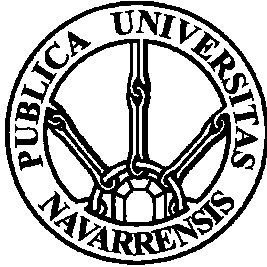


E.T.S. de Ingeniería Industrial,  
Informática y de Telecomunicación



# Detección robusta de pupila en entornos no controlados

Máster en Ingeniería Biomédica

Trabajo Fin de Máster

**Autor:** *Ion Martinikorena Aranburu*

**Tutores:** *Arantxa Villanueva Larre*  
*Rafael Cabeza Laguna*

*Pamplona, 27 de Abril de 2017*



**Proyecto: Detección robusta de pupila en entornos no controlados**

No está permitida la reproducción total o parcial de este proyecto, ni su tratamiento informático, ni la transmisión de ninguna forma o por cualquier medio, ya sea electrónico, mecánico, por fotocopia, por registro u otros métodos, sin el permiso previo y por escrito del autor del mismo.

## RESUMEN

En los sistemas de seguimiento de la mirada o eye tracking de tipo *head-mounted* la detección de pupila es un factor fundamental a la hora de estimar la dirección de mirada. A pesar de obtenerse buenos resultados en condiciones de laboratorio, esto no resulta sencillo en condiciones reales de uso debido a las numerosas complicaciones que surgen en dichos escenarios, tales como constantes cambios en la iluminación, reflejos, oclusiones y apariencia elíptica de la pupila etc. Por ello, es imprescindible disponer de algoritmos de detección de pupila robustos frente a dichas adversidades. Se presentan en este trabajo tres algoritmos de detección de pupila basados en buscar el punto de mayor contribución a la simetría radial de la imagen. En dos de ellos se incorporan dos etapas de refinado de centro para adaptar el método a pupilas de apariencia elíptica. La precisión en la detección es evaluada sobre un total de 225.569 imágenes de acceso público cuyo centro ha sido marcado de manera manual por los autores de las mismas. Los resultados son comparados con los obtenidos por el mejor de los algoritmos encontrados en la bibliografía.

## ABSTRACT

In *head-mounted* eye tracking systems, the correct detection of pupil position is a key factor to estimate gaze direction. However, this is a challenging issue when videos are recorded in real world conditions, due to many sources of noise presenting in such scenarios, like constant changes in illumination, reflections, oclusions and elliptic appearance of pupil etc. Thus, it is an indispensable prerequisite that the pupil detection algorithm must be robust against those challenging conditions. In this work, we present three pupil center detection methods based on searching the maximum contribution point to the radial symmetry of the image. Two different center refinement steps are incorporated in two of those algorithms with the aim to adapt the algorithms to images with highly elliptical pupil appearance. The performance of presented algorithms is evaluated on a data set consisting of 225569 annotated eye images from publicly available sources. Results are compared against the better algorithm found in bibliography.

<b>RESUMEN</b>	<b>I</b>
<b>ABSTRACT</b>	<b>I</b>
<b>CAPÍTULO 1 INTRODUCCIÓN</b>	<b>1</b>
EYE TRACKING	1
<b>CAPÍTULO 2 ESTADO DEL ARTE</b>	<b>8</b>
ALGORITMOS	9
<i>Starburst</i>	9
<i>SWIRSKI</i>	12
<i>SET</i>	15
<i>Pupil Labs</i>	16
<i>ExCuSE</i>	18
<i>ELSE</i>	21
COMPARATIVA	24
OBJETIVOS	25
<b>CAPÍTULO 3 MÉTODOS</b>	<b>26</b>
ALGORITMOS	26
COLECCIONES DE IMÁGENES	41
<b>CAPÍTULO 4 RESULTADOS</b>	<b>46</b>
CONCLUSIONES Y LÍNEAS FUTURAS	58
<b>REFERENCIAS</b>	<b>60</b>

# ÍNDICE DE FIGURAS

FIGURA 1 SISTEMA EYE TRACKING DISEÑADO POR ALFRED YARBUS .....	2
FIGURA 2 FUNDAMENTO DE LA ELECTRO-OCULOGRAFÍA .....	3
FIGURA 3 EYE TRACKING DE TIPO REMOTO Y DE TIPO <i>HEAD - MOUNTED</i> .....	4
FIGURA 4 DISPOSITIVO EYE TRACKING DE BAJO COSTE. ....	4
FIGURA 5 ESQUEMA DE ALINEACIÓN DE CÁMARA CON EL EJE VISUAL .....	6
FIGURA 6 EJEMPLOS DE IMÁGENES DE PUPILAS CON APARIENCIA ELÍPTICA .....	6
FIGURA 7 EJEMPLOS DE SISTEMAS DE EYE TRACKING DE TIPO <i>HEAD-MOUNTED</i> .....	6
FIGURA 8 STARBURST. PASOS DEL ALGORITMO .....	11
FIGURA 9 OBTENCIÓN DE PUNTOS DE CONTORNO DE PUPILA .....	11
FIGURA 10 ALGORITMO SWIRSKI. ESTIAMCIÓN APROXIMADA DE LOCALIZACIÓN DE PUPILA.....	12
FIGURA 11 ALGORITMO SWIRSKI. UMBRALIZACIÓN. ....	13
FIGURA 12 ALGORITMO SWIRSKI. AJUSTE DE ELIPSE .....	14
FIGURA 13 ALGORITMO SET. FASES DEL ALGORITMO .....	16
FIGURA 14 ALGORITMO PUPIL LABS. FASES DEL ALGORITMO .....	17
FIGURA 15 ALGORITMO EXCUSE. DETECCIÓN DE REFLEJO POR HISTOGRAMA .....	18
FIGURA 16 ALGORITMO EXCUSE. ESTIMACIÓN DE CENTRO DE PUPILA EN PRESENCIA DE REFLEJO .....	19
FIGURA 17 ALGORITMO EXCUSE. ESTIMACIÓN DE LA PUPILA POR FIPA .....	20
FIGURA 18 ALGORITMO EXCUSE. REFINAMIENTO DE CENTRO DE PUPILA.....	20
FIGURA 19 ALGORITMO ELSE. DIAGRAMA DE FLUJO.....	21
FIGURA 20 ALGORTIMO ELSE. LOCALIZACIÓN APROXIMADA.....	22
FIGURA 21 ALGORTIMO ELSE. FILTROS PARA LOCALIZACIÓN APROXIMADA.....	23
FIGURA 22 ALGORTIMO ELSE. REFINADO DE LA POSICIÓN APROXIMADA.....	23
FIGURA 23 COMPARATIVA RENDIMIENTO DE LOS ALGORITMOS.....	25
FIGURA 24 DIAGRAMA DE BLOQUES DE LOS ALGORITMOS PROPUESTOS.....	26
FIGURA 25 PASOS PARA OBTENER EL CENTRO DE PUPILA MEDIANTE EL MÉTODO DE SKODRAS .....	27
FIGURA 26 PASOS PARA OBTENER EL MAPA DE LUMINANCIA .....	29
FIGURA 27 PROCESADO DE LA IMAGEN PREVIO A SKODRAS .....	31
FIGURA 28 DIAGRAMA DE BLOQUES DEL ALGORITMO FRED A I .....	32
FIGURA 29 EJEMPLO DE APLICACIÓN DE FRST SOBRE PUPILAS DE APARIENCIA ELÍPTICA .....	33
FIGURA 30 DIAGRAMA DE FLUJO DEL REFINAMIENTO DE CENTRO DE FRED A I .....	34
FIGURA 31 EJEMPLO DE FUNCIONAMIENTO DEL PROCESO DE REFINAMIENTO DE CENTRO DEL ALGORITMO FRED A I.....	36
FIGURA 32 CENTRO OBTENIDO CON FRST VS CENTRO CORREGIDO .....	36
FIGURA 33 DIAGRAMA DE BLOQUES DEL ALGORITMO FRED A II .....	37
FIGURA 34 DIAGRAMA DE FLUJO DEL REFINAMIENTO DE CENTRO DE FRED A II .....	38
FIGURA 35 EJEMPLO DE FUNCIONAMIENTO DEL PROCESO DE REFINAMIENTO DE CENTRO DEL ALGORITMO FRED A II.....	39
FIGURA 36 CENTRO OBTENIDO CON FRST VS CENTRO CORREGIDO .....	40

FIGURA 37 EJEMPLOS DE IMÁGENES DE LA COLECCIÓN DE TÜBINGEN .....	42
FIGURA 38 EJEMPLOS DE IMÁGENES DE LA COLECCIÓN DE SWIRSKI .....	43
FIGURA 39 EJEMPLOS DE IMÁGENES DE LA COLECCIÓN LPW .....	45
FIGURA 40 GRÁFICA DE TASA DE DETECCIÓN SOBRE EL TOTAL DE IMÁGENES .....	47
FIGURA 41 GRÁFICAS DE TASA DE DETECCIÓN PARA CADA COLECCIÓN DE IMÁGENES .....	48
FIGURA 42 VARIABILIDAD DE LAS CURVAS DE ERROR SOBRE EL TOTAL DE IMÁGENES .....	49
FIGURA 43 VARIABILIDAD DE LAS CURVAS DE ERROR EN LAS COLECCIONES DE SWIRSKI Y LPW .....	49
FIGURA 44 EJEMPLOS DE IMÁGENES DIFÍCILES DE TÜBINGEN .....	53
FIGURA 45 EJEMPLOS DE IMÁGENES DIFÍCILES DE LPW .....	54
FIGURA 46 EJEMPLOS DE DETECCIÓN DE CENTRO PARA FRED A II Y ELSE EN IMÁGENES DE LPW .....	54
FIGURA 47 EJEMPLOS DE DETECCIÓN DE CENTRO PARA FRED A II Y ELSE EN IMÁGENES DE LPW .....	55
FIGURA 48 EJEMPLO DE CENTRO MAL ESTIMADO MEDIANTE LA FRST .....	55
FIGURA 49 COMPARATIVA DE CENTRO ESTIMADO CON FRED A Y FRED A II EN PUPILAS DE APARIENCIA ELÍPTICA .....	56
FIGURA 50 EJEMPLOS DE FUNCIONAMIENTO ERRONEO DEL ALGORITMO FRED A II SOBRE IMÁGENES DE TÜBINGEN .....	57

## ÍNDICE DE TABLAS

TABLA 1 TASA DE ACIERTOS DE LOS ALGORITMOS PARA CADA CONJUNTO DE IMÁGENES .....	24
TABLA 2 TASA DE ACIERTOS DE LOS ALGORITMOS PROPUESTOS Y ELSE PARA CADA CONJUNTO DE IMÁGENES .....	50
TABLA 3 TASA DE ACIERTOS PARA CADA COLECCIÓN DE IMÁGENES .....	51
TABLA 4 TASA DE ACIERTOS DE LOS ALGORITMOS PROPUESTOS Y ELSE SOBRE EL TOTAL DE IMÁGENES .....	51

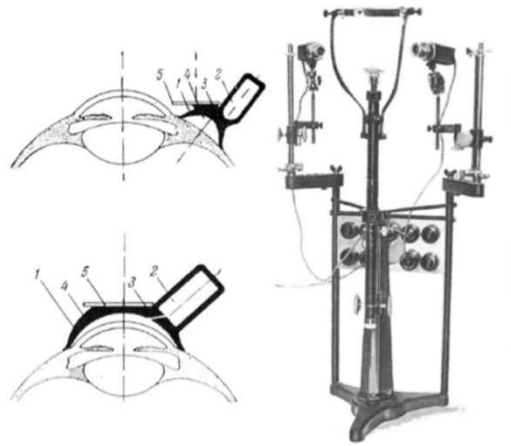
# CAPÍTULO 1 INTRODUCCIÓN

## Eye tracking

El término eye tracking hace referencia tanto al proceso de seguimiento del movimiento ocular como a la estimación del “point of gaze” (POG) o punto de mirada, entendiéndose éste como el punto de la escena en que el sujeto fija la mirada (1). Durante el último siglo, la observación y medida de los movimientos oculares ha sido utilizado para conocer de manera exhaustiva el funcionamiento oculo-motor así como el sistema de percepción visual, aportando además ideas clave sobre diversos procesos cognitivos (2). Varios campos de estudio como la medicina, psicología, publicidad, conducción autónoma o aplicaciones de control se han enfocado en comprender los procesos subyacentes a la percepción visual (3).

Inicialmente el movimiento de los ojos era estudiado de manera fisiológica y por el método de la observación. Antes de que la tecnología de eye tracking permitiera medidas más precisas, el estudio se restringía a clasificar los movimientos más básicos y la duración de estos se estimaba de manera aproximada(1). Los primeros sistemas de eye tracking resultaban muy invasivos e incómodos. Por ejemplo, en uno de los trabajos más conocidos y citados, publicado por Alfred Yarbus en 1956, para el seguimiento ocular se hacía uso de un capuchón adherido por succión al globo ocular en el cual se colocaban los instrumentos necesarios para realizar las medidas. Ello suponía tener que anestesiarse la córnea y fijar los párpados para evitar daños o despegar el capuchón (4).

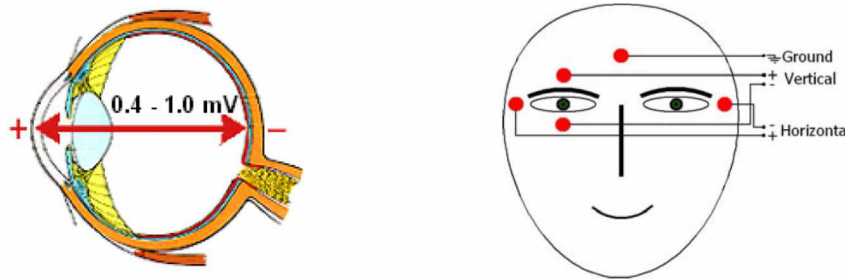




**Figura 1.** *Capuchones diseñados por Yarbuss, adheridos a la córnea y esclera (izquierda). Sistema de grabación para el seguimiento ocular (derecha).*

Sin embargo, el primer eye tracker no invasivo fue desarrollado por Dodge y Cline en 1901 en el cual las reflexiones producidas en la córnea por una fuente de luz, se registraban en una placa fotográfica pudiendo conocer así posteriormente el movimiento ocular. Este dispositivo, mejorado con los años, es considerado el primer antecesor de la técnica moderna de estimación de mirada basada en video y reflexiones corneales. Desde entonces han sido numerosas las técnicas desarrolladas con el fin de mejorar la precisión tanto en el propio seguimiento ocular como en la estimación del POG. Hoy en día las tecnologías de eye tracking se pueden clasificar en tres categorías: electro-oculografía, basadas en lentes de contacto y basadas en vídeo (5).

La electro-oculografía (EOG) mide el movimiento de los ojos mediante el registro de la diferencia de potencial existente entre la córnea y la retina. Así, considerando el ojo como un dipolo, se puede estimar la orientación de éste mediante un par de electrodos de superficie colocados cerca del globo ocular (1) (véase Figura 2). La EOG no ofrece mucha precisión espacial en la estimación de punto de mirada. Sin embargo, dado que registra poca información y puede funcionar incluso con los ojos cerrados, es ampliamente utilizada en el registro del sueño, por ejemplo para determinar las fases REM.



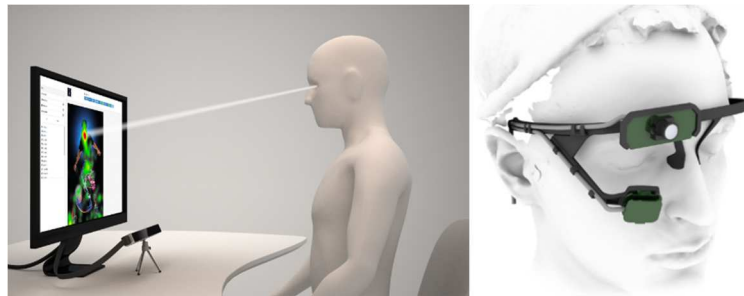
**Figura 2.** Polarización del globo ocular (izquierda) y colocación de los electrodos (derecha)(6)

El método más preciso pero a la vez, el más invasivo e incómodo para el usuario es el uso de lentes de contacto en las cuales se inserta algún sistema de seguimiento, como puede ser una bobina magnética o un pequeño espejo. Así, dado que la lente está adherida a la córnea, el problema de seguimiento ocular se restringe a la localización del elemento añadido a la lente. Esta tecnología únicamente es utilizable en entornos de laboratorio. Por ello la hace inservible para aplicaciones del día a día. Sin embargo, dada la alta precisión que se obtiene, es utilizada como referencia para probar sistemas más versátiles y no invasivos pero menos exactos (7).

El tercer método, el seguimiento ocular basado en vídeo, resulta el menos invasivo y debido a la miniaturización y abaratamiento de las cámaras y componentes electrónicos en general, se ha impuesto como método preferencial para el desarrollo de sistemas de eye tracking siendo utilizado en diferentes áreas, desde complejas aplicaciones en tiempo real, como la ayuda a la conducción basada en la detección de mirada (8) o interfaces persona computadora (9), hasta sencillas tareas como registrar las zonas de atención en páginas web (10).

Los sistemas de eye tracking basados en vídeo se dividen en dos categorías, los de seguimiento remoto y los denominados *head – mounted* (11). En los primeros, una o más cámaras fijas se alinean frente al sujeto registrando el rostro completo de éste (Figura 3a). Por tanto, para estimar la dirección de la mirada es necesario realizar una detección de cara, determinar las regiones oculares y, posteriormente, localizar el centro de pupila. En los sistemas tipo *head – mounted* sin embargo, se adquieren imágenes por medio de una o dos cámaras colocadas sobre un soporte fijado a la cabeza del sujeto (Figura 3b). En dichos

equipos se precisa que una cámara registre la escena que se está observando mientras que al menos una más, la cual es dirigida hacia el sujeto, adquiera las imágenes oculares.



**Figura 3** (a) Sistema eye tracking remoto. (b) dispositivo de eye tracking de tipo head-mounted

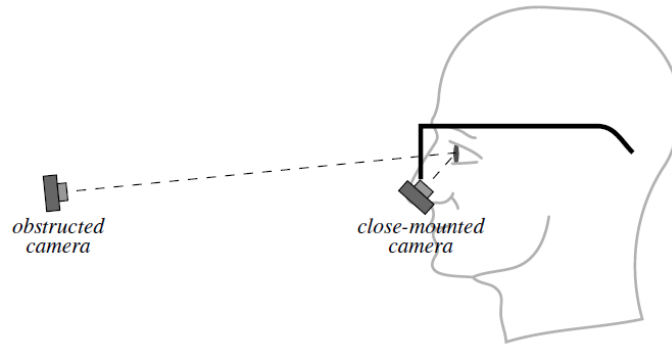
Uno de los mayores inconvenientes que se han encontrado a la hora de extender esta tecnología a la interacción persona - computadora para uso cotidiano es que históricamente han sido excesivamente invasivos y caros. A pesar de ello, en los últimos años, el tamaño de dichos dispositivos se ha reducido de manera drástica debido a la miniaturización de los componentes electrónicos. Sin embargo, aún resultan excesivamente caros principalmente por disponer de hardware muy específico y software diseñado expresamente para el dispositivo cuyo código fuente suele estar protegido (12, 13). Con el fin de superar este obstáculo, y extender esta tecnología al uso común, se han desarrollado sistemas de eye tracking de bajo coste y código abierto cuyos componentes provienen de equipos de computación de propósito general y para los cuales es posible desarrollar diferentes algoritmos de estimación de la mirada (12, 14).



**Figura 4.** Dispositivo de eye tracking PUPIL de bajo coste y código abierto

Una parte fundamental para una precisa estimación de mirada es la correcta detección del centro de pupila(2). A pesar de que los sistemas eye tracking de tipo *head-mounted* han mostrado resultados satisfactorios bajo condiciones de laboratorio (13, 15, 16), numerosos estudios reportan dificultades y bajas tasas de detección cuando las pruebas se realizan en situaciones reales tales como la conducción (8),(17),(18),(19), en la selección de productos en comercios (20) o simplemente caminando (21). La principal fuente de ruido en estos escenarios es la baja definición de la pupila debido a los constantes cambios en la iluminación, reflejos, emborronamiento por movimiento u oclusiones de la pupila a causa de las pestañas. Los cambios rápidos en la iluminación suelen ocurrir cuando el sujeto se mueve a velocidades altas o cuando el sujeto rota en un entorno con una distribución de fuentes de luz no uniforme (2). El uso de lentes de contacto o gafas por parte del sujeto genera numerosos reflejos de muy variada tipología que en ocasiones pueden llegar a ocluir la pupila por completo (3). Otras fuentes de errores en la detección de la pupila pueden ser la presencia de zonas oscuras en la imagen debida a sombras causadas por párpados, así como el uso de máscara de pestañas.

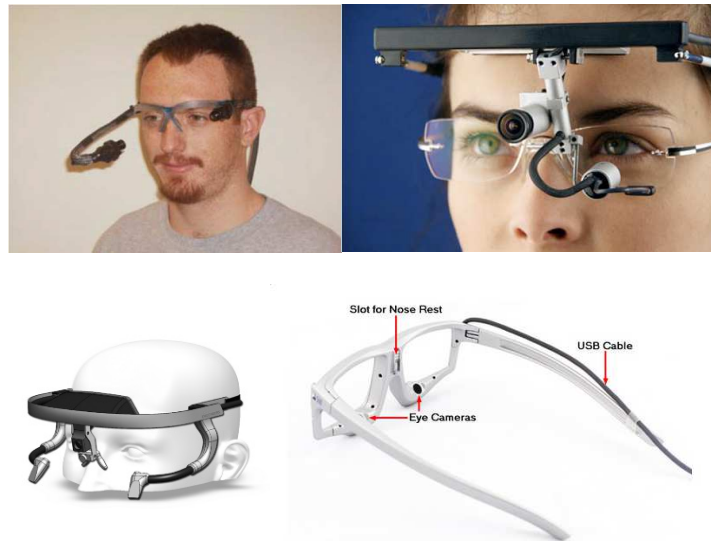
Una dificultad añadida a la hora de detectar la pupila mediante estos sistemas es la frecuente presencia de pupilas con apariencia elíptica. Para una óptima detección lo más indicado resulta colocar la cámara frente al ojo. Sin embargo, esto dificulta la visión natural del sujeto, haciendo que se tienda a colocar la cámara en el borde del campo de visión (22), como se muestra en la Figura 5. Además, para evitar oclusiones de la pupila por el uso de gafas, en algunos sistemas se coloca la cámara en el borde de éstas. Así, la alta angulación resultante entre la dirección de mirada y el eje central de la cámara hace que la pupila adquiera una apariencia excesivamente elíptica (véase Figura 6). Se muestran en la Figura 7 varios dispositivos con diferentes disposiciones de la cámara.



**Figura 5.** El uso de gafas puede obstruir la visión de la pupila en los sistemas de eye tracking normales con cámara frontal. Colocando la cámara entre la montura y el ojo se consigue una visión continua de esta pero con el inconveniente de obtener imágenes de la pupila con una apariencia elíptica.



**Figura 6.** Las cámaras muy próximas a los ojos obligan a colocarlas muy alejadas del eje de visión haciendo que la pupila adquiera una apariencia elíptica.



**Figura 7.** Diferentes modelos de sistemas tipo head-mounted. En la fila superior se muestran dos ejemplos donde la cámara se coloca muy próxima al centro visual del ojo. En la fila inferior sin embargo, las cámaras, dirigidas a cada ojo, se sitúan muy próximas a ellos y en consecuencia, muy alejadas del eje de visión.

Por todo ello, para integrar la tecnología de eye tracking para su uso cotidiano es imprescindible disponer de algoritmos de detección de pupila que sean robustos frente a las condiciones descritas, rápidos y compatibles con hardware de propósito general presentes en los equipos de bajo coste (13).

Con ese objetivo se presentan en este trabajo tres algoritmos de detección de pupila basados en la transformada rápida de simetría radial o FRST (Fast Radial Symmetry Transform) desarrollada por Loy y Zelinsky (23) cuyo propósito es determinar el centro de la zona con mayor grado de simetría radial de la imagen. Se describe en detalle el proceso en el capítulo 3. La FRST ha sido utilizada con anterioridad para la detección del centro del iris en sistemas de eye tracking remoto por Skodras et al. (24). Por tanto, se adapta dicho método a las características de las imágenes oculares obtenidas mediante los dispositivos tipo *head-mounted*. Además, se presentan dos métodos diferentes para compatibilizar el uso de la transformada FRST para la detección del centro en pupilas con apariencia elíptica. La precisión de los algoritmos es evaluada sobre un total de 225.569 imágenes oculares provenientes de tres colecciones de acceso público (3, 22, 25). Se realiza asimismo una comparativa de rendimiento sobre el mismo conjunto de imágenes contra el algoritmo ELSE desarrollado por Fuhl et al. (3) cuya capacidad de detección recientemente se ha mostrado superior frente a otros algoritmos presentes en la literatura (11).

# CAPÍTULO 2 ESTADO DEL ARTE

Durante los últimos años ha habido un gran avance en el ámbito de la detección de pupila con la aparición de numerosos algoritmos de detección (11). En general, estos se pueden clasificar en dos grupos: los basados en la detección de características y los basados en modelos.

Los algoritmos basados en características tratan de identificar puntos en la imagen que se relacionan con la posición del ojo, como por ejemplo el centro de la pupila o puntos del borde. Un elemento común en este tipo de algoritmos suele ser la necesidad de determinar un umbral para decidir la presencia o no de una cierta característica que, generalmente, depende de niveles de gris o gradientes de intensidad.

Por el contrario, en los algoritmos basados en modelos, no se buscan características distintivas en la imagen, sino que se intenta buscar el mejor modelo de ajuste que sea consistente con la imagen. Normalmente estos algoritmos suelen buscar el mejor ajuste de una elipse o circunferencia al contorno de pupila, considerando el centro de éste modelo de contorno como el centro de la pupila (5). Estos algoritmos, en general, suelen resultar más precisos en la detección de pupila que los basados en la búsqueda de características. Sin embargo, el coste computacional es notablemente superior al de los anteriores (14). Por ello, suelen ir acompañados de técnicas de procesamiento multi-escala en las que en diferentes etapas del algoritmo se trabaja con imágenes a menor resolución que la original (14).

En cuanto al tipo de imágenes con las que trabajan los algoritmos, éstas se puede dividir en dos tipos: de espectro visible e infrarrojo. Las primeras se obtienen de manera pasiva registrando en la cámara la luz ambiente reflejada en los ojos. Las condiciones no controladas de iluminación en este tipo de sistemas generan numerosas dificultades en la detección de pupila debido a reflexiones especulares y difusas. EL uso de iluminación infrarroja elimina dichas reflexiones además de obtener contrastes de pupila mayores con respecto del iris. Una característica en este tipo de imágenes es el uso de la reflexión corneal causada por los

iluminadores infrarrojos en la córnea para, junto con el centro de la pupila, estimar la dirección de mirada(14).

Se presentan a continuación la descripción de seis algoritmos de detección de pupila desarrollados para sistemas de seguimiento de mirada de tipo *head-mounted* así como una evaluación de la precisión de cada uno de ellos publicada por Fuhl et al. (11). Los algoritmos fueron probados sobre un total de 225.569 imágenes provenientes de tres colecciones independientes de imágenes oculares cuyo centro ha sido marcado de manera manual. Estas colecciones se describen en detalle en el capítulo 3.

## Algoritmos

### STARBURST

Originalmente ideado para sistemas de Eye tracking de bajo coste y compatible con equipos de propósito general, este algoritmo de Eye tracking combina técnicas tanto de métodos basados en modelos como de los basados en características, ofreciendo así un equilibrio entre tiempo de cómputo y precisión en la detección. El algoritmo funciona sobre imágenes obtenidas mediante iluminación infrarroja. El propósito es utilizar un reflejo en la córnea utilizado

Primeramente, a la imagen de entrada se le aplica un filtro gaussiano paso bajo para eliminar el ruido, tras lo cual, mediante una umbralización adaptativa aplicada en una región de interés de la imagen, se localiza el centro del reflejo corneal. Éste se estima tomando el centro geométrico de la región más extensa resultante de la umbralización adaptativa mencionada. Una vez localizado el centro del reflejo, éste se elimina mediante una interpolación radial. Para ello, se asigna al píxel marcado como centro del reflejo un valor de gris igual a la media de los píxeles del contorno de la región. Al resto de píxeles entre el centro y el borde de la región se les asigna un valor de gris por medio de una interpolación lineal. El resultado de este proceso se puede observar en la Figura 8. Con el reflejo eliminado, se aplica el paso fundamental del algoritmo, representado en la Figura 9, mediante el cual se obtiene un conjunto de puntos pertenecientes al contorno de la pupila. La idea para detectar dichos puntos consiste en extender una cantidad limitada de rayos comenzando desde una primera

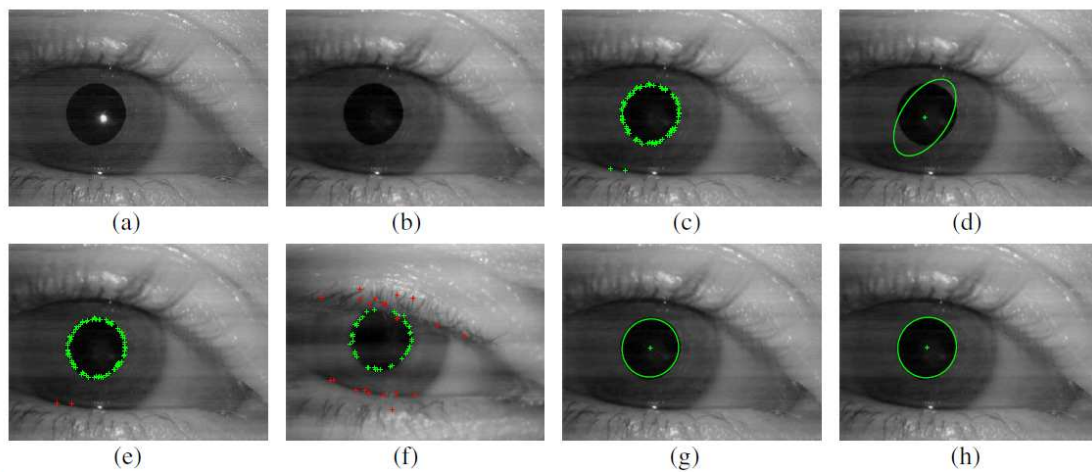


estimación del centro de pupila y marcar, en cada rayo, el punto en el cual la derivada de los valores de gris a lo largo de la trayectoria sea máxima. Este procedimiento supone un ahorro computacional respecto a los métodos tradicionales de detección de bordes por gradiente, laplaciano o similares. Seguidamente, desde cada candidato a punto de borde, se extiende una cantidad igual de rayos en los que se busca de manera idéntica más puntos de borde. Se toma el centro de todos los puntos detectados y se repite el proceso partiendo esta vez de esta nueva estimación de centro. Se itera este procedimiento hasta que la distancia entre los centros obtenidos entre iteraciones sucesivas sea inferior a 10 píxeles o hasta un máximo de 10 iteraciones. Una vez obtenido un conjunto de candidatos a puntos de borde de pupila, se calcula la elipse que mejor se ajusta a ellos mediante el método RANSAC. Dicho procedimiento selecciona múltiples subconjuntos aleatorios de datos para los cuales ajusta un modelo y encuentra aquel modelo que mayores similitudes tiene con el conjunto entero de datos. El subconjunto de datos para el cual se ha obtenido este modelo se denomina conjunto de consenso. En este algoritmo se toman varios subconjuntos de puntos para los cuales se estima una elipse de ajuste. Aquel subconjunto para el cual haya un menor número de valores atípicos, definiendo éstos como puntos que se encuentran más alejados de la elipse que cierta distancia umbral, se considera el subconjunto de consenso.

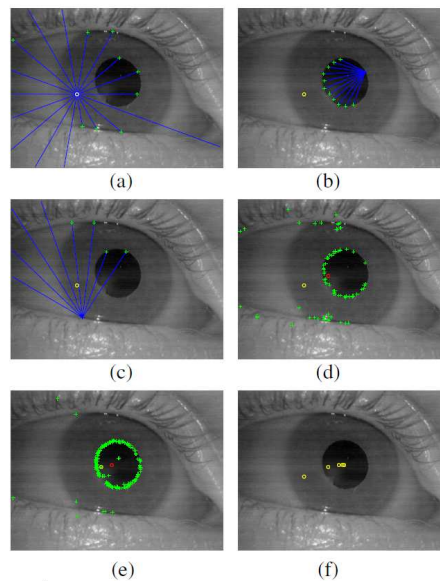
Para mejorar la estimación de la elipse obtenida por el método de búsqueda de características (puntos del borde de pupila) se mejora dicha elipse por medio de una optimización basada en modelo en la que se minimiza la expresión

$$-\frac{\int I(a + \delta, b + \delta, \alpha, x, y, \theta)d\theta}{\int I(a - \delta, b - \delta, \alpha, x, y, \theta)d\theta}$$

Donde  $\delta = 1$  e  $I(a, b, \alpha, x, y, \theta)$  representa el valor de gris en el contorno de la elipse de parámetros  $a, b, \alpha, x$  e  $y$  situado a un ángulo  $\theta$  respecto del semieje mayor. El modelo parte de la idea de que dada la elipse de contorno de pupila, la elipse interior ( $-\delta$ ) debe contener los píxeles más oscuros (valores de gris más bajos) y la exterior ( $+\delta$ ) más claros (valores de gris más altos). Así, el cociente negativo entre la suma de píxeles de la elipse exterior y la interior debe tender a  $-\infty$  según se hace más precisa la aproximación. Se toma como elipse de partida la obtenida a partir del método RANSAC.



**Figura 8** Diferentes fases del algoritmo Starburst (13) (a) imagen original (b) imagen con el reflejo corneal eliminado. (c) candidatos de puntos de borde (d) elipse de ajuste por mínimos cuadrados. (e) en rojo se marca los valores atípicos estimados mediante RANSAC (f) otro ejemplo con más valores atípicos (g) Elipse de ajuste obtenida únicamente con los valores no atípicos (verdes) (h) elipse optimizada mediante el ajuste por modelo.



**Figura 9** Procedimiento de obtención de puntos de características (13). (a) Trayectorias desde el centro inicial (amarillo) y puntos detectados (verde). (b) y (c) dos ejemplos de la segunda generación de puntos característicos que parten de los anteriores. (d) Total de puntos obtenidos en la primera iteración y centro estimado a partir de ellos (rojo). (e) Resultado tras la segunda iteración y segundo estimado (rojo). (f) Proceso de convergencia de los centros estimados tras cuatro iteraciones.

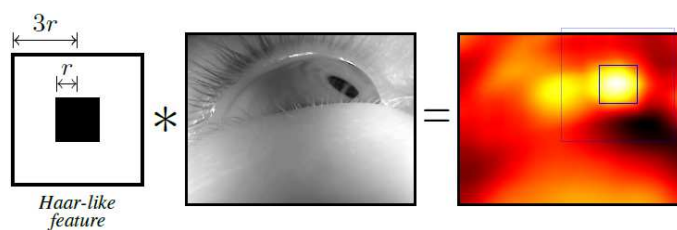
SWIRSKI

El algoritmo presentado por Swirski et al. (22) tiene como principal objetivo solventar la falta de precisión de los algoritmos de detección cuando, debido a la angulación de la cámara con respecto al eje visual: la apariencia de la pupila es excesivamente elíptica o se encuentra parcialmente ocluida por las pestañas.

El algoritmo consta de tres etapas principales.

### ***Estimación inicial de la región***

Como primera aproximación a la detección de la pupila se asume que la pupila, o la pupila en combinación con el iris, se puede describir como una mancha oscura rodeada de un contorno claro. Para determinar una región de estas características se hace uso de un detector tipo Haar. La idea es convolucionar la imagen inicial con un elemento Haar tipo centro-contorno (véase Figura 10). Este proceso se repite variando el radio del centro entre un mínimo y un máximo especificado por el usuario y, se obtienen tanto la posición en la imagen como el radio para los cuales se ha encontrado el máximo valor. Así, la primera aproximación sobre la estimación de la región de la pupila viene dada por una región rectangular centrada en el punto anterior y con un tamaño dado por el radio determinado en el mismo proceso.

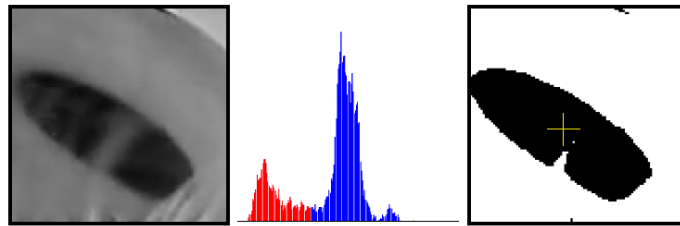


**Figura 10** Estimación aproximada de la localización de la pupila. Convolución entre un elemento tipo Haar (izquierda) con la imagen original.

### ***Segmentación de la pupila por histograma***

Puesto que la región de pupila estimada se limita a una zona cuadrada de la imagen, se aplica un segundo paso para determinar con más precisión la forma de la misma (Figura 11). La técnica utilizada es una segmentación de la imagen por histograma, asumiendo que, en la

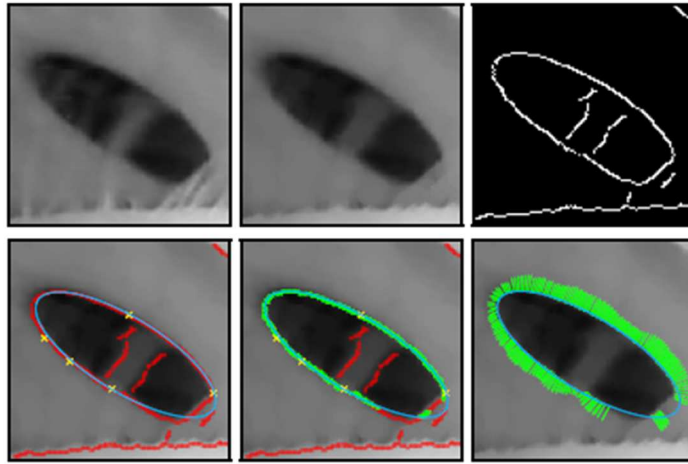
zona seleccionada, los píxeles más oscuros corresponden a la pupila. Para determinar el umbral que divida la parte oscura de la clara en la segmentación se recurre al método k-mean, siendo así un procedimiento completamente automático y dependiente del propio histograma de la imagen. Se considerará la región de la pupila al mayor conjunto conexo de píxeles negros obtenidos tras la umbralización. La posición de la pupila se calcula como el centro de masas del conjunto seleccionado de píxeles.



**Figura 11** Proceso de umbralización de la región de pupila. (a) recorte de imagen sobre la zona detectada. (b) histograma del recorte y umbral de división. (c) imagen umbralizada.

### ***Ajuste de elipse***

Para precisar aún más la localización de la pupila se realiza un refinamiento ajustando una elipse al borde entre la región de la pupila estimada y el iris (Figura 12). Se realiza un recorte de imagen tomando como centro el punto obtenido en el paso anterior. A dicho recorte se le aplica una apertura morfológica con la que se eliminan los artefactos producidos por pestañas y pequeños brillos para, seguidamente, realizar una detección de bordes mediante el método de Canny. El ajuste de la elipse se realiza sobre dichos puntos utilizando el método RANSAC al igual que en el algoritmo Starburst.

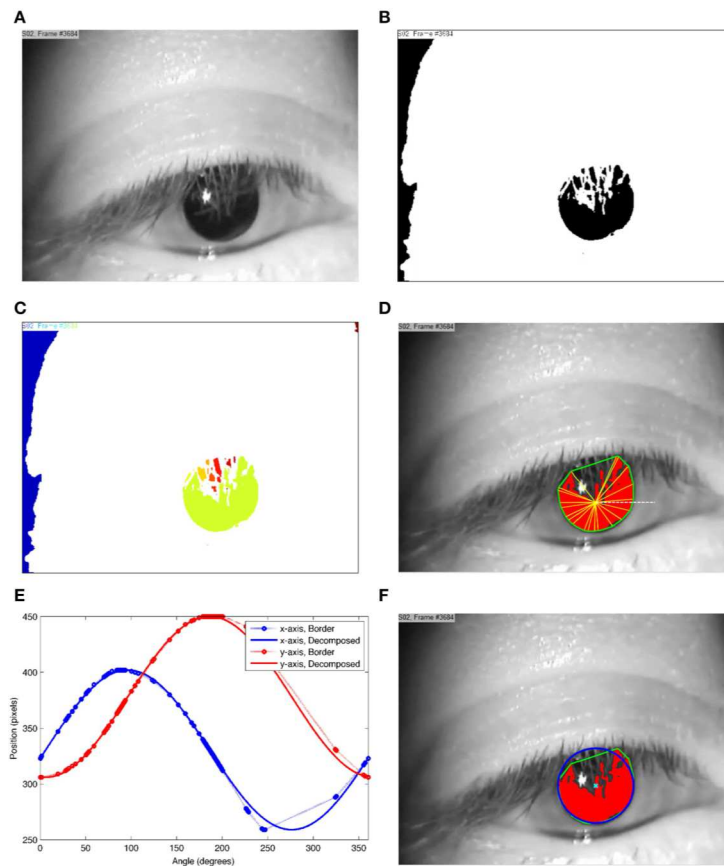


**Figura 12** Proceso de ajuste de elipse. (a) recorte original de la imagen. (b) Imagen tras aplicar una apertura. (c) Bordes de la imagen obtenidos mediante Canny. (d) Elipse de ajuste inicial sobre 5 puntos del borde (amarillo).

## SET

La principal característica del algoritmo SET es la combinación de pasos tanto automáticos como manuales que permiten adaptar el sistema a diferentes condiciones (26). Así, como paso previo al algoritmo de detección de pupila, se selecciona un umbral para la transformación de la imagen de escala de grises a binaria y el tamaño mínimo de los segmentos obtenidos que se tendrán en cuenta para la detección de la pupila. Se muestra en la Figura 13 el proceso del algoritmo.

Una vez seleccionados los parámetros, primeramente se extraen las áreas en las que es más posible localizar la pupila mediante un proceso de umbralización que da como resultado una imagen binaria (Figura 13-b). Se segmenta la imagen binaria diferenciando después las regiones independientes que hayan podido obtenerse tras la umbralización (Figura 13 C). Para cada una de ellas se obtiene la envolvente convexa y se extrae el borde el cual se utiliza para estimar una elipse de ajuste (Figura 13d). El proceso de ajuste se lleva a cabo descomponiendo los puntos del borde en sus componentes sinusoidales (Figura 13e). Para ello se toma como punto central la media aritmética de los puntos de borde y se miden los ángulos que forman las trayectorias entre dicho centro y los puntos del borde respecto de la horizontal. Para cada ángulo se representa la posición de la componente x e y de cada punto. Puesto que el borde no es circular, ésta representación contiene desviaciones sobre una senoidal. Mediante polinomios de Taylor se calculan las senoidales que mejor se aproximan a la distribución angular de cada componente y se obtiene así la elipse de ajuste al segmento inicial (Figura 13 F). Por tanto, de entre todas las elipses obtenidas de cada segmento, se selecciona aquella con una relación de aspecto entre ejes más cercano a la unidad, es decir, la elipse más circular. Su centro se considera el centro de la pupila.

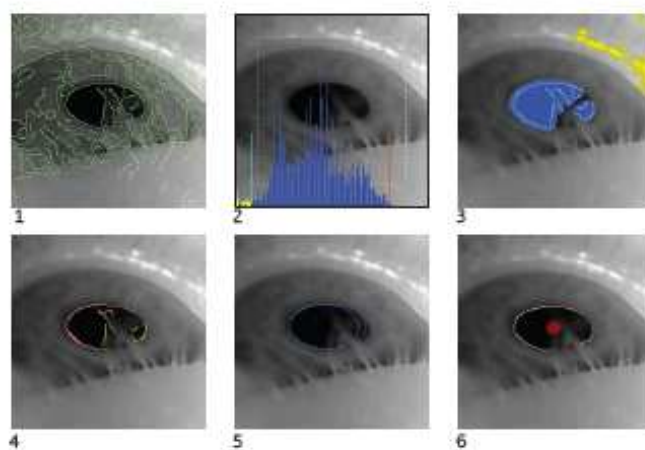


**Figura 13** Pasos del algoritmo SET(26). (A) Imagen original (B) Imagen umbralizada (C) Representación de la segmentación de los diferentes componentes conexos. (D) Uno de los segmentos (rojo) con el borde extraído (verde) de la componente convexa del mismo. En amarillo se representa la línea que une cada punto con el centro obtenido del borde. La línea blanca supone la referencia para medir el ángulo de cada trayectoria. (E) Distribución angular de las componentes x (azul) e y (rojo) de los puntos del borde. La línea gruesa muestra la aproximación senoidal a la distribución de puntos. (F) Elipse de ajuste (azul) obtenida de las dos componentes senoidales y centro estimado de la pupila (verde).

## PUPIL LABS

El algoritmo de detección Pupil Labs fue desarrollado junto con el sistema de código abierto de Eye Tracking denominado Pupil (12). Se muestran en la Figura 14 los pasos para determinar el centro de pupila. El algoritmo trabaja sobre imágenes obtenidas mediante iluminación infrarroja y tiene la característica de ser robusto frente a reflexiones en la pupila en la mayoría de ambientes.

Primeramente, tras pasar la imagen obtenida a escala de grises se realiza una primera aproximación a la localización de la pupila buscando la máxima respuesta entre la convolución de la imagen y un elemento de características centro-contorno al igual que se hace en el algoritmo de Swirski. Una vez obtenida el área de máxima respuesta, se realiza una detección de bordes por el método Canny (Figura 14-1). Estos bordes son filtrados en función de la intensidad de los píxeles que los rodean. Posteriormente se escogen los píxeles más oscuros de la imagen en función de un umbral establecido por el usuario (Figura 14-3). Los bordes en torno a los reflejos son también eliminados (zona amarilla de la Figura 14-4). Los restantes bordes son analizados y se generan bordes analizando la conectividad entre ellos. De ellos, se excluyen aquellos que no cumplan con cierto criterio de curvatura. (Figura 14-5). Con este subconjunto de bordes se realizan varios ajustes de elipse y se escoge la mejor candidata atendiendo a la aproximación de ésta al borde correspondiente así como al ratio entre la longitud de éste y la excentricidad de la elipse obtenida (Figura 14-6). Para determinar que se ha encontrado la pupila dicho ratio debe superar un umbral establecido por el usuario, en caso contrario, se reporta que no ha habido detección de pupila.



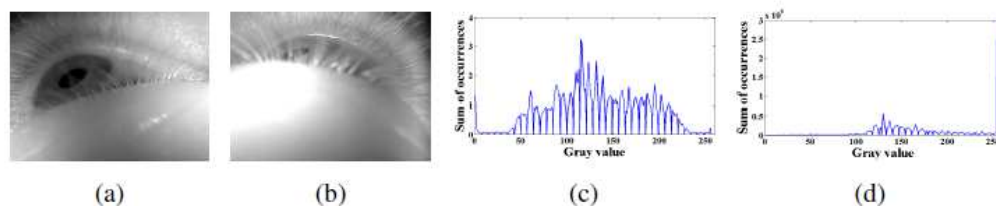
**Figura 14.** Pasos del algoritmo Pupil Labs (12).



## EXCUSE

El algoritmo ExCuSe es uno de los últimos algoritmos de detección de pupila desarrollados (16). Su principal característica radica en la elección del método de detección en función de la presencia o no de reflejos en la imagen de entrada. En caso de detectarse un reflejo, se escoge un procedimiento de búsqueda de pupila basado en la localización de bordes. En caso contrario, se realiza una búsqueda aproximada de la pupila sobre una umbralización de la imagen a la que posteriormente se le aplica un refinamiento del mismo en base a los valores de gris de alrededor.

Por tanto, como primer paso, tras normalizar la imagen de entrada, se obtiene el histograma y se deduce la presencia de un reflejo por la detección de un pico en la parte correspondiente a los píxeles más claros de la imagen (Figura 15).

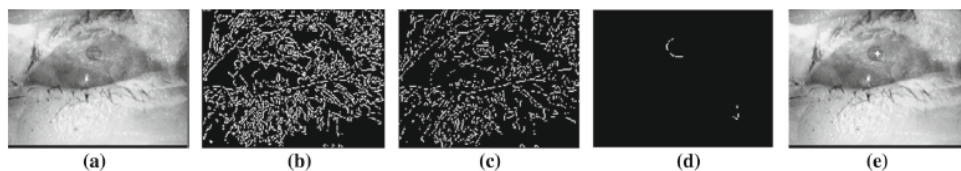


**Figura 15** Detección de reflejo por histograma. (a) imagen sin reflejo. (b) imagen con reflejo (c) histograma de la primera imagen. (d) histograma de la segunda imagen donde se observa el pico en la parte derecha de la imagen.

### Método por detección de bordes

El objetivo de éste método es determinar el borde que mejor defina el contorno de pupila sobre el cual se ajustará una elipse para determinar el centro. Se asume que el borde de pupila es curvo y recubre los píxeles más oscuros de la imagen. Para seleccionar dicho borde, primeramente, se filtra la imagen mediante un filtro de bordes de Canny (Figura 16b). Dicho filtrado genera numerosos bordes no significativos. Por tanto, se hace un cribado eliminando las líneas más finas y estrechando las más gruesas por medio de diversos operadores morfológicos. Los bordes restantes son suavizados y aquellos que sean ortogonales unos respecto de los otros se descartan. En base a la ortogonalidad, se establece también un

criterio para determinar si una discontinuidad en un borde debe ser rellenada o se consideran dos bordes independientes (Figura 16c). Para todos los componentes conexos que restan se calcula el punto medio de los mismos. Así, la distancia entre dicho punto y el borde correspondiente se utiliza para determinar si es una línea recta o curva. Si éste está más cerca que cierto umbral, se considera una recta y el borde es eliminado (Figura 16d). Para todas las curvas restantes se selecciona, como estimación del borde de pupila, aquella cuyos píxeles que recubre sean, de media, los más oscuros. Finalmente, al igual que en los algoritmos Starbust y Swirski, se ajusta una elipse a los puntos de la curva seleccionada y se toma como centro de la pupila el centro de dicha elipse (Figura 16e).

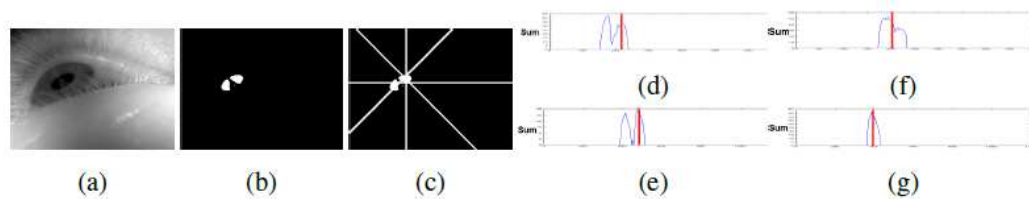


**Figura 16.** Pasos para la estimación de centro con presencia de reflejo

### ***Método por umbralización y refinamiento***

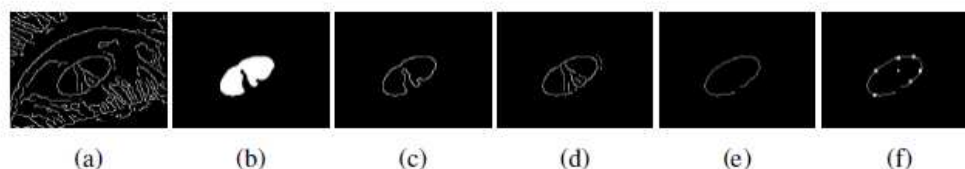
En caso que no haya detectado reflejos en la imagen se realiza una umbralización basada en la desviación estándar y dispersión de los niveles de gris presentes en la misma. Con ella, se genera una imagen binaria en la que se pretende diferenciar la región de la pupila del resto. La posición aproximada de la región se obtiene mediante la Función Integral de Proyección Angular o AIPF (Angular Integral Projection Function) (véase Figura 17). Esta función obtiene el centro ponderado por el nivel de gris para una proyección dada de una imagen. Se obtienen así los puntos medios en las proyecciones a  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$  y  $135^\circ$ , como se muestra en la Figura. A partir de estos puntos medios se calcula la posición de la pupila en base a las trayectorias obtenidas desde dichos puntos con el ángulo correspondiente de cada proyección. Se asume que las trayectorias ortogonales son más propensas a cruzarse cerca del centro de la región. Por tanto, se considera el centro el punto medio entre las intersecciones a  $0^\circ$  y  $90^\circ$  y las intersecciones a  $45^\circ$  y  $135^\circ$ . En caso de no haber cruces entre trayectorias se estima que no se ha detectado la región de la pupila en la umbralización y se registra como una oclusión de pupila o pestañeo.

Una vez obtenida una posición aproximada de la pupila se refina ésta en base a los valores de gris de los píxeles adyacentes. Se establece una región en torno a la posición estimada y se calcula la nueva localización teniendo en cuenta únicamente los valores de gris de los píxeles de la región ya detectada y aquellos que quedan fuera pero con un valor inferior a la media de estos.



**Figura 17.** *Estimación de la posición de la pupila mediante la Función Integral de Proyección Angular. (a) imagen de entrada. (b) umbralización de la imagen. (c) posición aproximada de la pupila en las 4 orientaciones dadas. (d) (f) (e) (g) resultado de la AIPF para las orientaciones a  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$  y  $135^\circ$  respectivamente.*

Finalmente, para estimar el centro de la pupila se calculan los bordes de un recorte de la imagen centrado en el punto obtenido en el paso anterior (Figura 18a). Se asume que la pupila está enteramente dentro de este recorte. Con una nueva imagen umbralizada, obtenida de este recorte, (Figura 18b) se seleccionan los bordes candidatos a contorno de pupila, tomando los puntos de los bordes más cercanos al contorno de la región umbralizada (Figura 18c). Tras realizar el filtrado de bordes del método anterior (Figura 18d y 18e), con el mejor candidato a borde de pupila, se estima el centro de pupila mediante el método de envío de rayos utilizado en el algoritmo Starbust (Figura 9).



**Figura 18.** *Refinamiento del centro de pupila.*

ELSE

El algoritmo ELSE, de reciente publicación (3), es un algoritmo de detección basado en la extracción de bordes, selección de elipses y validación de pupila. Se caracteriza por realizar, tras haber determinado la mejor elipse, una validación para determinar la bondad del ajuste a la pupila y, en caso de no superar un cierto umbral, realizar una nueva detección basada en una estimación aproximada seguida de un refinamiento de la posición. Se ilustra en el esquema de la Figura 19 el diagrama de flujo del algoritmo.

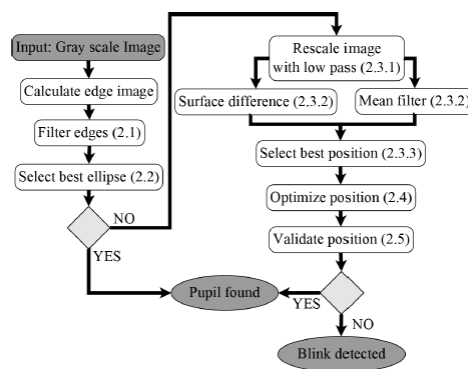


Figura 19 diagrama de flujo del algoritmo ELSE

**Extracción y filtrado de bordes**

Como primer paso se obtienen los bordes de la imagen mediante un detector de Canny para, seguidamente, realizar un filtrado de los mismos (paso 2.1) por métodos de análisis morfológico.

**Selección y evaluación de la mejor elipse**

Antes de realizar la búsqueda de elipses, se eliminan aquellos bordes que se consideren rectos. Para ello se obtiene el punto medio y se observa la distancia de éste hasta el borde correspondiente. Se asume que en una recta, dicho punto debe estar muy próximo a la línea. Con los bordes restantes, se detectan elipses en base al método de mínimos cuadrados. Para seleccionar la mejor elipse se excluyen aquellas que no corresponden al contorno de la pupila

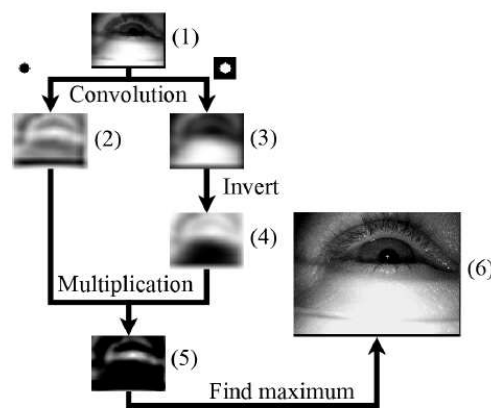
en base al área, forma y valores de gris de los píxeles interiores. Con las elipses restantes se realiza un ranking asignándoles una puntuación en función de los valores de gris interiores y la longitud y anchura de la misma, siguiendo la expresión:

$$elva(el) = gray_{value} * (1 + |el_{ancho} - el_{alto}|)$$

Se considera la mejor elipse aquella que obtenga la menor puntuación.

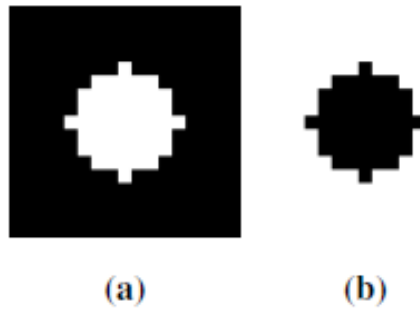
### **Localización aproximada**

En caso de no haberse obtenido ninguna elipse cuya puntuación supere cierto umbral, se procede a estimar la posición de la pupila por el método de la localización aproximada seguido de un refinamiento de centro. Se muestra en la Figura 20 el procedimiento para determinar una aproximación al centro de pupila.



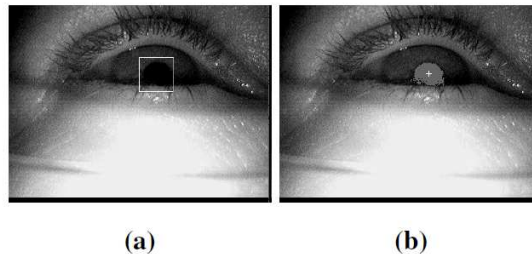
**Figura 20** Proceso de la localización aproximada del centro de pupila. (1) Imagen de entrada (2) convolución con un filtro de media. (3) convolución con un filtro de diferencias entre superficies. (4) Negativo del resultado. (5) Mapa de probabilidad estimada de localización de la pupila. (6) centro marcado como el punto máximo del anterior mapa.

Se realizan dos convoluciones independientes de la imagen por medio de los filtros mostrados en la Figura 21. Ambos filtrados se multiplican y se obtiene un mapa de probabilidad de localización, cuyo máximo se toma como candidato a centro de pupila.



**Figura 21.** (a) filtro de promediado. La suma de la parte blanca suma 1 mientras que la parte negra suma 0. (b) filtro de diferencia de superficie. El centro negro suma -1 y la parte blanca 1.

Seguidamente se realiza un refinado de dicha posición atendiendo a los valores de gris de una región en torno al centro estimado. Así, tras realizar una umbralización del área seleccionada, se calcula el centro de masas de los píxeles más oscuros que se considerará el centro final de la pupila (Véase Figura 22).



**Figura 22.** Refinado de la posición aproximada. (a) Área entorno al centro aproximado (b) en gris píxeles más oscuros de la región y en blanco centro de masas de los mismos.

## Comparativa

Se muestra en la tabla 1 el porcentaje de centros correctamente identificados por cada uno de los algoritmos en todos los conjuntos de imágenes que han sido analizados. Un centro se considera correctamente identificado si el error no es mayor de 5 píxeles (11). Se observa que el rendimiento del algoritmo ELSE es claramente superior al resto siendo más preciso en la detección en 42 de los 47 conjuntos de imágenes. Acorde con estos datos, en la Figura 23 se muestra la tasa media de aciertos para diferentes errores de píxel obtenidos sobre el total de imágenes. Para cualquier nivel de error se observa que el algoritmo ELSE obtiene una tasa de aciertos sensiblemente superior a sus competidores.

**Tabla 1.** Porcentaje de aciertos de los seis algoritmos sobre el total del conjunto de imágenes

	SET (%)	Starburst (%)	Swirski (%)	ExCuSe (%)	ELSE (%)	Pupil Labs (%)
<b>Tübingen</b>						
'I'	10,27	5,48	5,11	70,95	<b>85,41</b>	3,01
'II'	0,00	4,16	26,34	34,26	<b>64,55</b>	1,98
'III'	3,34	1,71	6,81	39,44	<b>63,57</b>	11,63
'IV'	4,03	4,44	34,54	81,58	<b>84,03</b>	3,31
'V'	7,73	14,66	77,85	77,28	<b>84,07</b>	1,83
'VI'	10,30	19,14	19,34	53,18	<b>77,00</b>	7,18
'VII'	1,70	2,41	39,35	46,91	<b>59,08</b>	0,29
'VIII'	7,62	9,52	41,90	56,83	<b>66,19</b>	9,21
'IX'	5,93	13,88	24,09	74,60	<b>86,47</b>	6,36
'X'	22,98	51,07	29,88	79,76	<b>82,38</b>	64,05
'XI'	14,66	27,79	20,31	56,49	<b>75,27</b>	17,40
'XII'	26,72	64,50	71,37	79,20	<b>79,39</b>	74,24
'XIII'	33,40	46,64	61,51	70,26	<b>73,52</b>	32,18
'XIV'	0,21	22,81	53,30	57,57	<b>84,22</b>	76,76
'XV'	26,17	7,71	<b>60,88</b>	52,34	57,58	44,35
'XVI'	26,02	8,93	17,86	49,49	<b>68,11</b>	45,66
'XVII'	0,00	0,75	70,90	77,99	<b>89,93</b>	85,07
'XVIII'	1,32	1,92	12,39	22,24	<b>49,76</b>	2,16
'XIX'	2,68	5,25	9,03	26,45	<b>32,50</b>	5,29
'XX'	2,18	3,73	17,93	52,37	<b>67,65</b>	0,94
'XXI'	0,54	2,41	8,09	<b>43,54</b>	42,61	0,01
'XXII'	0,64	5,91	1,98	27,93	<b>48,66</b>	0,21
'XXIII'	55,43	8,03	96,54	<b>93,86</b>	93,55	15,88
'XXIV'	0,94	1,88	44,43	45,21	<b>52,97</b>	1,04
<b>LPW</b>						
'XXV'	56,86	39,79	84,48	63,53	<b>87,95</b>	65,58
'XXVI'	48,68	19,70	41,58	29,90	<b>69,87</b>	24,72
'XXVII'	27,55	6,75	31,43	34,83	<b>57,50</b>	21,82
'XXVIII'	7,70	9,27	16,87	25,38	<b>37,42</b>	14,53
'XXIX'	6,75	0,00	8,38	19,08	<b>22,95</b>	13,15
'XXX'	11,10	13,30	63,48	53,44	<b>84,10</b>	38,72
'XXXI'	43,55	7,65	66,17	66,48	<b>73,60</b>	68,43
'XXXII'	42,17	34,32	77,68	75,32	<b>81,00</b>	64,22
'XXXIII'	35,65	30,90	56,40	60,42	<b>61,97</b>	45,20
'XXXIV'	10,42	3,65	71,23	59,00	<b>72,65</b>	41,62

'XXXV'	31,07	18,10	31,58	49,52	<b>71,48</b>	9,45
'XXXVI'	54,92	24,10	71,82	72,58	<b>89,73</b>	49,58
'XXXVII'	14,75	16,52	27,03	45,04	<b>51,51</b>	16,68
'XXXVIII'	30,25	23,50	<b>76,07</b>	58,60	70,50	57,52
'XXXIX'	27,17	8,15	37,80	44,83	<b>53,95</b>	43,78
'XXXX'	23,24	17,24	74,11	72,73	<b>82,13</b>	82,01
'XXXXI'	20,90	2,42	68,88	42,10	<b>72,97</b>	47,52
'XXXXII'	50,67	33,48	61,18	66,25	<b>78,57</b>	48,73
'XXXXIII'	11,97	3,45	24,87	21,88	<b>54,05</b>	2,60
'XXXXIV'	19,83	16,58	41,40	11,72	<b>83,35</b>	0,98
'XXXXV'	30,43	25,63	55,90	47,45	<b>88,92</b>	28,18
'XXXXVI'	41,60	11,85	6,48	31,23	<b>70,00</b>	0,83
'Swirski'	23,33	19,33	77,17	<b>86,17</b>	80,83	55,00

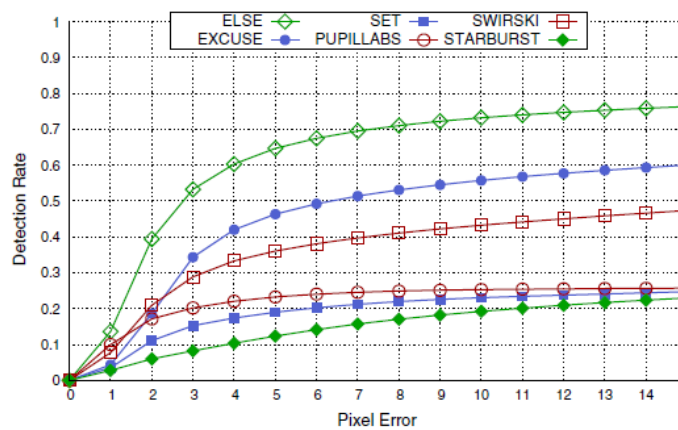


Figura 23 Tasa de error media sobre el total de imágenes analizadas.

## Objetivos

A tenor de expuesto en los anteriores apartados se definen los siguientes objetivos para este trabajo:

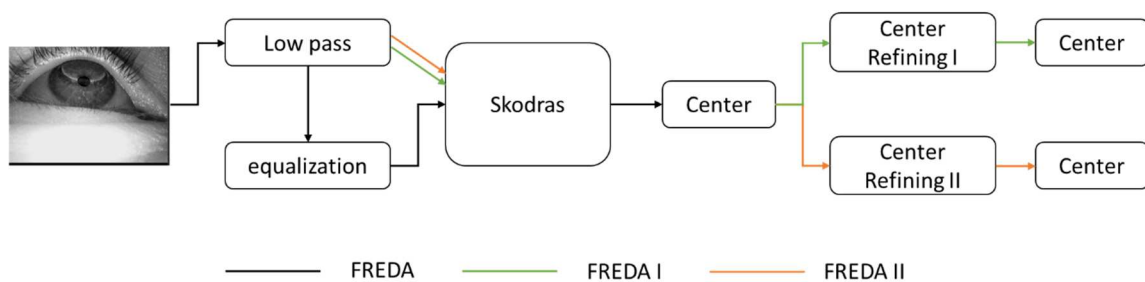
- Desarrollar un método rápido, robusto y eficaz para la detección del centro de pupila en entornos reales.
- Evaluar la precisión del algoritmo diseñado sobre una colección marcada de imágenes oculares obtenidas en diferentes escenarios realistas.
- Realizar una comparativa en cuanto a rendimiento frente a un algoritmo de referencia en la literatura.



# CAPÍTULO 3 MÉTODOS

## Algoritmos

Se proponen tres algoritmos para la detección del centro de pupila basados en el método presentado por Skodras et al. (24) que, a su vez, hace uso del procedimiento denominado Fast Radial Symmetry Transform o FRST (23), para estimar la posición del centro de pupila. La Figura 24 ilustra el diagrama de bloques de los tres algoritmos, denominados FREDA (Fast Robust Ellipse Detection Algorithm), FREDA I y FREDA II. Se observa como el primer algoritmo, FREDA, devuelve el centro dado por el método Skodras al que previamente se ha añadido una etapa de pre procesamiento de imagen que incluye un filtro paso bajo y una ecualización adaptativa. Las otras dos versiones, FREDA I y FREDA II, añaden una etapa más de post procesamiento una vez obtenido el centro mediante Skodras. El objetivo es mejorar la estimación en caso de pupilas elípticas. Se describen a continuación en detalle los tres algoritmos además del método de Skodras y la FRST.

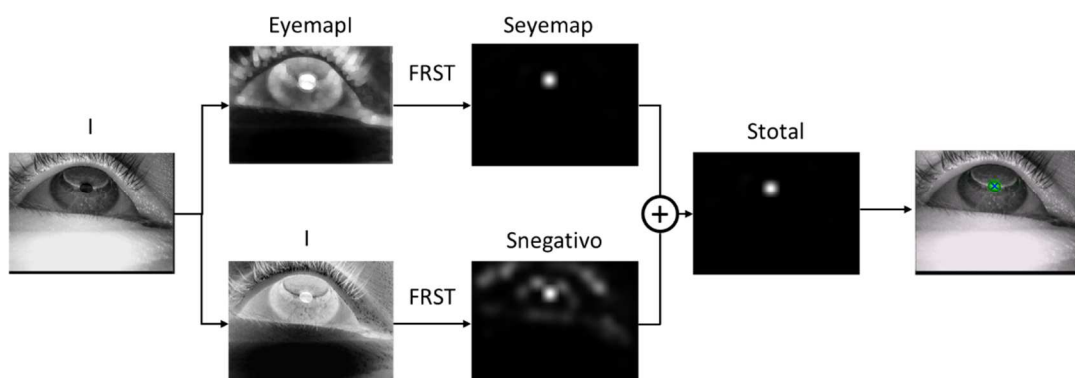


**Figura 24.** Diagrama de bloques de los tres algoritmos propuestos

SKODRAS

El método propuesto por Skodras et al. tiene como objetivo determinar el centro del iris en imágenes de baja resolución como las obtenidas mediante una webcam convencional (24). En este proyecto se modifica el algoritmo original para adaptarlo a las características de las imágenes utilizadas. Se ilustran en la Figura 25 los pasos seguidos para determinar el centro de pupila mediante el método de Skodras modificado.

El procedimiento original, al tratar con imágenes de rostro completo, realiza un reconocimiento facial en el que extrae las zonas de ambos ojos. Dado que en este trabajo se tratan imágenes oculares directamente, se excluye esta parte del algoritmo. Posteriormente, para cada región ocular, se realiza un realce de la zona de la pupila haciendo uso de la información de color. A este realce se le denomina mapa de luminancia (Eyemapl). Este procedimiento se modifica para adaptarlo a las imágenes en escala de grises utilizadas en este proyecto. Así, se genera un mapa de luminancia que resalta la zona de la pupila utilizando únicamente la información los valores de grises. Tanto a dicho mapa como al negativo de la imagen se les aplica de manera paralela la transformada de simetría radial o FRST. El resultado de ambas transformadas (Seyemap y Snegativo) es un mapa donde los valores más altos indican una mayor probabilidad de detección del centro de pupila. Finalmente se suman los valores de ambas transformadas (Stotal) y se obtiene el centro tomando las coordenadas de imagen del valor más alto del mapa resultante



**Figura 25.** Diagrama de los pasos para obtener el centro de pupila mediante el método de Skodras modificado y la FRST.

### **Construcción del mapa de luminancia (Eyemapl)**

Mediante el mapa de luminancia (Eyemapl) se consigue realzar la zona de la pupila facilitando que la transformada FRST posterior estime correctamente el centro. Los pasos que se siguen para obtener el mapa se muestran en la Figura 26. A la imagen de entrada en escala de grises ( $I$ ) se le aplican, de manera independiente, una erosión morfológica y una transformación de contraste ( $I'$ ) que permite resaltar las zonas oscuras de la imagen. Posteriormente, a esta transformación se le aplica una dilatación morfológica. Tanto la dilatación como la erosión realzan la apariencia de la pupila haciéndola más brillante en el caso de la imagen transformada ( $I'$ ) y oscureciendo la región en la imagen original ( $I$ ). Además, la forma circular de los elementos estructurales hace destacar las formas redondas de la imagen, facilitando así la estimación del centro mediante la transformada de simetría radial. Finalmente, para resaltar aún más la pupila, se divide la imagen dilatada por la erosionada logrando así el mapa de luminancia deseado (EyeMapl), de la siguiente manera:

$$EyeMapl = \frac{I' \oplus B1}{I \ominus B2 + \varepsilon}$$

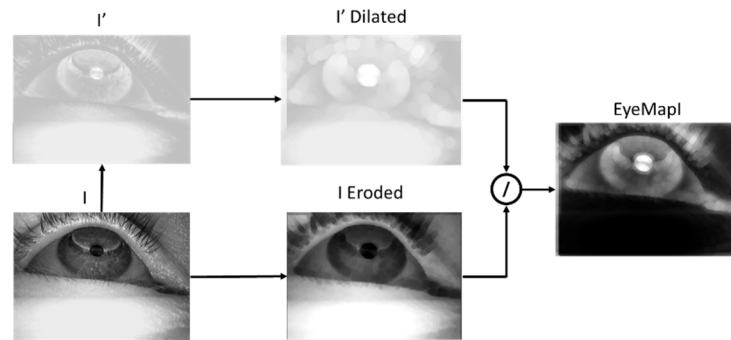
Donde  $\oplus$  y  $\ominus$  denotan las operaciones de dilatación y erosión respectivamente y,  $B1$  y  $B2$  representan los elementos estructurantes, en este caso, discos planos de radio:

$$Radio\ B1 = ancho\ de\ imagen/20;$$

$$Radio\ B2 = Radio\ B1/2;$$

Los radios para ambas operaciones no varían entre diferentes usuarios puesto que entre los humanos existe poca variación en el tamaño del iris (27). Además, estas pequeñas variaciones no afectan de manera perceptible al resultado. Se añade un factor  $\varepsilon$  para evitar que, debido a valores nulos o demasiado pequeños en el denominador, se obtengan valores excesivamente altos en la imagen resultante. Se escoge para  $\varepsilon$  un valor de:

$$\varepsilon = mean(I \ominus B2)$$



**Figura 26.** Diagrama de pasos para obtener el mapa de luminancia (EyeMapI).

### ***Fast Radial Symmetry Transform - FRST***

La transformada FRST mide la contribución de cada píxel a la simetría radial de los píxeles que lo rodean, los cuales pertenecen a los bordes de la imagen. Aplicado a la detección de pupila, se asume que el punto central debe aparecer como un máximo en la transformada puesto que está rodeado de los píxeles del borde de pupila que, de manera aproximada, forman una circunferencia en torno a él. Así, para determinar el punto de mayor simetría radial partiendo de los píxeles de borde de una imagen, se realiza un proceso de votación de puntos en dirección del gradiente de la imagen en cada uno de los puntos. El método utilizado es una modificación del propuesto por Loy et al. (23). La transformación se lleva a cabo para un conjunto de radios  $n \in N$ . Para las imágenes de este proyecto, el conjunto  $N$  de radios escogido es el compuesto por los números pares entre 20 y 30. La razón para no tomar todos los radios entre el máximo y el mínimo es la optimizar el tiempo de cómputo. El valor de la transformada para un radio  $n$  indica la contribución a la simetría radial de los gradientes a una distancia  $n$  de cada punto.

Primeramente, se calcula el gradiente  $g$  de la imagen mediante un operador de Sobel 3x3. De éste se consideran únicamente los valores de gradiente significativos, es decir, mayores que cierto umbral determinado. Para este trabajo se ha escogido empíricamente este umbral como el 5% del valor máximo de magnitud del gradiente obtenido en cada imagen. Con ello se reduce el número de puntos a considerar para la transformada. Para cada punto significativo  $p$  de gradiente, se define el píxel afectado  $p_{af}$  como aquel punto situado a una distancia  $n$  de  $p$  y al que apunta el gradiente en el punto  $p$ .

$$p_{af} = p + \text{round}\left(\frac{g(p)}{\|g(p)\|}n\right)$$

Donde *round* indica el redondeo al entero más cercano. Nótese que se buscan, únicamente, los píxeles afectados en la dirección en la que apunta el gradiente. Este punto supone una modificación respecto al método presentado originalmente por Loy et al., en la que se tienen en cuenta tanto la dirección positiva del gradiente como su contraria. Dado que el gradiente toma la dirección de las zonas oscuras a las claras, con esta modificación se consigue excluir de la transformada a los puntos situados en zonas oscuras de la imagen. Teniendo en cuenta que la imagen a transformar ha sufrido una inversión en la escala de grises (véase Figura 25), ello supone restringir la búsqueda del centro de pupila únicamente a las zonas más claras de la imagen evitando que otras zonas con alta simetría radial en la imagen (por ejemplo reflejos) puedan interferir en el resultado.

Para cada radio  $n$ , se crea una imagen de proyección de la orientación  $O_n$  y una imagen de proyección de magnitud  $M_n$  a partir de los píxeles afectados  $p_{af}$ . De la siguiente manera:

$$O_n(p_{af}) = O_n(p_{af}) + 1 \quad (1)$$

$$M_n(p_{af}) = M_n(p_{af}) + \|g(p)\| \quad (2)$$

La primera es una matriz de votación en la que se registra el gradiente de cuantos píxeles apuntan a cada pixel de la imagen de la manera definida en (1). La segunda, sin embargo, es una matriz de votación ponderada por la magnitud del gradiente en el punto que afecta al píxel (2).

La contribución de la simetría radial para un radio  $n$  se obtiene combinando ambas matrices y convolucionándolas con una máscara gaussiana  $A_n$  con media  $\mu = 2n$  y de la siguiente manera:

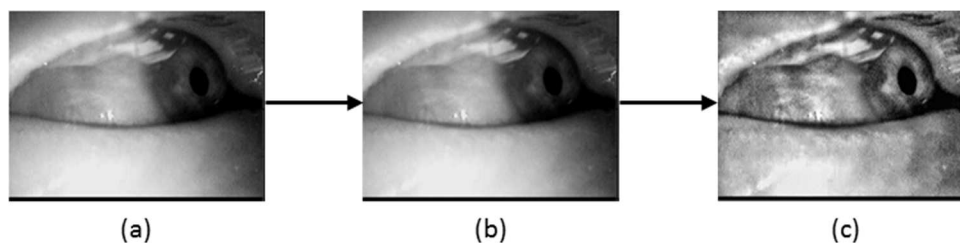
$$S_n = (M_n \cdot O_n^\alpha) * A_n$$

El parámetro  $\alpha$  se define como la “rigidez” radial y se establece en 2. El efecto de convolucionar las matrices de votación es la de distribuir la influencia de los píxeles afectados, resultando en un mapa de votación  $S_n$  con variaciones suaves. Se obtienen los mapas de votación para cada radio  $n$  y, de todos ellos, se toma como resultado final de la transformada aquel  $S_n$  que contenga el pico más alto (mayor contribución a la simetría

radial). El valor  $n$  con el que se ha obtenido dicho mapa de votación se considera la mejor estimación del radio de la pupila.

#### FREDA

El primer algoritmo propuesto se basa en la aplicación del método Skodras a una imagen ocular previamente procesada que consta de dos etapas: un filtrado gaussiano paso bajo y una ecualización adaptativa. La Figura 27 muestra el resultado de este procesado sobre una imagen de entrada.



**Figura 27.** *Procesado de la imagen previo al método de Skodras. (a) Imagen de entrada (b) imagen filtrada mediante una filtro gaussiano paso bajo (c) imagen ecualizada.*

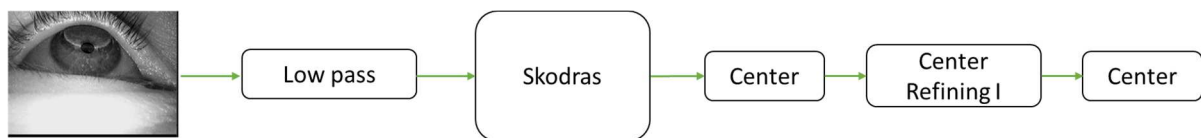
El filtrado paso bajo se realiza convolucionando la imagen con una máscara gaussiana de tamaño  $3 \times 3$ . El objetivo es reducir el ruido de la imagen para la posterior detección de bordes mediante gradiente al aplicar la transformada FRST.

Seguidamente se lleva a cabo una ecualización adaptativa de la imagen con el fin de aumentar el contraste de la pupila respecto del fondo y así obtener bordes de pupila más definidos para aplicar la transformada FRST. El procedimiento ecualiza el histograma de pequeñas regiones de la imagen en lugar de hacerlo sobre la imagen completa. Se escoge una subdivisión de 10 columnas y 10 filas resultando en un total de 100 regiones rectangulares sobre las que se aplica la ecualización. El histograma de salida de cada región se aproxima a una distribución uniforme. Para eliminar efectos de borde entre regiones adyacentes, éstas se combinan utilizando una interpolación bilineal. Para evitar que el ruido aumente en zonas uniformes de la imagen, el contraste se limita hasta cierto umbral que se

ha escogido empíricamente con un valor de 0.01. Finalmente, a la imagen resultante se le aplica el método Skodras modificado anteriormente descrito para determinar el centro de la pupila.

#### FREDA I

Se ilustra en la Figura 28 el diagrama de flujo del algoritmo. Como se puede observar, la principal diferencia respecto a FREDA es la etapa adicional tras el cálculo del centro mediante Skodras, añadida con el fin de mejorar la precisión en pupilas de apariencia elíptica. Tal y como muestra la Figura 29, para este tipo de imágenes, la transformada de simetría radial o FRST no es apta para determinar el centro de pupila puesto que tiende a marcar el centro cerca de algún foco de la elipse. Se justifica así la etapa posterior del algoritmo.



**Figura 28.** Diagrama de bloques del algoritmo FREDA I.

La idea de la etapa de refinado es tomar el centro de pupila estimado por Skodras como un punto interior de la pupila, desde el cual buscar el centro de manera más precisa. Este método de búsqueda está basado en un *Region Growing* utilizando el punto indicado como origen. Así, tras sucesivas iteraciones, el centro se toma como el centro de la elipse que mejor se ajusta al borde de la región en ese instante y cuya área sea la más parecida a la de la elipse calculada. Es decir, cuando la diferencia en áreas de la región y de la elipse que mejor se ajusta a ella sea mínima, se toma el centro de dicha elipse como estimación del centro de pupila. Este procedimiento se justifica bajo la hipótesis de que una vez la región haya aumentado lo suficiente como para abarcar toda la pupila, ésta tendrá una forma elíptica, con lo cual, la elipse de ajuste debe coincidir tanto con el borde de la región como con el de la pupila real. En este caso las áreas de la elipse y la región deben de ser idénticas. Se describe a continuación este procedimiento en detalle.



**Figura 29.** Ejemplo de centro obtenido únicamente mediante la FRST en una imagen con pupila de apariencia elíptica. Se resalta en verde el mapa de votación obtenido y el centro mediante la cruz roja.

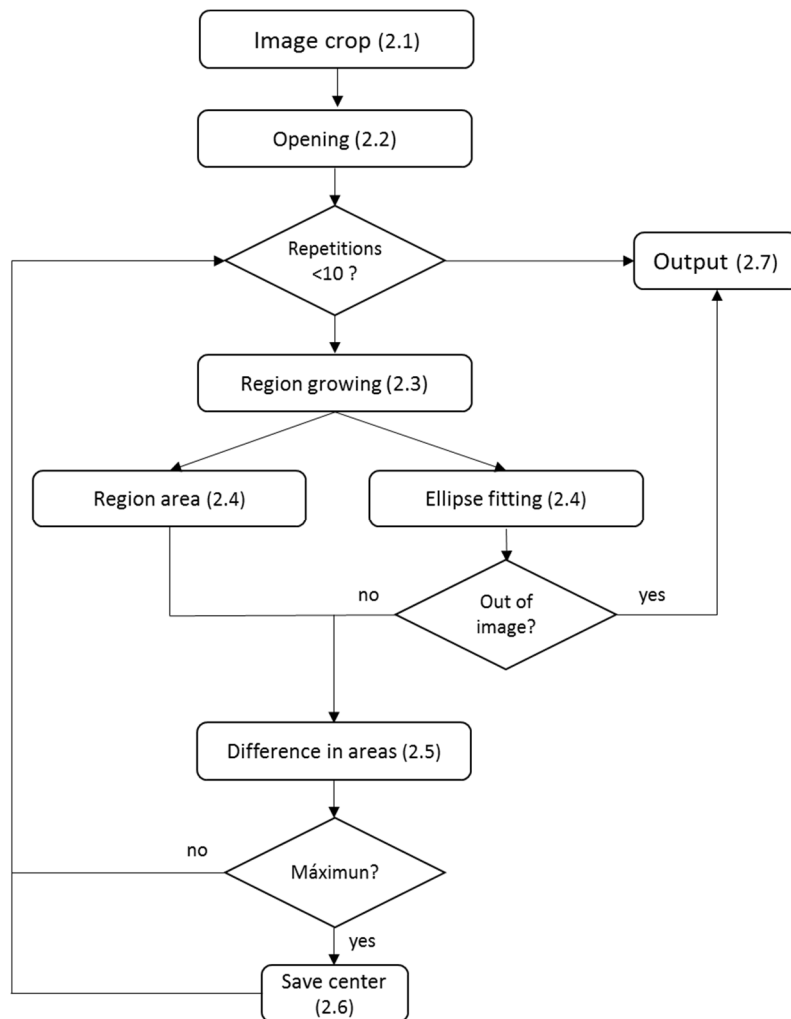
### ***Refinado de centro por Region Growing***

Primeramente (paso 2.1) se recorta la imagen centrando la región de interés en el punto obtenido mediante Skodras. Las dimensiones del recorte se escogen de manera adaptativa al tamaño estimado de pupila. El lado del recuadro es 6 veces el radio para el cual se ha obtenido el valor máximo en los mapas de votación de la FRST, con lo que es un buen estimador del radio de la misma. Seguidamente (paso 2.2), con el fin de limpiar la pupila de imperfecciones, se realiza una apertura morfológica. El elemento estructurante es un disco plano cuyo radio se escoge también en función del radio obtenido en la FRST. Se escoge un radio de disco cuyo valor es el entero superior más cercano a la mitad del radio estimado de pupila. Con la apertura se pretenden eliminar las zonas claras de la pupila debidas a pestañas o reflejos siempre que sean menores al elemento estructurante. Con la región de la pupila procesada y limpia de artefactos comienza el bucle mediante el cual se busca el centro preciso de la misma (Paso 2.3).

La región crece partiendo del centro obtenido mediante la FRST Skodras. Se añaden aquellos píxeles vecinos cuya diferencia en valor de gris respecto de la media de la región no excede cierto umbral. Este umbral se escoge inicialmente en 5. Cada iteración, este umbral es multiplicado por 1.1 para que en cada repetición la región sea mayor que la anterior. Una vez se ha obtenido una región, se calcula su área y el de la elipse (paso 2.4) que tiene el mismo segundo momento central que la región. Seguidamente (paso 2.5) se obtiene el parámetro para determinar el centro óptimo de pupila como la diferencia de áreas normalizada respecto

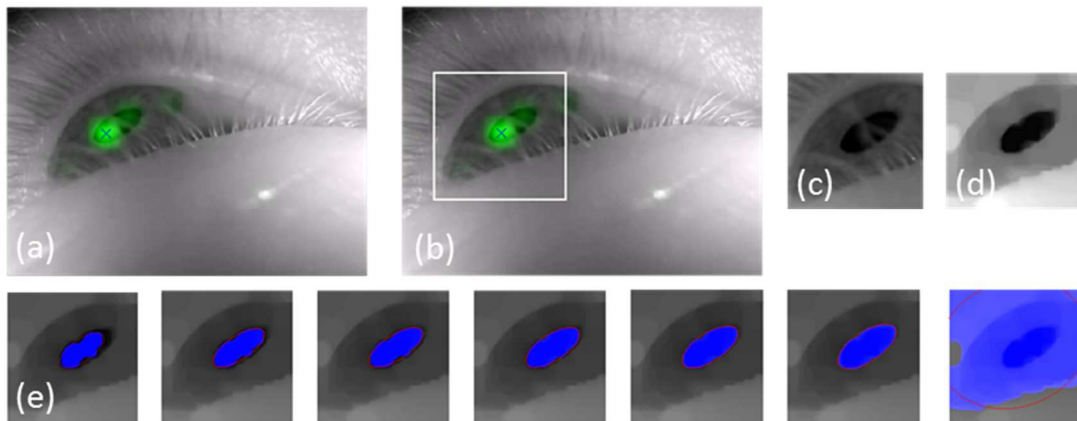


del área de la elipse. En caso de que éste valor sea inferior al mínimo guardado, se sustituye y se guarda el centro de la elipse actual como centro de la pupila (paso 2.6). Se escoge empíricamente 0.01 como valor inicial de este mínimo para la primera iteración. Con ello se consigue que crecimientos de región muy irregulares, en los que la elipse de ajuste difiere mucho del borde de la región, no sean tenidos en cuenta a la hora de estimar el centro. Es decir, prefijando un valor inicial a superar, se asegura que, si un centro es guardado, la relación entre la elipse de ajuste y la región de crecimiento será mínimamente aceptable. El bucle se repite un máximo de 10 veces siempre que la elipse de ajuste calculada en el paso 2.4 no se salga del recorte de imagen realizado en el paso 1 (paso 2.7). En tal caso, el centro escogido es el último guardado antes del último crecimiento de región.

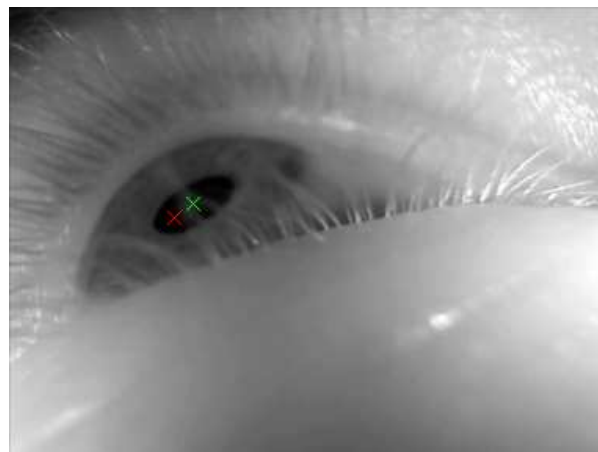


**Figura 30.** Esquema de los pasos para el refinamiento de centro del algoritmo FREDA I.

En la Figura 31 se muestra un ejemplo de aplicación del refinamiento de centro sobre una imagen con pupila de apariencia elíptica. En (a) se representa el mapa de votación obtenido mediante la FRST (verde) superpuesta a la imagen original, así como el centro estimado a partir de dicho mapa. En (b) se marca el recuadro de recorte de imagen (paso 2.1) centrado en el centro de pupila marcado anteriormente. Dicho recorte se representa por separado en la Figura 31 (c). A esta zona de pupila se le aplican la transformación morfológica de apertura (paso 2.2). El resultado se muestra en (d), donde se puede apreciar que se han eliminado los artefactos de la zona de la pupila. En (e) se representan las sucesivas repeticiones entre los pasos 2.3 y 2.6. En azul se dibuja la región de crecimiento a partir del punto semilla marcado en la Figura 31 (a). En rojo se ilustra la elipse de ajuste a dicha región de crecimiento. Se observa cómo, a medida que esta región se ajusta mejor a la zona de la pupila, la elipse circundante se aproxima mejor tanto al borde de la región como al de la pupila. El último fotograma de la secuencia muestra el momento de parada del procedimiento, cuando el crecimiento de la región sobrepasa la zona recortada y, en consecuencia, la elipse de ajuste sobresale también de dicha zona (paso 2.7). El centro se obtendrá, del fotograma en el que la diferencia normalizada entre el área de la elipse de ajuste y la región de crecimiento sea mínima. Se muestra en la Figura 32 el centro obtenido inicialmente mediante la aplicación de la transformada FRST y el centro calculado tras emplear el procedimiento de refinado de centro.



**Figura 31.** Ejemplo de funcionamiento del proceso de refinamiento de centro del algoritmo FREDA I. (a) Imagen con pupila de apariencia elíptica y centro mal marcado por la FRST. (b) y (c) Recorte de la zona de la pupila (paso 2.1). (d) Aplicación de la apertura (paso 2.2). (e) Sucesivas iteraciones de crecimiento de región (en azul) y ajuste de elipse (en rojo) (pasos 2.3 -2.7).



**Figura 32.** Centro obtenido mediante la FRST (rojo) y centro resultante de aplicar el proceso de refinamiento (verde).

## FREDA II

Al igual que en algoritmo FREDA I, FREDA II es una versión de FREDA a la que se le añade una etapa tras el cálculo del centro mediante Skodras con el fin de adaptar el método a elipses de apariencia elíptica. Como se aprecia en la Figura 33, el algoritmo sigue el mismo diagrama de flujo ilustrado en la Figura 28. La diferencia con el anterior radica en la manera en la que se refina el centro, aunque ambos métodos parten de la misma idea: realizar sucesivos crecimientos de región hasta encontrar la que mejor se adapte a la pupila, tras lo cual, se

obtiene el centro de la elipse que mejor se ajusta a dicha región. Ambos algoritmos difieren por tanto en la manera en la que estiman la región y, en consecuencia, la elipse que mejor se ajuste a la pupila.



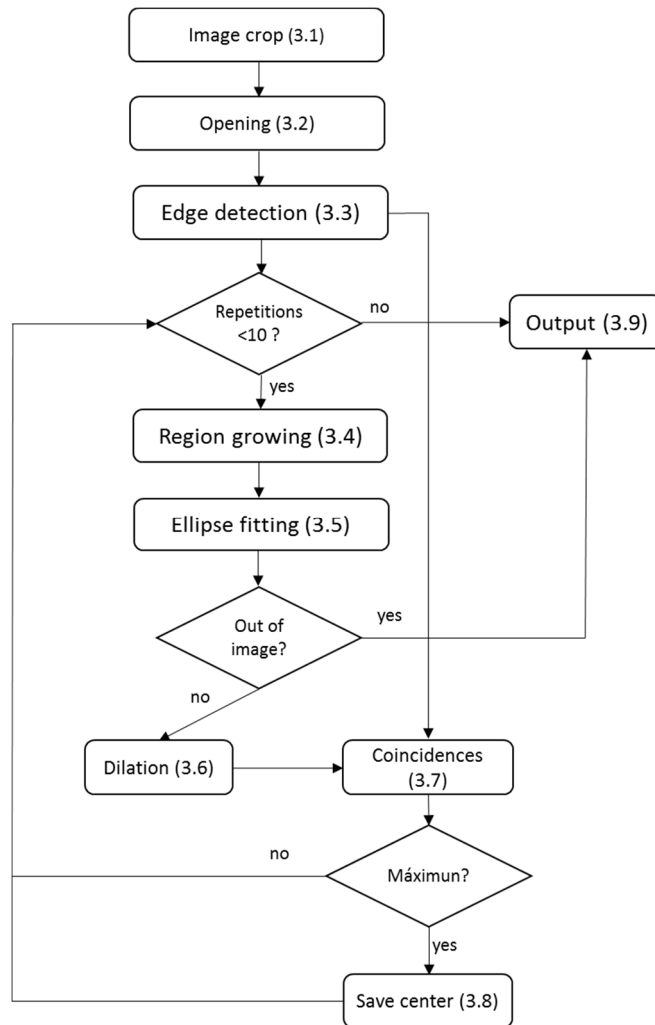
**Figura 33.** Diagrama de bloques del algoritmo FREDA II.

FREDA II no toma la diferencia en áreas como parámetro para decidir la bondad del ajuste. Por el contrario, esta se estima cuando la elipse de ajuste de la región creciente obtiene un máximo de coincidencias con el borde de pupila previamente calculado. Se muestra en la Figura 34 el diagrama de bloques del método de refinado de centro de FREDA II. A continuación, se detallan los pasos del método.

### ***Refinado de centro por Region Growing y detección de bordes***

En primer lugar (paso 3.1) se realiza un recorte de la imagen con una longitud de lado de 6 veces el radio con el que se ha obtenido el centro en la FRST. A continuación (paso 3.2) se realiza una apertura con el mismo elemento estructurante utilizado en FREDA I (paso 2.2). Se obtiene así una región de pupila libre de artefactos que impidan un crecimiento de región adecuado. En el siguiente paso (paso 3.3) se detectan los bordes de la pupila mediante el método de Canny. Tomando el centro dado por Skodras se realiza un crecimiento de región (paso 3.4) utilizando los mismos parámetros que en FREDA I (paso 2.3). Una vez se tiene la región se calcula la elipse que mejor se ajusta tomando aquella elipse con el mismo segundo momento central que la región (paso 3.5). Esta elipse va a ser utilizada para buscar las coincidencias con el borde de pupila detectado en el paso 3.3. Sin embargo, dada la dificultad para que dos bordes de 1 píxel de anchura coincidan, ésta es dilatada mediante un elemento estructurante de tamaño 3x3 (paso 3.6). Una vez obtenida la elipse de ajuste ensanchada, se buscan las coincidencias con el borde de la pupila mediante un operador “and” (paso 3.7).

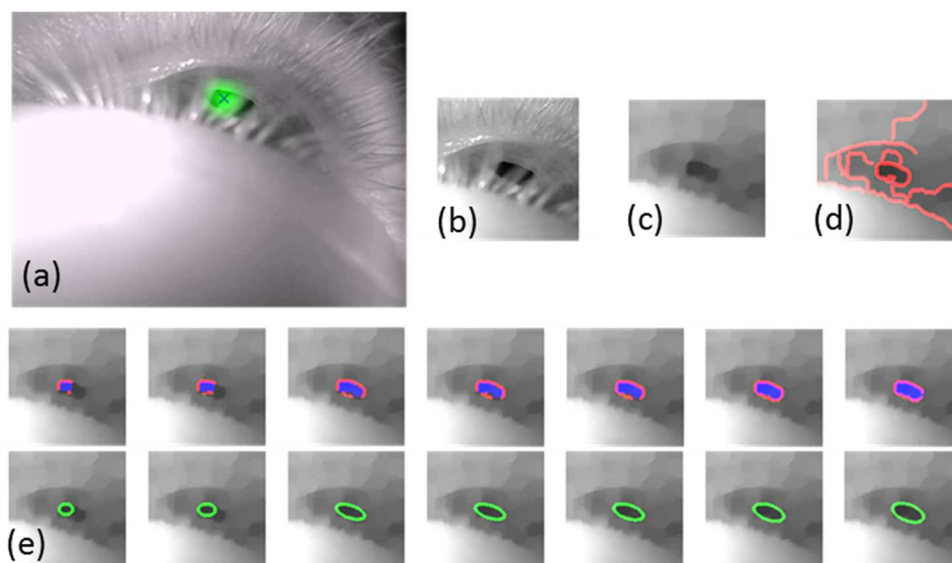
Se cuentan cuantos píxeles en común tienen la elipse y el borde de pupila y en caso de superarse el máximo anterior se guarda el centro de la elipse como nuevo centro estimado de la pupila (paso 3.8). Se vuelven a repetir los pasos del 3.4 al 3.8 hasta que se hayan completado 10 ciclos o la elipse de ajuste sobresalga del recorte realizado en 3.1. En tal caso la salida del algoritmo es el último centro guardado (paso 3.9).



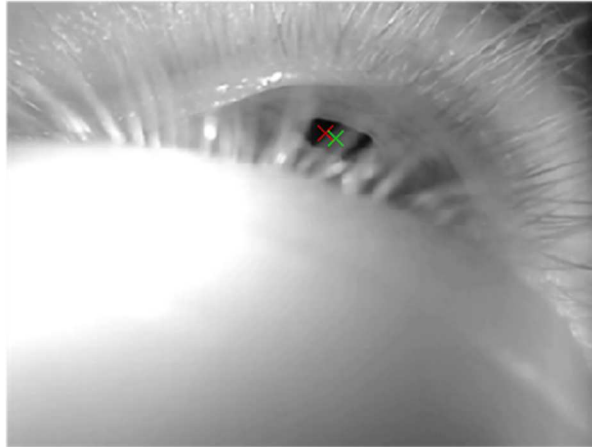
**Figura 34.** Esquema de los pasos para el refinamiento de centro del algoritmo FREDA II.

Al igual que en algoritmo FREDA I, se muestra en la Figura 35 un ejemplo de la secuencia de pasos del proceso de refinamiento de centro arriba descrito. En (a) se representa, en verde, el mapa de votación obtenido mediante la transformada FRST y el centro estimado a partir de dicho mapa. La Figura 35 (b) muestra el recorte realizado en el paso (3.1). Se aplican posteriormente la transformación morfológica de apertura (paso 3.2). Se observa en (c) el resultado de dicha transformación. En la Figura 35 (d) se superponen en rojo los bordes

detectados mediante el procedimiento de Canny (paso 3.3) sobre la imagen obtenida en el paso anterior. Cada par de imágenes (arriba y abajo) en la Figura 35 (e) muestran el resultado de una repetición de los pasos entre el 3.4 y 3.8. En la imagen superior se observa la región de crecimiento (en azul) junto con los píxeles coincidentes (rojo) entre los bordes detectados en el paso 3.2 (Figura 35 d) y la elipse de ajuste de la región que se representa en verde en la imagen inferior. Se aprecia claramente como a medida que la región va en aumento, la elipse de ajuste crece con ella y, los píxeles en común entre los bordes detectados y dicha elipse forman un borde de pupila cada vez más definido. Así, se toma como centro de la pupila, el centro de la elipse cuyas coincidencias con los bordes detectados sean máximas. En este ejemplo se observa en la última iteración la pérdida de varios píxeles en la parte derecha de la pupila. Por tanto, el centro se tomará de la elipse de la iteración anterior donde el borde queda completo. El criterio de parada, al igual que en el algoritmo Freda I, es la salida de la elipse de ajuste del recorte de la imagen por un crecimiento excesivo de ésta. En la Figura 36 se observa el centro corregido mediante el proceso de refinamiento.



**Figura 35.** Ejemplo de funcionamiento del proceso de refinamiento de centro del algoritmo FREDa II. (a) Imagen con pupila de apariencia elíptica y centro mal marcado por la FRST. (b) Recorte de la zona de la pupila (paso 3.1). (c) Aplicación de la apertura (paso 3.2). (d) Bordes detectados mediante el método de Canny (paso 3.3). (e) Sucesivas iteraciones de crecimiento de región (en azul) y coincidencias (rojo) entre la elipse de ajuste (verde) y los bordes detectados (pasos 3.4 -3.9).



**Figura 36.** Centro obtenido mediante la FRST (rojo) y centro resultante de aplicar el proceso de refinamiento (verde).

## Colecciones de imágenes

Para la evaluación de los algoritmos se han utilizado tres colecciones de imágenes oculares de acceso público que suman un total de 225.569 imágenes. Con cada colección de imágenes se adjuntan las coordenadas de imagen del centro de pupila, las cuales se utilizan como referencia para la evaluación de la precisión de los algoritmos.

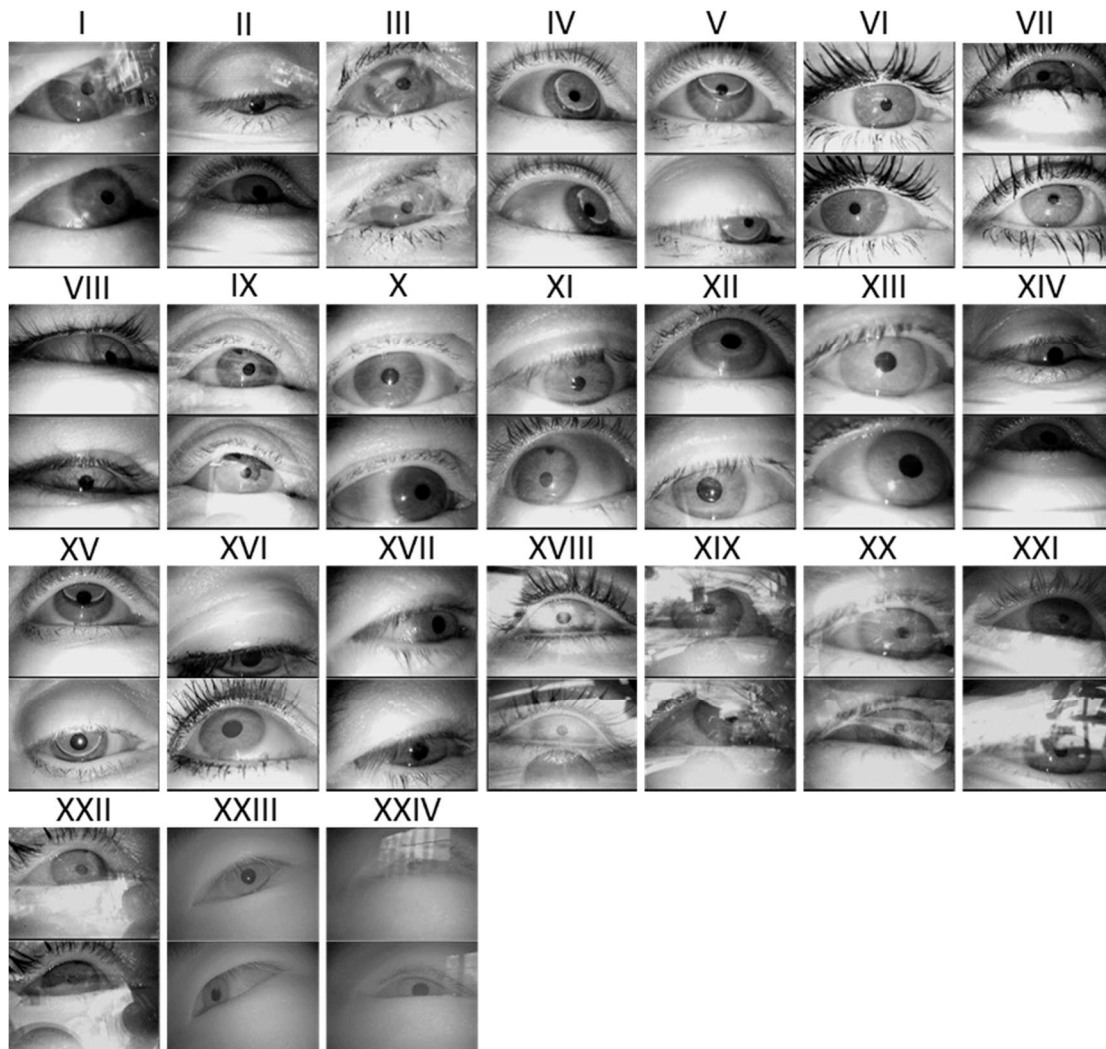
### *Tübingen*

Esta colección de imágenes fue publicada por Fuhl et al. (3) para la evaluación de los algoritmos ExCuSe (16) y ELSE (3). Está compuesta por un total de 94.113 imágenes oculares de 384 x 288 píxeles divididas en 24 sets correspondientes a 24 sujetos diferentes, de los cuales, los primeros 17 corresponden a la publicación del algoritmo ExCuSE y los restantes 7 fueron presentados junto con el algoritmo ELSE. Los sets I – IX fueron obtenidos en un experimento de conducción en carretera mediante el sistema Dikablis de seguimiento ocular (Ergoneers Inc., Manching, Germany), mientras que los sets X – XVII fueron registrados durante la búsqueda de productos en un supermercado (28) para lo que se utilizó el mismo sistema de registro de imágenes. Dado que en este último caso el objeto de estudio era la pérdida visual por glaucoma, las imágenes presentan numerosos reflejos debido a las gafas que portan los sujetos. Ello, unido a los frecuentes cambios en las condiciones de iluminación, hace que en estos últimos sets sea especialmente difícil determinar el centro de pupila. Se muestran en la Figura 37 dos imágenes de cada uno de los 24 sets de la colección.

Los sets de imágenes entre el XVIII y XIV, correspondientes al estudio de ELSE, presentan numerosos artefactos que dificultan en gran medida la estimación del centro de pupila. Los sets entre el XVIII y el XXII fueron registrados durante la conducción por carretera y se caracterizan por un alto nivel de emborronamiento por movimiento, reflejos y un bajo contraste de la pupila en comparación con el área circundante. Los sets XXIII y XXIV, sin embargo, fueron registrados a sujetos asiáticos en los que la principal dificultad radica en las oclusiones de pupila causadas por las sombras de párpados y pestañas. A ello hay que añadir



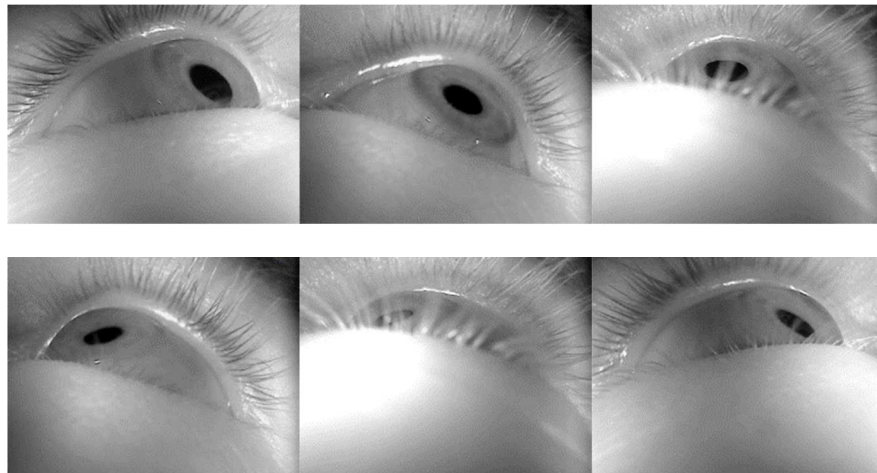
las reflexiones de gafas que porta el sujeto XXIV. El marcado de las imágenes se realizó de forma manual y el error puede llegar a ser a priori de hasta 5 píxeles (3).



**Figura 37.** Ejemplos de cada uno de los 24 set de imágenes de la colección de Tübingen donde se aprecian las dificultades para determinar el centro de pupila. Cada par de imágenes corresponde a un sujeto de estudio.

### Swirski

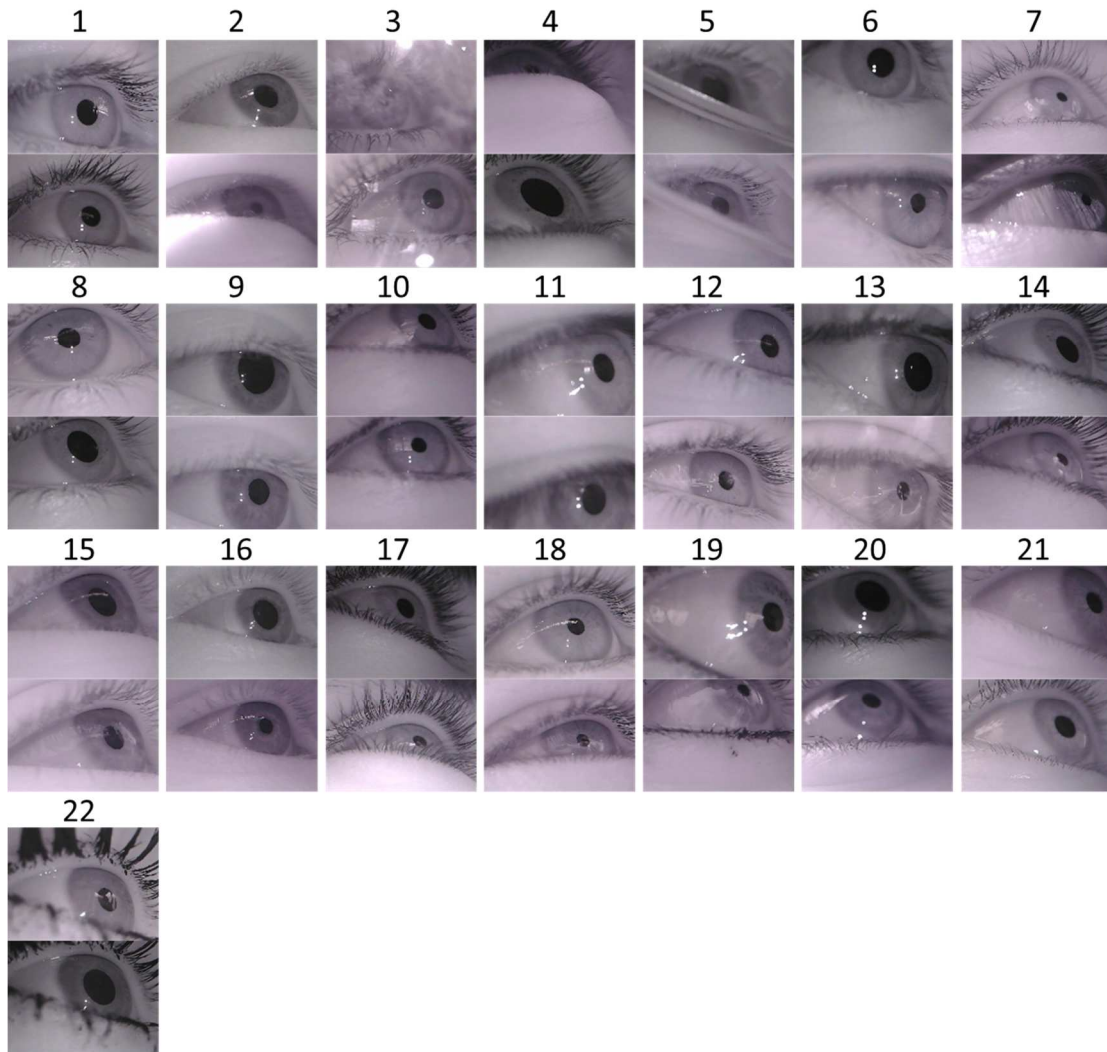
Este conjunto de imágenes fue publicado por Swirski et al. (22) y contiene 600 imágenes de alta calidad (640 x 480 píxeles) correspondientes a ambos ojos de 2 sujetos diferentes. Las imágenes fueron obtenidas mediante un sistema *head-mounted* de bajo coste con iluminación infrarroja y en condiciones de laboratorio. Destaca la buena calidad de las imágenes sobre todo por la ausencia de reflejos y el contraste de la pupila sobre el resto del ojo. La principal dificultad en la detección del centro de pupila está en la excentricidad de las pupilas debido al alto grado de angulación de la cámara respecto del eje de visión. Este hecho hace que aparezcan también oclusiones de la pupila por parte de las pestañas o párpados (véase Figura 38). El marcado del centro de pupila se llevó a cabo ajustando una elipse respecto a, por lo menos, 5 puntos manualmente colocados sobre el borde de la pupila. El resultado es un marcado de muy alta fiabilidad.



**Figura 38.** Ejemplos de imágenes de la colección de Swirski. Destaca el alto contraste entre la pupila y el resto de la imagen. Se aprecia también la elevada excentricidad en la apariencia de las pupilas.

### *Labeled pupils in the wild (LPW)*

El conjunto de imágenes denominado “Labeled Pupils in the Wild” o LPW publicado por Tonsen et al. (25) proviene de un total de 66 vídeos de alta calidad de 22 sujetos diferentes. Cada vídeo contiene unos 2.000 fotogramas de 640 x 480 píxeles, obtenidos a una frecuencia de imagen de 95 FPS teniendo un total de 130.856 fotogramas oculares. La colección abarca un amplio rango de situaciones tanto en pruebas de exterior como de interior. Cada usuario es grabado en dos localizaciones de interior y una de exterior. La iluminación varía también entre grabaciones añadiéndose en ocasiones fuentes artificiales. Una dificultad añadida es la alta excentricidad de pupila que presentan ciertas imágenes. Dado que el sujeto debe seguir con la mirada una pelota movida por el experimentador, para las posiciones más extremas, el eje de la pupila se desalinea respecto del eje de dirección de la cámara, mostrándose la pupila con una excentricidad muy elevada. El marcado de los centros es de muy alta precisión. Para ello se han utilizado diversos métodos los cuales incluyen técnicas tanto manuales como automáticas corrigiendo posteriormente posibles errores en la detección (25).

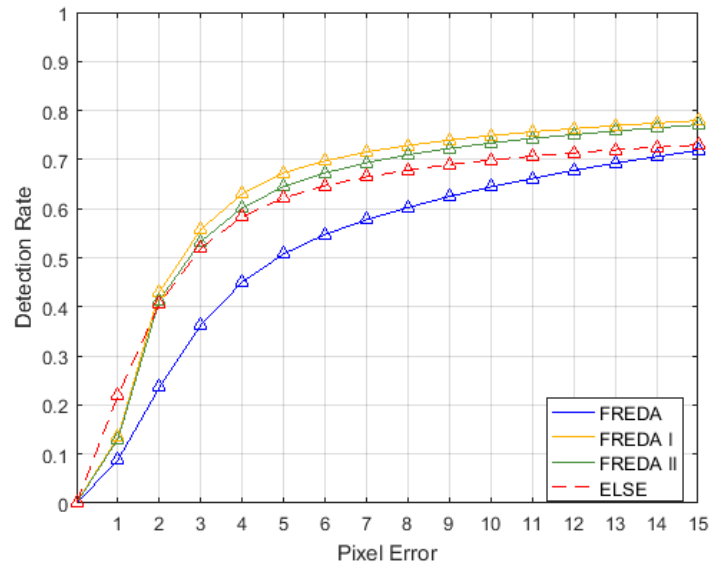


**Figura 39.** Ejemplos de imágenes de la colección LPW. Cada par de imágenes corresponde a un sujeto de estudio. Se aprecian las condiciones cambiantes del entorno de grabación, tanto en la iluminación global como en la dilatación de la pupila.

# CAPÍTULO 4 RESULTADOS

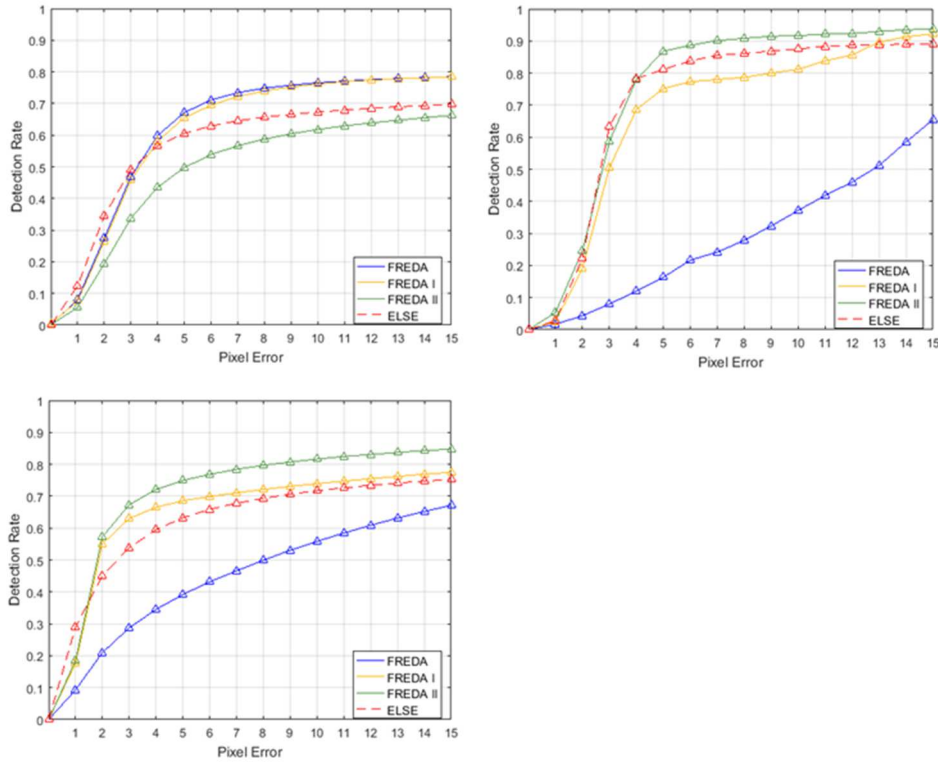
Se compara la precisión en la estimación del centro de pupila de los algoritmos FREDa, FREDa I, FREDa II y ELSE en los conjuntos de imágenes anteriormente descritos. A tenor de la comparativa mostrada en el capítulo 2, se escoge ELSE como algoritmo de referencia por ser claramente superior al resto. El error en la detección del centro de pupila se toma como la distancia euclídea en píxeles entre el centro estimado por el algoritmo y el centro marcado por los autores de cada colección de imágenes. Para hacer comparables estos errores entre diferentes colecciones se redimensionan las imágenes de las colecciones de LPW y Swirski (640 x 480 píxeles) para igualarlas a la resolución de las de Tübingen (384 x 288 píxeles).

La Figura 40 muestra el histograma acumulativo de errores normalizado respecto del total de imágenes (225 569). En el eje de abscisas se marca la distancia entre el punto estimado por el algoritmo y el centro marcado de pupila, mientras que el eje de ordenadas representa la tasa de aciertos obtenida para un cierto error de píxel. Es claramente apreciable que, a partir de los 3 píxeles de error, la tasa de aciertos obtenida es sensiblemente superior con el algoritmo FREDa I que con el resto. El algoritmo FREDa II parece mostrar un comportamiento parecido al de FREDa I pero a partir de los 3 – 4 píxeles de error el primero es ligeramente más preciso en la detección. De los métodos propuestos, el algoritmo FREDa es el que peor rendimiento muestra en términos globales. La curva de ELSE, a pesar de crecer antes que las del resto, a partir de los tres píxeles de error es claramente superado por FREDa I y II. Se debe tener en cuenta también que dada la incertidumbre en el marcado de la colección de Tübingen (de hasta 5 píxeles) para analizar el rendimiento por debajo de esa escala se deben considerar únicamente los resultados de las colecciones de Swirski y LPW.



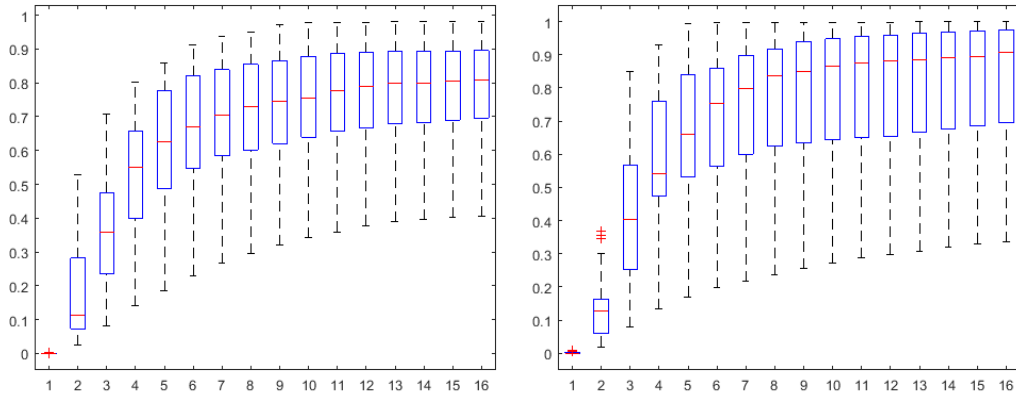
**Figura 40** Histograma acumulativo de errores de los cuatro algoritmos sobre el total de imágenes.

Las figuras 41a, 41b y 41c muestran el histograma acumulativo de errores para las colecciones de Tübingen, Swirski y LPW de manera independiente. En general, se aprecia que el número de aciertos es mucho mayor en las imágenes de Swirski y LPW (figuras 41b y 41c respectivamente) en comparación con las imágenes de Tübingen (41a), dada la calidad de las imágenes de las dos primeras colecciones. Más concretamente, en estos dos conjuntos, destaca la superioridad del algoritmo FREDA II frente al resto. En la colección LPW (130.856 imágenes) este algoritmo se muestra mucho más preciso que ELSE a partir de los 2 píxeles de error y que FREDA I a partir de los 3, mientras que en la de Swirski (600 imágenes) mejora la detección de ELSE a partir de los cuatro píxeles de error. Es notable el bajo rendimiento del algoritmo FREDA en estos dos conjuntos en los que abundan las pupilas de apariencia elíptica. Por el contrario, en el conjunto de imágenes de Tübingen (94.714 imágenes), donde la apariencia de las pupilas es generalmente circular y las imágenes contienen gran cantidad de artefactos, es el algoritmo que mejor comportamiento muestra, seguido muy de cerca por FREDA I. En contra de lo que ocurre en las dos anteriores colecciones, el algoritmo FREDA II pasa a ser, con una notable diferencia, el algoritmo menos preciso.

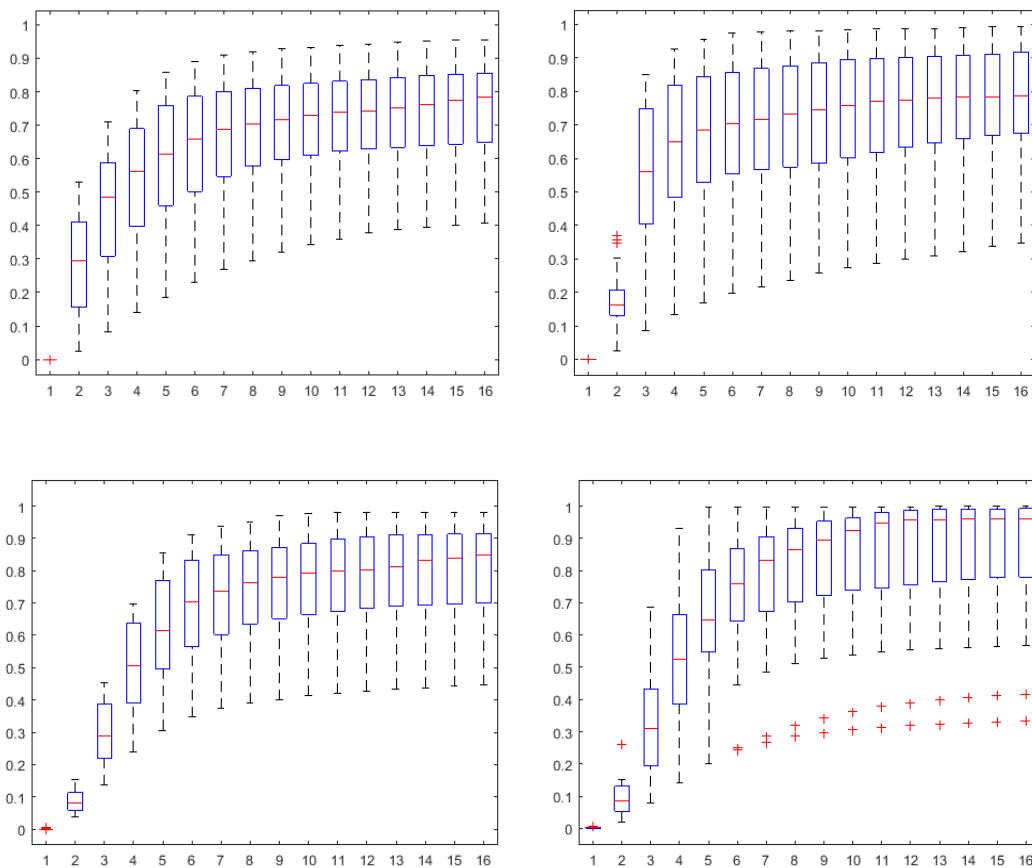


**Figura 41** Histograma acumulativo de errores de los cuatro algoritmos para cada colección de imágenes. (a) Tübingen, (b) Swirski, (c) LPW.

En la Figura 42 se muestra la variabilidad de las curvas de error sobre el total de conjuntos analizados por medio de diagramas de caja para cada distancia de error. La gráfica de la izquierda corresponde a los resultados obtenidos con el algoritmo ELSE, mientras que la de la derecha corresponde a los del algoritmo FRED A I. A pesar de que la variabilidad mostrada por el algoritmo ELSE es menor que la obtenida con FRED A I, es claramente apreciable que, en cada distancia de error, la mediana se sitúa por encima en el caso de FRED A I. En la Figura 43 se separan los resultados en las colecciones de LPW (fila de arriba) y Tübingen (fila de abajo).



**Figura 42** Variabilidad de las curvas de error sobre el total de conjuntos para los algoritmos ELSE (izquierda) y FREDA I (derecha).



**Figura 43** Variabilidad de las curvas de error sobre las colecciones LPW (arriba) y Tübingen (abajo) para los algoritmos ELSE (izquierda) y FREDA I (derecha).

La tabla 2 muestra la tasa de aciertos de los cuatro algoritmos evaluados en cada uno de los conjuntos de imágenes. Se considera un acierto cuando el error no supera los 5 píxeles de distancia respecto del centro marcado. Tal y como se apuntaba anteriormente, se observa



que el algoritmo FREDA es el más robusto de los cuatro en imágenes de baja calidad y pupilas circulares, siendo superior al resto en 12 de los 24 conjuntos de la colección de Tübingen. Por el contrario, el algoritmo FREDA II se muestra como el más preciso en imágenes de buena calidad y en presencia de pupilas elípticas, con una tasa de aciertos superior al resto en 14 de los 22 conjuntos de imágenes de LPW, así como en el conjunto de Swirski. Se observa claramente este resultado en la tabla 3, donde se muestra la tasa de aciertos de cada algoritmo sobre el total de imágenes de cada una de las tres colecciones de imágenes. Con el algoritmo FREDA se llega al 67,17 % de aciertos en la colección de Tübingen, frente al 60,60 % que se obtiene con ELSE. Para las imágenes de LPW y Swirski sin embargo, el algoritmo FREDA II acierta en el 76,84 % y el 86,83 % de los casos respectivamente, mientras que ELSE logra una tasa de aciertos del 65,86 % y del 81,17 %.

Respecto al algoritmo FREDA I, no parece destacar por su rendimiento en ninguna de las colecciones de imágenes, puesto que, como se observa en la tabla 3, a pesar de superar a ELSE en las imágenes de Tübingen (65,50% frente a 60,60 %) el algoritmo FREDA obtiene mejores resultados (67,17 %) y, en cuanto a las imágenes de LPW y Swirski, éste es superado en ambos casos por el algoritmo FREDA II. Sin embargo, como se aprecia en la tabla 4, tomando la tasa de aciertos sobre el global de las imágenes, el algoritmo FREDA I supera al resto con un 69,73 % de aciertos frente al 67,29% de aciertos de FREDA II y al 64,69 % de aciertos de ELSE. Con el algoritmo FREDA únicamente se obtiene una tasa de aciertos del 54,76 % considerando el global de imágenes.

**Tabla 2.** Tasa de aciertos de los cuatro algoritmos para cada conjunto de imágenes. La tasa de aciertos se define como el porcentaje de centros estimados con un error menor o igual a 5 píxeles.

Tübingen	FREDA	FREDA I	FREDA II	ELSE
'conjunto I'	<b>89,38</b>	85,70	69,58	85,95
'conjunto II'	74,45	<b>75,64</b>	52,87	63,76
'conjunto III'	<b>84,31</b>	80,30	55,05	65,31
'conjunto IV'	92,54	<b>92,92</b>	88,14	83,05
'conjunto V'	<b>96,06</b>	95,60	93,30	84,73
'conjunto VI'	<b>88,59</b>	87,95	86,34	77,27
'conjunto VII'	<b>78,22</b>	76,89	72,47	59,61
'conjunto VIII'	71,58	<b>72,22</b>	51,43	67,30
'conjunto IX'	<b>91,09</b>	89,33	68,63	86,72
'conjunto X'	<b>93,57</b>	87,50	88,33	78,93
'conjunto XI'	<b>88,55</b>	72,37	72,37	75,27
'conjunto XII'	73,47	86,26	<b>86,83</b>	79,01
'conjunto XIII'	73,31	74,54	67,62	73,73
'conjunto XIV'	91,68	<b>93,39</b>	61,62	84,22

' conjunto XV'	<b>77,41</b>	73,00	62,53	57,30
' conjunto XVI'	<b>89,03</b>	86,48	75,26	59,95
' conjunto XVII'	94,77	<b>96,64</b>	75,75	89,18
' conjunto XVIII'	50,63	48,70	24,09	<b>52,99</b>
' conjunto XIX'	33,47	32,17	25,14	<b>35,41</b>
' conjunto XX'	61,68	59,45	37,94	<b>68,79</b>
' conjunto XXI'	<b>67,96</b>	67,81	60,93	41,32
' conjunto XXII'	<b>62,38</b>	60,44	30,01	56,61
' conjunto XXIII'	91,19	<b>98,11</b>	97,33	93,40
' conjunto XXIV'	43,60	<b>51,72</b>	47,55	51,20
<b>LPW</b>				
1	63,70	<b>97,45</b>	95,88	88,23
2	74,45	<b>93,58</b>	93,15	50,15
3	55,72	<b>62,32</b>	61,00	50,20
4	18,25	35,53	<b>52,20</b>	34,25
5	14,03	19,72	21,02	<b>31,28</b>
6	47,41	84,85	<b>91,65</b>	63,09
7	56,48	85,30	<b>92,05</b>	70,07
8	54,43	89,27	<b>90,62</b>	84,17
9	32,48	66,52	70,05	<b>70,90</b>
10	27,72	67,90	<b>78,85</b>	65,80
11	20,28	53,43	<b>65,07</b>	56,18
12	49,25	85,13	86,68	<b>88,93</b>
13	28,44	55,94	<b>59,66</b>	52,31
14	31,33	55,52	65,60	<b>74,95</b>
15	15,47	60,18	<b>72,22</b>	65,93
16	66,86	85,73	<b>89,53</b>	87,47
17	21,83	72,82	<b>86,48</b>	67,05
18	58,12	88,63	<b>92,12</b>	83,33
19	32,03	52,98	<b>62,33</b>	41,13
20	53,83	86,18	<b>94,35</b>	23,05
21	24,57	78,27	<b>90,53</b>	62,10
22	15,12	30,52	35,63	<b>78,58</b>
<b>Swirski</b>	21,67	75,17	<b>86,83</b>	81,17

**Tabla 3.** Porcentaje de aciertos de los cuatro algoritmos para el total de cada colección de imágenes.

	<b>FREDA</b>	<b>FREDA I</b>	<b>FREDA II</b>	<b>ELSE</b>
Tübingen	<b>67,17</b>	65,50	49,78	60,60
LPW	43,15	69,88	<b>76,84</b>	65,86
Swirski	21,67	75,17	<b>86,83</b>	81,17

**Tabla 4.** Porcentaje de aciertos de los cuatro algoritmos sobre el total de imágenes

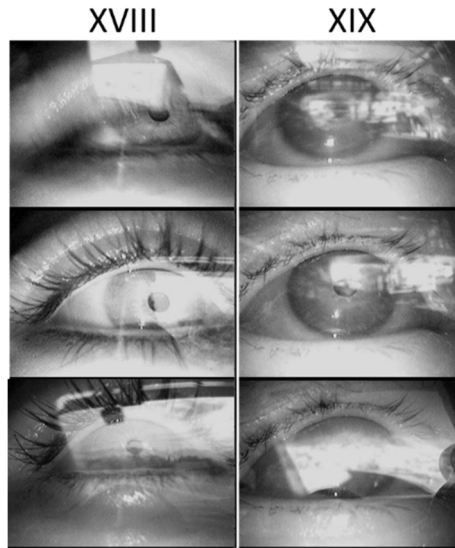
	<b>FREDA</b>	<b>FREDA I</b>	<b>FREDA II</b>	<b>ELSE</b>
Total	54,76	<b>69,73</b>	67,29	64,69

### *Tiempos de cómputo*

Los tiempos de procesamiento medio por imagen obtenidos para cada algoritmo, son de 44 ms para FREDA, 63 ms para FREDA I y de 68 ms para FREDA II. Se observa así un incremento del 43% en el tiempo de cómputo al utilizar el primer refinamiento de centro y de un 53% cuando se utiliza el segundo refinamiento. Como se observa en la tabla 4, el aumento en la detección por el contrario, es del 27 % (aumento del 54.76 % al 69.73 %) para el algoritmo FREDA I y del 23 % (aumento del 54.76 % al 67.29 %) para el algoritmo FREDA II. Dado que los algoritmos han sido implementados en Matlab el tiempo de cómputo no es directamente comparable al obtenido con ELSE puesto que se ha utilizado una versión compilada en lenguaje C++ con la que se observa un tiempo medio de procesamiento por imagen de 8 ms. Sin embargo, asumiendo una razonable diferencia en la velocidad de cómputo entre el entorno Matlab y una implementación en C++ de 5 o 10 veces más lenta, se puede estimar que los algoritmos implementados estarían en un rango de tiempos no muy alejados de ELSE.

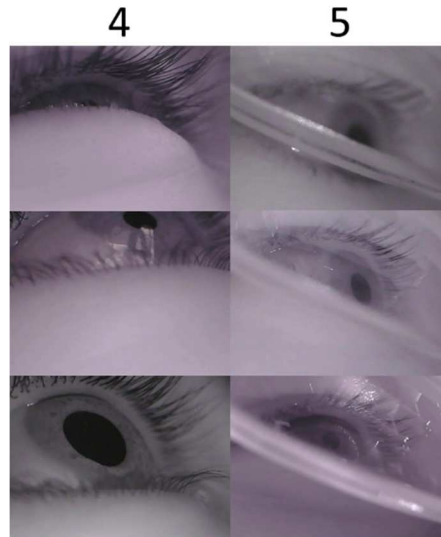
### *Limitaciones de los algoritmos*

Atendiendo a conjuntos individuales, se observa que existen grandes diferencias en la tasa de aciertos entre unos y otros. Habiendo numerosos conjuntos donde la tasa de aciertos supera el 90 %, es especialmente llamativa la baja tasa de aciertos en los conjuntos XVIII y XIX de Tübingen así como en el 4 y 5 de LPW. En los dos primeros el algoritmo ELSE obtiene los mejores resultados con solo un 52,99 % y 35,41 % de centros correctamente estimados, seguido del algoritmo FREDA con un 50,63 % y 33,47 % de aciertos. Tal y como se observa en la Figura 44, donde se muestran tres ejemplos de cada uno de los dos conjuntos, las oclusiones de la pupila debidas a reflejos, párpados o emborronamiento de la imagen, hacen especialmente difícil la detección del centro en estos dos casos.



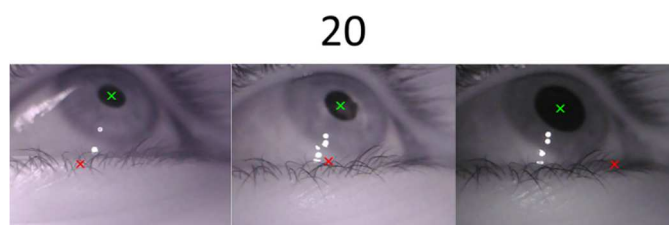
**Figura 44** ejemplos de imágenes de los conjuntos XVIII y XIX donde se aprecian las dificultades encontradas a la hora de estimar el centro de pupila.

En cuanto a las imágenes de los usuarios 4 y 5 de la colección LPW, la tasa de aciertos con el algoritmo FREDA II es del 52,20 % y 21,02 % respectivamente y, del 34,25 % y 31,28 % con ELSE. Se puede apreciar en la Figura 45 que en el usuario 4 la pupila puede llegar a no ser apreciable quedando ocluida por el párpado e, incluso quedar parcialmente cortada por la imagen. En el caso 5, sin embargo, es la montura de las gafas que porta el sujeto la responsable de ocultar total o parcialmente la pupila. El efecto de la lente también es apreciable en el emborronamiento de la imagen.



**Figura 45** Ejemplos de imágenes de los conjuntos 4 y 5 de la colección de Tübingen. Se aprecian las oclusiones y cortes de la pupila en el usuario 4 y la presencia de la montura de las gafas del usuario 5.

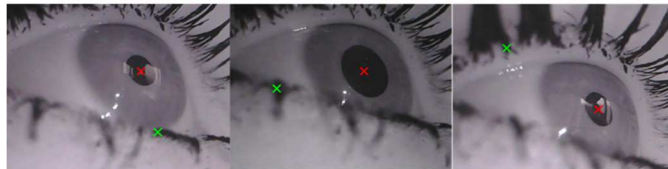
Los conjuntos de imágenes de los usuarios 20 y 21 de la colección LPW muestran tasas de aciertos muy diferentes para los algoritmos FREDa II y ELSE. En el primer caso, FREDa II obtiene una tasa de aciertos del 94,35 % mientras que ELSE se queda en un 23,05 % de aciertos. En la Figura 46 se presentan tres ejemplos de imágenes del usuario 20 con el centro estimado por ambos algoritmos (verde FREDa II y rojo ELSE). A pesar de la aparente buena calidad de la imagen, es apreciable que la pupila está fuera del plano focal, resultando que el borde de la misma aparezca desenfocado. Por tanto, se concluye que el algoritmo ELSE falla en imágenes sin mucha nitidez, mientras que el uso de la FRST aporta robustez en estos casos.



**Figura 46** Ejemplos de detección de centro para FREDa II (verde) y ELSE (rojo). FREDa II parece más robusto cuando la nitidez de la imagen no es muy alta.

Al contrario de lo que ocurre en el usuario 20, en el usuario 22 es el algoritmo ELSE el que obtiene una tasa de acierto elevada, 78,58 %, mientras que FREDa II logra acertar únicamente un 35,63 %. En la misma línea, los otros dos algoritmos basados en la FRST, FREDa y FREDa II obtienen bajos porcentajes de aciertos siendo estos del 15,12 % y del 30,52 % respectivamente. Como se puede observar en la Figura 47, el usuario 22 presenta gran cantidad de máscara de pestañas generando zonas oscuras de tamaño comparable al de la pupila. En este escenario el algoritmo ELSE, enfocado a buscar patrones elípticos, parece más robusto que los basados en la transformada FRST. Las votaciones para determinar el mejor candidato a centro de la FRST parecen fallar en estas imágenes como se muestra en la Figura 48. Se representa, de manera superpuesta sobre una imagen del usuario número 22, el resultado de la FRST en verde y el centro estimado a partir de ésta en rojo. Se observa claramente que las votaciones de la transformada se acumulan principalmente en las zonas oscuras de las pestañas, mientras que en la pupila apenas obtiene votos.

22



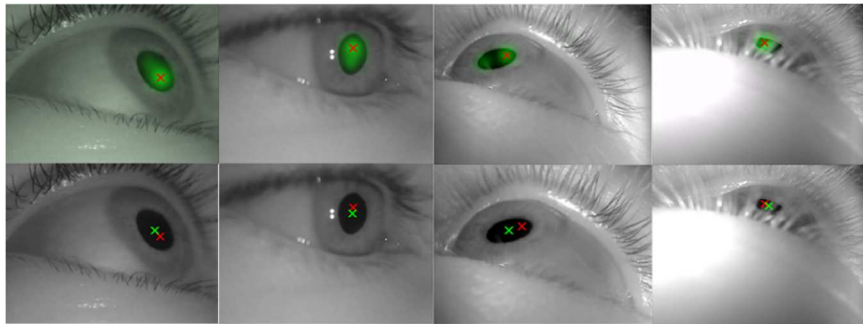
**Figura 47** Ejemplos de detección de centro para FREDa II (verde) y ELSE (amarillo). FREDa II parece más robusto cuando la nitidez de la imagen no es muy alta.



**Figura 48** Imagen del sujeto número 22 con el mapa de votaciones obtenido mediante la FRST superpuesta en verde y en rojo el centro estimado.

Como se puede observar en la Figura 41 y en la tabla 3 donde se muestra la tasa de aciertos sobre el total de cada colección de imágenes, tanto en la colección de Swirski, como en la de LPW, el algoritmo FREDA presenta un rendimiento muy bajo en comparación con el resto y con la tasa de aciertos obtenida en la colección de Tübingen.

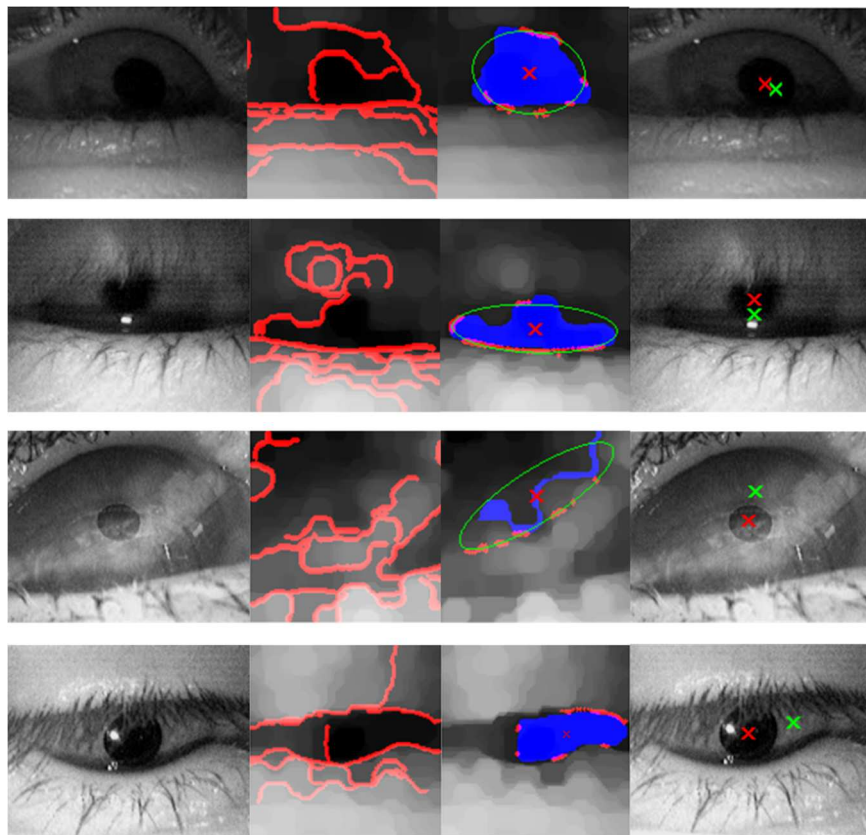
La Figura 49 muestra cuatro ejemplos de imágenes de la colección de Swirski en las que se aprecia la apariencia elíptica de la pupila y que la aplicación única de la transformada FRST no es suficiente para determinar correctamente el centro de pupila. En la primera fila se representa la imagen a la que se le ha superpuesto el resultado de la transformada FRST (verde) y el centro estimado de ella (rojo). En la segunda fila se ha marcado el centro dado por la FRST (rojo) así como el posterior resultado tras aplicar el refinamiento de centro del algoritmo FREDA II.



**Figura 49** Ejemplo de aplicación del algoritmo FREDA sobre imágenes de Swirski. La primera fila muestra el resultado de la transformada FRST (verde) y el centro obtenido (rojo). En la segunda se representa el centro estimado mediante FREDA (rojo) y el obtenido tras el refinamiento de centro del algoritmo FREDAII.

Al igual que ocurre con el algoritmo FREDA en las colecciones Swirski y LPW, el algoritmo FREDA II, a pesar de tener un rendimiento alto en estas imágenes, se muestra sin embargo muy poco preciso en la colección de Tübingen. (véase Figura 41a y tabla 3). En la Figura 50, se presentan cuatro ejemplos de aplicación del algoritmo FREDA II en imágenes de Tübingen en las cuales el centro ha sido correctamente identificado mediante la transformada FRST pero que, tras aplicar el refinamiento de centro, éste ha sido desplazado de manera incorrecta. En la primera columna se muestran, recortadas, las cuatro imágenes de ejemplo. En la segunda se representa sobre el recorte de imagen y la transformación morfológica, los

bordes detectados mediante el método de Canny y, en la tercera, el crecimiento de región (azul) junto con los píxeles coincidentes (rojo) entre la elipse de ajuste (verde) y los bordes detectados. La imagen ilustrada corresponde a la iteración en la que se obtiene un número mayor de coincidencias, es decir, de la que se extrae el centro de la elipse. Finalmente, la última columna muestra el centro obtenido primeramente mediante la transformada FRST (rojo) y el centro obtenido tras aplicar el refinamiento posterior (verde).



**Figura 50** Ejemplo de aplicación del algoritmo FREDa II sobre imágenes de Tübingen en las que el proceso de refinado de centro lo desplaza incorrectamente. En la primera columna se muestran las imágenes recortadas; en la segunda, los bordes detectados sobre la zona de la pupila, en la tercera se representan las coincidencias (rojo) entre la elipse de ajuste (verde) y los bordes detectados anteriormente. La última Figura muestra el centro detectado mediante la FRST (rojo) y el obtenido tras el refinado de centro (verde).



## Conclusiones y líneas futuras

Se han desarrollado tres algoritmos (FREDA, FREDA I y FREDA II), basados en la transformada de simetría radial para la detección del centro de pupila en imágenes oculares obtenidas mediante sistemas de eye tracking de tipo *head-mounted*. Tras evaluar el rendimiento de éstos sobre el total de imágenes recogidas en condiciones tanto de laboratorio como naturales, se tiene que el algoritmo FREDA I es el algoritmo con el que se consigue una mayor precisión en la detección del centro de pupila, superando al algoritmo ELSE que se ha tomado como referencia entre los publicados hasta la fecha.

El objetivo principal era desarrollar un método de detección de pupila robusto frente a las dificultades que se encuentran a la hora de usar dispositivos de eye tracking en condiciones ambientales fuera de laboratorio. De las imágenes utilizadas, las que presentan mayores artefactos son las de la colección de Tübingen, en la que abundan los reflejos, emborronamientos, oclusiones de pupila, etc. Para estas imágenes, el algoritmo más preciso, superando a ELSE, es FREDA, demostrando así que el uso de la FRST es un método apropiado para localizar la pupila en condiciones adversas. Sin embargo, el rendimiento cae notablemente en las colecciones de LPW y Swirski que, en general, son imágenes donde la pupila está mucho más definida. Esto es debido a que las imágenes de Tübingen presentan una apariencia de pupila circular frente a las de LPW y Swirski donde es común la aparición de pupilas con forma elíptica. Es en estas dos colecciones donde los dos algoritmos con refinamiento de centro, FREDA I y FREDA II se comportan mejor, superando en precisión a FREDA, confirmando así la necesidad de añadir un procesamiento posterior a la FRST para hacerlo compatible con pupilas de apariencia elíptica.

De los dos métodos propuestos para tal adaptación, se observa que el del algoritmo FREDA II basado en un crecimiento de región controlado por una detección de bordes es más preciso que la simple utilización del crecimiento de región aplicado en FREDA I. Sin embargo, cuando se aplica a imágenes de peor calidad como las de Tübingen, el método de FREDA II hace perder robustez a la detección, mientras que el crecimiento simple de FREDA I no parece empeorar de manera notable los resultados de FREDA. Así, a pesar de no ser el mejor método en las tres colecciones, el algoritmo FREDA I se muestra como la opción más fiable para cualquier tipo de situación puesto que es robusto frente a artefactos en las imágenes y se adapta de manera aceptable a imágenes con pupilas elípticas.

Una de las razones para la mala adaptación del algoritmo FREDa II a las imágenes de Tübingen es que la mala calidad de estas imágenes hace que la detección de bordes en estas imágenes sea muy poco precisa, siendo difícil definir el contorno de pupila. Por tanto, dado que este proceso de refinamiento se ha mostrado muy superior al resto en imágenes elípticas de buena calidad, sería deseable mejorar este algoritmo mediante la introducción de algún criterio para anular el crecimiento de centro cuando este es irregular y, tomar únicamente el centro dado por la FRST como en el algoritmo FREDa. Así se conseguiría combinar la capacidad de detección en condiciones adversas de uno con la buena adaptación a pupilas de apariencia elípticas del otro.

Otro de los aspectos a mejorar es la confusión que generan en la FRST las zonas oscuras no correspondientes a la pupila presentes en las imágenes, tales como, pestañas, manchas en la piel, sombras etc. Si bien esto puede controlarse en gran medida eligiendo un rango de radios en la FRST donde el tamaño de estos artefactos quede excluido, es deseable que éste rango sea lo mayor posible, puesto que las cambiantes condiciones de iluminación en ambientes reales generan una gran variabilidad en el tamaño de pupila, pudiéndose dar el caso de que en ciertos casos no se detectase la pupila por una mala estimación del tamaño.

Para mejorar la velocidad de detección una técnica común es la de utilizar un centro correctamente identificado como referencia para la búsqueda del siguiente, teniendo en cuenta que la frecuencia de imagen es lo suficientemente alta como para que no haya desplazamientos significativos entre fotogramas consecutivos. Se ha descartado la implementación de esta técnica en los algoritmos propuestos debido a que imposibilitaría la comparación del rendimiento con el algoritmo ELSE que no hace uso de ella.

---

## REFERENCIAS

1. Majaranta P, Bulling A. Chapter 3 Eye Tracking and Eye-Based Human–Computer Interaction.
2. Fuhr W, Santini T, Kasneci G, Kasneci E. PupilNet: Convolutional Neural Networks for Robust Pupil Detection. CoRR. 2016;abs/1601.04902.
3. Fuhr W, Santini TC, Kubler TC, Kasneci E. Ellipse Selection for Robust Pupil Detection in Real-World Environments. CoRR. 2015;abs/1511.06575.
4. Päivi Majaranta, Aoki H, Donegan M, Dan WH, John PH, Hyrskykari A, et al. Gaze Interaction and Applications of Eye Tracking: Advances in Assistive Technologies. Hershey, PA, USA: IGI Global; 2012.
5. Robert GL, Florina U. A Survey of eye tracking methods and applications. . 2013.
6. R. G. Lupu, R. G. Bozomitu, V. Cehan, D. A. Cehan. A new computer-based technology for communicating with people with major neuro-locomotor disability using ocular electromyogram. Proceedings of the 2011 34th International Spring Seminar on Electronics Technology (ISSE); ; 2011.
7. Jacob RJK. Virtual Environments and Advanced Interface Design. In: Barfield W, Furness I, Thomas A., editors. New York, NY, USA: Oxford University Press, Inc; 1995. p. 258-88.
8. Kasneci E. Towards the automated recognition of assistance need for drivers with impaired visual field.[dissertation]. Universität Tübingen, Germany, 2013.; 2013.
9. Turner J, Alexander J, Bulling A, Schmidt D, Gellersen H. Eye Pull, Eye Push: Moving Objects between Large Screens and Personal Devices with Gaze and Touch. In: Kotz P, Marsden G, Lindgaard G, Wesson J, Winckler M, editors. Human-Computer Interaction – INTERACT 2013: 14th IFIP TC 13 International Conference, Cape Town, South Africa, September 2-6, 2013, Proceedings, Part II. Berlin, Heidelberg: Springer Berlin Heidelberg; 2013. p. 170-86.
10. Pan B, Hembrooke HA, Gay GK, Granka LA, Feusner MK, Newman JK. The Determinants of Web Page Viewing Behavior: An Eye-tracking Study. Proceedings of the 2004 Symposium on Eye Tracking Research & Applications; San Antonio, Texas. New York, NY, USA: ACM; 2004.
11. Fuhr W, Tonsen M, Bulling A, Kasneci E. Pupil detection for head-mounted eye tracking in the wild: an evaluation of the state of the art. Mach Vision Appl. 2016;27(8):1275-88.
12. Kassner M, Patera W, Bulling A. Pupil: An Open Source Platform for Pervasive Eye Tracking and Mobile Gaze-based Interaction. Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication; Seattle, Washington. New York, NY, USA: ACM; 2014.
13. Dongheng Li, D. Winfield, D. J. Parkhurst. Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops; ; 2005.

- 
14. Li D, Babcock J, Parkhurst DJ. openEyes: A Low-cost Head-mounted Eye-tracking Solution. Proceedings of the 2006 Symposium on Eye Tracking Research & Applications; San Diego, California. New York, NY, USA: ACM; 2006.
  15. Goni S, Echeto J, Villanueva A, Cabeza R. Robust Algorithm for Pupil-Glint Vector Detection in a Video-oculography Eyetracking System. Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 4 - Volume 04; Washington, DC, USA: IEEE Computer Society; 2004.
  16. Fuhl W, Käßbler T, Sippel K, Rosenstiel W, Kasneci E. ExCuSe: Robust Pupil Detection in Real-World Scenarios. In: Azzopardi G, Petkov N, editors. Computer Analysis of Images and Patterns: 16th International Conference, CAIP 2015, Valletta, Malta, September 2-4, 2015 Proceedings, Part I. Cham: Springer International Publishing; 2015. p. 39-51.
  17. Xia Liu, Fengliang Xu, K. Fujimura. Real-time eye detection and tracking for driver observation under various light conditions. Intelligent Vehicle Symposium, 2002. IEEE; ; 2002.
  18. Trösterer S, Meschtscherjakov A, Wilfinger D, Tscheligi M. Eye Tracking in the Car: Challenges in a Dual-Task Scenario on a Test Track. Adjunct Proceedings of the 6th International Conference on Automotive User Interfaces and Interactive Vehicular Applications; Seattle, WA, USA. New York, NY, USA: ACM; 2014.
  19. C. Braunagel, E. Kasneci, W. Stolzmann, W. Rosenstiel. Driver-Activity Recognition in the Context of Conditionally Autonomous Driving. 2015 IEEE 18th International Conference on Intelligent Transportation Systems; ; 2015.
  20. Kasneci E, Sippel K, Heister M, Aehling K, Rosenstiel W, Schiefer U, et al. Homonymous Visual Field Loss and Its Impact on Visual Exploration: A Supermarket Study. Translational Vision Science & Technology. 2014 08/17;3(6):2.
  21. Sugano Y, Bulling A. Self-Calibrating Head-Mounted Eye Trackers Using Egocentric Visual Saliency. Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology; Charlotte, NC, USA. New York, NY, USA: ACM; 2015.
  22. Świrski L, Bulling A, Dodgson N. Robust Real-time Pupil Tracking in Highly Off-axis Images. Proceedings of the Symposium on Eye Tracking Research and Applications; Santa Barbara, California. New York, NY, USA: ACM; 2012.
  23. Loy G, Zelinsky A. Fast Radial Symmetry for Detecting Points of Interest. IEEE Trans Pattern Anal Mach Intell. 2003 aug;25(8):959-73.
  24. Skodras E, Fakotakis N. Precise Localization of Eye Centers in Low Resolution Color Images. Image Vision Comput. 2015 apr;36(C):51-60.
  25. Tonsen M, Zhang X, Sugano Y, Bulling A. Labelled Pupils in the Wild: A Dataset for Studying Pupil Detection in Unconstrained Environments. Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications; Charleston, South Carolina. New York, NY, USA: ACM; 2016.
  26. Javadi A, Hakimi Z, Barati M, Walsh V, Tcheang L. SET: a pupil detection method using sinusoidal approximation. Frontiers in Neuroengineering. 2015 03/23;8:4.
  27. Ma L, Tan T, Wang Y, Zhang D. Local intensity variation analysis for iris recognition. Pattern Recognit. 2004 6;37(6):1287-98.

---

28. Sippel K, Kasneci E, Aehling K, Heister M, Rosenstiel W, Schiefer U, et al. Binocular Glaucomatous Visual Field Loss and Its Impact on Visual Exploration - A Supermarket Study. PLOS ONE. 2014 08/27;9(8):e106089.