

# DISEÑO Y DESARROLLO DE UN SISTEMA DE INTERACCIÓN PARA SU IMPLEMENTACIÓN EN UN AULA DE CLASE INTELIGENTE

 CHRISTIAN ANDRÉS DÍAZ LEÓN<sup>1</sup>

EDWIN MAURICIO HINCAPIÉ MONTOYA<sup>2</sup>

EDISON ANDRÉS GUIRALES ARREDONDO<sup>3</sup>

GUSTAVO ADOLFO MORENO LÓPEZ<sup>4</sup>

## RESUMEN

Las aulas de clase inteligentes son un nuevo paradigma educativo cuyo objetivo es determinar mediante un conjunto de sensores que está sucediendo dentro del aula, y a partir de esta información —y usando estrategias de inteligencia artificial— inferir qué variaciones metodológicas o de contenido deben ser aplicadas por parte del docente, con el fin de optimizar las prácticas pedagógicas. Un componente importante de estas aulas es el uso de nuevas estrategias de interacción, que facilitan el manejo de los contenidos pedagógicos utilizados. En este artículo se presenta el diseño y desarrollo de un sistema de interacción, que usa diferentes estrategias para interactuar con los contenidos propuestos por el aula de clase inteligente. Estas estrategias de interacción están basadas en gestos, superficies interactivas no instrumentadas y sistemas de pulsación gestual. Finalmente, el artículo propone pruebas experimentales para validar las diferentes formas de interacción propuestas.

**PALABRAS CLAVE:** Aulas de clase inteligente; interacción hombre – computador; reconocimiento de gestos; superficies interactivas.

## DESIGN AND DEVELOPMENT OF AN INTERACTION SYSTEM IN ORDER TO BE IMPLEMENTED IN A SMART CLASSROOM

## ABSTRACT

Smart Classrooms are a new educational paradigm aimed to determine, by a set of sensors, what is happening in the classroom and from this information and using artificial intelligence strategies infer what methodological variations

---

<sup>1</sup> Ingeniería Biomédica, Magíster en Ingeniería Informática, Candidato a Doctor en Ingeniería. Docente Investigador, Institución Universitaria Salazar y Herrera.

<sup>2</sup> Ingeniero en Instrumentación y Control, Magíster en Matemáticas Aplicadas, Doctor en Ciencias de la Ingeniería. Director del Centro de Investigación, Institución Universitaria Salazar y Herrera.

<sup>3</sup> Estudiante Ingeniería de Sistemas Institución Universitaria Salazar y Herrera.

<sup>4</sup> Licenciado en Educación, Magíster en Educación. Director Académico Institución Universitaria Salazar y Herrera.



*Autor de correspondencia:* Díaz León, C.A. (Christian Andrés): Calle 70 Sur No. 38 - 305, Apto 1906, Conjunto Loma Linda, Sabaneta, Antioquia / Tel.: (574) 2888016. Correo electrónico: c.diaz@iush.edu.co

*Historia del artículo:*

Artículo recibido: 27-IV-2015 / Aprobado: 14-X-2016

Disponible online: 30 de febrero de 2017

Discusión abierta hasta abril de 2018



or content must be applied by the teacher, in order to optimize the pedagogical practices. An important component of these classrooms is the use of new interaction strategies that facilitate the use of educational content. This paper describes the design and development of an interaction system, which uses different strategies to interact with contents proposed for the smart classroom. These interaction strategies are based on gestures, interactive surfaces and gestural touch. Finally, the article proposes experimental tests to validate the different forms of interaction described.

**Keywords:** Smart classrooms; Human computer interaction; Gesture recognition; Interactive surfaces.

## DESENHO E DESENVOLVIMENTO DE UM SISTEMA DE INTERAÇÃO PARA SUA IMPLEMENTAÇÃO NUMA SALA DE CLASSE INTELIGENTE

### RESUMO

As salas de classe inteligentes são um novo paradigma educativo que visa determinar mediante um conjunto de sensores o que está acontecendo na sala de aula, e a partir dessa informação e utilizando estratégias de inteligência artificial inferir quais variações metodológicas ou de conteúdo devem ser aplicadas pelo professor, para otimizar as práticas pedagógicas. Um componente importante destas aulas é o uso de novas estratégias de interação que facilitam o uso de conteúdos pedagógicos utilizados. Neste artigo apresenta-se o desenho e desenvolvimento de um sistema de interação, que utiliza diferentes estratégias para interagir com conteúdos propostos para a sala de aula inteligente. Estas estratégias de interação estão baseadas em gestos, superfícies interativas não instrumentadas e sistema de pulsação gestual. Finalmente, o artigo propõe testes experimentais para validar as diferentes formas de interação descritas.

**PALAVRAS-CHAVE:** Salas de aula inteligentes; Interação homem-computador; Reconhecimento de gestos; Superfícies interativas.

### 1. INTRODUCCIÓN

Las aulas de clase inteligentes tienen como objetivo mejorar las actividades pedagógicas que se están realizando dentro de estas, mediante el análisis del contexto del aula de clase y su adaptación usando diferentes formas de presentación del contenido y/o variando las metodologías de enseñanza utilizadas (Chen y Huang, 2012; Papatheodorou *et al.*, 2010). Cuando hablamos de contexto, en este sentido, nos referimos a todas aquellas métricas que nos permiten determinar qué está sucediendo dentro del aula de clase, por ejemplo, el tipo de actividad que se está realizando en el aula de clase (clase magistral o trabajo en grupo), la disposición o nivel de atención de los estudiantes respecto a la clase, entre otros (Chen

y Huang, 2012). A partir de esta información un aula de clase inteligente determina qué debe ser mejorado en el aula de clase y se lo recomienda al docente. Estas recomendaciones pueden ir desde un cambio en la práctica de enseñanza hasta modificar la forma de presentar el contenido (por ejemplo, usando diapositivas o realidad aumentada).

Un componente muy importante para estos sistemas es aquel que ofrece diversas formas de interacción con el contenido, como por ejemplo, los sistemas de reconocimiento de gestos (Bailly *et al.*, 2012). Hoy en día estas aulas de clase inteligentes proponen el uso de diferentes formas de interacción del docente y el estudiante con los contenidos, con el fin de ofrecer formas de enseñanza más vivenciales

que refuerzan la memoria y el entendimiento de conceptos complejos y abstractos. Se puede considerar que los nuevos paradigmas de interacción se agrupan en dos grandes categorías: (i) aquellos que requieren una interacción física entre el usuario y el sistema y (ii) aquellos que no requieren de una interacción física entre el usuario y el sistema. Con sistema nos referimos al aula de clase inteligente y los contenidos, y con interacción física nos referimos a que el usuario entra en contacto físico con algún componente del sistema para indicarle su intención o lo que quiere hacer.

Varios trabajos han propuesto el uso de superficies multi-táctil, las cuales entran dentro de la primera categoría de interacción, como principio para la interacción entre el usuario y los contenidos expuestos por el aula inteligente (Bailly *et al.*, 2012; Zhao *et al.*, 2014; Nacher *et al.*, 2014; Novotny *et al.*, 2013). En Nacher *et al.* (2014) se describe el desarrollo de aplicaciones que usan superficies táctiles como mecanismo de aprendizaje para chicos de pre kínder (2 a 3 años de edad), los cuales muestran una asimilación mejor en la coordinación y relación de ideas tras haber experimentado entrenamiento con estas tecnologías. También se han combinado diferentes tipos de interacción con formas de presentación (Novotny *et al.*, 2013); en este caso los autores utilizan sistemas táctiles con realidad aumentada para la visualización de patrimonios virtuales, como una alternativa de aprendizaje acerca de los acontecimientos y lugares históricos ya desaparecidos en determinados lugares en ciudades. Otros trabajos, aunque en menor cantidad, se han enfocado a aplicar el segundo tipo de interacción descrito en este artículo. Por ejemplo, en Seo y Yeol (2013) se propone una solución de realidad aumentada que mediante el uso del Kinect permite manipular objetos multimedia. Por otra parte en Krejov *et al.* (2014) se describen varios tipos de aplicaciones que no hacen uso de superficies táctiles para interactuar con un sistema, y plantea que este tipo de interacción es mucho más intuitiva y fácil de aprender para el usuario.

Sin embargo, seleccionar un tipo de interacción para un contenido específico está muy relacionado con la tarea que se desea realizar. Por esta razón, este artículo propone el uso de paradigmas de interacción de las dos categorías para ser implementados en el sistema de interacción de un aula de clase inteligente. El sistema de interacción propuesto en este artículo esta compuesto por tres módulos que permiten a los usuarios interactuar con los contenidos usando tres paradigmas diferentes. El módulo de gestos permite reconocer diferentes gestos y mapearlos a una acción determinada en el aula inteligente. Este módulo fue descrito en el trabajo realizado por los autores en Autor *et al.* (2015), donde también fue reportada su validación como sistema de interacción en una aula inteligente. El módulo de pulsación permite interactuar con el contenido sin tener que tocar superficies, solo basta con desplazar la mano y posteriormente realizar un movimiento como si se quisiera presionar un botón. Para este propósito este módulo integra varias librerías para el reconocimiento de datos de profundidad descritas en Seo y Yeol (2013) y Al Delail *et al.* (2012). El módulo de superficies interactivas permite realizar una interacción mucho más natural, ya que la persona tocará la superficie donde será proyectada la imagen con la que va a interactuar, teniendo la oportunidad de hacer zoom, desplazar y rotar, solo utilizando las manos (Bailly *et al.*, 2012; Zhao *et al.*, 2014). Vale la pena mencionar que estas superficies no son instrumentadas. Este módulo tiene la característica de monitorear las manos de la persona en todo momento y sus dedos, ya que a partir de estos el usuario puede realizar comandos para la manipulación del contenido, como por ejemplo, hacer click en algún botón, realizar zoom, señalar o activar opciones. También monitorea el nivel de profundidad de la mano (Novotny *et al.*, 2013; Krejov *et al.*, 2014). A diferencia de lo planteado en el estado del arte, este artículo concatena los diferentes módulos en un sistema de interacción dando la sensación al usuario de una interacción con el sistema de forma intuitiva.

## 2. MATERIALES Y MÉTODOS

### *Descripción del sistema de interacción*

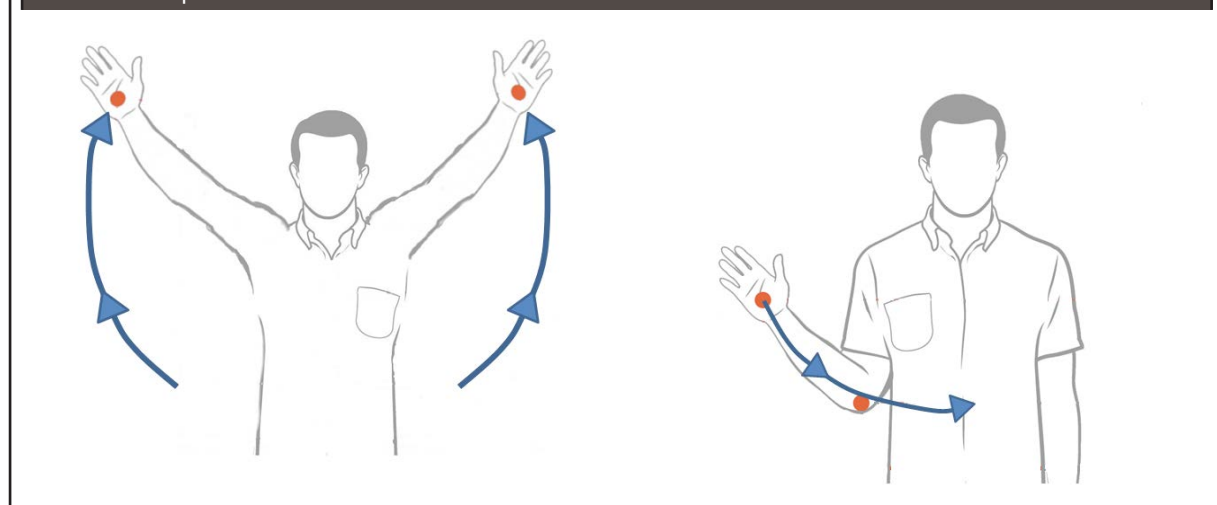
El rol que el sistema de interacción juega dentro de un aula de clase inteligente es permitir que los estudiantes y el docente puedan interactuar con los contenidos establecidos para la enseñanza de una forma fácil y natural. Con las tres formas de interacción propuestas en este artículo el usuario podrá mover, activar, seleccionar los diversos contenidos de la forma más apropiada. En esta sección se describirán los tres sistemas que conformarán los módulos de las aulas interactivas que se proponen en este artículo.

El primer módulo de interacción propuesto es el módulo de reconocimiento de gestos, el cual está encargado de caracterizar los movimientos previamente grabados (calibración de gestos) y posteriormente con base en la trayectoria de estos movimientos, reconocer los gestos o movimientos de la persona. Esto se logra aplicando un algoritmo matemático denominado *Dynamic Time Warping* (DTW) que nos permite tanto hacer la comparación como realizar los cálculos estadísticos para el reconocimiento de gestos sin importar la altura, constitución

física y género de la persona, ya que son factores diferenciadores que inciden en los movimientos de cada humano (Seo y Yeol, 2013; Krejov *et al.*, 2014). La implementación de esta técnica ayuda a que el procesamiento de la información captada por la cámara, y los datos de calibración tengan un tiempo de respuesta mucho más rápido, evitando reglas de inferencia matemáticas que nos puede llevar a un procesamiento de datos más lento. Posteriormente el movimiento es homologado con ciertos comandos para interactuar con módulos implementados en Unity3D (Goldstone, 2011). Unity3D es una plataforma para el desarrollo de aplicaciones en realidad virtual, realidad aumentada, videojuegos y visualización, y es el componente del aula de clase inteligente en el cual se despliegan los diferentes tipos de contenidos. En la **Figura 1** se presentan dos tipos de gestos que pueden ser realizados dentro del aula de clase, y el sistema de reconocimiento de gestos es capaz de reconocerlos y mapearlos a un comando para interactuar con los contenidos.

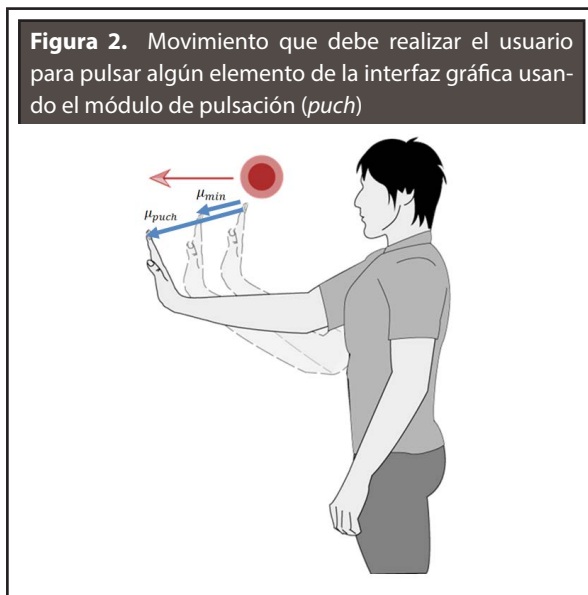
El segundo módulo de interacción propuesto es el módulo de pulsación (*puch*). Este módulo permite que las personas interactúen con los contenidos a una distancia determinada a partir de movimientos sin necesidad de tocar objetos o superficies.

**Figura 1.** Dos tipos de gestos que pueden ser reconocidos por el sistema de interacción propuesto. El gesto de la izquierda es llamado "Desplazar manos de abajo hacia arriba" y el gesto de la derecha es llamado "Deslizar la mano derecha a la izquierda".



El módulo tiene como característica el reconocimiento automático de la persona que va a manipular el sistema e incluso está en capacidad de reconocer la mano derecha o izquierda indicando de forma visual con cuál mano la persona manipulará el sistema. Durante el desarrollo del módulo de pulsación, se determinó que el factor más importante para el funcionamiento correct, es calcular, teniendo al frente del dispositivo Kinect, la distancia de la persona, y en base a ésta distancia determinar la profundidad de las manos al moverlas hacia adelante, para señalar y activar el contenido multimedia. Adicionalmente, el módulo tiene que ser lo suficientemente robusto para soportar un cierto margen de error y evitar que cualquier movimiento natural del usuario pueda activar los diferentes comandos que están configurados.

En la **Figura 2** se puede observar el gesto que debe realizar la persona para pulsar algún elemento en la interfaz gráfica de usuario. Además se pueden observar dos variables definidas como  $\mu_{min}$  y  $\mu_{puch}$  que caracterizan los umbrales de movimiento a partir de los cuales se define la activación del gesto de *puch* en el módulo. Si el movimiento es mayor que  $\mu_{min}$  y menor que  $\mu_{puch}$  significa que el evento de *puch* es activado y dependiendo del control que haga parte de la interfaz, por ejemplo un botón, es activado.



**Figura 2.** Movimiento que debe realizar el usuario para pulsar algún elemento de la interfaz gráfica usando el módulo de pulsación (*puch*)

En el desarrollo del módulo de superficies interactivas, se tuvieron en cuenta varias variables que pudieran afectar el reconocimiento de las manos, como son la condición de iluminación, refracción de superficies y texturas, además de las diferentes superficies en donde se pudiera dar la interacción a cabo. Por tal razón se utilizaron librerías avanzadas para el procesamiento de imágenes y características, ya que los módulos de programación del Kinect ofrecen herramientas muy básicas en este aspecto. Por ende se decidió utilizar OPenCV; esta biblioteca es compatible con los lenguajes de programación C, C++, Phyton y Java; entre sus principales aplicaciones están, el reconocimiento de objetos, calibración de cámaras y aplicaciones para visión robótica. Para integrar el OpenCV con el dispositivo Kinect se utilizaron los controladores OpenNi, estos drivers no solo proporcionan conexión a las cámaras del Kinect, sino también a los micrófonos para el reconocimiento de sonidos.

El sistema de superficies interactivas reconoce cada dedo de las manos de la persona que manipula el sistema, que se traducen finalmente en comandos que se programan para tareas específicas, ya sea en interacción con sistemas operativos, software específico (dibujo, visor de imágenes) o contenidos modelados en 3D, como por ejemplo utilizando el motor de videojuegos llamado Unity (Goldstone, 2011).

Adicionalmente, el sistema de superficies interactivas proporciona información cuando hay contacto de alguno de los dedos sobre la superficie calibrada o reconocida como área de trabajo. Las principales dificultades de reconocer el contacto con las superficies calibradas eran la refracción de la luz, ángulos de captura de las imágenes con el dispositivo Kinect y las superficies brillantes. Estos problemas fueron resueltos aplicando algoritmos de cálculo de distancias óptimas y ortogonalización, que permitieron minimizar los márgenes de error para que el sistema fuera funcional y práctico en las aulas de clase (Arranz, 2013). En este artículo podemos observar el uso de estos algoritmos aplicados

al procesamiento de imágenes tomadas desde diferentes ángulos y alturas, para posteriormente conformar un mapa o documento de la zona analizada, que brindará información de referencia posterior.

Debido a la importancia de esta información fotográfica la calidad de estas imágenes deben ser óptima. En la **Figura 3** se presentan los diferentes movimientos que se pueden realizar para interactuar con los contenidos usando el módulo 3 del sistema de interacción.

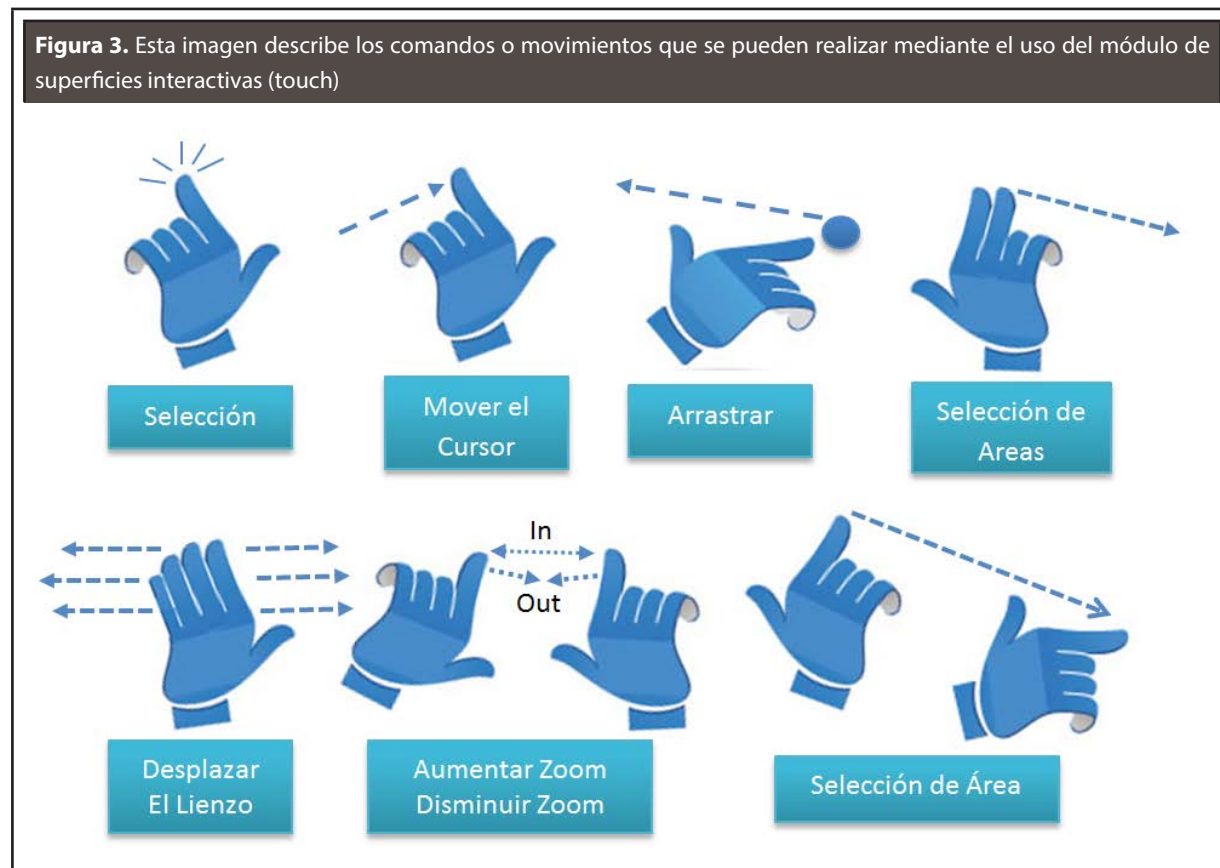
### *Descripción del método utilizado para calcular la profundidad e identificación de superficies*

A continuación se describen dos métodos que son utilizados tanto por el módulo de pulsación como por el módulo de superficies interactivas. Como fue mencionado anteriormente el método uti-

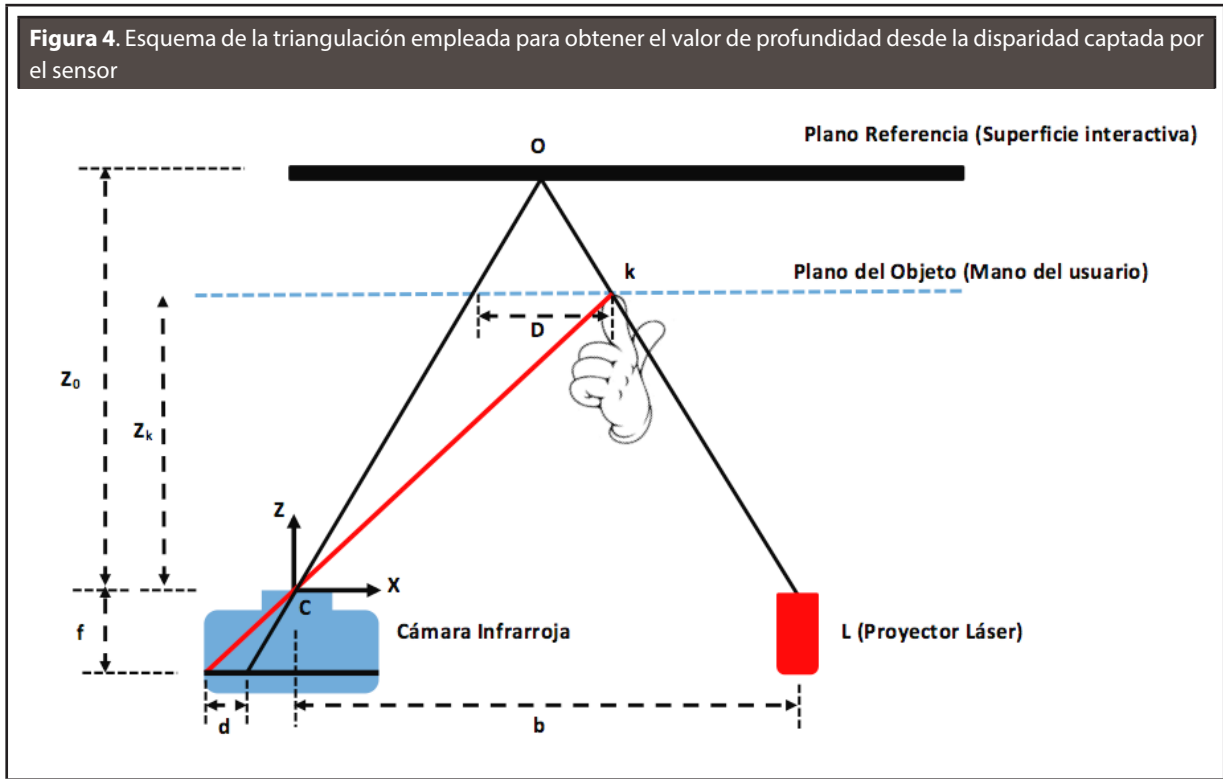
lizado para el reconocimiento de gestos es DTW, el cual fue descrito en (Autor *et al.*, 2015).

### *Cálculo de la profundidad*

Para describir el proceso matemático por medio del cual el Kinect es capaz de obtener la coordenada en el espacio de un objeto específico utilizamos como guía la **Figura 4** y nos basamos en la descripción presentada por Magallón (2013). En ella podemos observar la relación triangular entre el punto del objeto a tratar  $k$  y la disparidad medida  $d$ . Vale la pena anotar que el origen del sistema de coordenadas tridimensional está situado en la cámara infrarroja, con el eje  $Z$  ortogonal al plano de la imagen y dirigido al objeto, y el eje  $X$  en la línea base  $b$  que une la cámara con el proyector perpendicular al anterior.



**Figura 4.** Esquema de la triangulación empleada para obtener el valor de profundidad desde la disparidad captada por el sensor



En la **Figura 4** se puede observar que el plano del objeto está ubicado a una distancia  $Z_k$  menor que la distancia del plano de referencia  $Z_0$ . Por esta razón, cuando se proyecta el patrón desde  $L$ , el punto proyectado que en el plano de referencia visto por  $C$  se situaba en  $o$ , ahora se situará en  $k$ . Desde  $C$  se observa como un desplazamiento en el eje  $X$  a la derecha de magnitud  $D$ . Si el objeto estuviera ubicado más lejos del plano de referencia, el desplazamiento hubiese sido hacia la izquierda. Lo que mide y registra el sensor no es directamente la distancia  $D$  del espacio del objeto, sino la disparidad  $d$ . Para cada punto  $k$  se mide dicho parámetro y se obtiene un mapa de disparidad. A partir de una análisis de los triángulos visualizados en la figura se pueden deducir las siguientes ecuaciones:

$$\frac{D}{b} = \frac{Z_0 - Z_k}{Z_0} \quad (1)$$

$$\frac{d}{f} = \frac{D}{Z_k} \quad (2)$$

Donde  $Z_k$  es la distancia (profundidad) del punto  $k$  del objeto y es la variable que se quiere obtener;  $b$  es la longitud de la base entendida como la distancia entre la cámara y el proyector;  $f$  es la distancia focal del sensor infrarrojo;  $D$  es la distancia real de desplazamiento del punto  $k$  en el espacio del objeto y  $d$  es la disparidad observada en el espacio de la imagen. Sustituyendo  $D$  de **Ecuación 2** en **Ecuación 1** y despejando se obtiene:

$$Z_k = \frac{Z_0}{1 + \frac{Z_0}{f_b} d} \quad (3)$$

Los parámetros  $Z_0$ ,  $f$  y  $b$  pueden ser obtenidos aplicando un proceso de calibración.

La coordenada  $Z$  de cada punto junto con  $f$  define la escala de la imagen para ese punto. Las coordenadas planimétricas del objeto en cada punto se pueden calcular a partir de las coordenadas de la imagen y la escala:

$$X_k = -\frac{Z_k}{f}(x_k - x_0 + \delta x) \quad (4)$$

$$Y_k = -\frac{Z_k}{f}(y_k - y_0 + \delta y)$$

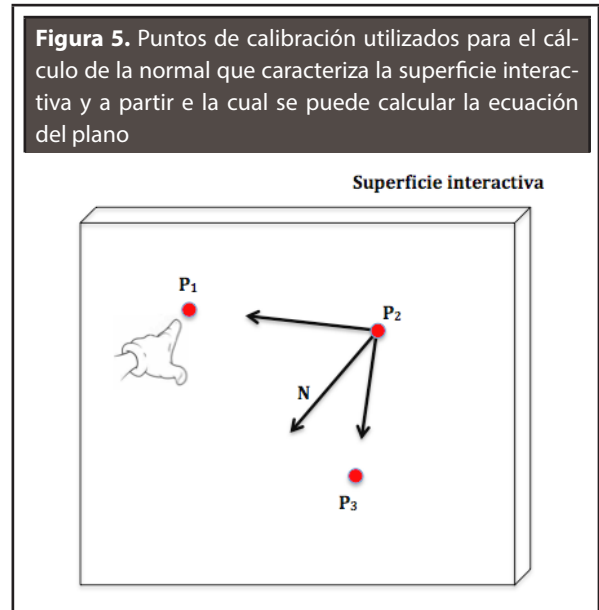
Donde  $x_k$  y  $y_k$  son las coordenadas en la imagen del punto,  $x_0$  y  $y_0$  son las coordenadas del punto principal, es decir, del *offset* de la imagen y  $\delta x$  y  $\delta y$  se refieren a las correcciones realizadas producto de la distorsión del lente. Tanto los valores de *offset* como los de las correcciones también se pueden obtener del proceso de calibración.

Con los parámetros conocidos de la calibración se puede completar la relación entre los puntos medidos de la imagen ( $x, y, d$ ) y las coordenadas del objeto ( $X, Y, Z$ ) de cada punto. Así se puede generar una nube de puntos de cada imagen de disparidad. La disparidad así descrita es una medida de la profundidad inversa, valores más grandes de la misma significan distancias más cortas. Pero el valor de disparidad que retorna la Kinect,  $d'$  está normalizado con un *offset* según la relación:

$$d' = d_{\text{offset}} - 8d \quad (5)$$

### Detección de superficies

El método de detección de superficies es utilizado por el módulo de superficies interactivas. Para detectar la superficie se utiliza un método matemático que calcula la ecuación de un plano a partir de tres puntos que hacen parte de la superficie. Estos tres puntos son capturados por el Kinect aplicando el método descrito previamente, durante una etapa de calibración en la cual el usuario le indica al sistema tres puntos que hacen parte de la superficie que será interactiva. En la **Figura 5** se puede observar la superficie interactiva que se desea detectar y los tres puntos  $P_1, P_2, P_3$ , seleccionados por el usuario para calcular el plano de la superficie interactiva.



A partir de estos puntos se calcula la normal  $N$  al plano, mediante la siguiente ecuación.

$$N = \frac{P_3 - P_2}{\|P_3 - P_2\|} \times \frac{P_1 - P_2}{\|P_1 - P_2\|} \quad (6)$$

$$N = (nx, ny, nz) \quad (7)$$

Considerando que la ecuación de un plano puede ser expresada de la siguiente manera:

$$Ax + By + Cz + D = 0 \quad (8)$$

Podemos obtener la ecuación del plano que caracteriza la superficie interactiva, a partir de la normal calculada y un punto que haga parte del plano, usando las siguientes expresiones:

$$A = nx \quad (9)$$

$$B = ny \quad (10)$$

$$C = nz \quad (11)$$

$$D = -Ax_1 - By_1 - Cz_1 \quad (12)$$

Donde  $P_1 = (x_1, y_1, z_1)$  y  $nx, ny$  y  $nz$  son los componentes de la normal calculada. Una vez calculado el plano que representa la superficie interactiva a



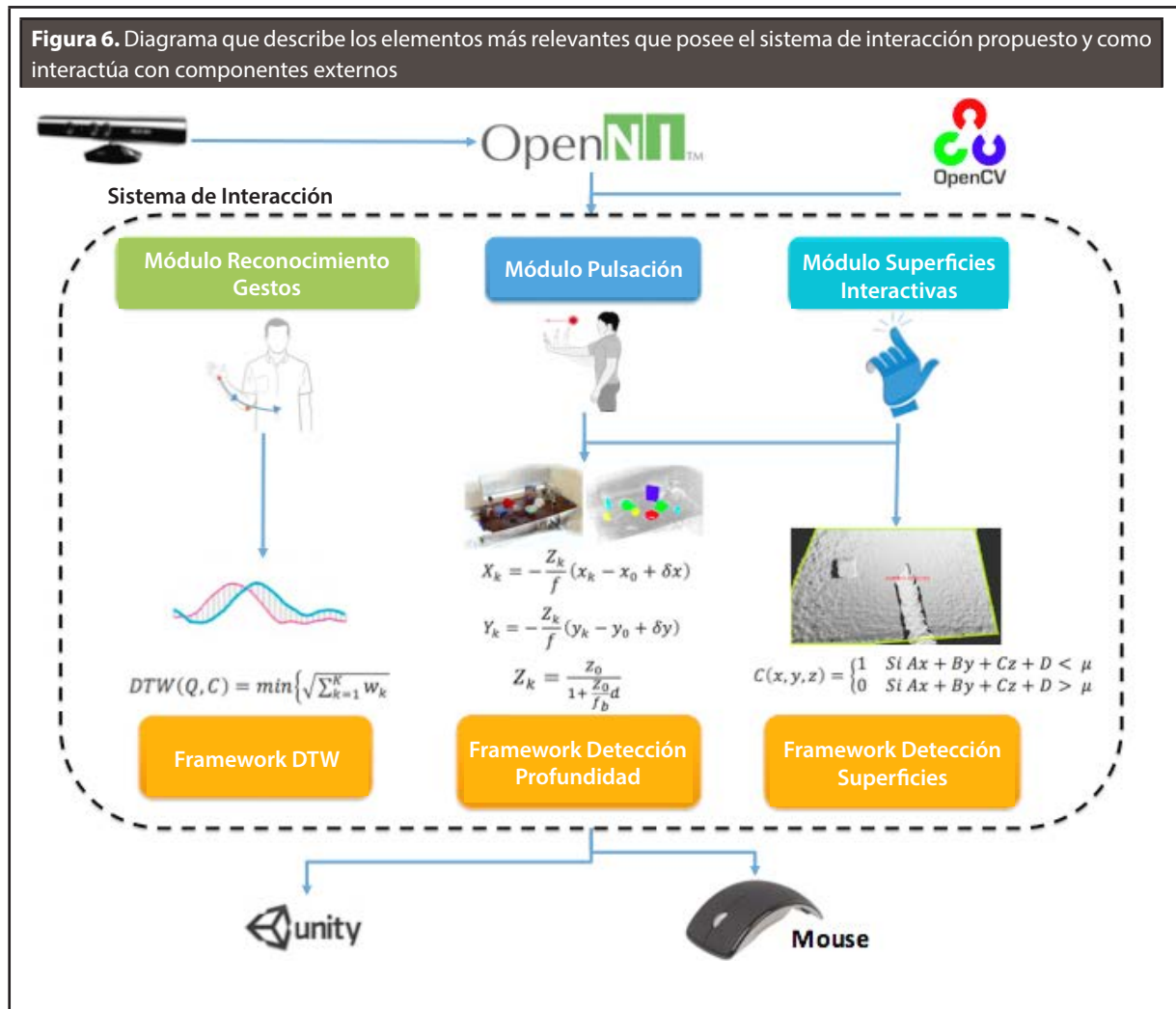
partir de los puntos de calibración, para saber si el usuario esta haciendo contacto con sus dedos, se reemplaza la coordenada actual de los dedos en la ecuación del plano, y si esta es igual 0, significa que el usuario esta haciendo contacto con la superficie. Debido a que el Kinect puede cometer errores de precisión en el cálculo de la profundidad, se establece un umbral  $\mu$  que corresponde a la distancia, entre la coordenada capturada por el Kinect y el plano de la superficie interactiva, en la cual se considera que hay contacto. Esto es representado mediante la siguiente expresión:

$$C(x, y, z) = \begin{cases} 1 & \text{Si } Ax + By + Cz + D < \mu \\ 0 & \text{Si } Ax + By + Cz + D > \mu \end{cases} \quad (13)$$

Donde  $C(x, y, z) = 1$  significa que el usuario está contactando la superficie interactiva y  $C(x, y, z) = 0$  significa lo contrario,  $x, y$  y  $z$  son las coordenadas del punto que define los dedos del usuario.

### Descripción de la arquitectura del sistema de interacción

Este sistema posee una arquitectura que permite la optimización de todos los recursos del dispositivo Kinect y el dispositivo de cómputo, con el objetivo de brindar una buena fluidez del sistema al momento de interactuar con los demás componentes con los cuales debe integrarse, como por ejemplo, sistemas operativos, software de visualización, entre otros.



Las librerías Open CV para los módulos de interacción nos permiten optimizar rutinas de código y mejorar la precisión y el éxito con el cual se mide la intención del usuario independiente del paradigma aplicado. La **Figura 6** muestra de manera esquemática en un diagrama de bloques cada uno de los componentes que hacen parte del sistema de interacción.

El sistema de interacción está compuesto por tres módulos: reconocimiento de gestos, pulsación y superficies interactivas. Junto a cada módulo se puede observar en la **Figura 6** que hay una imagen con el tipo de movimiento que tiene que realizar el usuario para interactuar con el contenido usando cada paradigma. Cada uno de estos módulos utiliza una o varias funcionalidades para medir la intención del usuario y mapearla a un comando; estas funcionalidades básicas son *Framework DTW*, *framework detección de profundidad* y *framework detección de superficies*. El primer framework fue descrito en (Arranz, 2013) y los dos últimos frameworks fueron descritos en la sección anterior. Adicionalmente el sistema de interacción usa OpenNI para poder extraer los datos provenientes del Kinect y OpenCV como librería de procesamiento de imágenes, para facilitar el procesamiento realizado por los frameworks descritos. Con los frameworks se visualizan las ecuaciones claves que permiten reconocer los gestos, determinar la profundidad de la mano y detectar las superficies interactivas. Finalmente, el sistema interactúa con Unity3D para controlar los contenidos desplegados como videojuegos, ambientes virtuales, visualización y realidad aumentada, o con el componente del sistema operativo que controla el *mouse* para controlar aplicaciones genéricas como Paint o PowerPoint.

### 3. DISCUSIÓN Y RESULTADOS

Para las pruebas experimentales descritas en esta sección se utilizó un dispositivo Kinect con conexión a un computador. Dicho computador tenía las siguientes características: 4Gb de RAM y procesador *dual core* o superior. Para esta prueba, el Kinect está ubicado a 2 metros de distancia del

usuario, con una iluminación media (ambiente). El sistema de interacción con el módulo de pulsación y el módulo de superficies interactivas fue instalado en el computador con el fin de realizar la prueba experimental.

En la **Tabla 1** se describen de manera general las pruebas que pueden ser realizadas con el módulo push para establecer falencias y fortalezas de este tipo interacción con los contenidos educativos ya planteados.

**TABLA 1. PARÁMETROS A UTILIZAR EN LA PRUEBA EXPERIMENTAL QUE SE DESARROLLARÁ CON EL MÓDULO PUSH**

| Prueba experimental para el módulo de interacción pulsar ( <i>push</i> )   |  |
|--|--|
| En este Framework el usuario dirige el cursor en la aplicación, y selecciona el contenido simulando el pulsar un botón |  |
| 1) Se podría medir la precisión al controlar el cursor.  | Cómo: tomando tiempos para la localización de los vínculos con forma de botones del framework y ejecutarlos.                                     |
| 2) reconocimiento de la mano derecha e izquierda.  | Cómo: realizando repetidos cambios de manos y de usuarios. Y así evidenciar bloqueos o inconsistencias.  |
| 3) medir la distancia mínima y máxima de eficacia del sistema.   | Cómo: tomando diferentes muestreos de distancia desde el Kinect al operador del sistema, determinando los rangos mínimos y máximos de operación. |

Para esta prueba experimental se contó con 5 usuarios, cuyo rango de edades es de los 23 a los 35 años y una estatura promedio de 1,72. En la **Tabla 2** se pueden observar las características de cada uno de los usuarios evaluados.

**TABLA 2. CARACTERÍSTICAS DE LAS PERSONAS QUE REALIZARAN LA PRUEBA EN EL MÓDULO PUSH**

| Usuario | Sexo      | Edad    | Altura |
|---------|-----------|---------|--------|
| 1       | Femenino  | 23 años | 1,68   |
| 2       | Femenino  | 16 años | 1,60   |
| 3       | Masculino | 28 años | 1,80   |
| 4       | Masculino | 35 años | 1,75   |
| 5       | Masculino | 24 años | 1,81   |

Estas personas estaban ubicadas a 1 metro de distancia del Kinect, inicialmente. Tomando como base el tiempo que se demora una persona entrenada, se midió el tiempo promedio que tarda una persona sin entrenamiento para accionar cada uno de los botones de la aplicación, que en este caso totalizaron 3 (física, química y astronomía). Para esta prueba, el *framework* deberá reconocer la mano que guiará el cursor, sin importar la textura física, ni la estatura de la persona. Cada usuario realizó 3 repeticiones de la prueba. Los resultados obtenidos para esta prueba son presentados en la **Tabla 3**. Considerando que en promedio un experto tarda 24 segundos en realizar la tarea, se puede observar que en promedio un grupo de usuarios sin entrenamiento alcanza a ejecutar la tarea en un tiempo.

**TABLA 3. RESULTADOS OBTENIDOS EN CADA REPETICIÓN DE LA PRUEBA DE CADA USUARIO PARA PROBAR EL MÓDULO DE PULSACIÓN**

| Usuario | Repetición 1 (segundos) | Repetición 2 (segundos) | Repetición 3 (segundos) | Promedio (segundos) |
|---------|-------------------------|-------------------------|-------------------------|---------------------|
| 1       | 24                      | 22                      | 28                      | 25                  |
| 2       | 25                      | 27                      | 24                      | 25                  |
| 3       | 29                      | 23                      | 18                      | 23                  |
| 4       | 31                      | 27                      | 23                      | 27                  |
| 5       | 28                      | 21                      | 22                      | 24                  |

Adicionalmente, debido a que el aplicativo deberá de tener la posibilidad de actualizar rápidamente la configuración de la persona que lo usa,

e identificar con qué mano se está manipulando el puntero, se realizó una prueba experimental para evaluar esta característica del módulo de interacción. Para esta prueba los 5 participantes se ubicaron a 1 metro de distancia del Kinect y cambiaron de posición el uno al otro, de manera que solo un participante sea capturado por el Kinect al tiempo. Esto lo realizarán los participantes 5 veces cada 50 segundos, garantizando que la aplicación es capaz de auto calibrarse y adaptarse a la persona que manipule el sistema, sin bloquearse o hacer un reconocimiento erróneo.

Los participantes repetirán la primera prueba a 1,5 metros, 2 metros, 3,5 metros y 4 metros.

Con los resultados obtenidos de la anterior prueba podemos plantear que el sistema es estable en el proceso de auto calibración e identificación del usuario que interactúa con el sistema.

En la **Tabla 5** se describe el experimento propuesto para evaluar el módulo de superficies interactivas. Para la prueba experimental el Kinect se ubicó a 1,5 metros de la superficie que deseamos convertir en superficie táctil, se pueden utilizar superficies irregulares, lisas, sin brillo y pintadas de cualquier color. Posteriormente, luego de ubicar el Kinect, se iniciará el reconocimiento de superficies; esto se hace con el propósito de que el módulo sea capaz de diferenciar las manos, distancias de la superficie y condiciones de iluminación que puedan alterar la lectura del Kinect.

**TABLA 4. RESULTADOS DE LA PRUEBA QUE MIDE LA ADAPTACIÓN DEL SISTEMA PROPUESTO. LA ETIQUETA IDENTIFICADO HACE ALUSIÓN A LA CALIBRACIÓN EXITOSA DE LA IDENTIFICACIÓN DEL USUARIO, LA ETIQUETA PROCESANDO HACE ALUSIÓN A FALLAS EN EL TIEMPO DE IDENTIFICACIÓN DEL USUARIO QUE NO DEBE DE SUPERAR 50 SEGUNDOS**

| Tiempo | Distancia  | Repetición1  | Repetición2  | Repetición3  | Repetición4  | Repetición5  |
|--------|------------|--------------|--------------|--------------|--------------|--------------|
| 50 s   | 1.5 metros | identificado | identificado | identificado | identificado | identificado |
| 50 s   | 2 metros   | identificado | identificado | identificado | identificado | identificado |
| 50 s   | 3.5 metros | identificado | procesando   | identificado | identificado | identificado |
| 50 s   | 4 metros   | identificado | identificado | identificado | identificado | identificado |

Cada vez que se vaya a hacer un reconocimiento con el módulo de superficies interactivas, bastará con refrescar el área de trabajo y verificar que cuando cada dedo toque la superficie cambie de color; esto nos garantizará que el Kinect está identificando de forma correcta la profundidad de la mano y el contacto con la superficie. En la **Tabla 5** se observan los elementos que pueden ser evaluados para la prueba experimental que se propone en este caso.

**TABLA 5. PARÁMETROS A UTILIZAR EN CADA PRUEBA EXPERIMENTAL QUE SE DESARROLLARÁ CON EL MÓDULO DE SUPERFICIES TÁCTILES**

Prueba propuesta para el *framework* superficies interactivas

En este Framework el usuario dirige el cursor en la aplicación, y selecciona el contenido realizando toques directos en superficies y ejecutando un comando con los dedos.

|   |   |
|---|---|
| 1) Se podría medir la precisión al controlar el cursor        | Cómo: ubicando el Kinect a diferentes distancias y realizando auto calibración para estas, para así verificar que reconozca las manos adecuadamente.                        |
| 2) Reconocimiento de superficies                              | Cómo: ubicando el Kinect en superficies proyectadas, y con texturas variantes. Y luego verificar que el reconocimiento de la mano y el toque de la superficie sea correctos |
| 3) Medir la distancia mínima y máxima de eficacia del sistema | Cómo: tomando diferentes muestreos y ángulos de operación   |

Con el fin de evaluar algunos de los elementos planteados en la **Tabla 5** se realizó una prueba experimental, en la cual el kinect fue ubicado a 1,5 mts de la superficie interactiva. En este caso, la superficie interactiva es una mesa sobre la cual se ubicó el dibujo del contorno de las tres figuras geométricas: un cuadrado, un triángulo y un círculo. En esta configuración el usuario debía dibujar la línea de contorno con el dedo, siguiendo el patrón ubicado sobre la superficie interactiva. La configuración descrita puede ser observada en la **Figura 7**.

**Figura 7.** Configuración experimental utilizada para evaluar la herramienta de superficies interactivas



Como herramienta de dibujo se utilizó la herramienta Paint de Windows. Una vez el usuario finalizaba de dibujar el contorno de las tres figuras, se le tomaba el tiempo y respondía una encuesta sobre su percepción utilizando la herramienta. Adicionalmente, se almacenó la imagen de referencia, es decir, el patrón dibujado sobre las superficies interactivas, y de igual manera se almacenó los contornos dibujados por el usuario, con el fin de comparar las diferencias en la trayectoria dibujada por cada uno de ellos y las figuras geométricas de referencia. En total fueron evaluados diez usuarios, con un rango de edades de entre los 19 y los 34 años, cuatro de ellos fueron de sexo femenino y seis de ellos de sexo masculino. Todos los participantes habían tenido una experiencia previa usando la herramienta Paint, y dos de los usuarios no tenían experiencia previa usando el dispositivo Kinect.

En la **Figura 8** de la **(a)** a la **(d)** podemos observar los resultados de las preguntas realizadas a cada uno de los usuarios. Se usó un esquema de preguntas tipo escala Likert para evaluar cada una de las afirmaciones. A partir de los resultados podemos observar que para el 90 % de los usuarios es fácil determinar cómo interactuar con la aplicación (**Figura 8a**); sin embargo, en la **Figura 8d** el 60 % no toma ninguna posición, considerando si es fácil saber cómo dibujar los contornos en la aplicación.

Esto se debe a que en la aplicación para dibujar el contorno se debe usar el dedo índice en contacto con la superficie y los otros cuatro dedos cerrados. Algunos usuarios durante las pruebas realizaron incorrectamente el gesto para dibujar, por ejemplo, algunos dejaban todos los dedos extendidos lo que podía dificultar el uso de la aplicación.

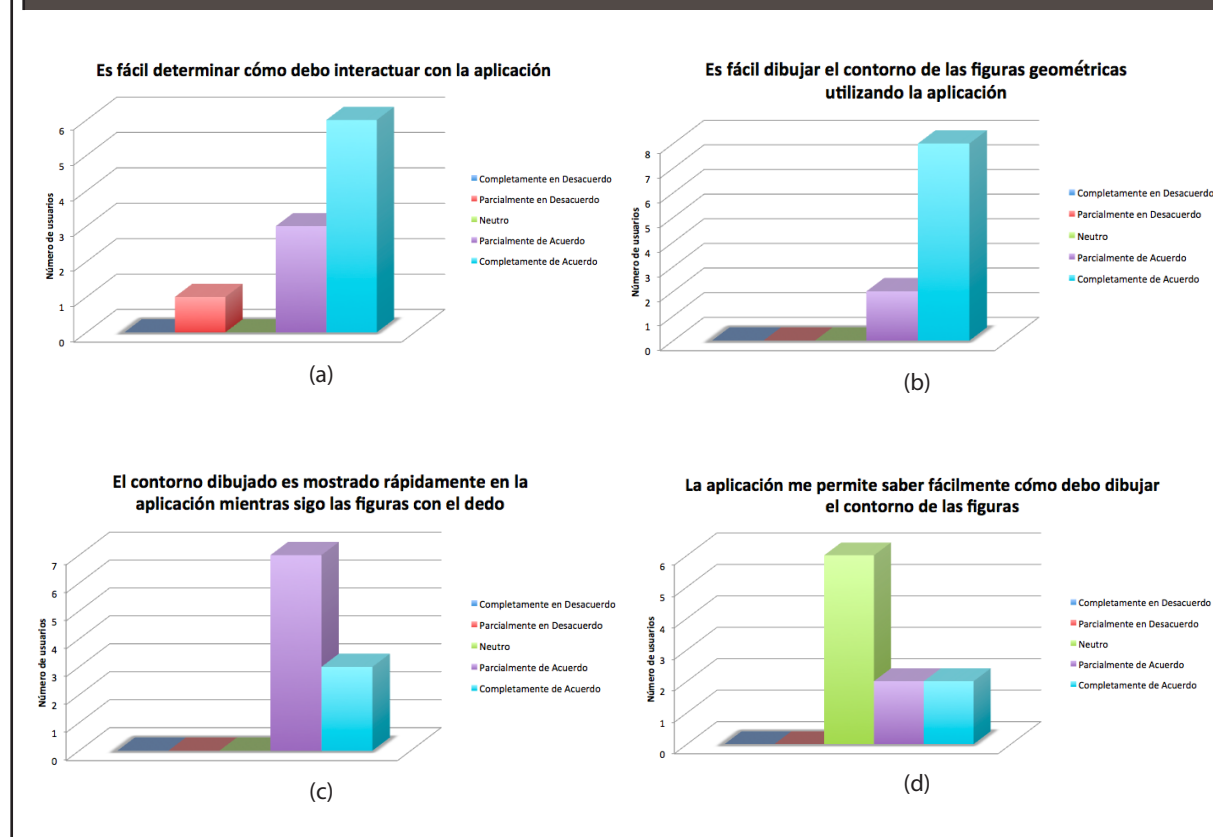
Sin embargo, cuando los usuarios eran conscientes de cómo debían realizar el gesto de la mano, dibujar los contornos fue muy sencillo para ellos como lo demuestra la **Figura 8b**. Finalmente, los usuarios estuvieron de acuerdo en que mientras ellos dibujaban el contorno de la figuras geométricas la aplicación rápidamente graficaba dicho contorno en la aplicación Paint (Ver **Figura 8c**).

Por último, durante la tarea de dibujado de los contornos se tomó el tiempo de ejecución de











la tarea a los usuarios y el número de errores. En promedio, los usuarios cometen menos de un error durante la delineación del contorno y se toman un tiempo de 10 segundos delineando del contorno de las tres figuras geométricas. Se considera que el tiempo de ejecución de la tarea de los usuarios es bueno, teniendo en cuenta que un usuario experto hace la tarea en 8 segundos. Por otra parte, la mayoría de errores que cometieron los usuarios fueron producto de un incorrecto gesto para delinear el contorno de la figuras geométricas.

En la **Tabla 6** se pueden observar las figuras dibujadas por los usuarios, el número de errores que cada uno cometió y el tiempo que le tomó completar la tarea.

**Figura 8.** Resultados a la preguntas realizadas a los usuario sobre la percepción que tuvieron respecto al uso de la aplicación



**TABLA 6.** RESULTADOS OBTENIDOS CON LOS USUARIOS CUANDO DIBUJABAN LAS FIGURAS GEOMÉTRICAS DESCRITAS

| Usuario | Resultado Dibujo Contorno   | Tiempo (segundos) | Errores |
|---------|---|-------------------|---------|
| 1       |    | 15                | 1       |
| 2       |    | 25                | 0       |
| 3       |    | 24                | 0       |
| 4       |    | 11                | 2       |
| 5       |   | 9                 | 0       |
| 6       |  | 16                | 1       |
| 7       |  | 17                | 2       |
| 8       |  | 9                 | 2       |
| 9       |  | 11                | 0       |
| 10      |  | 9                 | 0       |

#### 4. CONCLUSIONES

Con los datos arrojados por las diferentes pruebas realizadas a los módulos de superficies interactivas y sistema de pulsación multimedia podemos definir que la arquitectura propuesta, incluyendo hardware y software, permite una adecuada interacción del usuario con los contenidos desplegados por las aulas de clase inteligentes, utilizando tres tipos diferentes de paradigmas de interacción. Esto permite que se pueda seleccionar el paradigma de interacción que más se ajuste al contenido desarrollado.

Para el caso de los paradigmas de interacción del módulo de reconocimiento de gestos y el módulo de pulsación, el modelo matemático y los algoritmos propuestos que hacen parte del *framework* son lo suficientemente robustos para la interacción que se necesita. Sin embargo, el módulo de superficies interactivas requiere de un mejor algoritmo para reconocer cuando un dedo de la mano entra en contacto con la superficie interactiva, ya que si se usa un gesto que no deje ver claramente la punta de los dedos el sistema comete errores al determinar si se está interactuando o no con la superficie.

Adicionalmente, durante el desarrollo de las pruebas experimentales, se le preguntó a los usuarios si consideraban que la interacción con el sistema utilizando cualquiera de los tres paradigmas era fácil, a lo que ellos respondieron que era muy fácil aprender e interactuar con el sistema propuesto. Finalmente, a partir de las pruebas experimentales para evaluar la interacción con las superficies interactivas, se plantea desarrollar un sistema más robusto que permita reconocer los gestos de la mano, a pesar de que hayan ligeras diferencias realizadas por los usuarios. Considerando todo el sistema de interacción propuesto, como trabajo futuro se plantea evaluar en qué casos dentro del aula de clase es más conveniente usar un tipo de interacción u otro.

## AGRADECIMIENTOS

Un especial agradecimiento al Centro de Investigación de la Institución Universitaria Salazar y Herrera por su constante apoyo al trabajo del presente proyecto de investigación.

## REFERENCIAS

- Al Delail, B.; Weruaga, L.; Zemerly, J. (2012). CAViAR: Context Aware Visual indoor Augmented Reality for a University Campus. *IEEE/WIC/ACM International Conferences*, pp 2-4.
- Arranz, J. (2013). Diseño, optimización y análisis de sistemas basados en técnicas láser, para el modelado geométrico, registro y documentación, aplicados a entidades de interés patrimonial, Tesis doctoral, pp. 217-2225, pp. 334.
- Autor. (2015). Descripción de un Sistema de Reconocimiento de Gestos Para su Implementación en una Aula de Clase Inteligente. Pendiente de Publicación.
- Bailly, G.; Müller, J.; Lecolinet, E. (2012). Design and evaluation of finger-count interaction: Combining multi-touch gestures and menus. *International Journal of Human Computer Studies*, 70(10), pp. 673 – 689.
- Chen, C-C.; Huang, T-C. (2012). Learning in a u-Museum: Developing a context-aware ubiquitous learning environment. *Computers & Education*, 59(3), pp. 873 – 883.
- Goldstone, W. (2011). *Unity 3.x Game Development Essentials*. Packt Publishing, Second Edition.
- Krejov, P.; Gilbert, A.; Bowden, R. (2014). A Multitouchless Interface Expanding User Interaction, *IEEE Computer Society 0272-1716*, pp. 2-8.
- Magallón, M. (2013) *Sistema Interactivo para Manejo de electrodomésticos en Entornos Domésticos*. Trabajo de grado en Ingeniería de Telecomunicaciones, Universidad de Zaragoza.
- Nacher, V.; Jaen, J.; Navarro, E.; Catala, A.; González, P. (2014). Multi-touch gestures for pre-kindergarten children. *International Journal of Human-Computer Studies*, 73, pp.7-12.
- Novotný, M.; Lacko, J.; Samuelčík, M. (2013). Applications of Multi-Touch Augmented Reality System in Education and Presentation of Virtual Heritage, *Procedia Computer Science*, 25, pp. 231-235.
- Papatheodorou, C.; Antoniou, G.; Bikakis, A. (2010). On the Deployment of Contextual Reasoning in Ambient Intelligence Environments. *Sixth International Conference on Intelligent Environments (EI)*, pp. 13 – 18.
- Seo, W.; Yeol Lee, J. (2013). Direct hand touchable interactions in augmented reality environments for natural and intuitive user experiences. *Expert Systems with Applications: An International Journal archive*. 40(9), pp. 3784-3793.
- Zhao, J.; Soukoreff, R.; Ren, X.; Balakrishna, R. (2014). A model of scrolling on touch-sensitive displays. *International Journal of Human Computer Studies*, 72(12), pp. 805 - 821.

**PARA CITAR ESTE ARTÍCULO /  
TO REFERENCE THIS ARTICLE /  
PARA CITAR ESTE ARTIGO /**

Díaz León, C.A.; Hincapié Montoya, E.M.; Guirales Arredondo, E.A.; Moreno López, G.A. (2016). Diseño y desarrollo de un sistema de interacción para su implementación en un aula de clase inteligente. *Revista EIA*, 13(26), julio-diciembre, pp. 95-109. [Online]. Disponible en: <https://doi.org/10.24050/reia.v13i26.666>