



## **UNIVERSIDAD TECNOLÓGICA DE LA MIXTECA**

“EXTRACCIÓN AUTOMÁTICA DEL CONTORNO DE ROSTROS Y  
CARACTERÍSTICAS FACIALES”

TESIS:

PARA OBTENER EL TÍTULO DE  
**INGENIERO EN COMPUTACIÓN**

PRESENTA:

**ANTONIO CABALLERO BARBOSA**

DIRECTOR DE TESIS:

**M.I.A. HILDA CABALLERO BARBOSA**

HUAJUAPAN DE LEÓN, OAX. SEPTIEMBRE DE 2007



## UNIVERSIDAD TECNOLÓGICA DE LA MIXTECA

“EXTRACCIÓN AUTOMÁTICA DEL CONTORNO DE ROSTROS Y  
CARACTERÍSTICAS FACIALES”

TESIS:

PARA OBTENER EL TÍTULO DE  
INGENIERO EN COMPUTACIÓN

PRESENTA:

ANTONIO CABALLERO BARBOSA

JURADO

DR. NOMBRE DEL PRESIDENTE

PRESIDENTE

M.I.A. HILDA CABALLERO BARBOSA

VOCAL

DR. NOMBRE DEL SECRETARIO

SECRETARIO

HUAJUAPAN DE LEÓN, OAX.

SEPTIEMBRE DE 2007

*Por un gran ejemplo de vida y apoyo incondicional, dedico esta tesis con enorme gratitud y respeto a Toribio Caballero Aquino y a Carmen G. Barbosa Barragán, mis padres.*

*“El ejemplo no es la cosa más importante que influye sobre otros. Es la única cosa.”*

*Albert Schweitzer.*

# Agradecimientos

A mi querida directora de tesis: M. I. A. Hilda Caballero Barbosa por su asesoramiento científico, estímulo para seguir creciendo intelectualmente y enorme tolerancia.

A la Dra. Virginia Berrón Lara, al M. C. Moisés Emmanuel Ramírez Guzmán y a la M. C. Verónica Rodríguez López, por su colaboración y apoyo, así como por sus observaciones críticas en la redacción del trabajo, y su valiosa amistad.

Al Dr. Francesc Tarres y al Ing. Antonio Rama por haber autorizado el uso de la base de datos de imágenes *GTAV Face Database*, indispensable para el desarrollo de este proyecto.

A mis hermanos Elizabeth, Hilda y Rodolfo por su comprensión y apoyo en el transcurso de la vida.

Y especialmente a mis padres por su inmenso amor, apoyo y esfuerzo. Sin ellos esto no sería posible.

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Objetivo General . . . . .	7
1.2. Propuesta de Solución . . . . .	7
1.3. Esquema . . . . .	9
<b>2. Marco Teórico</b>	<b>10</b>
2.1. Definición de Imagen . . . . .	10
2.2. Definición de Espacio de Color RGB . . . . .	12
2.3. Sistema HSI: Conversión a Niveles de Gris . . . . .	13
2.4. Proyección del Histograma de una Imagen . . . . .	15
2.5. Binarización . . . . .	16
2.6. Preprocesamiento de la Imagen . . . . .	17
2.6.1. Convolución . . . . .	18
2.6.2. Filtro Gaussiano . . . . .	20
2.6.2.1. Diseño de Filtros Gaussianos . . . . .	22
2.7. Realce de Bordes . . . . .	24
2.7.1. Definiciones Básicas . . . . .	24
2.7.2. Gradiente . . . . .	25
2.7.3. Pasos en la Detección de Bordes . . . . .	27

2.7.3.1. Operador de Sobel . . . . .	27
2.8. Detección de Color de Piel . . . . .	28
2.9. Contorno Activo ó <i>Snake</i> . . . . .	32
2.9.1. Fuerza de Inflación o <i>Balloon Force</i> . . . . .	34
2.9.2. Algoritmo de Minimización de Energía . . . . .	36
2.10. Modelo Geométrico del Rostro . . . . .	38
<b>3. Desarrollo del Sistema</b>	<b>40</b>
3.1. Alcances de la Implementación . . . . .	42
3.2. Descripción de la Implementación del Sistema . . . . .	44
3.2.1. Localización de los Rasgos Faciales . . . . .	46
3.2.1.1. Detección de la Silueta de la Cabeza de la Persona . . . . .	46
3.2.1.2. Ubicación de la Parte Superior del Rostro. . . . .	56
3.2.1.3. Ubicación Aproximada del Contorno del Rostro . . . . .	57
3.2.1.4. Proyección Horizontal y Vertical del Histograma en el Área Superior del Rostro . . . . .	59
3.2.1.5. Modelo Geométrico del Rostro para Determinar la Posi- ción Aproximada de la Nariz y la Boca . . . . .	69
3.2.1.6. Proyección Vertical Sobre el Área de la Nariz y Boca . . . . .	70
3.2.2. Localización del Contorno del Rostro . . . . .	79
3.2.2.1. Inicialización del Contorno Activo . . . . .	80
3.2.2.2. Modelo de Contorno Activo para Detectar el Contorno del Rostro . . . . .	100
3.2.2.3. Algoritmo Voraz . . . . .	102
<b>4. Experimentos y Resultados</b>	<b>105</b>
4.1. Experimentos y Evaluación . . . . .	106

4.2. Casos de Funcionamiento No Adecuado . . . . .	118
4.3. Comparación con Otros Sistemas Similares . . . . .	121
<b>5. Conclusiones y Trabajos Futuros</b>	<b>127</b>
5.1. Conclusiones . . . . .	127
5.2. Trabajos Futuros . . . . .	129
<b>A. Apéndice</b>	<b>131</b>
A.1. Manual de Usuario . . . . .	131
<b>Bibliografía</b>	<b>140</b>

# Capítulo 1

## Introducción

La detección del contorno del rostro y de las características faciales es una tarea importante en sistemas de análisis y procesamiento automático de rostros. Por ejemplo, los resultados de la detección pueden utilizarse en el reconocimiento de rostros [16, 33], en el seguimiento de personas [26], en el análisis de expresiones faciales para el reconocimiento de emociones [14, 7], para monitorear actividades humanas, en sistemas de vigilancia, video conferencias, e interfaces inteligentes [13]; se requieren métodos más efectivos y amigables para la interacción humano-computadora (ó Human Computer Interaction -HCI-), los cuales no hagan uso de dispositivos tradicionales tales como teclados, ratones, y monitores.

La detección del rostro es el primer paso necesario en los sistemas de reconocimiento de rostros, con el propósito de localizar y extraer la región del rostro a partir del fondo. El rostro humano es un objeto dinámico y tiene un alto grado de variabilidad en su apariencia, lo cual hace de la detección de rostros un problema difícil dentro de la visión por computadora. La detección del rostro es una de las tareas visuales que los humanos pueden hacer sin esfuerzo. Sin embargo, en términos de visión por computadora, esta tarea no es fácil. Una definición general del problema proporcionada por Yang y Kriegman en [33] es: Dada una imagen arbitraria, la meta de la detección del rostro es determinar si hay o no algún rostro en la imagen, si lo hay, regresar la localización y tamaño de cada rostro. La solución al problema

involucra segmentación, extracción, y verificación de rostros y posiblemente rasgos faciales a partir de un fondo no controlado. Un sistema de detección de rostros debería también ser capaz de llevar a cabo la tarea a pesar de la iluminación, orientación, y distancia de la cámara. Los retos asociados con la detección del rostro pueden atribuirse a los siguientes factores [33]:

- **Pose.** Las imágenes de cada rostro varían según la pose relativa cámara-rostro (frontal, 45 grados, perfil), y de algunas características faciales (un ojo o la nariz podrían llegar a estar parcial o totalmente ocluidas).
- **Presencia o ausencia de componentes estructurales.** Las características faciales tales como barba, bigote y lentes podrían o no estar presentes, además de haber gran variabilidad entre estos componentes incluyendo forma, color y tamaño.
- **Expresión facial.** La apariencia del rostro es directamente afectada por la expresión facial de la persona.
- **Oclusión.** Los rostros podrían estar parcialmente ocultos por otros objetos. En una imagen con un grupo de gente, algunos rostros podrían ocluir parcialmente otros.
- **Orientación de la imagen.** Las imágenes de rostros varían directamente debido a diferentes rotaciones sobre el eje óptico de la cámara.
- **Condiciones de la imagen.** Cuando la imagen se forma, factores tales como la luz (espectro, distribución de la fuente e intensidad) y características de la cámara (respuesta del sensor, lentes) afectan la apariencia de un rostro.

Los primeros trabajos en el área de la detección de rostros datan de principios de los 70's, los cuales utilizaban técnicas sencillas basadas en heurísticas y medidas antropomórficas. Estos sistemas eran muy rígidos, lo cual dificultó el crecimiento de esta línea de investigación hasta 1990, cuando sistemas prácticos para el reconocimiento de rostros y la codificación

de video se hicieron una realidad. Desde entonces ha crecido el interés en investigar varios de los aspectos importantes de la detección de rostros. Se han desarrollado esquemas de segmentación más robustos, particularmente aquellos que utilizan movimiento, color o información generalizada. El uso de estadísticas y redes neuronales también ha facilitado la detección de rostros en escenas complejas. Además se han realizado numerosos avances en el diseño de extractores de características, tales como los modelos deformables y contornos activos, los cuales pueden localizar y seguir características faciales correctamente.

Debido a que las técnicas para la detección de rostros requieren de información a priori del rostro, éstas pueden organizarse en dos categorías que se diferencian por su enfoque en la utilización de dicha información [13]:

- La primer categoría o enfoque se conoce como basado en características. En las técnicas de esta categoría el éxito en la detección de rostros se consigue manipulando medidas de distancia, ángulos, y área de las características visuales derivadas de la escena. El desarrollo del enfoque basado en características puede dividirse a su vez en tres áreas:
  - Análisis de bajo nivel: Realiza la segmentación de las características visuales utilizando las propiedades del píxel tales como el nivel de gris y color.
  - Análisis de características: Utiliza información de la geometría del rostro. Se reducen ambigüedades y se determina la ubicación del rostro y de las características faciales.
  - Modelos de forma activa: Representan la apariencia real de las características. Una vez que el modelo de forma activa se libera próximo a una característica, éste interactuará con las características locales de la imagen (como bordes o brillantez) y gradualmente se deformará para tomar la forma de la característica.

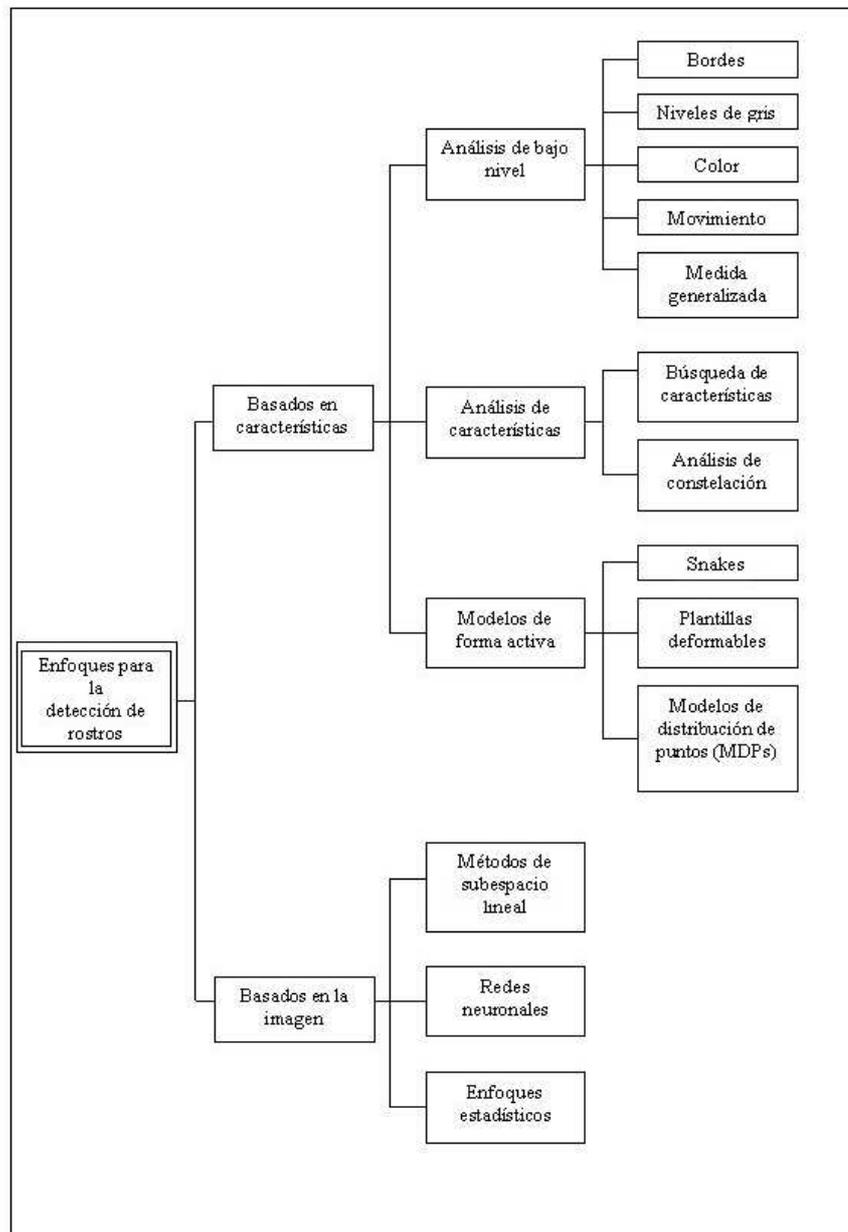
- Las técnicas de la segunda categoría hacen uso de los avances de la teoría de reconocimiento de patrones, tratan a la detección de rostros como un problema general de reconocimiento. Las técnicas de este enfoque clasifican como rostro a las representaciones de rostros basadas en la imagen (como por ejemplo, arreglos bidimensionales de intensidades), para ello utilizan algoritmos de entrenamiento sin el análisis y derivación de características. Estas técnicas pueden utilizar: Métodos de subespacio lineal, redes neuronales o enfoques estadísticos.

La Figura 1.1 resume lo anterior de manera esquemática.

Hay varios problemas estrechamente relacionados con la detección de rostros:

- **La localización del rostro** tiene como propósito determinar la posición de la imagen de un único rostro. Esta es una simplificación del problema de detección con el supuesto que una imagen de entrada contiene sólo un rostro.
- **La detección de los rasgos faciales** tiene como meta detectar la presencia y localización de características, tales como ojos, nariz, cejas, boca, labios, entre otras, con la suposición de que hay sólo un rostro en la imagen.
- **El reconocimiento o identificación del rostro** compara una imagen candidata contra una base de datos (galería de imágenes) y reporta una coincidencia, si la hay. El propósito de la autenticación del rostro es verificar la identidad de un individuo en una imagen de entrada.
- **Los métodos de seguimiento de rostros** continuamente estiman la localización y posiblemente la orientación de un rostro en una secuencia de imágenes en tiempo real.
- **El reconocimiento de expresiones faciales** se ocupa de identificar los estados afectivos (felicidad, tristeza, disgusto, entre otros) de los humanos.

Evidentemente, la detección del rostro es el primer paso en cualquier sistema automatizado de procesamiento de rostros. Por lo que el sistema que se desarrolló en este trabajo de tesis está orientado al problema de la localización del contorno del rostro y de características faciales: Ojos, nariz y boca, desde una perspectiva del enfoque basado en características: Empleando para la detección del contorno del rostro un modelo de forma activa y para la localización de las características faciales un método de análisis de características.



**Figura 1.1:** Enfoques empleados en la detección de rostros (tomada de [13]).

## 1.1. Objetivo General

Desarrollar un sistema que determine el contorno del rostro y las características faciales de manera automática, con técnicas del enfoque basado en características y sobre imágenes a color que presenten una vista frontal de la persona.

## 1.2. Propuesta de Solución

Para detectar el contorno del rostro se utilizó un modelo de forma activa conocido como contorno activo o *snake*, y las características faciales: Ojos, nariz y boca, se localizan mediante información basada en el histograma y en un modelo geométrico del rostro.

Los contornos activos (llamados así debido a la naturaleza de su comportamiento), se utilizan comúnmente para localizar el contorno de la cabeza [12, 21]. Se trata de modelos deformables formulados como un problema de minimización de energía. El modelo original del contorno activo fue presentado por Kass y Terzopoulos en [18] y ha sido utilizado exitosamente en diferentes problemas de visión por computadora y procesamiento de imágenes, tales como la detección de contornos, segmentación y seguimiento de objetos [13]. Los detalles sobre el modelo del contorno activo utilizado en este trabajo se abordan en las secciones 2.9 y 3.2.2.

Para la localización de los rasgos faciales se emplea información del histograma de la imagen y un modelo geométrico, este último consiste de rectángulos que envuelven a las características faciales: Ojos, nariz y boca. Se toma como una primera aproximación el modelo geométrico del rostro propuesto en el artículo de Shih y Chiang [28], es decir, se utiliza la configuración entre ojos, fosas nasales y boca, tomando como referencia que en la mayoría de los rostros, la distancia vertical entre los ojos y la nariz y entre los ojos y la boca son proporcionales a la distancia horizontal entre los ojos, partiendo de esta información se

aplica posteriormente una proyección del histograma para ubicar la posición exacta de las características faciales. Los detalles se mencionan en las secciones 2.10 y 3.2.1.

Por lo que para cumplir con el objetivo planteado el sistema que se desarrolló en este trabajo consta de dos etapas principales:

1. **Localización de Rasgos Faciales:** En la localización de los ojos se empleó información del histograma de la imagen en niveles de gris, habiendo previamente ubicado a la persona dentro de la imagen y delimitado la búsqueda de los ojos dentro del área del rostro, esta área fue encontrada con ayuda de la detección del color de piel. Conociendo la distancia de separación de los ojos se aplicó el modelo geométrico, que utiliza esta distancia como medida de proporción con respecto a los demás rasgos faciales para tener una aproximación a éstos, y finalmente haciendo uso de la información del histograma se determinó de manera precisa la posición de la nariz y la boca. El modelo geométrico consiste de cuatro rectángulos, dos para los ojos, uno para la nariz y uno para la boca.
2. **Localización del Contorno del Rostro:** Para determinar el contorno del rostro este sistema emplea un contorno activo, el contorno activo consiste de un conjunto de puntos o *saxels* inicializados cerca del área a segmentar, al ser sensible a la inicialización fue necesario aproximarlos lo más posible al contorno del rostro con ayuda de la información del color de piel, de la eliminación de regiones falsas de piel, límites del ancho del rostro y límite de la barbilla. Se construyó el contorno activo a través de una curva cerrada y con una representación discreta. El contorno activo es atraído a los bordes del rostro por medio de una minimización de energías, una de ellas es la energía de la imagen, la cual se obtuvo al aplicar un suavizado Gaussiano a la imagen en niveles de gris y un operador de Sobel para generar bordes. Y finalmente se utilizó un algoritmo voraz para hacer converger al contorno activo.

### **1.3. Esquema**

El presente trabajo está organizado en cinco capítulos y un apéndice, el primer capítulo contiene la introducción del trabajo y está enfocado a mostrar la importancia del problema y su propuesta de solución. Se plantea aquí el objetivo a alcanzar y la organización del documento.

En el segundo capítulo se hace una revisión de la teoría necesaria para llevar a cabo el objetivo propuesto.

El tercer capítulo se enfoca al diseño e implementación del sistema desarrollado en este trabajo de tesis.

En el cuarto capítulo se analiza la eficiencia del sistema a través de los resultados obtenidos.

El quinto capítulo describe las conclusiones obtenidas y los trabajos futuros.

Y finalmente, el apéndice A corresponde al manual de usuario del software desarrollado.

# Capítulo 2

## Marco Teórico

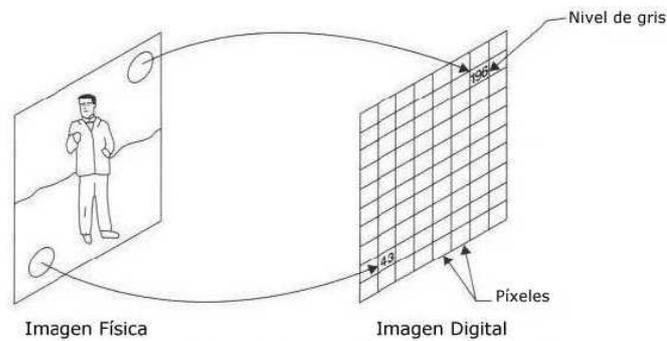
En esta sección se abordarán los fundamentos teóricos de las técnicas que se utilizaron en este proyecto.

### 2.1. Definición de Imagen

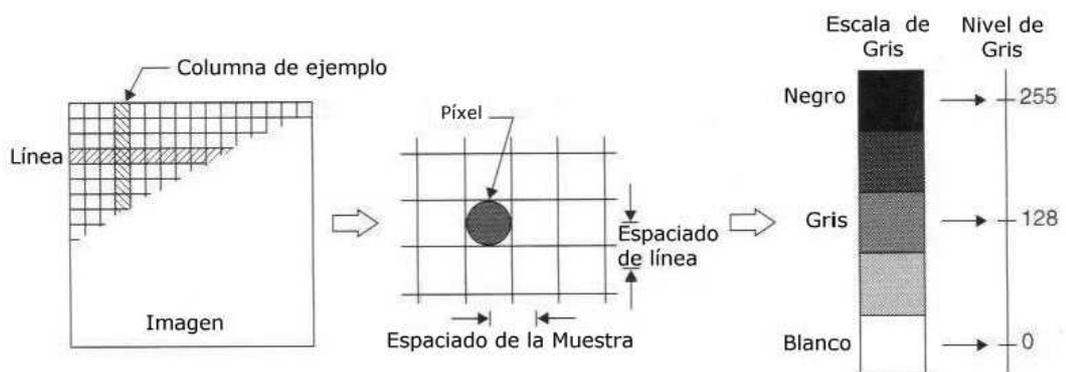
Algunas definiciones de imagen proporcionadas por Castleman en [3] son: “Una representación, similar o imitada de un objeto o cosa, ...una descripción real o gráfica, ...la introducción de algo que represente otra cosa”. Entonces en sentido general, una imagen es una representación de otra cosa. Una imagen contiene información descriptiva acerca del objeto que representa, una fotografía despliega esta información de manera que guía al observador a visualizar al objeto mismo.

Para que una imagen física, que representa una escena del mundo real, pueda ser procesada por la computadora requiere ser digitalizada. Una imagen digital es una función  $f(x, y)$ , generada por un medio óptico, que ha sido muestreada y cuantificada tanto en coordenadas espaciales como en la brillantez, muestreada como una matriz cuyos renglones y columnas están igualmente espaciados e identifican a un punto de la imagen, y el valor correspondiente al elemento de la matriz determina la información del nivel de gris en ese punto. Los ele-

mentos de tal arreglo digital son llamados elementos de imagen o píxeles [11]. Cada píxel tiene una ubicación entera o dirección (un número de renglón y un número de columna) [3]. Una imagen digital puede representar luminosidad de los objetos en una escena (tomadas por una cámara fotográfica), las características de absorción del tejido del cuerpo (imágenes de rayos-X), el perfil de temperatura de una región (imágenes infrarrojas), el campo gravitacional en una área (imágenes geofísicas). En general, cualquier función en dos dimensiones que muestre información puede ser considerada una imagen [1]. Un ejemplo de una imagen digital y del proceso de digitalización se pueden apreciar en las Figuras 2.1 y 2.2 respectivamente.



**Figura 2.1:** Imagen física y una correspondencia a imagen digital.



**Figura 2.2:** Digitalizando una imagen.

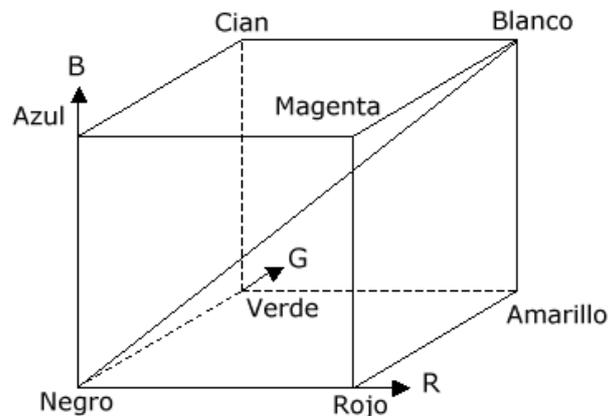
## 2.2. Definición de Espacio de Color RGB

El estudio del color es importante en el diseño y desarrollo de los sistemas de visión. El uso del color en la exhibición de imágenes no es sólo una satisfacción más, ésto también nos permite recibir más información visual. Mientras nosotros podemos percibir sólo una poca docena de niveles de gris, tenemos la habilidad para distinguir entre cientos de colores. Los atributos de color perceptuales son brillo, tonalidad, y saturación [1]. Hay varias maneras de poder especificar un color cuantitativamente, por ejemplo, en un píxel de una imagen digital a color. La manera más directa es usar los valores de brillantez del rojo, verde y azul, escalados entre, por ejemplo, de cero a uno. A esta convención se le llama formato RGB (Red, Green, Blue)<sup>1</sup>, donde cada píxel puede ser representado por un punto en el primer cuadrante del espacio llamado cubo RGB (ver Figura 2.3) [3]. El origen del espacio de color RGB representa la no brillantez para cualquiera de los colores primarios y es, por lo tanto, el color negro. El brillo completo de los tres colores primarios juntos aparece como el blanco. Cantidades iguales de los tres componentes de color con poco brillo producen una cortina de gris. La escala de grises es la diagonal que une al blanco y al negro. Tres esquinas del cubo de color corresponden a los colores primarios (rojo, verde, y azul), y las tres esquinas restantes corresponden a los colores secundarios (amarillo, cian, y magenta) [3].

En el procesamiento de imágenes, graficación por computadora y sistemas multimedia la representación RGB es a menudo la más usada. Una imagen digital a color es representada por un arreglo bidimensional de tres vectores variables los cuales constituyen los valores del rojo, verde y azul del píxel [30].

---

<sup>1</sup> Rojo, verde, azul.



**Figura 2.3:** Cubo que representa el espacio de color RGB, imagen tomada de [3].

### 2.3. Sistema HSI: Conversión a Niveles de Gris

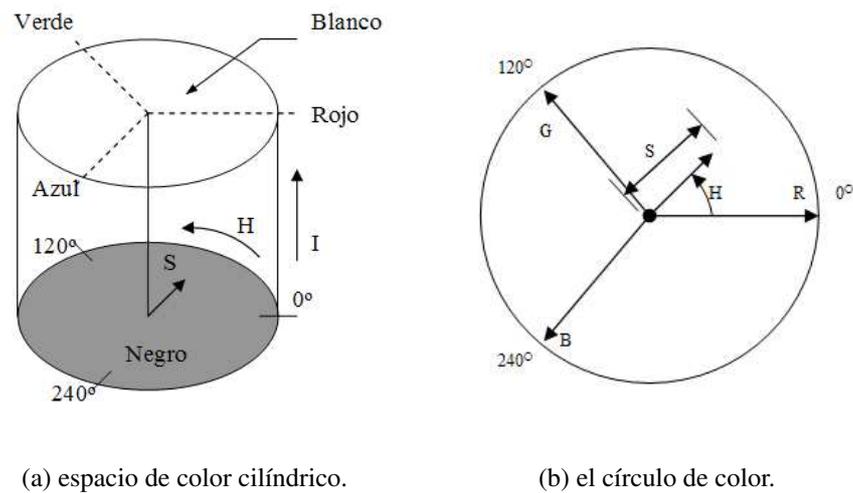
El formato HSI (*Hue-Saturation-Intensity*) es una formalización del sistema de color desarrollado por Munsell. Este diseño refleja la manera en que los humanos ven el color (debido a que el sistema de visión humana puede distinguir diferentes tonalidades fácilmente, mientras que la percepción de diferentes intensidades o saturación no implica el reconocimiento de diferentes colores), y también ofrece ventajas para el procesamiento de imágenes [3].

Existen algunas variaciones del sistema HSI, tales como HSB (*Hue-Saturation-Brightness*), HSL (*Hue-Saturation-Lightness*), y HSV (*Hue-Saturation-Value*).

El sistema HSI separa la información de color de una imagen de su información de intensidad. La información del color es representada por los valores de tonalidad (H) y saturación (S), mientras la intensidad (I), la cual describe la brillantez de una imagen, es determinada por la cantidad de la luz. La tonalidad (H) representa los colores básicos, y es determinada por la longitud de onda dominante en la distribución espectral de longitudes de onda de luz. La saturación es una medida de pureza del color, y significa la cantidad de luz blanca mezclada

con el color [4].

El espacio de color HSI puede ser representado geoméricamente según se muestra en la Figura 2.4(a). Generalmente la tonalidad (H) es considerada como un ángulo entre la línea de referencia y el punto de color en el espacio RGB. El rango de los valores de la tonalidad van desde 0° a 360°, como se aprecia en la Figura 2.4(b) el color rojo (R) es 0°, el verde (G) es 120° y el azul (B) es de 240°. El componente de saturación (S) representa la distancia radial desde el centro del cilindro. La intensidad es la altura en el eje de dirección. El eje del cilindro describe los niveles de gris, es decir, la intensidad cero (mínima) es negro, y la intensidad total (máxima) es blanco. Cada rebanada del cilindro (ó círculo de color, Figura 2.4(b)) perpendicular al eje de intensidad es un plano con la misma intensidad [4].



**Figura 2.4:** Espacio de Color HSI.

A partir del espacio de color RGB se pueden obtener las coordenadas HSI. Las fórmulas para la tonalidad (H), saturación (S), e intensidad (I) son [4]:

$$H = \arctan\left(\frac{\sqrt{3}(G - B)}{(R - G) + (R - B)}\right) \quad (2.1)$$

$$I = \frac{R + G + B}{3} \quad (2.2)$$

$$S = 1 - \frac{\min(R, G, B)}{I} \quad (2.3)$$

Una imagen a color se puede convertir en monocromática promediando los componentes RGB (ecuación (2.2)) de cada píxel, de tal modo que se desecha la información del color [3].

La intensidad de una imagen monocromática  $f$  en las coordenadas  $(x, y)$  se conoce como el nivel de gris  $l$  de la imagen en ese punto,  $l$  está dentro del rango [11]:

$$Lmin \leq l \leq Lmax \quad (2.4)$$

En teoría, el único requerimiento sobre  $Lmin$  es que sea positiva, y sobre  $Lmax$  es que sea finita. El intervalo  $[Lmin, Lmax]$  se llama escala de grises. Comúnmente en la práctica, numéricamente el intervalo se representa como  $[0, L]$ , donde  $l = 0$  se considera negro y  $l = L$  se considera como blanco en la escala. Todos los valores intermedios son sombras y variaciones continuas de gris que van del negro al blanco.

El proceso de digitalización necesita decisiones sobre los valores del número de niveles de gris permitidos para cada píxel, el cual puede calcularse mediante la siguiente fórmula:

$$G = 2^m \quad (2.5)$$

donde  $G$  indica el número de niveles de gris y  $m$  el número de bits empleados para representar a cada nivel de gris. De manera que si cada nivel de gris se representa con 8 bits, entonces el nivel de gris permitido es de 256 posibles valores, los cuales corresponden a valores enteros entre 0 y 255, donde el 0 se considera como negro y a 255 como el nivel de intensidad claro.

## 2.4. Proyección del Histograma de una Imagen

Una de las más simples y poderosas herramientas en el procesamiento de imágenes digitales es el histograma en niveles de gris. El histograma en niveles de gris es una función que muestra para cada nivel, el número de píxeles en la imagen que contienen ese nivel de gris. Y

al ser graficada esta función resume el contenido de niveles de gris en una imagen. La abscisa es el nivel de gris y la ordenada es la frecuencia de ocurrencia del número de píxeles [3]. Entonces el histograma especifica el número de píxeles que tiene cada nivel de gris, pero no sugiere donde se localizan los píxeles dentro de la imagen, una manera de conocer la posición de estos píxeles es a través de la densidad del histograma de la imagen, al aplicar una proyección del histograma a una sección de la imagen se obtiene la relación posición-densidad de nivel de gris en la imagen. La proyección vertical u horizontal ha sido usada en la extracción de características faciales, asumiendo que la región de búsqueda es un rectángulo  $H \times W$ , la proyección horizontal (ecuación (2.6)) y vertical (ecuación (2.7)) puede ser computada como [8]:

$$P(i) = \sum_{j=1}^H I(i, j) \quad \text{Donde } 0 \leq i \leq W \quad (2.6)$$

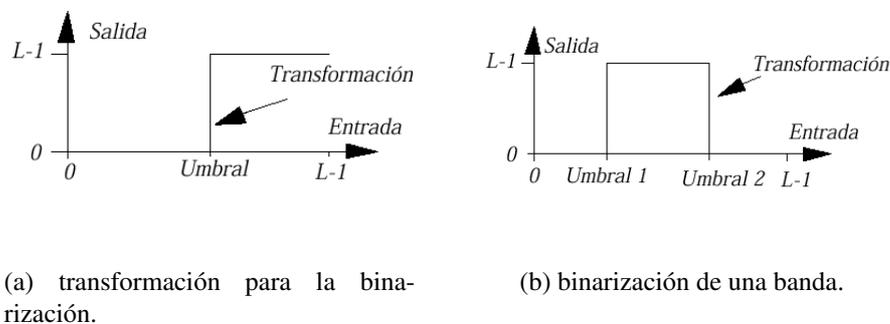
$$P(j) = \sum_{i=1}^W I(i, j) \quad \text{Donde } 0 \leq j \leq H \quad (2.7)$$

Donde  $I(i, j)$  es la función de intensidad para la ventana de búsqueda. Para el caso en que la proyección vertical u horizontal se utilice en la localización de los rasgos faciales, este proceso es equivalente a encontrar cierto mínimo y máximo local en  $P(i)$ . Este método funciona sólo cuando el rostro en la imagen tiene una vista frontal y no está obstruido [8].

## 2.5. Binarización

La binarización es una técnica que permite convertir imágenes con niveles de gris, en una imagen binaria (blanco y negro). De acuerdo a tal técnica, los valores de los píxeles en la imagen de entrada que son menores ó iguales a un cierto umbral pre-especificado, son convertidos a negro, mientras que los píxeles con valores mayores al umbral, son convertidos a blanco. En la Figura 2.5(a) se muestra la transformación que permite realizar la binarización.

En algunas ocasiones se desea realizar una binarización tal que a una banda especificada por dos umbrales, se les asigne el color blanco, mientras que los píxeles de la imagen de entrada cuyos valores están fuera de la banda especificada, se les asigne el color negro. Esta transformación se muestra en la Figura 2.5(b). Un ejemplo de binarización se muestra en la Figura 2.6(b) donde aparece el resultado obtenido al binarizar la imagen mostrada en la Figura 2.6(a), utilizando la transformación especificada en la Figura ?? con un umbral arbitrario de valor 128 [19].



**Figura 2.5:** Transformaciones utilizadas para binarizar una imagen con  $L$  valores de niveles de gris (imagen tomada de [19]).

## 2.6. Preprocesamiento de la Imagen

Cuando una imagen es adquirida por una cámara u otro sistema de formación de imágenes, frecuentemente el sistema de visión para el cual la imagen es requerida es incapaz de utilizarla directamente. La imagen podría estar corrupta por variaciones aleatorias en la intensidad (ruido), variaciones en la iluminación, o por pobre contraste [15]. Por ello, después de haber obtenido una imagen digital, el siguiente paso comúnmente consiste en aplicarle un preprocesamiento. La función principal del preprocesamiento es mejorar la imagen eliminan-



(a) imagen original.

(b) imagen binaria.

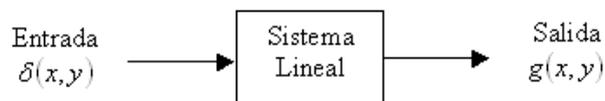
**Figura 2.6:** Ejemplo de binarización (imagen tomada de [19]).

do las características indeseables, de forma que resulte más adecuada que la original para una aplicación específica [11].

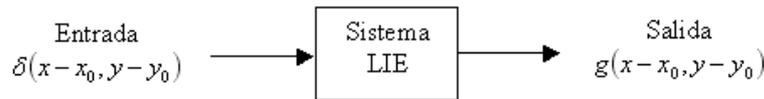
En este trabajo de tesis se utilizó como técnica de preprocesamiento un filtrado espacial, por ello enseguida se explican los conceptos relacionados con el filtro espacial lineal empleado: El filtro Gaussiano.

### 2.6.1. Convolución

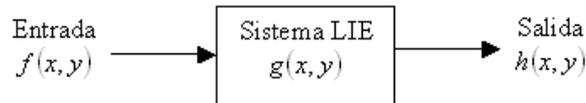
Varias operaciones del procesamiento de imágenes pueden modelarse como un sistema lineal [15] (ver Figura 2.7): Para un sistema lineal, cuando la entrada al sistema es un impulso centrado en el origen  $\delta(x, y)$ , la salida  $g(x, y)$  es la respuesta del sistema al impulso.

**Figura 2.7:** Sistema lineal.

Un sistema lineal cuya respuesta permanece igual sin importar la posición del pulso de entrada, es llamado un sistema invariante en el espacio (ver Figura 2.8): Un sistema Lineal Invariante en el Espacio (LIE) puede describirse completamente por su respuesta al impulso  $g(x, y)$  como se muestra en la Figura 2.9: Donde  $f(x, y)$  y  $h(x, y)$  son las imágenes de entrada y salida respectivamente.



**Figura 2.8:** Sistema LIE



**Figura 2.9:** Sistema LIE descrito por su respuesta al impulso  $g(x, y)$ .

Además, un sistema LIE debe satisfacer la siguiente relación:

$$a \cdot f_1(x, y) + b \cdot f_2(x, y) \Rightarrow a \cdot h_1(x, y) + b \cdot h_2(x, y) \quad (2.8)$$

donde  $f_1(x, y)$  y  $f_2(x, y)$  son imágenes de entrada,  $h_1(x, y)$  y  $h_2(x, y)$  son las imágenes de salida correspondientes a  $f_1$  y  $f_2$ , y  $a, b$  son factores constantes de escalamiento.

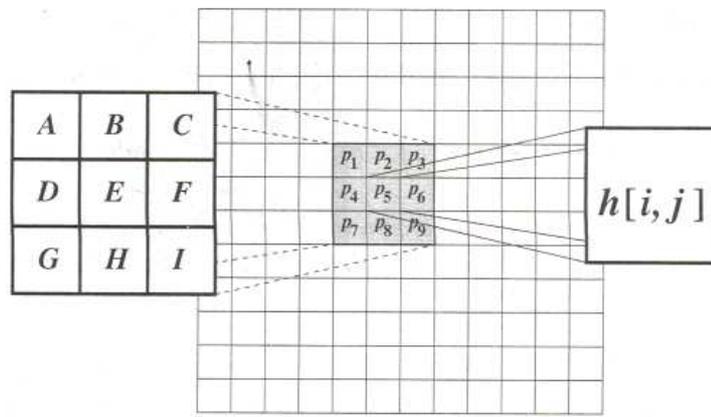
Para tal sistema, la salida  $h(x, y)$  es la convolución de  $f(x, y)$  con la respuesta al impulso  $g(x, y)$ , y se define como:

$$h(x, y) = f(x, y) * g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x', y') g(x - x', y - y') dx' dy' \quad (2.9)$$

Para el caso discreto la ecuación (2.9) se convierte en:

$$h[i, j] = f[i, j] * g[i, j] = \sum_{k=1}^n \sum_{l=1}^m f[k, l] g[i - k, j - l] \quad (2.10)$$

Si  $f$  y  $h$  son imágenes, la convolución consiste en el cálculo de sumas ponderadas de los píxeles de la imagen. La respuesta al impulso  $g[i, j]$ , es conocida como la máscara de convolución. Para cada píxel  $[i, j]$  en la imagen, el valor de  $h[i, j]$  se calcula trasladando la máscara de convolución al píxel  $[i, j]$  en la imagen, y entonces se toma la suma ponderada de los píxeles en la vecindad de  $[i, j]$ , donde los pesos individuales son los valores correspondientes a la máscara de convolución. Este proceso se ilustra en la Figura 2.10, utilizando una máscara de convolución de  $3 \times 3$ .



$$h[i, j] = A p_1 + B p_2 + C p_3 + D p_4 + E p_5 + F p_6 + G p_7 + H p_8 + I p_9$$

**Figura 2.10:** Ejemplo de una máscara de convolución de  $3 \times 3$ . El origen de la máscara de convolución corresponde a la celda E y los pesos A, B, . . . , I, son los valores de  $g[-k, -l]$ ,  $k, l = -1, 0, +1$ . Tomada de [15].

### 2.6.2. Filtro Gaussiano

Los filtros gaussianos son una clase de filtros de alisado lineal con la elección de peso de acuerdo a la forma de la función gaussiana. El filtro de alisado gaussiano es un muy buen filtro para remover ruido desde una distribución normal. La función gaussiana de media cero en una dimensión es [15]:

$$g(x) = e^{-\frac{x^2}{2\sigma^2}} \quad (2.11)$$

Donde la extensión del parámetro gaussiano  $\sigma$  determina la anchura del gaussiano. Para el procesamiento de imágenes, la función gaussiana discreta de media cero en dos dimensiones (2-D) es:

$$g[i, j] = e^{-\frac{(i^2+j^2)}{2\sigma^2}} \quad (2.12)$$

el cual es usado como una filtro de alisado.

Las funciones gaussianas tiene cinco propiedades que las hacen particularmente útiles en las primeras etapas de procesamiento de un sistema de visión. Estas propiedades indican que los filtros gaussianos de alisado son efectivos filtros paso-bajo<sup>2</sup> desde la perspectiva de ambos dominios -el espacial<sup>3</sup> y el de la frecuencia<sup>4</sup> -, son eficientes de implementar, y pueden ser usados efectivamente por ingenieros en aplicaciones prácticas de visión.

1. En dos dimensiones, las funciones gaussianas son rotacionalmente simétricas. Esto significa que la cantidad de alisado realizado por el filtro será el mismo en todas direcciones. En general, los bordes en una imagen no estarán orientados en alguna dirección en particular previamente conocida. Por lo tanto, no hay una razón a priori para realizar más alisado en una dirección que en otra. La propiedad de una simetría rotacional implica que un filtro gaussiano de alisado no predispondrá la detección subsecuente del borde en ninguna dirección particular.
2. La función gaussiana tiene un solo lóbulo. Esto significa que un filtro gaussiano alisa sustituyendo cada píxel con una carga promedio de los píxeles vecinos tal que el peso dado a un vecino disminuye monótonamente con la distancia desde el píxel central. Esta propiedad es importante debido a que un borde es una característica local en

---

<sup>2</sup> Estos filtros atenúan o eliminan las componentes de alta frecuencia en el dominio de Fourier a la vez dejan inalteradas las bajas frecuencias, es decir, dejan pasar frecuencias bajas.

<sup>3</sup> El dominio espacial se refiere al propio plano de la imagen, y las técnicas de esta categoría se basan en la manipulación directa de los píxeles de la imagen [11].

<sup>4</sup> El dominio de la frecuencia se basa en la modificación de la transformada de Fourier de una imagen [11].

una imagen, y una operación de alisado que de más realce a los píxeles mas lejanos distorsionará estas características.

3. La transformada de Fourier de un Gaussiano tiene un solo lóbulo en el espectro de la frecuencia. Las imágenes son a menudo corrompidas por señales indeseables de alta frecuencia (ruido y texturas finas). Las características deseables de la imagen, tales como bordes, tendrán componentes en ambas -baja y alta frecuencia-. El único lóbulo en la transformada de Fourier de un Gaussiano significa que la imagen alisada no será dañada por contribuciones de señales de alta frecuencia no requeridas, mientras la mayoría de las señales deseadas serán conservadas.
4. La anchura, y por lo tanto el grado de alisado de un filtro gaussiano es determinado por el parámetro  $\sigma$ , esta relación es muy simple. Un  $\sigma$  grande implica un filtro gaussiano más ancho y mayor alisado.
5. Los filtros gaussianos grandes se pueden implementar eficientemente porque las funciones gaussianas son separables. La convolución gaussiana en dos dimensiones se puede desarrollar convolucionando la imagen en una dimensión y entonces, convolucionando el resultado con el mismo filtro unidimensional orientado de modo ortogonal a la gaussiana usada en la primera etapa. Por lo que el cómputo requerido para un filtro gaussiano 2-D crece linealmente con respecto a la anchura de la máscara del filtro y no cuadráticamente.

### 2.6.2.1. Diseño de Filtros Gaussianos

Un enfoque para el diseño de filtros gaussianos consiste en calcular la máscara de pesos directamente a partir de la distribución gaussiana discreta [15]:

$$g[i, j] = ce^{-\frac{(i^2+j^2)}{2\sigma^2}} \quad (2.13)$$

Donde  $c$  es una constante de normalización, por lo que la ecuación anterior se puede reescribir como:

$$\frac{g[i, j]}{c} = e^{-\frac{(i^2+j^2)}{2\sigma^2}} \quad (2.14)$$

Al elegir un valor para  $\sigma^2$ , se puede evaluar sobre una ventana de  $n \times n$  para obtener un núcleo, o máscara, para el cual el valor en  $[0,0]$  es igual a 1. Por ejemplo al escoger  $\sigma^2 = 2$  y  $n = 7$ , la expresión antes mencionada da el siguiente arreglo:

$[i, j]$	-3	-2	-1	0	1	2	3
-3	.011	.039	.082	.105	.082	.039	.011
-2	.039	.135	.287	.368	.287	.135	.039
-1	.082	.287	.606	.779	.606	.287	.082
0	.105	.368	.779	1.000	.779	.368	.105
1	.082	.287	.606	.779	.606	.287	.082
2	.039	.135	.287	.368	.287	.135	.039
3	.011	.039	.082	.105	.082	.039	.011

Sin embargo, se desea que los pesos del filtro sean valores enteros por facilidad de cómputo. Por lo tanto, se toma el valor de una de las esquinas del arreglo, y se escoge una  $k$  tal que este valor llegue a ser 1. Considerando el ejemplo anterior, se tiene:

$$\frac{g[3, 3]}{k} = e^{-\frac{(3^2+3^2)}{2(2)^2}} = 0.011 \Rightarrow k = \frac{g[3, 3]}{0.011} = \frac{1.0}{0.011} = 91 \quad (2.15)$$

Ahora, multiplicando el resto de los pesos por  $k$ , se obtiene el siguiente arreglo:

$[i, j]$	-3	-2	-1	0	1	2	3
-3	1	4	7	10	7	4	1
-2	4	12	26	33	26	12	4
-1	7	26	55	71	55	26	7
0	10	33	71	91	71	33	10
1	7	26	55	71	55	26	7
2	4	12	26	33	26	12	4
3	1	4	7	10	7	4	1

Este es el resultado de la máscara de convolucion para el filtro Gaussiano. Sin embargo, los

pesos de la máscara no suman 1. Por lo tanto, cuando ejecutamos la convolucion, los valores de los píxeles de salida deberían ser normalizados por la suma de los pesos de la máscara para asegurarse que las regiones de intensidad uniforme no sean afectadas. Para el ejemplo antes mencionado:

$$\sum_{i=-3}^3 \sum_{j=-3}^3 g[i, j] = 1115 \quad (2.16)$$

Por lo tanto,

$$h[i, j] = \frac{1}{1115}(f[i, j] * g[i, j]) \quad (2.17)$$

Donde los pesos de  $g[i, j]$  son todos valores enteros.

## 2.7. Realce de Bordes

Los bordes son relevantes para estimar la estructura y propiedades de los objetos en una escena. Los bordes son cambios locales significativos en las intensidades de la imagen y son características importantes para el análisis de imágenes, éstos aparecen típicamente sobre el límite entre dos regiones diferentes en una imagen. La detección de bordes es frecuentemente el primer paso en la recuperación de información de imágenes y debido a su importancia, la detección de bordes es todavía una área activa de investigación [15].

Un borde en una imagen es un cambio local significativo en la intensidad de la imagen, usualmente asociado con una discontinuidad ya sea en la intensidad de la imagen o en la primera derivada de la intensidad de la imagen.

### 2.7.1. Definiciones Básicas

- a. **Un punto del borde** es un punto en una imagen con coordenadas  $[i, j]$  posicionado en un cambio local significativo de intensidad en la imagen.

- b. **Un fragmento del borde** coincide con las coordenadas  $i$  y  $j$  de un borde y con su orientación  $\theta$ , la cual podría ser el ángulo del gradiente.
- c. **Un detector de borde** es un algoritmo que produce un conjunto de bordes (puntos o fragmentos de borde) a partir de una imagen.
- d. **Un contorno** es una lista de bordes o la curva matemática que modela la lista de bordes.
- e. **Ligar bordes** es el proceso de formar una lista ordenada de bordes a partir de una lista desordenada.
- f. **Seguimiento de bordes** es el proceso de buscar en la imagen (filtrada) para determinar los contornos.

El término borde es usado indistintamente para referirse a los puntos o fragmentos de borde.

El conjunto de bordes producido por un detector de bordes puede ser dividido en dos subconjuntos: Bordes correctos, los cuales corresponden a bordes en la escena, y falsos bordes, los cuales no corresponden a bordes en una escena. Un tercer conjunto de bordes puede definirse como aquellos bordes en la escena que podrían haber sido detectados. Este es el conjunto de bordes faltantes. Los falsos bordes son llamados falsos positivos, y los bordes faltantes son llamados falsos negativos.

### 2.7.2. Gradiente

La detección de bordes es en esencia la operación de detectar cambios locales significativos en una imagen. El gradiente es una medida de cambio en una función, y una imagen puede ser considerada como un arreglo de muestras de alguna función continua de intensidad de la imagen. Por analogía, los cambios significativos en los valores de gris en una imagen pueden detectarse usando una aproximación discreta del gradiente. El gradiente es el equivalente bidimensional de la primera derivada y se define como el vector [15, 11]:

$$G[f(x, y)] = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (2.18)$$

Hay dos propiedades importantes asociadas con el gradiente:

1. El vector  $G[f(x, y)]$  apunta en la dirección de la máxima tasa de incremento de la función  $f(x, y)$ , es decir, indica la dirección de la máxima variación de  $f$  en  $(x, y)$ .
2. La magnitud del gradiente, dada por:

$$\nabla f(x, y) = \sqrt{G_x^2 + G_y^2} \quad (2.19)$$

igual a la máxima tasa de incremento de  $f(x, y)$  por unidad de distancia en la dirección de  $G$ . Sin embargo, es común aproximar la magnitud del gradiente por los valores absolutos:

$$\nabla f(x, y) \approx |G_x| + |G_y| \quad (2.20)$$

ó

$$\nabla f(x, y) \approx \max(|G_x| + |G_y|) \quad (2.21)$$

La magnitud (o módulo) de  $G$  generalmente es referenciada, para simplificar, como gradiente. A partir del análisis de vectores, la dirección del vector gradiente se define como:

$$\alpha(x, y) = \tan^{-1} \left( \frac{G_y}{G_x} \right) \quad (2.22)$$

Donde el ángulo  $\alpha$  es medido con respecto al eje  $x$ .

Para imágenes digitales, las derivadas de la ecuación (2.18) se aproximan mediante las siguientes máscaras (aunque un método alternativo es considerar una vecindad de 3 x 3):

$$G_x = \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix} \quad G_y = \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}$$

### 2.7.3. Pasos en la Detección de Bordes

Los algoritmos de detección de bordes contienen tres pasos [15]:

1. **Filtrado.** Puesto que el cálculo del gradiente basado en los valores de intensidad de sólo dos puntos son susceptibles a ruido y otros caprichos del computo discreto, el filtrado es comúnmente usado para mejorar el funcionamiento de un detector de bordes con respecto del ruido. Sin embargo, hay una descompensación entre la calidad del borde y la reducción de ruido. Un mayor filtrado para reducir el ruido resulta en pérdida de la calidad del borde.
2. **Realce.** Para facilitar la detección de bordes, es esencial determinar cambios en la intensidad en la vecindad de un punto. El realce enfatiza aquellos píxeles donde hay un cambio significativo en el valor de la intensidad local y es usualmente realizado calculando la magnitud del gradiente.
3. **Detección.** Sólo se requieren puntos con un fuerte contenido de borde. Sin embargo, muchos puntos en una imagen tienen un valor distinto de cero para el gradiente, y no todos estos puntos son bordes para una aplicación en particular. Por lo que se requiere un método que determine cuáles puntos son puntos de borde (frecuentemente se utiliza un umbral como criterio de detección).

Existe una amplia variedad de métodos para detectar bordes que han sido desarrollados. A continuación se describirá solo uno de ellos, el cual se utilizó en este proyecto.

#### 2.7.3.1. Operador de Sobel

El operador de Sobel es un operador de extracción de bordes que usa una vecindad de  $3 \times 3$  para el cálculo del gradiente. El operador de Sobel es la magnitud del gradiente computado

de acuerdo a [15]:

$$M = \sqrt{S_x^2 + S_y^2} \quad (2.23)$$

Donde las derivadas parciales  $S_x, S_y$  son definidas por:

$$S_x = (a_2 + ca_3 + a_4) - (a_0 + ca_7 + a_6) \quad (2.24)$$

$$S_y = (a_0 + ca_1 + a_2) - (a_6 + ca_5 + a_4) \quad (2.25)$$

Donde la constante  $c = 2$ , y el etiquetado de los píxeles vecinos al píxel  $[i, j]$  es:

a0	a1	a2
a7	$[i, j]$	a3
a6	a5	a4

$S_x, S_y$  pueden implementarse usando las siguientes máscaras de convolución:

$$S_x = \begin{array}{|c|c|c|} \hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array} \quad S_y = \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 0 & 0 & 0 \\ \hline -1 & -2 & -1 \\ \hline \end{array}$$

Este operador enfatiza los píxeles que están más cercanos al centro de la máscara.

## 2.8. Detección de Color de Piel

La detección de la piel es un paso importante en muchos sistemas de visión tales como sistemas de interpretación de gestos, seguimiento de la cabeza o de la mano, detección del rostro, entre otros. La detección de píxeles basados en el color de la piel puede dirigir el espacio de búsqueda con conocimiento a priori. Sin embargo, ésta no es una tarea fácil. Los valores del color de piel pueden variar con la luz del ambiente, por ejemplo, las lámparas de

colores actúan como filtros, también afectan las sombras, la luz del día, entre otros factores. Aunado a esto hay que considerar que diferentes cámaras retornan diferentes valores para la misma escena [9]. La meta final de la detección de piel es construir una regla de decisión que discrimine entre los píxeles de color de piel y los que no lo son. Normalmente este objetivo se alcanza mediante una métrica, la cual calcula la distancia de un píxel de color al tono de piel. El tipo de esta métrica se define por un método de modelado de color de piel. Debido a que la detección de color de piel tiene una larga tradición en el área de visión por computadora, existe una amplia variedad de métodos de modelado de color de piel, los cuales de manera general se clasifican en métodos de detección de piel basados en el píxel y métodos basados en la región. Los primeros clasifican individualmente a cada píxel como de color de piel o no, independientemente de sus vecinos. En contraste, los métodos basados en región tratan de considerar el arreglo espacial de los píxeles de color de piel durante la etapa de detección, con el propósito de mejorar el desempeño de tales métodos [31].

En esta sección se abundará sobre dos métodos basados en el píxel que definen explícitamente una región de color de piel (otros métodos de modelado de color de piel son especificados en [31]), ya que uno de ellos fue empleado en este trabajo de tesis, como se detallará posteriormente. Tales métodos construyen un clasificador de piel definiendo explícitamente, a través de un número de reglas, los límites cercanos de una región de color de piel en algún espacio de color.

El primer método al que se hace referencia fue descrito en [23, 31] y clasifica un píxel como piel en el espacio de color RGB si:

$$R > 95 \text{ y } G > 40 \text{ y } B > 20 \text{ y } \max\{R, G, B\} - \min\{R, G, B\} > 15 \text{ y } |R - G| > 15 \text{ y } R > G \text{ y } R > B \quad (2.26)$$

La ventaja obvia de este método es la simplicidad de las reglas de detección de piel que conducen a la construcción de un clasificador muy rápido [31]. Ejemplos de los resultados de

la detección de piel al aplicar la ecuación (2.26) se muestran en la Figuras 2.11(a) y 2.11(b). En la Figura 2.11(a) se observa una adecuada detección de color de piel; sin embargo en la Figura 2.11(b) aparecen ciertos falsos positivos en el área del pelo y ropa debido a la tonalidad del pelo y la iluminación.

La principal dificultad que presenta un método que define explícitamente una región de color de piel, consiste en la necesidad de encontrar tanto un buen espacio de color así como un conjunto adecuado de reglas de decisión empíricas, para poder alcanzar altos rangos de reconocimiento [31]. Desafortunadamente, como mencionan Gómez, Sánchez y Sucar en [9] no se tienen reportes sobre una selección apropiada de componentes de color para la detección de color de piel. Además, de que el trabajo previo en esta área evalúa un único espacio de color; por lo que en [9] presentan un paradigma de selección de atributos, que les permite determinar la mezcla de componentes de color que discrimina muy bien el color de piel en escenas al interior de un espacio cerrado y al exterior de éste. Éste es el segundo método al que se hace referencia. En su trabajo Gómez, Sánchez y Sucar realizaron una evaluación de cada componente de entre varios modelos de color, y a partir de éstos seleccionaron un modelo de color adecuado para la detección de piel. El espacio de color que definieron tiene tres ejes: E del espacio de color YES<sup>5</sup> (especificado por la corporación Xerox), el radio rojo/verde (R/G), y H del espacio de color HSV. Y debido a la convexidad de este espacio de color presentan un conjunto de reglas de decisión simples para la detección de piel. Como resultado su conjunto de reglas permite reconocer el 96% de puntos de color de piel, con 11% de falsos positivos. Algunos resultados tomados de [9] aparecen en las Figuras 2.12(a) y 2.12(b). Puede notarse que en 2.12(a) hay una adecuada detección de color de piel, sin embargo en 2.12(b) aparecen falsos positivos relacionados con la tonalidad del color del pelo, con tonalidad de la ropa, y con la iluminación del ambiente.

---

<sup>5</sup> Mayores detalles en [http://www.colour.org/tc8-03/survey/s\\_94h.html](http://www.colour.org/tc8-03/survey/s_94h.html) (último acceso: 11 de agosto de 2007; última modificación: 17 de agosto de 1999).

En este proyecto se optó por aplicar el primer método (ecuación (2.26)) dada su simplicidad y velocidad de clasificación, además de que al comparar los resultados de éste con los reportados por [11], su desempeño cualitativamente es similar en cuanto a la generación de falsos positivos.



(a) adecuada detección de color de piel.

(b) falsos positivos relacionados con la tonalidad del color del pelo, del color de la ropa e iluminación.

**Figura 2.11:** Detección de piel en el espacio de color RGB.



(a) adecuada detección de color de piel.

(b) falsos positivos relacionados con la tonalidad del color del pelo, ropa e iluminación del ambiente.

**Figura 2.12:** Detección de piel definida por [9].

## 2.9. Contorno Activo ó Snake

Existe una amplia gama de métodos que han sido utilizados para extraer el contorno del rostro y de la cabeza en imágenes, y entre ellos los que utilizan modelos de contornos activos ó *snakes* reportan buenos resultados [12, 22, 25, 21].

Los contornos activos son modelos de forma activa, al igual que los modelos deformables y los contornos dinámicos. Los contornos activos imponen cierto conocimiento a priori desde el inicio en lugar de esperar a que ciertas propiedades deseables, tales como la continuidad y suavidad, emerjan de los datos de la imagen. Esto es, imponen y ajustan un modelo elástico de curva continua y flexible a la imagen, y al variar los parámetros de elasticidad consiguen controlar la influencia de las suposiciones a priori [2].

Los contornos activos son un ejemplo de una técnica general para adaptar un modelo deformable a una imagen por medio de minimización de energía. A partir de cualquier punto de inicio, el contorno activo se deforma siguiendo el contorno sobresaliente más cercano [18].

El trabajo original desarrollado por Kass, Witkin y Terzopoulos [18] define a el modelo básico de contorno activo como: "Un spline (curva definida a trozos mediante polinomios) de continuidad controlada bajo la influencia de las fuerzas de la imagen y fuerzas de restricción externa". Las fuerzas de la imagen atraen al contorno activo hacia las características sobresalientes de la imagen como líneas, bordes y contornos subjetivos. Las fuerzas de restricción externa son responsables de colocar al contorno activo cerca del mínimo local deseado.

El contorno activo se define paramétricamente como un conjunto de puntos  $v(s) = (x(s), y(s))$  llamados *snaxels*, donde  $x(s)$ ,  $y(s)$  son las coordenadas  $x$ ,  $y$  a lo largo del contorno y  $s \in [0, 1]$ .

Una vez colocado en el espacio de la imagen, el contorno activo se deforma para encontrar el contorno deseado en su vecindad bajo la influencia de fuerzas internas y externas. El contorno deseado corresponde a la mínima energía del contorno activo. Por lo tanto, la deformación del contorno activo en el dominio de la imagen se especifica a través de una función

de energía, la cual se minimiza por medio de un proceso específico. La función de energía es una combinación ponderada de fuerzas internas y externas, definida como [18]:

$$E_{global}(v(s)) = \int_0^1 E_{interna}(v(s)) + E_{imagen}(v(s)) + E_{externa}(v(s)) ds \quad (2.27)$$

Donde:

$$E_{interna}(v(s)) = \frac{1}{2} \left( \alpha(s) \left| \frac{\partial v}{\partial s}(s) \right|^2 + \beta(s) \left| \frac{\partial^2 v}{\partial s^2}(s) \right|^2 \right) \quad (2.28)$$

- La  $E_{interna}$  es la parte que depende de propiedades intrínsecas del contorno activo, tales como su longitud o curvatura. Se compone de un término de primer orden y uno de segundo orden. El primer término que es controlado por  $\alpha(s)$ , ajusta la elasticidad del contorno activo y hace que se comporte como una membrana. Su propósito es conseguir que los puntos sean equidistantes. El término de segundo orden, el cual es controlado por  $\beta(s)$ , ajusta la rigidez del contorno activo, este término trata de suavizar la curva del contorno activo [21].

$$E_{imagen}(v(s)) = \gamma_{línea} E_{línea} + \gamma_{borde} E_{borde} + \gamma_{terminación} E_{terminación} \quad (2.29)$$

- La  $E_{imagen}$  permite al contorno activo ser atraído por las características sobresalientes de la imagen. La energía total de la imagen se puede expresar como una combinación ponderada de las tres energías funcionales dadas en (2.29).
  - La forma más simple y útil de expresar  $E_{línea}$  es como la intensidad de la imagen:

$$E_{línea} = I(x, y) \quad (2.30)$$

de modo que dependiendo del signo de  $\gamma_{línea}$ , el contorno activo será atraído por líneas claras u oscuras (el contorno activo tratará de alinearse al contorno cercano más claro o más oscuro).

- La energía funcional  $E_{borde}$  ayuda a localizar los bordes. El contorno activo será atraído a los contornos con gradientes de la imagen altos si se considera  $E_{borde}$  como:

$$E_{borde} = -|\nabla I(x, y)|^2 \quad (2.31)$$

Donde  $\nabla I$  es el gradiente de la imagen  $I(x, y)$  y se obtiene de acuerdo a la ecuación (2.19).

- La energía funcional  $E_{terminación}$  permite localizar las terminaciones de los segmentos de líneas y esquinas, para ello utiliza la curvatura de las líneas horizontales de la imagen ligeramente suavizada.
  - La energía externa  $E_{externa}$  depende de factores tales como la estructura de la imagen y restricciones particulares que el usuario impone (restricciones externas).

Un punto fuerte de los contornos activos es su **flexibilidad**: Es posible integrar al comportamiento del contorno activo cualesquiera de las características o propiedades de la imagen que se desean extraer, y que puedan expresarse como términos de energía en la función de energía del contorno activo.

### 2.9.1. Fuerza de Inflación o *Balloon Force*

La fuerza de inflación fue introducida por primera vez por L. D. Cohen en [5, 6]. Esta fuerza permite al contorno activo adaptarse a las propiedades de expansión y contracción de un globo. La dirección de expansión o contracción de la fuerza de inflación depende de dos aspectos [17]:

1. El vector tangente del contorno inicial.
2. La normal unitaria al vector tangente.

La dirección del vector tangente depende de la dirección de los puntos del contorno, los cuales primero son colocados sobre la imagen para formar el contorno inicial. La expresión que calcula el vector tangente es [17]:

$$\vec{t}_i = \frac{(\vec{v}_i - \vec{v}_{i-1})}{\|\vec{v}_i - \vec{v}_{i-1}\|} + \frac{(\vec{v}_{i+1} - \vec{v}_i)}{\|\vec{v}_{i+1} - \vec{v}_i\|} \quad (2.32)$$

Donde  $\vec{t}_i$  es el vector tangente al punto del contorno  $i$ . La normal unitaria ( $\vec{n}_i$ ) para cada punto individual del contorno se puede calcular fácilmente rotando el vector tangente en dicho punto por  $90^\circ$ . Por lo tanto, la expresión para la fuerza de inflación es [17]:

$$E_{balloon} = \lambda \vec{n}_i \quad (2.33)$$

Donde  $\lambda$  representa el peso de la fuerza de inflación (*balloon force*) similar a  $\alpha$ ,  $\beta$  y  $\gamma_{borde}$ , las cuales son las constantes arbitrarias para la fuerza de elasticidad, curvatura y del gradiente de la imagen respectivamente.

El contorno activo requiere de una posición inicial en el espacio de la imagen para encontrar el borde deseado, como ya se ha mencionado, lo cual tiene como consecuencias sobre la evolución de la curva del contorno activo que [5, 6]:

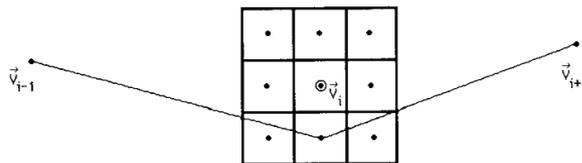
- Si la curva no está lo suficiente cercana a un borde, ésta no es atraída hacia él.
- Si la curva no está sujeta a cualquier fuerza de contrapeso, ésta tenderá a contraerse sobre sí misma.

Por consiguiente, la fuerza de inflación se añade a las energías  $E_{interna}$  y  $E_{imagen}$  para empujar a la curva hacia fuera, como si introdujéramos aire dentro (la curva se comporta como un globo que se está inflando). La curva se expande y es atraída a los bordes como antes, pero si el borde es muy pequeño o muy débil con respecto a la fuerza de inflación, la curva pasa sobre el borde, creciendo hacia fuera. La fuerza de inflación evita que la curva sea “atrapada” por supuestos puntos del borde aislados, y hace al resultado final menos susceptible a las condiciones iniciales.

### 2.9.2. Algoritmo de Minimización de Energía

Para llevar a cabo la minimización de la función de energía del contorno activo, denotada por  $E_{global}$  (ecuación (2.27)) se experimentó con el algoritmo voraz (*greedy*), el cual es un método iterativo que ofrece una solución rápida. A continuación se da una explicación de su funcionamiento.

Este algoritmo fue propuesto por Williams y Shah en [32], y realiza la minimización de  $E_{global}$  de manera iterativa, durante cada iteración se examina cada punto (*snaxel*) y su vecindad del siguiente modo: Se calcula la función de energía para la ubicación actual (*snaxel* actual)  $\vec{v}_i$  y para cada uno de sus ocho vecinos. La ubicación que tenga el valor más pequeño se elige como la nueva posición de  $\vec{v}_i$ . Este proceso se ilustra en la Figura 2.13 y el funcionamiento del algoritmo de voraz puede observarse, a través de su pseudocódigo, en la Figura 2.14.



**Figura 2.13:** La función de energía se calcula en  $\vec{v}_i$  y para cada uno de sus ocho vecinos. La ubicación que tenga el valor más pequeño se elige como la nueva posición de  $\vec{v}_i$ . El punto anterior y el posterior a  $\vec{v}_i$  se usan para calcular las restricciones de continuidad. (Imagen tomada de [32]).

Los términos  $E_{cont}$  y  $E_{curv}$  a que se hace referencia en la Figura 2.14 son el primero y segundo término de la  $E_{interna}$  definida en la ecuación (2.28), los cuales son aproximados por diferencias finitas [18, 32]:

- Para determinar una aproximación adecuada al primer término de la ecuación (2.28) (el término que representa la continuidad) éste se calcula de acuerdo a la ecuación (2.34), el algoritmo utiliza la diferencia entre la distancia promedio entre los *snaxels* ( $\bar{d}$ ), y la distancia entre los dos puntos en consideración. Por lo tanto, los puntos que tengan

```

Inicializar  $\alpha_i$ ,  $\beta_i$  y  $\gamma_i$  a 1 para todos los  $i$ 's.
hacer
  /*Ciclo para mover los puntos a una nueva posición*/
  for  $i=0$  to  $n$  /*el punto 0 es el primero y el último en ser procesado*/
     $E_{\min} = \text{Big}$ 
    for  $j=0$  to  $m-1$  /* $m$  es el tamaño de la vecindad*/
       $E_j = \alpha_i E_{\text{cont},j} + \beta_i E_{\text{curv},j} + \gamma_i E_{\text{imagen},j}$ 
      if  $E_j < E_{\min}$  then
         $E_{\min} = E_j$ 
         $j_{\min} = j$ 

    Mover el punto  $\vec{v}_i$  a la posición  $j_{\min}$ 
    if  $j_{\min}$  no es la posición actual,  $\text{ptsmoved} += 1$  /*contar los puntos que se movieron*/
  /*Proceso para determinar donde permitir esquinas en la siguiente iteración*/
  for  $i=0$  to  $n-1$ 
     $c_i = |\vec{u}_i / |\vec{u}_i| - \vec{u}_{i+1} / |\vec{u}_{i+1}||^2$ 

    for  $i=0$  to  $n-1$ 
      if  $c_i > c_{i-1}$  y  $c_i > c_{i+1}$  /*si la curvatura en el punto  $i$  es mayor que sus vecinos*/
      y  $c_i > \text{umbral1}$  /*y la curvatura es mayor a un umbral*/
      y  $\text{mag}(\vec{v}_i) > \text{umbral2}$  /*y la fuerza del borde está por encima de otro umbral*/
      then  $\beta_i = 0$  /* entonces se relaja la curvatura en el punto  $i$ */

hasta  $\text{ptsmoved} < \text{umbral3}$ 

```

**Figura 2.14:** Pseudocódigo para el algoritmo voraz. (Imagen tomada de [32]).

una distancia cercana al promedio tendrán el valor mínimo. El valor se normaliza dividiéndolo por el valor mayor en la vecindad en la cual el punto podría moverse, dándole un valor en el rango de  $[0, 1]$ . Al final de cada iteración se calcula una nueva distancia promedio.

$$\left| \frac{\partial v}{\partial s}(s) \right|^2 \approx \bar{d} - |\vec{v}_i - \vec{v}_{i-1}| \quad (2.34)$$

- El segundo término de la ecuación (2.28) (que representa a la curvatura) se calcula de acuerdo a la ecuación (2.35):

$$\left| \frac{\partial^2 v}{\partial s^2}(s) \right|^2 \approx |\vec{v}_{i-1} - 2\vec{v}_i + \vec{v}_{i+1}|^2 = (x_{i-1} - 2x_i + x_{i+1})^2 + (y_{i-1} - 2y_i + y_{i+1})^2 \quad (2.35)$$

El término  $E_{\text{imagen}}$  en la Figura 2.14 se refiere a la fuerza de la imagen, la cual es la magnitud del gradiente y se evalúa como lo indica la ecuación (2.36):

$$E_{\text{imagen}}(v(s)) = (\text{min} - \text{mag}) / (\text{max} - \text{min}) \quad (2.36)$$

donde  $mag$  es la magnitud del gradiente en un punto,  $max$  el gradiente máximo y  $min$  el gradiente mínimo en la ocho vecindad.

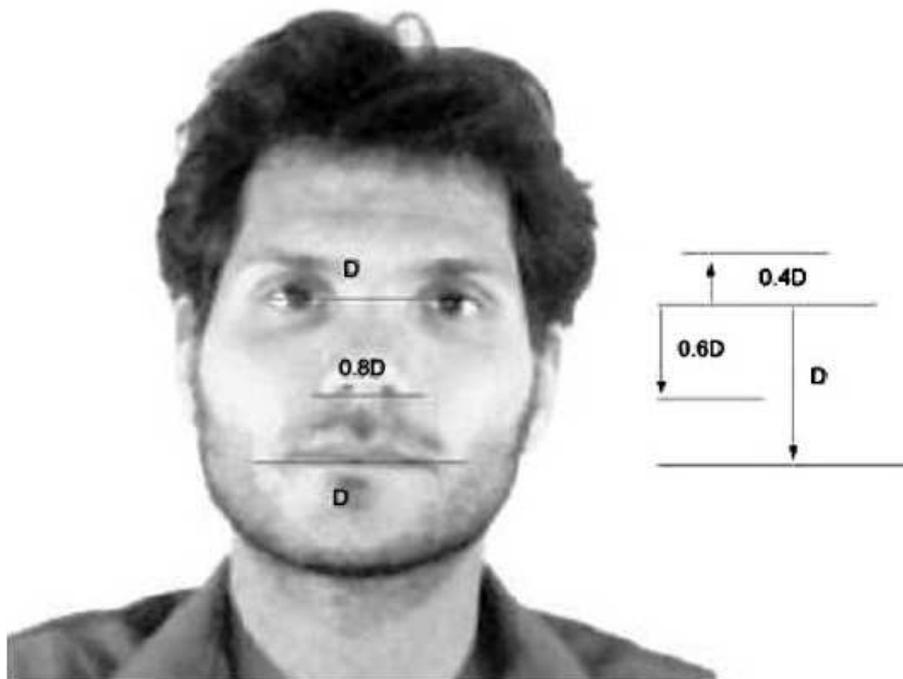
Al final de cada iteración se incluye un paso que determina la curvatura  $c_i$  en cada punto  $i$  sobre el nuevo contorno. La magnitud de la curvatura  $c_i$  se calcula con la ecuación que aparece en el pseudocódigo de la Figura 2.14. Los términos  $\vec{u}_i$  y  $\vec{u}_{i+1}$  en dicha ecuación son los vectores definidos por  $(x_i - x_{i-1}, y_i - y_{i-1})$  y  $(x_{i+1} - x_i, y_{i+1} - y_i)$  respectivamente. Si el valor de  $c_i$  es una curvatura máxima sobre cierto umbral de curvatura y sobre un umbral de gradiente, se establece la constante  $\beta_i = 0$  para la siguiente iteración, de esta manera se formará una esquina y se reducirá la curvatura en los segmentos entre estos puntos.

## 2.10. Modelo Geométrico del Rostro

Se experimentó en este proyecto de tesis con el modelo geométrico del rostro planteado en [28], en el cual primero se obtiene la distancia entre el centro de los ojos, la cual se denota por  $D$ , y a partir de ella se crea el modelo geométrico para localizar la nariz y boca considerando las siguientes relaciones:

1. La distancia vertical entre los dos ojos y el centro de la boca es  $D$ .
2. La distancia vertical entre los dos ojos y el centro de la nariz es  $0.6D$ .
3. El ancho de la boca es  $D$ .
4. El ancho de la nariz es  $0.8D$ .
5. La distancia vertical entre ojos y cejas es  $0.4 D$ .

Un ejemplo de estas distancias se ilustra en la Figura 2.15



**Figura 2.15:** Modelo geométrico del rostro (tomado de [28]).

# Capítulo 3

## Desarrollo del Sistema

Para alcanzar el objetivo planteado en este trabajo de tesis (sección 1.1), que consiste en determinar la ubicación del contorno del rostro y de las características faciales: Ojos, nariz y boca, se diseñó e implementó un sistema que consta de dos etapas. En la primer etapa se localizan los rasgos faciales haciendo uso de un modelo geométrico del rostro y de la proyección vertical y horizontal del histograma. Y en la segunda etapa se localiza automáticamente el rostro de la persona mediante un contorno activo. El diseño del sistema aparece en la Figura 3.1 compuesto por dos etapas, éstas se muestran a detalle en las Figuras 3.2 y 3.3 respectivamente, y en ellas se observan las técnicas de procesamiento digital de imágenes que se aplicaron y cómo se relacionan entre sí, los círculos en las figuras muestran las diferentes salidas producidas por cada proceso y las cuales funcionan como entrada para otros. Puede notarse que los resultados de la primera etapa son utilizados en la segunda. En seguida se detalla como fue implementado este diseño, primero se especifican los alcances de la implementación y posteriormente cada etapa.

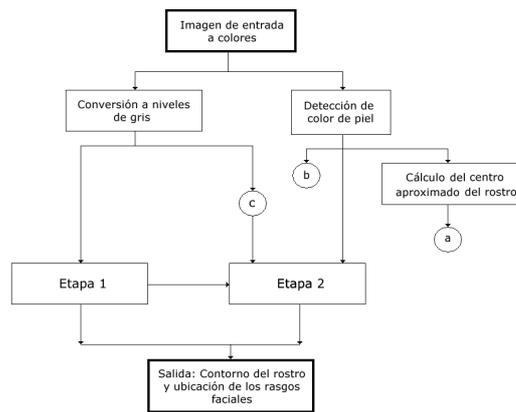


Figura 3.1: Diseño del sistema, formado por dos etapas.

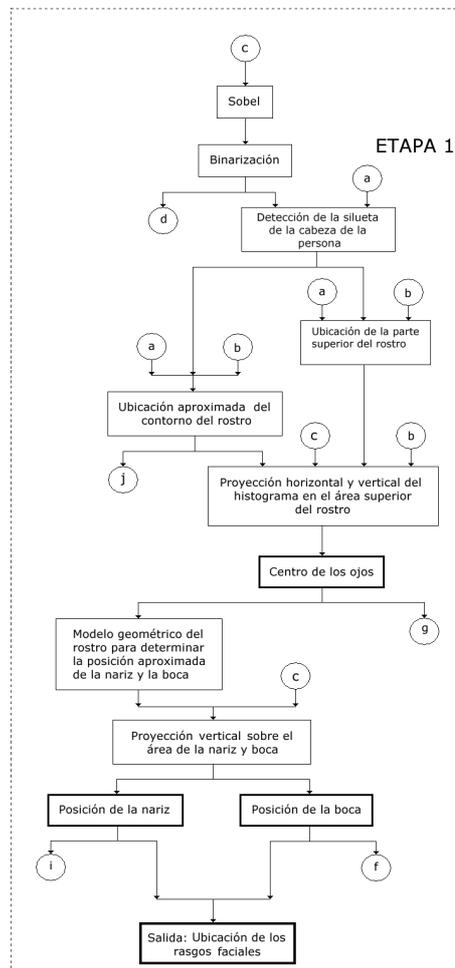
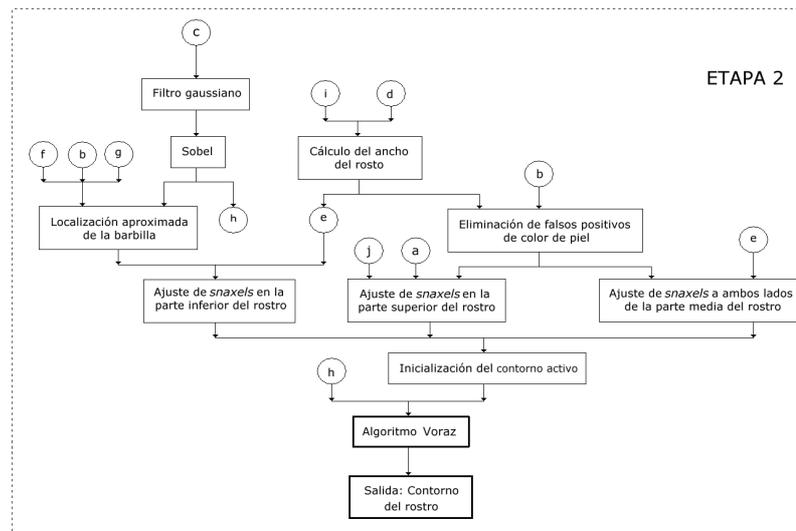


Figura 3.2: Etapa 1: Localización de las características faciales.



**Figura 3.3:** Etapa2: Localización del contorno del rostro.

### 3.1. Alcances de la Implementación

El sistema analiza imágenes de individuos con las siguientes características:

1. Personas con piel de tonalidad clara.
2. Que no usen accesorios en el área del pelo como gorras, pasadores, peinetas, diademas, entre otros.
3. Que no tengan un color de cabello o de cejas confundible con el color de la piel, como podría ser el caso de pelirrojos.
4. Que vestan ropa de un color que contraste con el color de la piel.

Por otro lado, las imágenes de entrada son imágenes a colores en formato BMP<sup>1</sup> por su facilidad de lectura y separación en sus colores primarios, tienen una resolución de 640x480 y las siguientes propiedades:

<sup>1</sup> BitMaP o Mapa de Bits

1. Contienen un fondo uniforme con color que contrasta con el color de piel.
2. Presentan a un único individuo en la escena con una pose frontal y con ausencia de expresiones faciales marcadas, sin presentar oclusiones en los rasgos faciales, con los ojos abiertos y sin lentes.
3. En algunas imágenes aparecen componentes estructurales como barba y bigote.
4. La cabeza del individuo presente en la escena debe estar completamente contenida en la imagen.
5. No presentan rotación sobre el eje de la cámara y poseen poca presencia de ruido generado por ésta.
6. Se consideraron imágenes tomadas bajo tres tipos diferentes de iluminación: Iluminación ambiental (o luz natural), bajo una fuente de luz intensa desde un ángulo de 45 grados, y bajo una fuente de luz semi-intensa casi frontal. Estos tipos de iluminación dependieron de las imágenes disponibles como se explicará en el capítulo 4. Sólo se consideraron las imágenes capturadas bajo estas condiciones de iluminación donde el color de piel no sea confundible con ciertos objetos como la ropa, pelo y accesorios.

Un ejemplo de las imágenes de entrada se muestra en la Figura 3.4.

De acuerdo a las secciones 2.1 y 2.2, la imagen de entrada es almacenada en un arreglo bidimensional con tres vectores variables para los componentes del espacio de color RGB (rojo, verde, azul), para realizar esto se implementó una librería que captura y almacena la imagen de entrada usando la estructura del formato BMP descrito por [20], y la cual funciona cuando la imagen de entrada es guardada en formato Windows, con 24 bits por píxel y con un tipo de compresión RGB.



**Figura 3.4:** Ejemplo de imagen de entrada.

## 3.2. Descripción de la Implementación del Sistema

Como se menciona al inicio de este capítulo el sistema se compone de dos etapas principales (localización de los rasgos faciales y localización del contorno del rostro), ambas etapas hacen uso del resultado de tres operaciones aplicadas a la imagen de entrada (como se aprecia en la Figura 3.1): Conversión a niveles de gris, detección de color de piel y cálculo del centro aproximado del rostro.

### **Conversión a niveles de gris.**

La imagen de entrada se convirtió en niveles de gris trasladándola del espacio de color RGB al espacio de color HSI. Como se explicó en la sección 2.3 la imagen monocromática se genera con el promedio de los componentes RGB de cada píxel, es decir, con el valor del componente de intensidad (I) del espacio HSI, como lo indica la ecuación (2.2). La Figura 3.5(b) muestra el resultado de aplicar este proceso a la imagen de la Figura 3.5(a).



(a) imagen de entrada.

(b) imagen monocromática.

**Figura 3.5:** Ejemplo de la conversión en niveles de gris.**Detección de color de piel.**

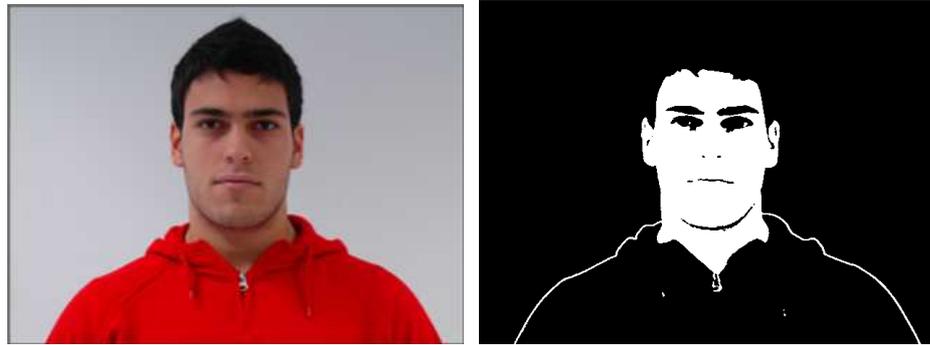
El resultado de la detección de piel es una imagen binaria, generada con la aplicación de la regla (2.26) vista en la sección 2.8. En la imagen binaria los píxeles que tienen valor de 1 son aquellos píxeles de la imagen de entrada cuyos componentes en RGB (rojo, verde y azul) cumplieron con dicha regla (es decir, representan el área de color de piel), mientras que los que tienen el valor de 0 no lo hicieron. Como ejemplo, la Figura 3.6(b) muestra la región de piel localizada en la imagen de la Figura 3.6(a).

El resultado de la detección de piel no solo sirve para ubicar las regiones de color de piel, sino también se utilizó para calcular **el centro aproximado del rostro** de la persona en la imagen, aplicando un promedio de la posición en  $x$  y en  $y$  de los píxeles detectados como piel:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i \quad (3.1)$$

$$\bar{y} = \frac{1}{N} \sum_{i=1}^N y_i \quad (3.2)$$

donde  $x$  y  $y$  son las coordenadas espaciales de los  $N$  píxeles de color de piel.



(a) imagen de entrada.

(b) imagen binaria.

**Figura 3.6:** Ejemplo de la detección de piel.

A continuación se explica cómo cada una de las etapas principales hace uso de estas tres operaciones. Los valores empleados en estas dos etapas se obtuvieron experimentalmente: Los valores para los diferentes umbrales, tamaños de áreas de búsqueda, tamaños de ventanas y vecindades, condiciones de paro, valores de ángulos, la forma de obtener y eliminar los diferentes máximos locales, medidas de proporción derivadas del modelo geométrico y recuadros que envuelven a los rasgos faciales, y cantidad de *snaxels* pertenecientes a la quijada. Se hace mención cuando los valores son tomados de los reportados en la literatura.

### 3.2.1. Localización de los Rasgos Faciales

Para localizar los rasgos faciales dentro de la imagen se implementaron los siguientes módulos:

#### 3.2.1.1. Detección de la Silueta de la Cabeza de la Persona

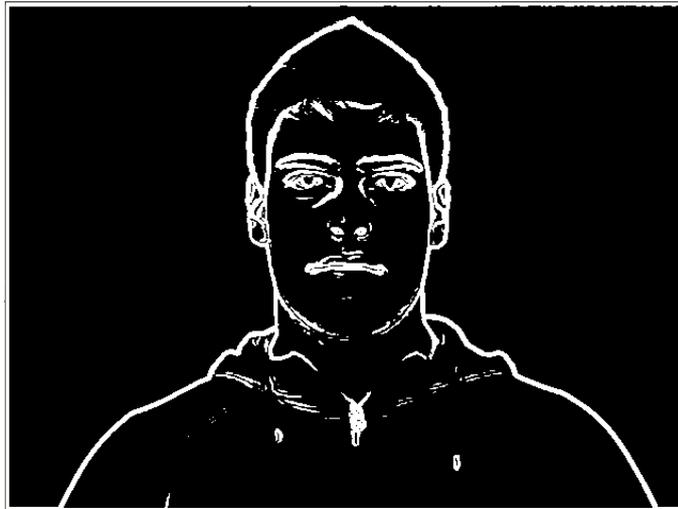
Como primer paso dentro de la localización de los rasgos faciales se ubicó a la cabeza de la persona dentro de la imagen a través de su silueta, ésto se consiguió de la siguiente manera:

1. Se aplicó el operador de Sobel a la imagen en niveles de gris para detectar bordes según las ecuaciones (2.24) y (2.25) vistas en la sección 2.7.3.1 (Figura 3.7).



**Figura 3.7:** Ejemplo de la aplicación de Sobel a la imagen de entrada convertida en niveles de gris.

2. Para obtener un contorno mejor definido de los bordes de la imagen, se aplicó a la imagen resultante del proceso anterior binarización con el siguiente umbral  $M^2 > 2693$  ( $M$  es especificada por la ecuación (2.23) y se determino de manera experimental para obtener bordes gruesos en la imagen de entrada), de acuerdo a como se explicó en la sección 2.5, donde los valores menores o iguales a dicho umbral tendrán un valor de cero (negro) dentro de la imagen, y los valores superiores al umbral serán uno (blanco), ésto se puede apreciar en la Figura 3.8.
3. Se estableció el punto inicial para la detección de la silueta de la cabeza con coordenadas  $(x, y)$ , donde las coordenadas  $x, y$  se obtuvieron de la siguiente manera:
  - a. Se fijó una región de búsqueda basada en el centro aproximado del rostro  $(\bar{x}, \bar{y})$ , definido por las ecuaciones (3.1) y (3.2), esta región corresponde al rectángulo en color rojo de la Figura 3.9. El tamaño horizontal de esta región es de ocho píxeles, es



**Figura 3.8:** Binarización aplicada al resultado de Sobel.

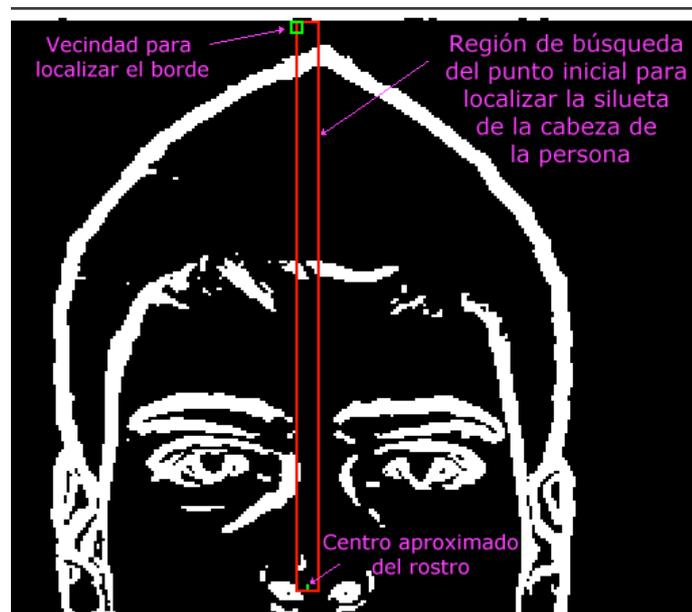
decir, cuatro píxeles anteriores y cuatro píxeles posteriores a  $\bar{x}$ , y el tamaño vertical de esta región está definido por la siguiente ecuación:

$$|(\text{Límite\_superior\_de\_la\_imagen} + 10\_píxeles) - \bar{y}| \quad (3.3)$$

Con las dimensiones de esta región de búsqueda se trata de evitar los bordes producidos en el límite superior de la imagen y aquellos bordes generados por el rostro de la persona.

- b. Se establece una vecindad para la localización del borde dentro de la región anterior de 24 vecinos (considerando la presencia de bordes gruesos en la silueta de la persona), el área de vecindad se aprecia en la Figura 3.9 con la ventana color verde.
- c. Finalmente para la localización de las coordenadas  $x$ ,  $y$  del punto inicial para la detección de la silueta se hace un barrido sobre la región especificada en el punto (a), analizando en cada iteración del barrido la vecindad definida en (b), el punto inicial del barrido es el centro de la ventana color verde que se observa en la Figura 3.9. El análisis consiste en determinar el número de vecinos pertenecientes al borde, es decir, cuántos píxeles tienen el valor binario uno (blanco), y se encuentra el punto

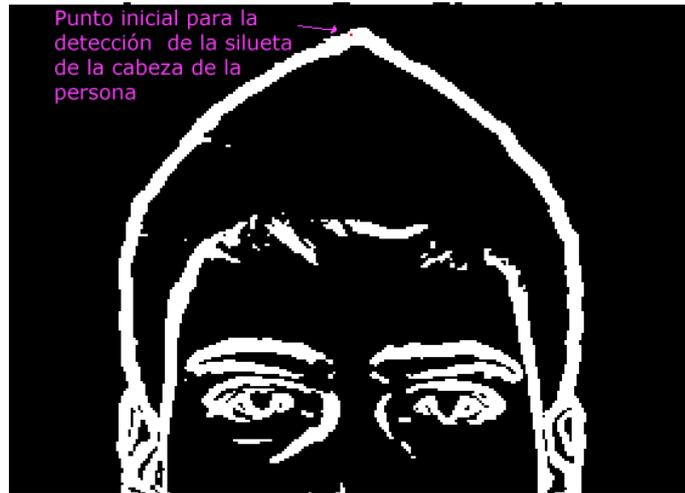
inicial de la silueta cuando 20 de los 24 vecinos son borde, se estableció este límite para colocar al punto inicial lo más cerca posible al centro del borde. Por lo que la coordenada  $(x, y)$  del punto inicial para la detección de la silueta de la cabeza es la posición en  $x, y$  del centro de la vecindad en la que se detuvo el barrido. Sin embargo, si el barrido recorre toda la ventana de búsqueda sin haber encontrado el punto inicial de la silueta, se realizará nuevamente el barrido disminuyendo cada vez en cinco (valor determinado empíricamente y es la diferencia que se produce cuando el ancho del borde es menor al estimado) el número de píxeles necesarios para encontrar el punto inicial de la silueta, es decir, si se tuviera que hacer por segunda vez el barrido se requeriría que 15 de los 24 vecinos fueran borde para determinar el punto inicial de la silueta, y así sucesivamente hasta que se requieran 5.



**Figura 3.9:** Imagen que muestra los elementos involucrados para determinar el punto inicial para la detección de la silueta de la cabeza.

El resultado de este proceso aparece en la Figura 3.10 donde se aprecia con rojo

el píxel que indica el punto inicial del borde para la detección de la silueta de la cabeza.



**Figura 3.10:** Punto inicial del borde para la detección de la silueta de la cabeza.

4. Considerando el punto inicial del borde de la silueta de la cabeza determinado en el punto 3, se realizó la detección de dicha silueta con el algoritmo de la Figura 3.11. En la Figura 3.12(a) se aprecian gráficamente algunas variables involucradas en este algoritmo, como los ángulos  $\theta_o$ ,  $\phi$  y  $\theta'$  medidos a partir del eje positivo de las abscisas del plano cartesiano con origen en el punto actual  $(P_x, P_y)$ ; el vector con dirección y sentido al centro aproximado del rostro aparece en color rojo; en azul se muestra el vector con dirección y sentido hacia la posible nueva posición con coordenadas  $(\overline{Pt_x}[n], \overline{Pt_y}[n])$ ; por otro lado también se observa la posición en  $y$  del centro aproximado del rostro, esta altura  $\bar{y}$  se utiliza para detener el algoritmo.

Ahora se detallará el funcionamiento del algoritmo: Se detectan puntos a lo largo del contorno de la silueta de la cabeza en dos partes, con un recorrido hacia la derecha y otro hacia la izquierda, la dirección del recorrido cambia según la inicialización del ángulo de dirección  $\theta$  al comienzo del algoritmo, para cada lado derecho o izquierdo

de la silueta el número máximo de puntos del borde es 46. La detección de los puntos comienza en el punto inicial de la silueta de la cabeza localizado previamente y denotado por  $(P_x, P_y)$ , en cada iteración se busca el siguiente punto que pertenece a la silueta de la cabeza partiendo de  $(P_x, P_y)$  y al ser encontrado dicho punto, éste será el punto inicial para la siguiente iteración, es decir,  $(P_x, P_y)$  será remplazado con el nuevo valor. La búsqueda del siguiente punto se realiza en tres direcciones a partir de  $\theta$ , la primera búsqueda se hace en la dirección  $\theta - \frac{\pi}{4}$ , la segunda en la misma dirección del recorrido  $\theta$  y la tercera en la dirección  $\theta + \frac{\pi}{4}$ , en cada dirección de búsqueda se establece una ventana de tamaño  $15 \times 15$  cuyo centro está dado por  $(Pt_x[k], Pt_y[k])$  (líneas 8 y 9 del algoritmo de la Figura 3.11) y se calcula la densidad del borde de cada una de estas tres ventanas, la densidad del borde se obtiene contabilizando el número de píxeles en la vecindad con valor de 1, también se obtiene la posición promedio de estos píxeles  $(\overline{Pt_x}[n], \overline{Pt_y}[n])$  (líneas 20 y 21 del algoritmo de la Figura 3.11). El siguiente punto de la silueta de la cabeza estará determinado por la posición promedio  $(\overline{Pt_x}[n], \overline{Pt_y}[n])$  de la ventana que contenga la densidad máxima de entre las tres ventanas analizadas.

Sin embargo, como el proceso anterior sólo considera bordes gruesos para determinar la nueva posición de  $(P_x, P_y)$  puede causar que durante la detección del contorno se produzcan comportamientos no deseados, tales como alejamiento de la silueta o la intromisión en el área del rostro, para evitar estos comportamientos se limita la dirección de la nueva posición dentro de cierto rango de curvatura considerando la forma del rostro, para ello se emplea a  $\theta'$ , el cual es el ángulo formado entre el vector con dirección al centro aproximado del rostro y el vector con dirección a la nueva posición, el valor de este ángulo debe estar contenido en el rango de  $[\frac{\pi}{8}, \frac{9}{10}\pi]$  (para el recorrido derecho), ó  $[\frac{\pi}{10}, \frac{7}{8}\pi]$  (para el recorrido izquierdo), este rango puede apreciarse en la Figura 3.12(b) (para el recorrido izquierdo) por las líneas etiquetadas como límite infe-

rior y límite superior en color rojo. Si la dirección de la nueva posición se sale de este rango se corrige esta desviación cambiando la nueva posición  $(\overline{Ptx}[n], \overline{Pty}[n])$  a una posición sugerida considerando una silueta circular, esta posición sugerida está dada por las componentes en  $x$  y en  $y$  del vector tangente al círculo con origen en el centro aproximado del rostro y que tiene como punto de aplicación  $(P_x, P_y)$  e intensidad 15, como se aprecia en la Figura 3.12(c) por el vector en color azul. Por otra parte, cuando no exista borde a seguir (cuando  $D[n] = 0$ ) entonces la dirección sugerida para la nueva posición tendrá el valor de  $\frac{\pi}{2}$ , es decir, se continuará avanzando en dirección vertical, esta dirección es dibujada por el vector en color verde en la Figura 3.12(c) y el cálculo de esta nueva posición se muestra en las líneas 28 y 29 del algoritmo de la Figura 3.11.

Lo anterior es válido cuando se ejecuta el recorrido hacia la izquierda excepto por algunos ajustes: Para iniciar el ángulo  $\theta$  se establece en  $\pi$ , el ángulo  $\theta'$  debe considerar el cambio de cuadrante del vector con punto de aplicación  $(P_x, P_y)$  y dirección al centro aproximado del rostro, situación que se aprecia en la Figura 3.12(d). Cuando se da este cambio de cuadrante el ángulo  $\theta_o$  es mayor a  $\frac{3}{2}\pi$  y dado que el valor que requiere el algoritmo para  $\theta'$  es el indicado por la línea en color verde de la Figura 3.12(d), es necesario realizar el ajuste indicado por la línea 51 del algoritmo de la Figura 3.11. La última modificación está relacionada con la posición  $P_x$  (línea 56 del algoritmo de la Figura 3.11), ya que ahora  $P_x$  no deberá sobrepasar a  $\bar{x}$  porque el recorrido se efectúa en sentido opuesto.

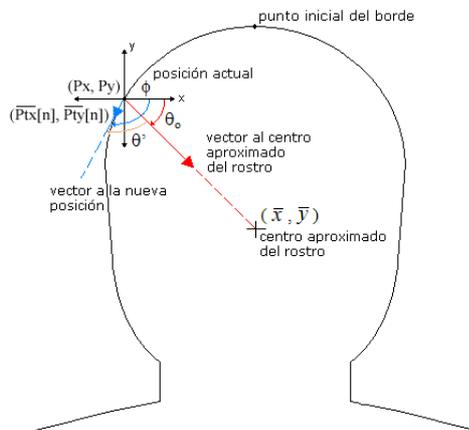
Es importante señalar que hay dos condiciones más para detener el algoritmo, una de ellas establece que el algoritmo se detendrá cuando el número de puntos analizados sea igual a 46, por lado, la otra condición considera que  $P_x$  no deberá ser menor a  $\bar{x}$  si el recorrido es hacia la derecha, ó  $P_x$  no deberá ser mayor a  $\bar{x}$  si el recorrido es hacia la izquierda.

```

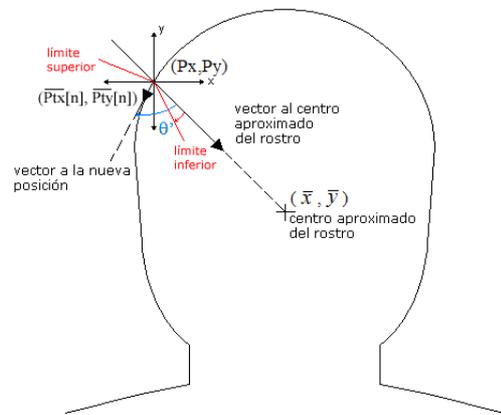
/* Algoritmo para el seguimiento del lado derecho de la silueta de la persona */
1 Inicializar  $\theta = 0$  /*  $\theta$  es el ángulo que indica la dirección del seguimiento del contorno */
2 Inicializar  $d = 15$  /* distancia de separación entre cada punto del contorno */
3 Inicializar  $P_x, P_y$  /*  $P_x$  y  $P_y$  se inicializan igual a la posición inicial del borde de la silueta */
4 for  $m = 0$  to 46 /*  $m$  es el número máximo de puntos en el borde para cada lado de la silueta */
5     Inicializar incremento =  $-\pi/4$ 
/* calcular la nueva posible posición de  $P_x, P_y$  */
6     for  $k = 0$  to 2 /* tres posibles posiciones para  $P_x, P_y$  variando según  $\text{ang}[k]$  */
7          $\text{ang}[k] = \theta + \text{incremento}$  /* el ángulo  $\text{ang}[k]$  cambia en  $-\pi/4, 0$  y  $\pi/4$  */
8          $\text{Pt}_x[k] = P_x + \cos(\text{ang}[k]) * d$ 
9          $\text{Pt}_y[k] = P_y + \sin(\text{ang}[k]) * d$ 
10         $D[k] = 0$  /* inicialización de la densidad del borde */
/* la densidad del borde es el número de vecinos pertenecientes al borde */
11         $\overline{\text{Pt}_x}[k] = 0$ 
12         $\overline{\text{Pt}_y}[k] = 0$ 
13        for  $i = -7$  to 7 /* tamaño de la vecindad a analizar */
14            for  $j = -7$  to 7
15                /*  $B$  es la imagen binaria obtenida a partir de Sobel */
16                if  $B(\text{Pt}_x[k] + i, \text{Pt}_y[k] + j) = 1$  then /* si el punto analizado es parte del borde */
17                     $D[k] += B(\text{Pt}_x[k] + i, \text{Pt}_y[k] + j)$  /* la densidad es aumentada */
18                     $\overline{\text{Pt}_x}[k] += \text{Pt}_x[k] + i$  /* se almacenan las posiciones encontradas */
19                     $\overline{\text{Pt}_y}[k] += \text{Pt}_y[k] + j$ 
20                end if
21            end for
22             $\overline{\text{Pt}_x}[k] = \frac{\overline{\text{Pt}_x}[k]}{D[k]}$  /* posición promedio en x de la vecindad analizada */
23             $\overline{\text{Pt}_y}[k] = \frac{\overline{\text{Pt}_y}[k]}{D[k]}$  /* posición promedio en y de la vecindad analizada */
24            incremento +=  $\pi/4$ 
25        end for
26         $n = \text{máximo}(D)$  /* máximo(D) devuelve la k posición del arreglo con un el valor más alto en D, por lo que */
/* se establece n como la k posición con densidad máxima */
27        if  $D[n] = 0$  then /* si la densidad máxima fue cero, es decir, no hay borde que seguir */
28             $\theta = \pi/2$  /* la nueva posición sugerida tiene una dirección vertical */
29             $\phi = \pi/2$ 
30             $P_x += \cos(\theta) * d$ 
31             $P_y += \sin(\theta) * d$ 
32        else
33             $\phi =$  ángulo que se dirige hacia la nueva posición  $\overline{\text{Pt}_x}[n], \overline{\text{Pt}_y}[n]$ 
34        end if
35        /* cálculo para  $\theta'$ , la cual es el ángulo formado por los vectores que parten de  $P_x, P_y$  y se dirigen, uno hacia la nueva posición y el otro hacia */
/* el centro aproximado del rostro */
36         $\theta' = \theta - \phi$ 
37        if  $(\theta' > \pi - (\pi/10))$  ó  $\theta' < \pi/8$  then /* si  $\theta'$  está fuera del rango establecido */
38            /* la dirección de la nueva posición se está alejando de la silueta y debe ser corregida */
39            /* mover el punto  $(P_x, P_y)$  a la posición sugerida por la tangente al vector que se dirige al centro aproximado del rostro */
40             $P_x += \cos(\theta - \pi/2) * d$  /* el objetivo de esto es mantener una silueta circular */
41             $P_y += \sin(\theta - \pi/2) * d$ 
42        else if  $D[n] \neq 0$  then /* sino si existe la posición promedio, es decir,  $D[0] \neq 0$ , mover el punto  $(P_x, P_y)$  a la nueva posición */
43             $P_x = \overline{\text{Pt}_x}[n]$  /* la nueva posición se establece en la posición promedio de la vecindad analizada */
44             $P_y = \overline{\text{Pt}_y}[n]$ 
45            /* mover el ángulo de búsqueda para el siguiente punto del borde con dirección a el ángulo con densidad máxima */
46             $\theta = \text{ang}[n]$ 
47        end if
48        /* condición para detener el seguimiento del borde */
49        if  $P_y > \bar{y}$  ó  $P_x \leq \bar{x}$  then /* si la posición  $P_y$  sobrepasa la altura del centro aproximado del rostro */
/* ó la posición  $P_x$  es inferior a el centro aproximado del rostro */
/* entonces se termina el ciclo de seguimiento */
50             $m = 46$ 
51        else
52            LadoD[m][0] =  $P_x$ ; /* caso contrario, se almacena la posición del borde */
53            LadoD[m][1] =  $P_y$ ;
54        end if
55    end for
/* el algoritmo anterior puede ser usado para el seguimiento del lado izquierdo del contorno con las siguientes modificaciones */
/* Inicializar  $\theta$  */
56  $\theta = \pi$  /* que indica la dirección izquierda del seguimiento */
/* cálculo para  $\theta'$ , se considera el cambio de cuadrante para el vector que se dirige al centro aproximado del rostro */
57 if  $\theta_0 > (2 * \pi) - (\pi/2)$  then
58      $\theta' = (2 * \pi - \theta_0) + \phi$  /* ajuste para la medición de  $\theta'$  */
59 else
60      $\theta' = \phi - \theta_0$ 
61 end if
62 if  $(\theta' > \pi - (\pi/8))$  ó  $\theta' < \pi/10$  then /* nuevo rango para la condición de la línea 34 */
/* condición para detener el seguimiento del borde */
63 if  $P_y > \bar{y}$  ó  $P_x > \bar{x}$  then /* si la posición  $P_y$  sobrepasa la altura del centro aproximado del rostro */
/* ó la posición  $P_x$  sobrepasa el centro aproximado del rostro */
/* entonces se termina el ciclo de seguimiento */
64      $m = 46$ 
65 else
66     LadoI[m][0] =  $P_x$ ; /* caso contrario, se almacena la posición del borde */
67     LadoI[m][1] =  $P_y$ ;
68 end if
69 end if

```

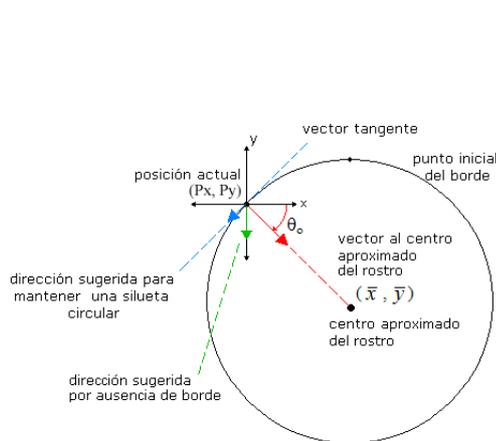
Figura 3.11: Algoritmo para la detección de la silueta de la cabeza.



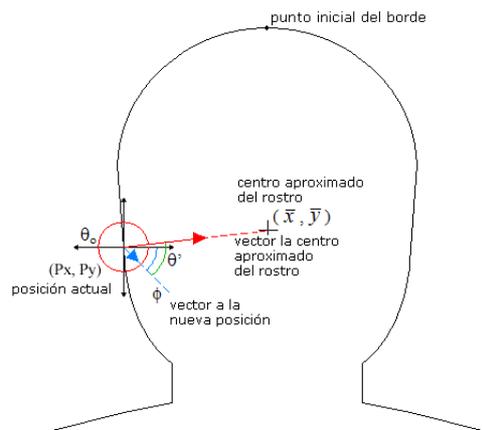
(a) variables involucradas.



(b) rango de curvatura permitido para  $\theta'$ .



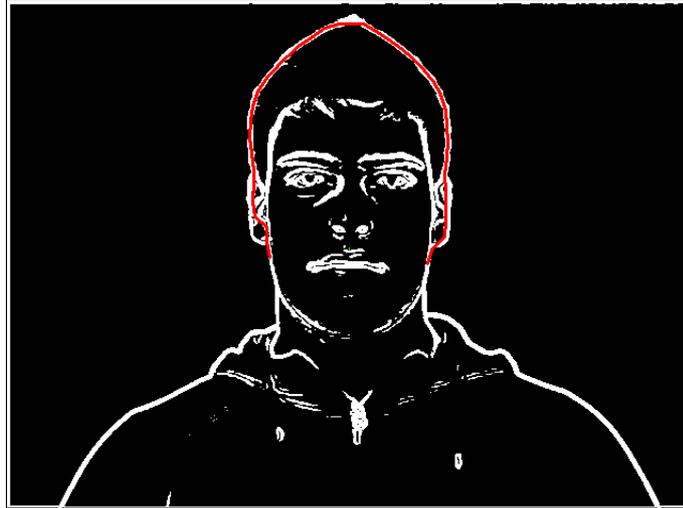
(c) dirección sugerida considerando una silueta circular.



(d) cambio de cuadrante del vector con dirección al centro aproximado del rostro.

**Figura 3.12:** Consideraciones gráficas del algoritmo para la detección de la silueta de la cabeza.

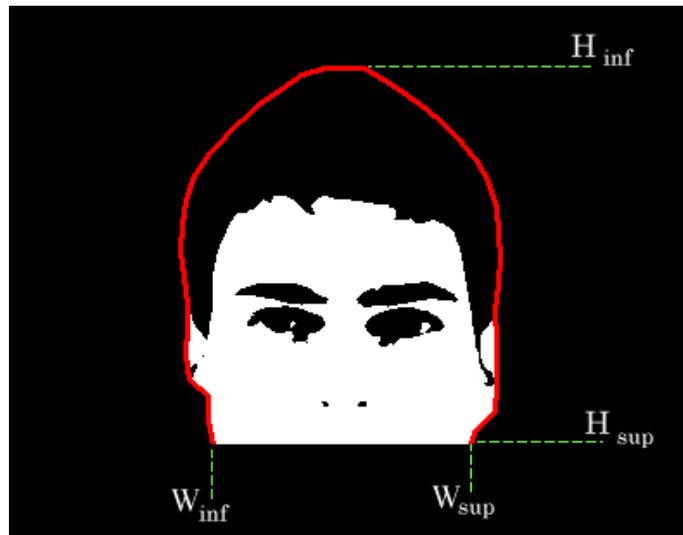
En la Figura 3.13 se resalta con la línea en rojo el resultado del proceso para la detección de la silueta de la cabeza.



**Figura 3.13:** Resultado del proceso para la detección de la silueta de la cabeza.

### 3.2.1.2. Ubicación de la Parte Superior del Rostro.

Para encontrar la posición vertical y horizontal de los ojos es necesario especificar el área del rostro en donde posiblemente se encuentren éstos, para ello se localiza la posición vertical que define la parte superior del rostro, esta posición se determina con ayuda de la ecuación (2.7) (descrita en la sección 2.4), donde  $P(y)$  es la proyección vertical del histograma en la posición  $y$  de la imagen,  $I$  es el valor de la intensidad de la imagen en la posición  $(x, y)$ ,  $H_{inf}$  y  $H_{sup}$  son los límites de altura inferior y superior respectivamente,  $W_{inf}$  y  $W_{sup}$  son el límite izquierdo y derecho de la región de búsqueda en la imagen, esta región es el área obtenida en el proceso anterior de la detección de la silueta de la cabeza y se muestra en la Figura 3.14.

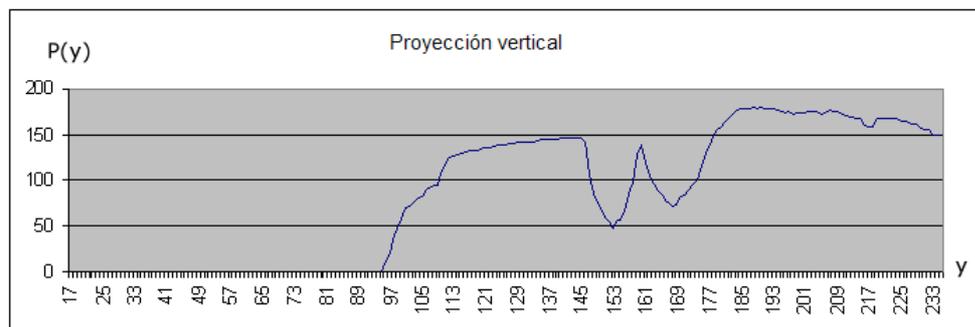


**Figura 3.14:** Región de búsqueda para la proyección vertical.

El valor de intensidad de la imagen se determina utilizando la información de la imagen binaria resultante de la detección de piel, como puede verse en la ecuación (3.4), donde la  $I$  toma un valor igual a uno si el píxel en la posición analizada es considerado como color de piel, de lo contrario, si el píxel analizado no es piel el valor de la intensidad será cero.

$$I(x, y) = \begin{cases} 1 & \text{Si el píxel en la posición } (x, y) \text{ es piel.} \\ 0 & \text{Si el píxel en la posición } (x, y) \text{ no es piel.} \end{cases} \quad (3.4)$$

Esta proyección vertical muestra la densidad de color de piel en la posición  $y$ , en la gráfica de la Figura 3.15 se muestra el resultado de esta proyección vertical aplicada a la región de búsqueda de la Figura 3.14.

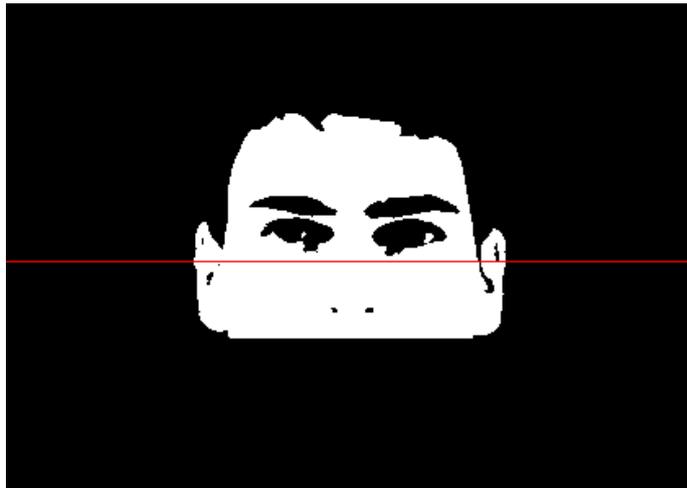


**Figura 3.15:** Gráfica resultante de la proyección vertical aplicada a la Figura 3.14.

El resultado de la ubicación de la parte superior del rostro es la posición en  $y$  del máximo global dentro de la gráfica resultante de la proyección vertical, es decir, el máximo global representa a la región con mayor densidad de color de piel y ésta se encuentra por debajo de los ojos y sobre la posición de la boca. El comienzo de la parte superior del rostro puede verse en la Figura 3.16 por la línea en color rojo.

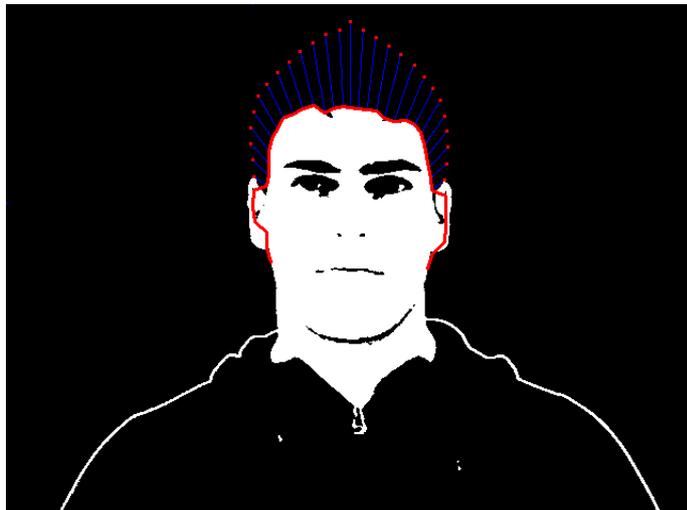
### 3.2.1.3. Ubicación Aproximada del Contorno del Rostro

A partir de los puntos considerados parte de la silueta de la cabeza y con ayuda de la imagen binaria resultado de la detección de piel, se hizo una reducción de la silueta al contorno del rostro. Para realizar dicha reducción, cada punto de la silueta es desplazado con dirección al centro aproximado del rostro y detenido sobre el borde del color de piel, empleando para la detención una ventana de  $7 \times 7$  y considerando borde cuando 16 ó más píxeles de la ventana



**Figura 3.16:** Resultado de la ubicación de la parte superior del rostro de la Figura 3.14.

son color de piel. En la Figura 3.17 se puede distinguir el desplazamiento de los puntos con las líneas en color azul y el resultado de la ubicación aproximada del contorno del rostro por la línea en color rojo.



**Figura 3.17:** Resultado del desplazamiento de los puntos de la silueta al contorno del rostro.

### 3.2.1.4. Proyección Horizontal y Vertical del Histograma en el Área Superior del Rostro

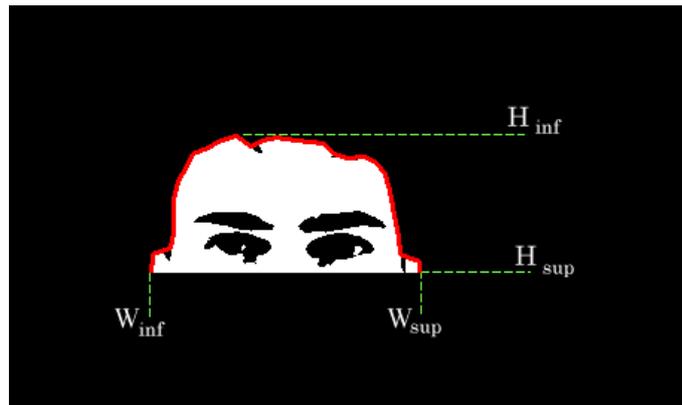
El propósito de esta etapa es localizar el centro de los ojos, lo cual se realizó en dos fases, primero se calcula la posición vertical del centro de ambos ojos y después la posición horizontal del centro de cada uno de ellos, para obtener la posición vertical u horizontal de los ojos se utilizó la proyección del histograma vista en la sección 2.4 sobre la imagen en niveles de gris. Considerando la morfología de la persona se aplicó la proyección vertical y horizontal del histograma sobre la ubicación aproximada del contorno del rostro (Figura 3.17) delimitada por la parte superior de éste (Figura 3.16).

En la primera fase, la posición vertical del centro de ambos ojos se determina con ayuda de la ecuación (2.7) (descrita en la sección 2.4) modificada, esta modificación invierte el valor para los píxeles en la imagen (considerando el rango de 0 a 255), haciendo énfasis en determinar la ubicación de los píxeles oscuros, esto debido a que en la imagen en niveles de gris la superficie de los ojos tiene alta concentración de tonos oscuros, por lo tanto, la proyección vertical es calculada con la siguiente ecuación:

$$P(y) = \sum_{x=W_{inf}}^{W_{sup}} (255 - I(x, y)) \quad \text{Donde } H_{inf} \leq y \leq H_{sup} \quad (3.5)$$

Donde  $P(y)$  es la densidad de niveles de gris o proyección vertical del histograma en la posición  $y$  de la imagen,  $I$  es el valor de la intensidad de la imagen en niveles de gris en la posición  $(x, y)$ ,  $H_{inf}$  y  $H_{sup}$  son los límites de altura inferior y superior respectivamente de la región de búsqueda en la imagen,  $W_{inf}$  y  $W_{sup}$  son el límite izquierdo y derecho, esta región se muestra en la Figura 3.18.

Además, la información de la imagen binaria resultante de la detección de piel se tiene en cuenta para determinar cuál es el valor de la intensidad de la imagen en niveles de gris que será considerado en la ecuación (3.5), como puede verse en la ecuación (3.6), donde la  $I$



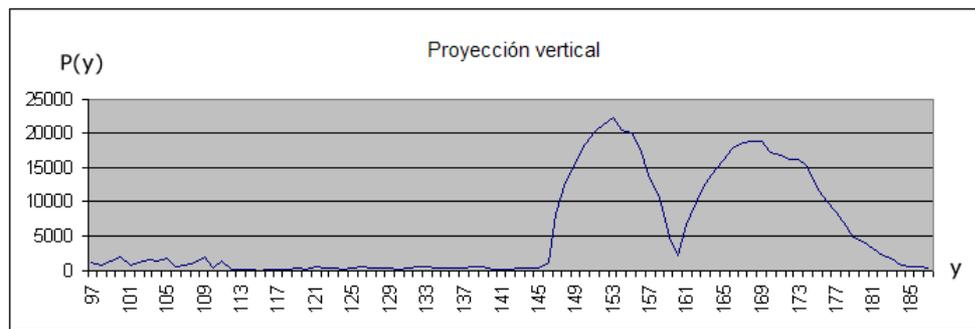
**Figura 3.18:** Área superior del contorno aproximado del rostro donde se aplicó la proyección del histograma.

toma un valor igual a cero si el píxel en la posición analizada es considerado como de color de piel, es decir, si el píxel de la imagen binaria resultado de la detección de piel en la posición analizada contiene el valor de 1, esto aumenta la eficiencia de la proyección restringiendo la selección de los píxeles.

$$I(x, y) = \begin{cases} 255 & \text{Si el píxel en la posición } (x, y) \text{ es piel.} \\ \text{Nivel de gris en } (x, y) & \text{Si el píxel en la posición } (x, y) \text{ no es piel.} \end{cases} \quad (3.6)$$

La gráfica obtenida por la aplicación de la proyección vertical a la imagen de la Figura 3.18 puede verse en la Figura 3.19. En esta gráfica la posición en  $y$  del rostro se encuentra en el eje de las abscisas y en el eje de las ordenadas se encuentra el valor de intensidad obtenido por la proyección vertical.

Los máximos locales contenidos en la gráfica nos revelan la posición vertical donde se encuentran las altas concentraciones de color oscuro en la imagen, en esas concentraciones de color se ubican las posiciones medias en  $y$  de las cejas, los ojos, entre otros rasgos propios de esta región (manchas en el rostro, cabello, sombras, entre otros). Por lo que para obtener el



**Figura 3.19:** Gráfica de la proyección vertical sobre la región superior del rostro.

máximo local perteneciente a la posición vertical del centro de los ojos, es necesario extraer los máximos locales de la gráfica y posteriormente analizarlos para determinar cuál de estos máximos corresponde al máximo buscado.

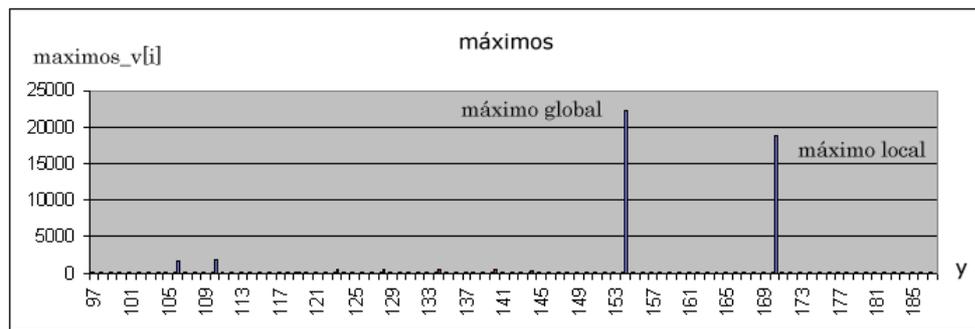
En la etapa de extracción se almacena a  $P(y)$  de manera discreta en un arreglo unidimensional denotado por **salida\_v**[ $i$ ], donde cada elemento  $i$  de **salida\_v** contiene el valor de intensidad de  $P(y)$ , el tamaño de este arreglo depende del alto del área superior del contorno aproximado del rostro, es decir, el tamaño es igual a  $H_{sup} - H_{inf}$ . Sobre este arreglo se hace un recorrido en el cual se considera al elemento  $i$  de este arreglo un máximo si cumple con la siguiente regla:

$$\mathbf{salida\_v}[i - 2] < \mathbf{salida\_v}[i] \text{ y } \mathbf{salida\_v}[i + 2] < \mathbf{salida\_v}[i] \quad (3.7)$$

Para  $i - 2 \geq 0$  y  $i + 2 < \text{Tamaño del arreglo salida\_v}$

Cada elemento máximo detectado es almacenado en un nuevo arreglo llamado **maximos\_v** de igual tamaño que **salida\_v**, y que contiene para cada elemento diferente a un máximo un valor de cero. El arreglo **maximos\_v** derivado de la gráfica de la Figura 3.19 es graficado en la Figura 3.20.

En la etapa de análisis se localiza el mayor de entre los elementos almacenados en el arreglo **maximos\_v**, debido a que en la posición y del máximo global se encuentra la posible



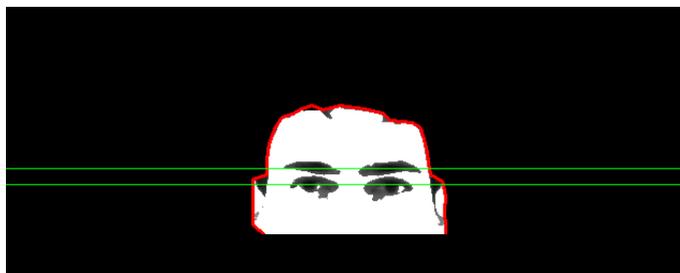
**Figura 3.20:** Gráfica del arreglo **maximos\_v**.

posición media de los ojos o cejas, para hacer la distinción de entre ojos y cejas de los diferentes máximos obtenidos en el arreglo **maximos\_v**, se eliminan todos aquellos elementos  $i$  del arreglo que tengan una distancia de separación con respecto al máximo global menor a 6 píxeles ó mayor a 30 (distancias obtenidas empíricamente), también aquellos elementos del arreglo cercanos al límite inferior de la región de búsqueda, es decir, no se consideran 5 elementos al comienzo del arreglo.

Debido a que haya ausencia o distorsión de cejas causadas por sombras, oclusiones, o por su similitud con el color de la piel, puede suceder que sólo el máximo global ó el máximo global y uno o más máximos locales sean extraídos del análisis anterior. Cuando sólo el máximo global es extraído, su posición en  $y$  corresponde de manera directa con la posición del centro vertical de los ojos, en caso contrario se recurre al ordenamiento de los máximos locales por su intensidad para encontrar al mayor de ellos, así las posiciones en  $y$  del máximo global y del máximo local mayor al resto corresponden a la posición media de los ojos o cejas según su ubicación dentro de la imagen, es decir, el que tenga la posición en  $y$  más cercana al límite superior de la región de búsqueda define la posición media de los ojos, un ejemplo de lo anterior aparece en la Figura 3.20 donde el máximo global se encuentra en la posición  $y$  igual a 153, el mayor de los máximos locales tiene la posición en  $y$  igual a 168, por lo tanto, la posición en  $y$  del mayor de los máximos locales corresponde al centro vertical de los ojos

y la posición en  $y$  del máximo global a la posición media de las cejas.

La Figura 3.21 muestra un ejemplo del resultado del análisis anterior, en el cual se localizaron ojos y cejas, y cuya posición media en  $y$  es dibujada por una línea verde.

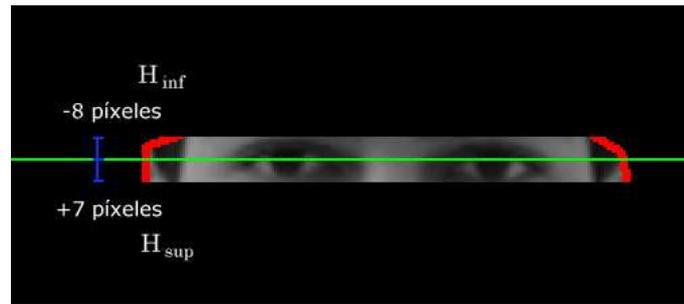


**Figura 3.21:** Posición media de ojos y cejas

En la segunda fase, se determina la posición horizontal del centro para cada uno de los ojos con ayuda de la ecuación (2.6) (descrita en la sección 2.4) modificada, y al igual que en la proyección vertical esta modificación invierte el valor para los píxeles en la imagen, por lo tanto, la proyección horizontal se calcula con la siguiente ecuación:

$$P(x) = \sum_{y=H_{inf}}^{H_{sup}} (255 - I(x, y)) \quad \text{Donde } W_{inf} \leq y \leq W_{sup} \quad (3.8)$$

Donde  $P(x)$  es la densidad de niveles de gris o proyección horizontal del histograma en la posición  $x$  de la imagen,  $I$  es el valor de la intensidad de la imagen en niveles de gris en la posición  $(x, y)$ ,  $W_{inf}$  y  $W_{sup}$  son los límites del ancho de la región de búsqueda en la imagen,  $H_{inf}$  y  $H_{sup}$  son los límites de altura inferior y superior respectivamente, los cuales han sido redefinidos, su nuevo valor recae sobre la región aproximada de los ojos, evitando que la proyección horizontal sea afectada por las cejas, por consiguiente los límites  $H_{inf}$  y  $H_{sup}$  se encuentran posicionados a una distancia de 8 píxeles sobre el centro vertical de los ojos y 7 píxeles por debajo del centro vertical de los ojos respectivamente (distancias obtenidas empíricamente), un ejemplo de esta región se aprecia en la Figura 3.22.



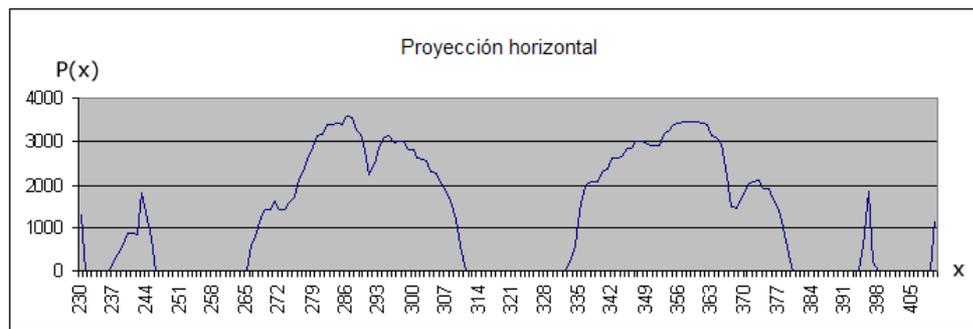
**Figura 3.22:** Región aproximada de los ojos empleada en la proyección horizontal.

Durante el cálculo de la proyección horizontal se utiliza la información de la imagen binaria resultante de la detección de piel, de esta manera se determina cual es el valor de la intensidad de la imagen en niveles de gris que será considerado, como puede verse en la ecuación (3.9), donde la  $I$  toma un valor igual a cero si el píxel en la posición analizada es considerado como de color de piel, es decir, si el píxel de la imagen binaria resultado de la detección de piel en la posición analizada contiene el valor de 1, esto aumenta la eficiencia de la proyección restringiendo la selección de los píxeles.

$$I(x, y) = \begin{cases} 255 & \text{Si el píxel en la posición } (x, y) \text{ es piel.} \\ \text{Nivel de gris en } (x, y) & \text{Si el píxel en la posición } (x, y) \text{ no es piel.} \end{cases} \quad (3.9)$$

La gráfica generada por la aplicación de la proyección horizontal a la región aproximada de los ojos (Figura 3.22) puede verse en la Figura 3.23, en esta gráfica la posición en  $x$  del rostro se encuentra en el eje de las abscisas y en el eje de las ordenadas el valor de intensidad obtenido por la proyección horizontal.

De modo similar que en la gráfica de la proyección vertical, los máximos locales contenidos en la gráfica corresponden a la posición horizontal de los ojos y otros rasgos detectados. Para determinar qué máximos locales corresponden al centro horizontal de cada uno de



**Figura 3.23:** Gráfica de la proyección horizontal de la Figura 3.22.

los ojos es necesario extraer nuevamente todos los máximos locales y analizarlos.

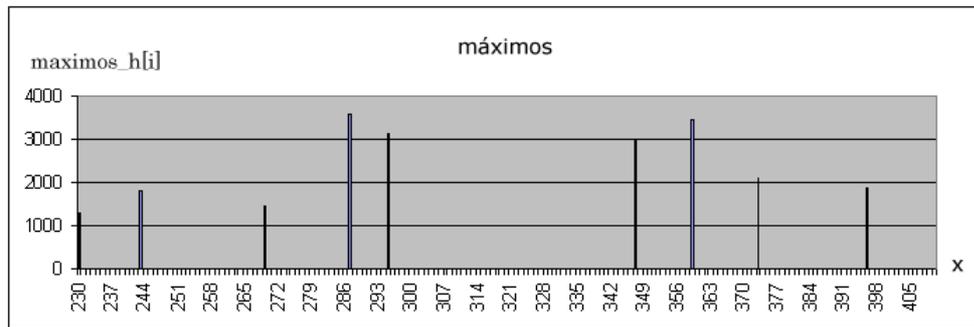
En la extracción de los máximos locales, primero se almacena a  $P(x)$  de manera discreta en un arreglo unidimensional denotado por **salida\_h**[ $i$ ], donde cada elemento  $i$  de **salida\_h** contiene el valor de intensidad de  $P(x)$ , el tamaño de este arreglo depende del ancho de la región aproximada de los ojos, es decir, el tamaño es igual a  $W_{sup} - W_{inf}$ . Sobre este arreglo se hace un recorrido en el cual se considera al elemento  $i$  de este arreglo un máximo si cumple con la siguiente regla:

$$\mathbf{salida\_h}[i] \geq \mathbf{salida\_h}[i - 4] \text{ y } \mathbf{salida\_h}[i] \geq \mathbf{salida\_h}[i + 4] \text{ y } \mathbf{salida\_h}[i] > 0 \quad (3.10)$$

Para  $i - 4 \geq 0$  y  $i + 4 < \text{Tamaño del arreglo } \mathbf{salida\_h}$

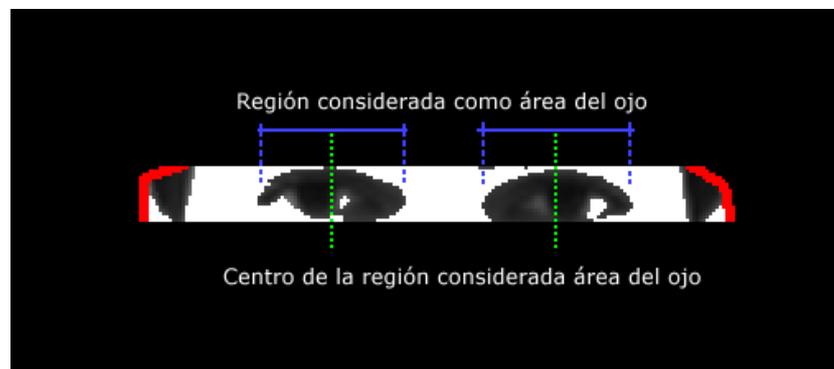
Cada elemento máximo detectado es almacenado en un nuevo arreglo llamado **maximos\_h** de igual tamaño que **salida\_h**, y que contiene para cada elemento diferente a un máximo un valor de cero. El arreglo **maximos\_h** derivado de la gráfica de la Figura 3.23 es graficado en la Figura 3.24.

Posteriormente se analiza el arreglo **maximos\_h** para eliminar máximos locales contiguos, haciendo que el elemento con mayor intensidad de entre éstos conserve su valor y el resto cambiará su valor a cero.



**Figura 3.24:** Gráfica del arreglo **maximos\_h**.

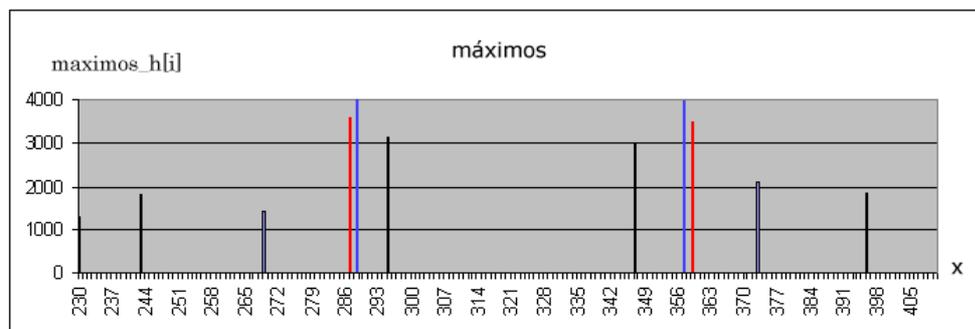
Antes de analizar los máximos locales para determinar el centro horizontal de cada ojo, es necesario calcular una primera aproximación a éste, la cuál se obtiene a partir del centro del área del ojo. Esta área se localiza con ayuda de la información de la detección de piel, así se tiene que el área del ojo se encuentra presente en las regiones consideradas como no piel dentro de la región aproximada de los ojos, un ejemplo de lo anterior se muestra en la Figura 3.25, donde el ancho de la región considerada como área del ojo se encuentra acotada en color azul, y cuyo centro se encuentra indicado por la línea en color verde.



**Figura 3.25:** Primera aproximación al centro del ojo.

Para determinar el centro de cada uno de los ojos, primero es tomado como referencia el centro aproximado de un ojo y se busca dentro del arreglo **maximos\_h** el elemento mayor en intensidad al resto y cuya posición en  $x$  sea la más cercana al punto de referencia, por lo tanto, la posición en  $x$  del elemento encontrado es asignada como la posición horizontal

del centro del ojo, la misma búsqueda se realiza para el ojo restante tomando como punto de referencia el centro aproximado de dicho ojo. Para evitar alejarse demasiado de la primera aproximación al centro del ojo, la búsqueda del elemento más cercano se hace en un rango de seis píxeles anteriores y seis píxeles posteriores a ésta, si ningún elemento es detectado la primera aproximación será entonces el centro del ojo. Un ejemplo de este proceso se aprecia en la Figura 3.26, donde la línea en color azul muestra el punto de referencia, el cual es la posición en  $x$  de la primera aproximación al centro horizontal en cada ojo, y en color rojo se remarca el máximo local mayor en intensidad y más cercano a este punto de referencia, por lo tanto en esta gráfica la posición en  $x$  de los elementos remarcados en rojo corresponden a la posición horizontal del centro de cada ojo.

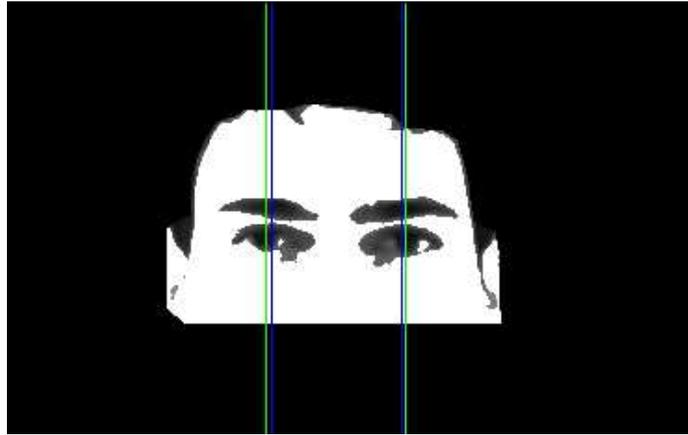


**Figura 3.26:** Gráfica del arreglo **maximos\_h** (color negro y rojo) donde se indica la primera aproximación al centro de cada ojo (color azul).

En la Figura 3.27 se dibujan los resultados obtenidos de la gráfica de la Figura 3.26, donde las posiciones horizontales del centro de cada uno de los ojos aparecen en tono verde, y en azul se encuentra la primera aproximación al centro de cada ojo.

### Centro de los ojos

Para establecer la posición  $(x, y)$  del centro de cada ojo, se hace la unión de las posiciones en  $x$ , y en  $y$  resultados de la aplicación de la proyección horizontal y vertical respectivamente, tomando en consideración que la posición en  $y$  es la misma para ambos ojos, es decir la



**Figura 3.27:** Centro horizontal de cada ojo (color verde) resultado de analizar la gráfica de la Figura 3.26.

persona contenida en la imagen no tiene una rotación significativa sobre el eje de la cámara, ni sobre su propio eje. El resultado de la localización del centro de los ojos puede verse en la Figura 3.28, en la cual se considera el centro del ojo el punto de intersección hecho por las líneas horizontal y vertical dibujadas en color rojo.



**Figura 3.28:** Centro de los ojos obtenidos por la proyección horizontal y vertical.

### 3.2.1.5. Modelo Geométrico del Rostro para Determinar la Posición Aproximada de la Nariz y la Boca

El cálculo de la posición vertical correcta de la nariz y la boca se realiza en dos partes, primero se determina la región donde deben estar contenidas éstas con ayuda de un modelo geométrico del rostro, y después se aplica una proyección vertical sobre esta región para localizar sus respectivas posiciones en  $y$ .

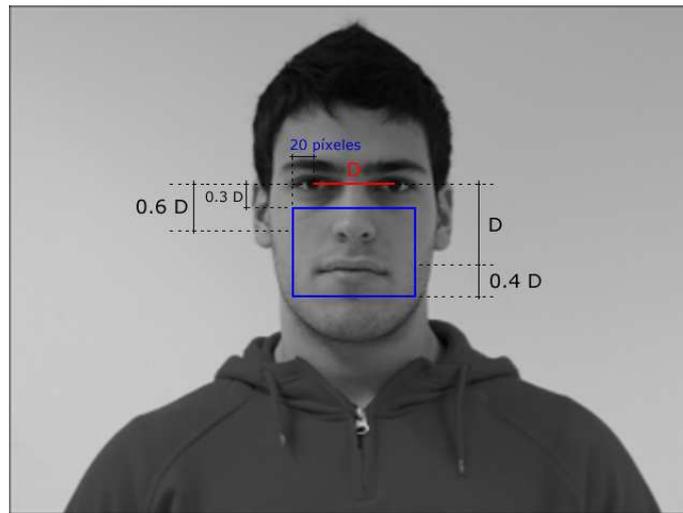
El modelo geométrico del rostro que se implementó en este proyecto de investigación fue el descrito en la sección 2.10, el cual especifica la separación de los rasgos faciales (nariz, boca y ojos). Sin embargo, debido a la variabilidad de las características faciales de cada persona, la coincidencia de los rasgos faciales obtenidos con este modelo geométrico no es precisa, por lo tanto, se emplea este modelo geométrico para definir una región de búsqueda y hallar así la posición exacta de la nariz y boca.

El modelo geométrico del rostro utiliza la distancia de separación entre el centro de los ojos, la cual fue encontrada en la etapa anterior, como medida de proporción con respecto a los demás rasgos faciales, esta medida de proporción es denotada por la letra  $D$ ; también emplea las reglas mencionadas en la sección 2.10, las cuales definen los límites de la región de búsqueda en la cual deben estar contenidas la nariz y la boca:

- a. Considerando que la distancia vertical entre el centro de los ojos y el centro de la nariz es de  $0.6 D$ , el límite vertical superior de la región de búsqueda se definió como  $0.3D$ .
- b. El límite vertical inferior de la región de búsqueda se establece por debajo del centro de la boca, por lo tanto tiene un valor de  $1.4 D$ .
- c. Al ser el ancho de la boca igual a la distancia de separación entre el centro de los ojos, el límite izquierdo para la región de búsqueda corresponde al centro horizontal del ojo izquierdo menos 20 píxeles.

d. De manera similar al punto anterior, el límite derecho de la región de búsqueda es el centro horizontal del ojo derecho más 20 píxeles.

Un ejemplo de la región obtenida por el proceso anterior es desplegado en la Figura 3.29, en la cual la distancia  $D$  se aprecia dibujada en color rojo, y la región de búsqueda para la boca y la nariz se encuentra enmarcada en azul.

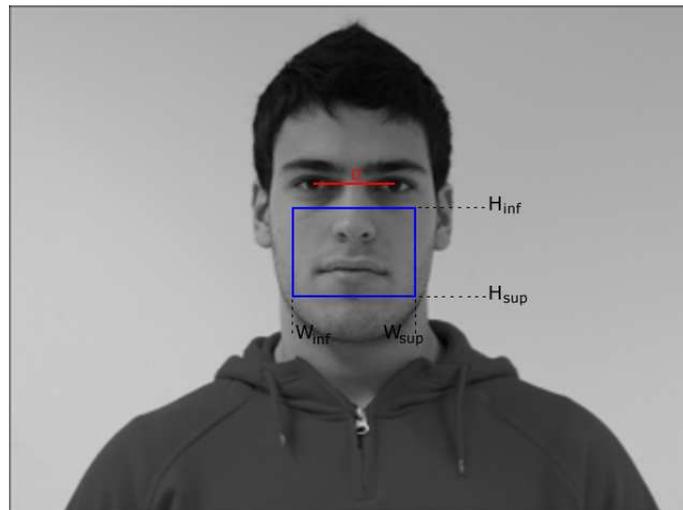


**Figura 3.29:** Región de búsqueda de la nariz y boca.

### 3.2.1.6. Proyección Vertical Sobre el Área de la Nariz y Boca

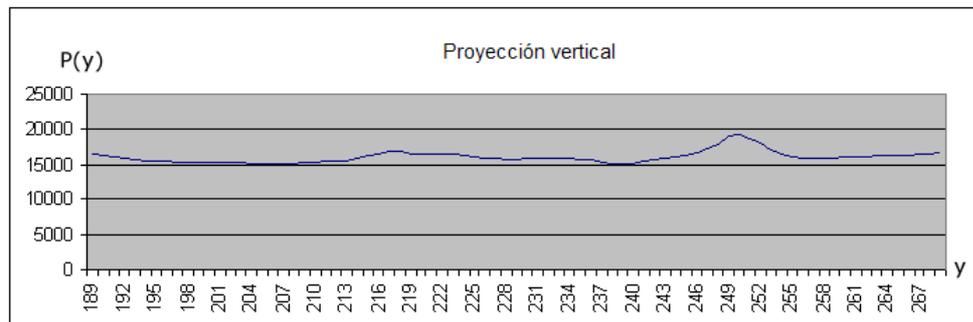
Para obtener la posición en  $y$  de la nariz y la boca se aplica la proyección vertical del histograma modificada (ecuación (3.5)) a la región de búsqueda.

Donde  $P(y)$  es la densidad de niveles de gris o proyección vertical del histograma en la posición  $y$  de la imagen,  $I$  es el valor de la intensidad de la imagen en niveles de gris en la posición  $(x, y)$ ,  $H_{inf}$  y  $H_{sup}$  son los límites de altura inferior y superior respectivamente,  $W_{inf}$  y  $W_{sup}$  son los límites del ancho de la región de búsqueda en la imagen. Los valores  $H_{inf}$ ,  $H_{sup}$ ,  $W_{inf}$  y  $W_{sup}$  se especifican en la Figura 3.30.



**Figura 3.30:** Límites de la región de búsqueda para la aplicación de la proyección vertical de la ecuación (3.5).

En la Figura 3.31 se muestra la gráfica obtenida por la aplicación de la proyección vertical en la región de búsqueda de la Figura 3.30.



**Figura 3.31:** Gráfica de la proyección vertical en la región de búsqueda de la Figura 3.30.

Los máximos de la gráfica 3.31 muestran la posición de la nariz, la boca, y otras características faciales contenidas en el área de búsqueda como el bigote, la barba, lunares o manchas, entre otras; por lo que para obtener los picos que representan la nariz y la boca se realiza un análisis sobre la gráfica que enseguida se describirá:

Se almacena a  $P(y)$  de manera discreta en un arreglo unidimensional denotado por **salida\_bn**[ $i$ ], donde cada elemento  $i$  de **salida\_bn** contiene el valor de intensidad de  $P(y)$ ,

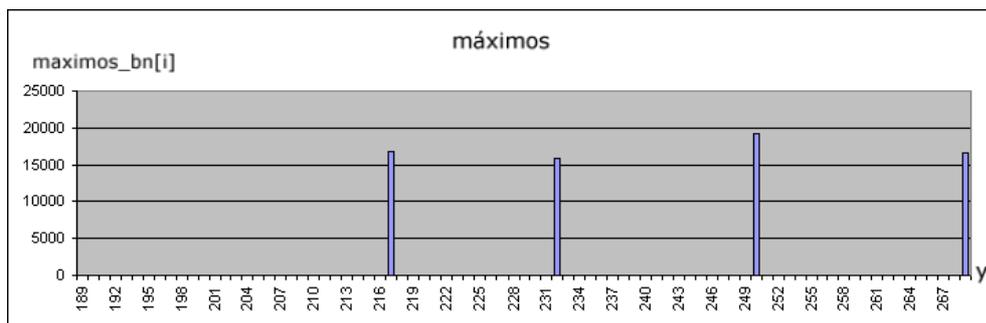
el tamaño de este arreglo depende del alto de la región de búsqueda, es decir, es igual a  $H_{sup} - H_{inf}$ .

Se hace un recorrido sobre **salida\_bn** y se extraen los máximos locales contenidos en él, considerando al elemento  $i$  de este arreglo un máximo si cumple con la siguiente regla:

$$\mathbf{salida\_bn}[i] \geq \mathbf{salida\_bn}[i - j] \quad \text{donde } j \text{ está en el rango } [-5,5] \quad (3.11)$$

Para  $i - j \geq 0$  y  $i + j < \text{Tamaño del arreglo } \mathbf{salida\_bn}$

Cada elemento máximo detectado es almacenado en un nuevo arreglo llamado **maximos\_bn** de igual tamaño que **salida\_bn**, y que contiene para cada elemento diferente a un máximo un valor de cero. El arreglo **maximos\_bn** derivado de la gráfica en la Figura 3.31 es graficado en la Figura 3.32.



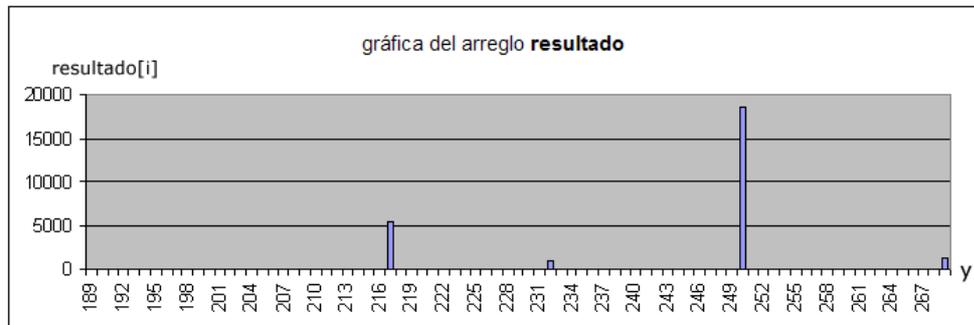
**Figura 3.32:** Gráfica resultante de la aplicación de la regla (3.11) a la gráfica de la Figura 3.31.

Para la determinación de la posición en  $y$  de la nariz y la boca es necesaria la construcción de un nuevo arreglo denotado por **resultado**, el cual está basado en los valores de **maximos\_bn** y **salida\_bn**. La asignación del valor para cada elemento  $i$  del arreglo **resultado** se realiza de acuerdo con la ecuación (3.12). El tamaño del arreglo **resultado** es igual que el tamaño de los arreglos **maximos\_bn** y **salida\_bn**.

$$\text{resultado}[i] = \begin{cases} 0 & \text{Si } \text{maximos\_bn}[i] \text{ no es máximo local} \\ & \text{es decir, contiene un valor igual a cero.} \\ \sum_{j=-5}^5 \text{maximos\_bn}[i] - \text{salida\_bn}[i + j] & \text{Si } \text{maximos\_bn}[i] \text{ es máximo local} \\ & \text{es decir, contiene un valor diferente de cero.} \end{cases} \quad (3.12)$$

Para  $i + j \geq 0$  y  $i + j < \text{Tamaño del arreglo salida\_bn}$

Los valores del arreglo **resultado** enfatizan la forma de los máximos locales contenidos en **salida\_bn**, resaltando la información que se busca, debido a que la tonalidad oscura de la nariz y el centro de la boca constituyen superficies verticales pequeñas dentro de la imagen, en contraste los rasgos faciales como barba y bigote abarcan superficies amplias y sus máximos locales correspondientes contienen planos en su pico más alto generando valores pequeños en la creación de **resultado[i]**. La gráfica del arreglo **resultado** generada con las gráficas de las Figuras 3.31 y 3.32 se aprecia en la Figura 3.33.



**Figura 3.33:** Gráfica del arreglo **resultado** generado con ayuda de las gráficas de las Figuras 3.31 y 3.32.

De acuerdo a la gráfica de la Figura 3.33, la posiciones verticales para la nariz y la boca se determinan a partir del valor de intensidad más alto de entre los elementos del arreglo **resultado**, y denotado como **elemento\_mayor**, considerando como boca a este elemento, y como nariz al elemento de mayor intensidad al resto de los elementos del arreglo, buscado desde

el comienzo del arreglo hasta 20 elementos anteriores al **elemento\_mayor**, estos 20 elementos son la separación mínima entre la boca y la nariz (distancia obtenida empíricamente). Si no se encuentran elementos anteriores al **elemento\_mayor**, se realiza un realce mayor de la información dentro de la región de búsqueda mediante una nueva especificación del arreglo **resultado** empleando la siguiente ecuación:

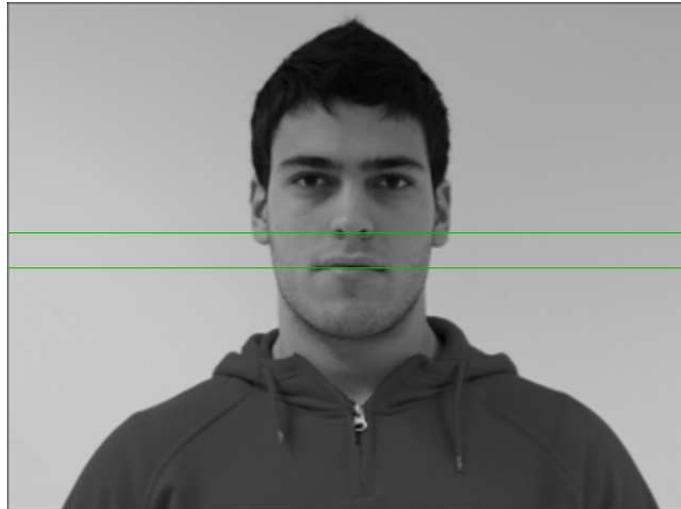
$$\text{resultado}[i] = \begin{cases} 0 & \text{Si } \text{maximos\_bn}[i] \text{ no es máximo local} \\ & \text{es decir, contiene un valor igual a cero.} \\ \sum_{j=-3}^3 \text{maximos\_bn}[i] - \text{salida\_bn}[i + j] & \text{Si } \text{maximos\_bn}[i] \text{ es máximo local} \\ & \text{es decir, contiene un valor diferente de cero.} \\ & \text{Para } i + j \geq 0 \text{ y } i + j < \text{Tamaño del arreglo } \text{salida\_bn} \end{cases} \quad (3.13)$$

Las posiciones verticales para la nariz y la boca se determinan a partir del **elemento\_mayor**, si el **elemento\_mayor** se encuentra dentro de la primera mitad del nuevo arreglo **resultado**, entonces la posición en y del **elemento\_mayor** corresponde a la posición vertical de la nariz, y la posición vertical de la boca es la posición en y del elemento con mayor intensidad de entre los elementos restantes, buscado a partir de 20 elementos posteriores al **elemento\_mayor** y con dirección al final del arreglo. De otra manera, si el **elemento\_mayor** se encuentra dentro de la segunda mitad del nuevo arreglo **resultado**, entonces la posición en y del **elemento\_mayor** corresponde a la posición vertical de la boca, y la posición vertical para la nariz es la posición en y del elemento con mayor intensidad de entre los elementos restantes, buscado a partir de 20 elementos anteriores al **elemento\_mayor** y con dirección al inicio del arreglo.

### Posición de la nariz y posición de la boca.

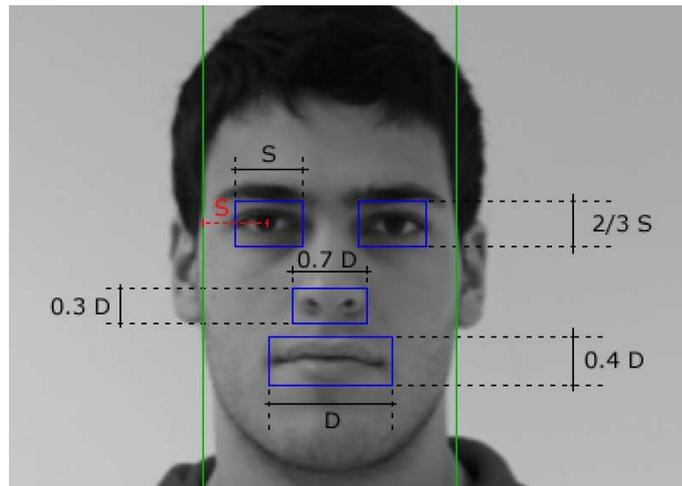
Como resultado del proceso anterior se obtienen las posiciones verticales para la nariz y para

la boca, un ejemplo de este resultado se aprecia en la Figura 3.34, en la cual las posiciones en y de la nariz y de la boca son indicadas por líneas en color verde.



**Figura 3.34:** Posiciones de la nariz y de la boca en líneas color verde.

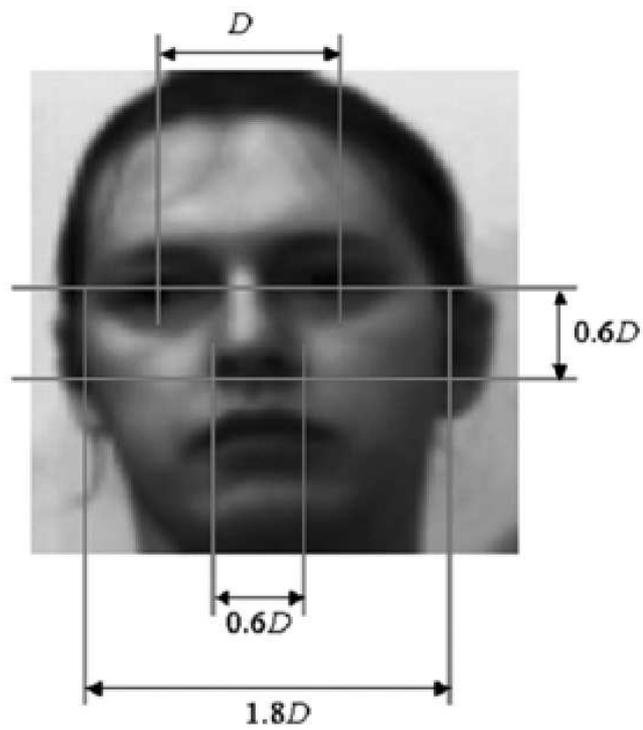
Como último paso dentro de la localización de los rasgos faciales se enmarcan por medio de rectángulos a los ojos, nariz y boca previamente encontrados, el tamaño de los rectángulos que envuelven a los rasgos faciales está relacionado con la medida de proporción dada por el modelo geométrico, la cual como se mencionó anteriormente es la distancia de separación entre la posición media de los ojos y es denotada por la letra  $D$ , por lo tanto, las medidas del ancho y alto del recuadro que envuelve a la nariz son de  $0.7 D \times 0.3 D$ , las medidas para el recuadro que envuelve a la boca es de  $D \times 0.4 D$  (ver Figura 3.35). Para el caso de los rectángulos que envuelven a cada uno de los ojos ambos tendrán el mismo ancho y alto, sin embargo, la medida de proporción dada por el modelo geométrico no es eficaz en este caso, por lo tanto, se requiere primero calcular el ancho de cada recuadro, se calcula la distancia de separación del centro del ojo izquierdo hacia el límite izquierdo del rostro, y la distancia de separación del ojo derecho hacia el límite derecho del rostro; finalmente se considera la distancia mayor de entre éstas como la nueva medida de proporción para el ancho y alto del recuadro, y es denotada por la letra  $S$  (ver Figura 3.35).



**Figura 3.35:** Medidas para los recuadros que envuelven a los rasgos faciales.

Para calcular la distancia de proporción  $S$ , se requiere especificar previamente las posiciones en  $x$  del límite izquierdo y del límite derecho del rostro, es decir, calcular el ancho del rostro. Una primera aproximación al ancho del rostro está basada en el modelo geométrico descrito por [29], en el cual el ancho del rostro es de  $1.8 D$  como se aprecia en la Figura 3.36. Como se mencionó antes la medida de proporción del modelo geométrico empleado para este proyecto no es una herramienta precisa para obtener las posiciones verticales de la boca y la nariz, por lo tanto al conseguir las posiciones exactas de la boca y la nariz dentro de la imagen, se adquiere para este caso de la localización del ancho del rostro un nuevo valor para  $D$  basado en la separación exacta de la nariz y la boca, este nuevo valor es denotado por  $D'$ , y calculado a partir de la distancia vertical entre la nariz y el centro de la boca que en el modelo geométrico es igual a  $0.4 D$ , entonces el valor para  $D'$  se calcula sustituyendo los valores exactos ya determinados como se aprecia en las ecuaciones (3.14) y (3.15), por lo tanto la primera aproximación del ancho del rostro está dada por  $1.8 D'$ .

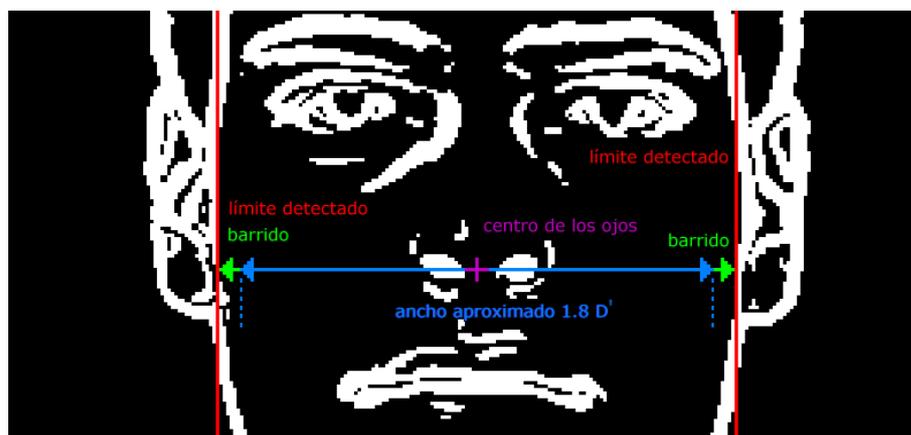
$$\text{Posición\_vertical\_boca} - \text{Posición\_vertical\_nariz} = 0.4D' \quad (3.14)$$



**Figura 3.36:** Modelo geométrico del rostro descrito en [29].

$$D' = \frac{\text{Posición\_vertical\_boca} - \text{Posición\_vertical\_nariz}}{0.4} \quad (3.15)$$

Con la primera aproximación al ancho del rostro se determina el ancho real del rostro, el cual puede ser mayor o igual al ancho aproximado, el ancho real es entonces la distancia de separación entre el límite izquierdo y el límite derecho del rostro. Para especificar las posiciones en  $x$  del límite izquierdo y del límite derecho del rostro se realizan dos barridos horizontales, uno con origen en el extremo izquierdo del ancho aproximado del rostro y con dirección hacia la izquierda, éste para localizar el límite izquierdo del rostro y otro con origen en el extremo derecho del ancho aproximado del rostro y con dirección hacia la derecha, éste para localizar el límite derecho, tomando como altura de ambos barridos la posición en  $y$  de la nariz, debido a que a esta altura el rostro se encuentra menos ocluido por el cabello o la barba; el barrido se realiza dentro de la imagen binaria (obtenida como se describe en la sección 3.2.1.1, ver Figura 3.8) como se aprecia en la Figura 3.37, en la cual tanto el límite izquierdo y el límite derecho son detectados sobre el borde del rostro en color blanco, empleando para esto una vecindad vertical de 17 píxeles sobre la altura de la nariz, entonces la detección del borde ocurre cuando 3 o más píxeles de los 17 analizados son borde. Ambos límites son dibujados por líneas en color rojo, el origen y fin del barrido se aprecian en color verde.



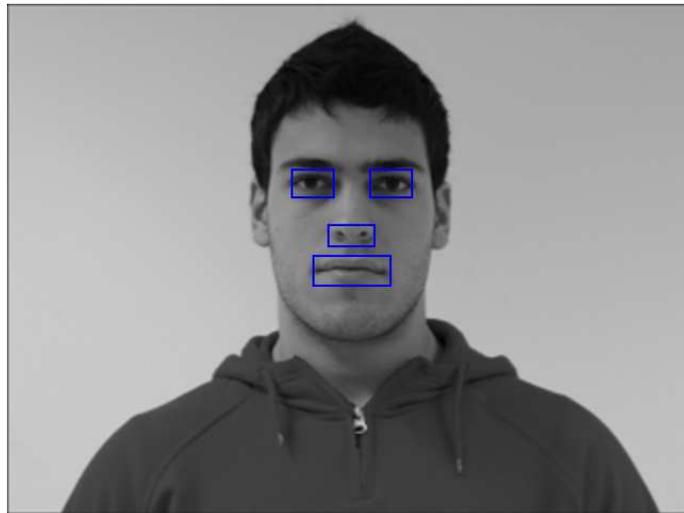
**Figura 3.37:** Proceso para la localización del ancho del rostro.

Puede ocurrir que la primera aproximación al ancho del rostro se encuentre fuera de los

márgenes de la imagen, ésto es causado por una mala localización del centro de los ojos en el proceso descrito en la sección 3.2.1.4, también puede ocurrir que durante el proceso de localización del ancho del rostro, éste se encuentre fuera de los límites de la imagen.

Finalmente, con la obtención de las posiciones en  $x$  de los límites izquierdo y derecho del rostro la distancia de proporción  $S$  es determinada como se mencionó en párrafos anteriores. Con base en esta medida de proporción se especifica que el ancho y el alto de los recuadros que envuelven a cada uno de los ojos es de  $S \times \frac{2}{3} S$  como se aprecia en la Figura 3.35.

El resultado final de la localización de los rasgos faciales se muestra en la Figura 3.38, donde se dibujan los rectángulos que envuelven a los rasgos faciales en color azul.



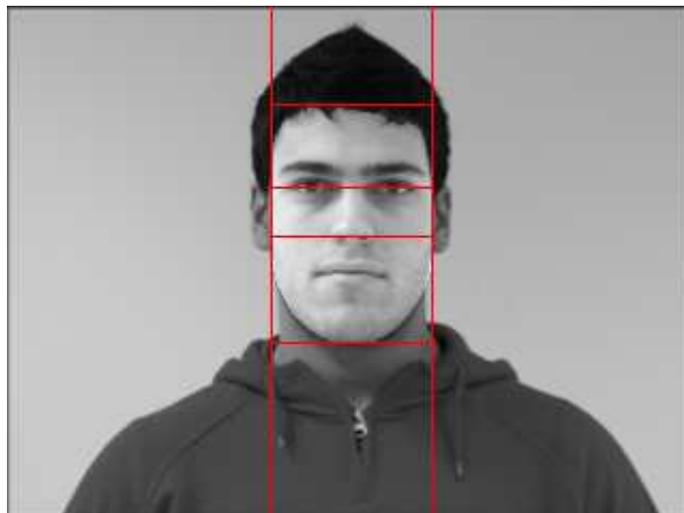
**Figura 3.38:** Resultado de la ubicación de los rasgos faciales.

### 3.2.2. Localización del Contorno del Rostro

La extracción del contorno del rostro consiste en localizar de manera automática los límites de la silueta del rostro contenido en la imagen, la segmentación del rostro se realizó mediante la aplicación de un modelo de forma activa llamado contorno activo (descrito en la sección 2.9).

### 3.2.2.1. Inicialización del Contorno Activo

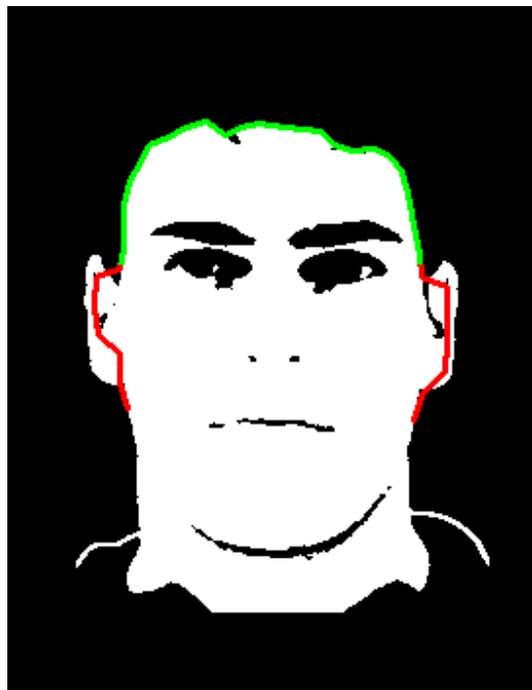
Debido a que el resultado obtenido por el contorno activo es sensible a su inicialización, se requiere ajustar la posición inicial de los *snaxels* cerca de la silueta del rostro, para esto se hizo una división del rostro en tres partes, como se aprecia en la Figura 3.39, de esta manera se tiene una inicialización para cada una de las partes: Una inicialización superior para posicionar a los *snaxels* sobre la silueta de la frente, la cual termina a la altura del centro vertical de los ojos; una inicialización media para posicionar a los *snaxels* sobre los costados del rostro, que abarca desde la altura del centro vertical de los ojos hasta la posición vertical de la nariz; y una inicialización inferior para posicionar a los *snaxels* sobre el contorno perteneciente a la quijada, que inicia desde la posición vertical de la nariz hasta el límite vertical de la barbilla.



**Figura 3.39:** División del rostro en tres partes.

### A. Inicialización superior

Esta inicialización consiste en posicionar a los *sna-xels* sobre el contorno de la frente, abarcando hasta el centro vertical de los ojos. Para lograr esto se retoma que la ubicación aproximada del contorno del rostro (previamente determinada en la sección 3.2.1.3) está construida por un conjunto de puntos, por lo tanto se considera a cada uno de estos puntos un *sna-xel* perteneciente al contorno de la frente, si la posición en *y* del punto es inferior al centro vertical de los ojos, como se aprecia en la Figura 3.40 por la línea en color verde.



**Figura 3.40:** Contorno de la frente.

Sin embargo, como se ha mencionado previamente el resultado de la detección de color de piel puede producir falsos positivos (sección 2.8), haciendo que la ubicación aproximada del rostro obtenida por esta información no permita un buen acercamiento al contorno de la frente. Por lo tanto, para extraer un mejor contorno de la frente se realiza una eliminación de falsos positivos de piel en la imagen binaria derivada de la detección de piel, y solo en la región que va desde el límite superior de la imagen hasta el centro vertical de los ojos y de

manera horizontal está región está delimitada por el ancho del rostro. Posteriormente se hace una reducción al contorno del rostro sin falsos positivos, es decir, cada punto del contorno de la frente es desplazado al centro aproximado del rostro y detenido sobre el borde de color de piel sin falsos positivos, un ejemplo de los falsos positivos se aprecia en la Figura 3.41.



**Figura 3.41:** Resultado de la detección de piel con falsos positivos.

#### *A.1. Eliminación de falsos positivos*

La eliminación de falsos positivos consiste en especificar qué píxeles de la imagen binaria procedente de la detección de piel pertenecen al área del rostro, tomando como referencia un conjunto de píxeles que se asume sí pertenecen al rostro, esto se realiza dentro del área de búsqueda, que va desde el límite superior de la imagen hasta el centro vertical de los ojos y de manera horizontal abarca desde el límite izquierdo del rostro hasta el límite derecho del rostro. Para especificar estos píxeles se recorre el área de búsqueda dos veces, una en dirección vertical ascendente y la otra en dirección vertical descendente:

- **Dirección vertical descendente**, comienza a la altura del centro de los ojos y termina en el límite superior de la imagen, durante este recorrido vertical los píxeles

pertenecientes al rostro se buscan en dos barridos horizontales consecutivos, de derecha a izquierda tomando como inicio el límite derecho del rostro de esta región de búsqueda, y de izquierda a derecha tomando como inicio el límite izquierdo del rostro.

- **Dirección vertical ascendente**, comienza en el límite superior de la imagen y termina en el centro vertical de los ojos, durante este recorrido vertical los píxeles pertenecientes al rostro se buscan en dos barridos horizontales consecutivos, de derecha a izquierda tomando como inicio el límite derecho del rostro de esta región de búsqueda, y de izquierda a derecha tomando como inicio el límite izquierdo del rostro.

Durante el barrido descendente se toma como referencia de piel que sí pertenece al rostro a los píxeles cuya posición en  $y$  se encuentren en la posición vertical inicial del recorrido, como se aprecia en la Figura 3.42(a) con la línea de color violeta, en la cual el punto inicial del barrido es la posición vertical del centro de los ojos. La información obtenida por el barrido descendente es la posición vertical del centro de los ojos. La información obtenida por el barrido descendente es considerada como referencia de piel para el barrido ascendente. El barrido en dirección ascendente se aprecia en la Figura 3.42(b). Para ambos barridos se considera que el píxel dentro del espacio de búsqueda es piel perteneciente al rostro si al menos uno de los ochos vecinos es piel que sí pertenece al rostro.



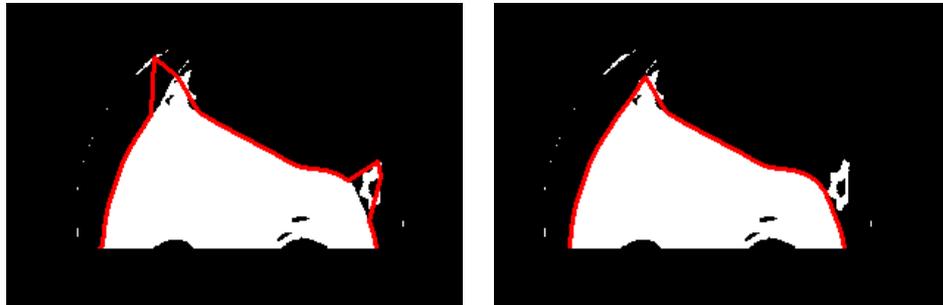
(a) recorrido en dirección vertical descendente

(b) recorrido en dirección vertical ascendente

**Figura 3.42:** Eliminación de falsos positivos.

Posteriormente el ajuste al contorno de la frente se realiza, desplazando los puntos que conforman la ubicación aproximada de la frente hacia la imagen binaria de la detección de piel sin falsos positivos, dando como resultado los *snaxels* posicionados sobre el contorno de la frente. El resultado de esta inicialización se aprecia en la Figura 3.43(b), al eliminar los falsos positivos que aparecen en la Figura 3.43(a).

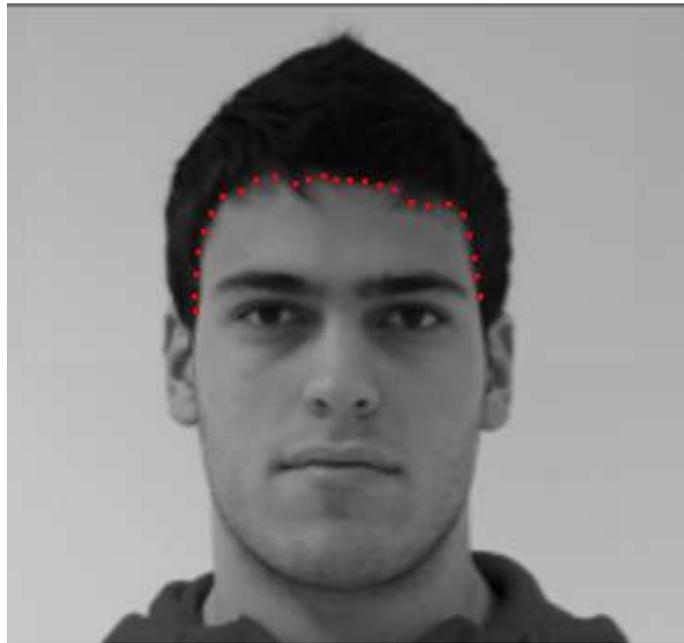
Por otra parte en la Figura 3.44 se muestra la inicialización superior de los *snaxels* para la imagen de la Figura 3.40.



(a) contorno de la frente con falsos positivos

(b) contorno de la frente sin falsos positivos

**Figura 3.43:** Reducción al contorno de la frente sin falsos positivos.



**Figura 3.44:** Inicialización de los *snaxels* sobre el contorno de la frente aplicada a la imagen de la Figura 3.40.

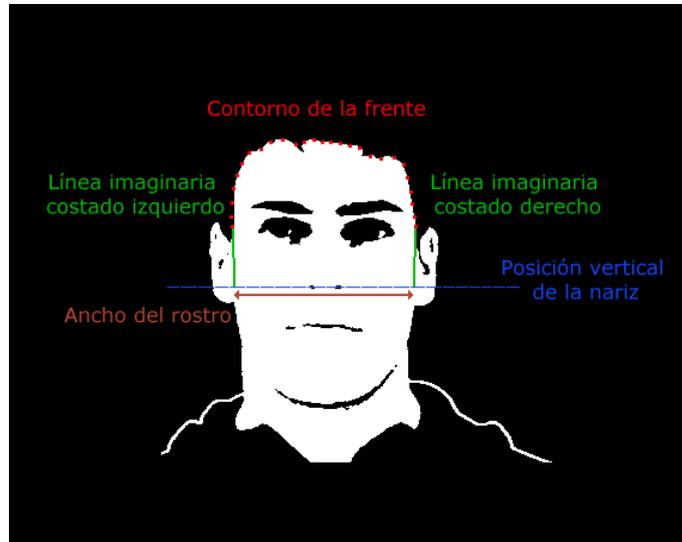
### **B. Inicialización media**

Esta inicialización consiste en posicionar a los *snaxels* sobre el costado izquierdo del rostro y sobre el costado derecho del rostro, a partir de la posición vertical del centro de los ojos y hasta la posición vertical de la nariz.

Para colocar los *snaxels* sobre el costado izquierdo del rostro se traza una línea recta imaginaria, que va desde el *snaxel* del extremo izquierdo del contorno de la frente hasta el punto con coordenada en  $x$  igual al límite izquierdo del rostro y con coordenada en  $y$  igual a la posición vertical de la nariz, como se aprecia en la Figura 3.45 con la línea en color verde. Sobre esta línea imaginaria se colocan los *snaxels* con una separación equidistante igual a 15.

De modo similar, para colocar los *snaxels* sobre el costado derecho del rostro se traza una línea recta imaginaria, que va desde el *snaxel* del extremo derecho del contorno de la frente hasta el punto con coordenada en  $x$  igual al límite derecho del rostro y con coordenada en  $y$  igual a la posición vertical de la nariz, como se aprecia en la Figura 3.45 con la línea en color

verde. Sobre esta línea imaginaria se colocarán los *snaxels* con una separación equidistante igual a 15.



**Figura 3.45:** Inicialización media de los *snaxels* sobre los costados del rostro.

El resultado de la inicialización superior y media se dibujan en la Figura 3.46.

### C. Inicialización inferior

En la inicialización inferior se colocan los *snaxels* sobre el contorno perteneciente a la quijada, que inicia desde la posición vertical de la nariz hasta el límite vertical de la barbilla (valor que se define posteriormente). Para lograr esto se requiere generar la forma de la quijada sobre la cual recae la posición de los *snaxels*, tal forma es dibujada por medio de una elipse con eje menor igual al ancho del rostro y con eje mayor igual al doble de la distancia de separación entre la posición vertical de la nariz y la posición vertical de la barbilla.

Para obtener la posición vertical de la barbilla se empleó una proyección vertical, ya antes utilizada, y al ser la forma de la barbilla no claramente distinguible dentro de la imagen en niveles de gris, por efectos de iluminación o sombras, se aplicó esta proyección sobre el resultado del realce de bordes de la imagen en niveles de gris, analizando de esta manera los



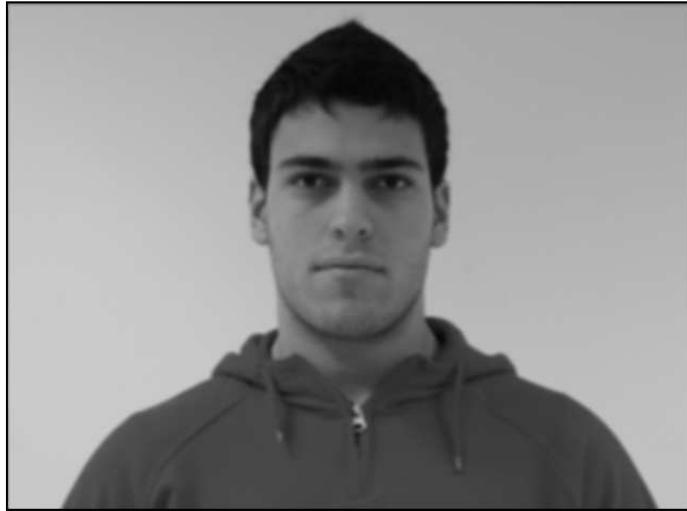
**Figura 3.46:** Inicialización superior y media de los *snaxels* aplicada a la imagen de la Figura 3.40.

bordes contenidos en el espacio de la barbilla.

### ***C.1. Realce de bordes sobre la imagen en niveles de gris***

Antes de aplicar el operador de Sobel para la detección de los bordes del rostro se alisa la imagen de entrada en niveles de gris, con el propósito de reducir el ruido sobre los bordes. El alisado o smoothing se realiza por medio del filtro Gaussiano descrito en la sección 2.6.2, aplicando la propiedad de separabilidad de las funciones gaussianas, es decir, se realizan dos barridos a la imagen, en el primero se convoluciona la imagen en una dimensión de forma horizontal con el filtro Gaussiano y en el segundo barrido se convoluciona el resultado de la convolución anterior con el filtro Gaussiano de modo ortogonal, el resultado de este alisado puede notarse en la Figura 3.47, en el cual se aplicó una ventana lineal de 5 elementos y un  $\sigma^2 = 4$ .

Después de alisar la imagen se aplica la detección de bordes, empleando para esto las ecuaciones (2.24) y (2.25) del operador de Sobel descrito en la sección 2.7.3.1, el resultado



**Figura 3.47:** Alisado Gaussiano aplicado a la imagen de entrada en niveles de gris.

de este realce de bordes se muestra en la Figura 3.48 donde  $c$  tomó un valor igual a 8, y se empleó una máscara de convolución de  $3 \times 3$ .



**Figura 3.48:** Resultado de aplicar el operador de Sobel a la imagen de la Figura 3.47.

### ***C.2. Proyección vertical con forma elíptica***

Como anteriormente se ha utilizado, la proyección vertical se obtiene aplicando la ecuación (2.7) mencionada en la sección 2.4, sin embargo por la forma elíptica de los bordes generados por la barbilla, el recorrido de esta proyección se modifica, el cual cambia de ser un barrido

lineal a un barrido con forma elíptica, por consiguiente la ecuación para la proyección vertical modificada es la siguiente:

$$P(y) = \sum_{k'=0}^{360} \sum_{j=0}^1 I(x'(k'), y'(k') - j) \quad \text{Donde : } H_{inf} \leq y \leq H_{sup} \quad (3.16)$$

$$x'(k') = [\cos(k' * \frac{\pi}{360}) * \frac{eje\_menor}{2}] + origen\_x$$

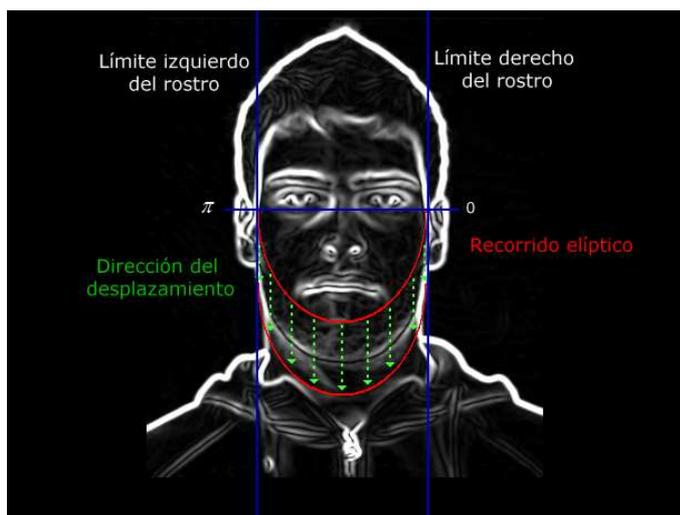
$$y'(k') = [\sin(k' * \frac{\pi}{360}) * \frac{eje\_mayor}{2}] + origen\_y$$

*origen\_x* = el punto medio de la distancia de separación entre los ojos.

$$origen\_y = y - 100$$

$$I(x'(k'), y'(k') - j) = \begin{cases} 0 & \text{Si } x'(k') \text{ está fuera del rango } [W_{inf}, W_{sup}] \\ & \text{ó si } y'(k') - j \text{ está fuera del rango } [H_{inf}, H_{sup}] \\ Sobel(x'(k'), y'(k') - j) & \text{Si } W_{inf} \leq x'(k') \leq W_{sup} \\ & \text{y si } H_{inf} \leq y'(k') - j \leq H_{sup} \end{cases}$$

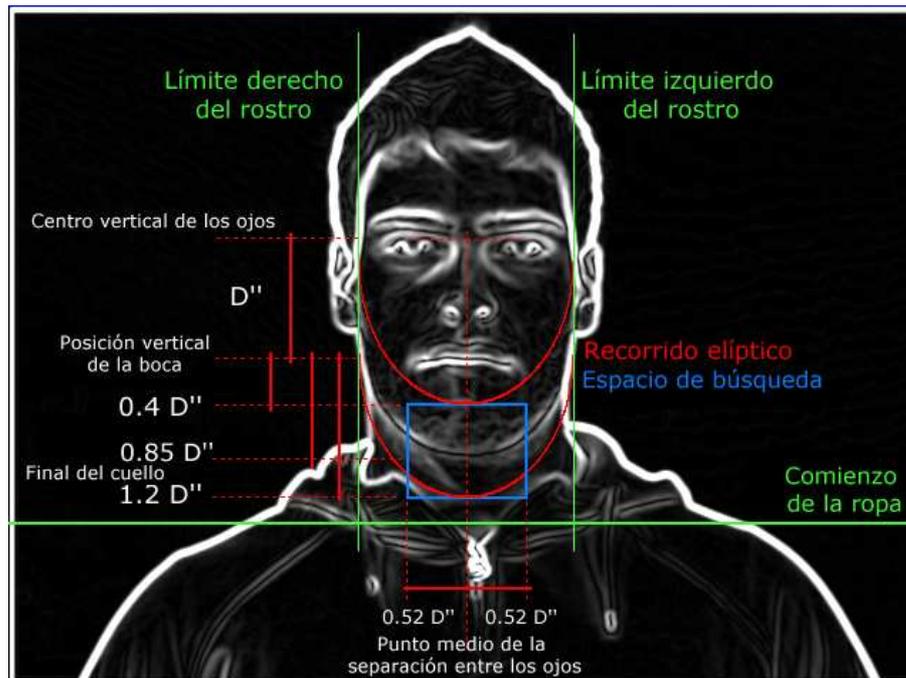
Donde  $P(y)$  es la proyección vertical con forma elíptica calculada en la posición  $y$  de la imagen;  $I$  es el valor de intensidad resultante del realce de bordes sobre la imagen suavizada en la posición  $(x'(k'), y'(k') - j)$ , el cual toma un valor igual a cero si la posición analizada está fuera de los límites de el espacio de búsqueda; el valor de  $j$  indica una vecindad vertical de 1 píxel sobre el punto analizado y la sumatoria sobre tal vecindad permite incrementar el valor de los bordes;  $H_{inf}$  y  $H_{sup}$  son los límites verticales superior e inferior respectivamente del espacio de búsqueda en el cual debe estar contenida la posición vertical de la barbilla,  $W_{inf}$  y  $W_{sup}$  son el límite izquierdo y el límite derecho de este espacio de búsqueda, el cálculo de los límites de este espacio se define más adelante;  $x'$  y  $y'$  son las coordenadas sobre una elipse como se aprecia en la Figura 3.49 por la línea en color rojo, la elipse que se describe tiene un eje menor igual al ancho del rostro, y un eje mayor constante igual a 200;  $k'$  es un factor que multiplicado por  $\frac{\pi}{360}$  traza la media elipse considerada para el barrido, como se muestra en la Figura 3.49. El desplazamiento del barrido está descrito por la variable *origen\_y*, la cual es igual a la posición analizada menos la mitad del eje mayor, es decir,  $y - 100$ .



**Figura 3.49:** Proyección vertical con forma elíptica.

Antes de aplicar la proyección vertical con forma elíptica, se definen los límites del espacio de búsqueda asegurándose que el final de la barbilla de la persona se encuentre contenido en éste, y evitando lo más posible el ruido proveniente de la ropa, por lo tanto para establecer el espacio de búsqueda se generó una medida de proporción con la cual se define al límite  $H_{inf}$  sobre la posición vertical de la barbilla a encontrar y al límite  $H_{sup}$  como la altura al final del cuello (ó si éste se encuentra ocluido,  $H_{sup}$  se define como la posición en y donde se detecta el inicio de la ropa) esta medida de proporción es denotada por  $D''$  y es igual a la distancia de separación entre la posición vertical del centro de los ojos y la posición en y de la boca, como se aprecia en la Figura 3.50.

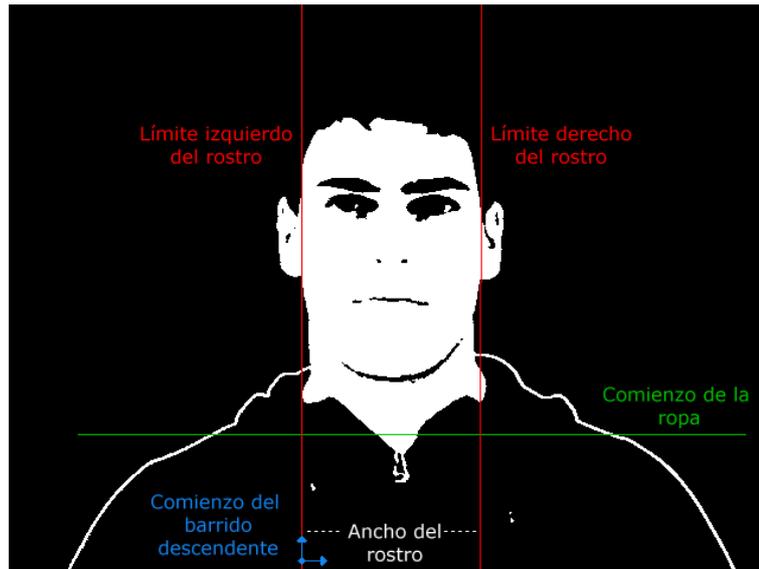
Haciendo uso de la medida de proporción  $D''$ ,  $H_{inf}$  se sitúa a  $0.4 D''$  desde la posición vertical de la boca y  $H_{sup}$  se sitúa a  $0.85 D''$  desde la posición vertical de la boca, estos valores fueron obtenidos empíricamente. Hay dos situaciones por las cuales el límite  $H_{sup}$  puede cambiar su posición, la primera es causada por la existencia de barba ó por sombras en el cuello producidas por la iluminación sobre la quijada, esto indica que el final de la barbilla está por debajo de  $H_{sup}$ , para evitar esto se realiza la cuantificación de los píxeles detectados como no piel en la posición vertical de  $0.85 D''$ , y entre los límites  $W_{inf}$  y  $W_{sup}$ , el



**Figura 3.50:** Región de búsqueda para la proyección vertical donde no hay oclusión del cuello.

porcentaje de píxeles no pertenecientes a la piel debe ser menor o igual al 5%, de lo contrario se extenderá la región de búsqueda aumentando el límite  $H_{sup}$  a  $1.2 D''$ , en la Figura 3.50 se aprecia este espacio de búsqueda por el rectángulo en color azul; la segunda situación para cambiar la posición de  $H_{sup}$  se produce cuando el cuello se encuentra ocluido por la ropa, es decir, cuando el límite  $H_{sup}$  es mayor a la posición en  $y$  del comienzo de la ropa, dando como resultado una medición errónea sobre la región de búsqueda, para resolver esto el límite  $H_{sup}$  cambia su posición vertical por la posición en  $y$  del comienzo de la ropa, esta posición en  $y$  se determina realizando un barrido desde el límite inferior de la imagen hasta que el inicio de la ropa sea detectado, ó hasta el límite superior de la imagen (situación que no se espera que ocurra, considerando que hay una persona en la imagen), tomando como límite izquierdo del barrido al límite izquierdo del rostro y como límite derecho al límite derecho del rostro. Durante este barrido se busca el color de la piel dentro de la imagen binaria resultante de la detección de piel, contabilizando en cada ciclo del barrido el número

de píxeles pertenecientes a la piel, de esta manera la posición en y del comienzo de la ropa se obtiene cuando el 20% de los píxeles analizados son color de piel, como se aprecia en la Figura 3.51 por la línea en color verde. Un ejemplo donde el cuello se encuentra ocluido aparece en la Figura 3.52 donde el espacio de búsqueda está dibujado por el recuadro en color rojo.

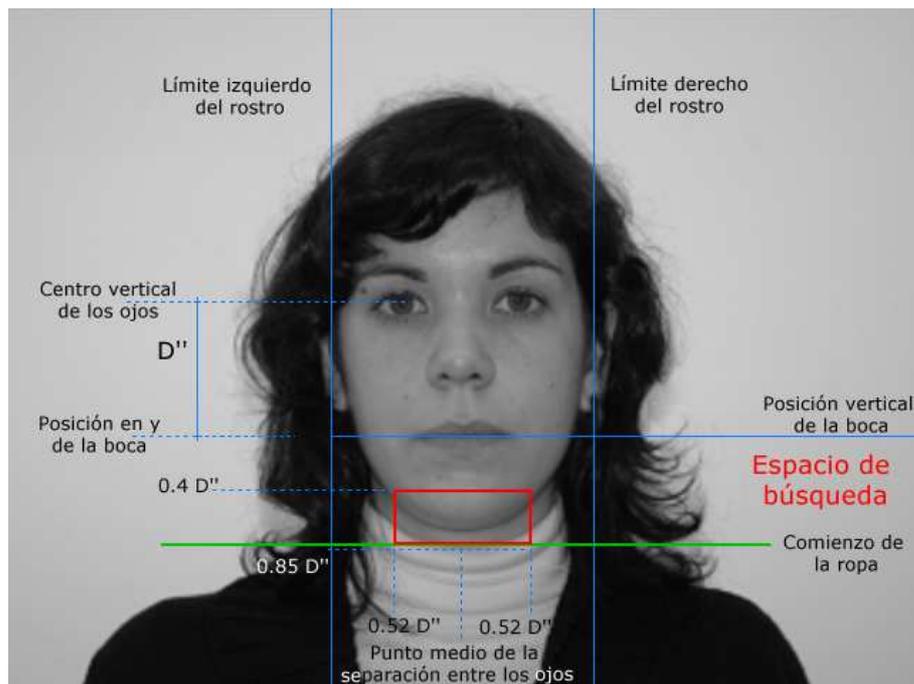


**Figura 3.51:** Detección del comienzo de la ropa.

$W_{inf}$  y  $W_{sup}$  se calculan con la misma medida de proporción  $D''$ , estableciendo para ambos límites un valor igual a  $0.52 D''$  a partir del punto medio de la distancia de separación entre los ojos (ver Figura 3.50 y 3.52).

Una vez establecido el espacio de búsqueda, se aplica la proyección vertical con forma elíptica sobre éste, y a partir de la gráfica resultante se obtendrá la posición en y más cercana a la barbilla. La gráfica resultante de la proyección vertical con forma elíptica aplicada al espacio de búsqueda de la Figura 3.50 se gráfica en la Figura 3.53.

La gráfica resultante contiene la posición vertical de la barbilla, la cual puede estar dada,



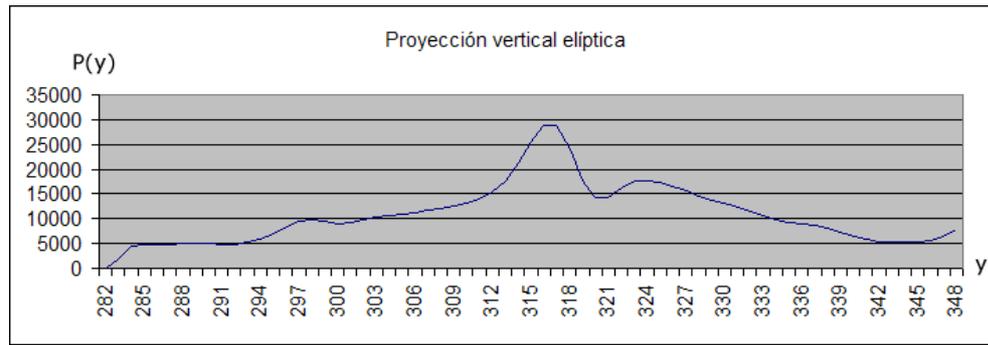
**Figura 3.52:** Región de búsqueda para la proyección vertical con oclusión del cuello.

según los bordes en el espacio de búsqueda, por un máximo global, ó por un mínimo local contenido entre el máximo global y un máximo local, como se aprecia en la Figura 3.54.

Entonces, para determinar si la posición en y de la barbilla corresponde a un máximo global ó a un mínimo local se realiza el siguiente análisis, que consiste de los siguientes pasos:

### 1. Localización de los máximos locales y el máximo global

Para la extracción de los máximos locales, se almacena a  $P(y)$  de manera discreta en un arreglo unidimensional denotado por **salida\_ba**[ $i$ ], donde cada elemento  $i$  de **salida\_ba** contiene el valor de intensidad de  $P(y)$ , el tamaño de este arreglo depende del alto del espacio de búsqueda, es decir, el tamaño es igual a  $H_{sup} - H_{inf}$ . Sobre este arreglo se hace un recorrido en el cual se considera al elemento  $i$  de este arreglo un máximo si cumple con la siguiente regla:



**Figura 3.53:** Gráfica resultante de aplicar la proyección vertical con forma elíptica al espacio de búsqueda definido en la Figura 3.50.

$$\text{salida\_ba}[i] > \text{salida\_ba}[i + j] \quad \text{donde } j \text{ está en el rango } [-4, 4] \quad (3.17)$$

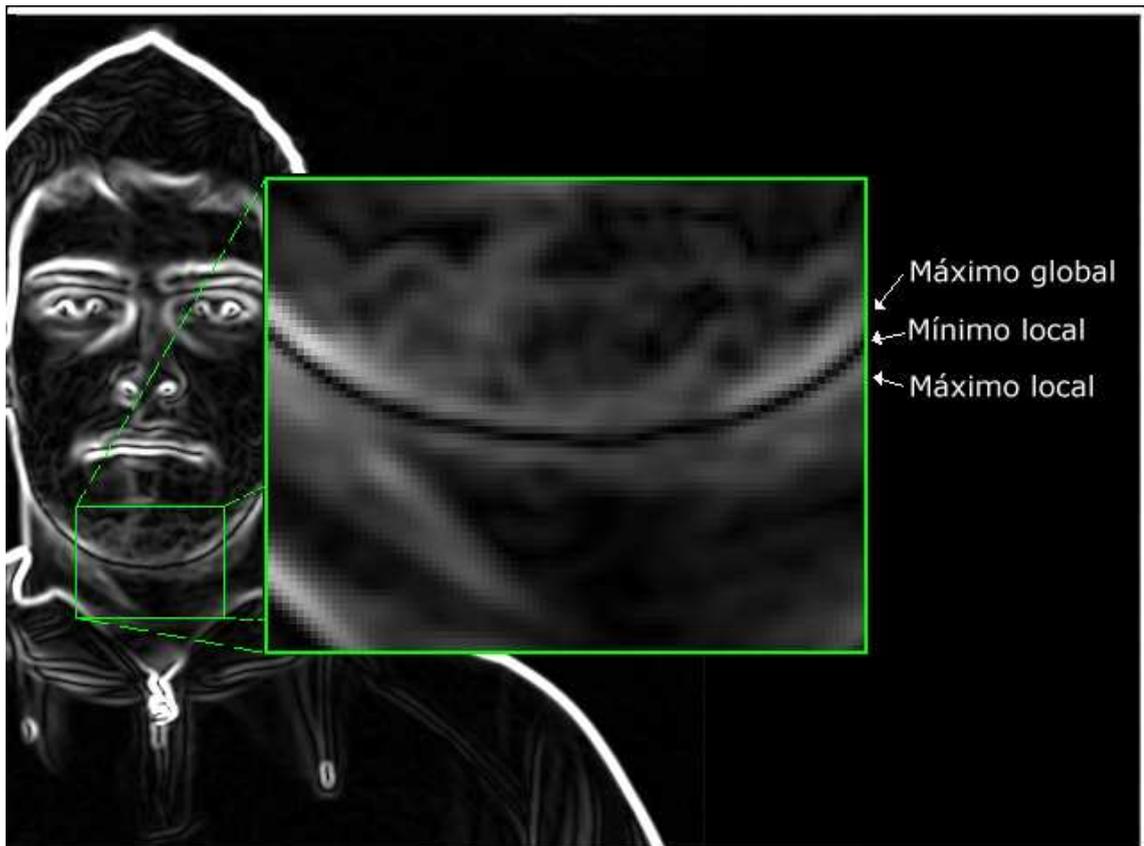
$$\text{y } 0 \leq i + j < (H_{sup} - H_{inf})$$

Cada elemento máximo detectado es almacenado en un nuevo arreglo llamado **maximos\_ba** de igual tamaño que **salida\_ba**, y que contiene para cada elemento diferente a un máximo un valor de cero. El arreglo **maximos\_ba** derivado de la gráfica 3.53 es visualizado en la Figura 3.55.

Después se determina qué elemento del arreglo **maximos\_ba** contiene el máximo global.

## 2. Eliminación de máximos locales.

Con el fin de reducir el número de máximos locales que intervienen en la determinación de la barbilla, se realiza la eliminación de algunos máximos locales: Se genera para cada máximo local, contenido en **maximos\_ba**, un nuevo valor dado por la suma de las diferencias de intensidad entre el máximo local y cada uno de los elementos cercanos dentro de **salida\_ba**, cuatro hacia la izquierda y cuatro hacia la derecha del máximo local. Entonces se aplica un umbral sobre estos nuevos valores, reduciendo a



**Figura 3.54:** Posible posición de la barbilla, máximo global ó mínimo local.

ceros a aquellos máximos locales cuyo nuevo valor esté por debajo del umbral, en otras palabras, se reducen a cero a aquellos máximos locales que no correspondan a un borde claramente definido.

Lo anterior se hace generando un nuevo arreglo basado en los arreglos **maximos\_ba** y **salida\_ba**, el nuevo arreglo es denotado por **resultado\_ba**, y la asignación del valor para cada elemento  $i$  de dicho arreglo se realiza de acuerdo con la ecuación (3.18). El tamaño de este arreglo es igual que el tamaño de los arreglos **maximos\_ba** y **salida\_ba**.

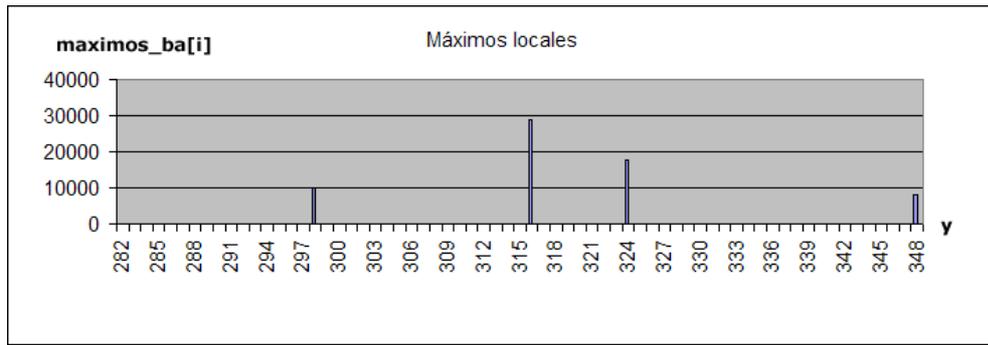


Figura 3.55: Gráfica del arreglo **maximos\_ba**.

$$\text{resultado\_ba}[i] = \begin{cases} 0 & \text{Si } \text{maximos\_ba}[i] \text{ no es máximo local, es decir,} \\ & \text{contiene un valor igual a cero.} \\ \sum_{j=-4}^4 \text{maximos\_ba}[i] - \text{salida\_ba}[i + j] & \text{Si } \text{maximos\_ba} \text{ es máximo local, es decir,} \\ & \text{contiene un valor diferente a cero.} \\ & \text{Para } i + j \geq 0 \text{ y } i + j < \text{Tamaño del arreglo } \text{salida\_ba}. \end{cases} \quad (3.18)$$

La gráfica del arreglo **resultado\_ba** generada con las gráficas de la Figura 3.53 y 3.55 se aprecia en la Figura 3.56.

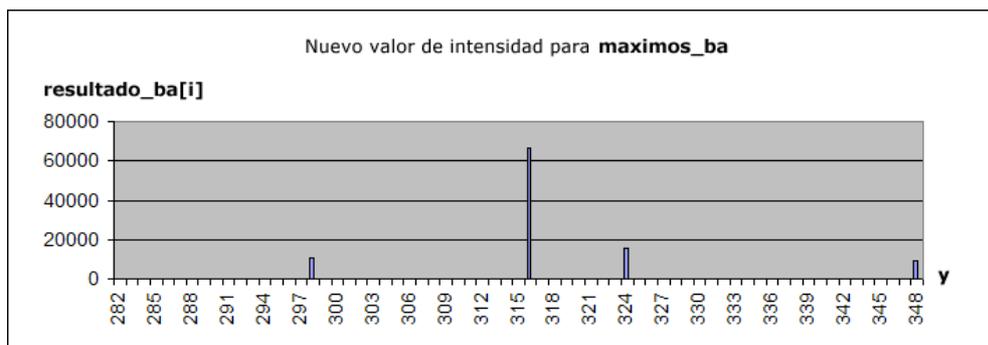


Figura 3.56: Gráfica del arreglo **resultado\_ba**.

Considerando el arreglo de intensidades **resultado\_ba** se calcula un umbral denotado por **umbral\_ba**, para la determinación de este umbral se toma el valor de intensidad

más alto contenido en **resultado\_ba** y el valor de intensidad más bajo del mismo arreglo, con estos dos valores el umbral es calculado de la forma siguiente:

$$\text{umbral\_ba} = \begin{cases} \frac{\text{Intensidad\_mas\_alta}}{2} & \text{Si Intensidad\_mas\_baja} \geq \frac{\text{Intensidad\_mas\_alta}}{2} \\ \frac{\text{Intensidad\_mas\_alta} + \text{Intensidad\_mas\_baja}}{2} & \text{Si Intensidad\_mas\_baja} < \frac{\text{Intensidad\_mas\_alta}}{2} \end{cases} \quad (3.19)$$

Por lo tanto la eliminación de los máximos locales contenidos en **maximos\_ba**, se realiza a través del arreglo **resultado\_ba**, donde para cada elemento  $i$  de **resultado\_ba** cuyo valor de intensidad sea menor al umbral **umbral\_ba**, el elemento  $i$  correspondiente al arreglo **maximos\_ba** tendrá un valor igual a cero, de lo contrario el elemento en **maximos\_ba** conservará su valor, nótese que el máximo global siempre conserva su valor, esto se aprecia en la fórmula (3.20).

$$\text{maximos\_ba}[i] = \begin{cases} 0 & \text{Si resultado\_ba}[i] < \text{umbral\_ba} \\ \text{maximos\_ba}[i] & \text{Si resultado\_ba}[i] \geq \text{umbral\_ba} \text{ ó} \\ & \text{Si maximos\_ba}[i] \text{ es máximo global (conserva su valor).} \end{cases} \quad (3.20)$$

El arreglo **maximos\_ba** resultante del proceso anterior se muestran en la Figura 3.57.

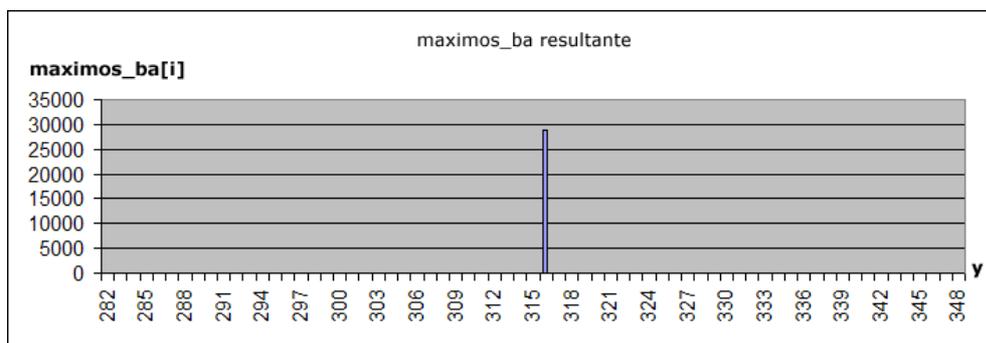


Figura 3.57: Gráfica del arreglo **maximos\_ba** con eliminación de máximos locales.

### 3. Búsqueda del máximo local con intensidad mayor.

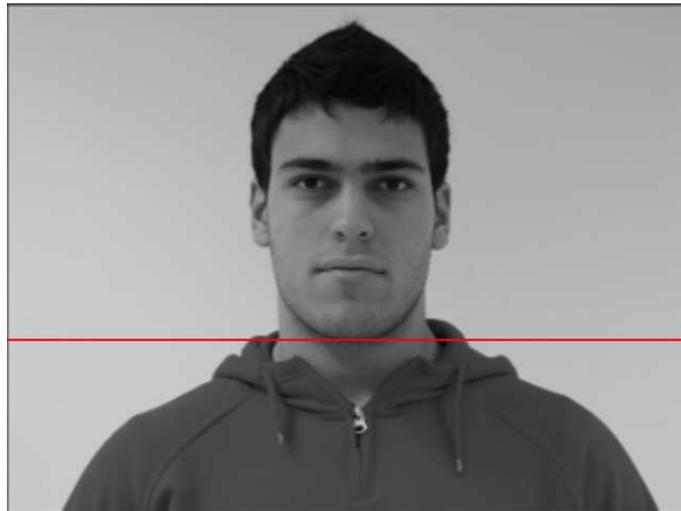
Después de haber localizado el máximo global y haber aplicado un umbral para reducir el número de máximos locales, buscamos el máximo local de mayor intensidad dentro del arreglo **maximos\_ba**, aunque puede no haber máximos locales.

### 4. Determinación de la barbilla.

Por último para determinar el elemento dentro del arreglo **maximos\_ba** cuya posición en  $y$  sea la posición vertical de la barbilla, se realiza una división entre el valor del máximo local mayor en intensidad y el valor del máximo global, para conocer la proporción de igualdad entre estas intensidades, si el resultado de esta división es mayor a 0.802 (valor obtenido empíricamente), la posición en  $y$  de la barba es la posición vertical del elemento con la menor intensidad o mínimo local dentro del arreglo **salida\_ba**, buscado en el rango  $[i_1, i_2]$ , donde  $i_1$  es el elemento  $i$  del arreglo que contiene al máximo global e  $i_2$  es el elemento  $i$  del arreglo que contiene al máximo local mayor en intensidad. De lo contrario si el resultado de la división es menor o igual a 0.802, pudiendo ser cero por la falta de un máximo local mayor en intensidad, la posición en  $y$  de la barba es la posición vertical del elemento que contiene al máximo global.

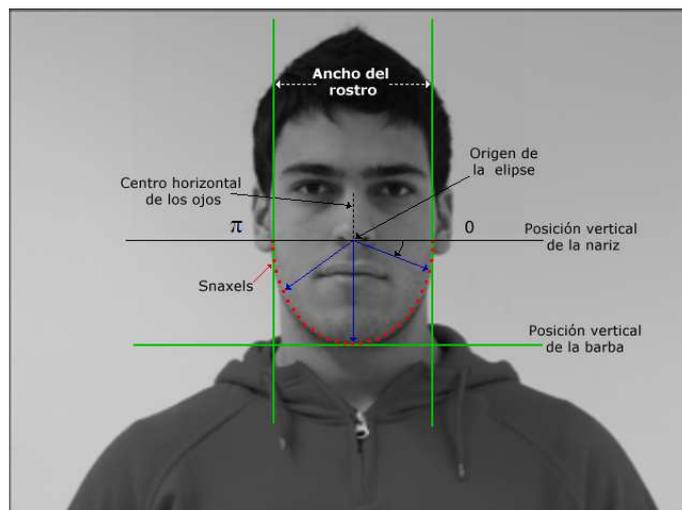
La posición en  $y$  de la barba obtenida por la aplicación del proceso anterior sobre la gráfica 3.53 se dibuja en la Figura 3.58 por la línea roja.

Retomando la inicialización inferior, se colocan 30 *snaxels* equidistantes sobre la forma dibujada por la elipse con eje menor igual al ancho del rostro y eje mayor igual al doble de la distancia de separación entre la posición vertical de la nariz y la posición vertical de la barba que ya fue encontrada, con origen en  $x$  igual al punto medio de la distancia de separación entre los ojos, y con origen en  $y$  igual a la posición vertical de la nariz, el trazado de los *snaxels* equidistantes sobre la media elipse se muestra en la Figura 3.59 por los puntos en



**Figura 3.58:** Posición vertical de la barba.

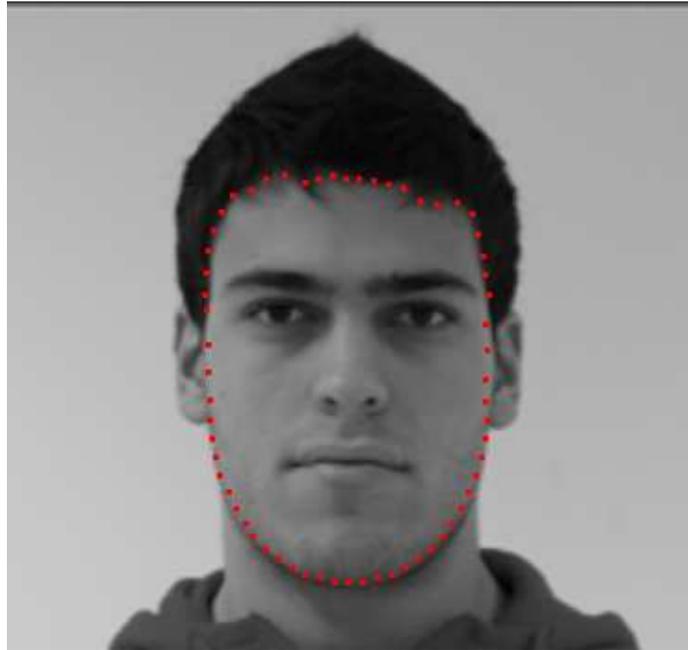
color rojo.



**Figura 3.59:** Trazado de los *snaxels* sobre la media elipse que conforma la quijada.

Un ejemplo de la unión de la inicialización inferior, media y superior puede verse en la Figura 3.60 por los puntos en color rojo.

La etapa de inicialización finaliza con la representación discreta del contorno activo: Una curva cerrada representada por el total de *snaxels* previamente fijados.



**Figura 3.60:** Ejemplo de la inicialización de los *snaxels*.

### 3.2.2.2. Modelo de Contorno Activo para Detectar el Contorno del Rostro

Considerando los conceptos especificados en la sección 2.9, se explicará cuál es el modelo de contorno activo utilizado en este proyecto:

El contorno activo tiene una representación discreta: La curva cerrada será representada por  $n$  puntos o *snaxels*. Una representación similar a la usada en [32, 34, 12, 22, 25, 21].

La función de energía que define su comportamiento (cantidad a minimizar) está dada por:

$$E_{global} = \int_0^1 E_{interna}(v(s)) + E_{imagen}(v(s)) + E_{balloon}(v(s)) \quad (3.21)$$

Donde las derivadas de la  $E_{interna}$  (de la ecuación (2.28)) se calculan de acuerdo a las ecuaciones (2.34) y (2.35) de la sección 2.9.2.

Y dado que se requiere localizar el contorno del rostro, la  $E_{imagen}$  definida en la ecuación (2.29) sólo considera el término  $\gamma_{borde}E_{borde}$  y se calcula por la ecuación (2.36) de la sección 2.9.2, donde el término  $mag$  está dado por la energía funcional  $E_{borde}$ , la cual se determina en términos del módulo del gradiente de la imagen convolucionada por una función gaussiana como se sugiere en [27]:

$$mag = E_{borde} = -|\nabla(g[5, 5] * I(x, y))| \quad (3.22)$$

Donde  $g[5, 5]$  es la máscara de convolución para el filtro gaussiano (definido en la sección 2.6.2.1) con desviación estándar  $\sigma^2 = 4$ , usando en este caso una ventana de convolución de  $5 \times 5$ ,  $I(x, y)$  es la imagen en niveles de gris, y  $\nabla$  es el gradiente especificado en la ecuación (2.23). Como puede observarse,  $E_{borde}$  fue generado previamente cuando se realizó el realce de bordes sobre la imagen en niveles de gris (explicado en la sección 3.2.2.1).

La fuerza de inflación (*balloon force*) implementada en este proyecto fue la propuesta por [10], la cual es adaptativa a los bordes y varía inversamente proporcional a la magnitud del gradiente de la imagen, y se define como:

$$E_{balloon}(v(s)) = \lambda n(s) \left( 1 - \frac{E_{borde}(v(s))}{255} \right) \quad (3.23)$$

Donde  $\lambda$  y  $n(s)$  son el peso de la fuerza de inflación y la normal unitaria respectivamente (ver sección 2.9.1).

Para llevar a cabo la minimización de la función de energía se empleó el algoritmo voraz propuesto en [32] y descrito en la sección 2.9.2. Enseguida se especificarán los parámetros

utilizados en este algoritmo.

### 3.2.2.3. Algoritmo Voraz

Cada *snaxel* tiene los siguientes valores iniciales:

- La constante  $\alpha$  que afecta a la elasticidad del contorno activo es igual a 1.
- La constante  $\beta$  que ajusta la rigidez del contorno activo es igual a 1.
- La constante  $\gamma$  que afecta al término  $E_{imagen}$  es igual a 1.2.
- La constante  $\lambda$  que define la amplitud de la fuerza de inflación es igual a 0.25 sólo para los *snaxels* cuya posición pertenezcan a la barbilla (es decir, los *snaxels* cuya posición en  $y$  sea mayor o igual a la posición vertical de la boca), y 0 para el resto. Únicamente se consideró la fuerza de inflación en esta área porque el contorno de la barbilla contiene bajo contraste en relación al cuello, haciendo que la confiabilidad del modelo activo sea baja, por lo que el propósito de la fuerza de inflación es empujar a los *snaxels* hacia fuera del rostro, evitando al contorno activo una contracción sobre sí mismo al no encontrar un borde cercano proveniente de la barbilla. La energía  $E_{balloon}$  es computada para cada uno de los ocho vecinos de  $v_i$  de acuerdo a la siguiente ecuación [10]:

$$E_{balloon\_neighbor}(v_i) = \lambda n_i(v_i - v_{neighbor}(v_i)) \left(1 - \frac{mag_{neighbor}}{255}\right) \quad (3.24)$$

Donde  $v_{neighbor}(v_i)$  es el punto en la vecindad de  $v_i$ ,  $E_{balloon\_neighbor}$  es la energía correspondiente a el  $v_{neighbor}(v_i)$ , y  $mag_{neighbor}$  es la magnitud del gradiente en el  $v_{neighbor}(v_i)$ .

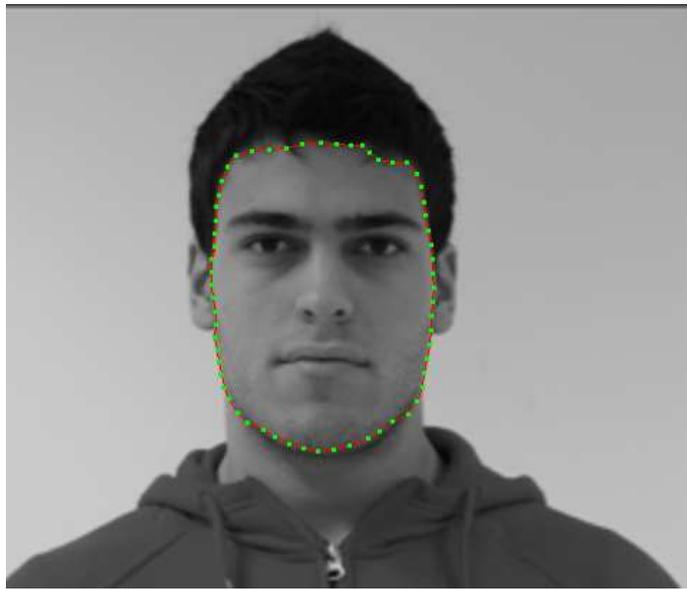
Los valores para  $\alpha$ ,  $\beta$  y  $\gamma$  se tomaron como se sugiere en [32], para  $\lambda$  el valor se obtuvo de manera empírica considerando que fuera menor a 1 como se propone en [5, 6].

El umbral de curvatura *umbral1* es igual a 0.9 y el umbral de borde *umbral2* es igual a 254, los cuales se adquirieron empíricamente.

El contorno activo se deja converger un número determinado de iteraciones de acuerdo a las siguientes condiciones de parada:

- Cuando el número de *snaxels* desplazados es cero, con un valor para *umbral3* igual a 1, es decir, todos los *snaxels* convergen al mínimo.
- Cuando se estabiliza la energía del contorno activo puede suceder que un número determinado de *snaxels* se encuentren oscilando y el contorno activo parecería no converger, por lo que las iteraciones se detienen si el número de *snaxels* desplazados son exactamente los mismos en 8 iteraciones seguidas, siendo este número de *snaxels* desplazados menor o igual al 20% del total de *snaxels* del contorno activo.
- Si las anteriores condiciones no se presentan, el algoritmo voraz es detenido al llegar a un número de iteraciones igual a 300, este umbral se obtuvo de manera empírica.

Un ejemplo del resultado del algoritmo voraz puede verse en la Figura 3.61 por la línea en color rojo, donde los *snaxels* están dibujados por puntos en color verde.



**Figura 3.61:** Ejemplo del resultado del algoritmo voraz, el tamaño de los *snaxels* en color verde fue aumentado para una mejor visualización.

# Capítulo 4

## Experimentos y Resultados

En este capítulo se detallan los experimentos realizados con el sistema descrito en el capítulo anterior y los resultados que se obtuvieron. Los experimentos se desarrollaron en dos computadoras con las siguientes características:

Características	Máquina 1	Máquina 2
Descripción	Dell Dimension XPS GEN 4	Dell Dimension 8400
Procesador	Intel(R) Pentium(R) 4 CPU 3.40 GHz - 3.39 GHz	Intel(R) Pentium(R) 4 CPU 3.00 GHz - 2.99 GHz
Memoria RAM	1.00 GB	512 MB
Disco duro	146 GB	70 GB
Sistema operativo	Windows Xp	Windows Xp

**Tabla 4.1:** Características de las máquinas en las que se realizaron las pruebas.

La implementación de los algoritmos descritos en el capítulo anterior fueron desarrollados en Visual C++ versión 6.0. Para los experimentos se usaron imágenes de la base de datos de imágenes pública: *GTAV Face Database*, la cual fue creada por el Dr. Francesc Tarres y el Ing. Antonio Rama del *Audio Visual Technologies Group* del departamento *Signal Theory and Communications Department* de la *Technical University of Catalonia*. Esta base de datos de imágenes se encuentra disponible en <sup>1</sup> :

<http://gps-tsc.upc.es/GTAV/ResearchAreas/UPCFaceDatabase/GTAVFaceDatabase.htm>

<sup>1</sup> Último acceso: 16 de Agosto de 2007.

La *GTAV Face Database* incluye imágenes de un total de 44 personas con 27 imágenes por persona, las cuales corresponden a diferentes poses de ésta ( $0^\circ$ ,  $\pm 30^\circ$ ,  $\pm 45^\circ$ ,  $\pm 60^\circ$  y  $\pm 90^\circ$ ) y bajo tres diferentes tipos de iluminación (luz natural o ambiental, una fuente de luz intensa desde un ángulo de  $45^\circ$ , y finalmente una fuente de luz semi-intensa casi frontal). Además, se incluyen por lo menos 10 imágenes adicionales por persona con vista frontal que muestran diferentes oclusiones y expresiones faciales. La resolución de las imágenes es de  $240 \times 320$  y están en formato BMP.

## 4.1. Experimentos y Evaluación

De las imágenes contenidas en la base de datos se utilizaron 85 que cumplen con las especificaciones descritas en la sección 3.1 y 74 que presentan diferentes variaciones (expresiones faciales marcadas, con oclusiones en los ojos - por lentes, pelo, y ojos cerrados -, ligera rotación horizontal del rostro, y la presencia de una mano cerca del rostro). La resolución original de las imágenes fue ampliada para su mejor análisis a  $480 \times 640$  utilizando para ello Microsoft Paint versión 5.1.

Para evaluar la correcta detección del contorno del rostro se empleó un método cuantitativo basado en la teoría de conjuntos y sugerido por [25], en el cual se realiza el cálculo de tres errores:

$$err_1 = \frac{|(A \cup B) \setminus (A \cap B)|}{|A| + |B|} * 100\% \quad (4.1)$$

$$err_2 = err_1 * \frac{|A| + |B|}{|A|} \quad (4.2)$$

Donde  $A$  y  $B$  son dos conjuntos cuyos elementos son puntos (pares ordenados) dentro de la región del rostro, el conjunto  $A$  contiene los puntos del límite del contorno del rostro se-

leccionado manualmente (éste se considera el contorno deseado) y el conjunto  $B$  contiene los puntos del límite del contorno del rostro detectado automáticamente por el sistema desarrollado en este proyecto. El  $err_1$  evalúa el error de coincidencia con respecto al total de puntos, es decir, es el porcentaje de los puntos que no coinciden entre los dos conjuntos; el proceso de intersección empleado en el  $err_1$  como es natural es el número de puntos que aparecen en ambos conjuntos  $A$  y  $B$ , pero al ser el contorno deseado seleccionado manualmente muy subjetivo al observador, esto da como resultado que la coincidencia de puntos sea muy complicada, para reducir esta complejidad se incrementa la posibilidad de intersección generando una ventana de  $7 \times 7$  sobre el punto analizado, de esta manera si dos puntos se encuentran dentro de la ventana de  $7 \times 7$  la intersección se considera como efectuada. Por otro lado, el  $err_2$  se estima como el porcentaje de no coincidencia con respecto al contorno deseado o contorno ideal.

El error  $err_3$  involucra la distancia euclideana promedio entre los dos contornos. El contorno del rostro seleccionado manualmente se denota por  $a$ , y el contorno del rostro detectado automáticamente se denota por  $b$ . La distancia euclideana desde el contorno  $a$  hacia el contorno  $b$  es:

$$D(a, b) = \frac{1}{N_a} \sum_{i=1}^{N_a} \min_{j=1, \dots, N_b} d(a(i), b(j)) \quad (4.3)$$

Donde  $d(, )$  es la distancia euclideana entre dos puntos,  $N_a$  y  $N_b$  son el número de puntos en el contorno  $a$  y  $b$  respectivamente.

La distancia euclideana promedio en píxeles entre los dos contornos es:

$$err_3 = (err_{31} + err_{32})/2 \quad (4.4)$$

Donde  $err_{31} = D(a, b)$  y  $err_{32} = D(b, a)$ .

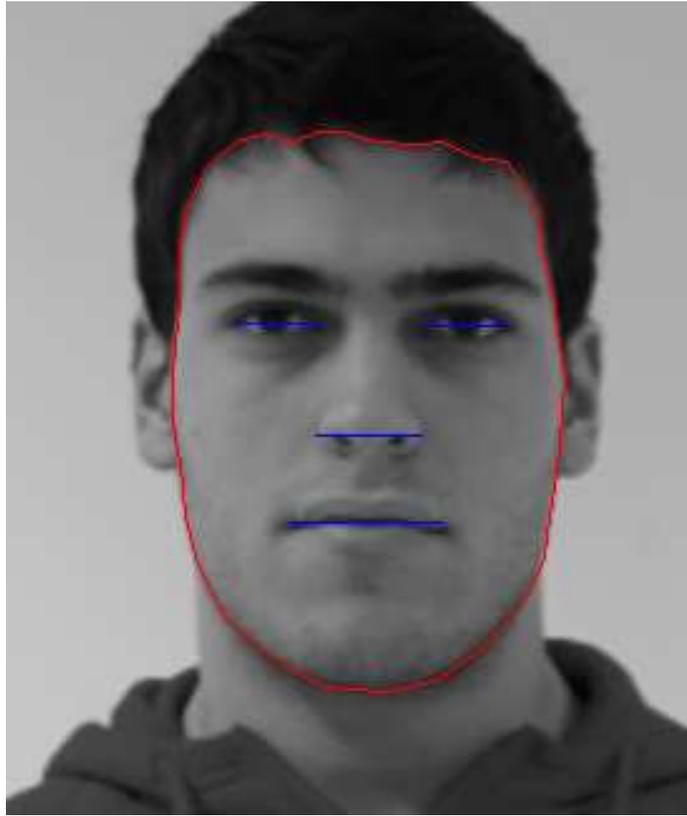
Para evaluar la correcta posición de los diferentes rasgos faciales se propuso un méto-

do cuantitativo, ya que en el artículo de Shih y Chuang [28] del que se obtuvo el modelo geométrico del rostro para localizar las características faciales la evaluación se hace cualitativamente. Este método cuantitativo consiste en establecer manualmente líneas horizontales para cada uno de los rasgos faciales y cuya longitud debe abarcar el tamaño deseado de éstos, entonces la medición del error para cada uno de los rasgos faciales se obtiene calculando el porcentaje de píxeles de la línea horizontal que se encuentran fuera de los recuadros obtenidos automáticamente, y que deben contener a los rasgos faciales.

En la Figura 4.1 se muestra un ejemplo de una imagen donde aparece el contorno deseado (color rojo) y las líneas horizontales (color azul) que abarcan cada uno de los rasgos faciales, los cuales fueron establecidos manualmente. Por lo que, para el cálculo de los porcentajes de error fue necesario crear una imagen similar a la de la Figura 4.1 para cada una de las imágenes de prueba.

Como se explicó antes, la detección automática del contorno del rostro y características faciales se aplicó a 159 imágenes: La medición de los errores para las 85 imágenes se muestra en las tablas 4.2 y 4.3, y para las 74 imágenes las mediciones aparecen en las tablas 4.4 y 4.5. Las tablas contienen las siguientes 12 columnas:

- **Nom.** Nombre de la imagen.
- **Snax.** Número de *snaxels* del contorno activo.
- **Iter.** Número de iteraciones en las que converge el algoritmo voraz.
- **T\_Proc\_Maq1.** Tiempo promedio de procesamiento de la máquina 1 (de diez ejecuciones).
- **T\_Dib\_Maq1.** Tiempo promedio en que se tarda la máquina 1 en dibujar los resultados (de diez ejecuciones).



**Figura 4.1:** Ejemplo del contorno deseado y las líneas horizontales para los rasgos faciales establecidos manualmente.

- **T\_Proc\_Maq2.** Tiempo promedio de procesamiento de la máquina 2 (de diez ejecuciones).
- **T\_Dib\_Maq2.** Tiempo promedio en que se tarda la máquina 2 en dibujar los resultados (de diez ejecuciones).
- **Err1.** Error descrito por la ecuación (4.1).
- **Err2.** Error descrito por la ecuación (4.2).
- **Err3.** Error descrito por la ecuación (4.4).
- **Er\_Ojo\_Izq.** Porcentaje de píxeles que no están contenidos en el recuadro del ojo izquierdo.

- **Er\_Ojo\_Der.** Porcentaje de píxeles que no están contenidos en el recuadro del ojo derecho.
- **Er\_Nar.** Porcentaje de píxeles que no están contenidos en el recuadro de la nariz.
- **Er\_Boc.** Porcentaje de píxeles que no están contenidos en el recuadro de la boca.

Para las columnas que muestran un porcentaje de error éste fue redondeado a dos dígitos de precisión.

Nom.	Snax.	Iter.	T_Proc_Maql.	T_Dib_Maql.	T_Proc_Maql.	T_Dib_Maql.	Err1	Err2	Err3	Er_Ojo_Izq.	Er_Ojo_Der.	Er_Nar.	Er_Boc.
1.bmp	69	38	0s 801ms	0s 551ms	0s 721ms	0s 462ms	12.64%	25.06%	1.9316	0.00%	0.00%	0.00%	0.00%
2.bmp	68	40	0s 690ms	0s 560ms	0s 628ms	0s 463ms	19.32%	38.36%	2.3539	0.00%	0.00%	0.00%	0.00%
3.bmp	72	49	0s 856ms	0s 561ms	0s 778ms	0s 465ms	8.07%	16.09%	1.7219	0.00%	0.00%	0.00%	0.00%
4.bmp	71	72	0s 914ms	0s 560ms	0s 815ms	0s 467ms	26.58%	53.92%	2.8089	0.00%	0.00%	0.00%	0.00%
5.bmp	62	73	0s 621ms	0s 560ms	0s 553ms	0s 468ms	25.75%	51.88%	2.661	0.00%	0.00%	0.00%	0.00%
6.bmp	62	31	0s 549ms	0s 562ms	0s 497ms	0s 464ms	16.44%	32.23%	2.3036	0.00%	0.00%	0.00%	0.00%
7.bmp	62	68	0s 743ms	0s 562ms	0s 673ms	0s 465ms	20.65%	41.63%	2.6242	0.00%	0.00%	0.00%	0.00%
8.bmp	59	90	0s 628ms	0s 561ms	0s 562ms	0s 467ms	3.99%	8.10%	1.3155	0.00%	0.00%	0.00%	0.00%
9.bmp	62	39	0s 595ms	0s 562ms	0s 540ms	0s 460ms	9.84%	19.75%	1.7925	0.00%	0.00%	0.00%	0.00%
10.bmp	61	54	0s 657ms	0s 559ms	0s 590ms	0s 465ms	23.16%	46.18%	3.0302	0.00%	0.00%	0.00%	0.00%
11.bmp	63	56	0s 668ms	0s 564ms	0s 606ms	0s 478ms	27.43%	54.98%	3.7608	0.00%	0.00%	0.00%	0.00%
12.bmp	61	26	0s 565ms	0s 562ms	0s 511ms	0s 476ms	13.33%	26.81%	2.0424	0.00%	0.00%	0.00%	0.00%
13.bmp	60	44	0s 592ms	0s 565ms	0s 532ms	0s 467ms	35.46%	71.65%	3.6395	0.00%	0.00%	0.00%	0.00%
14.bmp	58	75	0s 656ms	0s 560ms	0s 590ms	0s 467ms	10.83%	21.94%	1.8556	0.00%	0.00%	0.00%	0.00%
15.bmp	67	81	0s 874ms	0s 562ms	0s 789ms	0s 462ms	19.04%	37.97%	2.4868	0.00%	7.69%	0.00%	0.00%
16.bmp	63	82	0s 759ms	0s 559ms	0s 684ms	0s 459ms	20.46%	40.45%	2.3936	0.00%	0.00%	0.00%	0.00%
17.bmp	59	57	0s 548ms	0s 563ms	0s 490ms	0s 462ms	57.11%	111.73%	8.6927	0.00%	0.00%	0.00%	0.00%
18.bmp	59	80	0s 635ms	0s 561ms	0s 568ms	0s 465ms	42.43%	84.42%	4.236	0.00%	0.00%	0.00%	0.00%
19.bmp	64	80	0s 822ms	0s 562ms	0s 737ms	0s 467ms	3.20%	6.38%	1.6703	0.00%	0.00%	0.00%	0.00%
20.bmp	63	118	0s 917ms	0s 562ms	0s 826ms	0s 464ms	22.47%	44.70%	2.755	0.00%	0.00%	0.00%	0.00%
21.bmp	65	84	0s 780ms	0s 560ms	0s 695ms	0s 465ms	15.33%	30.83%	2.4299	0.00%	0.00%	0.00%	0.00%
22.bmp	65	92	0s 842ms	0s 562ms	0s 753ms	0s 465ms	6.36%	12.70%	1.7798	0.00%	0.00%	0.00%	0.00%
23.bmp	63	150	0s 857ms	0s 561ms	0s 773ms	0s 462ms	25.89%	51.53%	3.0433	0.00%	0.00%	0.00%	0.00%
24.bmp	65	47	0s 685ms	0s 564ms	0s 623ms	0s 470ms	20.35%	40.68%	2.5224	0.00%	0.00%	0.00%	0.00%
25.bmp	65	56	0s 746ms	0s 559ms	0s 665ms	0s 469ms	23.50%	46.96%	2.9245	0.00%	0.00%	2.17%	3.23%
26.bmp	61	71	0s 701ms	0s 563ms	0s 633ms	0s 465ms	45.53%	90.19%	5.2357	0.00%	0.00%	0.00%	0.00%
27.bmp	61	53	0s 721ms	0s 572ms	0s 635ms	0s 464ms	27.25%	53.87%	3.7345	0.00%	0.00%	0.00%	0.00%
28.bmp	60	104	0s 779ms	0s 570ms	0s 684ms	0s 467ms	22.78%	45.14%	2.9776	0.00%	0.00%	0.00%	0.00%
29.bmp	57	55	0s 603ms	0s 565ms	0s 548ms	0s 463ms	28.31%	56.00%	3.8729	0.00%	0.00%	4.55%	0.00%
30.bmp	56	96	0s 662ms	0s 564ms	0s 626ms	0s 467ms	24.23%	48.90%	2.9004	0.00%	0.00%	6.82%	0.00%
31.bmp	66	300	1s 1110ms	0s 556ms	1s 1006ms	0s 464ms	33.73%	66.57%	3.3257	0.00%	0.00%	0.00%	0.00%
32.bmp	65	61	0s 618ms	0s 562ms	0s 579ms	0s 464ms	50.21%	96.60%	6.0975	0.00%	0.00%	0.00%	0.00%
33.bmp	63	126	0s 848ms	0s 562ms	0s 776ms	0s 464ms	10.74%	21.52%	2.2915	0.00%	0.00%	0.00%	0.00%
34.bmp	62	92	0s 784ms	0s 562ms	0s 736ms	0s 460ms	23.09%	46.40%	3.2911	0.00%	0.00%	0.00%	0.00%
35.bmp	64	48	0s 750ms	0s 515ms	0s 656ms	0s 467ms	21.51%	42.90%	2.7706	0.00%	0.00%	0.00%	0.00%
36.bmp	60	150	0s 860ms	0s 569ms	0s 784ms	0s 468ms	29.60%	59.65%	3.7407	0.00%	0.00%	0.00%	0.00%
37.bmp	58	101	0s 817ms	0s 570ms	0s 701ms	0s 466ms	43.10%	85.60%	4.6755	0.00%	0.00%	0.00%	0.00%
38.bmp	66	86	0s 790ms	0s 570ms	0s 735ms	0s 465ms	14.00%	28.28%	2.4879	0.00%	0.00%	0.00%	0.00%
39.bmp	55	93	0s 545ms	0s 571ms	0s 498ms	0s 467ms	15.87%	31.36%	2.3697	0.00%	0.00%	0.00%	0.00%
40.bmp	54	53	0s 460ms	0s 571ms	0s 470ms	0s 468ms	6.64%	13.24%	1.4995	0.00%	0.00%	0.00%	0.00%
41.bmp	60	49	0s 576ms	0s 576ms	0s 542ms	0s 465ms	32.33%	65.77%	3.4845	0.00%	0.00%	0.00%	0.00%
42.bmp	60	90	0s 648ms	0s 570ms	0s 600ms	0s 465ms	13.67%	27.02%	2.3705	0.00%	0.00%	0.00%	0.00%
43.bmp	58	108	0s 840ms	0s 567ms	0s 800ms	0s 463ms	26.57%	53.34%	2.9298	0.00%	0.00%	0.00%	0.00%

Tabla 4.2: Evaluación de la localización del contorno del rostro y de las características faciales para las 85 imágenes.

Nom.	Snax.	Iter.	T_Proc_Maql1.	T_Dib_Maql1.	T_Proc_Maql2.	T_Dib_Maql2.	Err1	Err2	Err3	Er_Ojo_Izq.	Er_Ojo_Der.	Er_Nar.	Er_Boc.
44.bmp	61	172	1s 924ms	0s 569ms	0s 942ms	0s 456ms	33.10%	67.42%	3.4331	0.00%	0.00%	0.00%	0.00%
45.bmp	61	83	0s 688ms	0s 570ms	0s 639ms	0s 463ms	11.38%	22.64%	2.0686	0.00%	0.00%	0.00%	0.00%
46.bmp	63	108	0s 714ms	0s 570ms	0s 668ms	0s 465ms	21.44%	45.21%	3.6534	0.00%	0.00%	0.00%	0.00%
47.bmp	62	53	0s 620ms	0s 571ms	0s 584ms	0s 462ms	17.76%	36.48%	2.6659	0.00%	0.00%	0.00%	0.00%
48.bmp	60	150	0s 778ms	0s 565ms	0s 715ms	0s 465ms	27.56%	58.33%	4.1325	0.00%	0.00%	0.00%	0.00%
49.bmp	61	55	0s 607ms	0s 571ms	0s 571ms	0s 468ms	26.25%	53.85%	2.8272	0.00%	0.00%	0.00%	0.00%
50.bmp	66	88	0s 798ms	0s 568ms	0s 742ms	0s 468ms	29.58%	60.06%	3.0646	0.00%	0.00%	0.00%	0.00%
51.bmp	66	101	0s 742ms	0s 570ms	0s 690ms	0s 476ms	31.08%	63.03%	3.2812	0.00%	0.00%	0.00%	0.00%
52.bmp	61	49	0s 618ms	0s 570ms	0s 597ms	0s 474ms	7.37%	14.71%	1.8974	0.00%	0.00%	0.00%	0.00%
53.bmp	59	58	0s 592ms	0s 571ms	0s 557ms	0s 473ms	11.82%	23.93%	1.7207	0.00%	0.00%	0.00%	0.00%
54.bmp	61	61	0s 648ms	0s 570ms	0s 595ms	0s 464ms	10.40%	20.98%	1.7202	0.00%	0.00%	0.00%	0.00%
55.bmp	61	80	0s 690ms	0s 574ms	0s 628ms	0s 465ms	15.74%	31.72%	2.1682	0.00%	0.00%	0.00%	0.00%
56.bmp	64	300	1s 926ms	0s 568ms	0s 903ms	0s 465ms	32.27%	63.24%	4.7123	0.00%	0.00%	0.00%	0.00%
57.bmp	62	57	0s 540ms	0s 570ms	0s 493ms	0s 467ms	34.37%	69.37%	3.9091	0.00%	0.00%	0.00%	0.00%
58.bmp	58	59	0s 675ms	0s 580ms	0s 614ms	0s 464ms	17.76%	35.60%	2.5064	0.00%	0.00%	0.00%	0.00%
59.bmp	58	119	0s 817ms	0s 571ms	0s 734ms	0s 461ms	15.94%	31.68%	2.5121	0.00%	0.00%	0.00%	0.00%
60.bmp	56	130	0s 856ms	0s 568ms	0s 760ms	0s 467ms	19.65%	39.37%	2.8003	0.00%	0.00%	0.00%	0.00%
61.bmp	64	62	0s 792ms	0s 568ms	0s 698ms	0s 461ms	20.29%	40.79%	2.789	0.00%	0.00%	0.00%	0.00%
62.bmp	66	44	0s 870ms	0s 565ms	0s 770ms	0s 453ms	45.25%	88.10%	4.8563	0.00%	0.00%	5.08%	0.00%
63.bmp	71	87	0s 979ms	0s 559ms	0s 878ms	0s 454ms	21.83%	43.62%	2.9062	0.00%	0.00%	0.00%	0.00%
64.bmp	68	300	1s 1196ms	0s 562ms	1s 1067ms	0s 453ms	4.14%	8.28%	1.4965	0.00%	0.00%	0.00%	0.00%
65.bmp	70	60	0s 865ms	0s 565ms	0s 784ms	0s 446ms	2.42%	4.85%	1.3826	0.00%	0.00%	0.00%	0.00%
66.bmp	63	124	0s 873ms	0s 562ms	0s 793ms	0s 456ms	7.59%	15.43%	1.3313	0.00%	0.00%	0.00%	0.00%
67.bmp	69	39	0s 746ms	0s 560ms	0s 684ms	0s 456ms	43.77%	90.13%	5.8269	0.00%	0.00%	0.00%	0.00%
68.bmp	55	65	0s 534ms	0s 570ms	0s 490ms	0s 449ms	36.27%	67.58%	5.7266	0.00%	0.00%	8.33%	0.00%
69.bmp	59	67	0s 543ms	0s 567ms	0s 495ms	0s 454ms	39.42%	74.66%	5.9703	0.00%	0.00%	0.00%	0.00%
70.bmp	66	56	0s 817ms	0s 562ms	0s 746ms	0s 443ms	35.44%	72.74%	4.4245	0.00%	0.00%	0.00%	0.00%
71.bmp	67	113	0s 856ms	0s 564ms	0s 775ms	0s 449ms	6.03%	12.06%	1.5724	0.00%	0.00%	0.00%	0.00%
72.bmp	67	104	0s 804ms	0s 563ms	0s 731ms	0s 448ms	34.57%	70.49%	3.9904	0.00%	0.00%	0.00%	0.00%
73.bmp	68	75	0s 773ms	0s 562ms	0s 696ms	0s 447ms	14.85%	29.03%	2.1575	0.00%	0.00%	0.00%	0.00%
74.bmp	66	75	0s 764ms	0s 568ms	0s 687ms	0s 450ms	20.09%	40.06%	2.293	0.00%	0.00%	0.00%	0.00%
75.bmp	59	92	0s 622ms	0s 564ms	0s 560ms	0s 470ms	20.38%	40.72%	2.6841	0.00%	0.00%	0.00%	0.00%
76.bmp	64	77	0s 848ms	0s 561ms	0s 762ms	0s 459ms	13.38%	26.67%	2.029	0.00%	0.00%	3.17%	0.00%
77.bmp	60	80	0s 728ms	0s 565ms	0s 640ms	0s 459ms	24.19%	48.42%	3.566	0.00%	0.00%	0.00%	0.00%
78.bmp	65	59	0s 753ms	0s 564ms	0s 649ms	0s 459ms	14.39%	28.91%	2.1914	0.00%	0.00%	0.00%	0.00%
79.bmp	71	51	0s 761ms	0s 562ms	0s 663ms	0s 464ms	20.29%	40.92%	2.5222	2.50%	0.00%	0.00%	0.00%
80.bmp	62	49	0s 612ms	0s 565ms	0s 573ms	0s 462ms	20.73%	41.65%	2.2685	0.00%	0.00%	0.00%	0.00%
81.bmp	56	89	0s 672ms	0s 574ms	0s 576ms	0s 473ms	14.58%	29.32%	2.8971	0.00%	0.00%	0.00%	0.00%
82.bmp	64	50	0s 531ms	0s 560ms	0s 473ms	0s 459ms	26.38%	53.25%	3.1299	0.00%	0.00%	0.00%	0.00%
83.bmp	61	45	0s 687ms	0s 565ms	0s 598ms	0s 460ms	12.01%	23.92%	2.0795	0.00%	0.00%	0.00%	0.00%
84.bmp	59	57	0s 696ms	0s 564ms	0s 617ms	0s 461ms	3.86%	7.74%	1.5487	0.00%	0.00%	0.00%	0.00%
85.bmp	62	37	0s 651ms	0s 565ms	0s 582ms	0s 460ms	30.86%	61.48%	4.1764	0.00%	0.00%	0.00%	0.00%

Tabla 4.3: Evaluación de la localización del contorno del rostro y de las características faciales para las 85 imágenes.

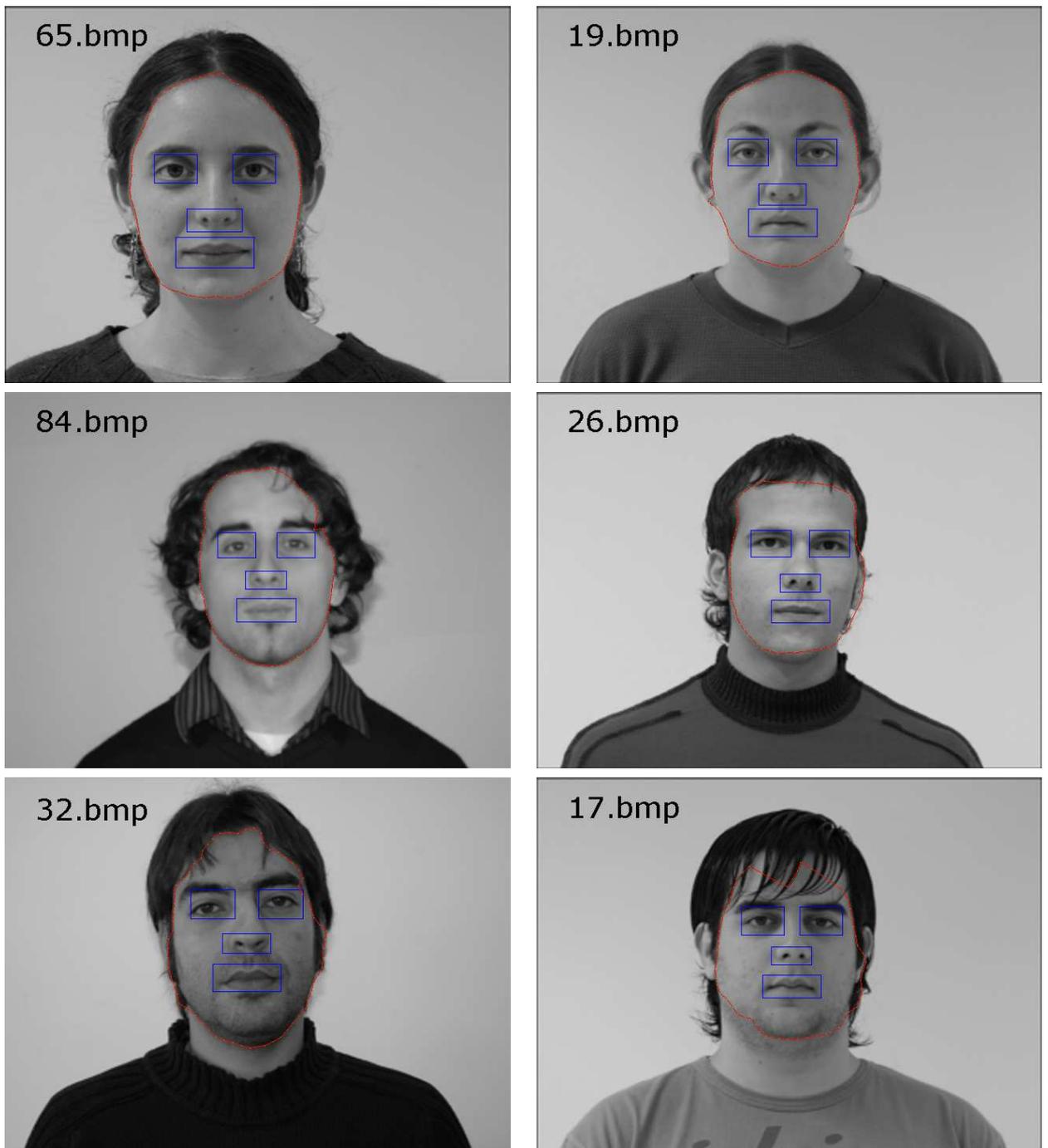
Nom.	Smax.	Itr.	T_Proc.Maql.	T_Dib.Maql.	T_Proc.Maql2.	T_Dib.Maql2.	Err1	Err2	Err3	Er_Ojo.Izq.	Er_Ojo.Der.	Er_Nar.	Er_Roc.
86.bmp	66	77	0s 789ms	0s 563ms	0s 704ms	0s 460ms	36.03%	70.91%	5.6628	0.00%	0.00%	5.08%	4.60%
87.bmp	62	71	0s 771ms	0s 568ms	0s 698ms	0s 456ms	26.36%	51.61%	3.4603	0.00%	0.00%	0.00%	0.00%
88.bmp	64	97	0s 770ms	0s 573ms	0s 686ms	0s 459ms	11.77%	23.91%	2.2134	0.00%	0.00%	4.55%	0.00%
89.bmp	64	81	0s 745ms	0s 573ms	0s 654ms	0s 457ms	11.32%	23.16%	2.2388	0.00%	0.00%	0.00%	0.00%
90.bmp	56	110	0s 779ms	0s 578ms	0s 673ms	0s 459ms	11.94%	23.66%	2.0142	0.00%	0.00%	0.00%	0.00%
91.bmp	59	43	0s 646ms	0s 571ms	0s 554ms	0s 459ms	38.69%	76.81%	4.0289	0.00%	0.00%	2.22%	0.00%
92.bmp	56	82	0s 648ms	0s 573ms	0s 567ms	0s 457ms	26.57%	51.90%	4.1618	0.00%	0.00%	2.13%	0.00%
93.bmp	64	63	0s 645ms	0s 571ms	0s 548ms	0s 459ms	31.87%	61.99%	4.0581	0.00%	0.00%	0.00%	0.00%
94.bmp	64	51	0s 690ms	0s 571ms	0s 603ms	0s 459ms	38.33%	77.61%	4.9515	0.00%	0.00%	6.98%	10.34%
95.bmp	59	77	0s 700ms	0s 570ms	0s 621ms	0s 456ms	17.07%	34.80%	2.5469	0.00%	0.00%	6.67%	5.00%
96.bmp	69	31	0s 906ms	0s 579ms	0s 795ms	0s 456ms	44.02%	85.71%	5.2134	0.00%	0.00%	7.81%	1.27%
97.bmp	59	64	0s 775ms	0s 573ms	0s 732ms	0s 457ms	6.19%	12.23%	1.7375	17.65%	9.38%	0.00%	0.00%
98.bmp	70	96	0s 968ms	0s 576ms	0s 829ms	0s 460ms	13.81%	27.72%	1.7039	11.11%	0.00%	24.14%	23.86%
99.bmp	66	65	0s 801ms	0s 572ms	0s 697ms	0s 460ms	40.42%	81.38%	5.2329	0.00%	0.00%	15.25%	12.35%
100.bmp	63	104	0s 889ms	0s 575ms	0s 775ms	0s 457ms	31.62%	63.59%	3.1956	0.00%	0.00%	0.00%	1.14%
101.bmp	71	68	0s 995ms	0s 573ms	0s 878ms	0s 460ms	23.01%	45.83%	2.9418	0.00%	0.00%	0.00%	4.08%
102.bmp	72	84	0s 989ms	0s 578ms	0s 878ms	0s 454ms	28.81%	57.66%	3.1257	0.00%	0.00%	0.00%	0.00%
103.bmp	74	70	0s 990ms	0s 575ms	0s 856ms	0s 459ms	59.43%	118.56%	9.4026	0.00%	0.00%	3.17%	5.32%
104.bmp	69	73	0s 892ms	0s 576ms	0s 789ms	0s 459ms	12.58%	25.39%	1.9111	0.00%	0.00%	0.00%	3.37%
105.bmp	66	300	1s 1165ms	0s 575ms	1s 1020ms	0s 460ms	23.21%	47.87%	3.1336	0.00%	10.81%	30.77%	18.87%
106.bmp	62	162	0s 942ms	0s 570ms	0s 823ms	0s 462ms	25.98%	50.98%	3.7756	0.00%	0.00%	4.17%	0.00%
107.bmp	67	32	0s 664ms	0s 581ms	0s 578ms	0s 458ms	6.60%	13.34%	1.5159	0.00%	0.00%	0.00%	0.00%
108.bmp	61	71	0s 685ms	0s 572ms	0s 689ms	0s 464ms	8.06%	16.03%	1.7949	0.00%	0.00%	0.00%	0.00%
109.bmp	64	86	0s 742ms	0s 563ms	0s 701ms	0s 462ms	41.02%	79.16%	6.2144	0.00%	2.50%	0.00%	0.00%
110.bmp	54	300	1s 920ms	0s 570ms	0s 974ms	0s 461ms	34.32%	67.90%	4.4863	0.00%	0.00%	4.08%	2.86%
111.bmp	59	85	0s 751ms	0s 570ms	0s 757ms	0s 459ms	36.70%	75.71%	7.9528	0.00%	0.00%	0.00%	0.00%
112.bmp	54	52	0s 517ms	0s 565ms	0s 489ms	0s 459ms	58.99%	101.88%	6.0133	0.00%	0.00%	6.67%	0.00%
113.bmp	64	90	0s 887ms	0s 563ms	0s 825ms	0s 462ms	29.31%	59.05%	4.5431	0.00%	0.00%	0.00%	0.00%
114.bmp	61	78	0s 725ms	0s 578ms	0s 670ms	0s 460ms	22.53%	44.46%	2.9032	0.00%	0.00%	0.00%	0.00%
115.bmp	63	98	0s 807ms	0s 565ms	0s 795ms	0s 459ms	22.76%	46.05%	3.1828	0.00%	0.00%	0.00%	0.00%
116.bmp	67	76	0s 784ms	0s 567ms	0s 735ms	0s 461ms	31.81%	64.72%	3.7957	0.00%	0.00%	0.00%	0.00%
117.bmp	65	66	0s 707ms	0s 565ms	0s 650ms	0s 462ms	12.38%	24.80%	1.8727	0.00%	2.86%	1.92%	0.00%
118.bmp	66	134	0s 865ms	0s 564ms	0s 803ms	0s 461ms	18.54%	37.91%	2.4858	0.00%	0.00%	3.85%	7.95%
119.bmp	63	43	0s 706ms	0s 572ms	0s 659ms	0s 465ms	26.82%	52.85%	3.9494	0.00%	0.00%	6.67%	6.82%
120.bmp	67	77	0s 793ms	0s 564ms	0s 724ms	0s 462ms	3.08%	6.21%	1.1887	0.00%	0.00%	0.00%	0.00%
121.bmp	59	72	0s 684ms	0s 567ms	0s 642ms	0s 457ms	34.46%	64.78%	9.7095	0.00%	0.00%	5.36%	1.39%
122.bmp	58	99	0s 720ms	0s 570ms	0s 677ms	0s 458ms	28.96%	4.8653	4.8653	0.00%	0.00%	6.25%	0.00%
123.bmp	62	61	0s 707ms	0s 570ms	0s 645ms	0s 464ms	35.67%	69.71%	5.9209	0.00%	0.00%	2.04%	8.00%
124.bmp	66	131	0s 851ms	0s 565ms	0s 770ms	0s 461ms	7.98%	16.48%	1.7321	0.00%	0.00%	0.00%	0.00%
125.bmp	70	84	0s 712ms	0s 567ms	0s 648ms	0s 468ms	24.31%	49.16%	2.841	0.00%	2.78%	0.00%	0.00%
126.bmp	64	74	0s 690ms	0s 563ms	0s 671ms	0s 471ms	6.61%	13.36%	1.4348	0.00%	0.00%	6.82%	0.00%
127.bmp	54	66	0s 706ms	0s 567ms	0s 578ms	0s 468ms	15.21%	30.22%	2.5092	0.00%	20.00%	21.74%	14.52%
128.bmp	58	70	0s 631ms	0s 564ms	0s 584ms	0s 467ms	4.30%	8.61%	1.7137	0.00%	0.00%	0.00%	0.00%

Tabla 4.4: Evaluación de la localización del contorno del rostro y de las características faciales para las 74 imágenes.

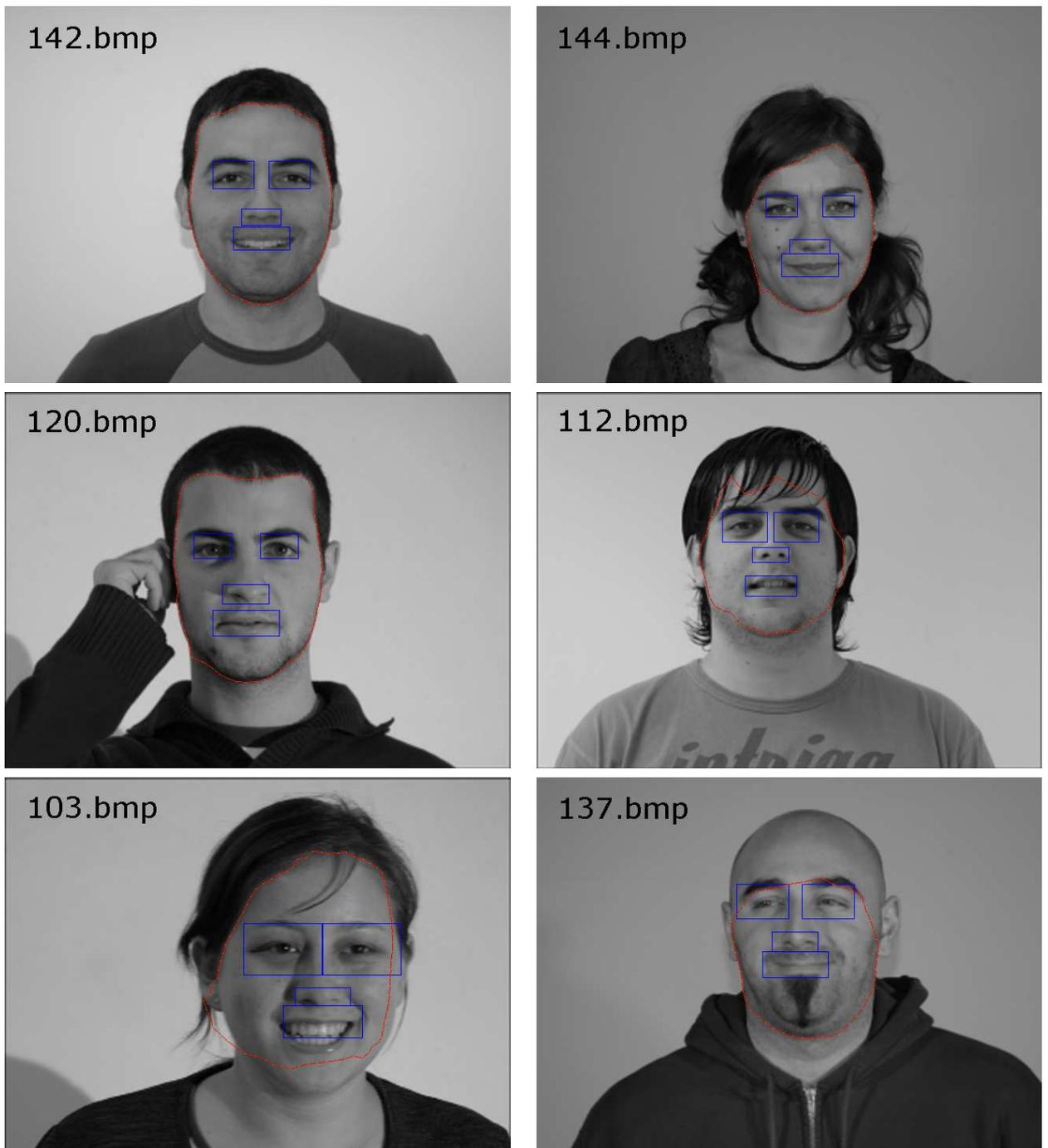
Nom.	Snax.	Iter.	T_Proc_Maq1.	T_Dib_Maq1.	T_Proc_Maq2.	T_Dib_Maq2.	Err1	Err2	Err3	Er_Ojo_Izq.	Er_Ojo_Der.	Er_Nar.	Er_Boc.
129.bmp	53	134	0s 710ms	0s 567ms	0s 668ms	0s 465ms	7.06%	14.02%	1.7586	0.00%	0.00%	0.00%	0.00%
130.bmp	59	51	0s 604ms	0s 573ms	0s 563ms	0s 463ms	26.82%	53.50%	3.1039	0.00%	12.90%	20.41%	25.97%
131.bmp	63	53	0s 623ms	0s 567ms	0s 609ms	0s 467ms	41.75%	75.68%	8.6904	2.56%	0.00%	0.00%	0.00%
132.bmp	59	300	1s 1002ms	0s 565ms	1s 909ms	0s 462ms	19.26%	38.20%	2.0809	100.00%	0.00%	15.69%	16.00%
133.bmp	63	300	1s 1128ms	0s 576ms	1s 1031ms	0s 461ms	21.46%	43.80%	2.4195	0.00%	0.00%	0.00%	0.00%
134.bmp	54	65	0s 461ms	0s 576ms	0s 436ms	0s 460ms	29.34%	58.86%	3.8281	10.00%	9.09%	6.00%	0.00%
135.bmp	68	88	0s 754ms	0s 579ms	0s 712ms	0s 457ms	20.79%	43.61%	3.8615	0.00%	0.00%	0.00%	0.00%
136.bmp	59	119	0s 860ms	0s 576ms	0s 790ms	0s 462ms	25.09%	51.03%	3.2587	0.00%	0.00%	0.00%	0.00%
137.bmp	49	92	0s 839ms	0s 578ms	0s 823ms	0s 456ms	85.10%	155.67%	24.8905	0.00%	0.00%	2.13%	7.04%
138.bmp	64	55	0s 679ms	0s 574ms	0s 728ms	0s 461ms	15.32%	31.75%	2.3083	0.00%	0.00%	0.00%	5.41%
139.bmp	64	84	0s 747ms	0s 578ms	0s 692ms	0s 456ms	11.55%	23.37%	2.4139	0.00%	0.00%	0.00%	0.00%
140.bmp	61	79	0s 809ms	0s 581ms	0s 726ms	0s 456ms	3.15%	6.30%	1.6231	0.00%	0.00%	0.00%	0.00%
141.bmp	64	99	0s 896ms	0s 579ms	0s 809ms	0s 458ms	11.83%	23.63%	2.0358	0.00%	0.00%	4.69%	10.10%
142.bmp	63	52	0s 685ms	0s 575ms	0s 623ms	0s 456ms	0.45%	0.90%	0.9817	0.00%	0.00%	2.04%	7.69%
143.bmp	60	226	0s 987ms	0s 579ms	0s 878ms	0s 456ms	31.60%	70.23%	5.8848	0.00%	0.00%	0.00%	0.00%
144.bmp	62	68	0s 673ms	0s 575ms	0s 589ms	0s 457ms	2.78%	5.59%	1.2913	0.00%	0.00%	0.00%	0.00%
145.bmp	56	81	0s 626ms	0s 579ms	0s 563ms	0s 464ms	26.92%	54.31%	3.5561	0.00%	0.00%	0.00%	6.06%
146.bmp	60	61	0s 698ms	0s 574ms	0s 611ms	0s 464ms	27.75%	56.49%	3.1416	6.25%	0.00%	0.00%	0.00%
147.bmp	66	300	1s 1168ms	0s 576ms	1s 1007ms	0s 462ms	15.78%	31.41%	2.4705	0.00%	12.82%	0.00%	0.00%
148.bmp	56	53	0s 616ms	0s 575ms	0s 549ms	0s 462ms	12.16%	24.09%	1.9296	0.00%	0.00%	0.00%	0.00%
149.bmp	63	69	0s 685ms	0s 579ms	0s 620ms	0s 459ms	45.97%	94.29%	6.8704	0.00%	0.00%	0.00%	0.00%
150.bmp	63	43	0s 587ms	0s 576ms	0s 528ms	0s 462ms	16.91%	33.76%	3.0618	0.00%	0.00%	0.00%	1.33%
151.bmp	69	40	0s 593ms	0s 568ms	0s 545ms	0s 462ms	27.27%	55.36%	3.4892	0.00%	0.00%	0.00%	0.00%
152.bmp	67	79	0s 660ms	0s 575ms	0s 589ms	0s 462ms	25.37%	50.59%	3.0095	0.00%	0.00%	0.00%	6.67%
153.bmp	66	91	0s 692ms	0s 579ms	0s 607ms	0s 464ms	11.92%	23.75%	1.9573	0.00%	0.00%	1.72%	7.23%
154.bmp	64	56	0s 535ms	0s 578ms	0s 482ms	0s 465ms	27.58%	57.00%	3.6788	0.00%	0.00%	1.89%	4.11%
155.bmp	68	71	0s 875ms	0s 576ms	0s 773ms	0s 464ms	8.04%	16.12%	1.7458	0.00%	0.00%	0.00%	1.08%
156.bmp	61	75	0s 714ms	0s 573ms	0s 629ms	0s 464ms	42.95%	83.70%	6.0144	0.00%	0.00%	0.00%	0.00%
157.bmp	65	76	0s 792ms	0s 576ms	0s 709ms	0s 460ms	41.76%	84.86%	11.5797	0.00%	0.00%	4.08%	6.06%
158.bmp	64	300	1s 1026ms	0s 583ms	1s 906ms	0s 462ms	24.80%	48.14%	3.9315	0.00%	3.12%	0.00%	0.00%
159.bmp	65	62	0s 765ms	0s 573ms	0s 678ms	0s 461ms	35.82%	72.06%	5.049	0.00%	0.00%	0.00%	0.00%

Tabla 4.5: Evaluación de la localización del contorno del rostro y de las características faciales para las 74 imágenes.

A partir de los resultados de las mediciones de error, se seleccionaron para ser mostradas en este documento 6 imágenes del conjunto de 85 y 6 del grupo de 74. Se eligieron de acuerdo al valor del porcentaje de error  $err_1$ , ya que es el error de coincidencia con respecto al total de puntos. En cada grupo de 6 imágenes se muestran 3 imágenes con valor de  $err_1$  menor (mejor coincidencia) y 3 con el valor de  $err_1$  mayor (peor coincidencia). Las 6 imágenes tomadas del conjunto de 85 se muestran en la Figura 4.2, y las 6 y tomadas del conjunto de 74 aparecen en la Figura 4.3. Para cada imagen se presentan los valores de error como: ( $err_1$ ,  $err_2$ ,  $err_3$ ). El conjunto completo de las imágenes resultado estarán disponibles en la carpeta **Imágenes\resultados** del CD que acompaña a esta tesis.



**Figura 4.2:** Localización del contorno del rostro y rasgos faciales del conjunto de 85 imágenes. Mejor coincidencia: 65.bmp (2.42%; 4.85%; 1.3826), 19.bmp (3.20%; 6.38%; 1.6703), 84.bmp (3.86%; 7.74%; 1.5487). Peor coincidencia: 26.bmp (45.53%; 90.19%; 5.2357), 32.bmp (50.21%; 96.60%; 6.0975), 17.bmp (57.11%; 111.73%; 8.6927).

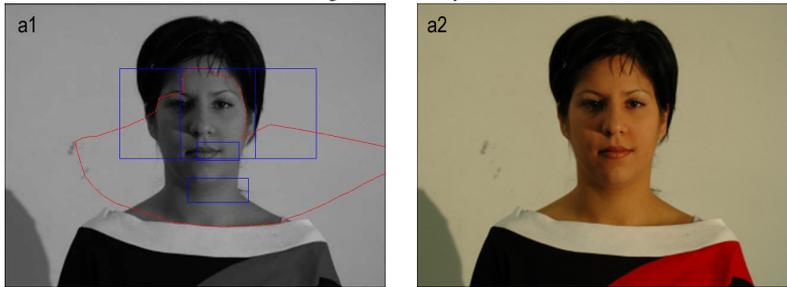


**Figura 4.3:** Localización del contorno del rostro y rasgos faciales del conjunto de 74 imágenes. Mejor coincidencia: 142.bmp (0.45%; 0.90%; 0.9817), 144.bmp (2.78%; 5.59%; 1.2913), 120.bmp (3.08%; 6.21%; 1.1887). Peor coincidencia: 112.bmp (58.99%; 101.88%; 6.0133), 103.bmp (59.43%; 118.56%; 9.4026), 137.bmp (85.10%; 155.67%; 24.8905).

## 4.2. Casos de Funcionamiento No Adecuado

Los casos donde se da una mala localización del contorno del rostro y de las características faciales se muestran en las Figuras 4.4, 4.5 y 4.6. Para la imagen *a1* se muestra a su vez la imagen original a color *a2* para una mejor visualización de las sombras; en la imagen *c1* aparece también la imagen original *c2* donde el color del pelo se confunde con el color de piel; para las imágenes: *b*, *d*, *i*, *o*, *p*, *q* y *t*, tan solo se muestran las imágenes originales debido a que no generan una imagen de salida, esto es causado por problemas al detectar la boca o la nariz, es decir, que las posiciones de estos rasgos se encuentren fuera de los márgenes de la imagen ó el valor sea cero para alguno de ellos, evitando que se dibujen los resultados; para la imagen *m* sucede algo similar, en este caso se presentó un problema al detectar el ancho del rostro, al tener una mala detección de uno o ambos lados del rostro (fuera de los límites de la imagen) producido por la rotación de éste, esto también ocasiona que no se dibujen los resultados.

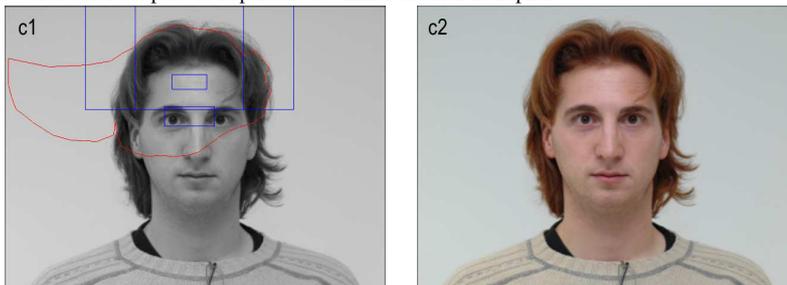
1. Cuando debido a la iluminación se genera una mayor cantidad de sombras en el área del rostro.



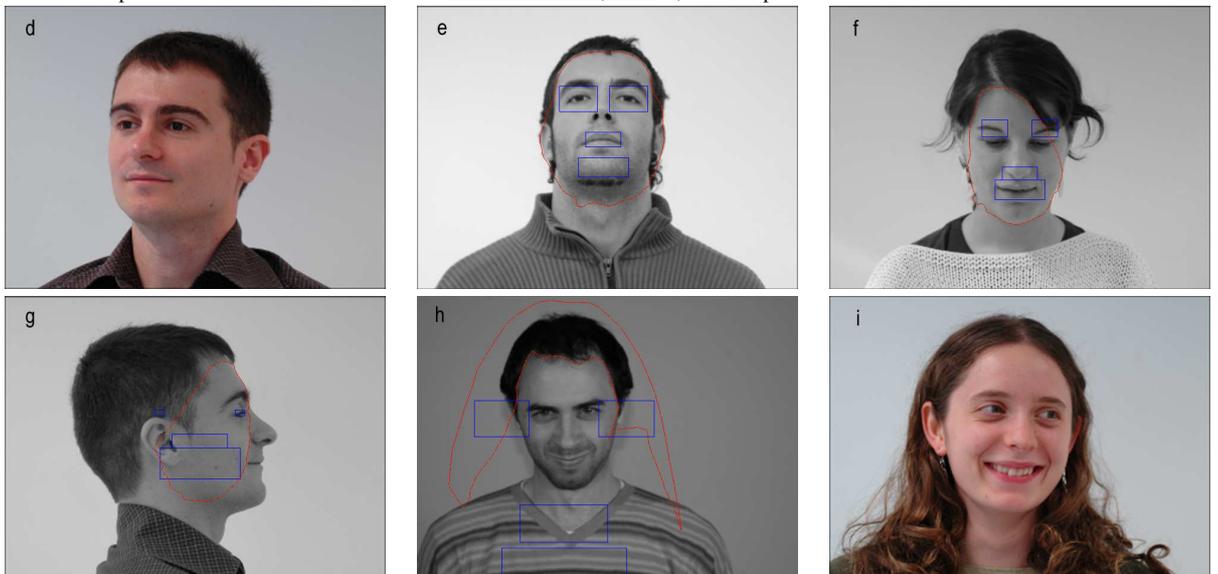
2. Si la iluminación hace que los objetos o el fondo en la escena sean confundibles con el color de la piel.



3. Si el color del pelo de la persona es similar al color de la piel.

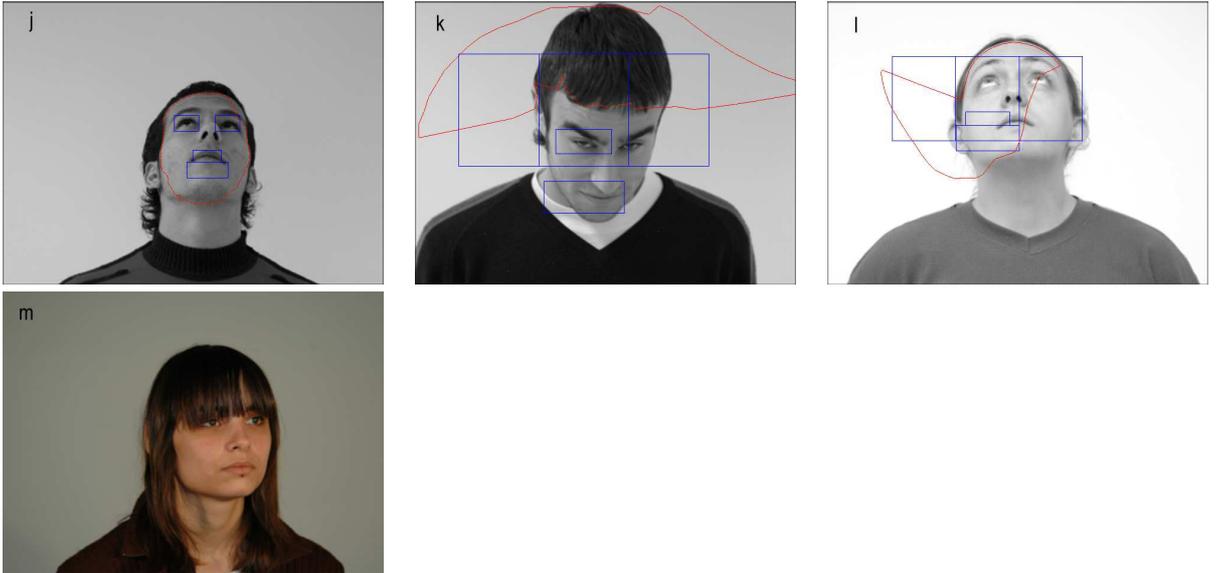


4. Cuando la persona contenida en la escena no tiene una vista frontal, es decir, el rostro presenta una marcada rotación horizontal o vertical.

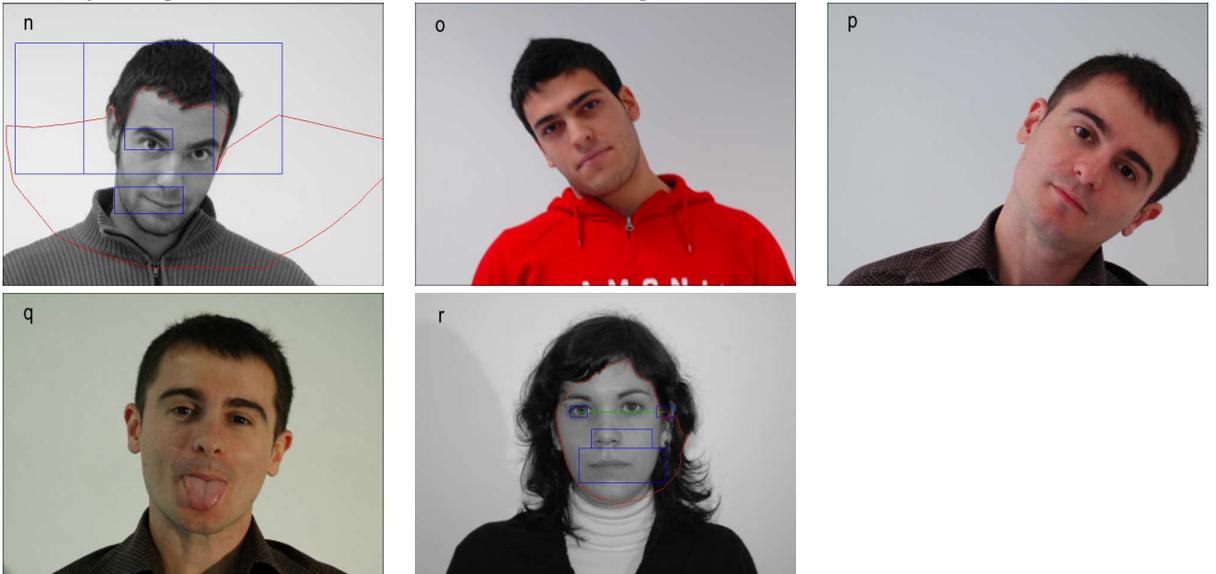


**Figura 4.4:** Ejemplos de una incorrecta localización del contorno del rostro y rasgos faciales.

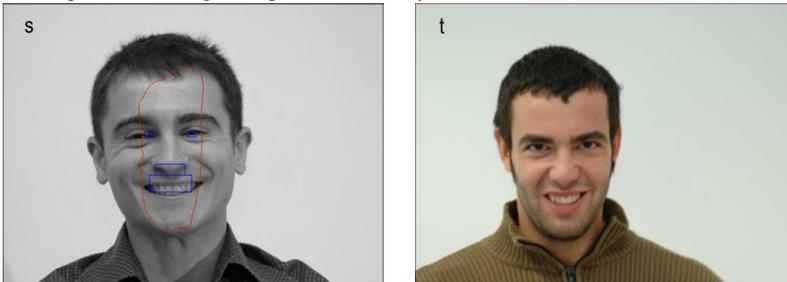
4. Cuando la persona contenida en la escena no tiene una vista frontal, es decir, el rostro presenta una marcada rotación horizontal o vertical.



5. Si los ojos de la persona se encuentran desalineados ó si la cabeza de la persona mantiene cierta inclinación.

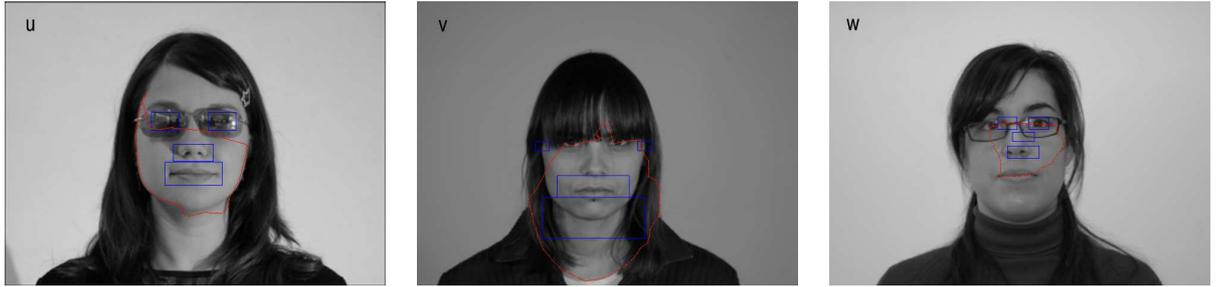


6. Si la persona tiene alguna expresión facial muy marcada.



**Figura 4.5:** Ejemplos de una incorrecta localización del contorno del rostro y rasgos faciales.

7. Cuando existen oclusiones en el área del rostro (lentes, fleco, entre otros).



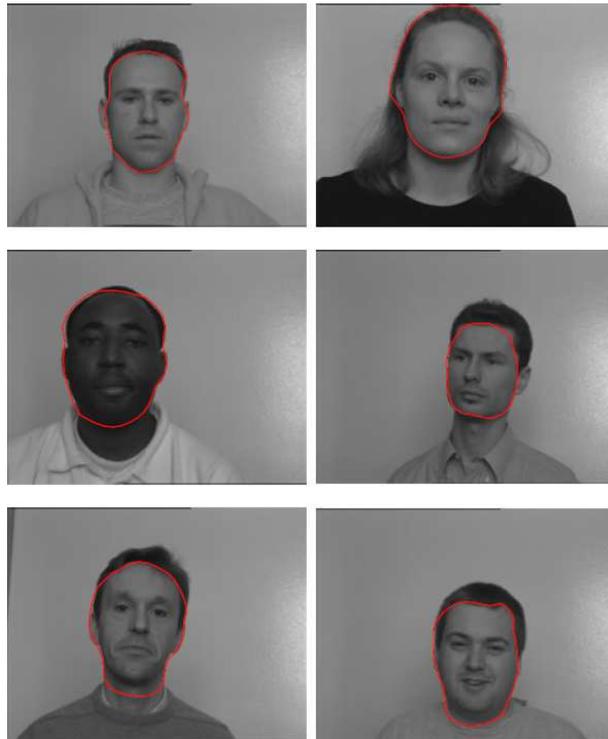
**Figura 4.6:** Ejemplos de una incorrecta localización del contorno del rostro y rasgos faciales.

### 4.3. Comparación con Otros Sistemas Similares

Para poder realizar una comparación cuantitativa de los resultados obtenidos en este trabajo de tesis, se buscaron las mediciones de trabajos relacionados:

Megherbi y Ambellouis, en su trabajo [21] combinan el contorno activo (similar al de este proyecto) con un algoritmo genético para extraer el contorno de la cabeza y del rostro, no muestran una medición de sus experimentos, la evaluación sólo es cualitativa y las imágenes con las que trabajan son de 768x576 píxeles, la ejecución de su aplicación dura alrededor de 1 minuto, y no se hace mención del software y hardware empleado. Ejemplos de los resultados que reportan aparecen en la Figura 4.7.

Gunn y Nixon, utilizan en [12] un contorno activo global y uno local empleando programación dinámica para extraer el contorno interno y externo de la cabeza, sus resultados son medidos de manera cualitativa empleando una calificación subjetiva de ocurrencia sobre el contorno deseado, calificando sus resultados como excelente, bueno y pobre. Utilizaron un conjunto de 75 imágenes con una resolución no especificada, no se hace mención del tiempo de ejecución de su aplicación, así como del software y hardware empleado. Ejemplos de los resultados que reportan aparecen en la Figura 4.8.



**Figura 4.7:** Ejemplo de los resultados de [21].



**Figura 4.8:** Ejemplo de los resultados de [12].

Shih y Chuang, en [28] realizan una extracción del contorno del rostro empleando diferentes filtros de detección de bordes y doble umbralización; la localización de los ojos se hace a través de un filtro de Gabor, y para los rasgos faciales usan un modelo geométrico del rostro. Aplicaron su método a más de 100 imágenes con una resolución de 384x256, el tiem-

po de ejecución de su aplicación es alrededor de 170 segundos -incluyendo 15 segundos para el procesamiento del modelo geométrico del rostro-, el trabajo fue desarrollado en Matlab con una máquina Sun Ultra 60 Workstation, y sólo evalúan sus resultados cualitativamente. Ejemplos de los resultados que reportan aparecen en la Figura 4.9.



**Figura 4.9:** Ejemplo de los resultados de [28].

Peng, Chen, Ruan y Kukharev, en [24] desarrollaron un algoritmo para la detección de

ojos en rostros sin lentes sobre imágenes en niveles de gris, aplicando métodos basados en características y *templates*, experimentaron con 227 imágenes en niveles de gris con una resolución de 92x112, el tiempo de ejecución de su aplicación es de 0.982 segundos empleando Matlab, en una PC Pentium IV a 1.8 GHz., la evaluación de sus resultados es cualitativa: Para cada una de las imágenes la valoración fue una correcta o incorrecta localización de los ojos. Ejemplos de los resultados que reportan aparecen en la Figura 4.10.



**Figura 4.10:** Ejemplo de los resultados de [24].

Perlibakas hizo una detección automática de las características faciales y extracción del contorno del rostro en [25], los rasgos faciales fueron encontrados aplicando morfología matemática del rostro y reglas heurísticas, las características faciales son filtradas y se genera un mapa de bordes, se aplica un contorno activo con energías internas (tensión y rigidez) y energías externas (mapa de bordes del campo generalizado del flujo del vector gradiente), los experimentos se realizaron a 427 imágenes de una base de datos de 600, cuya resolución fue de 128x192 píxeles, utilizando una computadora Pentium II a 400 MHz., bajo el sistema operativo Windows 98, y el sistema fue programado en Matlab 5.2. La evaluación del contorno del rostro fue hecha cuantitativamente empleado teoría de conjuntos y la distancia euclídeana entre el contorno detectado y un contorno manual. El tiempo promedio de convergencia para el contorno activo es de 37.9 segundos y el tiempo promedio de la localización de los rasgos faciales es de 26.7 segundos. Como la localización de los rasgos faciales sirvió de base para

la construcción del contorno activo, su medición se realizó de manera cualitativa. Ejemplos de los resultados que reportan aparecen en la Figura 4.11.



**Figura 4.11:** Ejemplo de los resultados de [25].

De acuerdo a los anteriores trabajos relacionados que localizan los rasgos faciales y evalúan los resultados cualitativamente, es difícil hacer una comparación directa con el sistema desarrollado debido a que la apreciación visual es muy subjetiva. Sin embargo, en este proyecto de tesis se implementó una medición cuantitativa de los rasgos faciales descrita en 4.1, por lo que el porcentaje promedio de acierto de los rasgos faciales para las 159 (85 y 74) imágenes seleccionadas se muestra en la tabla 4.6.

Rasgo facial	Porcentaje promedio de acierto
Ojo izquierdo	99.02%
Ojo derecho	99.04%
Nariz	98.32%
Boca	98.49%

**Tabla 4.6:** Porcentaje promedio de acierto de los rasgos faciales de las 159 imágenes de prueba.

Para el contorno del rostro se aplicó la medición sugerida por [25], y cuyos valores para

las 159 imágenes fueron colocados en las tablas 4.2, 4.3, 4.4 y 4.5, sin embargo no se puede hacer una comparación directa de estos resultados con los obtenidos por [25], debido a que no se utilizó la misma base de datos de imágenes, y cuantitativamente difieren mucho, esto es producido por que las mediciones se basan sobre un contorno deseado que es dibujado manualmente, y en [25] no se especifica como se obtiene dicho contorno manual. Por otro lado visualmente (cualitativamente) los resultados son similares.

Finalmente, no se puede realizar una comparación estricta con respecto al tiempo de detección del contorno del rostro y las características faciales, esto por la utilización de diferentes tipos de computadoras y diferentes tipos de compiladores para generar las diversas aplicaciones.

# Capítulo 5

## Conclusiones y Trabajos Futuros

### 5.1. Conclusiones

En este trabajo de tesis se cumplió con el objetivo planteado: La detección automática del contorno del rostro y características faciales, mediante técnicas del enfoque basado en características. El sistema se elaboró en dos etapas:

1. Durante la primera etapa se detectaron los rasgos faciales, primero se localizó a la persona en la escena, empleando detección de piel y seguimiento del contorno, para la detección de piel se empleó una regla simple para seleccionar los píxeles pertenecientes a la piel, esta regla resultó fácil y eficaz, pero se encuentra limitada a un rango de tonos de piel y es fácilmente afectada por la iluminación ó sombras; para la detección de la silueta de la persona se realizó un algoritmo que utiliza el ancho de los bordes del contorno para marcar el seguimiento, la desventaja de este algoritmo es que no funcionaría ante la presencia de un fondo no uniforme; los primeros rasgos faciales detectados son los ojos, aplicando proyecciones verticales y horizontales del histograma, las cuales resultan útiles para detectar concentraciones de niveles gris, pero los resultados se ven afectados en la localización de los ojos cuando la región de proyección no es restringida correctamente o la posición de los ojos tiene cierta inclinación; para los demás rasgos

faciales se combinó un modelo geométrico del rostro con la proyección vertical, esta combinación se produjo porque el modelo geométrico no proporciona una correcta localización de la nariz y boca para todos los casos y su resultado está limitado por la precisión al encontrar el centro de los ojos, en cambio este modelo si proporciona una región estable de búsqueda para aplicar la proyección vertical; también se propuso un método cuantitativo para medir el error al detectar los rasgos faciales, este método está basado en líneas horizontales, y resulta ser adecuado, la desventaja de esta medición es que no mide la eficiencia vertical de la detección, esto produce una mala estimación cuando la boca se encuentra mediana o completamente abierta.

2. La segunda etapa del sistema consiste en la extracción del contorno del rostro, y depende de la etapa anterior, la localización del contorno del rostro se hace a través un contorno activo tradicional, su representación es sencilla y se hizo converger con el algoritmo voraz, el cual es fácil de programar y genera una convergencia rápida del contorno activo; por otro lado, este contorno activo tradicional es muy sensible a su inicialización para generar buenos resultados, por lo tanto se hizo una inicialización basada en una aproximación del contorno del rostro (empleada para la detección de los ojos en la etapa anterior), y en la localización aproximada de la barbilla, para localizar ésta se generó un método basado en la proyección vertical del histograma; también durante la inicialización se utilizó la localización del ancho del rostro, proponiendo una modificación al ancho sugerido por el modelo geométrico, esta modificación emplea detección de bordes y da buenos resultados cuando el ancho del rostro real es mayor o igual al sugerido; la energía de inflación es agregada al contorno activo tradicional en la región de la barba para aumentar su atracción a los bordes de la barba que son débiles por su cercanía con el cuello; la medición de los errores del contorno del rostro cuantitativamente se hizo como lo sugiere [25].

Los experimentos fueron realizados empleando un total de 159 imágenes, y para cada una de ellas se construyó una imagen con el contorno deseado establecido manualmente, al igual que los rasgos faciales; el tiempo de procesamiento es adecuado en comparación con los tiempos obtenidos en trabajos relacionados (aunque no se pueden comparar directamente por que no hay uniformidad tanto en hardware como en software); cualitativamente los resultados obtenidos son buenos y numéricamente dependen de la construcción manual del contorno deseado y las características faciales.

## 5.2. Trabajos Futuros

Para generar un mejor desempeño en la detección automática del contorno del rostro y las características faciales propuesto por este proyecto de tesis, se ha pensado abundar en los siguientes puntos:

- Un método de detección de piel más preciso, que sea aplicable a diferentes tonos de piel.
- Cambiar el algoritmo de detección de la silueta de la persona para que trabaje con fondos no uniformes.
- Mejorar la localización de los ojos, con un método que no sea sensible a la inclinación del rostro, una forma sería el empleo de templates ó mediante un filtro de Gabor.
- Cambiar el modelo tradicional del contorno activo, y emplear un b-spline como representación, así como probar otro método de convergencia (con programación dinámica), experimentar con diferentes energías como sería la utilización del flujo del vector gradiente (*GVF -Gradient Vector Flow-*).
- Emplear otro método para la detección de la barbilla.

- Mejorar la técnica de detección del ancho del rostro.

# Apéndice A

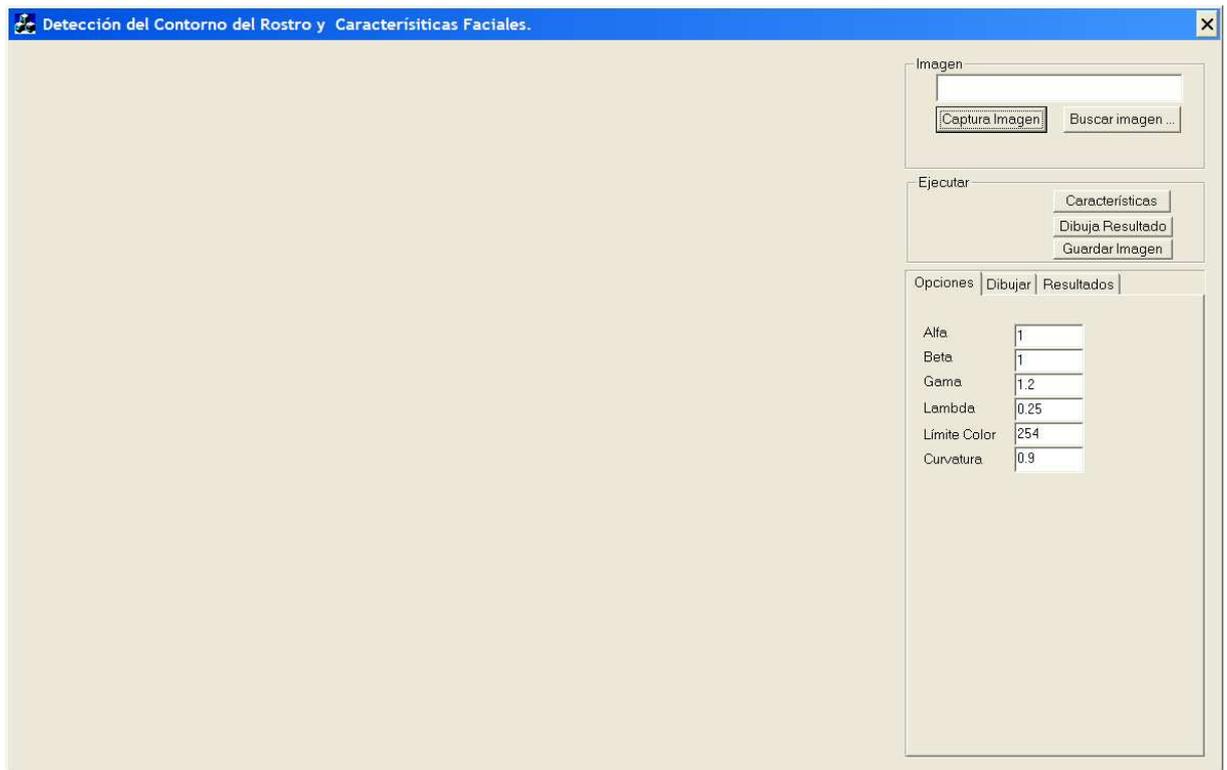
## Apéndice

### A.1. Manual de Usuario

El software generado por este proyecto de tesis requiere de una memoria en RAM de 25 Mb., y un espacio en disco duro de 217 Kb., y se ejecuta bajo el sistema operativo Windows 95 o superior. La aplicación y el código fuente se encuentran en un CD que acompaña a este trabajo (la organización del CD se detalla en el archivo Instrucciones.txt que se encuentra en el directorio principal). Para iniciar la aplicación se ejecuta el archivo *DCRCF.exe* dentro de la carpeta *Aplicacion*, esta aplicación consta de una única ventana, la cual se muestra en la Figura A.1, cuyo menú de opciones se encuentra en la parte derecha de la misma, y en la parte izquierda se localiza la zona de dibujado para la imagen de entrada/salida.

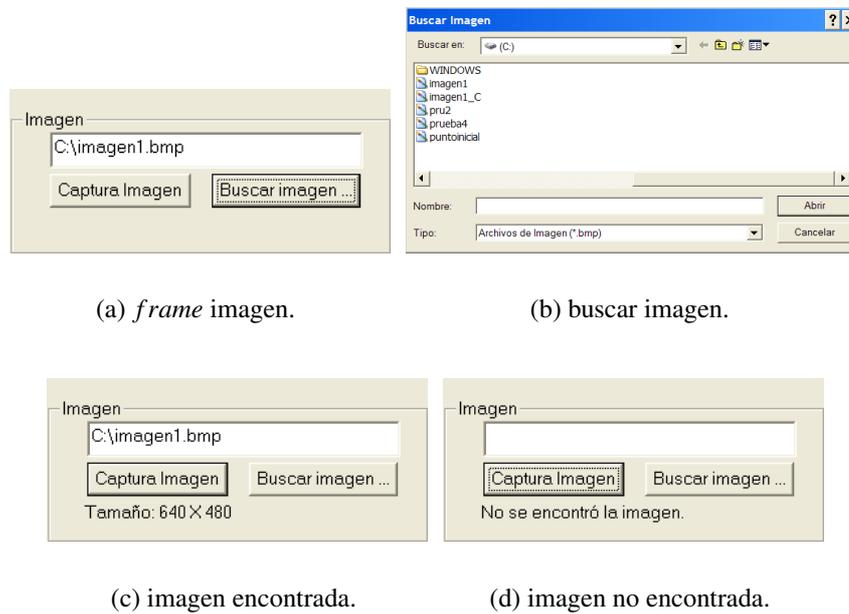
El menú de se encuentra dividido en tres partes:

1. **Captura de la imagen.** Para ingresar una imagen al sistema se recurre al *frame* titulado *Imagen*, el nombre y ruta de la imagen se ingresará dentro del cuadro de texto contenido en dicho *frame* como se aprecia en la Figura A.2(a); o se puede buscar la imagen presionando el botón *Buscar...*, y aparecerá un cuadro de diálogo para buscar la imagen (ver Figura A.2(b)); después de teclear o buscar el nombre de la imagen se



**Figura A.1:** Ventana del software desarrollado en este proyecto de tesis.

presiona el botón *Captura Imagen*, si la imagen es encontrada será almacenada y desplegada, en la parte inferior del botón aparecerá el tamaño de la imagen (ver Figura A.2(c)), de lo contrario aparecerá la leyenda "No se encontró la imagen"(ver Figura A.2(d)). El resultado de este proceso se muestra en la Figura A.3.

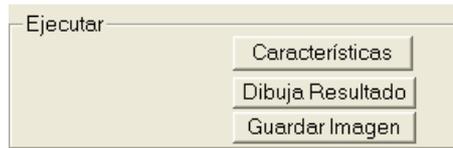


**Figura A.2:** Captura de la imagen de entrada.



**Figura A.3:** Resultado de la captura de la imagen de entrada.

2. **Ejecutar.** Después de almacenar y desplegar la imagen de entrada el siguiente paso es su procesamiento, para ello se hace uso de las opciones del frame con título *Ejecutar*. Este frame consta de tres botones (ver Figura A.4):



**Figura A.4:** *Frame Ejecutar.*

- a. **Características.** Este botón procesa la detección automática del contorno del rostro y las características faciales y dibuja el resultado de esta detección en la parte izquierda de la ventana.
  - b. **Dibuja Resultado.** Este botón redibuja el resultado de la detección automática en la parte izquierda de la ventana, según las opciones de dibujo, las cuales se detallarán más adelante.
  - c. **Guardar Imagen.** Este botón guarda la imagen que se esté mostrando en la parte izquierda de la ventana, el archivo se guardará con el mismo nombre y en la misma ruta que la imagen de entrada pero se le adjuntará el término ```R```. Ejemplo, imagen de entrada: `c : \1.bmp`; imagen guardada: `c : \1.R.bmp`.
3. **Pestañas: Opciones, Dibujar y Resultados:**

- **Opciones.** Esta pestaña muestra la posibilidad de manipulación de las diferentes variables involucradas con el contorno activo, las cuales fueron descritas en la sección 2.9, los parámetros del algoritmo voraz `umbral1` se denota por *Curvatura* y a `umbral2` como *Límite Color* (ver Figura A.5).
- **Dibujar.** Esta pestaña muestra las diferentes opciones de cómo y qué se dibujará en la imagen resultante (ver Figura A.6), para que los cambios surtan efecto

se requiere de presionar el botón *Dibujar Resultado* del *frame Ejecutar*.

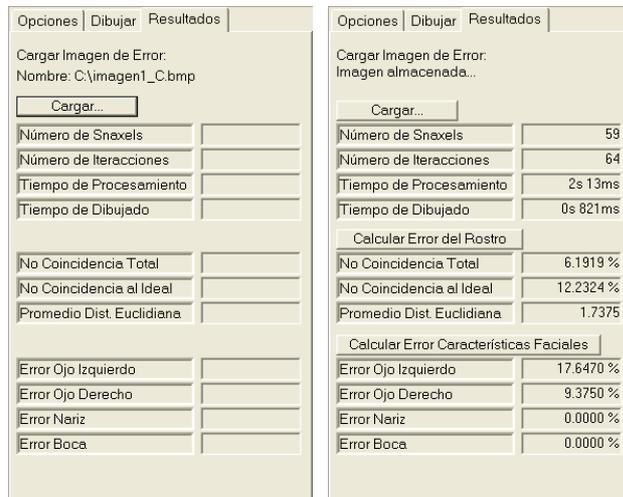
Alfa	1
Beta	1
Gama	1.2
Lambda	0.25
Límite Color	254
Curvatura	0.9

**Figura A.5:** Pestaña Opciones.

Imagen a color     Niveles de Gris  
 Separación entre ojos  
 Ancho del rostro  
 Color de la piel  
 Altura de la nariz  
 Altura de la boca  
 Altura del menton  
 Ventana Ojos  
 Ventana nariz  
 Ventana boca  
 Snaxels

**Figura A.6:** Pestaña Dibujar.

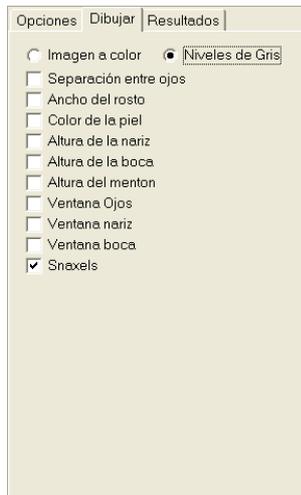
- **Resultados.** Esta pestaña es de información, aquí se muestran las mediciones de error involucradas con la detección automática del contorno del rostro y de las características faciales, además del número de *snaxels*, el número de iteraciones de convergencia del contorno activo, el tiempo de procesamiento y el tiempo de dibujado. Para poder generar la medición se requiere (como previamente se señaló) de una imagen de entrada con el contorno deseado y el conjunto de líneas horizontales que abarcan los rasgos faciales, ambos insertados manualmente, dicha imagen se buscará dentro de la misma ruta de la imagen de entrada y con el mismo nombre pero con la terminación ``\_C`` adjuntada al final; para efectos de este trabajo esta imagen se denominó como *imagen de error*. Ejemplo, imagen de entrada: *c : \1.bmp*; imagen de error: *c : \1.C.bmp* (ver Figura A.7(a)). Para almacenar esta imagen en memoria se presiona el botón *Cargar ...*, si la imagen con el contorno manual se encuentra aparecerá la leyenda ``Imagen almacenada`` y dos botones más: *Calcular Error del Rostro* y *Calcular Error Características Faciales* (ver Figura A.7(b)). Para poder realizar las mediciones de error, es necesario que la imagen resultante de la detección automática sea dibujada con las opciones de dibujado (pestaña *Dibujar*) *Niveles de Gris* y *Snaxels* (ver Figura A.8).



(a) nombre de la imagen manual.

(b) imagen de error en-contrada.

**Figura A.7:** Pestaña Resultados.



**Figura A.8:** Condiciones de dibujado necesarias para la medición de errores.

Por otro lado, los mensajes de error que se podrían producir durante el procesamiento de la imagen de entrada ocurren cuando:

1. No se detecta color de piel en la imagen (ver Figura A.9(a)).
2. No se detecta el punto inicial de la silueta (ver Figura A.9(b)).
3. La posición de los ojos tiene un valor igual a cero o se encuentra fuera de los márgenes de la imagen, también una mala localización del centro de los ojos produce que la primera aproximación al ancho del rostro se encuentre fuera de los márgenes de la imagen (ver Figura A.9(c)).
4. La posición de la nariz o la boca se encuentra fuera de los márgenes de la imagen o el valor sea cero para alguno de ellos (ver Figura A.9(d)).
5. Durante el proceso de localización del ancho del rostro, éste se encuentre fuera de los límites de la imagen (ver Figura A.9(e)).

Para cualquiera de los casos, no se podrá continuar con el resto del análisis.



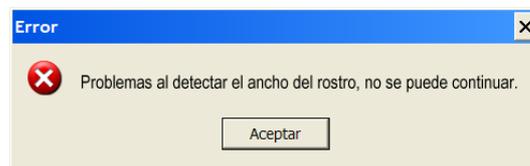
(a)

(b)



(c)

(d)



(e)

**Figura A.9:** Mensajes de error.

# Bibliografía

- [1] Jain K. Anil. *Fundamentals of Digital Image Processing*. Prentice Hall Information and System Sciences Series. Thomas Kailath, Editor, 1986.
- [2] Andrew Blake and Michael Isard. *Active contours*. Springer-Verlag, London, U.K., 1998. <http://research.microsoft.com/~ablake/contours/>, último acceso 7 de Julio del 2007.
- [3] Kenneth R. Castleman. *Digital Image Processing*. Prentice Hall, Upper Saddle River, NJ 07458, 1996.
- [4] H. D. Cheng, X. H. Jiang, Y. Sun, and Jing Li Wang. Color image segmentation: Advances and prospects. *Pattern Recognition*, 34(12):2259–2281, December 2001.
- [5] Laurent D. Cohen. On active contour models and balloons. *CVGIP: Computer Vision, Graphics, and Image Processing. Image Understanding*, 53(2):211–218, 1991.
- [6] Laurent D. Cohen and Isaac Cohen. Finite-element methods for active contour models and balloons for 2-d and 3-d images. *PAMI: Pattern Analysis and Machine Intelligence*, 15(11):1131–1147, November 1993.
- [7] Jeffrey F. Cohn and Takeo Kanade. Use of automated facial image analysis for measurement of emotion expression. In J. A. Coan and J. B. Allen, editors, *The handbook*

- of Emotion Elicitation and Assessment*. Oxford University Press Series in Affective Science, New York: Oxford, 2006.
- [8] J. B. Dowdall, I. Pavlidis, and G. Bebis. Face detection in the near-ir spectrum. *Image and Vision Computing*, 21(7):565–578, July 2003. [http://www.cpl.uh.edu/html/publications/invi\\_conf\\_papers/i1.pdf](http://www.cpl.uh.edu/html/publications/invi_conf_papers/i1.pdf), último acceso 7 de Julio del 2007, última modificación 10 de Noviembre del 2006.
- [9] G. Gomez, M. Sanchez, and L. E. Sucar. On selecting an appropriate colour space for skin detection. In *MICAI '02: Proceedings of the Second Mexican International Conference on Artificial Intelligence*, pages 69–78, London, UK, 2002. Springer-Verlag.
- [10] Xiaoyun Gong and Richard Yager. Boundary extraction using active contour models: Snake alive! Html, April 1999. <http://vislab.cs.vt.edu/Presentations/snakesBE/index.htm>, último acceso 7 de Julio del 2007, última modificación 22 de Abril de 1999.
- [11] Rafael C. González and Richard E. Woods. *Tratamiento Digital de Imágenes*. Addison-Wesley, 1996.
- [12] Steve R. Gunn and Mark S. Nixon. Global and local active contours for head boundary extraction. *International Journal of Computer Vision*, 30(1):43–54, 1998.
- [13] Erik Hjelmås and Boon Kee Low. Face detection: A survey. *Computer Vision and Image Understanding*, 83(3):236–274, 2001.
- [14] Spiros Ioannou, Amaryllis Raouzaiou, Kostas Karpouzis, and Stefanos D. Kollias. Adaptation of facial feature extraction and rule generation in emotion-analysis systems. In *International Joint Conference on Neural Networks (IJCNN 2004)*, Budapest, Hungary, 2004.

- [15] Ramesh Jain, Rangachar Kasturi, and Brian G. Schunck. *Machine Vision*. Computer Science Series. McGraw-Hill International Editions, Singapur, 1995.
- [16] Oliver Jesorsky, Klaus J. Kirchberg, and Robert W. Frischholz. Robust face detection using the hausdorff distance. In Josef Bigun and Fabrizio Smeraldi, editors, *Third International Conference on Audio-and-Video-Based Biometric Person Authentication*, pages 90–95, London, UK, 2001. Springer-Verlag.
- [17] Winnie Jiang and Kathy H. Y. Fung. Active contour for object delineation. Project Final Report 2004 Part IV, The University of Auckland, 2004.
- [18] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, January 1987. (Marr Prize Special Issue). <http://www.cs.toronto.edu/~dt/>, último acceso 7 de Julio del 2007.
- [19] Rubén Medina and Jesús Bellera. Bases del procesamiento de imágenes médicas. Taller de Telemedicina e Informática Médica, Abril 2001. Universidad de Los Andes, Facultad de Ingeniería, Grupo de Ingeniería Biomédica de la ULA (GIBU-LA). [http://www.saber.ula.ve/redtelemedicina/TallerTelemedicina/j\\_bellera-01.pdf](http://www.saber.ula.ve/redtelemedicina/TallerTelemedicina/j_bellera-01.pdf), último acceso 17 de Agosto del 2007.
- [20] John Miano. *Compressed Image File Formats*. SIGGRAPH Books Series with ACM Press Books. Addison Wesley, New York, 2nd edition, January 2000 2000.
- [21] Megherbi Najla and Ambellouis Sébastien. A genetic snake model to automatic human face and head boundary extraction. *7th International Symposium On Programming and Systems*, pages 289–296, 2005.

- [22] Athanasios Nikolaidis and Ioannis Pitas. Facial feature extraction and pose determination. *Pattern Recognition*, 33(11):1783–1791, November 2000. <http://citeseer.ist.psu.edu/cache/papers/cs/9153/http%3A%3Aposeidon.csd.auth.gr%3Apapers%3A%3APUBLISHED%3AJOURNAL%3ANikolaidisA99az%3ANikolaidisA99a.pdf/facial-feature-extraction-and.pdf>, último acceso 7 de Julio del 2007.
- [23] Peter Peer, Jure Kovac, and Franc Solina. Human skin colour clustering for face detection. *EUROCON 2003: International Conference on Computer as a Tool.*, 2:144–148, September 2003.
- [24] Kun Peng, Liming Chen, Su Ruan, and Georgy Kukharev. A robust algorithm for eyes detection on gray intensity face without spectacles. *Journal of Computer Science and Technology*, 5(3):127–132, October 2005. <http://journal.info.unlp.edu.ar/Journal/journal15/papers/JCST-Oct05-3.pdf>, último acceso 3 de Agosto del 2007.
- [25] Vytautas Perlibakas. Automatical detection of face features and exact face contour. *Pattern Recognition Letters*, 24(16):2977–2985, December 2003. <http://www.busim.ee.boun.edu.tr/~cdemirkir/files/FacialFeatureDetection/Automatical%20detection%20of%20face%20features%20and%20exact%20face%20contour.pdf>, último acceso 7 de Julio del 2007, última modificación 8 de Marzo del 2005.
- [26] Lazaros Polymenakos. 2d face tracking. Research and education society in information technologies, CHIL (Computers in the Human Interaction Loop) Technology Catalogue Overview, Athens, Greece, 2006. [http://chil.server.de/servlet/is/9454/TC\\_2DFaceTracking\\_AIT.pdf?](http://chil.server.de/servlet/is/9454/TC_2DFaceTracking_AIT.pdf?)

command=downloadContent&filename=TC\_2DFaceTracking\_AIT.pdf, último acceso 7 de Julio del 2007.

- [27] Petia Radeva, Joan Serrat, and Enric Martí. A snake for model-based segmentation. In *ICCV'95 International Conference on Computer Vision*, pages 816–821, Los Alamitos, CA, USA, June 1995. IEEE Computer Society. <http://citeseer.ist.psu.edu/cache/papers/cs/2737/http://zSzzSzwwww.cvc.uab.eszSzzCz7EpetiazSziccv95.pdf/radeva95snake.pdf>, último acceso 13 de Julio del 2007.
- [28] Frank Y. Shih and Chao-Fa Chuang. Automatic extraction of head and face boundaries and facial features. *Information Sciences-Informatics and Computer Science: An International Journal*, 158(1):117–130, January 2004. <http://www.busim.ee.boun.edu.tr/~cdemirkir/files/FacialFeatureDetection/Automatic%20extraction%20of%20head%20and%20face%20boundaries%20and%20facial%20features.pdf>, , último acceso 7 de Julio del 2007, última modificación 8 de Marzo del 2005.
- [29] Zoltán Szlávik and Tamás Szirányi. Face analysis using cnn-um. In *Cellular Neural Networks and Their Applications*, pages 190–195., Budapest, 2004. Proceedings of the 8th IEEE International Workshop Budapest (CNNA 2004).
- [30] Konstantinos N. Plataniotis and Anastasios N. Venetsanopoulos. *Color Image Processing and Applications*. Springer, Editor, 2000. (Digital Signal Processing).
- [31] Vladimir Vezhnevets, Vassili Sazonov, and Alla Andreeva. A survey on pixel-based skin color detection techniques. *Graphicon-2003*, pages 85–92, September 2003. Moscow, Russia. <http://graphics.cs.msu.su/en/publications/text/gc2003vsa.pdf>, último acceso 7 de Julio del 2007.

- [32] Donna J. Williams and Mubarak Shah. A fast algorithm for active contours and curvature estimation. *Computer Vision, Graphics, and Image Processing (CVGIP): Image Understanding*, 55(1):14–26, 1992.
- [33] Ming-Hsuan Yang, David J. Kriegman, and Narendra Ahuja. Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, January 2002. <http://vision.ai.uiuc.edu/mhyang/papers/pami02a.pdf#search=%22detecting%20faces%20in%20images%20a%20survey%22>, último acceso 7 de Julio del 2007, última modificación 5 de Septiembre del 2002.
- [34] David Young. Active contour models (snakes). HTML, March 1995. [http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL\\_COPIES/YOUNG/vision7.html](http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/YOUNG/vision7.html), último acceso 7 de Julio del 2007.