Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California

Programa de Posgrado en Ciencias en Ciencias de la Computación

---

Control de computadora basado en gestos con las manos en circunstancias de baja iluminación

Tesis

para cubrir parcialmente los requisitos necesarios para obtener el grado de

Maestro en Ciencias

Presenta:

América Ivone Mendoza Morales

Ensenada, Baja California, México

2015

Tesis defendida por

América Ivone Mendoza Morales

y aprobada por el siguiente Comité

Dr. Vitaly Kober

Director del Comité

Dr. Hugo Hidalgo Silva

Dr. Josué Álvarez Borrego

Dra. Ana Isabel Martínez García

Coordinador del Programa de Posgrado en Ciencias de la Computación

---

Dr. Jesús Favela Vara

Director de Estudios de Posgrado

Resumen de la tesis que presenta América Ivone Mendoza Morales como requisito parcial para la obtención del grado de Maestro en Ciencias en Ciencias de la Computación.

Control de computadora basado en gestos con las manos en circunstancias de baja iluminación

Resumen aprobado por:

Dr. Vitaly Kober

El reconocimiento de gestos con las manos ha sido un tema relevante en distintas áreas de las ciencias de la computación, por ejemplo en HCI es importante pues ayuda a crear una interacción natural entre la computadora y el usuario, por lo que se han desarrollado diversos métodos para encontrar el modelo que funcione en tiempo real y en diversas circunstancias. De manera que se pretende crear un modelo que fusione la información proporcionada por el dispositivo Kinect y haga el reconocimiento de gestos estáticos y dinámicos en tiempo real en circunstancias de baja iluminación y cuando existe oclusión. Dicho modelo será aplicado para crear un sistema que sirva como control de una computadora, es decir que los gestos puedan ser utilizados como el cursor de esta.

Palabras Clave: **Gestos con las manos, kinect, baja iluminación, oclusión.**

Abstract of the thesis presented by América Ivone Mendoza Morales as a partial requirement to obtain the Master of Science degree in Master in Computer Science in Computer Science.

Computer control based in hand gestures in circumstances of low illumination

Abstract approved by:

Dr. Vitaly Kober

The recognition of hand gestures has been prominent in different areas of computer science, eg. HCI is important because it helps create a natural interaction between the computer and the user, so have developed various methods to find the model that works in real time and in different circumstances. So it is to create a model that merges the information provided by the Kinect device, then the recognition of static and dynamic gestures in real time under conditions of low light and when there is occlusion. This model be applied to create a system that serves as a control computer, is that gestures can be used as the cursor.

Keywords: **Hand gestures, kinect, low illumination, occlusion.**

Dedicatoria

A ...

Agradecimientos

A ...

Al Centro de Investigación Científica y de Educación Superior de Ensenada.

Al Consejo Nacional de Ciencia y Tecnología (CONACyT) por brindarme el apoyo económico para realizar mis estudios de
maestría.

Tabla de contenido

Página

Tabla de contenido (continuación)

Lista de figuras

Figura Página

Lista de tablas

Tabla Página

Capítulo 1. Introducción

La interacción entre humanos se lleva a cabo gracias a la comunicación que existe entre ellos, esta puede ser oral o escrita, generalmente, por no decir siempre, viene acompañada de gestos realizados con la cara, manos, cuerpo. Estos gestos sirven como complemento de la comunicación ya ayudan a que nuestra idea se percibida de manera correcta.

El creciente desarrollo de la tecnología, a llevado a crear y estudiar distintas áreas de las ciencias computacionales, particularmente HCI (por sus siglas en íngles), la área encarga del estudio, diseño e interacción del humano con la computadora. Uno de los objetivos principales es que la interacción sea de manera natural. Por lo que no es extraño que los investigadores de HCI se hayan interesado en los gestos corporales, en especial los gestos realizados con las manos, para crear un ambiente natural entre el usuario y la computadora. Para obtener una interacción natural, entre estos dos actuadores, se necesita hacer el reconocimiento de los gestos, esto ha sido cada vez más sencillo gracias al avance de la tecnología, en especial en los dispositivos de visión como distintos tipos de cámaras, y al crecimiento en la capacidad de procesamiento de las computadoras. Aunque existen diversos métodos y sistemas para lograr el reconocimiento, no existe ninguno que nos pueda dar un reconocimiento totalmente preciso en todas las situaciones que se presentan en el mundo real.

Es por eso que se propone crear un sistema que haga el reconocimiento de gestos con las manos, en situaciones donde existe baja iluminación y cuando tenemos oclusión causada por los dedos. El sistema se enfoca principalmente en atacar los problemas de gestos con las manos que no tienen movimiento, y después se abordarán los gestos con las manos que involucran movimiento. El sistema aplicará los gestos como control de la computadora, esto con ayuda del dispositivo Kinect como herramienta para capturar la información de entrada.

1.1. Definición del problema

A finales de los anõs noventa se empezarón a desarrollar técnicas para reconocer gestos con las manos, las primeras fueron basadas en contacto y le siguieron las basadas

en la visión, estas fueron las más aceptadas debido a la facilidad de interacción entre el usuario, entre otras cosas, aunque estas tienen sus desventajas pues no es problema fácil de resolver debido a que existen distintas variables a considerar.

Aunque existen diversos métodos para el reconocimiento de gestos con las manos, con buena precisión, sigue siendo un problema abierto ya que no es fácil tener un sistema que se adecue a todo tipo de situaciones como: amigable con el usuario, invariante a la iluminación, rotación, al fondo, que funcione en tiempo real o cuando exista oclusión.

1.2. Justificación

1.3. Objetivo general

Desarrollar un sistema que permita controlar la computadora haciendo uso de gestos con las manos, estáticos y dinámicos. El sistema debe ser robusto, funcionar en circunstancias de baja iluminación, cuando exista oclusión en gestos dinámicos.

1.4. Objetivos especificos

- Identificar los métodos actuales de reconocimiento de gestos, estáticos y dinámicos cuando existe baja iluminación y en el caso de los gestos dinámicos cuando existe oclusión.
- Obtener conocimiento acerca del funcionamiento de sistema Microsoft Kinect.
- Desarrollar un sistema de reconocimiento de gestos estáticos y dinámicos, fusionando la información de los sensores de profundidad de dos dispositivo kinect. El sistema desarrollado deberá funcionar en circunstancias de baja iluminación y también cuando existe oclusión, causada por los dedos.
- Analizar el sistema diseñado, en cuanto a su eficiencia presentada en base al reconocimiento de los gestos y tiempo de respuesta, en circunstancias de baja iluminación y oclusión. En el análisis del sistema se usará información real.
- Comparar los modelos propuestos con los existentes, en base al tiempo de respuesta y la eficiencia en cuanto al reconocimiento del gesto.

1.5. Limitaciones y suposiciones

Gran porcentaje de los trabajos previos en el área de reconocimiento de gestos con las manos basados en el modelo de la visión utilizan cámaras digitales o cámaras web. Esta investigación utiliza el dispositivo Kinect, para obtener la información de entrada del sistema.

De manera que las limitaciones del sistema propuesto están dadas por las características dicho dispositivo, tales como la distancia a la que se encuentra el dispositivo con el usuario (poner la distancia), la resolución de las imágenes a color (poner resolucion) y la resolución del sensor infrarrojo (poner resolucion).
También el sistema depende de dos sensores Kinect, que se utilizarán en el caso que exista oclusión.
Otra limitante es el número de gestos que podrá reconocer el sistema.

Se supone el área de trabajo como un cuarto estándar con buena iluminación (enfocado a pruebas con la cámara color del sistema Kinect)

1.6. Reconocimiento de gestos con la manos

La definición de gestos (Mitra *et al.*, 2007) son movimientos del cuerpo expresivos y significativos que involucran a los dedos, manos, brazos, cabeza, cara o cuerpo con la intención de transmitir información relevante o de interactuar con el ambiente. De acuerdo con la literatura (Mitra *et al.*, 2007) los gestos con las manos se clasifican en estáticos y dinámicos, los primeros están definidos como la posición y orientación de la mano en el espacio manteniendo esta pose durante cierto tiempo, por ejemplo para hacer una señal de aventón, a diferencia de los gestos dinámicos donde hay movimiento de la pose, un ejemplo es cuando mueves la mano en señal de adiós. De aquí en adelante entiéndase el término gestos con las manos, como gestos.

El reconocimiento de gestos se divide en tres fasesRautaray y Agrawal (2012), detección o segmentación; extracción de características seguimiento; dependiendo si los gestos son dinámicos, por último la etapa final el reconocimiento del gesto. Este se clasifican en dos modelos, basados en la visión y en contacto, esta clasificación depende de la manera en que son capturados los datos, es decir la forma en que se obtiene el gesto,

para posteriormente poderlo reconocer.

Los primeros acercamientos para llevar acabo el reconocimiento de gestos fue usando modelos de contacto Rautaray y Agrawal (2012) y Nayakwadi (2014), como su nombre lo dice utilizan dispositivos que están en contacto físico con la mano del usuario, esto para capturar el gesto a reconocer, por ejemplo existen guantes de datos, marcadores de colores, acelerómetros y pantallas multi-touch, aunque estos no son tan aceptados pues entorpecen la naturalidad entre la interacción del humano y la computadora. Los modelos basados en la visión surgieron como respuesta a esta desventaja, estos utilizan cámaras para extraer la información necesaria para realizar el reconocimiento, los dispositivos van desde cámaras web hasta algunas más sofisticadas por ejemplo cámaras de profundida.

En este trabajo, se toma el enfoque basado en la visión ya que se quiere obtener un sistema que para el usuario sea facil de interactuar, y esta interacción sea natural y una manera de lograr esto es tomando este enfoque. estos tienen mayor complejidad (acomodra este parrafo :P)

Los métodos basados en la visión se pueden representar por dos modelos (Rautaray y Agrawal, 2012), los basados en 3D, da una descripción espacial en 3D de la mano, y los basados en apariencia, como su nombre lo dice se basan en la apariencia de la mano. Los modelos basados en apariencia se dividen en dos categorías, los estáticos (modelo de silueta, de contorno deformables) y de movimiento (de color y movimiento).

1.7. Estado del arte

La sección anterior explica los distintos enfoques para llevar acabo el reconocimiento de gestos, a continuación se encuentran los trabajos relevantes de cada uno de estos enfoques.

1.7.1. Modelos de contacto

(Yoon *et al.*, 2012) propone un modelo de mezclas adaptativo, usando un guante de datos, la principal limitante para este sistema es que solo reconoce gestos estáticos.

Aunque estos sistemas nos evitan algunos problemas que son consecuencia de los modelos basados en la visión, nos son perfectos, lo cual veremos enseguida.

Uno de los dispositivos recientes es MYO [1], aunque de este se hablará en la ultima parte de esta sección.

Como se describió en la sección anterior en los modelos de contacto la principal limitante es el uso de dispositivos en el cuerpo para el reconocimiento de los gestos, por esta razón la mayoría de los sistemas para el reconocimiento estan enfocados en modelos basados en visión. Por lo que resulta natural que la investigación propuesta tome un enfoque basado en la visión.

1.7.2. Modelos basados en la visión

Premaratne *et al.* (2013) realizan un modelo de reconocimiento de gestos estático y dinámico basados en el algoritmo de Lucas-Kanade. Las principales ventajas de este método son que es invariante a rotación, escala y al fondo. Aunque el modelo es afectado por los cambios en la iluminación.

(Huang *et al.*, 2011), propone un método para calculas gestos estáticos y dinámicos usando los filtros Gabor y haciendo una estimación del ángulo en el que se encuentra la mano. Las principales ventajas son que el sistema funciona con cambios en la iluminación y es robusto a la rotación y escala. La desventaja es que el problema de oclusión no es tratado.

(Mohd Asaari *et al.*, 2014) hacen el seguimiento de la mano para identificar los gestos dinámicos usando los filtros adaptativos Kalman y el método Eigenhand. Con esta combinación obtienen un excelente resultado pues el sistema es robusto a la iluminación, cambio de pose, y a la oclusión causada la mano oculta por algún objeto en movimiento.

A pesar que la mayoría de los modelos vistos en la parte de arriba solucionan muchos de los problemas de los modelos basados en la visión. Ninguno de ellos puede resolver el problema de iluminación y oclusión, formada por lo dedos. Allí la importancia de la investigación propuesta, pues dará solución a estos inconvenientes al momento de reconocer los gestos.

[1] https://www.thalmic.com/en/myo/

1.7.3. Sistemas comerciales

Existen dispositivos como: Leap Motion [2], MYO, y software, como Flutter [3], que realizan el reconocimiento de gestos, y estos los utilizan como reemplazo del ratón de la computadora.

Leap Motion es un dispositivo que detecta los movimientos de manos y dedos por medio de sensores infrarrojos. Leap Motion es robusto con el fondo, escala y rotación, pero no cuando existe oclusión pues cuando se realiza un zoom, como el que se hace en cualquier dispositivo touch, produce un error, y se presenta cuando un dedo es cubierto por otro, un problema grave es que tiene problemas de reconocimiento en circunstancias normales de luz.

MYO este dispositivo, solo se encuentra en pre-ordenamiento, detecta los impulsos eléctricos de tus músculos mediante tres sensores, giroscopio, acelerómetro y magnetómetros. MYO es un brazalete que promete controlar la computadora y dispositivos tales como el celular o la tableta. La principal desventaja del sensor es que gestos involuntarios pueden producir acciones no deseadas.

Flutter es un software que reconoce cuatro gestos estáticos detectando la palma de la mano, usando la cámara web como dispositivo de entrada. Flutter permite controlar aplicaciones multimedia de la computadora. Las limitaciones del software son que solo reconoce gestos estáticos, realiza acciones no deseadas al hacer gestos involuntarios y no siempre reconoce los gestos.

Aunque estos dispositivos y software para reconocer gestos solucionan algunos problemas importantes en el área, sigue existiendo el problema de oclusión e iluminación. De allí la importancia que existan modelos que puedan resolver estos problemas se presentan frecuentemente en el reconocimiento.

1.8. Organización de la tesis

La tesis se encuentra distribuida de la siguiente manera: la segunda sección presenta los fundamentos teóricos como base para la comprensión del tema. La tercera sección

[2] https://www.leapmotion.com/
[3] https://flutterapp.com/

presenta el sistema propuesto. En la cuarta sección se encuentran las pruebas realizadas al sistema junto con los resultados y la discusiones de estos. Finalmente la quinta sección presenta las conclusiones generales del sistema y el trabajo futuro.

Capítulo 2. Marco teórico

2.1. Gestos

Los gestos (Mitra *et al.*, 2007) son movimientos del cuerpo expresivos y significativos que involucran dedos, manos, brazos, cabeza, cara o cuerpo con la intención de transmitir información relevante o interactuar con el ambiente. De acuerdo con la literatura (Mitra *et al.*, 2007) los gestos con las manos se clasifican en estáticos y dinámicos, los primeros están definidos como la posición y orientación de la mano en el espacio manteniendo esta pose durante cierto tiempo, por ejemplo para hacer una señal de aventón, a diferencia de los gestos dinámicos donde hay movimiento de la pose, un ejemplo es cuando mueves la mano en señal de adiós. De aquí en adelante entiéndase el término gestos con las manos, como gestos.

2.2. Reconocimiento de gestos con la manos

El reconocimiento de gestos es

El reconocimiento de gestos se divide en tres fases (Rautaray y Agrawal, 2012), detección o segmentación; extracción de características seguimiento; dependiendo si los gestos son dinámicos, por último la etapa final el reconocimiento del gesto. Este se clasifican en dos modelos, basados en la visión y en contacto, esta clasificación depende de la manera en que son capturados los datos, es decir la forma en que se obtiene el gesto, para posteriormente poderlo reconocer.

Los primeros acercamientos para llevar acabo el reconocimiento de gestos fue usando modelos de contacto (Rautaray y Agrawal, 2012) y (Nayakwadi, 2014), como su nombre lo dice utilizan dispositivos que están en contacto físico con la mano del usuario, esto para capturar el gesto a reconocer, por ejemplo existen guantes de datos, marcadores de colores, acelerómetros y pantallas multi-touch, aunque estos no son tan aceptados pues entorpecen la naturalidad entre la interacción del humano y la computadora. Los modelos basados en la visión surgieron como respuesta a esta desventaja, estos utilizan cámaras para extraer la información necesaria para realizar el reconocimiento, los dispositivos van desde cámaras web hasta algunas más sofisticadas por ejemplo cámaras de profundida.

En este trabajo, se toma el enfoque basado en la visión ya que se quiere obtener un sistema que para el usuario sea fácil de interactuar, y esta interacción sea natural y una manera de lograr esto es tomando este enfoque. estos tienen mayor complejidad (acomodar este parrafo :P)

Los métodos basados en la visión se pueden representar por dos modelos (Rautaray y Agrawal, 2012), los basados en 3D, da una descripción espacial en 3D de la mano, y los basados en apariencia, como su nombre lo dice se basan en la apariencia de la mano. Los modelos basados en apariencia se dividen en dos categorías, los estáticos (modelo de silueta, de contorno deformables) y de movimiento (de color y movimiento).

2.2.1. Etapas del reconocimiento de gestos

Enseguida se describen las etapas del reconocimiento de gestos (detección, seguimiento y reconocimiento), con los métodos para llevar cada una de estas.

2.2.1.1. Detección

En esta etapa se detecta y segmenta la información relevante de la imagen (la mano), con la del fondo, existen distintos métodos para obtener dichas características como la de color de la piel, forma, movimiento, entre otras que generalmente son combinaciones de alguna de estas, para obtener un mejor resultado. Enseguida se describe brevemente cada una de estas.

- Color de la piel: Se basa principalmente en escoger un espacio del color, es una organización de colores especifica; como; RGB (rojo, verde, azul), RG (rojo, green), YCrCb (brillo, la diferencia entre el brillo y el rojo, la diferencia entre el brillo y el azul), etc. La desventaja es que si es color de la piel es similar al fondo, la segmentación no es buena, la forma de corregir esta segmentación es suponiendo que el fondo no se mueve con respecto a la cámara.
- Forma: Extrae el contorno de las imágenes, si se realiza correctamente se obtiene el contorno de la mano. Aunque si se toman las yemas de los dedos como características, estas pueden ser ocluidas por el resto de la mano, una posible solución es usar más de una cámara.

- Valor de píxeles: Usar imágenes en tonos de gris para detectar la mano en base a la apariencia y textura, esto se logra entrenando un clasificador con un conjunto de imágenes.
- Modelo 3D: Depende de cual modelo se utilice, son las características de la mano requeridas.
- Movimiento: Generalmente esta se usa con otras formas de detección ya que para utilizarse por sí sola hay que asumir que el único objeto con movimiento es la mano.

2.2.1.2. Seguimiento

Consiste en localizar la mano en cada cuadro (imagen). Se lleva acabo usando los métodos de detección si estos son lo suficientemente rápidos para detectar la mano cuadro por cuadro. Se explica brevemente los métodos para llevar a cabo el seguimiento.

- Basado en plantillas: Este se divide en dos categorías (Características basadas en su correlación y basadas en contorno), que son similares a los métodos de detección, aunque supone que las imágenes son adquiridas con la frecuencia suficiente para llevar acabo el seguimiento. Características basadas en su correlación, sigue las características a través de cada cuadro, se asume que las características aparecen en mismo vecindario. Basadas en contorno, se basa en contornos deformables, consiste en colocar el contorno cerca de la región de interés e ir deformando este hasta encontrar la mano.
- Estimación óptima: Consiste en usar filtros Kalman, un conjunto de ecuaciones matemáticas que proporciona una forma computacionalmente eficiente y recursiva de estimar el estado de un proceso, de una manera que minimiza la media de un error cuadrático, el filtro soporta estimaciones del pasado, presente y futuros estados, y puede hacerlo incluso cuando la naturaleza precisa del modelo del sistema es desconocida; para hacer la detección de características en la trayectoria.
- Filtrado de partículas: Un método de estimación del estado de un sistema que cambia a lo largo del tiempo, este se compone de un conjunto de partículas (muestras) con pesos asignados, las partículas son estados posibles del proceso. Es utilizado

cuando no se distingue bien la mano en la imagen. Por medio de partículas localiza la mano la desventaja es que se requieren demasiadas partículas, y el seguimiento se vuelve imposible.

- Camshift: Busca el objetivo, en este caso la mano, encuentra el patrón de distribución mas similar en una secuencia de imágenes, la distribución puede basada en el color.

2.2.1.3. Reconocimiento

Es la clasificación del gesto, la etapa final del reconocimiento, la clasificación se puede hacer dependiendo del gesto. Para gestos estáticos basta con usar algún clasificador o empatar el gesto con una plantilla. En los dinámicos se requiere otro tipo de algoritmos de aprendizaje de máquina. A continuación se encuentran los principales métodos para llevar acabo el reconocimiento del gestos.

- K-medias: Consiste en determinar los k puntos llamados centros para minimizar el error de agrupamiento, que es la suma de las distancias de todo los puntos al centro de cada grupo. El algoritmo empieza localizando aleatoriamente k grupos en el espacio espectral. Cada píxel en la imagen de entrada es entonces asignadas al centro del grupo mas cercano

- K-vecinos cercanos (KNN, por sus siglas en inglés): Este es un método para clasificar objetos basado en las muestras de entrenamiento en el espacio de características.

- Desplazamiento de medias: Es un método iterativo que encuentra el máximo en una función de densidad dada una muestra estadística de los datos.

- Máquinas de soporte vectorial (SVM, por sus siglas en inglés). Consiste en un mapeo no lineal de los datos de entrada a un espacio de dimensión más grande, donde los datos pueden ser separados de forma lineal.

- Modelo oculto de Markov (HMM, por sus siglas en inglés) es definido como un conjunto de estados donde un estado es el estado inicial, un conjunto de símbolos de salida y un conjunto de estados de transición. En el reconocimiento de gestos se

puede caracterizar a los estados como un conjunto de las posiciones de la mano; las transiciones de los estados como la probabilidad de transición de cierta posición de la mano a otra; el símbolo de salida como una postura especifica y la secuencia de los símbolos de salida como el gesto de la mano.

- Redes neuronales con retraso: Son una clase de redes neuronales artificiales que se enfocan en datos continuos, haciendo que el sistema sea adaptable para redes en linea y les da ventajas sobre aplicaciones en tiempo real.

2.3. Imagen

(Gonzalez) Una imagen se puede definir como una función bidimensional, $S(x, y)$ donde el valor de la función es la intensidad o el nivel de gris en el punto (x, y). Si el valor de la función y los puntos de la imagen son finitos, la imagen es una imagen digital.

2.4. Sensor Kinect

En noviembre del 2010 la compañia Microsoft lanzó el sensor Kinect para consolas de vídeo juego Xbox 360 y en febrero del 2011 lanzó la versión para Windows 1.

Figura 1: Sensor Kinect versión 1

El sensor esta equipado con los siguientes componentes: un cámara de color o sensor de color, un emisor infrarrojo, un sensor infrarrojo de profundidad, un motor que controla la inclinación, un arreglo de cuatro micrófonos y un LED 2.

Enseguida se describen brevemente cada uno de los componentes del sensor Kinect.

- La cámara de color captura y transmite datos en vídeo a color, detectando los colores rojo,verde y azul. La transmisión de datos que brinda la cámara es una secuencia de imágenes (cuadros), a una velocidad de 30 cuadros por segundo con

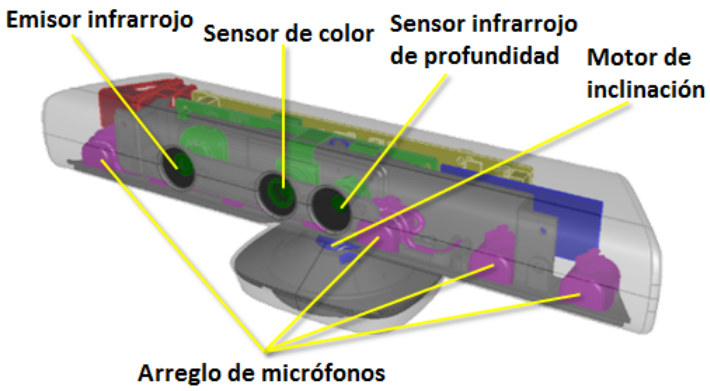

Figura 2: Componentes del sensor Kinect

una resolución de hasta $640\ x\ 480$ píxeles. La velocidad de los cuadros por segunda varia según la resolución de la imagen.

- El emisor infrarrojo proyecta puntos de luz infrarroja frente al sensor, con estos puntos y el sensor de profundidad se puede medir la profundidad.
- El sensor infrarrojo lee los puntos infrarrojos proyectados y calcula la distancia que existe entre el objeto y el sensor. El sensor transmite los datos de profundidad con una velocidad de 30 cuadros por segundo con una resolución de hasta $640\ x\ 480$.
- El motor de inclinación controla el ángulo de la posición vertical de los sensores del dispositivo. El motor puede moverse desde el ángulo de $-27°$ a $+27°$.
- Arreglo de micrófonos, consta de 4 micrófonos, captura el sonido y localiza la dirección en la que proviene.
- LED indica el estado del sensor.

2.5. Detección rápida de objetos usando características simples utilizando el clasificador de cascada impulsada

El método desarrollado por (ref Viola Jones) fue creado originalmente para atacar el problema de detección de rostros, este puede ser usando para detectar cualquier objeto, debido a la forma en que este fue creado, pues detecta un objeto clasificando imágenes basándose en el valor de características simples.

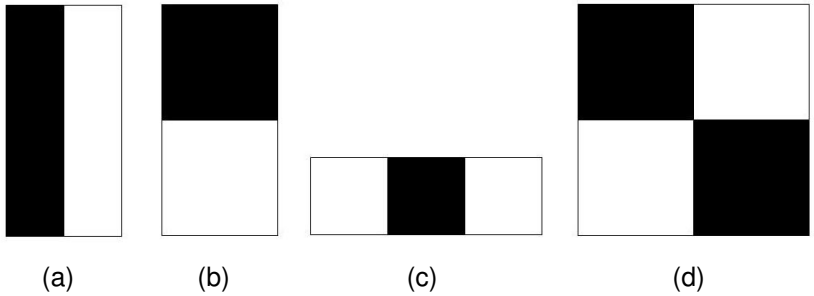

Figura 3: Ejemplo de operadores Haar

La técnica clasifica si el objeto se encuentra en la escena, usando AdaBoost en forma de cascada, y discrimina el objeto tomando en cuenta el valor de las características, se usan las características Haar, el valor de estas es calculado mediante una imagen integral.

(Parece que falta explicar mas el algoritmo de cascada)

En seguida se explica a detalle cada etapa del método.

2.5.1. Características Haar

Las características Haar son operadores rectangulares como los de la figura 3.
Las características con dos rectángulos 3(a), 3(b) , contienen dos regiones rectangulares adyacentes, y el valor de la característica se calcula tomando la diferencia de la suma de ambas regiones.
Las características con tres rectángulos 3(c), contienen tres regiones rectangulares adyacentes, y el valor de la característica se calcula la suma de las regiones exteriores y se resta la suma de la región interior.
Las características con cuatro rectángulos 3(d), contienen cuatro regiones rectangulares adyacentes, y el valor de la característica se obtiene con la diferencia entre las regiones pares diagonales.

2.5.2. Imagen integral

Uno de los aportes de este método es el concepto de imagen integral con la cual se calcula el valor de las características. La imagen integral, SI, se calculada como la suma del valor de los pixeles que se encuentran arriba y a la izquierda de cierta posición.

$$SI(x,y) = S(x,y) + S(x-1,y) + SI(x,y-1) - SI(x-1,y-1)$$

La imagen integral permite calcular la suma de los pixeles de cierta región usando solo los valores de las esquinas de dicha región, la cual se obtiene como:

$$REG(\alpha) = SI(A) + SI(D) - SI(B) - SI(C)$$

donde $REG(\alpha)$ es la región a la cual se le quiere calcular el valor de la suma de sus pixeles; A, B, C, D son las esquinas de dicha región, como se muestra en la figura 4

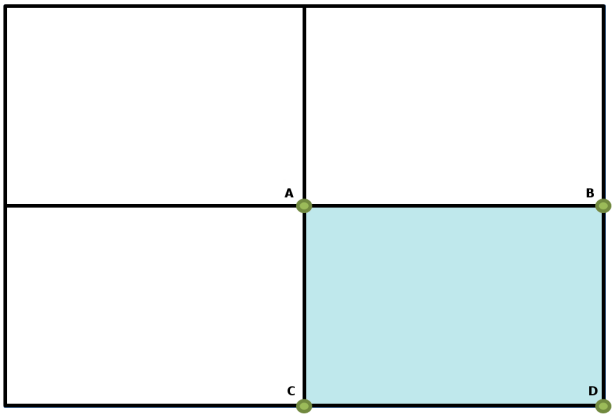

Figura 4: Regiones de imagen integral

2.5.3. Clasificador AdaBoost

El algoritmo AdaBoost realiza su clasificación construyendo un clasificador fuerte $h(x)$ de clasificadores débiles $h_i(x)$. Los clasificares débiles son calculados de la siguiente manera:

$$h_i(x) = \begin{cases} 1, & Si \quad p_i f_i < p_i \theta_i \\ 0, & de \quad otra \quad forma. \end{cases}$$

donde $f_i(x)$ es una característica, θ es un umbral, y $p_i(x)$ representa el signo de la desigualdad.

El clasificador fuerte es una combinación lineal de los clasificadores débiles, y se define de la siguiente forma:

$$h(x) = \alpha_1 h_1(x) + \alpha_2 h_2(x) + \cdots + \alpha_n h_n(x)$$

donde n es el número de características, α_i es el valor asociado a cada característica, el cual va entre 0 y 1.

2.5.4. Clasificador AdaBoost en Cascada

2.6. Binarización

La binarización es una técnica de procesamiento de imágenes, la cual se encarga de transformar una imagen en escala de grises $S(x, y)$ en una imagen binaria $B(x, y)$ es decir, los pixeles de la imagen toman un valor de 0 ó 1. Para formar la imagen binaria un valor, umbral, de la imagen en escala de grises es seleccionado. Ya que se tiene el umbral, T, los pixeles de la imagen son discriminados dependiendo si su valor es mayor o igual al umbral entonces el valor de los pixeles en la imagen binaria es 1 el resto toma valor de 0. Es decir:

$$B(x,y) = \begin{cases} 1, & Si \quad S(x,y) \geq T \\ 0, & de \quad otra \quad forma. \end{cases}$$

Existen diversas técnicas para binarizar una imagen, estas se pueden clasificar en dos grupos dependiendo de la manera es que es calcula el umbral, global o local. Los métodos globales calculan un umbral que es usado en toda la imagen y los métodos locales calculan varios umbrales para ciertas regiones de la imagen.

(citar) Un método de binarización muy utilizado es el de NiBlack, es un método local y adaptativo ya que adapta el umbral basándose en la media $m(i, j) =$ y la desviación estándar $\sigma(i, j)$ de una ventana deslizante de tamaño bxb. El umbral T se calcula como:

$$T(i,j) = m(i,j) + k \cdot \sigma(i,j)$$

1	1	1
1	1	1
1	1	1

(a) Rectángulo de $3x3$

0	0	1
1	1	1
0	0	1

(b) Figura de $3x3$

0	0	1	0	0
0	0	1	0	0
1	1	1	1	1
0	0	1	0	0
0	0	1	0	0

(c) Cruz de $5x5$

Figura 5: Ejemplos de elementos estructurales

donde $k \in [0, 1]$ el valor de la constante determina que tanta parte del contorno es preservado.

2.7. Operaciones Morfológicas

Otra técnica muy utilizada en procesamiento de imágenes son las operaciones morfológicas que son un conjunto de operaciones no lineales, la idea es que al aplicar alguna de estas operaciones el ruido se removido tomando en cuanta la forma y estructura de la imagen. La operaciones morfológicas utilizan un elemento estructural el cual se aplica por toda la imagen, los elementos estructurales pueden ser de distintas formas como 5

Existen distintas operaciones morfológicas, las principales o básicas son la dilatación y erosión las cuales se explican enseguida junto con la apertura y el cierra.

2.7.1. Dilatación

La dilatación es una operación que añade pixeles a la orilla de los objetos que se encuentran en la imagen. La dilatación se define como:

$$S \oplus EX = \{S | EX_S \subseteq S\}$$

donde EX_S es el elemento estructural trasladado con la imagen.

2.7.2. Erosión

La erosión remueve pixeles a la orilla de los objetos que se encuentran en la imagen. La erosión se define como:

$$S \ominus EX = \{S | EX_S \subseteq S\}$$

donde EX_S es el elemento estructural trasladado con la imagen.

2.7.3. Apertura

La operación apertura abre huecos entre objetos conectados por un enlace delgado de pixeles.

$$S \circ EX = (S \ominus EX) \oplus EX$$

2.7.4. Cierre

La operación cierre elimina huecos pequeños y rellena huecos en las

$$S \bullet EX = (S \oplus EX) \ominus EX$$

2.8. Casco convexo y defectos de convexidad

Antes de definir el casco convexo se presenta la definición de conjunto convexo.

Sea C un conjunto de puntos en el plano Euclidiano, el casco convexo es el conjunto convexo más pequeño que contiene a todos los puntos en C.

Los defectos de convexidad de un con casco convexo, es el conjunto de puntos que no pertenecen al casco convexo. El defecto es el espacio entre la linea y el objeto

2.9. Máquinas de soporte vectorial

N puntos de entrenamiento de dimensión D, dos clases distintas $y_i = -1$ o $+1$ es decir:

x_i, y_i donde $i = 1, \cdots, N$, $y \in -1, 1$, $x \in \Re^D$

Hiperplano óptimo

$$w \cdot x + b = 0$$

donde w es la normal al hiperplano, $\frac{b}{|w|}$ es la distancia perpendicular desde el hiperplano al origen.

$$w \cdot x + b = +1 \text{ para } y_i = +1$$

$$w \cdot x + b = -1 \text{ para } y_i = -1$$

Maximizar el margen, encontrar el mínimo de w.

$Min\ \|w\|$ tal que $y_i(w \cdots x_i + b) - 1 \geq 0$

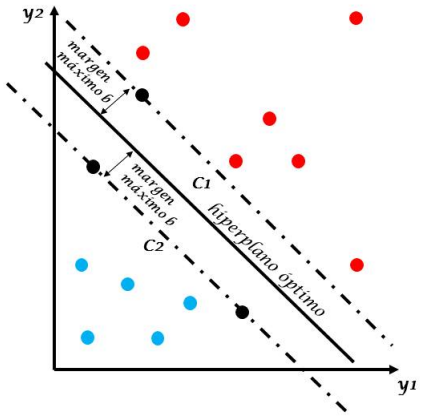

Figura 6: Clasificación de maquina de soporte usando kernel lineal

Capítulo 3. Sistema de reconocimiento de gestos propuesto

En este capítulo se describe el sistema de reconocimiento de gestos propuesto, este consta de cuatro etapas principales. La primera etapa es la adquisición de los datos, en la cual se capturan las imágenes de entrada del sistema; la segunda etapa es la detección aquí la mano es localizada y segmentada del fondo; en la etapa tres se extraen las características de la mano para ser procesadas en la etapa final donde el gesto realizado es reconocido.

3.1. Adquisición de los datos

En esta etapa se capturan los datos que son la entrada del sistema. Los datos provienen de los sensores de profundidad de dos dispositivos Kinect, estos se encuentran ubicados uno frente al usuario y otro al lado izquierdo como se muestra en la figura 7.

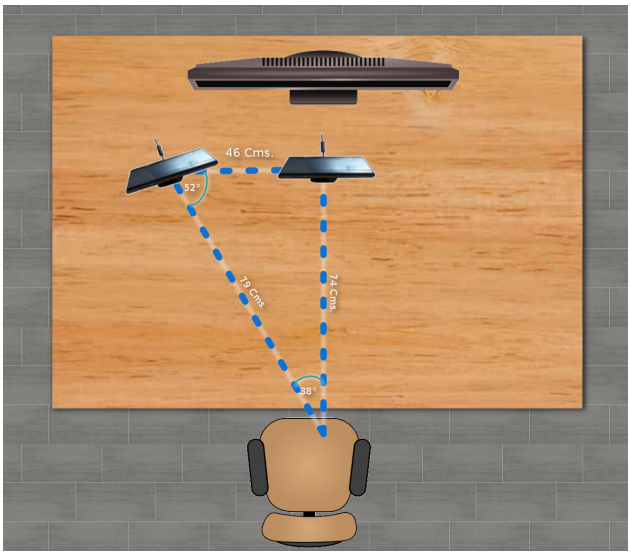

Figura 7: Configuración del sistema de reconocimiento de gestos con las manos

una vez que le flujo de datos de los sensores de profundidad es capturado este es representado como una imagen en escala de grises de 8 bits de 640 píxeles de ancho por 480 píxeles de largo. En las imágenes se puede apreciar detalles pequeños, es decir cambios en la profundidad de hasta $1\ mm$ esto debido a que la escala de grises inicia cada $26\ cm$. En la siguiente imagen se puede apreciar un ejemplo de las imágenes de profundidad. 8

Figura 8: Representación de los datos capturados por los Kinect

Debido a la naturaleza del funcionamiento del Kinect, las imágenes obtenidas contiene ruido del tipo (poner), el cual nos da una imagen como la figura , el ruido es reducido usando un filtros de mediana este es aplicado en toda la imagen en una ventada de tamaño 13. La imagen resultante $S(x, y)$ es como la que se muestra en la figura 8

Figura 9: Representación de los datos capturados por los Kinect

3.2. Detección

En esta etapa del sistema el objetivo es localizar y segmentar la mano para poder extraer las características necesarias para el reconocimiento. En este trabajo se utiliza el algoritmo de detección de objetos desarrollado por (citar viola jones), como se mostró en el capítulo 2 sección 2.5, el algoritmo clasifica las imágenes basándose en el valor de características.

La selección de las características se llevó acabo por medio del algoritmo AdaBoost; la implementación se realizó utilizando el software OpenCV Haar training classifier [1]. Se entrenó con 1000 imágenes positivas (imágenes de profundidad de la mano), y 2000 negativas, (imágenes de fondo a distintas profundidades). Las imágenes positivas fueron generadas de 100 imágenes de la mano usando el software Create Samples [2]. Todas las imágenes usadas fueron tomadas de nuestra base de datos [3].

Nuestra base de datos contiene gran cantidad de imágenes de profundidad. Imágenes de fondo y de mano, estas fueron tomadas a una distancia de entre $60\ cm$ y $200\ cm$. Las imágenes de profundidad de la mano fueron tomadas de 6 personas distintas con tres distintas poses: palma abierta con dedos abiertos, palma abierta con dedos juntos y finalmente el puño, como se muestra en la figura 10. Las imágenes de fondo fueron tomadas de distintos escenarios como se muestra en la figura 11. El programa para la captura de las imágenes puede ser encontrado en github [4].

Figura 10: Ejemplo de imágenes de la mano obtenidas de nuestra base de datos

Figura 11: Ejemplo de imágenes de la mano obtenidas de nuestra base de datos

[1] https://github.com/mrnugget/opencv-haar-classifier-training
[2] http://note.sonots.com/SciSoftware/haartraining.html
[3] https://github.com/americamm
[4] https://github.com/americamm

Para localizar la mano en cada cuadro proveniente de los Kinect, una ventana de tamaño $kahkjgv$ se desliza por la imagen, una vez que la mano se localiza la región de interés $Roi(x, y)$ es seleccionada alrededor de la mano, como se puede ver en la figura 13.

Figura 12: Mano seleccionada

Ya que se tiene localizada el área donde se encuentra la mano, el siguiente paso es segmentar la mano del Roi. El siguiente paso es binarizar el Roi, pero antes se aplican las operaciones morfológicas apertura y cerradura, en ese orden. la apertura se aplico con un elemento estructural rectangular de 3 píxeles de ancho 11 píxeles de altura, como el que se muestra en la figura (). La cerradura se aplicó también con un elemento estructural rectangular del tamaño tal. El resultado de estas operaciones es quitar las uniones pequeñas no deseadas causadas por el sensor, también juntar o cerrar hoyos. Las imágenes siguientes muestran el resultado de aplicar las operaciones apertura y cerradura al Roi.

Las operaciones anteriores son aplicadas antes de la binarización pues se obtiene un mejor resultado que aplicarlas después de la binarización. Para binarizar el Roi, se aplicó el algoritmo desarrollado por NiblacK (citar), se decidió usar este método debido a la naturaleza de la imagen. Los parámetros que fueron usados fue con $k = 0.5$ y una ventana de $3x3$ píxeles.

Como resultado obtenemos la imagen binarizada, donde se pueden ver los cambios en esta.

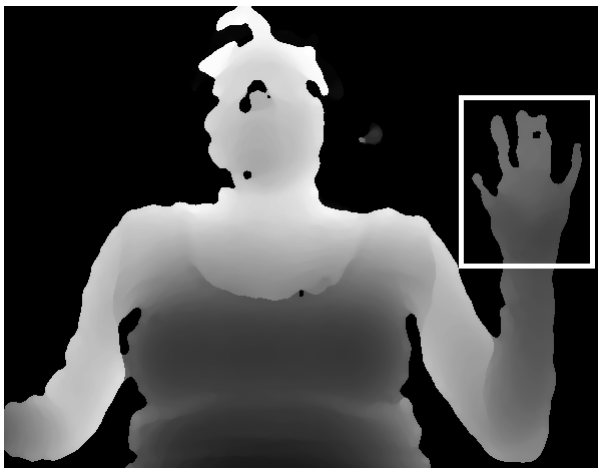

Figura 13: Mano seleccionada

3.3. Extracción de características

La idea de esta etapa es encontrar la características necesarias para reconocer el gestos realizado por la mano.

Las características son simple se extraen, el número de dedos, los ángulos entre ellos y también aquí se extrae la posición del centro de la mano. Para realizar la extracción se usan los algoritmos de casco convexo y el de defectos de convexidad.

3.4. Reconocimiento

3.4.1. Cálculo de la demanda de oxígeno

Capítulo 4. Resultados

Capítulo 5. Conclusiones

5.1. Trabajo futuro

Lista de referencias bibliográficas

(????). An Introduction to the Kalman Filter.

Arulampalam, M. S., Maskell, S., Gordon, N., y Clapp, T. (2002). A Tutorial on Particle Filters for Online Nonlinear / Non-Gaussian Bayesian Tracking. **50**(2): 174–188.

Bao, J., Song, A., Guo, Y., y Tang, H. (2011). Dynamic Hand Gesture Recognition Based on SURF Tracking. *Robot*, **33**(4): 482–489.

Bergh, M. V. D. (2010). Combining RGB and ToF Cameras for Real-time 3D Hand Gesture Interaction. pp. 66–72.

Chen, Q., Georganas, N. D., Petriu, E. M., Edward, K., Ottawa, A., y Kin, C. (2007). Real-time Vision-based Hand Gesture Recognition Using Haar-like Features.

Cheng, J., Xie, C., Bian, W., y Tao, D. (2012). Feature fusion for 3D hand gesture recognition by learning a shared hidden space. *Pattern Recognition Letters*, **33**(4): 476–484.

Chuang, Y., Chen, L., y Chen, G. (2014). Saliency-guided improvement for hand posture detection and recognition. *Neurocomputing*, **133**: 404–415.

Dominio, F., Donadeo, M., y Zanuttigh, P. (2013). Combining multiple depth-based descriptors for hand gesture recognition. *Pattern Recognition Letters*.

Duda, R. O., Hart, P. E., y Stork, D. G. (????). *Pattern Classification*. Segunda edición. p. 654.

Freeman, W. T. y Roth, M. (1994). Orientation Histograms for Hand Gesture Recognition.

Huang, D.-Y., Hu, W.-C., y Chang, S.-H. (2011). Gabor filter-based hand-pose angle estimation for hand gesture recognition under varying illumination. *Expert Systems with Applications*, **38**(5): 6031–6042.

Just, A. y Marcel, S. (2009). A comparative study of two state-of-the-art sequence processing techniques for hand gesture recognition. *Computer Vision and Image Understanding*, **113**(4): 532–543.

Kang, J., Zhong, K., Qin, S., Wang, H., y Wright, D. (2013). Instant 3D design concept generation and visualization by real-time hand gesture recognition. *Computers in Industry*, **64**(7): 785–797.

Mitchell, H. (2012). *Data Fusion: Concepts and Ideas*. Springer, segunda edición.

Mitra, S., Member, S., y Acharya, T. (2007). Gesture Recognition : A Survey. **37**(3): 311–324.

Mohd Asaari, M. S., Rosdi, B. A., y Suandi, S. A. (2014). Adaptive Kalman Filter Incorporated Eigenhand (AKFIE) for real-time hand tracking system. *Multimedia Tools and Applications*.

Murthy, G. R. S. y Jadon, R. S. (2009). A REVIEW OF VISION BASED HAND GESTURES RECOGNITION. **2**(2): 405–410.

Nayakwadi, V. (2014). Natural Hand Gestures Recognition System for Intelligent HCI : A Survey. **3**(1): 10–19.

Patwardhan, K. S. y Roy, S. D. (????). Dynamic Hand Gesture Recognition using Predictive EigenTracker.

Premaratne, P., Ajaz, S., y Premaratne, M. (2013). Hand gesture tracking and recognition system using Lucas[U+0096]Kanade algorithms for control of consumer electronics. *Neurocomputing*, **116**: 242–249.

Rautaray, S. S. y Agrawal, A. (2012). Vision based hand gesture recognition for human computer interaction: a survey. *Artificial Intelligence Review*.

Reifinger, S., Wallhoff, F., Ablassmeier, M., Poitschke, T., y Rigoll, G. (2007). Static and Dynamic Hand-Gesture Recognition for Augmented Reality Applications. pp. 728–737.

Ren, Z., Yuan, J., y Zhang, Z. (2011). Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera. *Proceedings of the 19th ACM international conference on Multimedia - MM '11*, p. 1093.

Sangineto, E. y Cupelli, M. (2012). Real-time viewpoint-invariant hand localization with cluttered backgrounds. *Image and Vision Computing*, **30**(1): 26–37.

Sgouropoulos, K., Stergiopoulou, E., y Papamarkos, N. (2013). A Dynamic Gesture and Posture Recognition System. *Journal of Intelligent & Robotic Systems*.

Shan, C., Tan, T., y Wei, Y. (2007). Real-time hand tracking using a mean shift embedded particle filter. *Pattern Recognition*, **40**(7): 1958–1970.

Shen, X., Hua, G., Williams, L., y Wu, Y. (2012). Dynamic hand gesture recognition: An exemplar-based approach from motion divergence fields. *Image and Vision Computing*, **30**(3): 227–235.

Tang, M. (????). Recognizing Hand Gestures with Microsoft [U+0092] s Kinect.

Wachs, J. P., Kölsch, M., Stern, H., y Edan, Y. (2011). Vision-based hand-gesture applications. *Communications of the ACM*, **54**(2): 60.

Yao, Y., Fu, Y., y Member, S. (2014). Contour Model based Hand-Gesture Recognition Using Kinect Sensor. **8215**(c): 1–10.

Yoon, J. W., Yang, S. I., y Cho, S. B. (2012). Adaptive mixture-of-experts models for data glove interface with multiple users. *Expert Systems with Applications*, **39**(5): 4898–4907.

Apéndice A. Apéndice

El apéndice...