



**UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FISICAS Y MATEMATICAS
DEPARTAMENTO DE INGENIERÍA ELECTRICA**

**ESTUDIO DE MÉTODOS PARA DETECCIÓN Y SEGUIMIENTO DE ROSTROS
FRONTALES EN IMÁGENES DIGITALES: TRANSFORMADA ELÍPTICA
LINEAL Y MOLDES PSO**

**TESIS PARA OPTAR AL GRADO DE MAGÍSTER EN CIENCIAS DE LA
INGENIERÍA, MENCIÓN INGENIERÍA ELÉCTRICA**

**MEMORIA PARA OPTAR AL TITULO DE INGENIERO
CIVIL ELECTRICISTA**

CARLOS MARCELO ARAVENA CERECEDA

**PROFESOR GUÍA:
CLAUDIO PÉREZ FLORES**

**MIEMBROS DE LA COMISIÓN:
PABLO ESTEVEZ VALENCIA
CLAUDIO HELD BARRANDEGUY
PABLO ZEGERS FERNANDEZ**

**SANTIAGO DE CHILE
ABRIL 2009**

RESUMEN DE LA TESIS PARA OPTAR AL GRADO
DE MAGÍSTER EN CIENCIAS DE LA INGENIERÍA,
MENCION INGENIERÍA ELÉCTRICA Y AL TÍTULO
DE INGENIERO CIVIL ELECTRICISTA
POR: CARLOS MARCELO ARAVENA CERECEDA
FECHA: 07/03/2009
PROF. GUÍA: DR. CLAUDIO PÉREZ F.

**ESTUDIO DE MÉTODOS PARA DETECCIÓN Y SEGUIMIENTO DE ROSTROS
FRONTALES EN IMÁGENES DIGITALES: TRANSFORMADA ELÍPTICA LINEAL
Y MOLDES PSO**

La detección y localización de rostros es una de las áreas de investigación más activa en el ámbito del procesamiento digital de imágenes, debido a que es un componente principal para muchos sistemas de identificación y seguimiento de rostros, así como también para aplicaciones con interfaces hombre-máquina. Esto, apoyado por las predicciones de un fuerte crecimiento en el mercado biométrico mundial, hacen que sea de gran relevancia la búsqueda de nuevos métodos de detección de rostros, más rápidos y precisos, que se adapten de manera robusta a ambientes y condiciones complejas. En particular, está demostrado que las aplicaciones de reconocimiento de rostros dependen en gran medida de la exactitud con que se realiza su localización, motivo por el cual, es necesario contar con métodos que permitan un mayor grado de certeza en la estimación de la posición del rostro.

En esta tesis, se estudiaron dos nuevos métodos que apuntan a resolver de mejor manera este problema. El primero de ellos permite obtener información precisa acerca del rostro, como son su posición, excentricidad, tamaño y rotación coronal. Este método, basado en la Transformada Elíptica Lineal (LET), asocia una elipse al rostro y funciona transformando el contorno semi elíptico inferior de la cara en un borde vertical en el espacio de salida. La detección de dicho borde, permite estimar los parámetros de la elipse que se ajusta de mejor manera al rostro. La metodología fue probada sobre las bases de datos de Purdue, Caltech, FERET y Yale-B, obteniendo tasas de localización 98.2 %, 95.3 %, 95.7 % y 79.6 % respectivamente.

La segunda parte de este trabajo consistió en estudiar la optimización de moldes direccionales mediante la utilización de enjambres de partículas (*Particle Swarm Optimization*). La optimización tuvo por objetivo el mejorar el desempeño y selectividad de los moldes y el disminuir el tiempo de procesamiento, a la vez de estudiar el efecto de generar moldes personalizados para un individuo o situación particular. Los resultados muestran una leve mejora sobre los moldes originales de un 2 % a 3 % en las tasas de localización, pero con una reducción de 40 % en el tiempo de cálculo y una mucho mayor selectividad, lo que los hace más robustos. Las tasa de localización sobre la base de datos de Yale-B es de aproximadamente un 82 %. También, se probó la eficacia del método al ser usado en conjunto con un detector de iris para estimar la posición de los ojos en imágenes estáticas de la base de datos de Purdue, obteniéndose tasas de detección de iris por sobre el 98 % para criterios de error muy exigentes.

Ambas metodologías fueran comparadas con el detector internacionalmente utilizado propuesto por Viola y Jones, obteniendo tasas de localización muy similares. En el caso de Yale-B los métodos propuestos en esta Tesis son ampliamente superiores, demostrando su robustez frente a condiciones extremas de iluminación y contraste.

*A Dios, creador del Universo y creador del aún más
fascinante mundo de la inteligencia humana...*

*A mi tata Lolo, por compartir mis sueños y haberme
acompañado en espíritu a lo largo de este camino...*

Agradecimientos

En primer lugar, quisiera agradecer a quienes son las personas más importantes en mi vida: a mis padres Carlos y Malva, por su infinito apoyo durante toda mi vida. Gracias por todos los sacrificios que han realizado para poder brindarme la oportunidad de obtener esta carrera. Los quiero muchísimo y estoy muy orgulloso de tener unos papás como ustedes. A mi hermano Pablo, mi mejor amigo, discípulo y maestro. Gracias por todos los momentos vividos, por tu ayuda en los momentos difíciles y tu alegría en los momentos felices. Te quiero mucho.

Quisiera agradecer también a todo el resto de mi familia. A mis tíos y abuelos, por acompañarme y otorgarme su cariño desde que tengo memoria. A mis primos y primas, por vivir conmigo numerosos e inolvidables momentos de alegría. Quisiera agradecer especialmente a mi abuelita Sara, por acogerme en su casa durante casi toda mi carrera. Gracias por abrirme su hogar y hacerme sentir en familia.

Quisiera agradecer a mis numerosos amigos, antiguos y algunos nuevos, por su muy sincero cariño y apoyo. A mis amigos del norte, por su ya inquebrantable amistad y compañerismo durante muchos años de mi vida. A mis amigos de la U, por los muy gratos momentos vividos dentro y fuera de la Facultad. A mis compañeros del laboratorio, por todas las anécdotas, enseñanzas y momentos alegres entregados durante el desarrollo de mi tesis. Un saludo especial también para mi niña. Gracias por todo tu apoyo, cariño y alegría, y por ser mi inspiración durante todo este último año de trabajo.

Doy las gracias a mi profesor guía, Claudio Pérez, por darme la oportunidad de trabajar con él en esta tesis. Gracias por todos los consejos, tanto académicos como personales.

Esta tesis fue financiada por los proyectos FONDECYT 1080593 y 1040693.

Índice general

1. Introducción	1
1.1. Objetivo General	3
1.2. Objetivos Específicos	3
1.3. Estructura del documento	4
2. Antecedentes	6
2.1. Métodos de detección de rostros	6
2.1.1. Métodos basados en Conocimiento	7
2.1.2. Métodos basados en Características Invariantes	8
2.1.3. Métodos basados en Moldes	9
2.1.4. Métodos basados en Apariencia	9
2.1.5. Revisión de métodos utilizados en esta Tesis	10
2.1.5.1. Detector de Viola y Jones (Adaboost)	10
2.1.5.2. Templates direccionales	15
2.2. Bases de datos utilizadas	19
2.2.1. Caltech	19
2.2.2. The AR face database (Purdue)	19
2.2.3. FERET	19
2.2.4. YALE B	20
2.3. Criterios de medición	20
2.3.1. Criterio de detección de rostros	20
2.3.2. Error en la estimación de parámetros	21
2.3.3. Criterio de localización de iris	22
2.4. Librería OpenCV para procesamiento de imágenes	23
3. La Transformada Elíptica Lineal (LET)	25
3.1. La Transformada Elíptica Lineal	25
3.2. Metodología	28
3.2.1. Pre-procesamiento de la imagen	30

3.2.2.	Etapa de detección gruesa	30
3.2.3.	Estimación de los parámetros del rostro	31
3.3.	Procesamiento de Secuencias de Video	34
3.4.	Pruebas Realizadas	36
4.	Optimización de Moldes Direccionales mediante PSO	39
4.1.	Generación de moldes PSO	40
4.1.1.	Preprocesamiento	41
4.1.2.	Inicialización de los moldes	44
4.1.3.	Evaluación	45
4.1.4.	Modificación de las partículas	47
4.1.5.	Multiresolución	48
4.2.	Pruebas realizadas	48
4.2.1.	Secuencias de video	48
4.2.2.	Base de datos Yale B	49
4.2.3.	Base de datos PURDUE	50
5.	Resultados Experimentales	52
5.1.	Transformada Elíptica Lineal (LET)	52
5.1.1.	Selectividad de los parámetros de la LET	52
5.1.2.	Estimación de parámetros	56
5.1.3.	Tasas de detección de rostros	57
5.1.4.	Tiempo de procesamiento	62
5.1.5.	Tipos de error detectados	62
5.2.	Moldes PSO	62
5.2.1.	Estimación del Tamaño del Rostro	66
5.2.2.	Integral de línea	68
5.2.3.	Selectividad Espacial	68
5.2.4.	Tasas de Detección de Rostros	73
5.2.4.1.	Secuencias de Video	73
5.2.4.2.	Base de Datos Yale B	73
5.2.5.	Detección de Iris	75
6.	Conclusiones	76
	Bibliografía	81

Anexos	85
A. Imagen Direccional	86
B. Método de Deteccion Gruesa (Maio y Maltoni)	89
C. Detección de Iris	92

Capítulo 1

Introducción

La detección y localización de rostros es una de las áreas de investigación más activa en el ámbito del procesamiento digital de imágenes, debido a que es un componente principal para muchos sistemas de identificación y seguimiento de rostros, estimación de la posición del iris, reconocimiento de gesticulaciones y estados de ánimo, entre otros. Diversas aplicaciones de interfaces hombre-máquina han sido desarrolladas, en áreas tales como vigilancia por video, verificación de identidad, sistemas de control de acceso, o en sistemas de ayuda para individuos con discapacidad o de guía visual para el control de dispositivos en ambientes tecnológicos complejos, tales como cabinas de aviones, salas de operaciones en hospitales o unidades de control en ambientes industriales [1–5].

La importancia en la investigación y desarrollo de estas aplicaciones se ve apoyada en las predicciones de un fuerte crecimiento en el mercado biométrico mundial para los próximos años (figura 1.1), especialmente en productos basados en reconocimiento de rostros o que lo integren en conjunto con otras técnicas de identificación biométrica [6, 7].

Los métodos existentes para detección y localización de rostros han sido previamente clasificados en cuatro categorías no necesariamente excluyentes entre sí [8]: métodos basados en conocimiento [9, 10], métodos basados en características invariantes [11–14], utilización de moldes (“*templates*” por su denominación anglosajona) [15–17] y métodos basados en apariencia [18–23]. A pesar de la popularidad que poseen los de esta última categoría (ej. Adaboost), los métodos basados en apariencia están diseñados principalmente para detección de uno o varios rostros, pero a menudo carecen de precisión para estimar su localización y/o tamaño. Varios de estos métodos tampoco ofrecen información acerca de la rotación o excentricidad del rostro [8]. Algunos métodos basados en *templates* han presentado resultados satisfactorios en aplicaciones de localización y seguimiento de rostros, incluso en casos de rostros con rotación

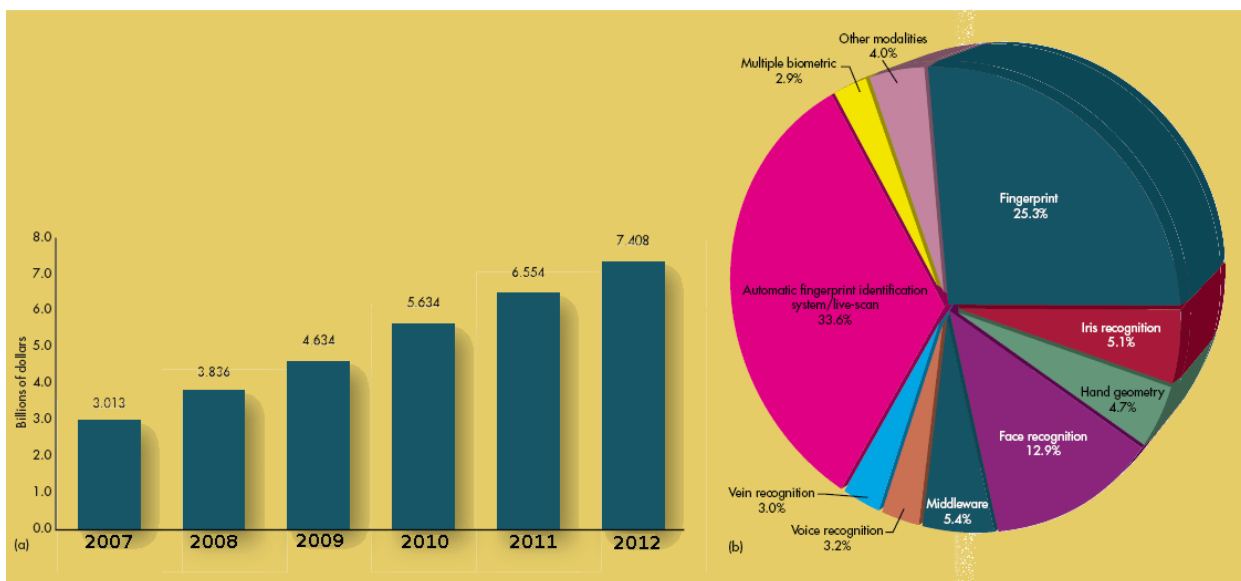


Figura 1.1: Predicción del mercado mundial de aplicaciones biométricas para el período 2007 - 2012. International Biometric Group.

coronal y no coronal [24].

Los métodos referidos han presentado resultados satisfactorios en ambientes relativamente controlados y/o para un subconjunto reducido de personas. Las principales dificultades que se presentan en el desarrollo de estas técnicas en ambientes reales son: fondo no homogéneo y cambiante en el tiempo, distinta luminosidad, cambios de escala en el o los sujetos (por variaciones en la distancia a la cámara), rotaciones excesivas, oclusiones, variabilidad del color de piel, incapacidad de detectar los ojos (por uso de anteojos o lentes de sol), etc. Además muchas de las técnicas no son aplicables en tiempo real con un computador personal. Está demostrado, asimismo, que las aplicaciones de reconocimiento de rostros dependen en gran medida de la exactitud con que se realiza la localización del rostro [25]. Es por estas razones, que se hace importante la búsqueda de nuevas metodologías que por un lado apunten a una mayor efectividad en la detección de los rostros y que, por otro, permitan un mayor grado de certeza en la estimación de la localización del rostro. Es deseable también que estos métodos puedan ser aplicados satisfactoriamente en ambientes no controlados y en tiempo real.

En esta tesis, se estudiaron dos nuevos métodos que apuntan a resolver de mejor manera el problema de la localización de rostros. El primero de ellos permite obtener información precisa acerca del rostro, como son su posición, excentricidad, tamaño y rotación coronal. Este método, basado en la Transformada Elíptica Lineal (LET) (creada por Gárate en su trabajo de Tesis de Magíster [26]), transforma el contorno semi elíptico inferior del rostro en un borde vertical en el espacio de salida. La metodología fue probada sobre cuatro bases de

datos internacionales, y sobre varias secuencias de video de individuos con rotación coronal, mostrando buenos resultados en la estimación de los parámetros antes señalados.

La segunda parte de este trabajo consistió en estudiar la optimización de moldes direccionales mediante la utilización de enjambres de partículas (*Particle Swarm Optimization* - PSO). Dichos moldes han mostrado resultados satisfactorios en el seguimiento de rostros con rotación coronal y la optimización apunta a mejorar su desempeño y a disminuir el tiempo de procesamiento. El método fue probado en bases de datos internacionales y sobre secuencias de video de individuos del Departamento de Ingeniería Eléctrica. También, se probó la eficacia del método al ser usado en conjunto con un detector de iris para estimar la posición de los ojos en imágenes estáticas.

Ambas metodologías fueron comparadas con el detector de rostros basado en Adaboost, propuesto por Viola y Jones en [27]. Este método fue elegido por sus numerosas implementaciones, probada eficacia y por ser uno de los más utilizados en aplicaciones alrededor del mundo. Para el caso de los moldes PSO, la aplicación de detección de iris fue comparada con un método que combina información de intensidad y bordes binarios [25].

1.1. Objetivo General

El objetivo general de este trabajo es el estudio de dos nuevas metodologías de detección y seguimiento de rostros que son: la Transformada Elíptica Lineal (LET) que permite independizar la localización del rostro de sus características antropométricas y el uso de moldes direccionales optimizados por Particle Swarm Optimization (PSO).

El estudio implica la implementación y la medición de la efectividad de ambos métodos en la determinación de la posición, dirección, excentricidad y tamaño del rostro en relación a otras metodologías desarrolladas previamente, utilizando criterios de medición y bases de datos estandarizados. El proceso involucra que la detección sea en tiempo real, es decir, sea capaz de detectar y seguir el rostro en una secuencia de video estándar (NTSC). El estudio involucra la optimización computacional del método y la adición de características que apunten a mejorar su desempeño.

1.2. Objetivos Específicos

Los Objetivos Específicos son:

- Definir, estudiar e implementar el método basado en la Transformada LET.

- Estudiar e implementar la detección de rostros basada en Templates PSO.
- Implementar el detector de iris utilizado en [28], basado en moldes circulares que buscan la frontera iris-esclera.
- Realizar mediciones con bases de datos estándar, que permitan obtener el real desempeño de las metodologías, en cuanto al error en la determinación del centro del rostro y su tamaño.
- Comparar los resultados con los obtenidos por otras técnicas desarrolladas a nivel mundial, en este caso, el detector desarrollado por Viola y Jones [27] y el detector de iris desarrollado por [25].
- Analizar los resultados de manera de obtener información acerca de los rangos de aplicación, casos límites, ventajas y desventajas de ambos métodos bajo diversas condiciones de operación.
- Optimizar las metodologías de manera de aumentar su desempeño computacional y/o aumentar su precisión.

1.3. Estructura del documento

Esta Tesis se encuentra dividida en los siguientes capítulos:

- En el Capítulo 2 se incluyen los antecedentes teóricos necesarios para contextualizar el trabajo realizado. Se explican las principales técnicas de detección de rostros, explicando en mayor profundidad los métodos de detección mediante templates direccionales [15, 24, 28–30] y el popular detector propuesto por Viola y Jones [27], basado en Adaboost y que es usado como método principal de comparación de las metodologías utilizadas. En esta sección también se explican los criterios de medición de desempeño investigados y se incluye información acerca de las bases de datos utilizadas. Por último, se ha agregado una breve reseña de la librería OpenCV, utilizada para implementar las distintas metodologías de esta Tesis.
- En el Capítulo 3 se explica en detalle la metodología de detección mediante la Transformada Elíptica Lineal (LET). En una primera parte se introduce la LET y sus propiedades de selectividad de los parámetros del rostro. Posteriormente se detalla la metodología de aplicación de la LET para la detección de rostros frontales con rotación coronal, tanto en

imágenes fijas, como las hipótesis realizadas para su aplicación en la detección sobre imágenes de video. Finalmente se explican las pruebas realizadas para medir el desempeño de este método.

- En el Capítulo 4 se presenta un nuevo método, que consiste en la optimización mediante PSO de los moldes direccionales utilizados en [15, 28]. El capítulo se centra en la optimización de dichos moldes, mientras que la metodología de aplicación de dichos moldes a la detección de rostros es explicada brevemente en el capítulo 2 y en el material bibliográfico señalado. Al final del capítulo se detallan las pruebas realizadas y los criterios de medición utilizados.
- El Capítulo 5 se encuentra dividido en dos partes. En la primera sección se explican y discuten los resultados de las pruebas realizadas sobre la metodología mediante Transformada LET. En la segunda parte se exponen los resultados y análisis de la utilización de los templates PSO para la detección de rostros
- El Capítulo 6 expone las conclusiones finales de este trabajo de Tesis, comparando ambos métodos y destacando sus fortalezas, debilidades y dominios de aplicación. Asimismo, se delimitan futuras líneas de investigación que se puedan seguir a partir de este trabajo.
- Finalmente se incluye por razones de completitud, secciones de anexos con información relevante para la aplicación de estas metodologías, como son una mayor explicación de los métodos de detección gruesa del rostro y de la detección de iris.

Capítulo 2

Antecedentes

2.1. Métodos de detección de rostros

Un estudio exhaustivo de los métodos de detección de rostros fue realizado por Yang et al. en [8]. En él se explica de manera bastante clara las tendencias actuales de investigación en torno al problema de detección de rostros, en especial en cuanto a las distintas categorías de soluciones, criterios de medición, bases de datos utilizadas y también se resumen los principales problemas y desafíos que deben enfrentar las metodologías actuales.

Entre las principales dificultades que se pueden observar en la aplicación de los distintos métodos se encuentran:

- *Variabilidad en la Pose:* Las imágenes del rostro varían según la pose frente a la cámara, por lo que algunas características faciales pueden quedar ocluidas (presencia de un solo ojo, oreja, distinta relación espacial entre características del rostro, etc.)
- *Presencia o ausencia de componentes estructurales:* Características faciales tales como barba, bigote y lentes, al estar o no estar presentes, generan una gran variabilidad en el color, forma y tamaño de los conjuntos de rostros.
- *Expresión Facial:* La apariencia del rostro es directamente afectada por la expresión (gesticulación) del individuo.
- *Oclusión:* Los rostros pueden estar ocluidos parcialmente por distintos objetos como anteojos de sol, gorros y bufandas. También se observan problemas de oclusión al aplicar la detección a multitudes de personas (oclusión entre caras).

- *Adquisición de la Imagen:* Condiciones tales como la iluminación, tipo de sensor de la cámara, resolución, velocidad de captura, etc., afectan la apariencia del rostro.
- *Tiempo de procesamiento:* Si bien el avance en la capacidad de los computadores actuales minimizan este problema, aún existen barreras tecnológicas en cuanto a desplegar ciertos métodos en aplicaciones de tiempo real.

Los distintos métodos de detección de rostro pueden ser más o menos robustos que otros para resolver alguno de los problemas antes mencionados. De ahí la existencia de distintos enfoques que permiten abordar de mejor manera algunas de estas dificultades. El trabajo, antes citado, de Yang et al. [8], divide a los métodos de detección de rostros en las siguientes categorías no excluyentes entre sí:

1. *Métodos basados en Conocimiento:* Se basan en codificar el conocimiento humano mediante reglas. Generalmente, dichas reglas capturan relaciones entre características faciales.
2. *Métodos basados en Características Invariantes:* Buscan características que no son modificadas a pesar de que el rostro esté sometido a cambios de iluminación, pose y/o ubicación de la cámara, las cuales son utilizadas para la ubicación del rostro.
3. *Métodos basados en Templates:* Utilizan la correlación entre una imagen de entrada y patrones almacenados que describen un rostro.
4. *Métodos basados en Apariencia:* Utilizan modelos obtenidos mediante un conjunto de entrenamiento de imágenes, los cuales capturan una representación variable de la apariencia facial (ej. el detector de Viola y Jones [27])

2.1.1. Métodos basados en Conocimiento

Se basan en capturar las reglas expertas que supuestamente usan las personas para etiquetar una imagen dada como un rostro. Estas reglas son obtenidas del conocimiento que se tenga acerca de como está constituido el rostro de una persona y describen las características de una cara y sus relaciones. Una cara, por ejemplo, puede representarse por las distancias relativas y posiciones en que aparecen sus componentes principales y así saber que una imagen con dos ojos que son simétricos entre ellos, una boca y una nariz, es probablemente un rostro. Por lo general, el algoritmo consiste en extraer las características faciales de la imagen de entrada, para luego obtener los candidatos a rostro mediante un conjunto de reglas. Posteriormente, algún proceso de verificación es usado para reducir las detecciones falsas.

La principal dificultad de este tipo de métodos es diseñar un conjunto de reglas bien definidas que interpreten de la mejor manera posible el conocimiento humano. Reglas muy estrictas, por lo general, tienden a eliminar todas las muestras presentadas, teniendo como resultado el que no se detecte ninguna o muy pocas caras. Reglas menos precisas y por ende, más generales, tienden a generar una gran cantidad de falsos positivos. También se presenta el problema de la generalización de dicho conjunto de reglas a todas las condiciones de iluminación, pose y gesticulaciones posibles.

Chiang et. al propusieron en [9] un algoritmo basado en conocimiento con niveles de reglas jerárquicas. Las reglas de mayor nivel son descripciones generales de como luce un rostro mientras que las reglas de menor nivel son descripciones de los detalles de las componentes faciales. En el nivel mayor, todos los posibles candidatos a rostro son buscados mediante el escaneo de una ventana sobre la imagen de entrada, aplicando un conjunto de reglas en cada ubicación. En este nivel, se utiliza una imagen de menor resolución para determinar los candidatos a cara, los cuales son procesados con imágenes de mayor resolución, aplicando las reglas de menor nivel.

Kotropoulos y Pitas [10] presentaron un método basado en reglas similar al de [9], que localiza las características faciales mediante un método de proyecciones. Primero se determinan dos mínimos locales de la proyección horizontal, los cuales corresponderían al lado derecho e izquierdo de la cara. Luego se obtiene la proyección vertical y los mínimos locales son usados para determinar las ubicaciones de los labios, nariz y ojos. Estas características constituyen el candidato a cara.

2.1.2. Métodos basados en Características Invariantes

Estos métodos se basan en el supuesto en que los seres humanos son capaces de detectar rostros u objetos en general en distintas poses y condiciones de iluminación, por lo que deberían existir características en el rostro que no varían frente a estos cambios. Diversos métodos han sido diseñados de tal manera que primero, se detectan las componentes faciales tales como cejas, ojos, boca, nariz y la línea del pelo, para luego inferir la presencia de una cara. Utilizando las componentes faciales extraídas se construye un modelo estadístico para describir las relaciones y verificar la existencia de un rostro. Estos algoritmos presentan serios problemas cuando la extracción de las componentes faciales se ve dificultada por problemas de iluminación (contraste, sombras), ruido y oclusión entre otros. Entre las características más investigadas se encuentran:

- *Componentes Faciales*: utilizadas principalmente sobre imágenes en escalas de grises. Buscan relaciones estadísticas y/o probabilísticas entre los componentes faciales, co-

múnmente ojos, nariz y boca [11, 12].

- *Textura*: El rostro presenta una textura especial que la diferencia de otros objetos de interés. La ventaja de este tipo de métodos es que son más robustos a la ausencia de algún componente facial, detectando por ejemplo, caras con lentes y barba [13].
- *Color de piel*: El color de la piel humana es una característica efectiva en muchas aplicaciones, tales como detección de rostro y seguimiento de manos. La mayoría de estos métodos convierten la imagen de entrada a un espacio de representación alternativo en que sea patente la segmentación de los píxeles de piel [14].

2.1.3. Métodos basados en Moldes

Estos métodos de detección de rostros se basan principalmente en aplicar alguna medida de distancia o correlación entre la imagen de entrada y un patrón o molde (template) que se construye o define en forma previa. Este molde captura una o varias características del rostro, por ejemplo, contornos o información acerca de los bordes de los ojos, nariz, boca, cejas, etc. Estos métodos generalmente operan de manera que si la correlación (o distancia) entre alguna zona particular de la imagen y uno de los moldes supera un umbral determinado, entonces se concluye la presencia de un rostro en dicha localización.

Estos métodos poseen las ventajas de ser rápidos en su aplicación (permitiendo procesamiento en tiempo real) y son relativamente fáciles de implementar. Una desventaja sin embargo, es su generalización, ya que es muy difícil que un sólo template represente de manera correcta todas las variaciones en escala, forma y pose del rostro. Estos problemas son abordados construyendo un conjunto de moldes a múltiple resolución, escala o que representen variadas poses [15, 16]. También es común la utilización de moldes deformables que se ajustan de manera elástica al rostro de entrada [17].

2.1.4. Métodos basados en Apariencia

En general, los métodos basados en apariencia se basan en técnicas de análisis estadístico y *machine learning* para aprender características relevantes de imágenes de rostro y no-rostro. Estas características aprendidas dan la forma de los modelos de distribución o las funciones discriminantes que son utilizadas para la detección de rostro.

Muchos métodos basados en apariencia se basan en probabilidades. Una imagen o un vector de características derivado de una imagen es visto como una variable aleatoria x , y esta

variable aleatoria es caracterizada como rostro o no-rostro mediante una función de densidad de probabilidad condicional $P(x/rostro)$ o $P(x/no - rostro)$, respectivamente.

Otro enfoque utilizado consiste en buscar una función discriminante entre las clases rostro y no-rostro, tales como superficies de decisión, hiperplanos separadores o funciones umbrales. Los patrones de la imagen son, casi siempre, proyectados a un espacio de menor dimensionalidad de manera de obtener una función discriminante para clasificación, o bien formar una superficie de decisión no-lineal mediante una red neuronal multicapa (MLP). Esta reducción de dimensionalidad también tiene por objeto reducir la complejidad computacional ya que por lo general, se trabaja en espacios de muy alta dimensionalidad.

Entre los métodos basados en apariencia se encuentran: Eigenfaces [21], redes neuronales [18–20], máquinas de soporte vectorial (SVM) [23], modelos ocultos de Markov (HMM) [22], y Adaboost [27], etc.

2.1.5. Revisión de métodos utilizados en esta Tesis

2.1.5.1. Detector de Viola y Jones (Adaboost)

Viola y Jones propusieron en [20, 27] un detector de rostros basado en Adaboost [31] que permite una detección rápida y confiable de múltiples rostros frontales en una imagen digital. Esta metodología se ha convertido en una de las más utilizadas en aplicaciones de detección de rostros y e han sugerido múltiples mejoras para extender su funcionalidad o aumentar su confiabilidad. Debido a esto este detector fue elegido como uno de los métodos de comparación de los algoritmos estudiados en esta Tesis.

El método de Viola-Jones usa una colección de características para clasificar ventanas de la imagen. Un conjunto de clasificadores débiles es obtenido a partir de un conjunto de entrenamiento. Cada uno de estos clasificadores es una función simple compuesta de sumas de rectángulos, seguidas de la aplicación de un umbral. Estos clasificadores débiles son combinados mediante *boosting* para formar un único clasificador más complejo y preciso. La idea se basa en la afirmación de que varios clasificadores sencillos, cada uno de ellos con una precisión ligeramente superior a una clasificación aleatoria, pueden combinarse para formar un clasificador de mayor precisión, siempre y cuando se disponga de un número suficiente de muestras de entrenamiento. Viola y Jones se basaron en el algoritmo de Adaboost de Freund y Schapire [31] para la aplicación del *boosting*.

La Figura 2.1 describe las distintas etapas de aplicación del algoritmo de Viola y Jones para un clasificador ya entrenado. A continuación se explican las distintas etapas del algoritmo

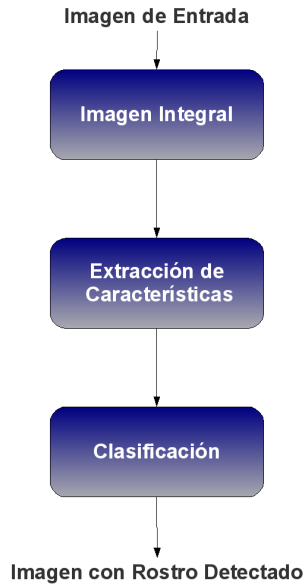


Figura 2.1: Diagrama de bloques del detector de rostros de Viola y Jones. El clasificador final viene dado por una combinación lineal de clasificadores débiles entrenada mediante el algoritmo de Adaboost.

y se describe en mayor detalle la metodología de entrenamiento del clasificador final.

Imagen Integral: Presentada en [27], esta representación de la imagen original permite extraer de forma rápida y eficiente características rectangulares a diferentes escalas (como las características de Haar [27]), ya que no se trabaja directamente con los valores de intensidad si no que con una imagen acumulativa que se construye a partir de operaciones básicas.

La imagen integral (ver figura 2.2), contiene en la posición (x, y) , la suma de los píxeles de la parte superior izquierda de la imagen y se puede calcular como se indica en 2.1:

$$II(x, y) = \sum_{x' \leq x, y' \leq y} Im(x', y'), \quad (2.1)$$

donde $II(x, y)$ es la imagen integral e $Im(x, y)$ es la imagen original.

La imagen integral es calculada sólo una vez sobre la imagen original y permite la rápida extracción de características de Haar que son utilizadas por la metodología de Viola y Jones.

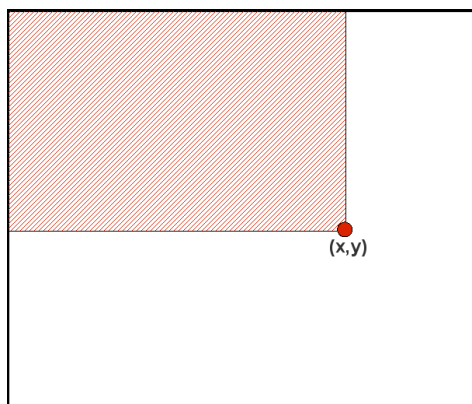


Figura 2.2: Cálculo de la Imagen Integral. La posición (x,y) contiene la suma de los valores de intensidad de todos los píxeles contenidos en el área achurada.

Extracción de Características de Haar: La extracción de características se realiza aplicando a la imagen filtros con bases Haar. Estos filtros, son selectivos en la orientación espacial y frecuencia, permiten ser modificados en escala y orientación y poseen la ventaja de poder ser calculados sobre la imagen integral de manera eficiente. Algunos ejemplos de los filtros usados en la extracción de las características de Haar son mostrados en la Figura 2.3.

Los filtros con bases Haar realizan una codificación de diferencia de intensidades en la imagen, generando características de contornos, puntos y líneas, mediante la captura de contraste entre regiones. En la figura 2.4 se ilustra el cálculo de una sub-ventana en la imagen integral. De esta operación se puede extraer una característica en un tiempo constante sobre la imagen integral adicionando y sustrayendo los valores de los vértices para cada rectángulo (2.2):

$$sumD = (4 + 1) - (2 + 3) , \quad (2.2)$$

donde 1, 2, 3, 4, son los valores que toma la imagen integral de la figura (2.4) en las posiciones marcadas por esos números.

Clasificación: Un conjunto de distintas características (o filtros de Haar) corresponden a un clasificador simple y su complejidad vendrá dada por el número de características que posea. Un conjunto de clasificadores simples en cascada constituye el detector final. Este detector final que resulta del entrenamiento previo, es escaneado por sobre la imagen variando su escala y posición (pudiendo existir traslape). Las variaciones de escala se realizan variando el tamaño del clasificador y no el de la imagen, ya que las características Haar pueden ser evaluadas a cualquier escala con tiempo constante debido al uso de la imagen integral. Todas aquellas posiciones y escalas en que una subventana de entrada sea clasificada como rostro

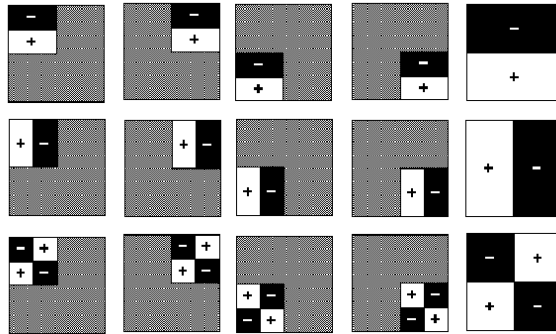


Figura 2.3: Filtros Haar con rotación, traslación y cambios de escala. El signo + corresponde a las áreas del filtro que se suman. El signo - corresponde a las áreas del filtro que se restan.

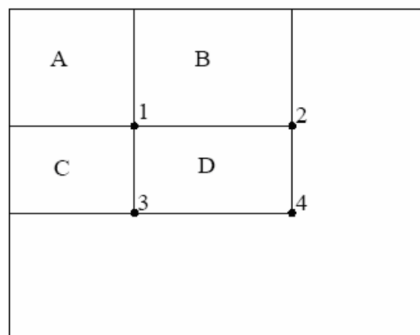


Figura 2.4: Cálculo de sub-ventanas sobre la Imagen Integral.

son firmes candidatas a ser reconocidas como tal. Sin embargo, debido a que el detector es insensible a pequeños cambios en la traslación y la escala, múltiples detecciones ocurrirán frecuentemente alrededor de un mismo rostro. Es por ello que es importante aplicar algún tipo de post-procesamiento [27] que combine las múltiples detecciones en una sola subventana de salida.

Entrenamiento: Esta es la etapa principal del algoritmo, y se realiza de forma previa a la aplicación del detector final. Permite, en primer lugar, seleccionar las mejores características discriminantes de un conjunto de miles de posibles candidatas. En segundo lugar, permite entrenar un clasificador, utilizando las características mencionadas, que decide si una región de la imagen de entrada es un rostro o no. Ambos objetivos se logran utilizando el algoritmo de Adaboost, que utiliza una combinación de clasificadores débiles simples para construir un clasificador final robusto.

En cada etapa de *boosting*, una característica se selecciona de manera de obtener el menor error ponderado posible. A esta característica se le asigna un peso en el clasificador final, según lo estipulado en el procedimiento de Adaboost. Para las etapas siguientes, se le asigna un peso mayor a las imágenes de ejemplo que sean etiquetadas de manera errónea, mientras que a las que son etiquetadas de forma correcta se le asigna un peso menor.

Para reducir la tasa de falsos positivos y mantener la eficiencia, se utiliza una cascada de clasificadores. Una ventana de entrada se envía, de un clasificador al siguiente, mientras sea clasificada como un rostro. El umbral de cada clasificador se ajusta de manera de obtener una alta tasa de detección. Los primeros clasificadores de la cascada tienen pocas características, mientras que los últimos son más complejos para descartar las imágenes “fáciles” que no son rostros, de manera temprana en la cascada. Cada clasificador en la cascada es entrenado con un conjunto de no-rostros que consiste en falsos positivos de etapas previas más un conjunto de rostros. Esto permite que los últimos clasificadores se enfoquen en los ejemplos más difíciles, aumentando la complejidad del clasificador.

Para entrenar una cascada completa y alcanzar una muy pequeña tasa de falsos positivos, se requiere un gran número de ejemplos de rostros y no-rostros, especialmente de estos últimos. Después de 5 etapas, la tasa de falsos positivos es a menudo menor a 1%. Por lo tanto, más del 99% de los ejemplos de no-rostro son descartados y se requiere adicionar nuevos ejemplos para el entrenamiento de las etapas posteriores.

El algoritmo completo de entrenamiento se describe a continuación:

Algoritmo 2.1 Algoritmo de *Boosting*

- Dado un conjunto de imágenes con sus respectivas etiquetas de clase $(x_1, y_1), \dots, (x_n, y_n)$ donde $y_i \in [0, 1]$ para muestras negativas (no-rostros) y positivas (rostros) respectivamente .
- Inicializar los pesos $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$, para $y_i = 0, 1$, donde m es el número de muestras negativas y l es el número de muestras positivas.
- Para $t = 1, \dots, T$:

1. Normalizar los pesos $w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^n w_{t,j}}$

2. Seleccionar el mejor clasificador débil con respecto al error ponderado:

$$\epsilon_t = \min_{f,p,\theta} \sum_i w_i |h(x_i, f, p, \theta) - y_i|,$$

donde $h()$ es un clasificador débil, f es su característica de Haar asociada, θ un umbral y p una paridad que indica el signo de la desigualdad que se aplica para el clasificador:

$$h(x_i, f, p, \theta) = \begin{cases} 1 & pf(x_i) < p\theta \\ 0 & \text{si no} \end{cases}$$

3. Definir $h_t(x) = h(x, f_t, p_t, \theta_t)$ donde f_t, p_t y θ_t son los que minimizan ϵ_t .

4. Actualizar los pesos:

$$w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$$

donde $e_i = 0$ si la muestra x_i es clasificada correctamente, $e_i = 1$ si no, y $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$

5. El clasificador robusto final es:

$$C(x) = \begin{cases} 1 & \sum_{t=1}^T \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^T \alpha_t \\ 0 & \text{en otro caso} \end{cases},$$

donde $\alpha_t = \log \frac{1}{\beta_t}$

2.1.5.2. Templates direccionales

Este método para detección y seguimiento de rostros rotados coronalmente, es una extensión del método propuesto previamente por Maio y Maltoni [32]. Se basa en la utilización de información direccional (bordes) de la imagen, por lo que puede ser aplicado a imágenes en escala de grises. No requiere entrenamiento previo y permite tasas de procesamiento de tiempo real. Al trabajo inicial [32] se han agregado diversas modificaciones para hacer más robusto su desempeño y extenderlo a rotaciones coronales [15, 29, 30] y no coronales [33].

La metodología de moldes direccionales cuenta con dos etapas principales: detección gruesa y detección fina del rostro. Es posible agregar una etapa adicional al trabajo original y que consiste en la detección del iris. La detección gruesa considera el cálculo de la transformada de Hough para la estimación gruesa de la posición del rostro. La detección fina utiliza la estimación gruesa de la posición del rostro para determinar el tamaño y la inclinación del rostro.

Detección Gruesa: Esta etapa tiene por objetivo la obtención de la posición aproximada del rostro, el cual se representa por un objeto elíptico de una excentricidad dada dentro del total de la imagen. Una imagen direccional, que contiene información de los bordes presentes, es barrida con un conjunto de moldes que contienen información angular de posibles elipses. Los barridos de estos moldes se suman en un acumulador, cuyo máximo representa el centro de la elipse que aproxima al rostro. Como resultado de esta etapa se obtiene una posición inicial del centro, que es luego refinada por los templates antropométricos que propone esta metodología. La detección gruesa se explica con mayor detalle en el anexo B de esta Tesis.

Moldes Antropométricos: La detección fina de la posición, inclinación y tamaño del rostro se basa en la aplicación de un conjunto de moldes elipsoidales para cada uno de los tamaños y rotaciones consideradas. Un conjunto de moldes para el rostro, a distinta escala y rotación coronal, se crean *a priori* y en forma manual, representando características antropométricas conocidas. Estos moldes contienen información vectorial (ángulo y módulo) almacenados en cada pixel. El proceso de creación de moldes se inicia con la definición del rango de tamaños a utilizar y del rango y paso de rotación coronal que considerarán dichos moldes.

Los moldes direccionales consideran la mitad inferior del rostro como un contorno elíptico. Para determinar las posiciones de las características antropométricas del rostro más destacables, se realizó un estudio estadístico en imágenes de la base de datos de Purdue [34], dando como resultado que las características direccionales más repetidas fueron: cejas, nariz, boca y trago de la aurícula. Posiciones mínimas y máximas fueron determinadas en forma porcentual con respecto al tamaño total del rostro de manera de fijar la forma final de los moldes frontales [28]. La Figura 2.5 muestra los moldes frontales generados para el rostro. Los ángulos de los moldes direccionales considerados para las características antropométricas estudiadas son: 0° para las cejas y la boca; y 90° para la nariz y el trago de la aurícula. En (a), el valor del ángulo se representa en niveles de gris, con el nivel más claro para 0° y el más oscuro para 180° . Las cejas y la boca presentan un ángulo de 0° , apareciendo de color blanco; mientras que la nariz y el trago de la aurícula tienen un ángulo de 90° . Para el borde semi-elíptico inferior del rostro, se puede apreciar la variación angular, representada en los distintos tonos de grises.

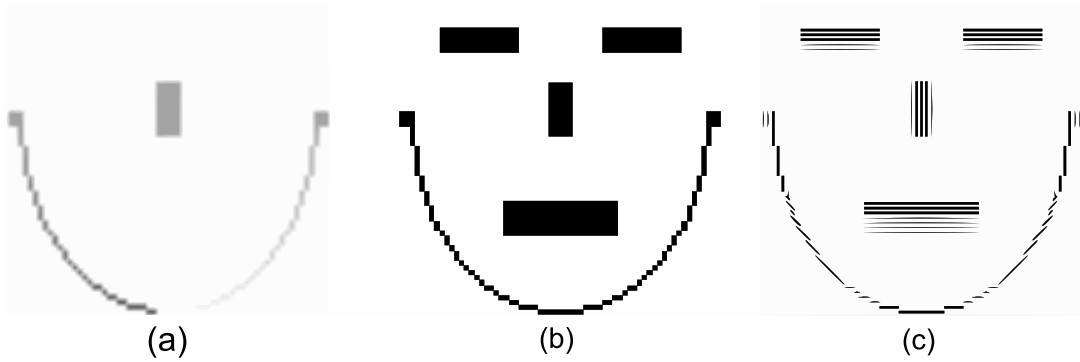


Figura 2.5: Molde direccional para rostro frontal no rotado. (a) Información angular (representada en distintos tonos de gris). (b) Módulo del molde direccional. (c) Representación gráfica de un molde direccional completo. La información angular es aquí representada según la inclinación de los vectores.

En 2.5(b) se muestra el módulo del molde, mostrando todas las características antropométricas consideradas. Dichas características presentan el mismo valor en su módulo. Finalmente en (c), se muestra una representación gráfica del molde completo considerando magnitud y ángulo para todas las características antropométricas. Los ángulos (antes codificados en tonos de grises) son representados por la inclinación de los pequeños vectores del rostro.

Los moldes con rotación coronal se generan a partir de un molde frontal, el que es rotado en el eje coronal con respecto al centro geométrico del molde, manteniendo las mismas características antropométricas. El ángulo de rotación coronal, con respecto a la horizontal, se denomina ϕ . Con esto, el ángulo de las cejas y la boca queda dado por ϕ , mientras que para la nariz y el trago de la aurícula el ángulo pasa a ser $90^\circ - \phi$.

El rango de rotación de los moldes va desde $\phi_{min} = -40^\circ$ a $\phi_{max} = +40^\circ$ con un paso de 10° . El tamaño de los moldes, medido como la altura del mismo, varía entre $[b_{min}, b_{max}]$. La Figura 2.6 muestra los moldes coronales generados para diferentes ángulos de rotación para un tamaño de rostro dado. Se muestra la magnitud (a-f), el ángulo (g-l) y el molde coronal completo (m-r), generados para los casos de rotación coronal de -30° , -20° , -10° , 10° , 20° y 30° .

Integral de línea: Como en la gran mayoría de los métodos que utilizan *templates*, es necesario definir una medida de distancia o correlación entre la imagen de entrada y el(los) molde(s) que se apliquen sobre ella. En este caso, la medida utilizada se denomina *integral de línea* que considera las diferencias angulares, punto a punto, entre la imagen direccional fina de la imagen de entrada y el molde antropométrico del rostro. La imagen direccional fina (Anexo A) contiene información angular de los bordes presentes en la imagen, codificada de

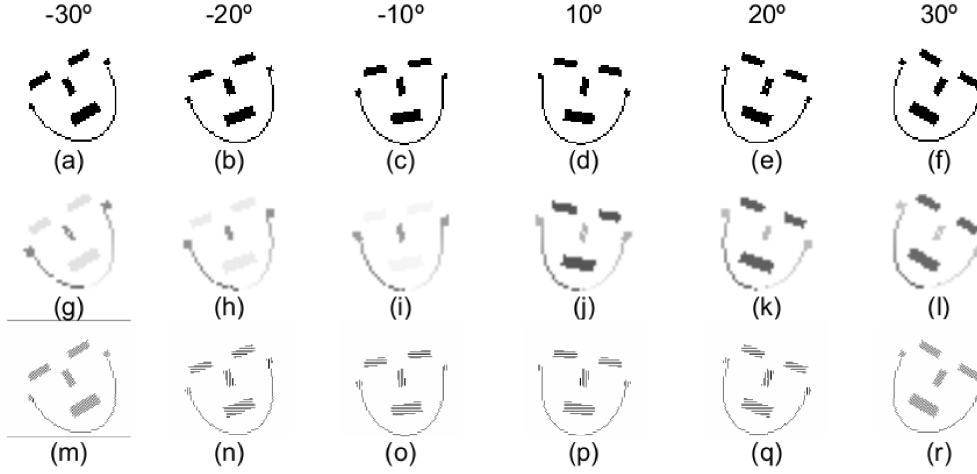


Figura 2.6: Moldes del rostro con rotación coronal para diferentes ángulos y un tamaño fijo

igual manera que los moldes direccionales. El cálculo de la integral de línea se realiza viene dado por:

$$I_{T_{R\varphi}}(i_0, j_0) = \frac{\sum_{i,j \in T_{R\varphi}(i_0, j_0)} [I_{D_{mod}}(i, j)\alpha(i, j) - (1 - I_{D_{mod}}(i, j)) \cdot 90^0]}{N_{T_{R\varphi}}}, \quad (2.3)$$

donde $T_{R\varphi}$ es el template utilizado, de tamaño R y ángulo de rotación φ , centrado en (i_0, j_0) . $N_{T_{R\varphi}}$ es el número de pixeles en el template $T_{R\varphi}$; mientras que $I_{D_{mod}}(i, j)$ es el valor del módulo en la imagen direccional de entrada.

La función $\alpha(i, j)$ se denomina “disparidad angular” y mide la diferencia angular entre el molde y la imagen direccional de entrada y viene dada por:

$$\alpha(i, j) = 90 - 2 \cdot \min(|T(i, j) - I_{D_{ang}}(i, j)|, 180 - |T(i, j) - I_{D_{ang}}(i, j)|), \quad (2.4)$$

donde $T(i, j)$ es el ángulo asociado al punto (i, j) en el molde $T_{R\varphi}$ (coordenadas relativas al centro del molde en (i_0, j_0)) y $I_{D_{ang}}(i, j)$ es el ángulo asociado al punto (i, j) en la imagen direccional I_D . La división por la cantidad de puntos del molde $N_{T_{R\varphi}}$ permite la normalización de los resultados con respecto al tamaño del molde. El factor $\alpha(i, j)$ provoca que, para una gran disparidad entre los ángulos, el valor de la integral de línea sea pequeño, disminuyendo la posibilidad de elección del molde asociado.

La aplicación sucesiva de los distintos moldes sobre la imagen, da la posibilidad de obtener

el tamaño de molde R , el ángulo de rotación φ y la posición (i, j) que maximiza la integral de línea y, por ende, el mejor calce entre el rostro y el molde.

La aplicación de esta metodología muestra muy buenos resultados en ambientes controlados de interfaz hombre-máquina, siendo una primera etapa en aplicaciones como la detección de iris entre otras [15, 24, 29, 30, 35].

2.2. Bases de datos utilizadas

2.2.1. Caltech

Creada por Markus Weber en el California Institute of Technology, USA [36]. Contiene fotografías de 27 individuos con rostros frontales sin rotación coronal. Existen variaciones en la pose e iluminación. Los fondos son variables y realistas. Esta base de datos incluye un total de 450 imágenes de rostros, de 896 x 592 píxeles. El formato es JPEG y se incluye un archivo Matlab con las coordenadas de la caja que contiene al rostro. Su uso es libre y gratuito.

2.2.2. The AR face database (Purdue)

Esta base de datos fue creada por Aleix Martinez y Robert Benavente en el Centro de Visión Computacional (CVC) de la Universidad de Purdue, USA [34]. Contiene sobre 4,000 imágenes a color de rostros correspondientes a 126 personas distintas (70 hombres y 56 mujeres). Las imágenes están almacenadas en formato RGB RAW (píxeles sin compresión) y poseen una resolución 768×576 píxeles, con 24 bits de profundidad de color. Las imágenes muestran rostros frontales con distintas expresiones faciales, condiciones de iluminación y oclusiones (lentes de sol y bufandas). Las fotografías fueron tomadas en el CVC bajo condiciones estrictamente controladas. El fondo es uniforme (blanco) y no se impusieron restricciones en la vestimenta (trajes, anteojos, etc.) ni en el maquillaje y tipo de peinado de los participantes. Cada individuo fue fotografiado en dos sesiones con dos semanas de diferencia. La base de datos es gratuita y pública para fines académicos (previa autorización por escrito) y está estrictamente prohibido su uso comercial.

2.2.3. FERET

El programa FERET (Facial Recognition Technology) [37] estableció como objetivo el crear una gran base de datos de imágenes faciales que sería utilizada con el fin de establecer un set de prueba independiente de los desarrolladores de algoritmos para detección y reconocimiento

de rostros. Las imágenes se obtuvieron en un ambiente semi-controlado, utilizando la misma configuración física para cada sesión. La base de datos FERET se obtuvo en 15 sesiones fotográficas llevadas a cabo entre Agosto de 1993 y Julio de 1996 e incluye 1564 conjuntos de imágenes (14.126 imágenes en total) adquiridas de 1199 individuos. La base de datos incluye 365 conjuntos de imágenes “duplicados” tomadas sobre individuos registrados previamente. Estos duplicados para algunos individuos, fueron adquiridos con días, incluso años de diferencia, permitiendo el estudio de la influencia del paso del tiempo en algoritmos de reconocimiento. La base de datos se encuentra disponible en forma gratuita, previa autorización.

2.2.4. YALE B

Esta base de datos [38] contiene 5760 imágenes de 10 individuos, cada uno fotografiado bajo 576 condiciones distintas (9 poses x 64 condiciones de iluminación). La iluminación consiste en una sola fuente lumínica (foco) dirigida hacia el rostro, la cual varía en 64 ángulos distintos. Producto de esto, las imágenes presentan una iluminación dura, con altos contrastes y sombras en la imagen. Para cada individuo también se adquiere una imagen sin iluminación alguna (sólo ambiental). Las imágenes están codificadas en escala de grises de 8 bits y almacenadas en formato PGM RAW (formato sin pérdidas). El tamaño de cada imagen es de 640×480 pixeles. Esta base de datos es libre de ser utilizada para fines académicos.

2.3. Criterios de medición

2.3.1. Criterio de detección de rostros

Las tasas de detección de rostros fueron calculadas para ambos métodos y comparadas con el método de Viola y Jones. Los métodos entregan como salida una caja rectangular centrada alrededor del rostro detectado y por tanto es importante medir la similitud entre dicha caja y la marcada en forma manual. Para uniformar el criterio de marcado manual, se utilizan consideraciones de tipo antropométrica [28], basadas en la posición de los ojos, por lo que basta con el marcar la posición de estos de manera precisa. Los parámetros que describen a una detección (caja) son la posición central del rostro (x, y) , el ancho de la caja (w) y el alto (h) . Para considerar un rostro como detectado correctamente, la caja de salida debe satisfacer dos criterios [39, 40]:

1. La distancia entre el centro de la caja real y el centro de la caja estimada debe ser menor

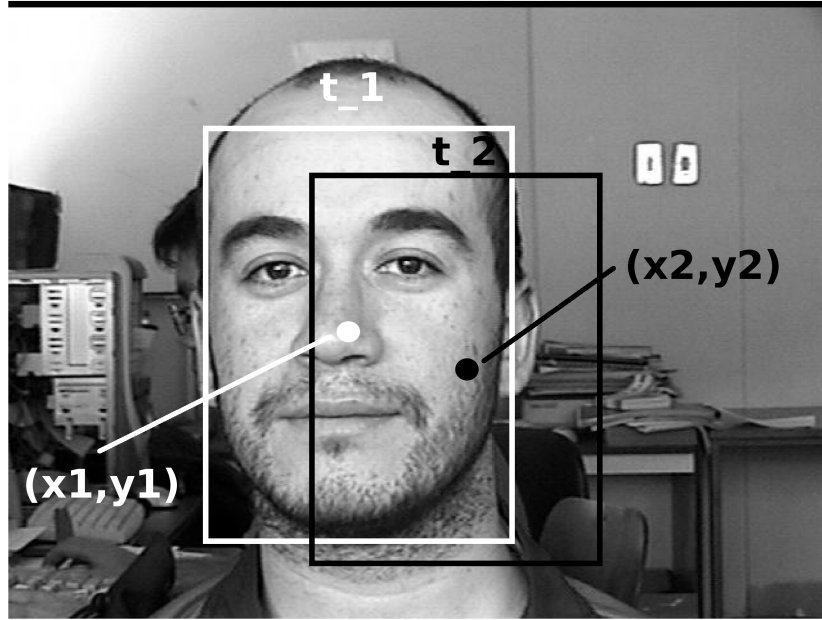


Figura 2.7: Ejemplo de la caja marcada en forma manual (blanca) y la entregada por el algoritmo (negra).

a P veces la suma de ambos anchos. Para este trabajo, P ha sido elegido igual a 0,25.

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} < P \cdot (w_1 + w_2) \quad (2.5)$$

2. El ancho de una de las cajas no puede ser más grande que el doble del ancho de la otra.

$$(w_1 < w_2 \wedge w_2 < 2 \cdot w_1) \vee (w_2 < w_1 \wedge w_1 < 2 \cdot w_2) \quad (2.6)$$

La figura 2.7 muestra un ejemplo de validación del rostro detectado. En este caso, el rostro estimado (caja negra) no cumple el criterio de distancia (2.5) entre los centros por encontrarse muy alejado de la segmentación manual (caja blanca).

2.3.2. Error en la estimación de parámetros

Para el caso de la metodología LET, se obtuvo estadísticas del error en la medición de los diversos parámetros del rostro, dados por la desviación entre el valor real y el valor estimado para cada parámetro. Las coordenadas reales de posición, así como los valores precisos de rotación coronal, excentricidad y tamaño del rostro, fueron determinados en forma manual para cada uno de los rostros presentes en las imágenes estáticas y secuencias de video utilizadas. El error absoluto entre el valor real de un parámetro (P_{Real}) y su valor estimado (P_{Est}) viene

dado por:

$$E_P = |P_{Est} - P_{Real}|. \quad (2.7)$$

Para independizar esta medida de error del tamaño del rostro y para incluir información relativa a los rangos utilizados para los parámetros no-espaciales, se crea la medida de error relativo. Para los parámetros espaciales, esto es, posición y tamaño del rostro, el error absoluto se normaliza por el tamaño real del rostro R_{Real} . Para el caso de la excentricidad y rotación coronal, el factor de normalización es la diferencia entre el máximo y el mínimo valor del rango permitida para ambos parámetros. El cálculo del error relativo se resume en:

$$\begin{aligned} E_{R_X} &= E_X / R_{Real} \\ E_{R_Y} &= E_Y / R_{Real} \\ E_{R_r} &= E_r / R_{Real} \\ E_{R_\varphi} &= E_\varphi / (\varphi_{max} - \varphi_{min}) \\ E_{R_\epsilon} &= E_\epsilon / (\epsilon_{max} - \epsilon_{min}). \end{aligned} \quad (2.8)$$

2.3.3. Criterio de localización de iris

Para medir el error en la localización de los ojos, se utiliza una métrica aplicada en [25], que define el error relativo en la determinación de la posición del iris como:

$$\text{err} = \frac{\max(d_l, d_r)}{d_{lr}}, \quad (2.9)$$

donde d_l es la disparidad del ojo izquierdo, i.e. la distancia entre la posición real del iris del ojo izquierdo y la estimada por el método; d_r la disparidad del ojo derecho y d_{lr} la distancia euclidiana entre las posiciones reales de los ojos izquierdo y derecho. Se considera una detección correcta si $\text{err} < 0,25$, lo que corresponde, aproximadamente, a que la posición del iris se encuentre en el borde del ojo, en vez del centro.

2.4. Librería OpenCV para procesamiento de imágenes

La librería OpenCV (Open Source Computer Vision) es una librería de código abierto, desarrollada por Intel. Está orientada al tratamiento de imágenes en tiempo real, lo que la hace especialmente apropiada para aplicaciones de Interfaces Hombre-Máquina, segmentación y reconocimiento de objetos, reconocimiento de rostros y gesticulaciones, seguimiento de movimiento y aplicaciones de robótica en tiempo real. Su licencia BSD permite que sea usada libremente para propósitos comerciales y de investigación con las condiciones en ella expresadas. La librería es multiplataforma, y puede ser usada en plataformas Mac OS X, Windows y Linux.

Algunas otras características generales de esta librería son:

- Es de código abierto.
- Programada en C/C++, con interfaces a otros lenguajes de programación (ej. Python)
- Altamente optimizada. Provee una interfaz a las primitivas IPP de Intel (Intel's Integrated Performance Primitives) que incluyen funciones altamente optimizadas con código específico para procesadores de tipo Intel.
- Incluye funciones genéricas para la lectura/escritura/adquisición de imágenes y video.

En cuanto a funcionalidad, cabe destacar lo siguiente:

- Manipulación de los datos internos de la imagen.
- Entrada/Salida de imagen y video.
- Manipulación matricial y vectorial con rutinas de álgebra lineal (productos, eigenvalues, descomposición en valores singulares).
- Estructuras de datos dinámicas (listas, colas, conjuntos, árboles, grafos).
- Procesamiento básico de imágenes (filtrado, detección de bordes, muestreo e interpolación, conversión de espacio de colores, operaciones morfológicas, histogramas, etc.)
- Análisis estructural (procesamiento de contornos, transformación de distancia, template matching, transformada de Hough, aproximación poligonal, ajuste de líneas y elipses, etc.)
- Calibración de cámaras (identificación y seguimiento de patrones de calibración, estimación de la matriz fundamental, correspondencia estéreo, etc.)

- Análisis de movimiento (flujo óptico, segmentación de movimiento, tracking).
- Reconocimiento de objetos (eigen-methods, HMM).
- Componentes básicos de interfaz gráfica (despliegue de imagen/video, manejo de mouse y teclado, barras de desplazamiento).
- Rutinas básicas de dibujo (lineas, curvas, polígonos, texto).

Capítulo 3

La Transformada Elíptica Lineal (LET)

3.1. La Transformada Elíptica Lineal

La Transformada Elíptica Lineal (LET) está basada en la Transformada Log-Polar (LPT) [41]. Debido a la naturaleza esférica del ojo, la LPT representa a la imagen de entrada en coordenadas polares, con una escala logarítmica para el eje axial. La LET sigue los mismos principios de la LPT, pero está definida como un mapeo de coordenadas cartesianas a elípticas y utiliza una escala lineal para toda la imagen. La transformada LET está definida por (3.1),

$$\begin{aligned} r &= \sqrt{(\hat{x}/\epsilon)^2 + \hat{y}^2} \\ \theta &= \text{atan}\left(\frac{\hat{y}}{\hat{x}}\right) \end{aligned} \tag{3.1}$$

donde:

$$\begin{aligned} \hat{x} &= (x - x_0) \cdot \cos \varphi + (y - y_0) \cdot \text{sen} \varphi \\ \hat{y} &= -(x - x_0) \cdot \text{sen} \varphi + (y - y_0) \cdot \cos \varphi \\ \epsilon &= \text{Excentricidad.} \\ \varphi &= \text{Parámetro rotación angular de la LET.} \\ (x_0, y_0) &= \text{Párametros de posición de la LET en la imagen de entrada (centro de la elipse)} \end{aligned}$$

La transformada LET inversa, utilizada en la implementación computacional del método está dada por (3.2)

$$\begin{aligned} x &= \frac{r \cdot \epsilon \cdot \cos(\theta + \varphi)}{\sqrt{\cos^2(\theta) + \epsilon^2 \sin^2(\theta)}} + x_0 \\ y &= \frac{r \cdot \epsilon \cdot \sin(\theta + \varphi)}{\sqrt{\cos^2(\theta) + \epsilon^2 \sin^2(\theta)}} + y_0. \end{aligned} \quad (3.2)$$

Como se muestra en la ecuación (3.1), la LET posee cuatro parámetros a ajustar. Dos parámetros para la posición del centro de la elipse (x_0, y_0) , uno para la rotación angular (φ) y uno para la excentricidad de la elipse (ϵ) . Estos cuatro parámetros describen una familia de elipses de distintos tamaños, localizadas sobre un plano cartesiano, donde las elipses están centradas en (x_0, y_0) , rotadas en φ radianes y con una relación de aspecto igual a ϵ entre los semiejes secundario y principal de la elipse. Ajustando estos cuatro parámetros, el objetivo de encontrar un contorno semi elíptico en la imagen de entrada puede ser visto como un problema de optimización en cuatro dimensiones, en el que el conjunto preciso de parámetros debe ser encontrado. Si se utiliza el conjunto correcto de parámetros para calcular la LET, aparecerá un borde vertical en la imagen LET de salida. Si se utiliza un conjunto incorrecto de parámetros, la imagen LET no presentará dicho borde vertical. Este efecto se deduce de la ecuación 3.1, ya que al fijar el valor de r a una constante, la ecuación resultante es la de una elipse de excentricidad ϵ , ángulo de rotación φ y centro en (x_0, y_0) . La Fig. 3.1 (a) muestra un ejemplo simple en el que una elipse en la imagen de entrada (espacio X-Y cartesiano) es convertida en una línea vertical en la imagen LET $(r-\theta)$ de salida cuando el conjunto de parámetros es el adecuado (3.1(b)). La figura 3.1 (c) muestra el caso en que se utilizan valores incorrectos (distintos de la elipse original) en uno o más parámetros para calcular la LET.

La posición del borde vertical sobre la imagen LET se calcula mediante un acumulador que almacena los valores del histograma proyectado por columnas de la imagen. Para normalizar estos valores y hacerlos independientes del tamaño de la imagen LET, se establece un valor de medida de la verticalidad del borde sobre la imagen LET que denominaremos V_{LET} . La medida V_{LET} (3.3) es definida como la suma de todos los pixeles en una columna, normalizada por la altura de la imagen:

$$V_{\text{LET}}(r) = \frac{1}{N} \sum_{\theta=0}^N I_{\text{LET}}(r, \theta), \quad (3.3)$$

donde:

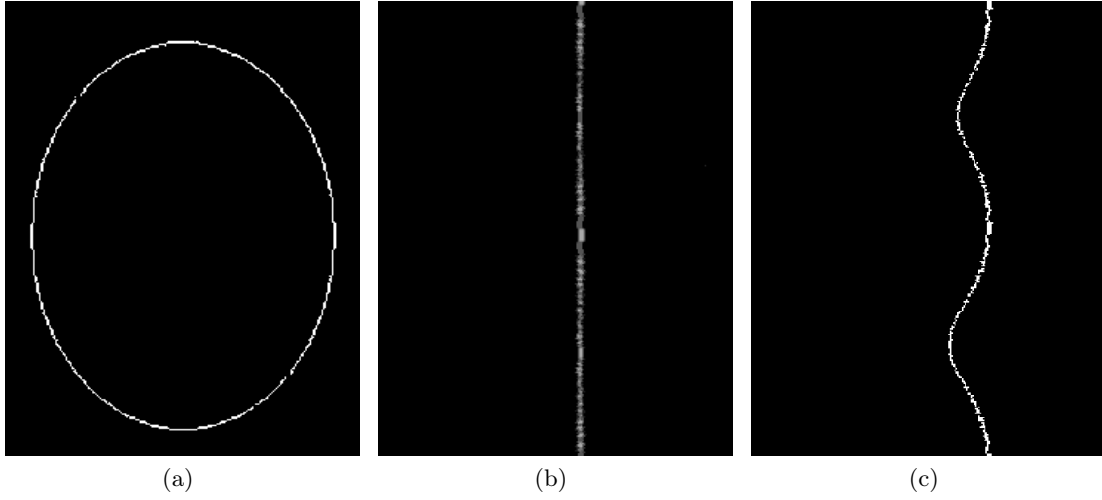


Figura 3.1: Efecto de aplicar la transformada LET a una elipse. (a) Imagen Original en coordenadas cartesianas X-Y. (b) Imagen LET con el conjunto correcto de parámetros. Se observa como la elipse es transformada en un borde vertical si los parámetros de la LET son iguales a los de la elipse original. (c) Imagen LET con un conjunto incorrecto de parámetros. Para las figuras (b) y (c) el eje radial r corresponde al eje horizontal y el eje angular θ al eje vertical.

$$\begin{aligned}
 N &= \frac{h}{2} \\
 h &= \text{Altura de la imagen LET en pixeles} \\
 r &= \text{Eje radial (una columna en la imagen LET)} \\
 \theta &= \text{Eje angular (una fila en la imagen LET)}.
 \end{aligned}$$

Debido a que el objetivo es la detección del contorno semi-elíptico inferior del rostro, sólo los ángulos entre 0 y π son considerados para θ en la imagen LET, es decir, sólo se incluye en la suma, la mitad de las columnas de la imagen. Se considera sólo el borde semi-elíptico inferior del rostro, debido a que presenta mayor estabilidad para la detección, al contrario del borde superior, que se ve modificado en gran forma por la forma de la cabeza, pelo, gorro, etc. Adicionalmente, sólo las columnas entre $[r_{\min}, r_{\max}]$ son calculadas ya que estos valores definen los límites de tamaño mínimo y máximo del rostro que el método puede detectar. Los valores de V_{LET} abarcan desde 0 a 255 para imágenes en escala de grises de 8 bits, donde 255 corresponde a una columna con todos sus pixeles en blanco.

Cuando un borde vertical se encuentra presente en la imagen, $V_{LET}(r)$ será máximo para la columna (r) en que se encuentre dicho borde. El tamaño en pixeles de la elipse (semieje mayor) estará entonces determinado por la posición en el eje radial (r) en que se presentó el

máximo valor.

3.2. Metodología

La utilización de la LET como método para estimar los parámetros del rostro puede ser visto como un problema de optimización de cuatro dimensiones. Debido a esto, el objetivo del método consiste en encontrar el mejor conjunto de parámetros $(x_0, y_0, \varphi, \epsilon)$ de manera que maximicen V_{LET} .

La metodología utilizada se divide en dos etapas principales: una de detección gruesa, seguida de una etapa de localización fina del rostro. Esta forma de abordar el problema ha sido utilizada con éxito en trabajos anteriores [15, 32]. El objetivo de la etapa de detección gruesa es obtener una aproximación inicial de la posición del rostro para su posterior localización. Esta detección gruesa, en conjunto con una estimación antropométrica de la excentricidad promedio del rostro, nos da un punto de partida para la búsqueda de los parámetros óptimos del rostro mediante LET. El objetivo de la etapa de detección fina es refinar la posición del centro del rostro y estimar su rotación coronal y excentricidad en forma precisa.

Un diagrama de bloques explicativo del método se muestra en la Fig. 3.2. En primer lugar, la imagen de entrada es preprocesada con el objeto de reducir los requerimientos computacionales, eliminando información innecesaria. Posteriormente se aplica la etapa de detección gruesa para obtener una estimación de la posición del rostro. La localización fina y la estimación de los parámetros faltantes del rostro se lleva a cabo en la etapa siguiente, calculando varias transformadas LET sucesivas sobre la imagen preprocesada. Cada una de estas transformadas LET se aplican con distintos valores en sus parámetros. Después de cada cálculo de la imagen LET, se obtiene una imagen direccional de esta última [15, 32, 35], detectando el borde vertical que corresponde al contorno semi elíptico inferior del rostro. Para esto se calcula un histograma proyectado a lo largo de las columnas de la imagen direccional, obteniendo un V_{LET} máximo para cada imagen. Los parámetros de la LET que maximicen el valor V_{LET} de la imagen direccional asociada, son consideradas como los parámetros correctos del proceso de estimación de las características de la elipse. Por lo tanto, los parámetros del rostro: posición, rotación, excentricidad y tamaño son determinados especificando la elipse que mejor se ajusta al rostro de entrada. La etapa de localización fina, puede también ser utilizada para estimar sólo un subconjunto de los parámetros si la información acerca de uno o varios de ellos se encuentra disponible, como por ejemplo en rostros frontales sin rotación coronal o con un valor conocido, excentricidad de rostro predeterminada, etc.

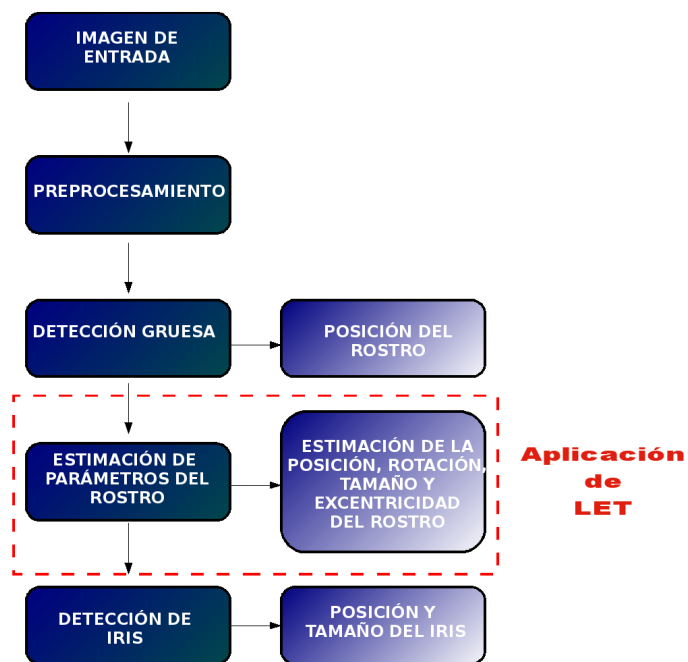


Figura 3.2: Diagrama de bloques en que muestra la aplicación de la transformada LET en la etapa de detección fina. Esto permite una estimación precisa de los parámetros del rostro.

3.2.1. Pre-procesamiento de la imagen

Primero la imagen de entrada se reescala a 320 pixeles en su dimensión principal (dependiendo si la imagen tiene orientación vertical u horizontal), manteniendo la relación de aspecto entre ancho y altura intacta. Esto tiene por objeto el disminuir los requerimientos de memoria del sistema y los tiempos de procesamiento. Debido a que el método utiliza sólo información geométrica, la imagen redimensionada puede ser transformada desde el espacio de color RGB de 24 bits a un espacio de escala de grises de 8 bits, mediante la ecuación:

$$I_{\text{gris}}(i, j) = 0,2980 \cdot I_{\text{R}}(i, j) + 0,5922 \cdot I_{\text{G}}(i, j) + 0,1098 \cdot I_{\text{B}}(i, j). \quad (3.4)$$

Finalmente, se aplica un filtro promediador de 3x3 de manera de eliminar los bordes débiles de la imagen y disminuir el ruido.

3.2.2. Etapa de detección gruesa

Esta etapa tiene como meta principal el proveer de un punto inicial para la búsqueda de los parámetros óptimos del rostro que se llevará a cabo en la etapa siguiente. Para esta etapa se utiliza un enfoque basado en *templates*, propuesto originalmente por Maio y Maltoni [32], que asume que los rostros frontales pueden ser modelados por elipses. Debido a que la aplicación principal del método es en el ámbito de las interfaces hombre-máquina, se asume que un rostro frontal con posible rotación coronal es la principal forma elíptica presente en la imagen. Este método, basado en la Transformada elíptica de Hough, aplica una serie de moldes (o *templates*) elípticos predefinidos sobre una imagen direccional obtenida a partir de la imagen de entrada pre-procesada. La imagen direccional contiene información acerca de los vectores tangentes a todos los bordes presentes. Moldes direccionales predefinidos se encuentran asociados a los ángulos de dichos vectores y son aplicados sobre cada uno de los vectores de la imagen direccional. Estos moldes entregan información a una matriz acumulador acerca de todos los puntos en la imagen que puedan haber generado un vector tangente con la posición y ángulo adecuados. Después de aplicar los correspondientes moldes a todos los vectores en la imagen, las coordenadas del acumulador que presenten el mayor puntaje corresponderán al posible centro del rostro. La Fig. 3.3.a muestra una imagen original pre-procesada, la fig. 3.3.b la correspondiente imagen I_{LDR} , la fig. 3.3.c el acumulador superpuesto a I_{LDR} , y la fig. 3.3.d al acumulador sobre la imagen original. Este método para detección gruesa fue empleado debido a su simplicidad y a su probada efectividad en investigaciones previas [15, 30, 35, 42, 43]. En el Anexo de esta Tesis, se explica de forma detallada el cálculo de la imagen direccional y

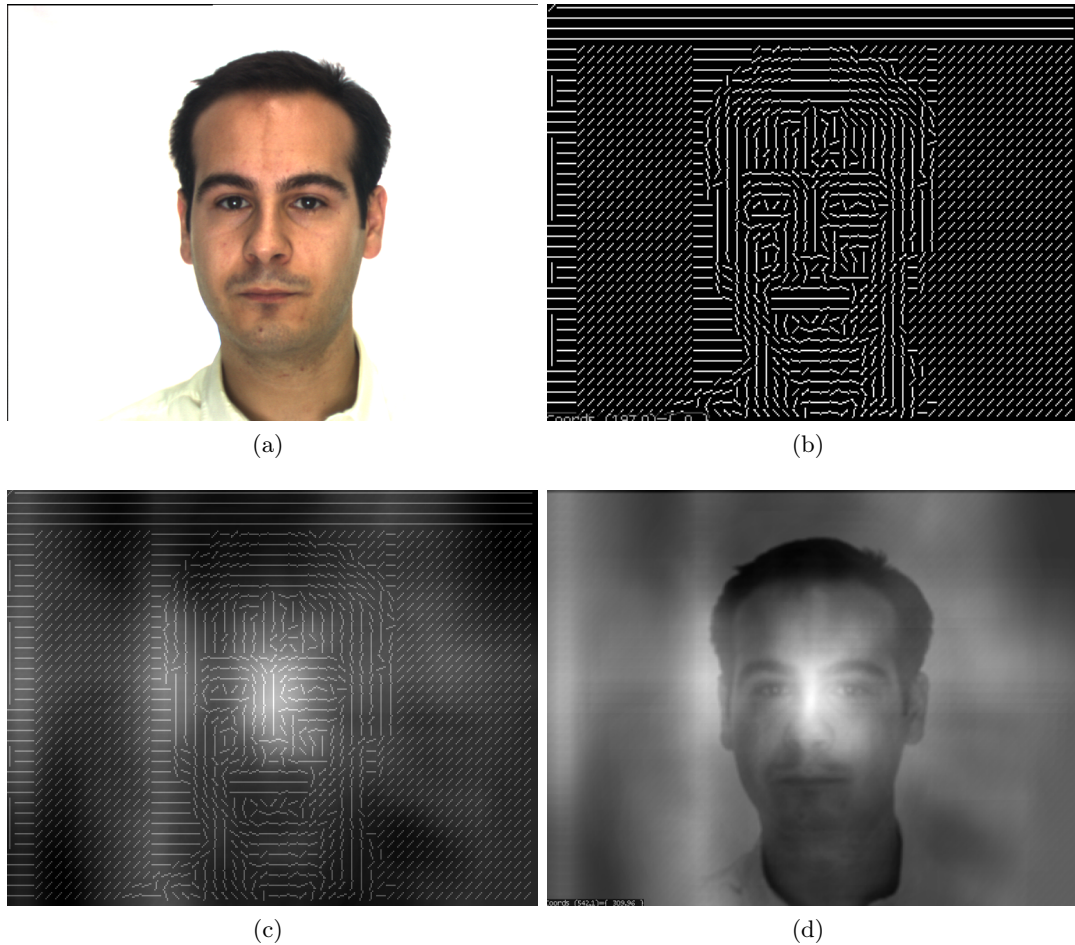


Figura 3.3: Detección gruesa mediante el método de Maio y Maltoni. (a) Imagen Original. (b) Imagen direccional I_{LDR} . (c) Acumulador e imagen direccional superpuestas. (d) Acumulador e imagen original. El máximo del acumulador (zona brillante) está localizada sobre el centro del rostro.

la etapa de detección gruesa.

3.2.3. Estimación de los parámetros del rostro

La posición, excentricidad, rotación y tamaño del rostro pueden ser estimados usando la Transformada LET para seleccionar la mejor elipse que se ajuste al rostro. La fig. 3.4 muestra el resultado de aplicar la transformada (con el conjunto correcto de parámetros) a una imagen de la base de datos de PURDUE [34]. Se observa como el contorno inferior del rostro se convierte en un borde vertical en la imagen LET. Los pixeles negros se deben a la compresión de la imagen producto de la transformación de coordenadas, por lo que no todos los pixeles

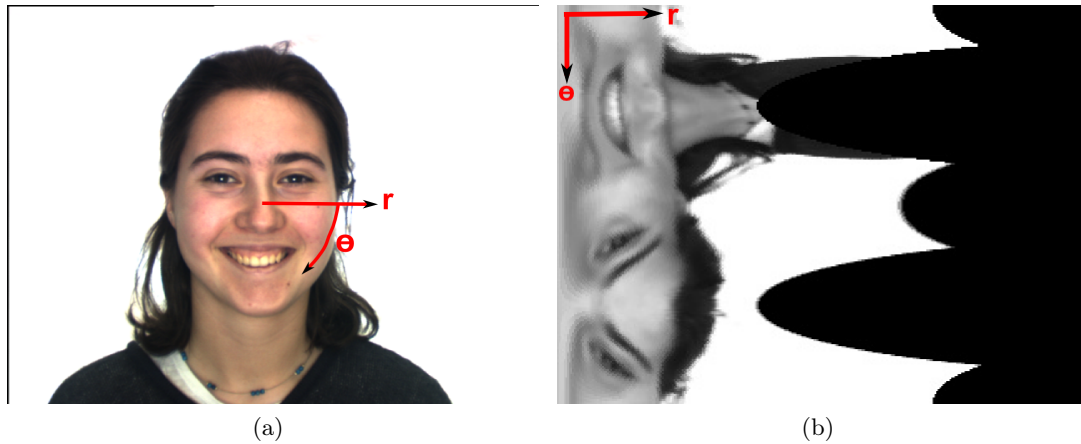


Figura 3.4: Transformada LET aplica a un rostro de la base de datos de PURDUE. (a) Imagen Original. (b) Imagen LET. Se aprecia como el contorno inferior del rostro se mapea a un borde vertical en la mitad superior de la imagen.

de la imagen LET (de ejes r y θ) tienen un correspondiente pixel válido en la imagen original (de coordenadas X-Y).

La metodología LET de estimación de parámetros del rostro requiere la aplicación de un detector de bordes sobre la imagen LET que permita identificar y aislar el borde vertical si es que se encuentra presente. Para este efecto, se utiliza una imagen direccional, similar a la empleada en la etapa de detección gruesa, pero de mayor resolución. Esta imagen direccional de alta resolución, que se denominará (I_{DirLET}), es calculada sobre vecindades de 3x3 píxeles y entrega información angular acerca de los bordes presentes en la imagen LET. Debido a que en este caso, el objetivo es detectar los bordes verticales, sólo los vectores con ángulos cercanos a 90^0 son considerados. Los bordes verticales se presentan como líneas blancas verticales en la imagen I_{DirLET} . Por último, y con el propósito de eliminar ruido para el posterior cálculo de V_{LET} , los vectores con módulo pequeño también son eliminados de la imagen final, manteniendo sólo los bordes más relevantes. La Fig. 3.5.a muestra una imagen de entrada preprocesada, la fig 3.5.b la imagen LET correspondiente, calculada con los valores de los parámetros precisos de manera de entregar un borde vertical; la fig 3.5.c muestra la imagen direccional asociada y la fig 3.5.d histograma proyectado (V_{LET} por columnas) asociado a la imagen direccional. Se puede observar que la posición del histograma que presenta un V_{LET} máximo coincide con la columna en que se presenta el borde vertical en la imagen LET.

Con esta información previa, la estimación de la elipse que mejor se ajusta al rostro es realizada calculando transformadas LET sucesivas sobre la imagen original preprocesada, variando los parámetros de posición, excentricidad y rotación. Cada vez que se aplica la LET, se obtiene el máximo valor V_{LET} de la correspondiente imagen I_{DirLET} , siendo la transformada que

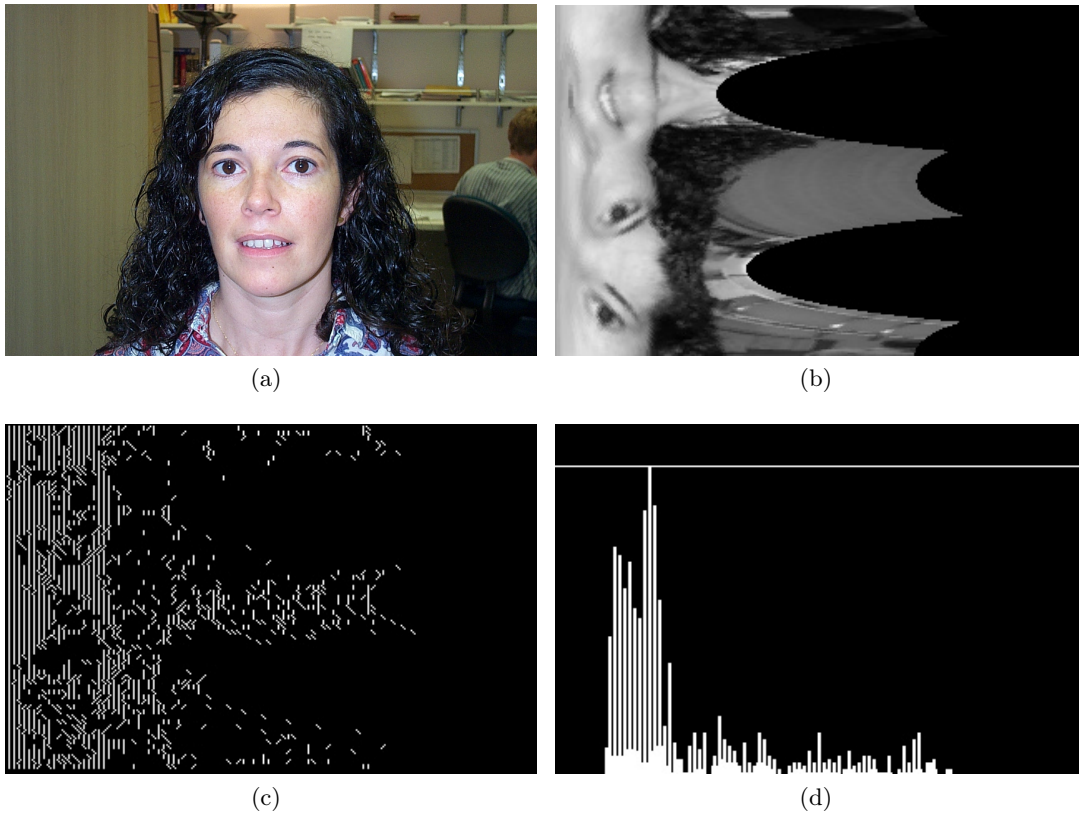


Figura 3.5: Detección del borde vertical mediante imágenes direccionales. (a) Imagen Original. (b) Imagen LET. (c) Imagen direccional. (d) Histograma proyectado. Se puede apreciar un máximo en la columna donde se encuentra ubicado el borde vertical.

presente el mayor valor la que determina el valor del conjunto de parámetros $(x_0, y_0, \epsilon, \varphi)$ que define a la elipse objetivo. En teoría, sólo un conjunto de parámetros tendrá como resultado un borde perfectamente vertical en la imagen LET, siempre y cuando se asuma que el borde inferior del rostro queda representado por una única elipse con tamaño, rotación y posición definidos. En general, los cuatro parámetros de la LET son inicialmente desconocidos, pero la etapa de detección gruesa nos da una primera aproximación de las coordenadas del rostro, tanto vertical (x_0) como horizontal (y_0). La excentricidad estadísticamente presenta poca variación entre personas y por lo tanto, se puede utilizar un valor promedio como punto de partida. La rotación es por tanto, el parámetro más difícil a estimar. Si se asume una posición inicial no rotada del rostro, o si la rotación observada es muy pequeña para los individuos en cuestión (como en el caso de las bases de datos utilizadas), se puede asumir un valor de 0^0 para este parámetro. En el caso en que exista rotación coronal, el rango de búsqueda va desde -30^0 a $+30^0$. El rango de búsqueda para la posición del rostro es de $[x_c - 20, x_c + 20]$ para la posición horizontal y de $[y_c - 40, y_c + 40]$ para la posición vertical, a partir de las coordenadas entregadas en la etapa de detección gruesa (x_c, y_c) . Estos valores son los utilizados durante el desarrollo de esta tesis, pero pueden ser modificados según la aplicación. La asimetría en el rango de búsqueda entre ambas direcciones se debe al hecho de que la etapa de detección gruesa es mucho menos precisa en estimar la coordenada vertical del rostro. El parámetro de excentricidad se mueve en el rango $[0,65, 0,95]$ para rostros normales sin gesticulaciones exageradas. Cuando se encuentran los cuatro parámetros, el tamaño del rostro puede ser calculado a partir de la columna con más alto V_{LET} en el histograma proyectado. El índice de dicha columna (posición en el eje r) determina el tamaño en píxeles del semieje principal de la elipse (R).

3.3. Procesamiento de Secuencias de Video

La detección de rostros en secuencias de video sigue la misma lógica explicada hasta ahora. Sin embargo, como es el mismo sujeto el que se está detectando, es posible utilizar un modelo para la predicción de las variables del proceso como son la posición, ángulo de rotación y tamaño, con el fin de disminuir los rangos de búsqueda para el siguiente cuadro. Esto permite por un lado, mayor robustez una vez que se ha detectado correctamente la vecindad de los parámetros y mayor velocidad de procesamiento de manera de obtener una tasa de procesamiento que sea lo más cercana a tiempo real.

El primer cuadro se procesa de manera distinta al resto, siendo utilizada la detección gruesa para obtener valores aproximados de la posición del rostro. La excentricidad se estima mediante LET en el rango $[0,65, 0,95]$. Se considera una rotación inicial nula del rostro en

secuencias de video.

Los siguientes cuadros de la secuencia se procesan según el algoritmo de la Figura 3.6 utilizando la estimación de parámetros de LET. La estimación de parámetros se realiza en una vecindad de 20×20 pixeles para la posición y $[\varphi - 10^0, \varphi + 10^0]$ para la rotación, siendo φ el valor estimado para el ángulo a partir de la información de los cuadros previos. La excentricidad se mantiene constante en el valor detectado en el primer cuadro, ya que es un parámetro que no varía mucho para un mismo individuo. Una vez obtenida la posición fina y la rotación ajustada, se calcula el tamaño del rostro mediante LET.

La estimación de los parámetros de posición para el siguiente cuadro se realizan de la siguiente forma:

$$\begin{aligned}\hat{x}_t &= 2 \cdot x_{t-1} - x_{t-2} \\ \hat{y}_t &= 2 \cdot y_{t-1} - y_{t-2},\end{aligned}\tag{3.5}$$

donde (\hat{x}_t, \hat{y}_t) corresponde a la posición en el cuadro actual, (x_{t-1}, y_{t-1}) a la posición en el cuadro anterior y (x_{t-2}, y_{t-2}) a la posición dos cuadros atrás en la secuencia de video.

El ángulo de rotación para el siguiente cuadro se calcula utilizando una ecuación similar a la anterior:

$$\varphi_t = 2 \cdot \varphi_{t-1} - \varphi_{t-2},\tag{3.6}$$

donde φ_t es el ángulo de rotación para el cuadro actual, φ_{t-1} el ángulo de rotación del cuadro anterior y φ_{t-2} el ángulo de rotación coronal dos cuadros atrás.

El tamaño del rostro (semieje de la elipse) se mueve entre $[r - 4, r + 4]$ según el tamaño detectado en el cuadro anterior.

Se considera una detección fallida del rostro si el valor LET asociado es menor a 100. Si falla la detección en uno de los cuadros del proceso, se aplica detección gruesa nuevamente y se continúa con el proceso explicado anteriormente.

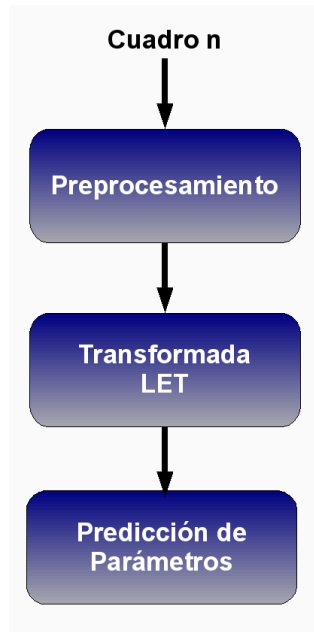


Figura 3.6: Algoritmo de Detección para Secuencias de Video

3.4. Pruebas Realizadas

El método propuesto fue probado sobre 300 imágenes de la base de datos de FERET [37] (subconjuntos fa y fb), 275 de la base de datos de PURDUE [34] (subconjuntos 1 al 7 en PURDUE), 425 de la de CALTECH [36] y 1200 de la base de datos de YALE-B [38]. Estas imágenes fueron seleccionadas aleatoriamente, eliminando aquellas imágenes que presentaban rostros no frontales, oclusiones o lentes de sol. Las figuras 3.7(a, c, e y g) muestran un ejemplo de un rostro frontal de cada una de las cuatro bases de datos. Debido a que los rostros en estas bases de datos casi no presentan rotación coronal, se utilizaron además cuatro secuencias de video para probar la selectividad angular de la LET. Cada secuencia de video presenta un individuo frontal a la cámara y rotando su cabeza en el plano coronal. Estas secuencias fueron capturadas utilizando una cámara de video NTSC con una resolución de 640×480 pixeles por cuadro. A modo de ejemplo las figuras 3.8(a, c, e y g) muestran un cuadro de cada una de las secuencias de video. Para cada imagen de las cuatro bases de datos descritas y para cada cuadro de las secuencias de video, se estima la posición, excentricidad, rotación y tamaño del rostro utilizando la metodología LET. La metodología LET fue comparada en cada una de las cuatro base de datos con el detector de Viola y Jones [20, 27] que denominaremos $Adaboost_{VJ}$. El tiempo de procesamiento también fue medido, para obtener información acerca de la aplicación en tiempo real del método. Todas las pruebas fueron realizadas sobre un computador con procesador Athlon64 3000+ con un 1GB de memoria RAM.

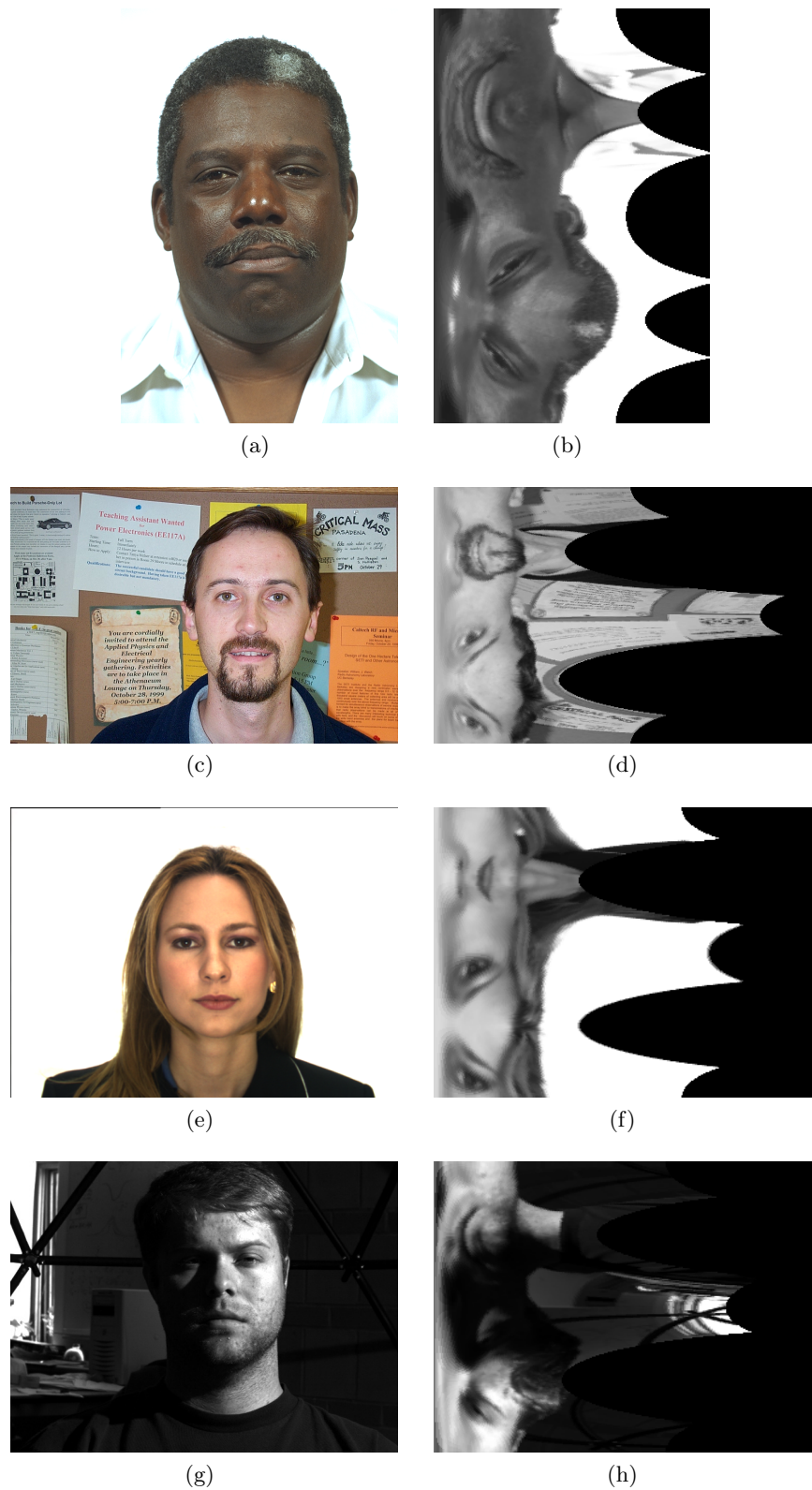


Figura 3.7: Ejemplos de imágenes de las bases de datos de FERET (a), CALTECH (c), PURDUE (e) y YALE B (g). En las figuras (b), (d), (f) y (h) se muestra la transformada LET correspondiente a (a), (c), (e) y (g).

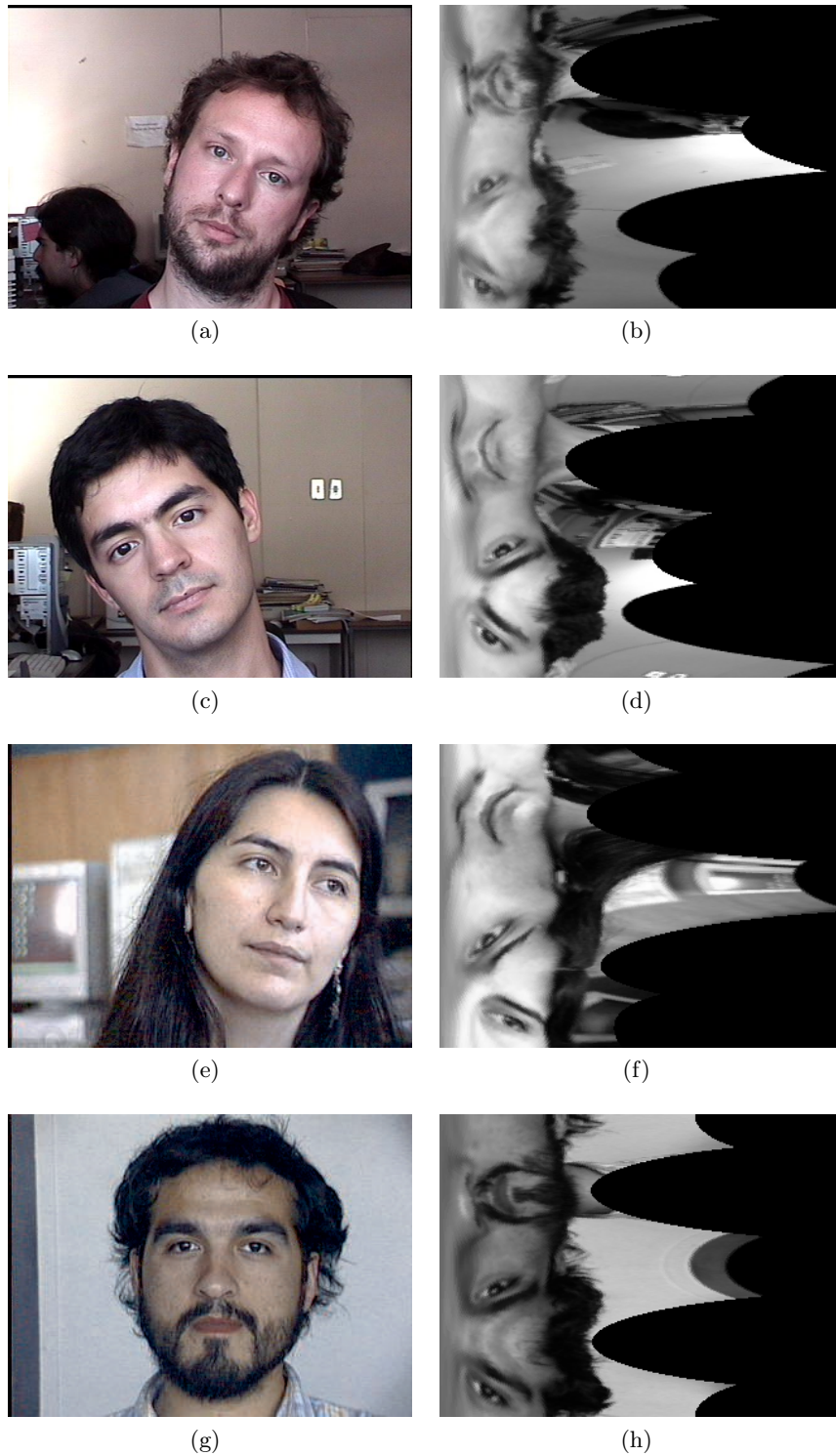


Figura 3.8: Ejemplos de cuadros extraídos de las cuatro secuencias de video (a), (c), (e) y (g). Sus respectivas Transformadas LET se muestran en (b), (d), (f) y (h).

Capítulo 4

Optimización de Moldes Direccionales mediante PSO

El método original de detección de rostros en base a moldes direccionales [32] fue mejorado de manera de poder detectar rostros con rotación coronal en tiempo real [15]. Este método probó ser adecuado para el uso en interfaces hombre-máquina.

La detección mediante moldes direccionales se basa en el cálculo de la imagen direccional del rostro. Esta imagen direccional contiene información angular de los gradientes principales de la cara de un individuo en particular. Un molde ideal debería entregar un valor de detección alto cuando se usa para calcular la integral de línea justo sobre la imagen direccional del rostro de un individuo dado. Asimismo, debería producir un valor de detección lo más bajo posible en cualquier otro lugar de la imagen en que no se encuentre el rostro objetivo. Como la imagen direccional cambia con los movimientos faciales y condiciones de iluminación, el molde debería representar características principales del rostro que se mantengan invariantes frente a estos cambios. Los moldes direccionales, utilizados en trabajos previos, fueron creados con esta idea en mente, incluyendo información acerca de la posición de los ojos, cejas, nariz, boca y la parte inferior del mentón. Se crean moldes de distintos tamaños y rotaciones para ser aplicados en tiempo real en la localización de rostros. Debido al uso de características antropométricas en estos moldes, se denominan Moldes Antropométricos. A pesar del buen desempeño mostrado en la utilización de dichos moldes, es natural buscar una forma óptima de estimar estos moldes de manera que se incluyan tanto las características que presenten un mayor desempeño en la localización del rostro; como optimice el desempeño del algoritmo de detección. Adicionalmente, las características relevantes para cada individuo pueden diferir, pudiendo obtenerse una mejor respuesta del sistema si los moldes se adaptan para cada rostro en particular.

La búsqueda de moldes mejorados para la localización de rostros, involucra una optimización en un espacio de alta dimensión, haciendo imposible aplicar una búsqueda exhaustiva del molde óptimo. Incluso para el caso del molde más pequeño propuesto en este método y considerando sólo un número fijo de puntos en el molde, el número de cálculos involucrados es inmenso. Si se considera que el molde más pequeño a optimizar es de 30×40 píxeles y que cada píxel puede tomar 36 valores distintos (ya que la información angular se cuantiza en ángulos de 5^0), al tomar un número fijo de 200 píxeles como parte del molde, el número de moldes posibles de generar es enorme y viene dado por:

$$\text{Número de moldes} = \binom{1200}{200} \times 36^{200} = \frac{1200!}{200! \times 1000!} \times 36^{200} \quad (4.1)$$

El hecho de que el número de puntos presente por molde es variable, y que se optimizan varios moldes de tamaño superior a 30×40 píxeles, hace que el espacio de búsqueda sea mucho mayor al expuesto en 4.1, lo que justifica la elección de un algoritmo de optimización que guíe la búsqueda del óptimo. La metodología propuesta en esta tesis utiliza el algoritmo de Optimización por Enjambre de Partículas (PSO) debido a su probada efectividad y relativa simpleza [44]. El algoritmo PSO fue presentado originalmente por Eberhart y Kennedy [44]. Diversos trabajos han expandido el algoritmo original y analizado su convergencia y estabilidad [45–51].

4.1. Generación de moldes PSO

El algoritmo PSO es una heurística combinatorial relativamente nueva, basada en la interacción social de sistemas biológicos tales como cardúmenes de peces o bandadas de aves. Desde su creación, varias mejoras han sido propuestas, introduciendo nuevos parámetros para intensificar la búsqueda, así como para evitar la explosión combinatorial de las soluciones [45, 48].

El algoritmo PSO simula el comportamiento social de un individuo (partícula) moviéndose en un espacio multidimensional. Cada vector solución se identifica como una partícula individual con coordenadas específicas en el espacio de búsqueda. Cada partícula posee una posición (X_i) y una velocidad (V_i) determinadas en el espacio. La posición representa una posible solución al problema de optimización, mientras que la velocidad indica la cantidad de cambio entre la posición actual y la anterior. El algoritmo almacena la mejor posición anterior (P_i) de cada partícula y la mejor posición global de entre todas las partículas que están en su vecindad ($P_{g(i)}$). Esta información es utilizada para adaptar la velocidad de cada partícula y, por ende, la posición para la iteración siguiente. La idea general es que cada partícula recorrerá el espacio de soluciones, siendo “atraída” a los puntos más prometedores que haya recorrido

anteriormente y a los puntos que globalmente hayan resultado con mejor *fitness* para el resto de las partículas.

Al inicio, el algoritmo genera una población de partículas en que cada posición y velocidad son inicializadas aleatoriamente. La velocidad de cada partícula se inicializa en un rango dado por los parámetros $[-V_{max}, V_{max}]$ donde (V_{max}) es la velocidad máxima que pueden tomar las partículas. Posterior a la inicialización de las partículas, y para las iteraciones subsiguientes, una función de fitness (función a optimizar) se evalúa para cada posición de las partículas. Cada vez que el resultado del fitness actual sea mejor que el mejor fitness previo almacenado (P_i), la posición actual pasará a ser la mejor posición histórica de la partícula ($P_i = X_i$). Lo mismo ocurre con el cálculo de la mejor posición en la vecindad de la partícula ($P_{g(i)}$), la que será actualizada cuando se encuentre un mejor valor de fitness en la posición actual de alguna de las partículas de la vecindad. Para las iteraciones siguientes, la velocidad y la posición de cada partícula se ajustarán de la forma dada por (4.2) y (4.3):

$$V_i(t + 1) = w \cdot V_i(t) + \varphi_1 \cdot (P_i - X_i(t)) + \varphi_2 \cdot (P_{g(i)} - X_i(t)) \quad (4.2)$$

$$X_i(t + 1) = X_i(t) + V_i(t + 1) , \quad (4.3)$$

donde φ_1 y φ_2 son números aleatorios en el rango $[0, \dots, \varphi_{max}]$. Generalmente φ_{max} , es igual a 2, pero puede cambiar dependiendo de la implementación utilizada, siempre y cuando se cuide la estabilidad. En las ecuaciones previas, w es el parámetro de inercia utilizado para balancear la búsqueda entre exploración y convergencia a lo largo de las iteraciones [48].

El método PSO se aplica a los moldes direccionales de rostros frontales de manera de generar un conjunto de moldes para distintos tamaños de rostro. Debido a lo complejo de los cálculos y al tiempo computacional empleado en la optimización, sólo se optimizarán moldes de rotación cero (0^0). La figura 4.1 muestra un diagrama de flujo del método utilizado para la generación de moldes PSO y el algoritmo completo se explica en las secciones siguientes.

4.1.1. Preprocesamiento

El método de optimización requiere de varias imágenes de rostros frontales a optimizar. Por lo tanto, la primera etapa es segmentar las imágenes de rostros, determinando los parámetros (x_1, x_2, y_1, y_2, T) mostrados en la fig. 4.2. El rostro se enmarca entre los intervalos $[y_1 - \Delta y_1, y_2 + \Delta y_2]$ y $[x_1, x_2]$ para las coordenadas vertical y horizontal respectivamente. Esta segmentación puede ser realizada en forma manual o utilizando la posición de los ojos como referencia

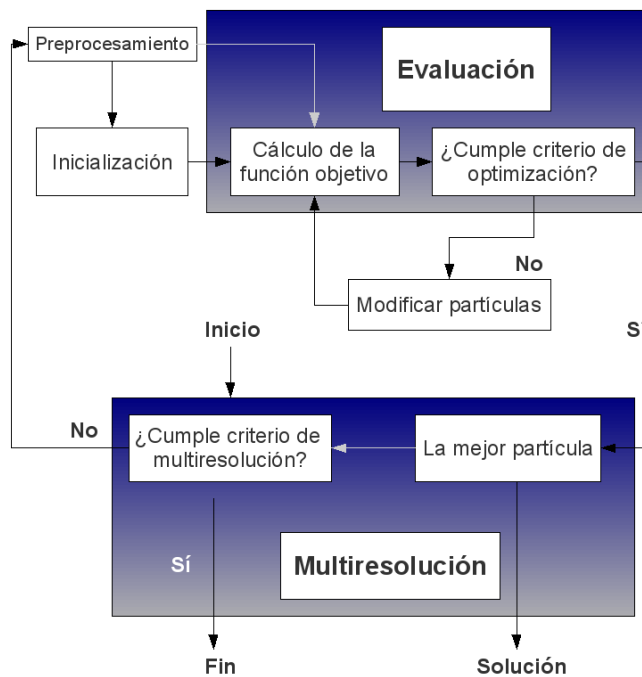


Figura 4.1: Diagrama de bloques del método de generación de moldes PSO. Una vez inicializadas la partículas para un tamaño de molde dado, ésta se modifican hasta alcanzar el criterio de optimización. La mejor partícula se almacena, y las partículas son reinicializadas para un molde de tamaño distinto en la etapa de multiresolución.

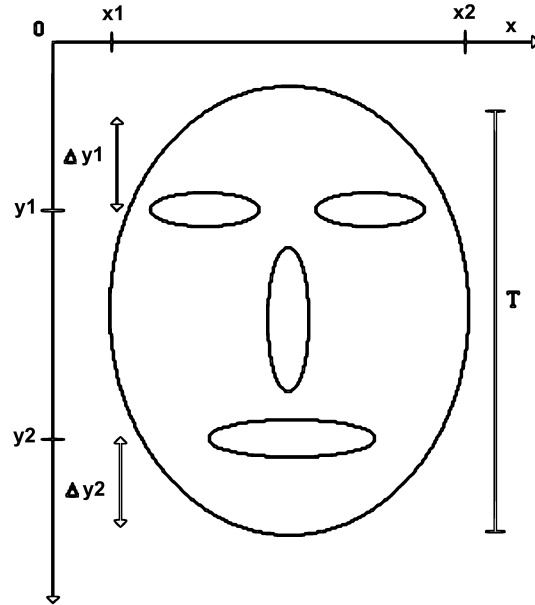


Figura 4.2: Parámetros para la segmentación del rostro. Se utilizan características antropométricas para determinar los parámetros.

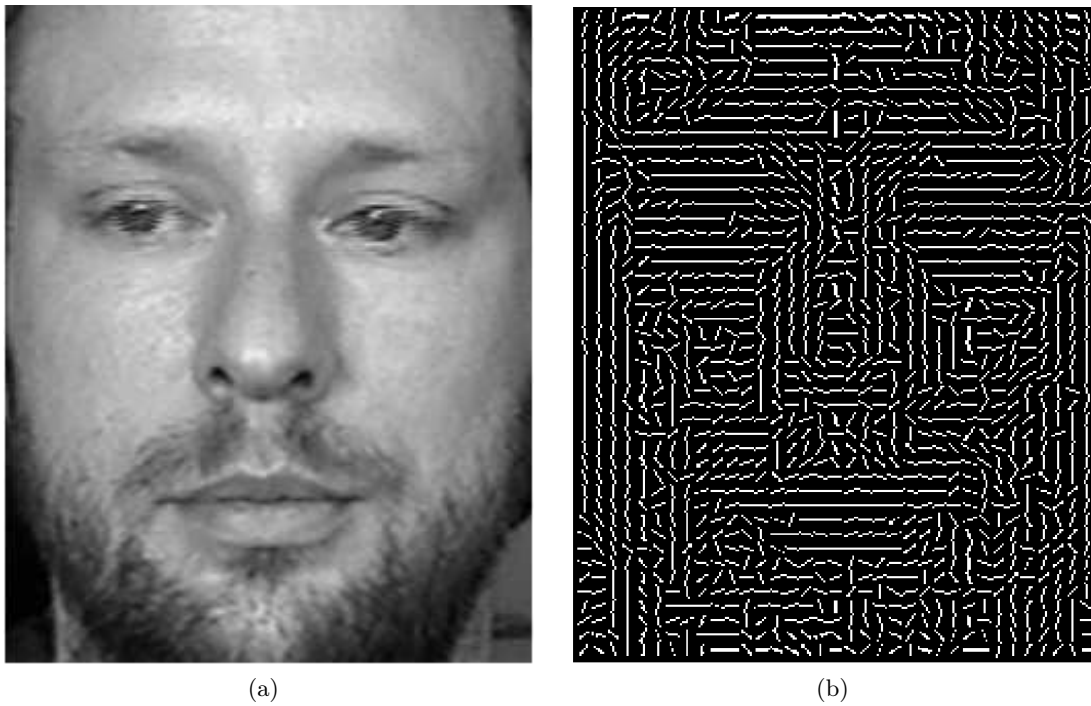


Figura 4.3: (a) Rostro frontal recortado. (b) Imagen direccional correspondiente a (a) .

[28]. En este último caso, se utilizan relaciones antropométricas del rostro para realizar el recorte y posee la ventaja adicional de que los rostros quedan alineados (con los ojos en el mismo lugar), lo que acelera notablemente la convergencia del algoritmo. Los rostros recortados son convertidos a escala de grises y normalizados a varias escalas que generan moldes de distinto tamaño. Una imagen direccional se [15, 24, 29, 30, 32] calcula como se muestra en 4.3. La imagen direccional (ver Anexos) se calcula sobre vecindades de 3×3 pixeles y contiene información acerca de los vectores tangentes a los bordes de la imagen. El conjunto de imágenes direccionales, una para cada imagen de entrada y para cada tamaño a optimizar, es utilizada posteriormente por el algoritmo PSO.

4.1.2. Inicialización de los moldes

Un molde es una imagen direccional que se construye a partir de una imagen rectangular en que cada pixel representa un ángulo entre 0 y 180 grados cuantizados en pasos dados por $180/w_d$. Para este trabajo se utilizó $w_d = 36$, resultando un paso de 5° . La información angular es almacenada en cada pixel del molde en un valor entre 0 y 36 y con un valor de 255 para los pixeles que no serán considerados como parte del molde y, por tanto, no poseen información angular.

Para la optimización PSO, la posición de cada partícula está definida como un vector de dimensión N, igual al número total de pixeles en el molde. Este valor varía según el tamaño del molde que se esté optimizando. Cada componente ($X_{i,l,m}$) de la partícula X_i representará entonces, a un pixel determinado (l, m) y el rango de posibles valores a adquirir estará determinado por el paso de cuantización utilizado ($w_d = 36$) y por un margen (m_R), que permite definir si dicho componente (o pixel) estará o no presente en el molde. Este rango viene dado por $[0, w_d + 2 \cdot m_R]$. Para la implementación computacional utilizada, se considera $m_R = 110$, ya que de esta forma los pixeles se mueven en el rango $[0, 255]$. Sólo los componentes del vector que son parte del molde son utilizados para calcular el fitness (integral de línea) del algoritmo. La figura 4.4 y las siguientes ecuaciones muestran el rango donde un pixel del molde PSO, o componente del vector posición de la partícula, será considerado como parte del molde:

$$\begin{aligned}
 \text{si } X_{i,l,m} \in [m_R, w_d + m_R] &\Rightarrow \\
 MM_{i,l,m} &= X_{i,l,m} - m_R \\
 IM_{i,l,m} &= 1
 \end{aligned} \tag{4.4}$$

$$\text{si } X_{i,l,m} \notin [m_R, w_d + m_R] \Rightarrow IM_{i,l,m} = 0. \tag{4.5}$$

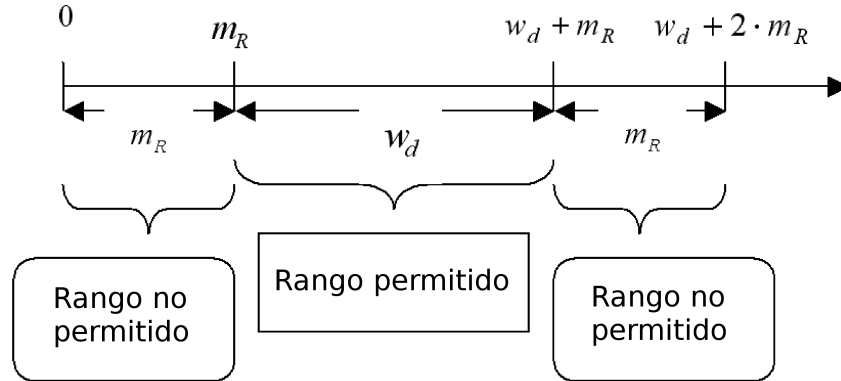


Figura 4.4: Rango de inicialización de los píxeles del molde PSO. Los píxeles contendrán información angular relevante sólo si se encuentran dentro del rango permitido.

$IM_{i,l,m}$ corresponde al módulo del vector direccional en la posición (l, m) del molde (X_i) y representa si el píxel (l, m) posee información angular relevante (es decir, pertenece al molde), mientras que $MM_{i,l,m}$ es el correspondiente ángulo del vector direccional (cuantizado de 0 a 36) para el píxel (l, m) de la partícula (X_i)

Un cambio de la posición de la partícula en el espacio de búsqueda, puede modificar el valor de uno o varios píxeles del molde PSO variando su información angular. Ésta puede tomar un valor dentro del intervalo $[m_R, w_d + m_R]$, o puede tomar un valor fuera de ese rango, desapareciendo por tanto, del molde PSO (representado por un cambio de 1 a 0 en el valor de $IM_{i,l,m}$). La posición y velocidad de cada partícula son inicializadas aleatoriamente en el rango $[0, w_d + 2 \cdot m_R]$ para la posición y en el rango $[-w_d, w_d]$ para la velocidad. El tamaño (dimensión) de la partícula dependerá por supuesto, del tamaño del molde a generar.

4.1.3. Evaluación

Para medir cuantitativamente la calidad de una solución (partícula) frente a otra, se define una función de fitness. Dado que la integral de línea provee de una medida de similitud entre el molde y la imagen direccional [15], será utilizada como función de fitness para el algoritmo de optimización. La integral de línea f_i^k para una partícula X_i está dada por:

$$f_i^k = \frac{1}{N_R \cdot PM_i} \sum_{h=0}^{N_R-1} \sum_{l=0}^{N_w} \sum_{m=0}^{N_H} IM_{i,l,m} \cdot (I_{mod_{h,l,m}} \cdot \alpha_{i,h,l,m} - (1 - I_{mod_{h,l,m}}) \cdot 18), \quad (4.6)$$

donde $I_{mod_{h,l,m}}$ indica que la imagen direccional del rostro de entrada h , presenta información angular en la posición (l, m) . N_R es el número de rostros frontales de entrada al algoritmo, N_w es el ancho de la partícula (molde) mientras que N_H es su alto. PM_i es el número total de puntos con información angular de la partícula o molde X_i . El índice k indica el número de iteración. El factor $\alpha_{i,h,l,m}$, denominado disparidad angular [29], corresponde a la diferencia angular de una coordenada particular (l, m) en el molde PSO con la correspondiente componente en la imagen direccional h . Este factor viene definido por:

$$\alpha_{i,h,l,m} = 18 - 2 \cdot \min(|MM_{i,l,m} - I_{ang_{h,l,m}}|, 36 - |MM_{i,l,m} - I_{ang_{h,l,m}}|), \quad (4.7)$$

donde

$I_{ang_{h,l,m}}$ = ángulo de la imagen direccional en la coordenada (l, m) para el rostro h .

Además, los valores de la integral de línea para cada pixel (l, m) del molde se almacenan en cada partícula para la próxima iteración de la siguiente manera:

$$f_{i,l,m}^k = \frac{\sum_{h=0}^{N_R-1} (I_{mod_{h,l,m}} \cdot \alpha_{i,l,h,m} - (1 - I_{mod_{h,l,m}}) \cdot 90^0)}{N_R}. \quad (4.8)$$

El valor de la integral de línea es penalizado por un factor (N_R) que toma en cuenta el número total de puntos en cada partícula, es decir, el número total de puntos con información angular del molde PSO. Este factor permite seleccionar aquellos moldes con menor número de puntos y que por tanto permiten acelerar el cálculo de la integral de línea, criterio de vital importancia en aplicaciones de tiempo real.

El algoritmo se detiene si se satisface el criterio de detención o continúa iterando hasta que se cumpla dicho criterio. Posibles criterios de detención a utilizar son:

- *Valor umbral:* Si la integral de línea de la mejor partícula sobrepasa un umbral dado, el algoritmo se detiene. El máximo valor para la integral de línea es de 18 y el umbral fue establecido en 13.5, que es un valor mayor a los obtenidos, en promedio, por los moldes antropométricos [15].
- *Número máximo de iteraciones:* El algoritmo se detiene cuando se alcanza un número máximo de iteraciones.

Ambos criterios pueden ser combinados de manera de mejorar la estabilidad y la convergencia del algoritmo.

4.1.4. Modificación de las partículas

La modificación de la posición y velocidad de las partículas se lleva a cabo de acuerdo a los siguientes criterios [43, 45]:

- *Actualización de la Mejor Partícula Individual:* Para cada partícula, si el valor de la integral de línea (f_i^k) en la iteración actual k es menor que el de la mejor partícula individual, la mejor partícula es modificada sólo si los valores de los componentes de la integral de línea ($f_{i,l,m}^k$) son mayores que los valores anteriores, i.e.,

$$\text{si } f_{i,l,m}^k \geq \max_{n=0\dots k-1} \{f_{i,l,m}^n\} \Rightarrow P_{i,l,m} = X_{i,l,m}. \quad (4.9)$$

Si el valor actual de (f_i^k) es mayor que el de la mejor partícula individual, los componentes individuales de la mejor partícula son modificados sólo si los componentes de la integral de línea ($f_{i,l,m}^k$) son mayores que los valores anteriores o si estos valores están por debajo de un umbral T_{PI} . Esto permite eliminar las componentes con valor de integral de línea pequeño, dejando sólo las más relevantes para el molde final:

$$\text{si } f_{i,l,m}^k \geq \max_{n=0\dots k-1} \{f_{i,l,m}^n\} \quad \text{o} \quad f_{i,l,m}^k \leq T_{PI} \Rightarrow P_{i,l,m} = X_{i,l,m}. \quad (4.10)$$

- *Actualización de la mejor partícula dentro de una vecindad:* Para cada partícula los valores de la integral de línea se comparan dentro de una vecindad en torno a la partícula. El índice y la posición de la mejor partícula en la vecindad se almacena como $g(i)$. El parámetro T indica el tamaño de la vecindad:

$$g(i) = \max_{l=i-T, \dots, i+T} \left\{ \max_{j=0\dots k} \{f_l^j\} \right\}. \quad (4.11)$$

- *Parámetro de inercia:* este parámetro fue elegido como una función decreciente de la integral de línea, de manera de mejorar la convergencia. Este parámetro queda definido según lo propuesto en [46–48]:

$$w = \left[1 - \frac{f_i^k}{52} \right] \cdot \Gamma \quad (4.12)$$

donde Γ toma valores alternantes, con un período de 15 iteraciones, entre $\Gamma = 0,5$ para realizar la búsqueda local, y $\Gamma = 1,5$ para la búsqueda global. Una función decreciente no lineal también ha sido propuesta para Γ en trabajos anteriores [46, 49].

Con estos criterios bien definidos, la velocidad y posición de las partículas se modifican entonces de acuerdo a:

$$V_{i,l,m}(t+1) = w \cdot V_{i,l,m}(t) + \varphi_1 \cdot (P_{i,l,m} - X_{i,l,m}(t)) + \varphi_2 \cdot (P_{g(i),l,m} - X_{i,l,m}(t)) \quad (4.13)$$

y

$$X_{i,l,m}(t+1) = X_{i,l,m}(t) + V_{i,l,m}(t+1). \quad (4.14)$$

4.1.5. Multiresolución

La última etapa de la metodología corresponde a la utilización de multiresolución. Aumentando la dimensionalidad de las partículas, los parámetros se ajustan de manera de transformar la escala de los moldes y obtener moldes PSO para distintos tamaños de rostros. En esta etapa, se busca y almacena la partícula con mejor fitness que pasa a ser el molde PSO del tamaño especificado. Los moldes son optimizados para tamaños finales que van desde 120 a 240 pixeles de semieje mayor. Sin embargo, la información angular se encuentra “comprimida” en un factor dado por la vecindad utilizada para calcular la imagen direccional (3×3 en este caso), por lo que el tamaño de los moldes utilizados por el algoritmo PSO va desde 30×40 pixeles hasta 60×80 pixeles.

Las iteraciones sucesivas del algoritmo, optimizando los moldes mediante PSO, permiten crear un conjunto de moldes optimizados de distinto tamaño, que pueda ser utilizado para un individuo en particular.

4.2. Pruebas realizadas

4.2.1. Secuencias de video

Los moldes PSO fueron comparados con los Moldes Antropométricos (Fig. 4.5) calculando el valor de la integral de línea de ambos tipos de moldes sobre la imagen direccional del rostro

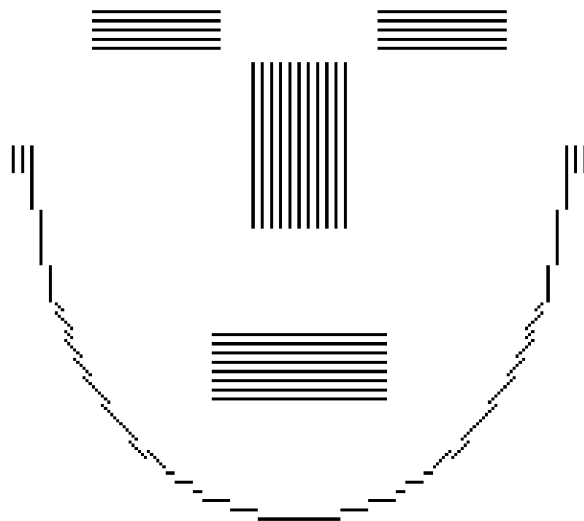


Figura 4.5: molde Antropométrico.

Tabla 4.1: Base de datos de videos utilizadas para las pruebas de los moldes PSO

Secuencia de video	Número de Cuadros
1	466
2	202
3	306
4	395

para cuatro secuencias de video con rostros sin rotación coronal (Tabla 4.1). Cada secuencia de video presenta el rostro frontal de un individuo distinto, y fueron capturadas en el Laboratorio de Procesamiento de Imágenes, Universidad de Chile. Para esta prueba, se crearon moldes PSO personalizados, un conjunto para cada persona, utilizando los 6 primeros cuadros tomados de la secuencia de video (no incluidos en el conjunto de prueba final).

Adicionalmente, se estudió la selectividad espacial de los moldes PSO y Antropométricos, mediante el cálculo de la integral de línea en una pequeña ventana de 11x11 píxeles en torno al centro real del rostro. Este cálculo se realizó para cada frame de las secuencias de video y para cada imagen de la base de datos.

4.2.2. Base de datos Yale B

Para esta base de datos, se utilizó un conjunto de moldes PSO genérico, así como moldes PSO personalizados para cada sujeto de la base de datos de Yale B. Los moldes personalizados se construyeron, utilizando 4 imágenes por sujeto. Estas imágenes fueron seleccionadas a partir



Figura 4.6: (a) Imágenes utilizadas para generar el conjunto de moldes PSO genérico (b) Cuatro imágenes seleccionadas (pose 0) de la base de datos Yale B, utilizados para la creación del conjunto de moldes PSO del individuo yaleB01.

del subconjunto de imágenes con pose frontal de cada individuo (pose 0). La figura 4.6.a muestra las imágenes utilizadas para generar el conjunto de moldes PSO genérico y corresponde a fotografías de cuatro estudiantes del Departamento de Ingeniería Eléctrica. La figura 4.6.b muestra cuatro imágenes del individuo YaleB01, utilizadas para la optimización de los moldes mediante PSO.

Se realizaron pruebas de tasa de detección de rostros sobre esta base de datos de manera de medir la robustez y generalización de los moldes PSO, incluyendo diferentes condiciones de iluminación y poses con algún grado de rotación no coronal (aproximadamente 24^0). Los moldes PSO fueron comparados contra los moldes Antropométricos [15] y el método de detección de Viola y Jones ($Adaboost_{V,J}$) [20]. Se utilizó la totalidad de la base de datos de Yale B para realizar esta prueba (más de 5000 imágenes).

4.2.3. Base de datos PURDUE

Los moldes PSO fueron compararon en una aplicación de detección de iris con un método que usa información combinada de intensidad y bordes binarios [25]. En [25], el método se aplicó a dos subconjuntos de la base de datos de PURDUE, denominados AR-63 y AR-564, mostrando buenos resultados en la localización del iris. Para esta prueba, se implementó un detector de iris utilizado en trabajos anteriores [15], basado en la detección del borde iris-esclera. Se utilizaron los mismos subconjuntos AR-63 y AR-564 y la misma métrica (ver Capítulo 2)

de manera de poder comparar los resultados directamente. Se crearon un conjunto de moldes PSO especialmente adaptado a la base de datos PURDUE, utilizando rostros contenidos en dicha base de datos, pero que no formaran parte de los subconjuntos mencionados. También se utilizó para estas pruebas, el conjunto de moldes PSO genérico explicado anteriormente.

Capítulo 5

Resultados Experimentales

5.1. Transformada Elíptica Lineal (LET)

Se procesaron imágenes de las siguientes cuatro bases de datos de imágenes: FERET, CALTECH, PURDUE y Yale-B, así como rostros rotados de cuatro secuencias de video de la base de datos interna del laboratorio [15], usando la metodología LET para la estimación de parámetros. Los resultados fueron comparados con el detector de rostros de Viola y Jones (Adaboost_{VJ}) [20].

En la figura 5.1 se muestran ejemplos de una estimación correcta de la elipse del rostro. Para las distintas imágenes de entrada (a,c,e,g) se observa su correspondiente transformada LET (b,d,f,h). Se puede apreciar como la elipse estimada se ajusta de muy buena manera al rostro del individuo y que el contorno semi-elíptico inferior del rostro es convertido a una línea vertical en el cuadrante superior izquierdo de la imagen transformada por LET. La fig 5.2 muestra los resultados sobre imágenes de las secuencias de video procesadas. Se puede apreciar como la rotación de elipse se ajusta de manera de coincidir con la del individuo.

5.1.1. Selectividad de los parámetros de la LET

La Fig. 5.3 muestra la selectividad que posee la LET para seleccionar los distintos parámetros del rostro para una imagen cualquiera de la base de datos de CALTECH. Se observa como el máximo valor de V_{LET} coincide en cada caso, con los parámetros correctos de posición horizontal (a), posición vertical (b), excentricidad (c) y rotación coronal del rostro (d).



Figura 5.1: Ejemplos de localización correcta del rostro en las bases de datos de FERET (a,c), CALTECH (e) y PURDUE (g), con su respectiva elipse de salida. En (b,d, f, h) se muestra la imagen LET asociada. El punto blanco corresponde al centro del rostro dado por LET. El punto negro a la posición estimada por la etapa de detección gruesa.

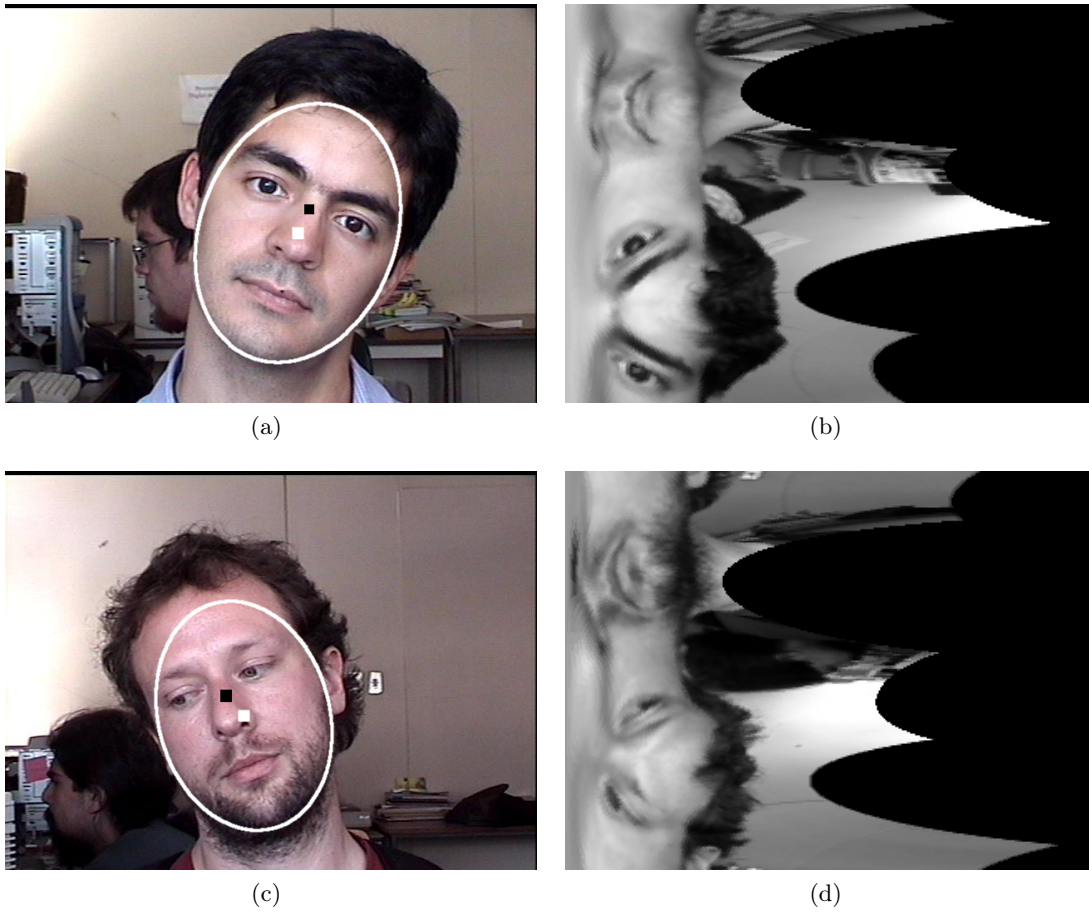
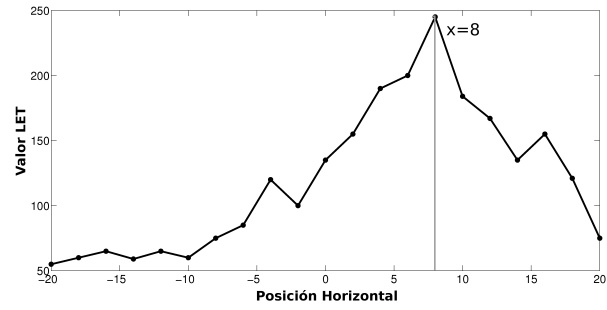
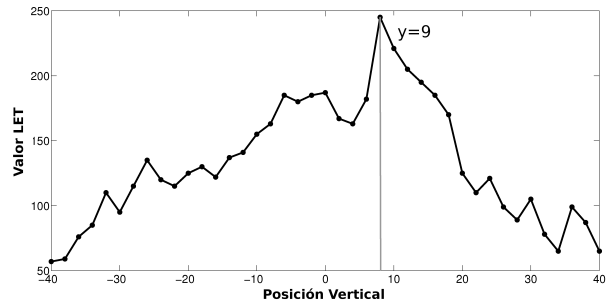


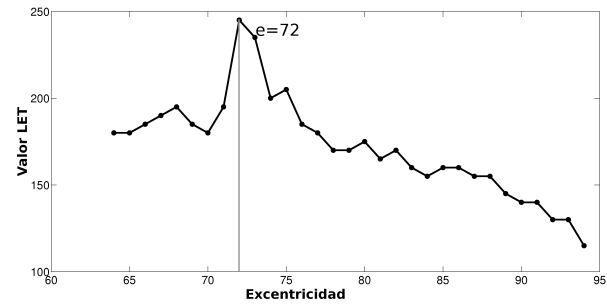
Figura 5.2: Ejemplos de localización correcta del rostro en dos de las cuatro secuencias de video. En (a, c) se muestra la salida del método (elipse dada por LET). (b,d) Imagen LET asociada. El punto blanco corresponde al centro del rostro dado por LET. El punto negro a la posición estimada por la etapa de detección gruesa.



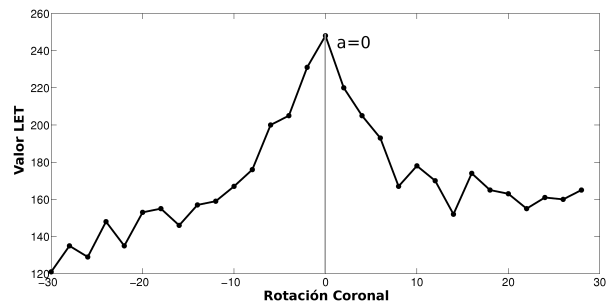
(a)



(b)



(c)



(d)

Figura 5.3: Selectividad de la LET al calcular cada uno de los parámetros del rostro. Los parámetros restantes se mantienen fijos en el valor correcto. (a) Posición Horizontal (x_0). (b) Posición Vertical (y_0). (c) Excentricidad. (d) Ángulo de rotación coronal.

Tabla 5.1: Error absoluto (en píxeles) en la estimación de parámetros del rostro mediante LET y Adaboost_{VJ}. Estas estadísticas fueron calculadas sobre las cuatro bases de datos de imágenes.

	PURDUE		CALTECH		FERET		YALE B	
	LET	Ada.VJ	LET	Ada.VJ	LET	Ada.VJ	LET	Ada.VJ
Posición Horizontal (x_0)	4.5	4.8	8.6	10.5	8.1	8.7	12.1	16.7
Posición Vertical (y_0)	14.9	12.5	16.6	13.1	15.2	13.5	14.7	21.2
Excentricidad (ϵ)	8.8	15.7	9.3	14.6	7.7	13.9	7.8	13.1
Tamaño (r)	14.8	18.2	13.4	12.1	18.4	22.5	11.9	10.0

5.1.2. Estimación de parámetros

El cálculo del error absoluto en la estimación de los parámetros del rostro (x_0, y_0, e, r) se resume en la tabla 5.1 para las bases de datos de FERET, CALTECH, PURDUE y Yale-B. El error absoluto es medido en píxeles para la posición y tamaño del rostro. La Tabla 5.2 muestra los resultados de error absoluto en la posición, excentricidad, rotación y tamaño para las 4 secuencias de video procesadas mediante LET y el método Adaboost_{VJ}. Se puede observar que el error absoluto es más pequeño en la estimación de la posición horizontal y la excentricidad, mientras que se comete un error más grande al estimar la posición vertical y el tamaño del rostro. Para el caso de las secuencias de video utilizadas, los errores fueron en general de menor orden aún que los presentados sobre las bases de datos, aún incluyendo la estimación adicional del parámetro de rotación coronal.

El error absoluto en la localización del rostro nos da un estimador del error que se comete, en píxeles, al estimar la posición y tamaño del rostro. El error en la excentricidad se puede interpretar como la diferencia en la relación de aspecto entre el rostro detectado y el seleccionado en forma manual.

El error promedio en píxeles para la posición y el tamaño (Tabla 5.1) muestra que LET tiene mejor desempeño que Adaboost_{VJ} al estimar la posición horizontal y es similar al estimar el tamaño del rostro. El detector Adaboost_{VJ} presenta una ventaja sobre LET estimando la posición vertical, básicamente debido a errores producidos en la etapa de detección gruesa. Sobre la base de datos de Yale B, LET claramente es superior a Adaboost_{VJ} para la estimación de la posición y la excentricidad, siendo similar en el caso del tamaño, debido principalmente a las condiciones extremas de iluminación de dicha base de datos, que impide una buena aplicación del detector Adaboost_{VJ}.

Tabla 5.2: Error absoluto (en pixeles) en la estimación de parámetros del rostro mediante LET. Estas estadísticas fueron calculadas sobre las cuatro secuencias de video utilizadas.

	Video 1	Video 2	Video 3	Video 4
Posición Horizontal (x_0)	5.1	5.3	5.0	5.7
Posición Vertical (y_0)	12.4	13.7	12.5	14.6
Excentricidad (ϵ)	7.8	7.9	7.6	8.1
Tamaño (r)	11.2	13.2	12.3	13.4
Ángulo de rotación (φ)	5.3	5.7	5.6	6.0

Tabla 5.3: Tasa de detección de rostros (%) para las metodologías LET y Adaboost_{VJ}.

	PURDUE	CALTECH	FERET	YALE B
LET	98.2	95.3	95.7	79.6
Adaboost _{VJ}	99.1	96.7	97.2	61.5

Se puede apreciar, asimismo, que LET presenta un menor error en la estimación de la excentricidad que el método Adaboost_{VJ}. Esto debido principalmente a que el primero, es capaz de entregar una “caja” de relación de aspecto variable entre alto y ancho, de manera que se ajusta mejor al rostro en cuestión.

Las figuras 5.4, 5.5, 5.6 y 5.7 muestran para cada uno de los cuatro parámetros estimados del rostro (x_0, y_0, e, r), el porcentaje de imágenes que presentan una estimación correcta como función del error relativo (ver Capítulo 2) para las bases de datos de PURDUE, CALTECH, FERET y YALE B. Para tolerancias de error bajas, la LET es más precisa que el método Adaboost_{VJ} para 3 de los 4 parámetros estimados, especialmente en el caso de la excentricidad y la posición horizontal. Asimismo, la LET se comporta en forma muy superior a Adaboost_{VJ} sobre YALE B. Este resultado es de extrema importancia para aplicaciones tales como reconocimiento de rostros, debido a que unos pocos pixeles de error en la detección del rostro, puede resultar en una identificación errónea del individuo. Por lo tanto, es importante comparar la mejora del método en el rango de un error relativo bajo.

5.1.3. Tasas de detección de rostros

La Tabla 5.3 muestra los porcentajes de detección de rostros para ambas metodologías (LET y Adaboost_{VJ}) según el criterio explicado en la sección 2.3.1. Se aprecia que la metodología Adaboost_{VJ} se comporta levemente mejor en las bases de datos con rostros bien iluminados, mientras que el desempeño de LET es mucho mejor en escenarios con iluminación y contraste complejos y no homogéneos (Yale B).

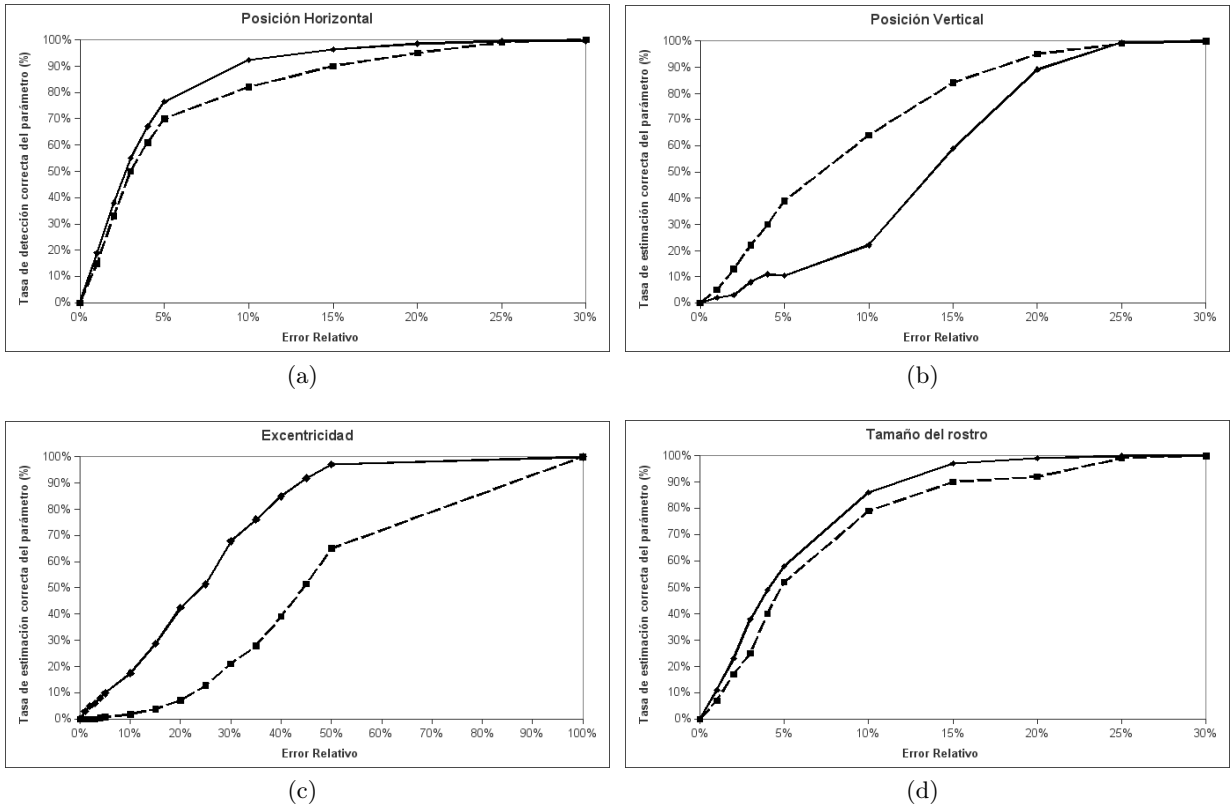


Figura 5.4: Porcentaje de imágenes con estimación correcta de parámetros versus el error relativo sobre la base de datos de PURDUE. (a) Posición Horizontal x_0 . (b) Posición Vertical y_0 . (c) Excentricidad ϵ . (d) Tamaño r del rostro. Se compara el desempeño de LET (línea continua) versus Adaboost_{VJ} (línea segmentada).

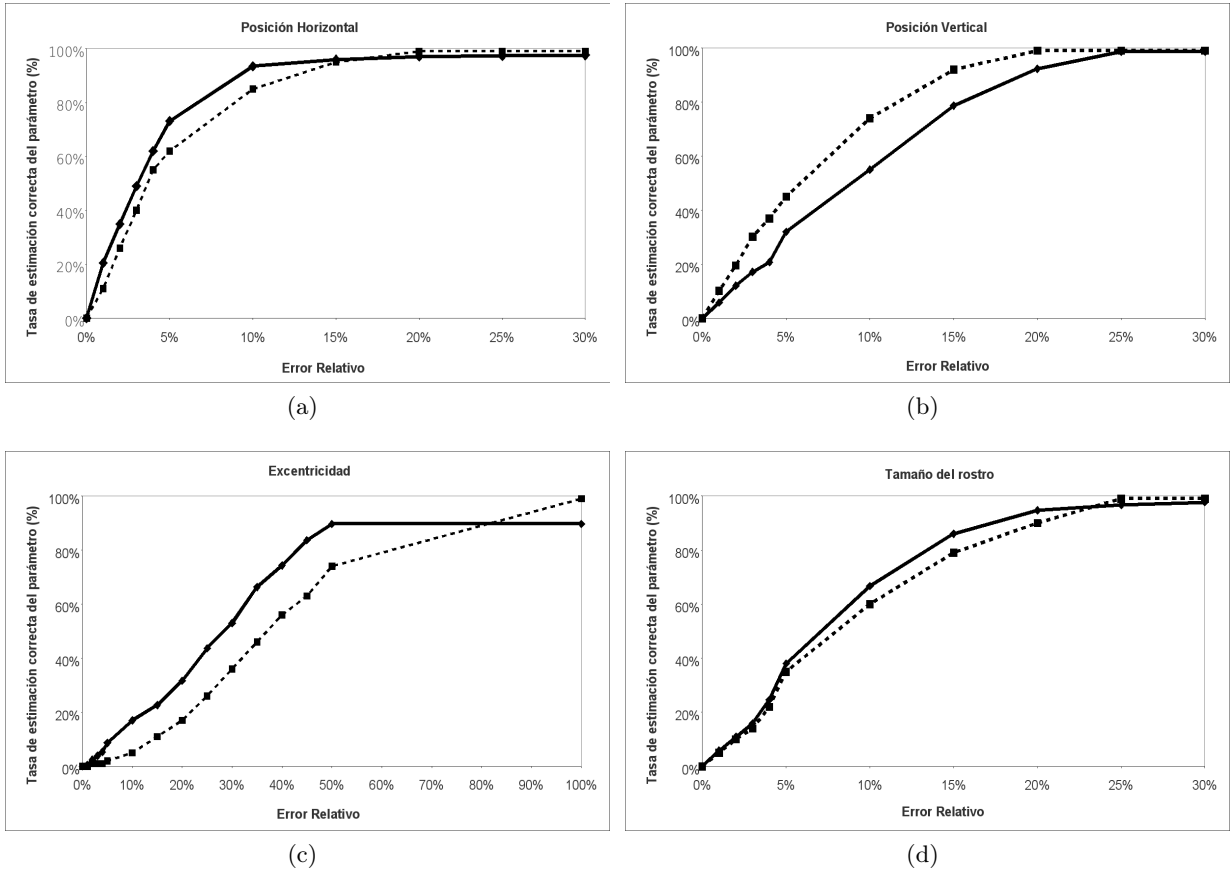


Figura 5.5: Porcentaje de imágenes con estimación correcta de parámetros versus el error relativo sobre la base de datos de CALTECH. (a) Posición Horizontal x_0 . (b) Posición Vertical y_0 . (c) Excentricidad ϵ . (d) Tamaño r del rostro. Se compara el desempeño de LET (línea continua) versus Adaboost_{VJ} (línea segmentada).

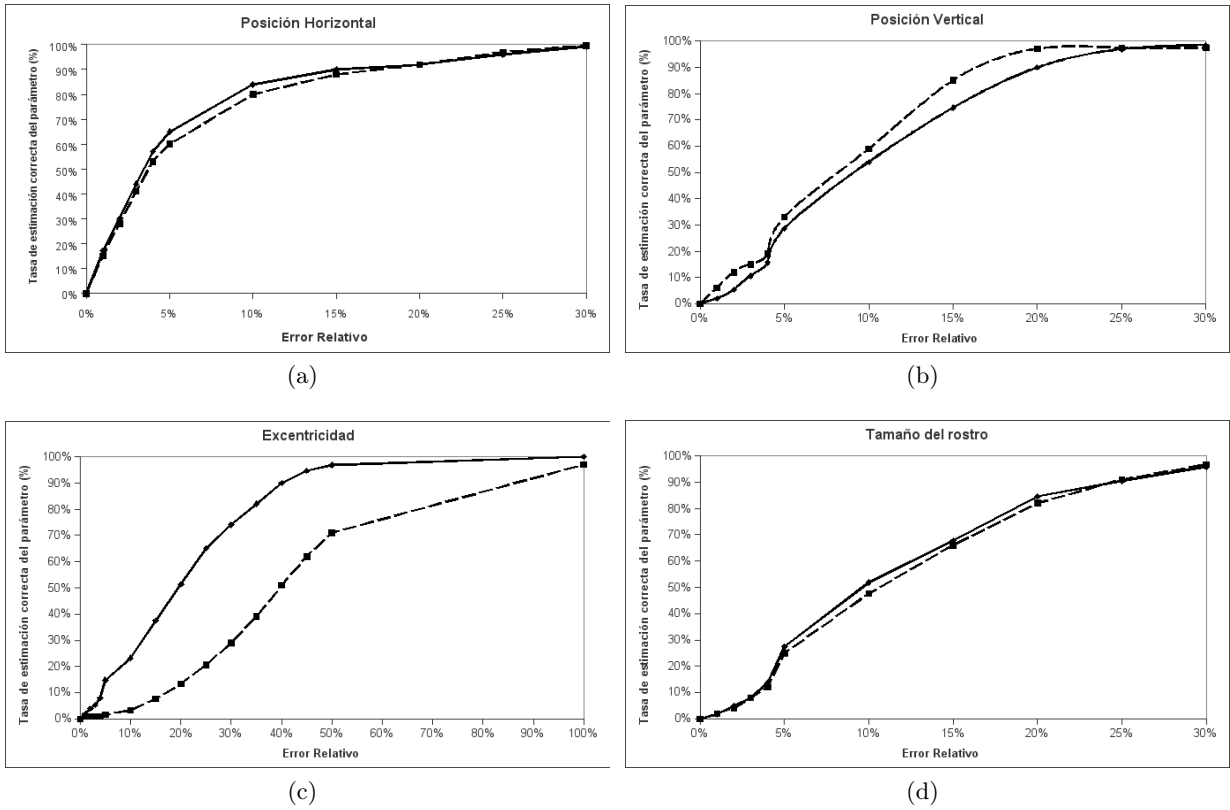


Figura 5.6: Porcentaje de imágenes con estimación correcta de parámetros versus el error relativo sobre la base de datos de FERET. (a) Posición Horizontal x_0 . (b) Posición Vertical y_0 . (c) Excentricidad ϵ . (d) Tamaño r del rostro. Se compara el desempeño de LET (línea continua) versus Adaboost $_{VJ}$ (línea segmentada).

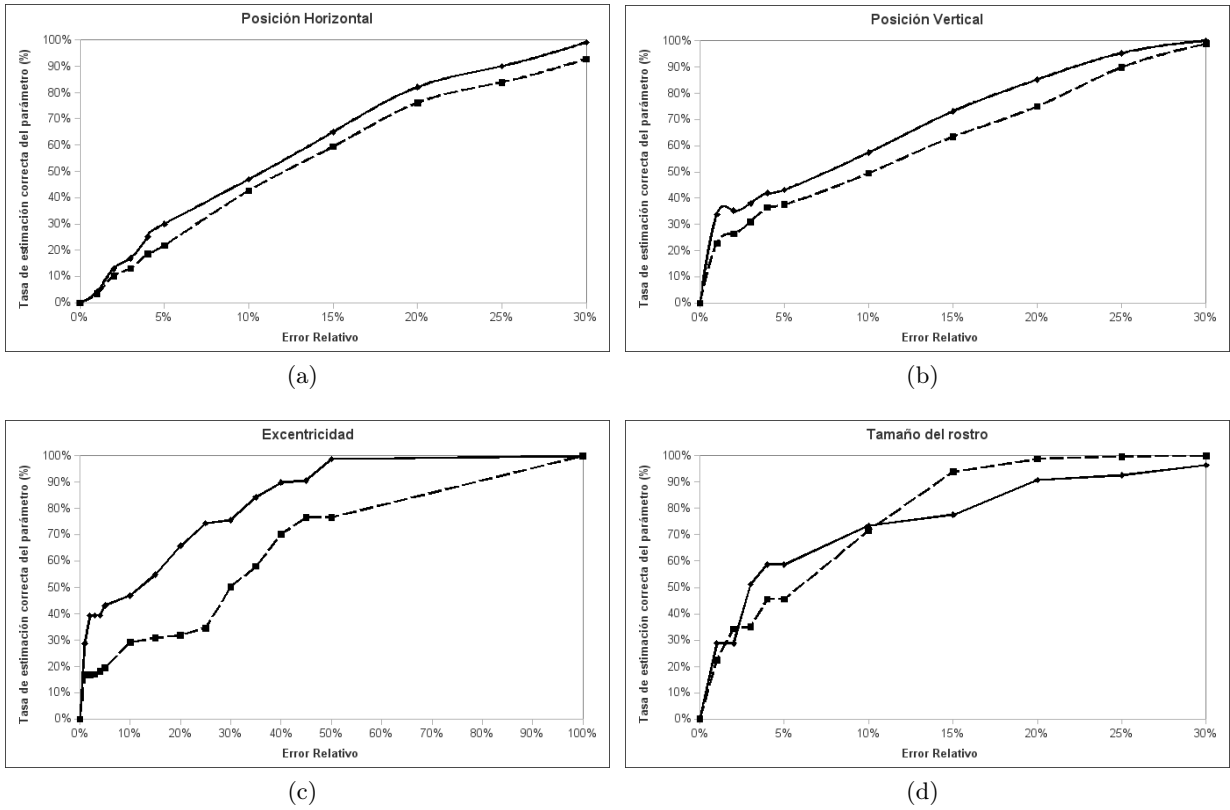


Figura 5.7: Porcentaje de imágenes con estimación correcta de parámetros versus el error relativo sobre la base de datos de YALE B. (a) Posición Horizontal x_0 . (b) Posición Vertical y_0 . (c) Excentricidad ϵ . (d) Tamaño r del rostro. Se compara el desempeño de LET (línea continua) versus Adaboost_{VJ} (línea segmentada).

Tabla 5.4: Tiempo de procesamiento de las principales funciones de la metodología LET.

Función	Tiempo de Procesamiento [ms]
Preprocesamiento	5.1
Detección Gruesa	175.3
LET	0.9
Imagen Direccional I_{DirLET}	2.7
Cálculo de V_{LET}	0.1

5.1.4. Tiempo de procesamiento

El tiempo de procesamiento de cada función principal de la LET fue medido, de manera de evaluar la posibilidad de aplicar el método en tiempo real para la localización y seguimiento de rostros. La Tabla 5.4 resume los tiempos empleados por cada función principal de la metodología, calculados sobre un computador Athlon64 3000+ con un 1GB de memoria RAM. La LET y la detección del borde vertical (cálculo de V_{LET}) son bastante rápidos, siendo la etapa de detección gruesa la más lenta de la metodología. Como se explica anteriormente, se debe renunciar a la búsqueda exhaustiva de parámetros si se quiere lograr una tasa de procesamiento aceptable para aplicaciones de video. Por lo tanto, los parámetros de localización del rostro se estiman sólo en una pequeña vecindad de 10×10 píxeles en torno a la posición del frame anterior, obteniéndose tasas cercanas a los 10 cuadros por segundo. La detección gruesa sólo se aplica al primer frame por lo que el desempeño del sistema es altamente dependiente de esta primera estimación. La excentricidad es estimada sólo en el primer cuadro y se mantiene constante para los cuadros subsiguientes.

5.1.5. Tipos de error detectados

Ejemplos de una estimación incorrecta de los parámetros del rostro se puede observar en la fig 5.8. Algunos errores se deben a una estimación incorrecta de la posición inicial dada por la detección gruesa (a, b), rostros no elípticos (c), o errores en la estimación dada por la LET (d).

5.2. Moldes PSO

La figura 5.9 muestra la evolución de un molde PSO de tamaño 200 píxeles (semieje mayor) a lo largo de 1000 iteraciones, para la optimización realizada sobre el conjunto de moldes genérico. El molde es inicializado en forma aleatoria (Fig. 5.9.a) y converge a una forma relativamente estable a medida que pasan las iteraciones (Figs. 5.9.b a 5.9.f) Se puede

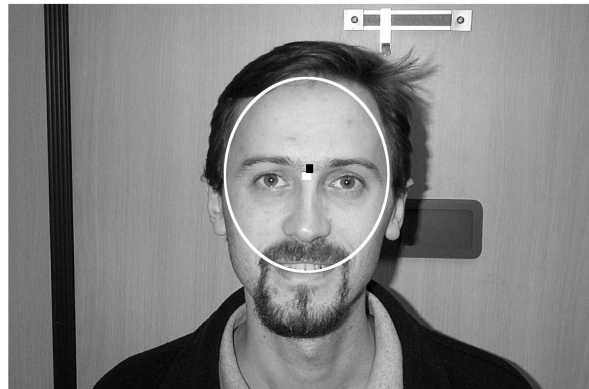


(a)

(b)



(c)



(d)

Figura 5.8: Ejemplos de estimación incorrecta de la elipse. (a-b) Errores en la etapa de detección gruesa. (c) Rostro no elíptico. (d) Estimación errónea de los parámetros mediante LET. El punto blanco corresponde al centro del rostro dado por LET, el punto negro a la posición estimada por la etapa de detección gruesa.

Tabla 5.5: Número total de pixeles del conjunto de moldes utilizado para procesar cada secuencia de video

Moldes	Número de puntos
Antropométricos	24429
PSO Genéricos	18768
PSO Personalizados	15528

observar asimismo, como el número de pixeles del molde decrece con el número de iteraciones, disminuyendo el número de cálculos involucrado en la integral de línea.

En la figura 5.10 se muestra como varía el valor del fitness (integral de línea) en función del número de iteraciones para la mejor partícula del enjambre. Para la mayoría de los casos, el valor máximo del fitness tiende a estabilizarse alrededor de las 1000 iteraciones, dependiendo del tamaño del molde (moldes más grandes demoran más).

La figura 5.11 muestra el resultado final de la optimización para el caso de 3 rostros distintos. En la columna izquierda (a,d,g) se observa el rostro de entrada en escala de grises y en la columna central su correspondiente imagen direccional (b,e,h). La columna de la derecha (c, f, i) se pueden apreciar los correspondientes moldes PSO. Al comparar el molde PSO específico de cada rostro con el molde antropométrico de la figura 4.5, se observa como el primero se adapta de mejor manera a la imagen direccional.

Se comparó la cantidad de puntos (y por ende de cálculos) de los moldes PSO y antropométricos en las cuatro secuencias de video. Debido a que el número total de cálculos necesarios para obtener el valor de la integral de línea es directamente proporcional al número de pixeles del molde, es deseable que el molde direccional presente la menor cantidad de puntos posible. Esto sin perjuicio de mantener un alto valor de la integral de línea y una buena selectividad espacial. La tabla 5.5, resume el número de pixeles utilizados para cada uno de los tres conjuntos de moldes generados. Se puede observar como los moldes PSO generan muchas menos operaciones de cálculo que los moldes antropométricos.

El tiempo empleado en la generación de un molde PSO depende del tamaño final que se requiera para el molde. La figura 5.12 es un ejemplo del tiempo empleado en generar un molde PSO como función del tamaño del molde. Todos los moldes fueron optimizados utilizando como criterio de detención un número de 1000 iteraciones. El tamaño (altura) de los moldes varió desde los 120 a los 240 pixeles.

Un resultado importante al usar los moldes PSO es que debido al menor número de puntos en el molde, el tiempo computacional requerido para detectar un rostro fue reducido a un 60%

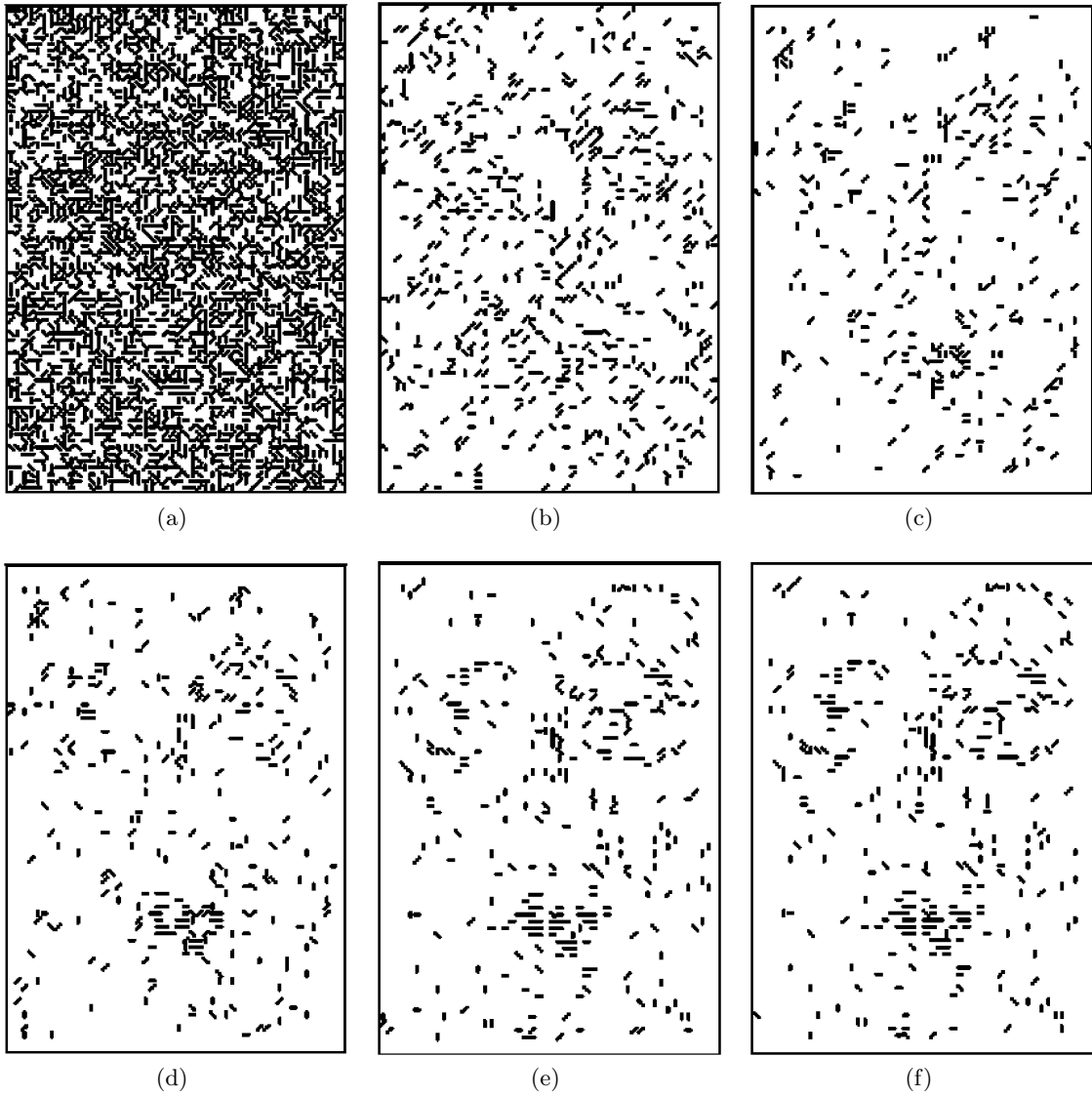


Figura 5.9: Evolución de un molde PSO de 200 píxeles (semieje mayor) para 1,000 iteraciones. (a) Inicialización aleatoria del molde, (b) iteración 8 , (c) iteración 111, (d) iteración 307, (e) iteración 690, y (f) molde final en la iteración 1000.

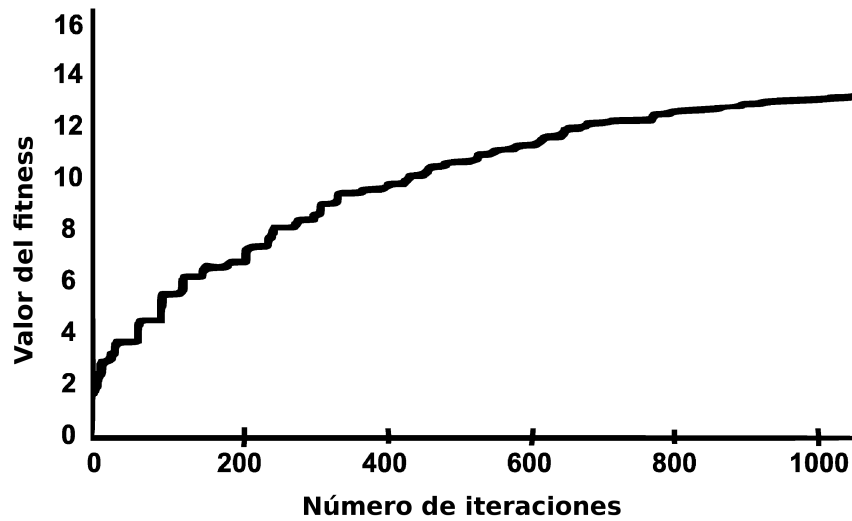


Figura 5.10: Evolución del fitness (valor de la integral de línea) en 1000 iteraciones.

Tabla 5.6: Comparación del tiempo computacional promedio empleado en detectar un rostro

molde	Tiempo Promedio [ms]	Desviación estándar
Antropométrico	28.5	0.0068
PSO	18.6	0.0053

del empleado al utilizar moldes antropométricos (Tabla 5.6). Estos valores fueron obtenidos para la secuencia de video 2, utilizando un procesador Athlon XP 2000+ de 1.67GHz con 512 MB RAM.

5.2.1. Estimación del Tamaño del Rostro

El error en la estimación del tamaño del rostro, fue calculado como el valor absoluto de la diferencia entre el valor de tamaño entregado por el molde y el valor marcado en forma manual (*ground-truth*). La figura 5.13 muestra el error en pixeles en la estimación del tamaño del rostro mediante moldes antropométricos (curva negra) y mediante moldes PSO personalizados (curva gris) aplicados a cuatro secuencias de video distintas (a-d). En el eje de las abscisas se grafica el número del cuadro. Se observa claramente como el error en la estimación del tamaño del rostro es casi siempre menor para el caso de los moldes PSO en relación con los moldes Antropométricos.

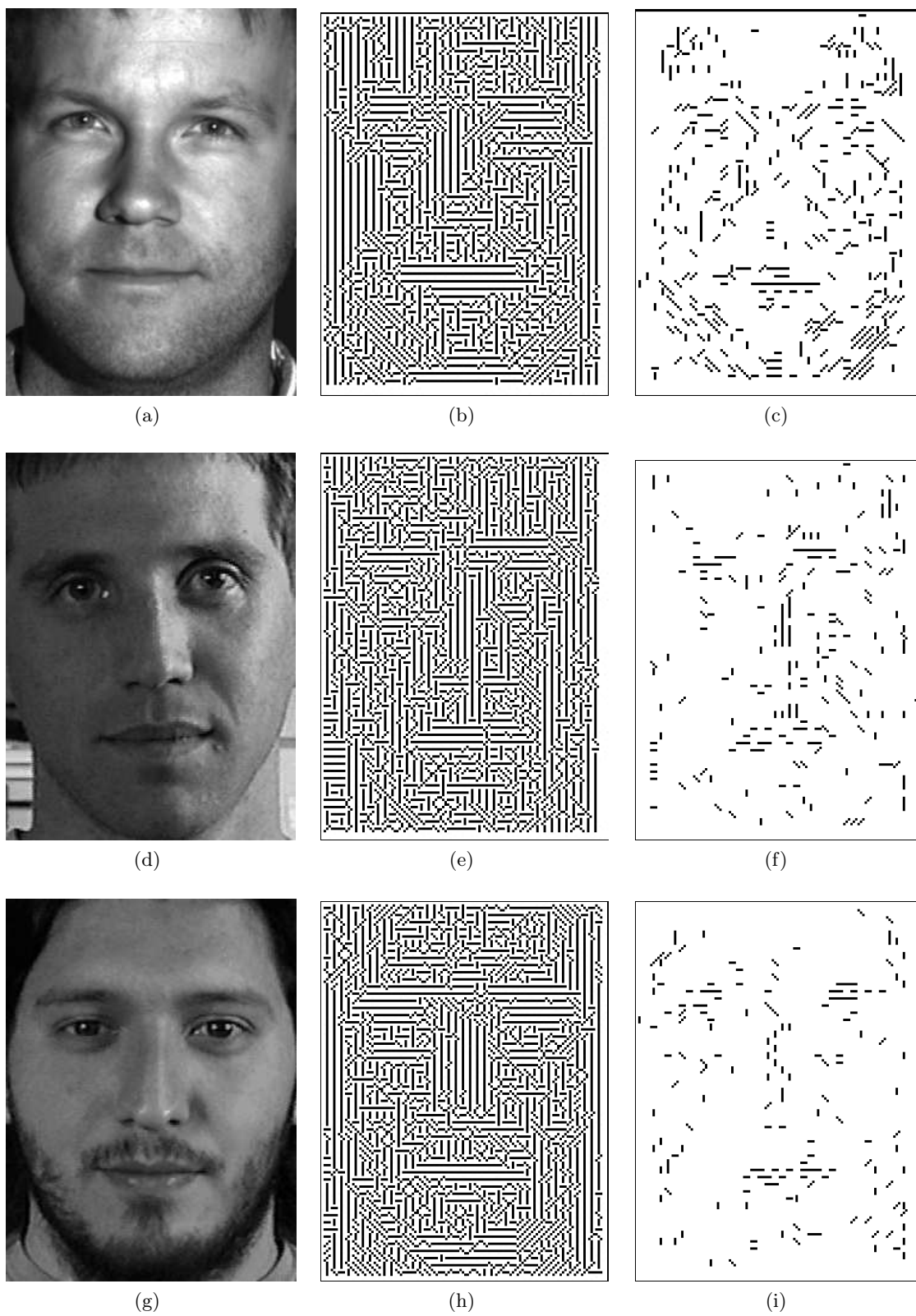


Figura 5.11: Rostros Frontales (a, d, g), su correspondiente imagen direccional (b, e, h), y el correspondiente molde PSO final (c, f, i).

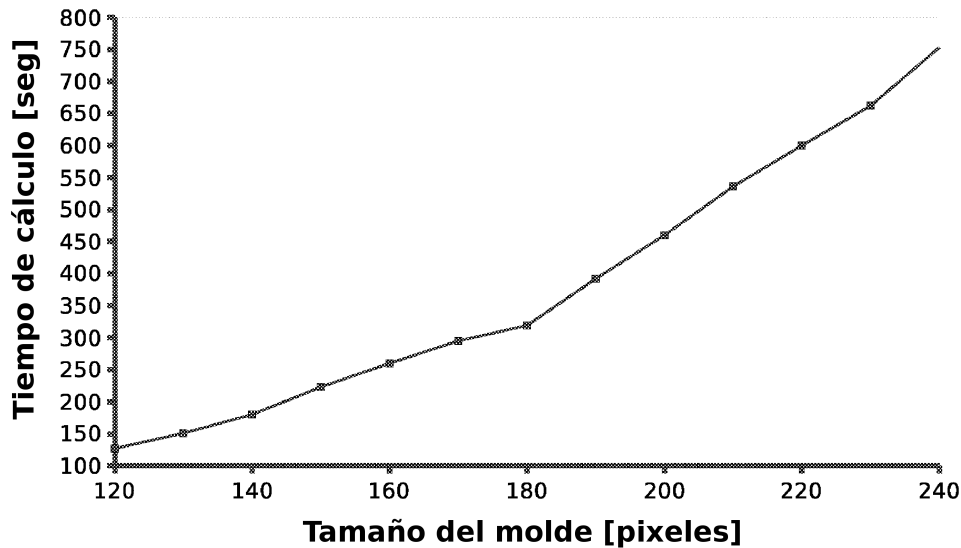


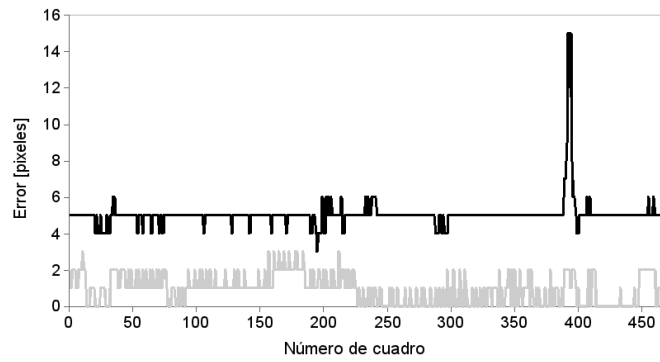
Figura 5.12: Tiempo de optimización mediante PSO para los distintos tamaños de molde.

5.2.2. Integral de línea

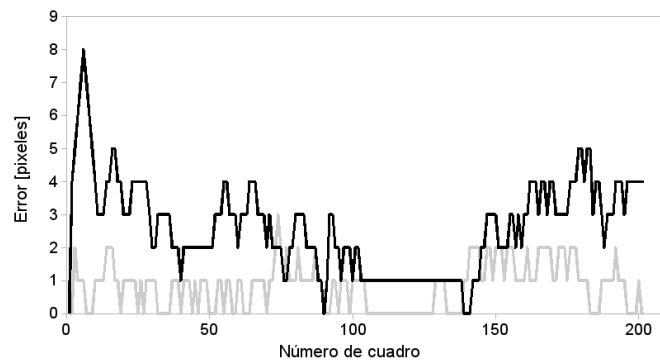
La figure 5.14 muestra la variación de la integral de línea calculada para cada *frame* de las 4 secuencias de video utilizadas (a-d) para el caso de los moldes PSO personalizados (curva gris) y los moldes antropométricos (curva negra). Se puede observar que para las cuatro secuencias de video y para todos los frames, el valor de la integral de línea es mayor al utilizar moldes PSO que cuando se usan los moldes antropométricos. Esto confirma la hipótesis de que los moldes PSO personalizados, al ser optimizados para cada individuo, presentan un mejor ajuste a las características direccionales del rostro.

5.2.3. Selectividad Espacial

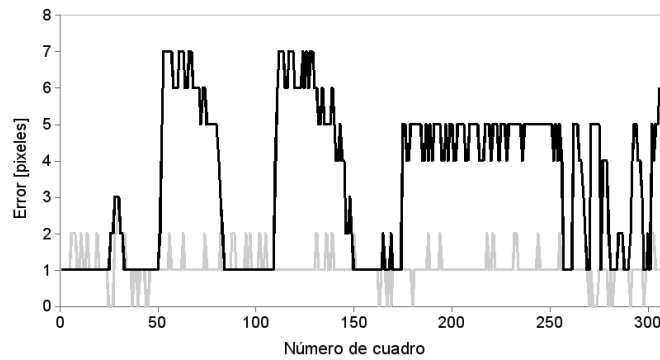
La selectividad espacial de los moldes PSO y Antropométricos puede ser apreciada calculando la integral de línea dentro de una vecindad de 11×11 píxeles alrededor del centro real del rostro (marcado manualmente). En la figura 5.15(a) se observa la imagen en escala de grises original del rostro, mientras que en las fig. 5.15(b) y 5.15(c) se observa el valor de la integral de línea dentro de la ventana de 11×11 píxeles en torno al centro, representado en niveles de gris, para los casos de los moldes PSO y Antropométricos respectivamente. Niveles de gris más oscuros representan valores más pequeños de la integral de línea, correspondiendo los valores negro (0) y blanco (255), al menor y mayor valor de la integral de línea dentro de



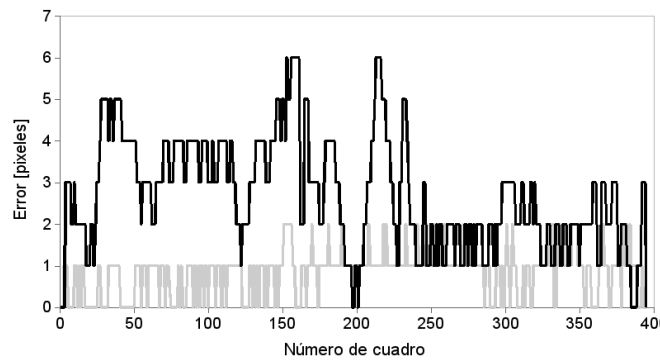
(a)



(b)

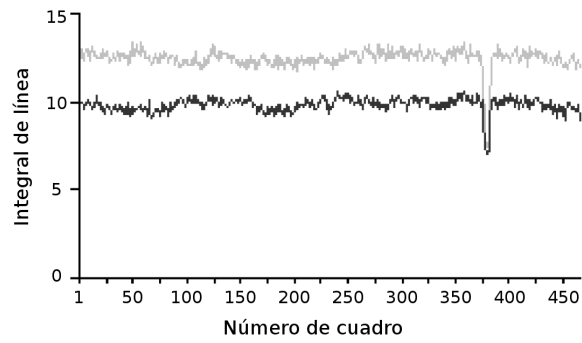


(c)

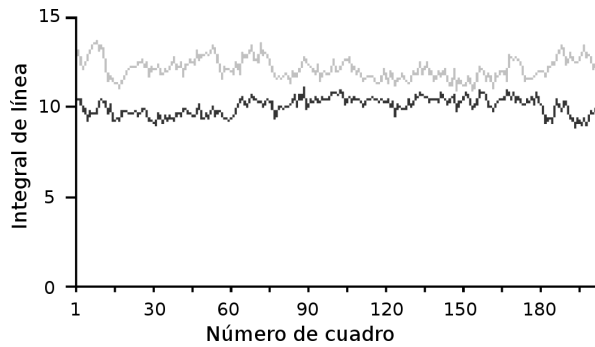


(d)

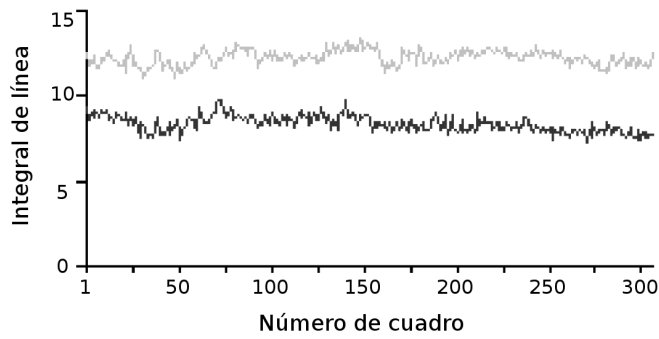
Figura 5.13: Error cuadro a cuadro, para la estimación del tamaño del rostro entre los moldes Antropométricos (negro) y los moldes PSO (gris) aplicados a cuatro secuencias de video (a-d).



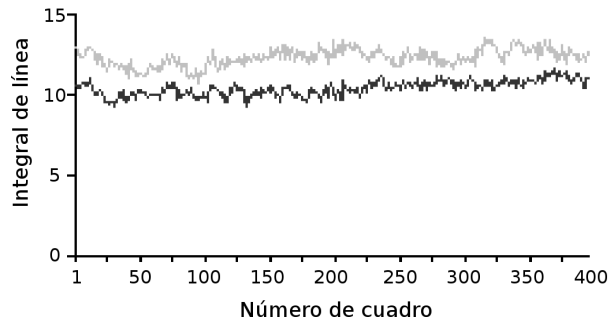
(a)



(b)

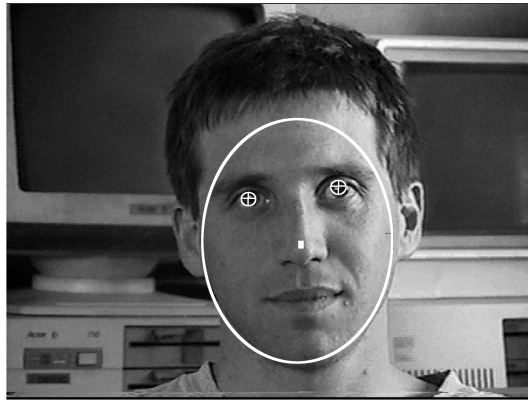


(c)

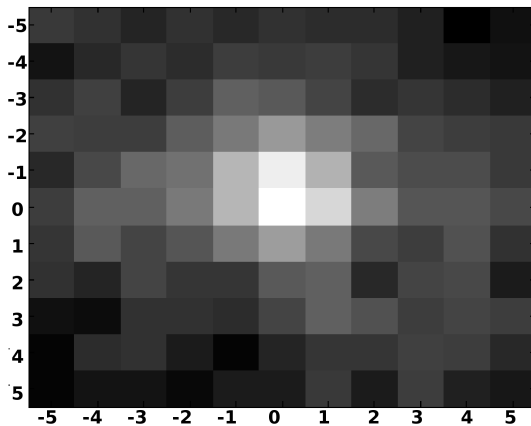


(d)

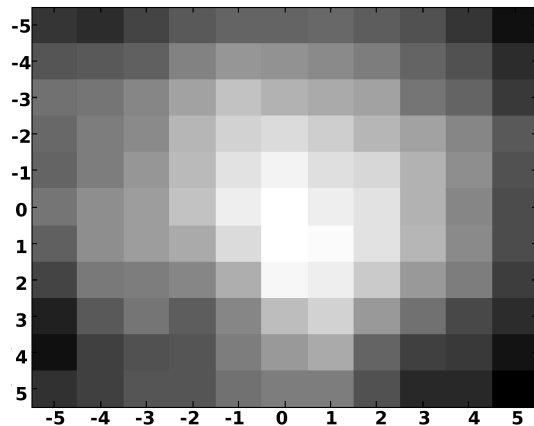
Figura 5.14: Valor de la integral de línea al utilizar moldes PSO (gris) y moldes Antropométricos (negro) aplicados sobre los cuadros de cuatro secuencias de video.



(a)



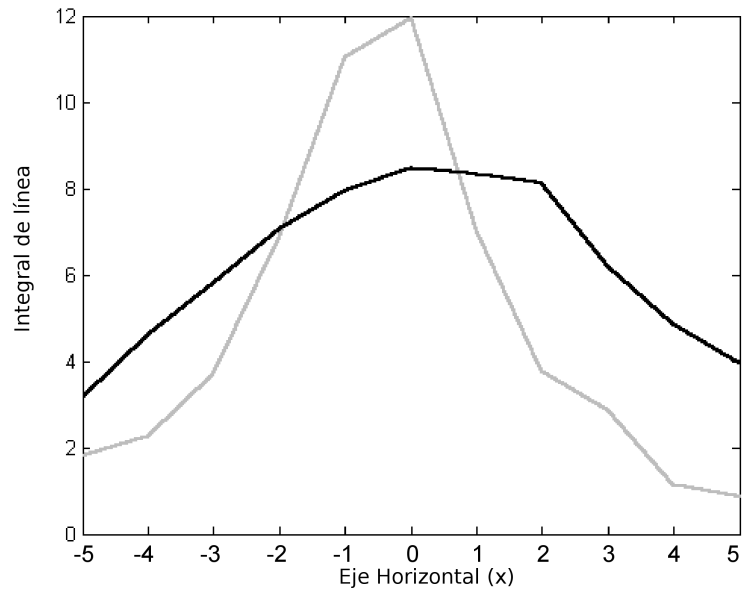
(b)



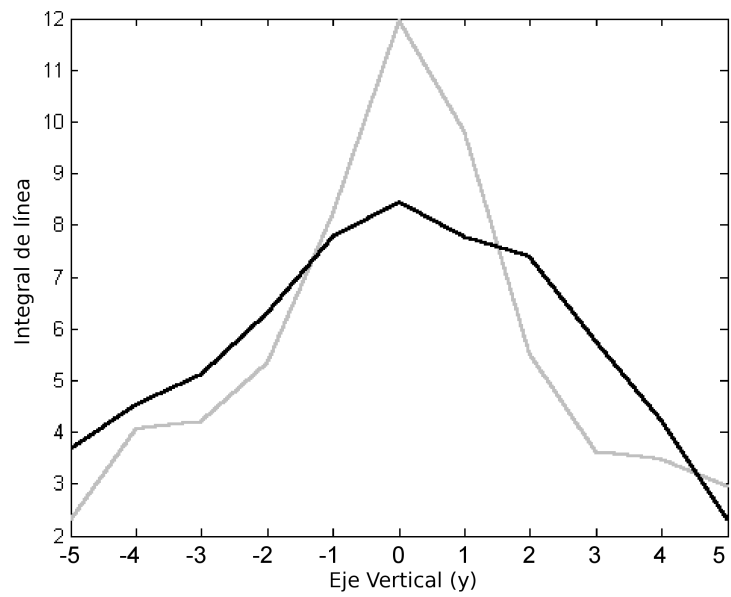
(c)

Figura 5.15: (a) Imagen Original. (b) Valor de la integral de línea calculado con los moldes PSO en una vecindad de 11x11 píxeles en torno al centro real del rostro. (c) Mismo cálculo, utilizando moldes Antropométricos.

la mencionada ventana. La curva graficada en 5.16(a) muestra el valor de la integral de línea a lo largo del eje x, cruzando el centro del rostro horizontalmente, mientras que en 5.16(b) se observa como varía a lo largo del eje y, también pasando por el centro del rostro. En ambas visualizaciones, (Figs. 5.15 y 5.16), se puede apreciar como los moldes PSO entregan un ajuste espacial más preciso, presentando un máximo más agudo en el valor de la integral de línea en torno al centro del rostro. Para el caso de los molde Antropométricos el máximo en el centro del rostro también está presente, pero es mucho menos pronunciado y en condiciones de mayor ruido, podría no ser fácil su detección.



(a)



(b)

Figura 5.16: Valor de la integral de línea a lo largo del (a) eje x y (b) eje y, cruzando el centro del rostro para el caso de los moldes PSO (gris) y Antropométricos (negro). El cálculo fue realizado dentro de una vecindad de 11×11 píxeles en torno al centro del rostro.

Tabla 5.7: Tasas de detección de rostros en secuencias de video

Secuencia de video.	No. de Cuadros	Tasa de detección de rostros (%)		
		PSO	Antropométricos	Adaboost _{VJ}
1	466	99	98	99
2	202	100	100	100
3	306	100	100	100
4	395	100	100	100
Total	1369	100	99	100

Tabla 5.8: Tasas de detección de rostros sobre Yale B

Método	Tasa de Detección (%)
Moldes PSO personalizados	82.65
Moldes PSO genéricos	81.92
Moldes Antropométricos	81.74
Adaboost _{VJ}	60.54

5.2.4. Tasas de Detección de Rostros

5.2.4.1. Secuencias de Video

La tabla 5.7 muestra el porcentaje de detección correcta de rostros para los moldes PSO y Antropométricos y el detector Adaboost_{VJ}. Se observa el número de cuadros en cada secuencia de video, acompañado del porcentaje de detección de cada secuencia.

5.2.4.2. Base de Datos Yale B

La tabla 5.8 muestra los resultados de detección de rostros para la base de datos Yale B, utilizando los tres métodos: moldes PSO, moldes Antropométricos y Adaboost_{VJ}. Se aprecia que la mayor tasa de detección fue obtenida utilizando los moldes PSO. También se observa que los moldes Antropométricos presentan mejores resultados que el método de Viola y Jones. La base de datos Yale B presenta cambios fuertes en las condiciones de iluminación e imágenes con fuerte contraste o mal iluminadas, lo que demuestra la mayor robustez de los métodos de moldes direccionales por sobre Adaboost en este tipo de condiciones. Algunos ejemplos de detecciones correctas en la galería Yale B se pueden observar en la fig. 5.17.

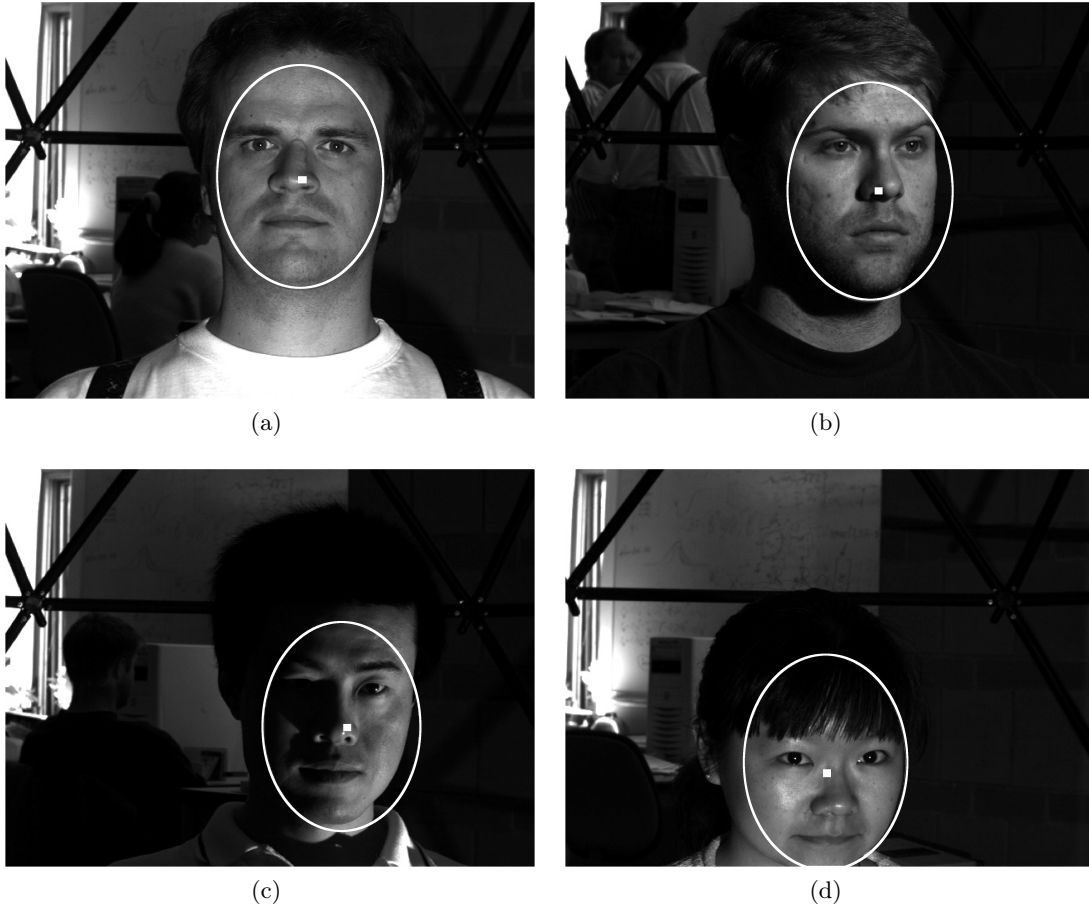


Figura 5.17: Ejemplos de detección correcta de rostros utilizando moldes PSO.

Tabla 5.9: Tasa de localización correcta de iris para los dos subconjuntos de la base de datos de PURDUE.

Método	Tasa de localización de iris versus umbral de error (%)							
	1	0.8	0.50	0.40	0.33	0.25	0.15	0.05
(Song et al. 2006) AR-63	96.8	96.8	96.8	96.8	96.8	96.8	N/A	N/A
(Song et al. 2006) AR-564	96.6	95.9	92.4	90.2	89.7	86.5	N/A	N/A
Moldes Antropométricos AR-63	100	100	100	100	100	100	100	74.1
Moldes Antropométricos AR-564	100	99.7	99.7	99.3	99.3	98.7	98.1	88.1
Moldes PSO genéricos AR-63	100	100	100	100	100	100	100	75.6
Moldes PSO genéricos AR564	100	100	100	99.7	99.5	99.3	98.8	87.3
Moldes PSO _{PURDUE} AR-63	100	100	100	100	100	100	100	77.8
Moldes PSO _{PURDUE} AR-564	100	100	100	99.8	99.7	99.7	99.1	88.6

5.2.5. Detección de Iris

La tabla 5.9 muestra los resultados de la localización de iris para los conjuntos de moldes Antropométricos, moldes PSO genéricos y los moldes PSO ajustados para la base de datos de PURDUE (PSO_{PURDUE}), en los subconjuntos AR-63 y AR-564. Los resultados propuestos por Song et al. en [25] también se incluyen para efectos de comparación. Se observa como los moldes Antropométricos tienen un desempeño superior al método propuesto por Song et al, superándolo en aproximadamente un 10% para el criterio más exigente reportado en [25]. Para el caso de los moldes PSO, tanto genéricos como ajustados, el error en la localización es aún menor, llegando a superar levemente a los moldes antropométricos.

Capítulo 6

Conclusiones

Durante la investigación realizada en este trabajo de tesis se han desarrollado dos nuevos métodos de localización de rostros en imágenes digitales: Transformada LET y Moldes PSO, útiles en aplicaciones de interfaces hombre-máquina y con desempeño en muchos casos mejor que otros métodos ampliamente utilizados a nivel internacional, como el detector de Viola y Jones [27] y los moldes antropométricos [15, 35].

Transformada LET: Una de las principales contribuciones de esta tesis es el desarrollo de un nuevo método de localización de rostros para aplicaciones hombre-máquina que presenta interesantes resultados a nivel de tasas de detección, robustez y selectividad de los parámetros del rostro. Este método, basado en una nueva transformada denominada LET (Linear Elliptic Transform) [26], convierte el contorno semi elíptico inferior del rostro en una línea vertical en el espacio de salida. La detección de esta recta vertical, mediante un histograma proyectado, permite conocer las características del rostro ya que vienen determinadas por los parámetros aplicados en la transformada. Este método permite obtener información acerca de la localización, tamaño y rotación del rostro, no requiere de entrenamiento previo (sólo determinar los rangos de aplicación) y es robusto a variaciones en las condiciones de iluminación, color de piel, ruido y fondos no homogéneos.

La tasa de detección de rostros está por sobre 95 % en base de datos de imágenes convencionales como son las de CALTECH, FERET y PURDUE, teniendo un desempeño superior al 98 % en esta última. Estos resultados están a unos pocos puntos porcentuales (1 % a 2 % aprox.) por debajo de las tasas obtenidas al aplicar el detector de Viola y Jones [27], ampliamente utilizado en varias aplicaciones de detección de rostros. Sin embargo, y a diferencia de LET, el detector de Viola y Jones no entrega información acerca de la rotación coronal y excentricidad del rostro. En la base de datos de YALE B, con condiciones extremas de contraste

e iluminación, el desempeño de LET es de 79,6 % en las tasas de detección, muy superior a la obtenida al aplicar Viola y Jones (61,5 %).

La estimación de los parámetros del rostro mediante LET entrega muy buenos resultados, obteniéndose una elipse que se ajusta de forma casi perfecta en la mayoría de los casos. Se observa que la posición horizontal x_0 , excentricidad ϵ y tamaño del rostro r presentan el mejor desempeño, siendo altamente precisos. La estimación de la posición vertical y_0 y la rotación coronal φ , son la que presentan mayor error, principalmente debido a que la detección gruesa tiene menor precisión al determinar la posición vertical. En el caso de la rotación, el paso utilizado es más grueso, y se observa que el contorno inferior del rostro se ajusta de buena manera a varios valores de rotación, lo que induce a error.

Otra ventaja de la metodología LET es la posibilidad de aplicarla en forma parcial, en conjunto con otros métodos de detección de rostros, de manera de determinar sólo uno de los parámetros que entrega. Por ejemplo, puede ser utilizada para determinar la excentricidad y tamaño del rostro si se conoce a priori la posición.

El método propuesto, basado en LET, está diseñado para ser utilizado en aplicaciones de interfaz hombre-máquina, y asume que en la imagen de entrada se encuentra sólo un rostro como forma principal. Sin embargo, la extensión a múltiples rostros puede ser llevada a cabo, si se reemplaza la etapa de detección gruesa por una que entregue la posición aproximada de múltiples rostros. Los tiempos de procesamiento de este método no permiten su aplicación directa en aplicaciones de tiempo real, ya que la búsqueda de los parámetros es muy lenta. Sin embargo, si se aplica alguna forma de predicción de parámetros cuadro a cuadro y se restringe la búsqueda a una vecindad de valores cercanas a los parámetros conocidos en el cuadro anterior, se puede obtener un desempeño aceptable con tasas de procesamiento cercanas a los 10 cuadros por segundo.

La mayoría de los errores cometidos por el algoritmo LET se deben a bordes del rostro demasiado débiles u ocluidos que hacen imprecisa la aplicación de la transformada. Varios de los problemas de localización se deben también a una incorrecta detección del rostro en la etapa de detección gruesa, que la aplicación de la LET no logra corregir. Otra fuente de error detectada son los rostros no elípticos en que el sujeto presenta, por ejemplo, un mentón muy agudo o prominente y en el que la hipótesis del rostro elíptico no puede ser aplicada.

La mayor deficiencia de LET se debe a la dificultad de detectar la línea vertical en la imagen de salida, ya que se debe iterar sobre sucesivas combinaciones de parámetros, lo que entrega distintas imágenes LET. Se propone como trabajo futuro, estudiar métodos que permitan recorrer el hiperespacio de la imagen paramétrica LET, de manera de seleccionar los valores de la elipse de manera más rápida y precisa. Además, es recomendable el estudio de otros

métodos de detección de la línea vertical, ya que la imagen direccional, si bien es robusta, provoca una pérdida de información en cuanto a la selectividad de la transformada. Esto ya que líneas verticales con pequeñas ondulaciones tienden a ser todas “mapeadas” a una línea vertical en la imagen direccional. Por último, y en combinación con lo anterior, se propone estudiar la aplicación de otros métodos de detección gruesa que sean más precisos en la tarea de determinar un punto inicial para la aplicación de las LET.

Moldes PSO: En los trabajos previos basados en moldes antropométricos [15,35] se utilizan características faciales comúnmente reconocibles por el sistema visual humano, como son ojos, cejas, nariz, boca y mentón. En esta tesis, basada en el trabajo previo de Juan Vallejos [52], se estudia en mayor profundidad el método de generación de moldes direccionales, optimizados mediante el algoritmo de PSO (Particle Swarm Optimization). Se estudió el desempeño del método en la detección de rostros sobre 4 secuencias de video y sobre la base de datos de Yale B. Asimismo, se midió el desempeño de los moldes PSO, al integrarlos en una aplicación de detección de iris, sobre dos conjuntos de la base de datos de Purdue.

Las diferentes pruebas realizadas concluyen que los moldes PSO tienen un mejor comportamiento en la detección, ajuste fino y tamaño del rostro, que los moldes antropométricos estudiados en trabajos previos. Esto debido a que los templates PSO maximizan la respuesta de la integral de línea, cuando son aplicados sobre la imagen direccional del rostro de entrada. Los resultados muestran un mayor valor de la integral de línea por sobre los moldes antropométricos, y una mejor estimación del tamaño del rostro, esto debido a que capturan características invariantes del conjunto de rostros de entrada, las que no necesariamente corresponden a características “clásicas” como los bordes presentes en ojos, nariz y boca.

La optimización PSO permite adecuar los moldes a un individuo en particular (molde PSO personalizado) o a un conjunto de individuos que pueden o no presentar características comunes (moldes PSO genéricos). Esto se realiza simplemente cambiando el conjunto de rostros de entrada a optimizar. Los resultados muestran que mientras más personalizada sea la optimización, el conjunto de moldes generado presenta una mejor respuesta al individuo.

Otra ventaja del método es que permite controlar mediante el fitness el número de puntos total del conjunto de moldes direccionales, disminuyendo el tiempo computacional requerido y permitiendo procesamiento en tiempo real de aproximadamente 30 cuadros por segundo en un computador estándar.

El método fue probado sobre un subconjunto de la base de datos Yale B y comparado con el método de detección propuesto por Viola y Jones [27]. Los resultados obtenidos muestran una mayor robustez de la metodología PSO con un 82,7% de detección correcta de rostros. Este

valor se encuentra levemente por sobre los registrados con moldes antropométricos (81,74 %) y LET (79,6 %); y muy por encima de los registrados con el detector de Viola y Jones (61 % aprox). Sin embargo, el número de puntos total de los moldes PSO es un 25 % a un 40 % menor que los moldes antropométricos, lo que significa que son mucho más eficientes para un desempeño ligeramente mejor. También se probó la eficacia del método al ser usado en conjunto con un detector de iris para estimar la posición de los ojos en imágenes estáticas de la base de datos de Purdue, obteniéndose tasas de detección de iris por sobre el 98 % para criterios de error muy exigentes. El método fue comparado con el publicado por Song et. al. [25], mostrando una clara superioridad en la detección del iris. Nuevamente, el uso de los moldes PSO es levemente mejor que si se utilizan los moldes antropométricos originales, debido a una mayor precisión en la determinación de la posición del rostro y por tanto, de la zona de búsqueda de los ojos.

La metodología de PSO presenta la desventaja de requerir de un pequeño subconjunto de imágenes frontales (en esta tesis se utilizaban entre 4 y 8), para poder realizar la optimización, la que es costosa computacionalmente. Sin embargo, este método podría ser utilizado en conjunto con los moldes direccionales antropométricos, aplicando los moldes PSO sólo para los individuos registrados previamente. Los moldes PSO estudiados sólo son aplicables a rostros frontales sin rotación coronal, por lo que se propone como trabajo futuro, el estudiar la extensión de estos moldes a otros ángulos. La rotación coronal puede ser obtenida rotando de manera inteligente los moldes PSO frontales, de manera de mantener en forma consistente la información direccional. Las rotaciones de moldes direccionales en otros ejes, abordada en trabajos como [33], permiten la aplicación de esta metodología en una mayor variedad de casos. Sin embargo, la optimización PSO para moldes con rotación sagital y transversal requeriría de un gran número de ejemplos a distintos ángulos de rotación.

En general, ambos métodos estudiados, Transformada LET y Moldes PSO, presentan un muy buen desempeño al ser utilizados en aplicaciones de interfaz hombre-máquina, en que se asume la presencia de un solo rostro en la imagen. Ambos métodos son simples de implementar y no requieren equipamiento complejo o invasivo. Los moldes direccionales, los cuales pueden ser optimizados con PSO para mejorar su desempeño, presentan la ventaja de ser aplicables en tiempo real, a diferencia de LET, que requiere un mayor tiempo computacional. Sin embargo, LET es más versátil, ya que determina parámetros adicionales como rotación, excentricidad y tamaño de manera precisa y puede ser utilizado en forma parcial para determinar sólo uno de estos parámetros. Además, la metodología LET, no requiere de ningún entrenamiento ni procesamiento previo, mientras que los moldes PSO deben ser calculados con anterioridad.

Ambos métodos, LET y moldes PSO, al estar basados en imágenes direccionales, presentan gran robustez a condiciones de iluminación extremas, ya que la información direccional tiende

a ser independiente de la iluminación. Ambos, asimismo, dependen en gran manera de la etapa de detección gruesa (igual para ambos métodos) que entrega la posición aproximada del rostro. Se propone como trabajo futuro el estudio de ambos métodos en conjunto con otros detectores como el de Viola y Jones de manera de complementar la capacidad de detección de múltiples rostros de este último, con la precisión entregada por los métodos propuestos. Es de interés también, el trasladar la optimización PSO estudiada en esta tesis a trabajos más recientes y avanzados en moldes direccionales como los propuestos en [33], que integran rotaciones en varios ejes del rostro. Esto permitiría una utilización más eficiente de estos moldes en aplicaciones de tiempo real, por ejemplo como una etapa de estimación de pose dentro de un sistema de reconocimiento de rostros.

Bibliografía

- [1] Z. Zhu and Q. Ji, “Eye and gaze tracking for interactive graphic display,” *Machine Vision and Application*, vol. 15, pp. 139–148, July 2004.
- [2] C. Perez, C. Peña, C. Holzmann, and C. Held, “Design of a virtual keyboard based on iris tracking,” in *Proceedings of the second Conference of the IEEE/EMBS and BMES*, Houston, USA, pp. 2428–2429, October 2002.
- [3] Q. Ji, P. Lan, and Z. Zhu, “Real time non-intrusive monitoring and prediction of driver fatigue,” *IEEE Trans. Veh. Technology*, vol. 53, pp. 1052–1068, July 2004.
- [4] M. Valstar and M. Pantic, “Biologically vs. logic inspired encoding of facial actions and emotions in video,” in *IEEE International Conference on Multimedia and Expo*, Toronto, Ontario, Canada, pp. 325–328, July 2006.
- [5] K. S. Park and C. J. Lim, “A simple vision-based head tracking method for eye-controlled human/computer interface,” *Int. J. Hum.-Comput. Stud.*, vol. 54, no. 3, pp. 319–332, 2001.
- [6] International Biometric Group, “Biometrics market and industry report 2007-2012,” Jan. 2007.
- [7] B. Christie, “Biometrics boom,” *IEEE Spectrum*, vol. 41, p. 13, March 2004.
- [8] M. H. Yang, D. J. Kriegman, and N. Ahuja, “Detecting faces in images: a survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 34–58, Jan. 2002.
- [9] C. Chiang, W. Tai, M. Yang, Y. Huang, and C. Huang, “A novel method for detecting lips, eyes and faces in real time,” *Real-Time Imaging*, vol. 9, no. 4, pp. 277–287, 2003.
- [10] C. Kotropoulos and I. Pitas, “Rule-based face detection in frontal views,” in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. IV, Munich, Germany, pp. 2537–2540, Apr. 1997.

- [11] T. Leung, M. Burl, and P. Perona, "Finding faces in cluttered scenes using random labeled graph matching," in *Proceedings of the Fifth IEEE International Conference on Computer Vision (ICCV)*, Los Alamitos, CA, USA, pp. 637–644, IEEE Computer Society, 1995.
- [12] K. Yow and R. Cipolla, "Feature-based human face detection," *Image and Vision Computing*, vol. 15, no. 9, pp. 713–735, 1996.
- [13] Y. Dai and Y. Nakano, "Face-texture model-based on SGLD and its application in face detection in a color scene," *Pattern Recognition*, vol. 29, pp. 1007–1017, June 1996.
- [14] M. J. Jones and J. M. Rehg, "Statistical color models with application to skin detection," *International Journal of Computer Vision*, vol. 46, no. 1, pp. 81–96, 2002.
- [15] C. Perez, V. Lazcano, and P. Estevez, "Real-time iris detection on coronal axis rotated faces," *IEEE Transactions on Systems, Man and Cybernetics Part C*, vol. 37, pp. 971–978, Sept. 2007.
- [16] I. Craw, H. Elis, and J. R. Lishman, "Automatic extraction of face-features," *Pattern Recogn. Lett.*, vol. 5, no. 2, pp. 183–187, 1987.
- [17] A. L. Yuille, P. W. Hallinan, and D. S. Cohen, "Feature extraction from faces using deformable templates," *International Journal of Computer Vision*, vol. 8, no. 2, pp. 99–111, 1992.
- [18] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 23–38, 1998.
- [19] R. Flores, "Detección de rostros mediante redes neuronales min-max difusas," Tesis de Magister, Departamento de Ingeniería Eléctrica, Universidad de Chile, 2007.
- [20] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, Kauai, Hawaii, USA, pp. 511–518, Dec. 2001.
- [21] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.
- [22] A. N. Rajagopalan, K. S. Kumar, J. Karlekar, R. Manivasakan, M. M. Patil, U. B. Desai, P. G. Poonacha, and S. Chaudhuri, "Finding faces in photographs," in *ICCV '98: Proceedings of the Sixth International Conference on Computer Vision*, Washington, DC, USA, p. 640, IEEE Computer Society, 1998.

- [23] E. Osuna, R. Freund, and F. Girosi, "Training support vector machines: an application to face detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997*, San Juan, Puerto Rico, pp. 130–136, June 1997.
- [24] C. A. Perez and V. A. Lazcano, "Real-time iris detection on faces with coronal and transversal axis rotation," *Optomechatronic Machine Vision*, vol. 6051, p. 60510C, Dec. 2005.
- [25] J. Song, Z. Chi, and J. Liu, "A robust eye detection method using combined binary edge and intensity information," *Pattern Recognition*, vol. 39, pp. 1110–1125, June 2006.
- [26] P. Garate, "Diseño de moldes direccionales para la deteccion de caras y movimientos oculares en rostros rotados," Tesis de Magister, Departamento de Ingeniería Eléctrica, Universidad de Chile, 2005.
- [27] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vision*, vol. 57, no. 2, pp. 137–154, 2004.
- [28] A. Palma, "Sistema de detección y seguimiento de movimientos oculares en imágenes digitales," Tesis de Magister, Departamento de Ingeniería Eléctrica, Universidad de Chile, 2003.
- [29] C. Perez, A. Palma, C. Holzmann, and C. Pena, "Face and eye tracking algorithm based on digital image processing," in *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 2, Tucson, AZ, USA, pp. 1178–1183, Oct. 2001.
- [30] C. A. Perez, V. Lazcano, P. Estevez, and C. Held, "Real time iris detection on rotated faces," in *Proceedings of the SPIE, Optomechatronics Systems IV*, vol. 5264, Boston, MA, pp. 42–53, Oct. 2001.
- [31] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *European Conference on Computational Learning Theory*, Barcelona, España, pp. 23–37, 1995.
- [32] D. Maio and D. Maltoni, "Real time face location on gray scale static images," *Pattern Recognition*, vol. 33, pp. 1525–1539, Sept. 2000.
- [33] V. Lazcano, "Seguimiento 3d de rostros e iris mediante moldes antropométricos," Tesis de Magister, Departamento de Ingeniería Eléctrica, Universidad de Chile, 2006.
- [34] A. Martinez and R. Benavente, "The AR face database," Tech. Rep. 24, CVC Tech. Report, June 1998.

- [35] C. Perez, V. Lazcano, P. Estevez, and C. Held, "Real-time iris detection on faces with coronal axis rotation," in *IEEE International Conference on Systems, Man and Cybernetics*, vol. 7, The Hague, Netherlands, pp. 6389–6394, Oct. 2004.
- [36] M. Weber, "The CALTECH face database." Online, 1995. <http://www.vision.caltech.edu/html-files/archive.html>.
- [37] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090–1104, 2000.
- [38] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 643–660, 2001.
- [39] R. Verschae and J. R. del Solar, "A hybrid face detector based on an asymmetrical adaboost cascade detector and a wavelet-bayesian-detector," *Lecture Notes in Computer Science*, vol. 2686, pp. 742–749, 2003.
- [40] R. Verschae, J. R. del Solar, and M. Correa, "A unified learning framework for object detection and classification using nested cascades of boosted classifiers," *Machine Vision and Applications*, vol. 19, pp. 85–103, Jan. 2008.
- [41] E. Schwartz, "Computational anatomy and functional architecture of striate cortex: A spatial mapping approach to perceptual coding," *Vision Research*, vol. 20, pp. 645–669, 1980.
- [42] C. A. Perez and C. M. Aravena, "Log-linear elliptic transform for frontal-face parameter estimation," in *IEEE International Conference on Systems, Man and Cybernetics*, Montreal, Canada, pp. 1130–1134, Oct. 2007.
- [43] C. Perez and J. Vallejos, "Face detection using PSO template selection," *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, 2006*, vol. 5, pp. 4220–4224, 8-11 Oct. 2006.
- [44] R. Eberhart and J. Kennedy, "Particle swarm optimization," in *Proceedings of IEEE International Conference on Neural Networks*, vol. 4, Perth, Australia, pp. 1942–1948, 1995.
- [45] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.

- [46] A. Chatterjee and P. Siarry, “Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization,” *Computers and Operations Research*, vol. 33, pp. 859–871, 2006.
- [47] R. Eberhart and Y. Shi, “Parameter selection in particle swarm optimization,” in *Evolutionary Programming VII: Proceedings of the Seventh Annual Conference on Evolutionary Programming*, San Diego, California, pp. 591–600, Mar. 1998.
- [48] Y. Shi and R. Eberhart, “A modified particle swarm optimizer,” in *Proceedings of the IEEE International Conference on Evolutionary Computation*, pp. 69–73, 1998.
- [49] R. Eberhart and Y. Shi, “Empirical study of particle swarm optimization,” in *Congress of Evolutionary Computation*, Washington, DC, USA, pp. 1945–1950, July 1999.
- [50] I. Trelea, “The particle swarm optimization algorithm: convergence analysis and parameter selection,” *Information Processing Letters*, vol. 85, pp. 317–325, 2003.
- [51] L. P. Zhang, Huan-Jun, and Shang-Xu, “Optimal choice of parameters for particle swarm optimization,” *Journal of Zhejiang University SCIENCE*, vol. 6A, pp. 528–534, 2005.
- [52] J. Vallejos, “Optimización de moldes por enjambre de partículas y maximización de componentes para detección de rostros e iris,” Tesis de Magister, Departamento de Ingeniería Eléctrica, Universidad de Chile, 2006.
- [53] V. Popovici, J.-P. Thiran, Y. Rodriguez, and S. Marcel, “On performance evaluation of face detection and localization algorithms,” in *Proceedings of the 17th International Conference on Pattern Recognition*, vol. 1, Cambridge, UK, pp. 313–317, Aug. 2004.

Anexos

A. Imagen Direccional

Una imagen puede ser vista como una superficie $h(x, y)$ sobre el plano x-y, donde los máximos locales corresponden a las áreas iluminadas y los valles a las áreas oscuras. Las propiedades topológicas de esta superficie, en particular las curvas de nivel, contienen información que puede ser usada en el procesamiento adaptivo de la imagen con el objeto de realizar reconocimiento de patrones. Las tangentes a las curvas de nivel, se orientan en forma paralela a los bordes de la imagen, por lo que la dirección de estas tangentes pueden ser utilizadas para la detección de dichos bordes.

Conceptualmente, el método consiste en estimar los vectores que representan a las tangentes de los bordes dentro de la imagen. Para esto, en cada punto dentro de una ventana se calcula un vector n que es normal a la superficie $z = h(x, y)$. El vector tangente a la curva de nivel en este punto (si es que está definido), yace en el plano x-y y es perpendicular a la normal n . El vector tangente promedio para una cierta ventana será el vector unitario en el plano x-y que resulte más cercano de ser perpendicular a todas las normales dentro de la ventana. Existen ventajas al utilizar los vectores normales para calcular la tangente promedio, en vez de promediar las tangentes en forma directa. En primer lugar, un vector normal siempre está bien definido para una superficie dada y en segundo lugar nos entregan información acerca de lo pronunciado del borde. La fig. 1 ilustra estos conceptos. Lejos del eje x , las normales a la superficie $h(x, y)$ son paralelos al eje z por lo que la tangente no está definida (no existe un borde marcado para determinar su dirección). Al moverse cerca del eje x , los vectores normales se inclinan alejándose del eje z . Si el borde es muy pronunciado, en el eje x , las normales serán casi perpendiculares al eje z . En particular, mientras más pronunciado el borde, menor será el ángulo entre la superficie que contiene a las normales y el plano x-y. Es claro, que al calcular el vector tangente a todas las normales cercanas al borde, el vector resultante es el mismo eje x .

Para el caso de una imagen digital, la función $h(x, y)$ se encuentra discretizada espacialmente, por lo que el cálculo de las normales para cada punto deben ser realizadas sobre un

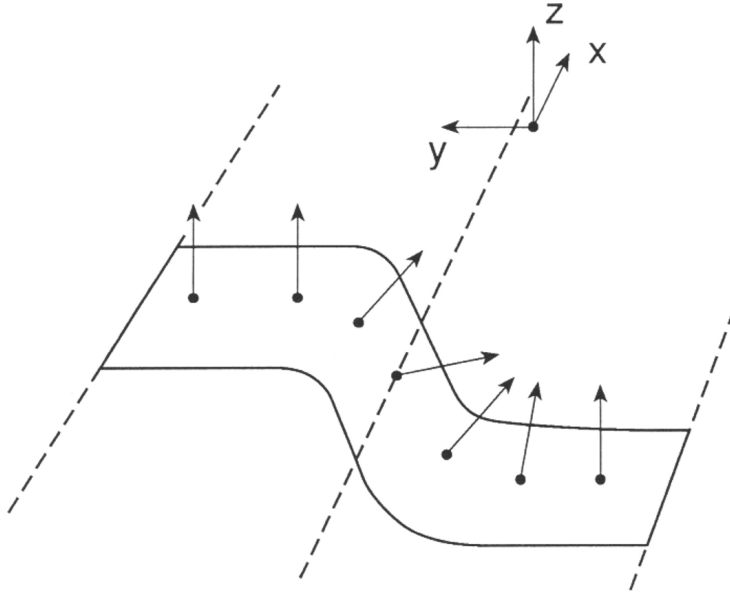


Figura 1: Representación gráfica de un borde dentro de la imagen $z = h(x, y)$. Se han dibujado las normales a la superficie. Se aprecia como al acercarse al eje x , las normales dejan de ser paralelas a z .

plano aproximado $\Pi_{\text{local}}(x, y)$ que represente la región delimitada por los 4 píxeles diagonales en torno al elemento (x, y) (fig 2).

Para calcular la tangente dentro de una ventana, la imagen de entrada es dividida en vecindades cuadradas de $W \times W$ píxeles, cuyo tamaño dependerá de la aplicación (tamaños mas grandes son menos precisos pero mas inmunes al ruido). Sobre cada punto (x, y) dentro de esta vecindad, se calcula el vector normal al plano aproximado $\Pi_{\text{local}}(x, y)$. En base a estos vectores es calculado el vector tangente a la vecindad como aquél que minimiza su producto punto con cada uno de los vectores normales. El conjunto de estos vectores tangentes, uno por vecindad, da origen a la imagen direccional I_{dir} , de tamaño W^2 veces menor que la imagen original. En el trabajo original de Maio y Maltoni [32], los vectores tangentes contienen tanto información angular como de módulo, éste último representando la intensidad del borde. Para todos los trabajos expuestos aquí, el módulo se considera constante, dando igual peso a los bordes débiles y fuertes.

La fig 3.a muestra la imagen original de entrada y la 3.b su correspondiente imagen direccional. Cabe destacar que la imagen direccional así expresada es una simple visualización de la imagen direccional real, en que por cada vecindad se dibuja un vector tangente, orientado según el ángulo que posea en la imagen direccional real. Se puede apreciar como los vectores tangentes siguen los bordes principales del rostro.

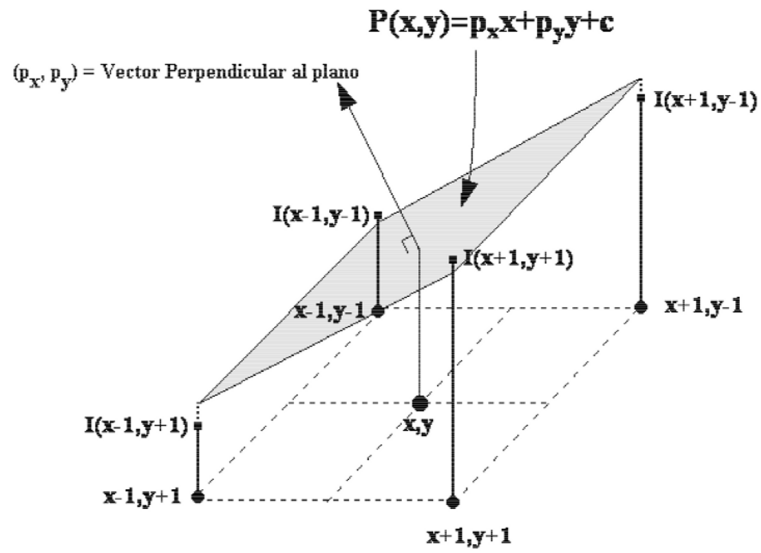


Figura 2: Plano aproximado al pixel ubicado en (x, y) . Se utilizan los 4 pixeles diagonales al pixel central para la estimación de la normal al plano.

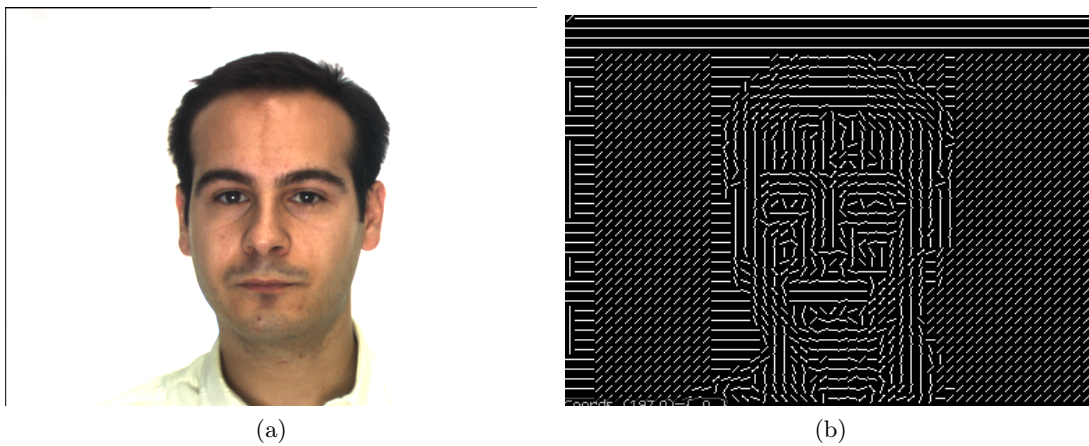


Figura 3: (a) Imagen original. (b) Visualización de la imagen direccional asociada.

El desarrollo matemático del cálculo de los vectores normales y de la minimización del producto punto entre el vector tangente y los vectores normales de la vecindad, se explican en detalle en el trabajo de Tesis de Alvaro Palma [28]

B. Método de Detección Gruesa (Maio y Maltoni)

La detección gruesa se realiza mediante el método propuesto por Maio y Maltoni [32] que utiliza el hecho de que los rostros frontales pueden ser representados mediante una forma elipsoidal. El diagrama de bloques en la figura 4 especifica los pasos necesarios para la detección gruesa de la posición del rostro.

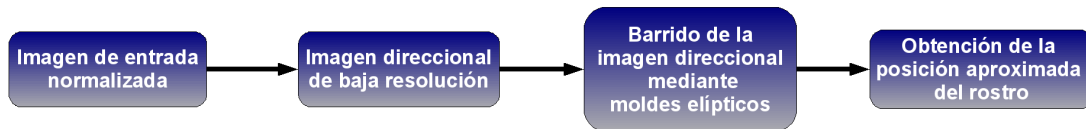


Figura 4: Diagrama de bloques para la detección gruesa del centro del rostro.

El proceso consiste en obtener una imagen direccional a partir de la imagen preprocesada, que contenga información acerca de las tangentes a los bordes presentes en la imagen. Esta imagen direccional de baja resolución I_{LDR} , se pasa del espacio de la imagen al espacio paramétrico dado por la Transformada de Hough, mediante el incremento de un acumulador. Esta Transformada, al igual que LET, requiere como parámetros de aplicación el ángulo de inclinación de la tangente φ , la excentricidad ϵ y el tamaño de elipses a detectar.

El incremento del acumulador se realiza mediante el barrido de moldes o plantillas elípticas T_φ sobre la imagen direccional que representan las posibles posiciones del centro de una elipse que es tangente al vector direccional de ángulo φ . La figura 5 muestra uno de estos moldes elípticos donde, en rojo, se aprecia la tangente asociada a una vecindad centrada en (x_0, y_0) e inclinada φ grados con respecto a la horizontal. Las elipses de excentricidad ϵ , y semejeje vertical situado en el rango $[b_{min}, b_{max}]$ que sean tangentes al vector dado por φ tendrán sus centros ubicados a lo largo del vector dado por el ángulo θ (línea magenta) y estarán limitados a los segmentos dados por el tamaño mínimo y máximo que determinan la excentricidad y los valores de b_{min} y b_{max} (segmentos en amarillo) El ángulo θ se calcula mediante la ecuación 1:

$$\theta = \arctan\left(\frac{-1}{\epsilon \cdot \tan(\varphi)}\right) \quad (1)$$

Como la imagen direccional calculada aproxima los ángulos de las tangentes, es necesario

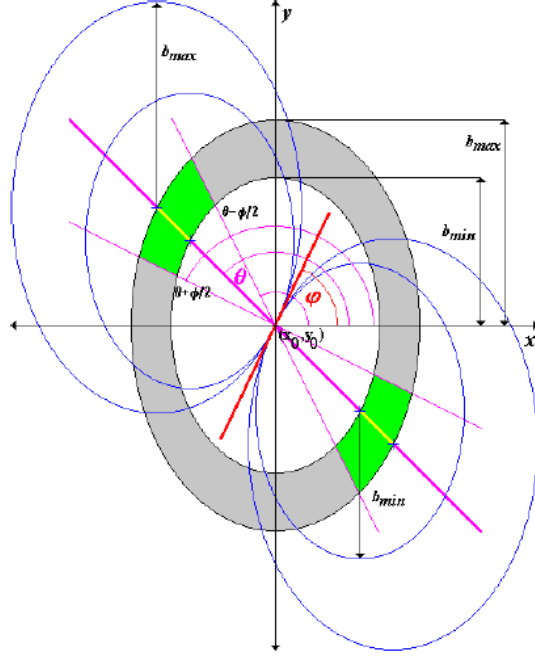


Figura 5: Molde elíptico de posibles posiciones del centro del rostro (en verde) con respecto a una tangente inclinada en φ grados.

introducir un término de tolerancia a la determinación de las posibles ubicaciones del centro de la elipse, término dado por el ángulo ϕ . Este término, convierte el lugar geométrico de los posibles centros de elipse, del segmento de recta antes mencionado, a una sección angular elíptica comprendida entre los ángulos $[\theta - \frac{\phi}{2}, \theta + \frac{\phi}{2}]$ (zona verde en la Figura 5)

Con esto, el lugar geométrico de los moldes T_φ queda entonces definido como:

$$T_\varphi = \left\{ (x, y) \in I_{LDR} \left| r_{min}^2 \leq \left(\frac{x - x_0}{e} \right)^2 + (y - y_0)^2 \leq r_{max}^2 \quad y \quad \left| \arctan \left(\frac{y - y_0}{x - x_0} \right) - \theta \right| \leq \frac{\phi}{2} \right. \right\} \quad (2)$$

Los puntos de este molde, situados en ángulos más cercanos a la recta dada por θ constituyen, como se ha discutido, una ubicación más probable del centro de la elipse que los puntos situados más cerca del borde de la sección angular dada por ϕ . Para integrar este hecho en las ecuaciones, se introduce una función de peso asociada a cada punto perteneciente al lugar geométrico, la cual decaerá linealmente desde el centro de la zona hacia sus orillas:

$$Pes_{OT}(x, y) = 1 - \frac{2 \cdot |\arctan((y - y_0)/(x - x_0)) - \theta|}{\phi} \quad (3)$$

La figura 6 muestra un ejemplo de estos moldes para un ángulo de inclinación φ de 25° y considerando una excentricidad fija de 0.75.

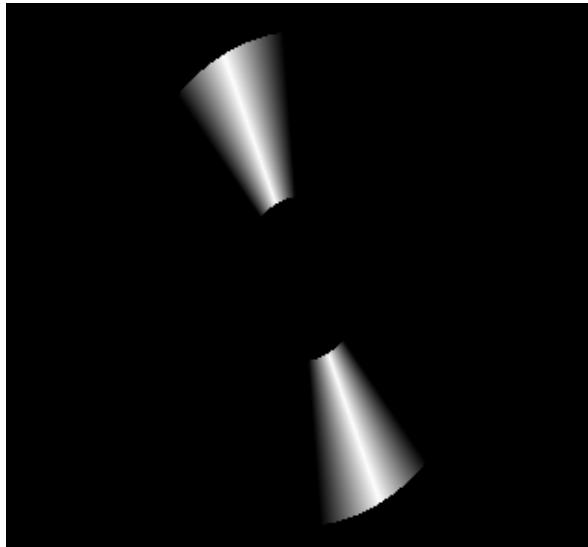


Figura 6: Molde de posibles centros de elipses considerando una excentricidad de 0.75 y rotación de 25° .

Estos moldes elípticos son creados una única vez, con una excentricidad fija al inicio del procesamiento y no sufren transformaciones durante el transcurso de éste. La detección gruesa de la posición del rostro es realizada mediante el barrido de I_{LDR} por este conjunto de moldes. El proceso de barrido se inicia con la creación de un acumulador de dimensiones idénticas a I_{LDR} y con todos sus valores inicialmente nulos. Este acumulador es dividido en vecindades cuadradas del mismo tamaño que las utilizadas en la creación de la imagen direccional, de 7 píxeles por lado. Sobre cada punto (i, j) del acumulador que representa el centro de una vecindad es posicionado el centro del molde T_φ asociado al ángulo tangente presente en ese punto de la imagen direccional. Una vez que se ha ubicado el molde en la posición señalada, se procede a incrementar el valor de cada punto del acumulador en la cantidad dada por el valor del píxel perteneciente al molde elíptico que se encuentre sobre él. El proceso de ubicación de los moldes se repite para todos los puntos del acumulador cuyas coordenadas representan centros de vecindades. Este algoritmo puede resumirse en el siguiente pseudo código:

```

Inicialización de A;
∀ vector d ∈ ILDR{
    [x0, y0] = posición(d);
    φ = dirección(d);
    σ = módulo(d);
    T = plantilla(x0, y0, φ);
    ∀ pixel[x, y] ∈ T{
        A[x, y] = A[x, y] + σ · PesoT[x, y];
    }
}

```

Una vez recorrida la imagen completa con los moldes, se puede determinar la posición de la(s) elipse(s) que se encuentren presentes en la imagen, simplemente observando los puntos en el acumulador que presenten un valor mayor a un cierto umbral. En definitiva, los máximos locales en el acumulador, que cumplan la condición señalada, corresponderán a centros de elipses.

En la imagen 7 se muestra la detección de una elipse mediante el método descrito. En la Figura 7 (b) se observa en tonalidades de grises los pixeles del acumulador, correspondiendo a las tonalidades más claras los pixeles con mayor valor acumulado. En la Figura 3.3 se observa este método aplicado a la detección de un rostro. En este caso, se considera dentro de las hipótesis la presencia de un sólo rostro en la imagen, por lo que las coordenadas de la detección gruesa (x_C, y_C) vienen dadas por la ubicación del máximo en el acumulador.

Con la obtención de estas coordenadas se pasa a la etapa de refinamiento (o detección fina) en que se estima en forma precisa los parámetros de la elipse.

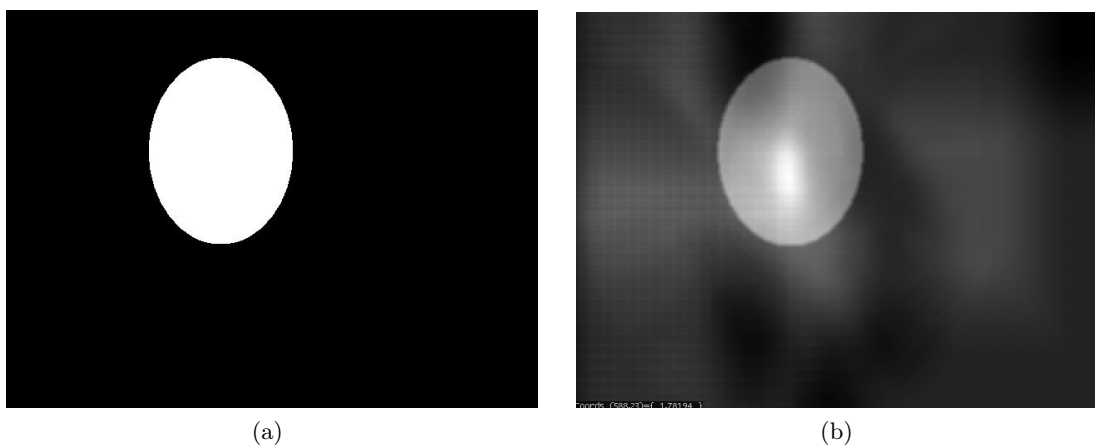


Figura 7: Detección de elipses (a) Imagen Original. (b) Imagen con acumulador superpuesto.

C. Detección de Iris

La zona de detección ocular, se calcula a partir de la posición y rotación del rostro y de sus características antropométricas, obtenidas en la etapa de detección fina. Para la determinación de esta zona, existen diversos métodos, siendo el utilizado hasta ahora el propuesto en [53] que da las coordenadas probables del ojo en base las características de la caja que representa al rostro. Sobre estas coordenadas, se estima una región de radio R en que se busca al ojo. En [26] se propone calcular la zona de búsqueda de manera distinta, utilizando información estadística de rostros de distintos sujetos de la base de datos de Purdue.

La zona de búsqueda de los ojos viene dada por las siguientes ecuaciones referidas a la Figura 8 :

$$\begin{aligned}h &= 2 \cdot r \\w &= h \cdot \epsilon \\y_{upper} &= h \cdot (tr_{gn} - en_{gn})/tr_{gn} \\s &= zy/w \\x_{ee} &= 2 \cdot pupil_{se}/s\end{aligned}$$

donde h y w son el alto y ancho del rostro, r es el semieje mayor calculado en la etapa de detección fina y las constantes antropométricas vienen dadas por :

$$\begin{aligned}pupil_{se} &= 33,4 \\en_{gn} &= 117,7 \\tr_{gn} &= 187,2 \\zy &= 139,1\end{aligned}$$

Luego, las coordenadas de los ojos, relativas al rostro no rotado son:

$$\begin{aligned}(x_1, y_1) &= (x_{ee}/2, y_{upper} - r) \\(x_2, y_2) &= (-x_{ee}/2, y_{upper} - r)\end{aligned}$$

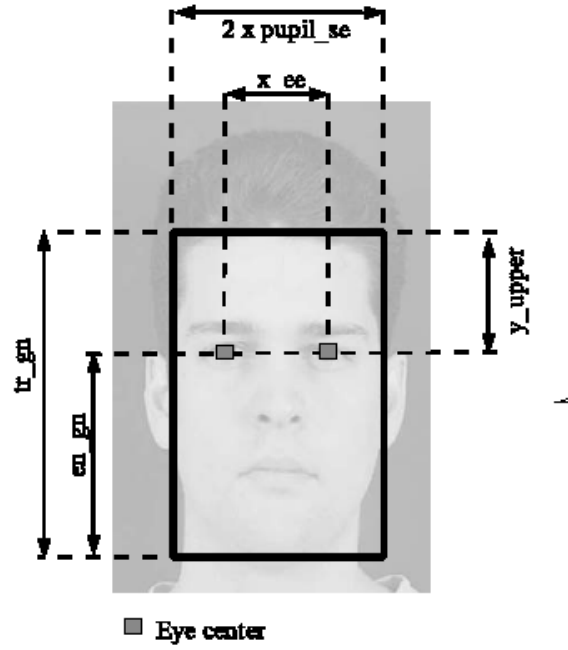


Figura 8: Modelo del rostro.

Agregando ahora, la rotación y traslación del rostro, las coordenadas finales son:

$$\begin{bmatrix} x_{ojos_i} \\ y_{ojos_i} \end{bmatrix} = \begin{bmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} x_f \\ y_f \end{bmatrix}$$

donde el índice i indica el ojo 1 ó 2 , (x_f, y_f) es la posición refinada del rostro y φ es el ángulo de rotación del rostro.

Una vez segmentada la zona de búsqueda de los ojos I_{ojo} se genera una imagen con los posibles centros de iris, reduciendo las reflexiones mediante un filtrado mínimo (4) y binarizando por nivel de histogramas.

$$I_{ojom}(i, j) = \text{mín}[I_{ojo}(i - k, j - l)] \quad k, l \in [-3, 3] \quad (4)$$

Esta imagen de grises, más homogénea que la original, es entonces binarizada para quedarse sólo con los pixeles más oscuros, candidatos a pertenecer al iris :

$$I_{iris}(i, j) = \begin{cases} 0 & I_{ojom}(i, j) < t \\ 255 & \text{otro} \end{cases}$$

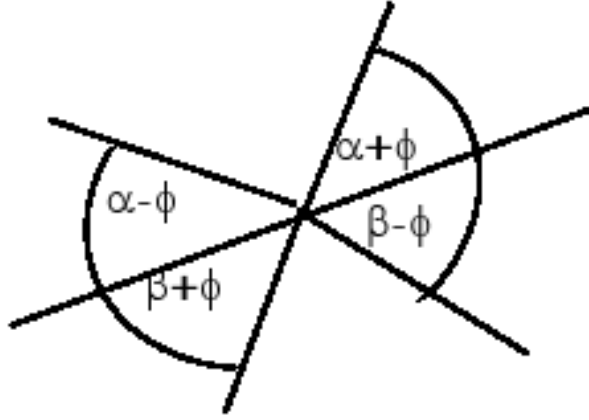


Figura 9: Molde del Iris para un rostro rotado en ϕ .

El umbral t segmenta el 35% de pixeles más oscuros del histograma acumulativo.

Para obtener la posición del centro del iris, se utiliza el hecho que la frontera iris-esclera presenta un alto contraste y, además, posee simetría circular. Con esto, se realiza el cálculo de dos integrales de línea sobre los caminos definidos por moldes del iris T_{I_ϕ} que modelan la oclusión de los párpados mediante arcos circulares donde se calcularán las integrales.

Estos arcos (Fig 9) presentan una apertura de α y β en 50° y 70° respectivamente y fueron obtenidos mediante mediciones empíricas [29]. Los moldes del iris contienen estos arcos para distintos radios y rotaciones del rostro ϕ .

La integral de línea se calcula entonces en forma discreta como:

$$\oint_{T_{I_\phi}_R(i_0, j_0)} I_n(i, j) \approx \frac{1}{N_{T_{I_\phi}_R}} \cdot \sum_{i, j \in T_{I_\phi}_R(i_0, j_0)} I_{ojo}(i, j) \quad (5)$$

donde (i_0, j_0) es un punto candidato a ser centro del iris y el término $N_{T_{I_\phi}_R}$ corresponde al número de puntos pertenecientes al molde $T_{I_\phi}_R$ y se utiliza para efectos de normalización. Se computan dos integrales para moldes de radio R y $R + \delta$ con δ igual a 2 (paso entre moldes) La razón entre estas dos integrales (6) será máxima cuando los moldes asociados a cada integral estén ubicados de manera que el externo (de radio $R + \delta$), se encuentre en la zona de la esclera y el interno se encuentre en la zona del iris. Este proceso se repite para todos los

puntos candidatos a ser centro del iris. El cálculo de la integral de línea se realiza sobre I_{ojo} dado que no está filtrado y por ende mantiene la precisión de la imagen original.

$$R_{LI_Φ}(i_0, j_0) = \frac{\oint_{I_{IΦ_R+\delta}(i_0, j_0)} I_n(i, j)}{\oint_{I_{IΦ_R}(i_0, j_0)} I_n(i, j)} \quad \forall (i_0, j_0) \in I_{Iris} \quad (6)$$

Para disminuir los cálculos computacionales, se define empíricamente una relación entre el tamaño del iris y el tamaño del rostro [29] dado por:

$$\begin{aligned} R_{min} &= 0,11 \cdot \left(\frac{r_{rostro}}{2} - 70 \right) \\ R_{max} &= R_{min} + 2 \end{aligned}$$

donde r_{rostro} corresponde al semieje mayor del rostro detectado en las etapas previas. El umbral de detección de los ojos fue definido en forma empírica como:

$$Iris? \begin{cases} \text{Sí} & R_{LI_Φ}(i_0, j_0) \geq 1,4 \\ \text{No} & \text{otro caso} \end{cases}$$