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 mis Padres y Abuela

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 o cuerpo. Estos gestos sirven como complemento de la comunicación y ayudan a que el mensaje sea percibido de manera correcta.

El creciente desarrollo de la tecnología, a llevado a crear y estudiar distintas áreas de las ciencias computacionales, particularmente el área de interacción humano computadora (HCI, por sus siglas en inglés Human Computer Interaction), la área encargada del estudio y diseño de la forma en que el humano interactua con la computadora. Uno de los objetivos principales de esta área es que la interacción se lleve acabo de manera natural. No resulta 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. Por lo que es necesario que la computadora pueda identificar la o las manos del usuario y reconocer el gesto que este realiza.

A finales de los años noventa se empezarón a desarrollar técnicas para el reconocimiento de gestos con las manos. Los primeros acercamientos utilizaban sensores como: guantes de datos, marcadores de colores y acelerómetros; los cuales se colocaban en la o las manos para poder capturar la posición, la pose de la o las manos, entre otros datos para poder reconocer el gesto realizado. Las técnicas desarrolladas posteriormente obtienen la información necesaria de la mano usando distintos tipos de imágenes o vídeos, que son obtenidos mediante diversos tipos de cámaras.

Los métodos que utilizan imágenes o vídeo son los más utilizados ya que el usuario y la computadora interactuan de manera natural, el detalle con estos métodos es que es un problema difícil de resolver pues existen distintas variables que entran en juego para obtener una buena precision en el reconocimiento tomando en cuenta todas la variables.

Aunque existe gran variedad de métodos y sistemas que hacen el reconocimiento de gestos de las manos no existe alguno que su reconocimiento tenga un alto grado de precisión en todas las situaciones que se presentan en el mundo real.

Es por eso que se propone crear un sistema que reconozca gestos realizados con las manos, en situaciones que presentan baja iluminación y cuando existe oclusión de los dedos. El sistema se enfoca en atacar estos problemas cuando las manos no se encuentran en movimiento, pero también se abordarán los gestos con las manos que involucran movimiento. El objetivo del sistema es el de controlar la computadora mediante gestos, con la ayuda del sensor Kinect como herramienta para la captura de la información de entrada.

1.1. Definición del problema

Existen diversas técnicas que logran obtener buena precisión en el reconocimiento de gestos realizados con las manos, pero hay técnicas que puedan tener buena precision y que se adecue a todo tipo de situaciones de la vida real como: amigable con el usuario, invariante a la iluminación, rotación, al fondo, que funcione en tiempo real o cuando exista oclusión.

1.2. Justificación

Debido a la complejidad del problema de reconocimiento de gestos con las manos, las técnicas desarrolladas y actuales se enfocan en aspectos específicos para obtener un buen grado de precisión. De manera que se necesitan nuevos métodos que aborden los aspectos dejados de lado y funcionen no solo en condiciones ideales si no en situaciones que se presentan de manera natural y al mismo tiempo se obtenga un alto grado de precisión.

Una vez logrado lo anterior se pueden desarrollar nuevas aplicaciones y tecnologías que ayuden a interactuar con naturalidad al usuario y la computadora.

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 específicos

- 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 dispositivos 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 dos dispositivos Kinect, para obtener la información de entrada del sistema.

De manera que las limitaciones del sistema propuesto están dadas por las características de dicho dispositivo, tales como la distancia a la que se encuentran los dispositivos con el usuario y la resolución del sensor.

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.

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 aquí en adelante entiéndase el término gestos con las manos, como gestos.

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 profundidad.

En este trabajo, se toma el enfoque basado en la visión ya que se quiere obtener un sistema que para el usuario la interaccion interacción sea natural y una manera de lograr esto es tomando este enfoque.

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

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

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 (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.

1.7.3. 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. La cuarta sección se encuentra a detalle la implementación del sistema. En la quinta sección se presentan las pruebas realizadas al sistema junto con los resultados y la discusiones de estos. Finalmente la sexta sección presenta las conclusiones generales del sistema y el trabajo futuro.

Capítulo 2. Marco teórico

En este capítulo se definen una serie de conceptos importantes del área de procesamiento de imágenes y reconocimiento de patrones, estas definiciones son importantes para la comprensión del tema.

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 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 profundidad.

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

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

Una imagen se puede definir como una función bidimensional, $S(x, y)$ donde x, y representan las coordenadas en el plano y el valor de la función es la intensidad o nivel de gris en el punto (x, y). Si el valor de la función y los puntos de la imagen son finitos, esta es una imagen digital, la cual se puede representar en una matriz donde cada valor o pixel es el nivel de gris de la imagen, y los indices de esta indican la posición, (Gonzalez y Woods, 2002).

2.4. Oclusión

Se puede definir una oclusión como discontinuidades del movimiento y profundidad que se es percibida por un observador que se encuentra en movimiento en un ambiente estático.
Los puntos de oclusión en una imagen o cuadro son pixeles que aparecen o desaparecen en dos cuadros consecutivos, estos son llamados puntos de oclusión o punto de no oclusión, (Silva y Santos-Victor, 2001).

Existen tres tipos distintos de oclusiones la cuales depende de la forma en que es causada. Estas son: oclusión por el mismo objeto, entre objetos y por el fondo. La oclusión por el mismo objeto se presenta cuando parte del objeto ocluye a otra. La oclusión entre objetos es cuando dos objetos que se siguen se ocluyen entre ellos mismos. La oclusión por el fondo es cuando parte del fondo ocluye al objeto que se sigue, (Yilmaz *et al.*, 2006)

Capítulo 3. Sistema de reconocimiento de gestos propuesto

En este capítulo se describen las etapas del sistema de reconocimiento, junto con los métodos o herramientas que son utilizados en cada una de ellas.

El sistema de reconocimiento de gestos propuesto 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 el gesto realizado es reconocido.

3.1. Adquisición de los datos: sensor Kinect

Es la primera etapa del sistema, donde se capturan los datos que son la entrada del sistema. Los datos provienen de los sensores de profundidad de dos dispositivos Kinect. A continuación se describe las características de este dispositivo.

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, que se muestra en la figura 1.

El dispositivo Kinect esta equipado con una serie de sensores que permiten obtener imágenes a color y de profundidad (imágenes que indican a la distancia que esta un objeto del sensor), los cuales permiten hacer detección y seguimiento de personas. Detecta 6 personas y hace el seguimiento de 2 personas [1].

Figura 1: Sensor Kinect para Windows

El sensor esta equipado con los siguientes componentes: un cámara de color o sensor

[1] https://msdn.microsoft.com/en-us/library/hh973074.aspx

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.

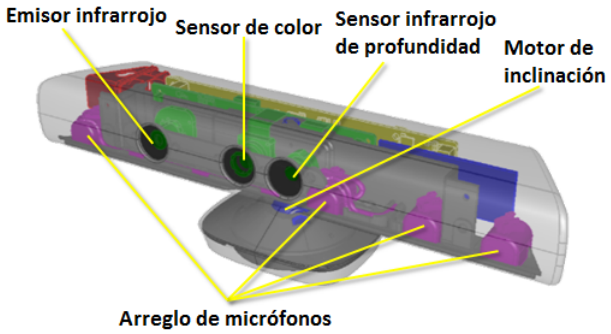


Figura 2: Componentes del sensor Kinect

Enseguida se describen brevemente cada uno de los componentes del sensor Kinect, (Jana, 2013).

- La cámara de color captura y transmite datos de vídeo a color, detectando los colores rojo, verde y azul (RGB, por sus siglas en inglés, red, green and blue). La transmisión de datos que brinda la cámara es una secuencia de imágenes (cuadros), a una velocidad de hasta 30 cuadros por segundo con una resolución de hasta $1280\,x\,960$ píxeles. La velocidad de los cuadros por segundo 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 que existe del sensor.

- 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 $640\,x\,480$ pixeles.

- 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.

3.2. Detección: detección rápida de objetos usando características simples utilizando el clasificador AdaBoost en forma de cascada.

En esta etapa del sistema el objetivo es localizar y segmentar la mano para extraer las características necesarias para el reconocimiento.

En este trabajo se utiliza el método detección rapida de objetos usando caracteristicas simples utilizando el clasificador AdaBoost en forma de cascada, (Viola y Jones, 2001), el cual 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.

La técnica clasifica si el objeto se encuentra en la escena, usando el clasificador AdaBoost (Freund y Schapire, 1995) en forma de cascada, y discrimina el objeto tomando en cuenta el valor de las características Haar (Viola y Jones, 2001), el valor de estas es calculado mediante el uso de una imagen integral (Viola y Jones, 2001).

Enseguida se explica a detalle cada etapa del método.

3.2.1. Características Haar

Las características Haar, son operadores rectangulares como los que se muestran en la figura 3. A continuación se explicaran los operadores Haar básicos:

- 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.

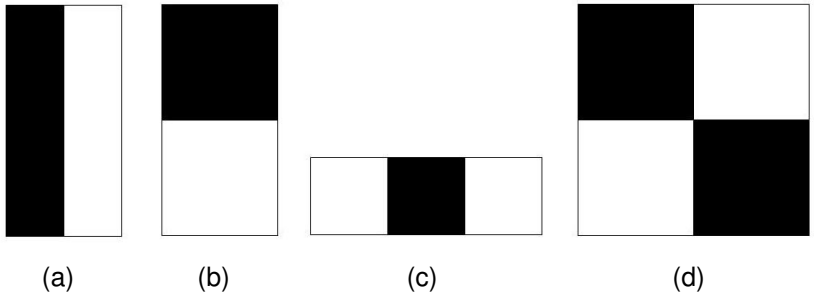


Figura 3: Ejemplo de operadores Haar

3.2.2. Imagen integral

Uno de los aportes del método desarrolla por Viola y Jones es el concepto de imagen integral con la cual se calcula el valor de las características. La imagen integral, SI, de un imagen, $S(x, y)$, es calculada como la suma del valor de los pixeles que se encuentran arriba y a la izquierda de cierta posición de la imagen a la cual se quiere hacer el cálculo. Lo anterior se puede escribir como: [2]

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

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: [3]

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

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

3.2.3. 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:

[2]https://computersciencesource.wordpress.com/2010/09/03/computer-vision-the-integral-image/
[3]https://computersciencesource.wordpress.com/2010/09/03/computer-vision-the-integral-image/

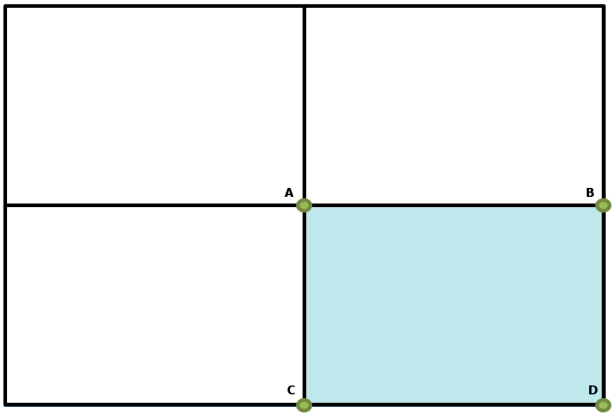


Figura 4: Regiones de imagen integral

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

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.

3.2.4. Clasificador AdaBoost en Cascada

3.3. 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.

Un método de binarización muy utilizado es el de NiBlack, (Chaki *et al.*, 2014) 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:

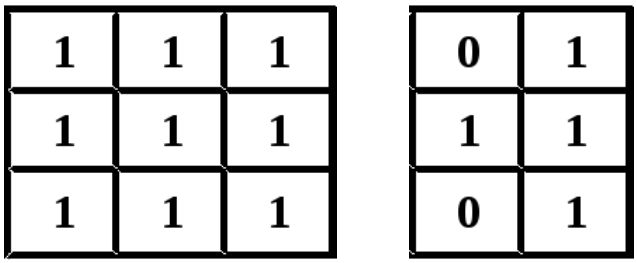
$$T(i,j) = m(i,j) + k \cdot \sigma(i,j)$$

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

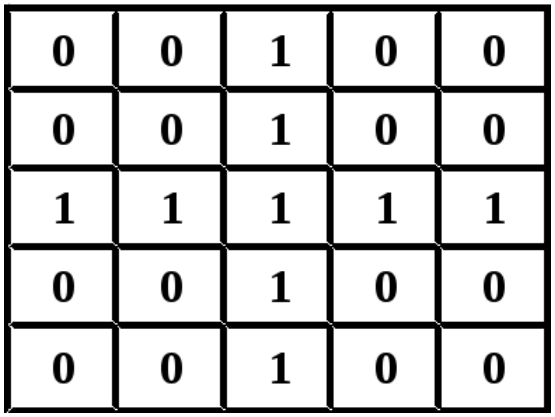
3.4. Operaciones Morfológicas

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.



(a) Rectángulo de $3x3$. (b) Figura de $3x2$.



(c) Cruz de $5x5$.

Figura 5: Ejemplos de elementos estructurales

3.4.1. 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.

3.4.2. 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.

3.4.3. Apertura

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

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

3.4.4. Cierre

La operación cierre elimina huecos pequeños y rellena huecos en las

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

3.5. Extracción de características: casco convexo y defectos de convexidad

La idea de esta etapa es encontrar las características que sean capaces de describir la mano, de manera que con estas características se pueda reconocer los gestos realizados por la o las manos. Las características se guardan en un vector, llamado vector de características, donde la dimension del vector es el número de características que describen, es este caso, la mano.

Las características que se extraen son geométricas, se extraen: el número de dedos, los ángulos entre ellos, el centro de la mano, la distancia del centro a la punta de los dedos, el área y perímetro de la mano.

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

3.6. Reconocimiento: máquinas de soporte vectorial

Es la etapa final del reconocimiento,es donde finalmente el gesto puede ser interpretado por la computadora.

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 $y_i(w \cdots x_i + b) - 1 \geq 0$

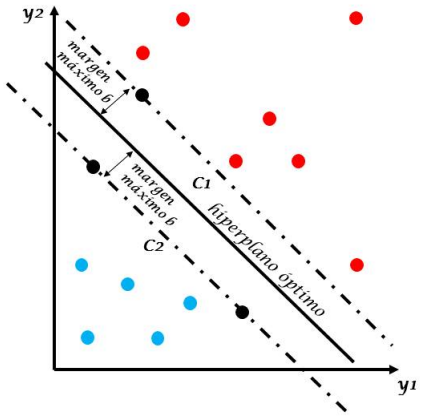


Figura 6: Clasificación de maquina de soporte usando kernel lineal

Capítulo 4. Implementación del sistema de reconocimiento de gestos propuesto

En este capítulo se describe los detalles de implementación del sistema.

4.1. 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.

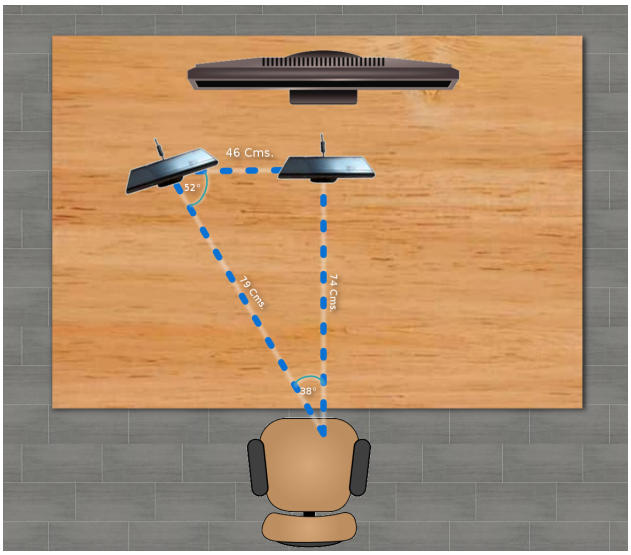


Figura 7: Configuración del sistema de reconocimiento de gestos con las manos

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

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 8: Representación de los datos capturados por los Kinect

Figura 9: Representación de los datos capturados por los Kinect

4.2. Detección

En esta etapa del sistema el objetivo es localizar y segmentar la mano para extraer las características necesarias para el reconocimiento. En este trabajo se utiliza el algoritmo de detección de objetos desarrollado por (Viola y Jones, 2001), como se mostró en el capítulo 2 sección 3.2, 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 ne-

[1] https://github.com/mrnugget/opencv-haar-classifier-training

gativas, (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: Imágenes de la mano de nuestra base de datos

Figura 11: Imágenes del fondo de nuestra base de datos

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 12.

[2] http://note.sonots.com/SciSoftware/haartraining.html
[3] https://github.com/americamm
[4] https://github.com/americamm

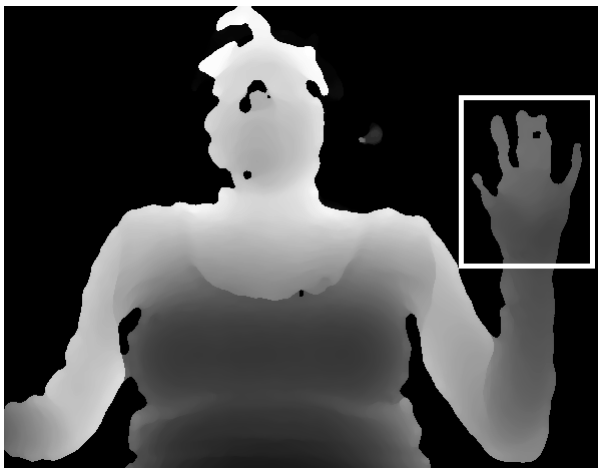


Figura 12: Mano seleccionada

Ya que se tiene localizada el área donde se encuentra la mano, el siguiente paso es segmentar la mano del ROI. La segmentación se realiza calculando los contornos existentes en la ROI y se toma el contorno más grande como el contorno de la mano, antes de calcular el contorno se realizan una serie de procesamientos al ROI para eliminar ruido y que el cálculo del contorno sea más preciso.

El primer procesamiento que se aplica al ROI, son las operaciones morfológicas de apertura y cerradura, en ese orden. Se utilizan para eliminar uniones pequeños que existen en la imagen como el que se muestra 13 o unir pequeños hoyos que existen en la imagen, como los que se encuentran en la figura 14.
La operaciones utilizan un elemento estructural rectangular; para la operación de apertura el tamaño del elemento $3\,x\,7$ pixeles; para la cerradura se aplicó con un tamaño $7\,x\,7$ pixeles.

El resultado de estas operaciones es mejorar la región de interés. Las imágenes siguientes muestran el resultado de aplicar las operaciones apertura y cerradura al ROI.

Una vez aplicadas las operaciones el siguiente paso es binarizar la región de interés, se lleva acabo aplicando el algoritmo desarrollado por (Niblack, 1985), se decidió usar este método debido a la naturaleza de la imagen. Los parámetros que fueron usados fueron $k = 0.5$ y una ventana de $3x3$ píxeles. La imagen siguiente es una imagen binarizada.

Una vez que la ROI es binarizada, el siguiente paso es encontrar los contornos existentes dentro de esta. Los contornos se calcularon utilizando el algoritmo de $blabla$ con

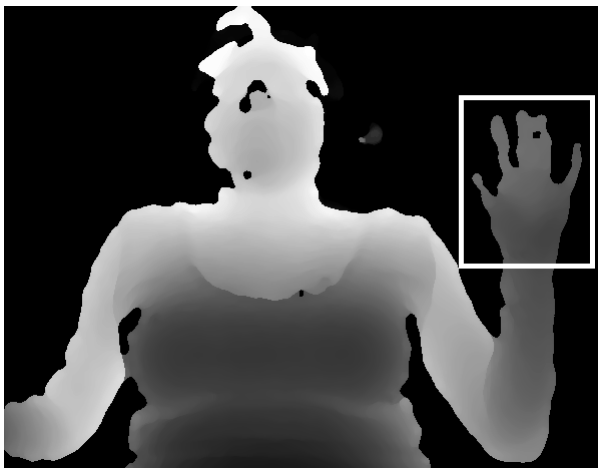


Figura 13: ROI que muestra una unión entre los dedos

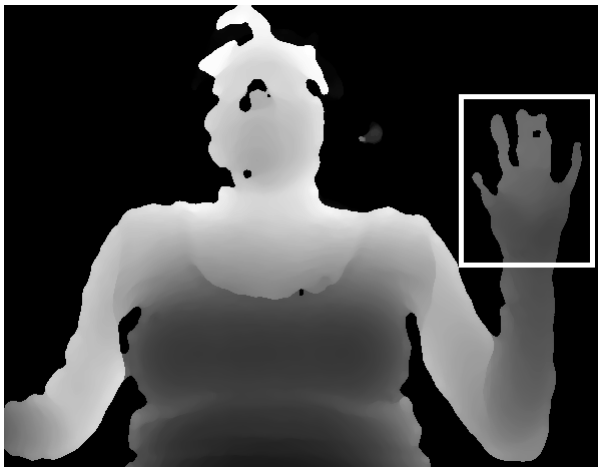


Figura 14: ROI donde se aprecio un hoyo en la mano

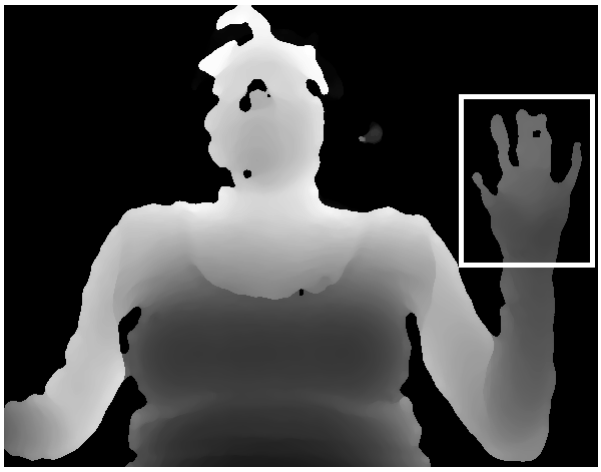


Figura 15: Apertura y cerradura

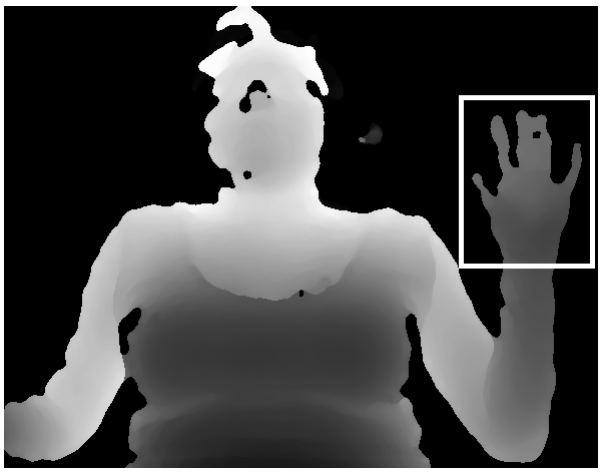


Figura 16: Binarización de ROI

tales parámetros.

Ya que le contorno de la mano es identificado, se calcula el casco convexo de la mano y posteriormente los defectos de convexidad

4.3. Extracción de características

Para la extracción de características se utilizan los algoritmos de casco convexo y el de defectos de convexidad, la figura muestra la mano con el casco convexo y sus defectos.

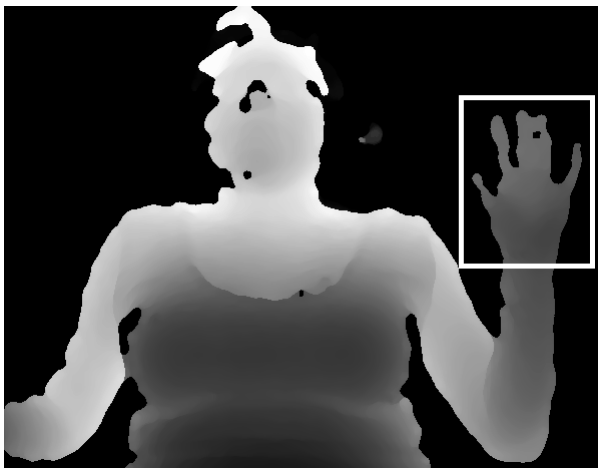


Figura 17: En esta dibujado el casco convexo los punto en son los defectos de convexidad

Una vez aplicados estos dos algoritmos se pueden calcular el número de dedos y las puntas de estos que son fundamentales para calcular las demás características. Enseguida se describe el algoritmo () para calcular el numero de dedos. Sea $CD =$

$cd_1, cd_2, \cdots, cd_n$ los defectos de convexidad de un conjunto casco convexo. Cada defecto esta compuesto de tres elementos $s_i(x,y), d_i(x,y), e_i(x,y) \in cd_i$. δ_i es la distancia de $d_i(x,y)$ a la orilla del casco convexo.

Entrada: d_i
Salida: Número de dedos, Nf.

1: **para** $i = 1$ hasta n **hacer**
2: $minDist = 20$, $maxAng = 60$, $antecesor = 0$, $sucesor = 0$.
3: **si** $\delta_i < minDepth$ **entonces**
4: continuar
5: **fin si**
6: **si** i=0 **entonces**
7: $antecesor = n - 1$
8: **si no**
9: $antecesor = i - 1$
10: **fin si**
11: **si** i=n-1 **entonces**
12: $sucesor = 0$
13: **si no**
14: $sucesor = i + 1$
15: **fin si**
16: Calcular el $angulo$ entre $s_{antecesor}(x,y)$ y $s_{sucesor}(x,y)$
17: **si** $angulo \geqslant maxAng$ **entonces**
18: **regresar falso**
19: **fin si**
20: $Nf = Nf + 1$.
21: **fin para**

Calculo del $angulo$

$$\alpha_{f_j} = \tan^{-1} \left| \frac{m_{j+1} - m_j}{1 + mj + 1m_j} \right|$$

$$\theta_{f_j} = \tan^{-1} |m_j - 90°|$$

Las características se guardan en un vector de características de dimensión 26.

4.4. Reconocimiento

Es la etapa final del reconocimiento,es donde finalmente el gesto puede ser interpretado por la computadora. El reconocimiento de los gestos estáticos se lleva acabo usando

el algoritmo de clasificación de MSV.

Capítulo 5. Resultados

Capítulo 6. Conclusiones

6.1. Limitaciones del sistema

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 de dicho dispositivo, tales como la distancia a la que se encuentra el dispositivo con el usuario, $0.4m$ a $3m$, la resolución de las imágenes a color $640\,x\,480$ pixeles y la resolución del sensor infrarrojo $640\,x\,480$ pixeles.
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).

6.2. Trabajo futuro

Lista de referencias bibliográficas

Chaki, N., Shaikh, S. H., y Saeed, K. (2014). *Exploring Image Binarization Techniques*. Springer, primera edición. p. 90.

Chang, C.-C. y Lin, C.-J. (2011). Libsvm. *ACM Transactions on Intelligent Systems and Technology*, **2**(3): 1–27.

Freund, Y. y Schapire, R. (1995). A desicion-theoretic generalization of on-line learning and an application to boosting. *Computational learning theory*, **55**(1): 119–139.

Gonzalez, R. y Woods, R. (2002). *Digital image processing*. p. 190.

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.

Jana, A. (2013). *Kinect for Windows SDK - Programming Guide - Face Tracking*. Packt, primera edición. p. 392.

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*.

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

Niblack, W. (1985). *An introduction to digital image processing*. Strandberg Publishing Company Birkeroed, Denmark,. p. 215.

Ong, K. C., Teh, H. C., y Tan, T. S. (1998). Resolving occlusion in image sequence made easy. *The Visual Computer*, **14**(4): 153–165.

Premaratne, P. (2013). *Human Computer Interaction Using Hand Gestures*. Springer, primera edición. p. 182.

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

Silva, C. y Santos-Victor, J. (2001). Motion from occlusions. *Robotics and Autonomous Systems*, **35**(3-4): 153–162.

Viola, P. y Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, **1**.

Ye, M., Zhang, Q., Wang, L., y Zhu, J. (????). A Survey on Human Motion Analysis. pp. 149–187.

Yilmaz, A., Omar, J., y Mubarak, S. (2006). Object tracking: a survey. *ACM Computing Surveys (CSUR)*, **38**(4): 45.

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.