



*Revista Tecnológica ESPOL, Vol. xx, N. xx, pp-pp, (Mes, 200x)*

## **Diseño para la fácil implementación de Superficies Sensitivas por medio de la Reutilización de Componentes.**

José Oramas M.<sup>(1)</sup>

Xavier Ochoa Ch.<sup>(2)</sup>

Centro de Tecnologías de Información

Escuela Superior Politécnica del Litoral

Edificio 37, Km 30.5 Vía Perimetral, Guayaquil, Ecuador

[joramas@cti.espol.edu.ec](mailto:joramas@cti.espol.edu.ec)<sup>(1)</sup>

[xavier@cti.espol.edu.ec](mailto:xavier@cti.espol.edu.ec)<sup>(2)</sup>

### **Resumen**

*En este artículo se presenta un diseño para la implementación de superficies sensitivas de bajo costo. Considerando como costo: (a) el valor del hardware y (b) los conocimientos necesarios para el desarrollo del sistema. El diseño propuesto apunta a reducir estos costos por medio de la reusabilidad de componentes. En el lado de hardware, ensamblando una estructura abierta compatible con diferentes cámaras y medios de proyección para ser empleados como superficie. En el lado de software, se plantea el uso de HandVu como la base para la detección de la mano del usuario y el reconocimiento de posturas y gestos dactilológicos. Adicionalmente se describe la evaluación de VirtualTable, un prototipo desarrollado siguiendo las guías propuestas, para presentar las características y funcionalidades que los sistemas desarrollados con el diseño propuesto pueden proveer.*

**Palabras Claves:** interfaces naturales de usuario, superficies sensitivas, bits tangibles, gestos, HandVu, reusabilidad.

### **Abstract**

*In this paper a design for the development of low-cost surface computing systems is presented. Considering as cost: (a) the hardware and (b) the knowledge needed for the development of the system. The design aims to reduce the costs focusing on the reusability of components. On the hardware side, setting up a compatible infrastructure with different cameras and displays to be used as the surface. On the software side using HandVu as the base for the detection and tracking of user's hands and the recognition of hand's postures and gestures. The testing of VirtualTable, a prototype developed following the proposed guidelines is described in order to present the features and functionalities that the systems developed with the design can provide.*

## 1. Introducción

Interacción Hombre-Computador es la rama de las Ciencias en Computación cuyo objetivo es mejorar la productividad de las personas al trabajar con sistemas computacionales. Para lograr esto, durante los últimos años se han planteado diferentes formas de interactuar con los computadores. Uno de estas formas o paradigmas de interactuar con el computador que se ha hecho popular recientemente es la Interacción Háptica. Esta consiste en emplear el sentido del tacto para trabajar directamente con la información sin necesidad de usar dispositivos como el teclado y el ratón.

En los últimos meses varias alternativas comerciales han surgido siguiendo este paradigma. Entre estas Surface de Microsoft, DiamondTouch de Mitsubishi, los muros sensitivos al tacto de Perceptive Pixel, entre otros sistemas que cambian significativamente la experiencia de uso sin embargo poseen un costo relativamente alto.

El objetivo de este documento plantear el desarrollo de un sistema que siga este paradigma y que disminuya los costos de implementación por medio de la reutilización de dispositivos de fácil acceso y/o bajo costo.

## 2. Paradigma de Interacción Háptica

El estilo de interacción háptica fue propuesto por Iroshi Ishii en 1997 bajo el nombre Bits Tangibles []. Bajo este concepto Ishii promovía el uso de interfaces que exploten la riqueza de los sentidos humanos y no solo la vista como los “bits pintados” que normalmente se puede ver en los monitores de los computadores que se usa comúnmente. Para la propuesta de Bits Tangibles, Ishii presentó MetaDESK, TransBOARD y AmbientRoom, sistemas en los cuales el usuario podía manipular las aplicaciones por medio de sus manos directamente o por medio de herramientas muy similares a las empleadas para realizar dicha actividad en el mundo real. De esta manera se explotaba el conocimiento adquirido previamente en lo referente al uso de estas herramientas o a la forma en que se debía interactuar con dicho sistema.

Entre los diferentes estilos de interacción que cubre este paradigma, se encuentra el de superficies sensitivas. Este estilo consiste en proyectar sobre una mesa o un muro la imagen de una aplicación y por medio de sus manos el usuario puede interactuar con los datos presentados sobre esta.

## 3. Antecedentes

Como fue presentado en [10, 13, 1, 5, 6, 9, 12, 7, 4] diferentes investigaciones han sido realizadas persiguiendo el uso de las manos como medio de interacción sobre muros y superficies empleando algoritmos de visión por computador. La ventaja de esta estrategia es el bajo costo que el hardware empleado en estos sistemas. Sin embargo, algunas de estas propuestas toman como base algoritmos muy complejos de visión por computador, introduciendo de esta manera carga relacionada con los conocimientos necesarios para la implementación. Para evitar el requerimiento que esta carga cognitiva representa, este documento se enfocará en herramientas que faciliten el desarrollo de superficies sensitivas cuya interacción se realice por medio de las manos.

Este documento presenta un diseño para la fácil implementación de superficies sensitivas de bajo costo, que tomen como base de interacción posturas y gestos dactilológicos. Este diseño disminuye los costos de implementación ya que sigue un principio de reutilización de recursos de software y recursos de hardware existentes. El uso de este tipo de interfaces con aplicaciones de software particulares mejoran el tiempo requerido para el aprendizaje de uso de dicho software y/o disminuyen el tiempo empleado para realizar una tarea con él. Para investigadores, profesores y estudiantes de HCI el diseño presentado en este documento propone ser una guía para la fácil implementación de superficies sensitivas para ser usadas como plataformas de prueba de bajo costo para este paradigma.

El objetivo de este documento es la reducción de costos tanto económicos como intelectuales en la implementación de superficies sensitivas con las siguientes características:

- Interfaz sin elementos agregados: El sistema propuesto apunta a proveer una interacción por medio de las manos sin la necesidad de agregar a estas elementos adicionales. Como fue presentado en [8] “las aplicaciones en las cuales el uso de un dispositivo externo es necesario para la interacción son menos naturales y ese dispositivo representa una limitación no solo para el usuario sino también para la aplicación”. Adicionalmente, los dispositivos mecánicos son afectados por interferencias externas y reducen los grados de libertad que posee el usuario generando “ruido ergonómico” [11].
- Reutilización de conocimientos/habilidades: El sistema propuesto va a recibir entradas enviadas con la mano por medio de posturas y/o gestos dactilológicos similares a los que el usuario realizaría para realizar la misma tarea en el mundo real. Como fue

mencionado en [3], “las interfaces en las que se emplean directamente las manos, explotan de mejor forma la riqueza de los sentidos humanos comparadas con las tradicionales interfaces teclado-mouse”.

- Integración de hardware existente: Por medio de esta característica el diseño propuesto apunta a permitir a los desarrolladores a emplear hardware que tengan disponible en sus laboratorios como cámaras DV o webcams, para el reconocimiento de las manos del usuario, el uso de pantallas LCDs o CRTs, proyectores, etc. para el desarrollo de sistemas.

### 3. Trabajo Relacionado

De los proyectos mencionados en la sección anterior, HandVu [4] es el único que se enfoca en facilitar el desarrollo de superficies sensitivas en base a lenguajes dactilológicos. HandVu consiste en un librería para desarrollar aplicaciones de Computación Vestible. Emplea un algoritmo que combina características KLM con comportamiento de rebaño para detectar y seguir objetos articulados, en este caso, la mano del usuario, proveiendo la funcionalidad de reconocer posturas dactilológicas. Provee de archivos fuente para asegurar su portabilidad e incluye un aplicación de prueba que puede ser instalada en diferentes sistemas operativos.

A finales del 2007, el boom de los hacks de Wiimote iniciaron con la propuesta de Johnny Lee. Esta propuesta planteaba el uso del control remoto del Nintendo Wii (wiimote) como una cámara infrarroja de altorendimiento capaz de detectar hasta cuatro puntos de mayor intensidad en una resolución de 1024x768 píxeles conectándose a un computador por un enlace bluetooth. Para el uso de las manos como medio de interacción, Lee propuso el uso de cinta reflectiva en la punta de los dedos para de esta manera mejorar la precisión en la detección. El código fuente para realizar la conexión Wiimote-computador para el sistema operativo MS Windows y otras aplicaciones de prueba fueron liberadas desde el día de su lanzamiento, actualmente varias migraciones a sistemas GNU Linux se encuentran disponibles.

Touchlib es un framework para el desarrollo de superficies multipunto en base a Reflexión Total Interna Frustrada (FTIR) [2] para detectar y seguir manchas de luz infrarroja en pantallas de retroproyección. Provee capacidad de procesamiento distribuido al enviar desde un servidor eventos como “dedo presionando”, “dedo levantado”, “dedo movido”, etc. FTIR es la tecnología empleada detrás

de los productos de la compañía Perceptive Pixel. Originalmente el framework se encontraba disponible para MS Windows, sin embargo, actualmente se encuentran disponibles versiones para otras plataformas.

Bespoke 3DUI XNA es un framework a facilitar la integración de superficies en base a Wiimotes o tecnología FTIR a entornos MS Windows. Se encuentra desarrollado sobre el XNA Game Studio de Microsoft y se encuentra liberado bajo la Licencia Pública de Microsoft (MPL). Está escrito en el lenguaje C# y requiere de las clases del framework .NET. Adicionalmente provee renderizado estereoscópico, seguimiento de cabeza con seis grados de libertad, reconocimiento de gestos en tres dimensiones, etc.

## 4 Descripción del Sistema

### 4.1 Componentes del Sistema

El funcionamiento de sistemas desarrollados siguiendo el diseño que este documento propone consiste en varias etapas. En la etapa inicial cuando el usuario interactúa sobre la superficie, una cámara apuntando perpendicularmente hace la superficie captura imágenes constantemente. Esta cámara envía las imágenes a un computador que posee algunos módulos de software encargados de realizar diferentes tareas.

Durante la segunda etapa, uno de estos módulos escaneará las imágenes recibidas para detectar la mano del usuario. Una vez detectada va a ser seguida con el objetivo de reconocer gestos y posturas dactilológicas.

Dependiendo de la lógica de la aplicación alojada en el sistema, el reconocimiento de posturas y gestos generará eventos. Finalmente cuando un evento es lanzado, su respectiva retroalimentación es presentada en la superficie.

Dando una mirada a los elementos mencionados previamente, el sistema propuesto se encuentra formado por los siguientes cuatro componentes:

- (C1) Área de trabajo; un área en la cual la interfaz gráfica de usuario (GUI) es mostrada. hay varias tecnologías empleadas para la implementación del área de trabajo en otros proyectos: pantallas de retroproyección, superficies sensibles al tacto, etc. . El sistema propuesto sigue un diseño que permita el uso de dispositivos de proyección tradicionales como proyección frontal, pantallas LCDs o CRTs de dimensiones significantes.

- (C2) Entrada Visual; se encarga de realizar la detección y seguimientos de las manos del usuario sobre la superficie. En el mercado existen cámaras de alto rendimiento que permiten sensor luz infrarroja; sin embargo, el sistema propuesto persigue el uso de dispositivos de captura mas comunes.
- (C3) Computador; en el cual el procesamiento va a ser realizado.
- (C4) Software; corre en el computador y provee funcionalidad específica a todo el sistema. Puede ser dividido en dos parte: (a) la parte de detección y seguimiento; encargada del reconocimiento de posturas y gestos dactilológicos y (b) la parte de aplicación; que se encarga de manejar la lógica de la aplicación y envía la retroalimentación respectiva al usuario.

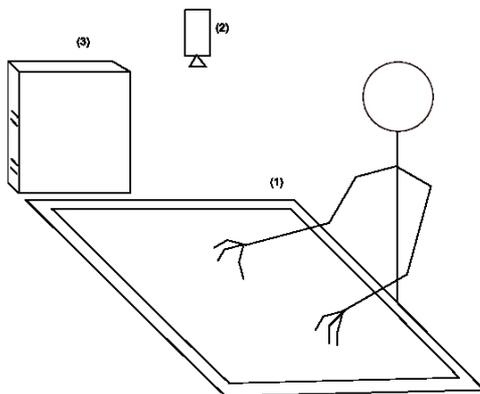


Figura 1. Componentes del Sistema

Para seguir el objetivo de este documento, los componentes de hardware (C1), (C2) y (C3) que conforman al sistema deben poseer las siguientes características: (a) Tener un costo relativamente bajo. (b) Evitar el uso de hardware especializado; hardware que debido a sus características especiales no se encuentra disponible en cualquier sitio y (c) tener independencia de hardware; esta característica asegura la reusabilidad de elementos de hardware para la implementación de los componentes físicos del sistema. Adicionalmente, esta característica permite que se pueda integrar al sistema dispositivos mas potentes y con mayor precisión que con el paso del tiempo experimentarán una disminución en sus costos. Para la implementación de (C4), las alternativas mencionadas en la sección de trabajo relacionado fueron comparadas en tres aspectos: (A1) ¿Qué tan natural la interacción con sistemas desarrollados es?, (A2) Portabilidad de los sistemas desarrollados, y (A3) Independencia de hardware de los sistemas desarrollados.

Respondiendo a la pregunta (A1), los sistemas construidos empleando tecnología FTIR son mas intuitivos y naturales. Entre las alternativas

consideradas solo Touchlib y Bespoke 3DUI XNA le emplean. HandVu sigue debido a su funcionalidad de reconocimiento de posturas dactilológicas. Al usar HandVu el usuario debe conocer las posturas dactilológicas que el sistema es capaz de reconocer para poder realizar una interacción fluida. En tercer lugar está ubicada la propuesta realizada por Lee, debido al requerimiento no natural de colocar cinta reflectiva en la punta de los dedos para mejorar la precisión de detección.

El aspecto (A2) es relacionado al número de sistemas operativos en los cuales el sistema puede correr. Según este criterio, HandVu encabeza la lista brindando la posibilidad de ser compilado y ejecutado en diferentes sistemas operativos incluyendo Windows, GNU Linux, etc.. Siguiendo en segundo lugar con versiones que están siendo portadas a GNU Linux se encuentra Touchlib y la propuesta de Johnny Lee terminando con Bespoke 3DUI XNA debido a sus dependencias con las clases del framework .NET que hacen que funcione correctamente sobre el sistema operativo MS Windows. En lo relacionado a Independencia de hardware (A3) la propuesta de Lee y Bespoke 3DUI XNA toman el último lugar debido a su dependencia total en el Wiimote. Touchlib sigue debido a la tecnología FTIR que toma como base dispositivos de proyección posterior/inferior. Esto deja a HandVu, que funciona con cualquier dispositivo de proyección y es compatible con algunos modelos de cámaras DV y webcams, en la punta.

Tabla 1  
Herramientas Disponibles

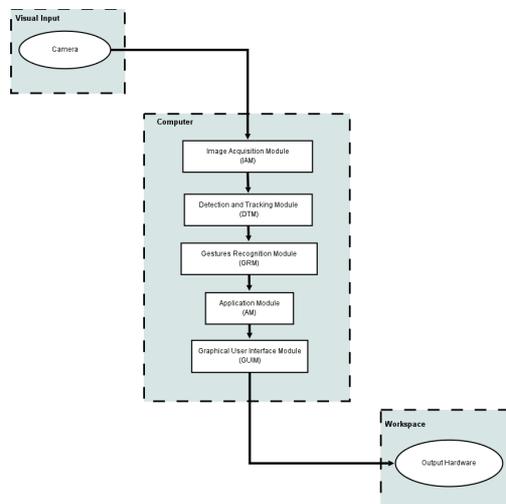
Alternativa	Aspecto de Evaluación		
	A1	A2	A3
Hacks de Wiimote	1	2	1
Touchlib	3	2	2
Bespoke XNA	3	1	1
Handvu	2	3	3

#### 4.2 Procesamiento del Sistema

Para lograr el reconocimiento de posturas y gestos dactilológicos para proveer retroalimentación y una interacción fluida, el diseño propuesto divide el procesamiento en cinco módulos. El Módulo de Adquisición de Imágenes (MAI) es el módulo que maneja la conexión con el dispositivo de captura; esto incluye tanto la conexión por medio de la interfaz de hardware como la transferencia de datos desde el dispositivo empleado como Entrada Visual encargado hasta el resto de software funcionando en el sistema. Es en este módulo donde la compatibilidad con diferentes dispositivos de entrada como cámaras IEEE1293 y webcams es manejada. Luego las imágenes capturadas por el MAI son enviadas al Módulo de Detección y Seguimiento (MDS). El MDS

escanea las imágenes recibidas para detectar y seguir la mano del usuario. Como se mencionó previamente, HandVu fue la herramienta seleccionada para encargarse de la detección y seguimiento de la mano del usuario y constituye la base sobre la cual se implementó el MDS.

Cuando una postura es detectada, la información relacionada con la ubicación e identificador de la postura es enviada al siguiente módulo. El módulo de Reconocimiento de Gestos (MRG) registra la información recibida del MDS y verifica si un gesto ha sido realizado. En caso de que un gesto dactilológico haya sido reconocido, el identificador y la ubicación de este son enviados al Módulo de Aplicación (MA). Este módulo implementa la lógica de la aplicación alojada en el sistema. Dependiendo de la aplicación algunas posturas y gestos dactilológicos van a ser reconocidos y otros van a ser ignorados. El MA genera la retroalimentación que va a ser presentada al usuario por medio del Área de Trabajo. Al final de su procesamiento, el MA actualiza la imagen que va a ser mostrada con la retroalimentación respectiva para luego enviarla al Módulo Gráfico (MG). El Módulo Gráfico se encarga de manejar los controles gráficos mostrados en la aplicación. Para esto se encarga de integrar controles disponibles en diferentes librerías gráficas.



**Figura 2.** Flujo de Procesamiento de los módulos del sistema

## 5 VirtualTable

Para poder evaluar el diseño planteado anteriormente, virtualTable una aplicación de manipulación de imágenes fue desarrollada. Esta aplicación es muy similar a las presentadas en varios productos donde los usuarios pueden mover, rotar y escalar imágenes. Cuando un usuario realiza una postura dactilológica

sobre una imagen, un instrucción es ingresada dependiendo de dicha postura. Luego, cuando un gesto es realizado, es empleado como una medida en la cual se va a realizar la instrucción. Suponiendo que el usuario realiza la postura de traslación sobre una imagen, a medida que el usuario desplace su mano la imagen cambiará de posición. Un principio similar es empleado para las otras acciones.

Para la implementación de virtualTable, un monitor LCD de 24 pulgadas fue usado como Área de Trabajo. Para la Entrada Visual se utilizó una webcam. Entre las cámaras probadas están la Logitech Quickcam, Creative NX y la Genius Messenger, con una resolución de 320x240, 640x480 y 640x480 respectivamente. Una de estas cámaras fue sujeta a un trípode y apuntada perpendicularmente al Área de Trabajo. Para realizar el proceso computacional se empleó un computador con un procesador dual core de 1.6 GHZ y 2GB de memoria. La adquisición de imágenes del MAI fue implementada empleando OpenCV, como fue mencionado previamente, HandVu es la herramienta base en la implementación del MDS. En el MRG, secuencias para el reconocimiento de gestos relacionados con la traslación, rotación y cambio de dimensiones de las imágenes fueron implementadas. En el MA, un set de funciones y métodos combinados con OpenCV son usados para manejar el procesamiento de imágenes en la aplicación. Finalmente en el MG, una interfaz gráfica de usuario minimalista fue desarrollada empleando la librería highgui incluida en el paquete de OpenCV.

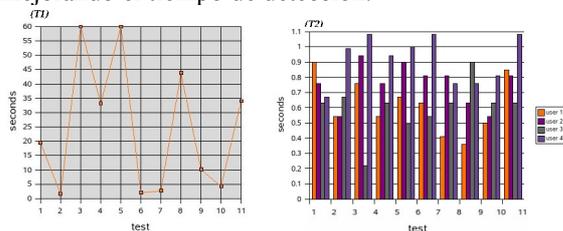
## 6 Proceso de Pruebas

La primera prueba relacionada a VirtualTable se realizó para verificar la eficacia y eficiencia de HandVu para reconocer posturas dactilológicas, adicional a esto, se implementó una pequeña aplicación para la traslación de imágenes para verificar si la integración de los módulos poseía un procesamiento que proveyera retroalimentación al usuario en tiempo real. Para esto la aplicación fue instalada en un arreglo de hardware en el cual el Área de Trabajo era un fondo oscuro controlado, muy similar a lo presentado en [7] pero con solo una webcam apuntando perpendicularmente esta área. En esta prueba al usuario se le solicitó trasladar imágenes de un lugar a otro, y fue durante esta prueba que las seis posturas dactilológicas que HandVu detecta por defecto fueron evaluadas con el objetivo de conocer las cuatro posturas detectadas con mayor precisión para ser usadas en la aplicación final.

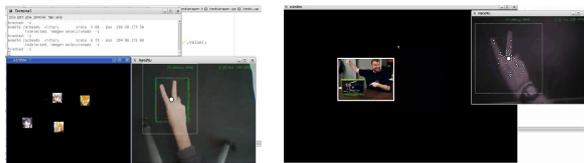
La siguiente prueba fue realiza sobre un Área de Trabajo de cambio constante. Para esto se empleó un televisor CRT obteniendo como resultado que el

tiempo de detección sobre fondos dinámicos fuera igual o mayor que un minuto, lo cual no hacia al sistema apropiado para interacción en tiempo real.

Para solucionar este problema, una webcam fue modificada para que funcione como cámara infrarroja. Esto se logró removiendo el filtro de luz infrarroja que esta poseía y agregando una lámina oscura de cinta fotográfica para que filtre la luz visible. Como las pantallas y proyectores solo proyectan luz visible, esta no es captada con la cámara y ya que la piel humana refleja la luz infrarroja, esto colocó las condiciones de esta prueba a condiciones similares a las del fondo controlado mejorando el tiempo de detección.



**Figura 3.** Tiempo de detección en escenarios de prueba.



**Figura 4.** Fondo controlado VS. Fondo dinámico filtrado con una cámara infrarroja.

## 7 Conclusiones y Trabajo Futuro

En este artículo un diseño para la implementación de superficies sensitivas de bajo costo fue propuesto. Este diseño se enfoca en la reducción de costos por medio de la reutilización de componentes convirtiéndose en una guía para la fácil implementación de superficies sensitivas, como fue mostrado anteriormente, el diseño permite el uso de pantallas LCDs o CRTs o proyectores y dependiendo de las cámaras disponibles, una mejor precisión de reconocimiento puede ser lograda. El módulo para el reconocimiento en el diseño propuesto fue desarrollado con HandVu para disminuir la carga cognitiva requerida para la implementación del sistema. Para de esta manera enfocar mas el trabajo en los aspectos de HCI mas que en lo de los algoritmos de Visión por Computador necesarios para el reconocimiento. Esta librería demostró ser lo suficiente robusta y las pruebas demuestran que su procesamiento es lo suficientemente rápido para

interacción en tiempo real. La portabilidad lograda por el diseño provee de sistemas que no solo corren en diferentes sistemas operativos, pero en diferentes arreglos de hardware también incluyendo superficies horizontales y verticales.

Durante la implementación de virtualTable algunos problemas fueron detectados y solucionados, sin embargo, mejoras y evaluaciones adicionales son recomendadas. Para mejorar la precisión de reconocimiento se recomiendo implantaciones del sistema con cámaras DV y/o webcams de mayor resolución. La mayoría de las pruebas fueron realizadas durante el día, por esto futuras pruebas durante la noche son recomendadas.

Una debilidad detectada en virtualTable, muy ligada a HandVu, es la restricción de ángulo que posee para el reconocimiento.

Finalmente, en lo relacionado a la interfaz gráfica de usuario, se propone la integración de librerías de sonido y motores de física en el MG con el objetivo de mejorar el flujo y experiencia de uso del sistema.

## 8 Referencias

- [1]. Christian von Hardenberg. Bare-hand human-computer interaction.
- [2]. Han, Jeff. Low-cost multi-touch sensing through frustrated total internal reflection. <http://portal.acm.org/citation.cfm?doid=1095034.1095054>.
- [3]. Ishii, Hiroshi. Tangible Bits: Towards Seamless Interfaces, between People, Bits and Atoms.
- [4]. Kolsch, Mathias. Fast 2D Hand Tracking with Flocks of Features and Multi-Cue Integration. <http://www2.computer.org/portal/web/csdl/abs/proceedings/cvprw/2004/2159/10/2159100158abs.htm>.
- [5]. Krueger Myron. VIDEOPLACE - an artificial reality. <http://portal.acm.org/citation.cfm?id=317463>.
- [6]. Lien C. Model-based articulated hand motion tracking for gesture recognition. <http://cat.inist.fr/?aModele=afficheN&cpsidt=216520>.
- [7]. Malik, Shahzad. Visual touchpad: a two-handed gestural input device. <http://portal.acm.org/citation.cfm?id=1027980>.
- [8]. Manresa, Cristina. Real Time Hand Tracking and Gesture Recognition for Human-Computer Interaction. <http://dmi.uib.es/ugiv/papers/ELCVIAManresa.pdf>.
- [9]. Nölker C. Illumination independent recognition of deitic arm postures.
- [10]. Quek F. Inductive learning in hand pose recognition. <http://portal.acm.org/citation.cfm?id=796035&dl=>.
- [11]. Sturman, D. Hands-on Interaction with Virtual Environments. <http://portal.acm.org/citation.cfm?id=73660.73663>.

[12]. Wang, Xiyang. Tracking of deformable human hand in real time as continuous input for gesture-based interaction.

[13]. Wu M. Multinger and whole hand gestural interaction techniques for multi-user tabletop displays.

<http://portal.acm.org/citation.cfm?id=964718>.