capacidades del *hardware* de maneras no previstas con anticipación.

Desde luego, el refinamiento constante del *wrapper*, así como la ampliación de la colección de aplicaciones disponibles y su utilización, en ambientes reales de tipo educativo y de creación, son altamente recomendados como desarrollos continuos a futuro.

Referencias

1. [TC]2.: 3d Body scanning & technology development. [en línea] Recuperado el 18 de 04 de 2012, de TC2: http://www.tc2.com/index_3dbodyscan.html (2011).

2. The Kinect Sensor in Mobile Robotics: Initial Experiments. [en línea] Recuperado el 18 de 04 de 2012, de Youtube: http://www.youtube.com/watch?feature=player_embedded&v=dRPEns8MS2o (2010).
3. Kinect Education. [en línea] Recuperado el 10 de 04 de 2012, de Kinect Education: <http://www.kinecteducation.com/> (2011)
4. Ardito, C., Costabile, F., Lanzilotti, R.: Gameplay on a multitouch screen to foster learning about historical sites. Proceedings of the International Conference on Advanced Visual Interfaces (AVI '10) (págs. 75-78). New York: ACM. (2010).
5. Atken, M.: ofxKinect 3D draw. [en línea] Recuperado el 12 de 04 de 2012, de [www.vimeo.com](http://www.vimeo.com/16818988): <http://www.vimeo.com/16818988> (2010).
6. Bellamy, R., Desmond, M., Martino, J., Matchen, P., Ossher, H., Richards, J., et al: Sketching tools for ideation. Proceedings of the 33rd International Conference on Software Engineering (ICSE '11). New York: ACM. (2011).
7. Business Value magazine. Kinect and Future of Human-Machine Interaction. [en línea] Recuperado el 18 de 4 de 2012, de Microsoft: http://www.microsoft.com/china/ard/en/innoforum/innoforum_24.msp (2011).
8. Dawe, M.: Desperately seeking simplicity: how young adults with cognitive disabilities and their families adopt assistive technologies. Proceedings of the SIGCHI conference on Human Factors in computing systems (CHI '06) (págs. 1143-1152). New York: ACM. (2006).
9. Design I/O.: Puppet Parade – Interactive puppetry experience. [en línea] Recuperado el 05 de 04 de 2012, de design-io.com: http://design-io.com/site_docs/work.php?id=15 (2011).
10. DotNetNuke Corporation.: OpenNI. [en línea] Recuperado el 02 de 05 de 2012, de <http://www.openni.org/> (2011).
11. Dumitrescu, A.: Minnesota University Team Adapts Kinect for Medical Use. [en línea] Recuperado el 17 de 04 de 2012, de <http://news.softpedia.com/news/Minnesota-University-Team-Adapts-Kinect-for-Medical-Use-189553.shtml> (2011).
12. Evolvece. WIN&I Gesture Control Software for Kinect and Windows. [en línea] Recuperado el 18 de 04 de 2012, de http://www.evolvece.com/_win-and-i/en/software/overview/index.php (2012).
13. Flatley, J. L.: Kinect hack turns you into a punching, waving MIDI controller. [en línea] Recuperado el 17 de Abril de 2012, de <http://www.engadget.com/2011/01/03/kinect-hack-turns-you-into-a-punching-waving-midi-controller/> (2011).
14. Jordà, S., Geiger, G., Alonso, M., Kaltenbrunner, M.: The reacTable: exploring the synergy between live music performance and tabletop tangible interfaces. Proceedings of the 1st international conference on Tangible and embedded interaction (TEI '07) (págs. 139-146). New York: ACM. (2007).
15. Keene, J.: Watch this: Kinect turns a dancer into a 22,000-point musical sculpture. [en línea] Recuperado el 17 de Abril de 2012, de The Verge: <http://www.theverge.com/2012/4/1/2918170/watch-this-kinect-dancer-musical-sculpture> (2012).
16. Khan, U.: HCI Project - Using gesture and speech with Kinect for education. [en línea] Recuperado el 16 de 4 de 2012, de <http://www.cs.utoronto.ca/~uzmakhan/HCIProject.html> (2011).
17. Ko, E., Ju, J.S., Kim, E.Y.: Situation-based indoor wayfinding system for the visually impaired. The proceedings of the 13th international ACM SIGACCESS conference on Computers and accessibility (ASSETS '11) (págs. 35-42). New York: ACM. (2011).

18. Laff, M., Hanson, V., Cavender, A., Trewin, S.: Exploring Visual and Motor Accessibility in Navigating a Virtual World. *ACM Transactions on Accessible Computing (TACCESS)*, 35-39. (2009).
19. Lieberman, Z.: openFrameworks. [en línea] Recuperado el 05 de 05 de 2012, de <http://www.openframeworks.cc/> (2012).
20. Murchu, I. O.: Stephen Howell: Using Scratch with Kinect for Education. [en línea] Recuperado el 22 de 03 de 2012, de New Tech Post: <http://newtechpost.com/2011/07/11/stephen-howell-using-scratch-with-kinect-for-education> (2012).
21. radRounds Radiology Network.: Xbox Kinect-based radiology & medical image exploration. [en línea] Recuperado el 14 de 04 de 2012, de <http://www.radrounds.com/video/xbox-kinect-based-radiology-medical-image-exploration> (2012).
22. Sheetal, A., Anupam, J., Arun, K., Priyanka, M., Nitendra, R.: Spoken Web: creation, navigation and searching of VoiceSites. *Proceedings of the 16th international conference on Intelligent user interfaces (IUI '11)* (págs. 431-432). New York: ACM. (2011).
23. TC2.: Multiple Kinects - 3D Body Scanning Demo by [TC]2. [en línea] Recuperado el 18 de 4 de 2012, de Youtube: http://www.youtube.com/watch?v=9MwS_nk9n2A&feature=related (2011).
24. Vik, C.: Kinectar. [en línea] Recuperado el 17 de Abril de 2012, de Kinectar: <http://kinectar.org/> (2012).
25. Wigdor, D.: Architecting next-generation user interfaces. *Proceedings of the International Conference on Advanced Visual Interfaces (AVI '10)* (págs. 16-22). New York: ACM. (2010).
26. Wingfield, N.: Microsoft Ponders Future of Kinect, Tech. [en línea] Recuperado el 03 de 04 de 2012, de The New York Times: <http://bits.blogs.nytimes.com/2012/02/28/microsoft-ponders-future-of-kinect-tech/> (2012).
27. Wobbrock, J. O., Morris, M., Wilson, A. D.: User-defined gestures for surface computing. *Proceedings of the 27th international conference on Human factors in computing systems (CHI '09)* (págs. 1083-1092). New York: ACM. (2009).