

UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA
FACULTAD DE INGENIERÍA

INGENIERÍA BIOMÉDICA

**Desarrollo de un software para la clasificación del paralelismo y desviación ocular mediante el
reconocimiento del reflejo corneal**

Presentado por:

Maria Paula Blanco Olaya

John Sebastian Rojas Sanchez

Jose Fernando Camargo Mantilla

Códigos:

U00114495

U00106168

U00110269

Director:

Ing. Manuel Hernando Franco Arias

Codirectores:

Ing. Edwing Ramón Ayala Estupiñan

Dra. Angélica María Prada Rocha

Bucaramanga

2022

Dedicatoria

A mi familia por apoyarme y estar siempre presente en cada paso, especialmente a mis padres, Javier Blanco y Dolly Olaya, mi hermana María Alejandra Blanco, por apoyar mi sueño de ser profesional y guiarme en momentos difíciles, a mi abuela, Maria Dolly Arango por ser mi mayor impulso, mi motivación diaria y enseñarme a jamás rendirme. Este logro es para todos ustedes, por su amor y apoyo incondicional.

-Maria Paula Blanco Olaya

Deseo dedicar el cierre de este proyecto a toda mi familia, principalmente a mi madre Yolanda Mantilla por ser la inspiración y el motivo en toda mi formación académica tanto de Maestro en Música como de Ingeniería Biomédica, a mi padre Pedro Camargo por su inquebrantable apoyo desde antes, durante y final de este proceso educativo el cual no habría sido posible sin su ayuda, a mis hermanos Lina y Ricardo que mantuvieron su amor, comprensión y hombro siempre dispuesto para cada momento difícil que se presentó y no permitirme caer en la tentación de rendirme, a mi hija Mariana por no dejarme perder la capacidad de sorprenderme por cada aprendizaje por muy pequeño que fuera y por último a mi primera carrera de Maestro en Música, ya que esta me dio el tiempo, capacidad y la disciplina necesaria para poder estudiar esta segunda titulación.

-Jose Fernando Camargo

A mi familia por siempre estar conmigo y apoyarme a cumplir mis sueños, en especial a mi papá Jorge Enrique Rojas Moreno y mi mamá Nora Alicia Sanchez Pabon por ser mi empuje diario, a mis hermanas Wendy Nathalia Rojas Sanchez y Adriana Lucia Rojas Sanchez, a mis amigos por brindarme su compañía cuando la necesito. Este logro se los dedico a todos ustedes por siempre brindarme su cariño y apoyo incondicional.

- John Sebastian Rojas Sanchez

Agradecimientos

En primer lugar, queremos agradecer a los tutores Manuel Franco, Ingeniero Biomédico, Edwing Ramón Ayala, Ingeniero Biomédico y Dr. Angélica María Prada, MD. Oftalmóloga, los cuales con su conocimiento tanto académico como humano, nos brindaron su apoyo y guiaron a través de cada una de las etapas realizadas para alcanzar un resultado satisfactorio a lo largo del proyecto.

En segundo lugar, queremos agradecer a la Universidad Autónoma de Bucaramanga por brindarnos los recursos académicos que fueron necesarios para llevar a cabo el proceso de investigación de este trabajo de grado. No se podría haber llegado al resultado esperado sin la excelente formación brindada por la Institución.

Por último, queremos agradecer a todos nuestros compañeros y conocidos por brindarnos el apoyo incondicional para crecer profesionalmente, en especial a nuestros padres que siempre tuvieron la disposición de darnos apoyo y las palabras correctas para comenzar de nuevo.

Muchas gracias a todos.

Resumen

La desviación ocular se conoce como la insuficiencia de paralelismo en los ejes horizontal y vertical de los ojos, esta alteración puede presentarse en uno o en ambos ojos, también puede darse desde el nacimiento o puede ser adquirida durante el transcurso de la vida de la persona. Para el diagnóstico de la desviación ocular se tienen en cuenta cinco pasos generales, primero la exploración física y determinación de la fijación ocular, segundo lograr establecer o determinar el ángulo KAPPA, como tercer paso es la realización del test de HIRSCHBERG, el cuarto paso es el test de KRIMSKY y por último el COVER TEST ALTERNANTE. Sin embargo, este proceso de diagnóstico suele ser largo, subjetivo y a veces tardía en su identificación como diagnóstico, retrasando el comienzo del tratamiento para la corrección de esta patología. Es por esto, que se desarrolla un software para brindar una herramienta de ayuda al profesional para que pueda realizar una clasificación de forma más rápida y exacta, buscando automatizar y reducir el tiempo de diagnóstico; Para el desarrollo de este proyecto se utilizó como herramienta de programación el software Matlab, permitiendo obtener como resultado final una aplicación ejecutable que no requiere de la licencia del software mencionado.

Las pruebas de la aplicación se realizaron con una base de datos de imágenes obtenidas de libros clínicos y parte de otros estudios científicos, que permitieron seleccionar las estructuras de interés de cada paciente y clasificar a este, según el tipo de desviación ocular que presente.

Abstract

The ocular deviation is known as the insufficiency of parallelism in the horizontal and vertical axes of the eyes, this alteration can occur in one or both eyes, it can also occur from birth or can be acquired during the course of the person's life. For the diagnosis of ocular deviation five general steps are taken into account, first the physical examination and determination of ocular fixation, second to establish or determine the KAPPA angle, as a third step is the realization of the HIRSCHBERG test, the fourth step is the KRIMSKY test and finally the ALTERNATIVE COVER TEST. However, this diagnostic process is usually long, subjective and sometimes late in its identification as a diagnosis, delaying the start of treatment for the correction of this pathology. For this reason, a software is developed to provide a tool to help the professional to make a classification in a faster and more accurate way, seeking to automate and reduce the diagnosis time; For the development of this project, Matlab software was used as a programming tool, allowing to obtain as a final result an executable application that does not require the license of the mentioned software.

The application tests were performed with a database of images obtained from clinical books and part of other scientific studies, which allowed to select the structures of interest of each patient and classify them according to the type of ocular deviation they present.

Capítulo I	13
Aspectos generales	13
1.1 Problema u Oportunidad	13
1.2 Descripción del problema	14
1.3 Justificación	14
1.4 Pregunta problema	16
1.5 Objetivo general	16
1.6 Objetivos específicos	17
1.7 Limitaciones y Delimitaciones	17
Capítulo II	18
Marco Teórico y Estado del Arte	18
2.1. Marco teórico	18
2.1.1 Desviación Ocular	18
2.1.2 Visión Binocular	19
2.1.3 Órgano Ocular	19
2.1.4 Punto de Visión Cercana	20
2.1.5 Ángulo Kappa	20
2.1.6 Ambliopía	21
2.1.7 Tratamiento Quirúrgico	21
2.2 Estado del Arte	22
Capítulo III	28
Metodología	28
3.1 Diseño	29
3.1.1 Recolección de imágenes para la base de datos	29
3.1.2 Determinar el Software a usarse para el procesamiento y análisis de las imágenes	29
3.1.3 Investigar técnicas de procesamiento para identificar regiones de interés	29
3.2 Implementar	30
3.2.1 Caracterización de la imagen	30
3.2.2 Obtención de los valores oculares de las regiones de interés en la imagen.	30
3.2.3 Procesamiento de los datos para la clasificación	30
3.3 Validar	31
3.3.1 Obtener imágenes con un diagnóstico ya clasificado	31
3.3.2 Validación con imágenes previamente diagnosticadas	31
Capítulo IV	32

Resultados y Análisis de Resultados	32
4.1 Diseño	32
4.1.1. Recolección de imágenes para la base de datos	32
4.1.2 Selección del software a utilizar	33
4.2 Implementación	
4.2.1. Adecuación de la imagen e identificación del Iris	35
4.2.2. Adecuación de la imagen e identificación de la pupila	37
4.2.3. Adecuación de la imagen e identificación del Reflejo Corneal	38
4.2.4 Obtención de los valores oculares posibles con la identificación de las estructuras de los ojos en la imagen	39
4.2.5 Iris	39
4.2.6 Pupila	40
4.2.7 Reflejo Corneal	41
4.2.8 Análisis y procesamiento de imágenes para la clasificación de la desviación ocular.	42
4.2.9 Diseño de la interfaz	42
4.3. Validar	
4.3.1 Funcionamiento del software	45
4.3.2 Pruebas y Análisis de resultados	48
4.3.3 Tabla comparativa de diagnóstico definido vs clasificación del software	52
Capítulo V	53
Conclusiones y Recomendaciones	53
5.1 Conclusiones	53
5.2 Recomendaciones	54
Capítulo VI	55
Referencias Bibliográficas	55
Anexo 1	58
Diagrama de flujo software Strap-Vision	58
Anexo 2	59
GUIA RAPIDA PARA EL USO Y MANEJO DEL SOFTWARE STRAB-VISION PARA LA CLASIFICACIÓN DEL PARALELISMO Y DESVIACIÓN OCULAR MEDIANTE EL RECONOCIMIENTO DEL REFLEJO CORNEAL	59
Anexo 3	64
Código interfaz de presentación	64

Anexo 4	67
Código interfaz datos del paciente	67
Anexo 5	72
Código interfaz principal	72

Contenido de Figuras

Figura 1. Reflejo Corneal para Ángulo Kappa.	15
Figura 2. Test de HIRSCHBERG realizado por un profesional de la salud.	15
Figura 3. Test de KRIMSKY.	16
Figura 4 . Punto de Visión Cercana.	20
Figura 5. Representación de Ángulo Kappa de magnitud positiva.	20
Figura 6. Ambliopía Estrábica.	21
Figura 7. Paciente mientras se le realiza la prueba con un Sinoptóforo en consulta.	25
Figura 8. Ejemplo de informe que realiza el Gazelab.	27
Figura 9. Diagrama de bloques de la metodología aplicada.	28
Figura 10. Base de datos de imágenes a usar para pruebas en el software.	32
Figura 11. Paciente uno, imagen original para pruebas iniciales.	34
Figura 12. Paciente uno, imagen original en tercer plano, escala de grises modificando brillo.	35
Figura 13. Paciente uno, imagen binarizada.	35
Figura 14. Paciente uno, imagen con binarización invertida y eliminación de áreas sobrantes para obtener el Iris.	36
Figura 15. Filtro gaussiano, donde se aplica un filtro pasa altas y un filtro pasa bajas.	36
Figura 16. Paciente uno, Imagen original en escala de grises.	37
Figura 17. Paciente uno, Imagen en escala de grises binarizada.	37

Figura 18. Paciente uno, imagen con binarización invertida y eliminación de áreas sobrantes para obtener la Pupila.	37
Figura 19. Paciente uno, imagen con binarización invertida para resaltar Reflejo Corneal.	38
Figura 20. Paciente uno, Centroides para resaltar el Iris.	38
Figura 21. Paciente uno, círculos resaltando el Iris.	39
Figura 22. Paciente uno, círculos resaltando el Iris en la imagen inicial.	39
Figura 23. Paciente uno, Centroides para resaltar la Pupila.	39
Figura 24. Paciente uno, círculos resaltando la Pupila.	40
Figura 25. Paciente uno, círculos resaltando la Pupila en la imagen inicial.	40
Figura 26. Paciente uno, Centroides para resaltar el Reflejo Corneal.	40
Figura 27. Paciente uno, círculos resaltando el Reflejo Corneal en la imagen inicial.	41
Figura 28. Paciente uno, suma de ROIs en la imagen inicial.	41
Figura 29. Interfaz principal del Software.	44
Figura 30. Primera ventana de la interfaz, que incluye datos de autores y colaboradores.	45
Figura 31. Segunda ventana de la interfaz, permite ingresar datos del paciente.	45
Figura 32. Primera prueba, paciente uno.	47
Figura 33. Segunda prueba, paciente dos.	48
Figura 34. Tercera prueba, paciente tres.	49
Figura 35. Cuarta prueba, paciente diez con esotropia.	50
Figura 36. Cuarta prueba, paciente diez con esotropia.	50

Figura 37. Quinta prueba, paciente diez, luego de cirugía de corrección.	51
Figura 38. Cuarta prueba, paciente diez, Orthotropia.	51

Contenido de Tablas

Tabla 1. <i>Matriz de Pugh para comparar dos software, Matlab y Python.</i>	32
Tabla 2. <i>Comparativa diagnóstico Vs clasificación.</i>	52

Capítulo I

Aspectos generales

1.1 Problema u Oportunidad

La visión binocular es conocida como aquel mecanismo que da la capacidad de integrar imágenes captadas por ambos ojos en una sola percepción, este tipo de visión permite el movimiento sincrónico de los ojos obteniendo la integración de dichas imágenes desde un solo punto focal. Teniendo en cuenta que es un mecanismo muy complejo, esto lo hace también muy vulnerable durante el desarrollo sensoriomotor ya sea durante el nacimiento o que se desarrolle durante la vida, dando lugar a diferentes patologías, una de estas es el estrabismo. El estrabismo se define como “la desviación de la alineación perfecta de los ojos en cualquier dirección resultante de un trastorno visual” [1].

Según William Lawrence (2015), el estrabismo se define como la desviación ocular que resulta cuando uno o los dos ojos no se sitúan en armonía con el otro. El ojo puede estar girado hacia el centro en forma convergente, hacia afuera en forma divergente, hacia abajo o hacia arriba según sea el caso [2].

Esta patología oftalmológica se presenta entre un 2 % a 4 % de la población infantil en el mundo y su detección se realiza a través de un examen oftalmológico sistemático, el cual inicia con una exploración física de los ojos, seguido de una evaluación de la agudeza visual de la persona, se observan movimientos oculomotores y por medio de test específicos (ángulo KAPPA, test de KRIMSKY, Cover test alternante) se puede cuantificar la desviación usando lentes prismáticos, encontrando un valor adecuado según el test utilizado cuando el Reflejo Corneal está centrado o cuando hay ausencia de movimiento[3]. Según estudios, esta enfermedad puede ser detectada y corregida con mayor facilidad en las edades de 1 a 4 años, sin embargo la mayoría de los casos de estrabismo no son diagnosticados a tiempo y aún con un adecuado seguimiento y proceso médico, conlleva a una baja probabilidad de una mejora por parte del tratamiento [1].

En Colombia los estudios sobre el tema son escasos, según la asociación colombiana de oftalmología pediátrica y estrabismo, más conocida como ACOPE un 4 % de nuestra población padece de estrabismo además de tener una detección y tratamiento tardío [4].

1.2 Descripción del problema

Actualmente pocas personas reciben un tratamiento a tiempo para la corrección de esta patología realizando una cirugía que busca cumplir varios objetivos, dentro de los cuales se tiene la corrección del paralelismo ocular, la mejora del campo visual y la visión binocular, sin embargo; durante la realización de esta cirugía, según Kampanartsanyakorn y Hertle, existe un porcentaje de éxito durante el tratamiento que oscila entre el 60.2 % y el 85 %; el primero lo plantea Kampanartsanyakorn y el segundo Hertle [5], puesto que el oftalmólogo realiza la cirugía sin tener claridad exacta de cuanto es la desviación ocular que se debe corregir, no tener un buen control sobre factores mecánicos y sensoriales de las estructuras intervenidas del paciente y la falta de una detección temprana de este [5].

1.3 Justificación

El diagnóstico para definir la desviación ocular se realiza bajo cinco procedimientos generales, el primero es la exploración física y determinación de la fijación ocular, la cual es realizada con el uso de un oftalmoscopio para observar el fondo del ojo y estudiar la fijación visual [6].

Como segundo procedimiento es establecer el ángulo KAPPA, este es un test de tipo monocular, el cual el profesional de la salud lo determina ubicando de forma manual a una distancia entre 32 y 40 centímetros, una luz de forma horizontal a la altura aproximada de su eje visual y su eje anatómico, el cual genera un reflejo luminoso en la Córnea del paciente llamado Reflejo Corneal (Ver Figura 1), permitiendo al profesional identificar bajo criterio propio el Ángulo Kappa según que tan centrado se encuentre este [7].

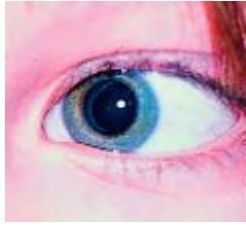


Figura 1. Reflejo Corneal para Ángulo Kappa [7].

Un tercer paso es la realización del test de HIRSCHBERG (Ver Figura 2), este es de tipo binocular que consiste en iluminar los ojos del paciente y observar el reflejo luminoso en la Córnea, este proceso presenta las mismas falencias que se nombraron anteriormente, ya que es el profesional el cual bajo su criterio y experticia el que determina, si hubo un desplazamiento asimétrico del reflejo luminoso, lo cual indicaría la presencia de desviación [7].



Figura 2. Test de HIRSCHBERG realizado por un profesional de la salud [7].

Como cuarto paso se tiene el test de KRIMSKY (Ver Figura 3), el cual se realiza con prismas, se ubican frente al ojo afectado, haciendo que este se centre automáticamente al ver a través del prisma, esto hace posible la realización de una cuantificación de la desviación al ir cambiando las potencias de los prismas hasta encontrar nuevamente bajo un criterio desde la perspectiva visual del profesional una alineación de los ojos con un valor en dioptrías prismáticas o milímetros [7].

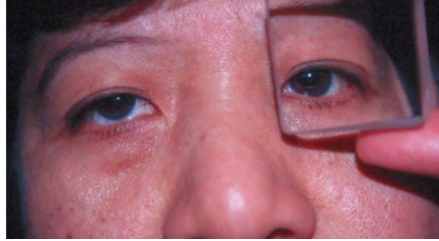


Figura 3. Test de KRIMSKY [7].

El quinto procedimiento que se realiza es el test Gold Standar o cover test alternante (CTA), se puede conjugar también con prismas como un PCT (prisma cover test), este se hace alternando rápidamente la oclusión en cada ojo, logrando disociarlos y maximizando la desviación, se puede con fijación cercana o lejana y el uso de los prismas permite medir el grado de la desviación [8].

Los cinco procedimientos mencionados anteriormente tienen en común, que su diagnóstico es determinado por el profesional bajo criterio propio, debido a que actualmente no existe un dispositivo o software asequible con la capacidad de clasificar y cuantificar el tipo de desviación ocular.

1.4 Pregunta problema

¿Cómo clasificar el paralelismo y desviación ocular mediante el reconocimiento del Reflejo Corneal empleando técnicas de procesamiento de imágenes?

1.5 Objetivo general

Desarrollar un software para la clasificación del paralelismo y desviación ocular mediante el reconocimiento del Reflejo Corneal

1.6 Objetivos específicos

1. Diseñar un software para el reconocimiento de la Pupila y el Reflejo Corneal mediante el uso técnicas de procesamiento de imágenesl.
2. Implementar las técnicas de reconocimiento de la Pupila y el Reflejo Corneal, mediante el funcionamiento del diseño propuesto para la clasificación del tipo desviación.
3. Validar las clasificaciones obtenidas por el software diseñado mediante comparaciones con bases de datos científicos para la aprobación de la funcionalidad del mismo.

1.7 Limitaciones y Delimitaciones

Para la presente investigación y realización de las pruebas del software se tuvieron en cuenta dos opciones para la creación de la base de datos, i) Obtenerlas por parte de un banco de imágenes o libros; ii) Obtenerlas por parte de una institución médica. Sin embargo, al escoger las imágenes, ambas opciones presentaron las mismas limitaciones, los bancos de imágenes encontrados, no estaban especializados, las imágenes no estaban clasificadas por un profesional puesto que la mayoría de bases de datos eran de rostros que no contaban con un Reflejo Corneal.

No se utilizaron técnicas de inteligencia artificial, puesto que no fue necesaria y por las escasas imágenes funcionales que se podrían implementar para el uso de IA en el desarrollo del software, este se llevó a cabo con técnicas de procesamientos de imágenes que permitieron identificar cada una de las regiones de interés y clasificar los resultados de cada paciente. Sin embargo en el desarrollo del proyecto, se tuvo en cuenta la investigación de diferentes tipos de inteligencia artificial, como lo fueron KNN, forward selection, reconocedor euclidiano, entre otros.

Capítulo II

Marco Teórico y Estado del Arte

2.1. Marco teórico

Al momento de incursionar en esta temática de investigación se hace relevante conocer los siguientes términos: Desviación Ocular, Visión Binocular, Órgano Ocular, punto de Visión Cercana, Ángulo Kappa, Ambliopía, Tratamiento quirúrgico, con el fin de entender de mejor manera el trayecto histórico del avance científico en el campo de la clasificación del paralelismo y desviación ocular.

2.1.1 Desviación Ocular

El estrabismo es una desviación ocular definida como una insuficiencia de paralelismo en los ejes de los ojos [2]. Esta alteración se puede presentar en uno o en los dos ojos, también puede estar presente desde el nacimiento, es decir de carácter congénito o puede ser adquirida durante el transcurso de la vida de la persona [1].

Según la Asociación Colombiana de Oftalmología Pediátrica y Estrabismo (ACOPE), el estrabismo se presenta en el 4% de la población colombiana, siendo menos frecuente en adultos, presentándose en ellos de forma adquirida [4].

Estas desviaciones oculares también se pueden presentar como desórdenes neuromusculares, como problemas sistémicos relacionados con el enfoque y defectos refractivos, los hay de tipo paralítico por la afectación en los pares craneales ya que estos controlan los ocho músculos oculares, así mismo, se debe tener en cuenta posibles causas tumorales, infecciosas o traumáticas [3].

Las desviaciones oculares aparte de su clasificación congénita y adquirida, también se clasifican en 5 tipos según su dirección, aunque el quinto tipo sería la mezcla de dos de los 4 principales. Las desviaciones que se presentan de forma horizontal con dirección hacia adentro se denominan ENDOTROPIA, y las que tienen

orientación hacia fuera serían de tipo EXOTROPIA, en el caso de ser desviaciones verticales, las que tiene orientación hacia arriba con HIPERTROPIA y las contrarias a esta serían HIPOTROPIA[11]. También existen desviaciones torsionales, EXCICLOTROPIA en contra de las manecillas del reloj e INCICLOTROPIA con las manecillas del reloj.

Para poder entender el objeto de estudio en este proyecto, se debe tener conocimiento de los aspectos que relacionan la patología expuesta y el órgano afectado con las características que relacionan el estado estrábico.

2.1.2 Visión Binocular

Se le dice Visión Binocular a la capacidad de formar imágenes mediante el uso de dos ojos y formando estos una sola imagen a partir del envío de esta información por separado[12].

2.1.3 Órgano Ocular

Este es el que da el sentido de la vista con todas sus estructuras que la conforman y en especial la retina que conecta con todo el sistema nervioso que hace posible ver la luminosidad de los cuerpos del entorno [13].

Este órgano se encuentra en pares y están ubicados simétricamente, su parte más externa es la Cornea, sus medidas en promedio son: 25 mm en su parte anteroposterior, de 23 mm a 25 mm transversalmente, vertical 23mm y su peso puede estar entre los 7 gr a 7.5 gr [14].

2.1.4 Punto de Visión Cercana

Hace referencia a la distancia en la que un objeto es enfocado con una acomodación máxima del Cristalino (Ver Figura 4)[15].

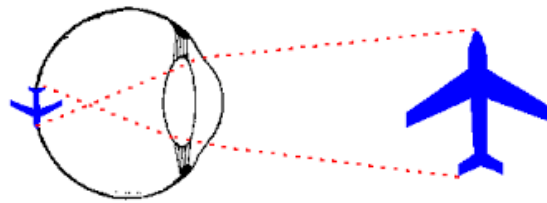


Figura 4 . Punto de Visión Cercana [15].

2.1.5 Ángulo Kappa

Es el ángulo entre el centro de la Pupila y el Reflejo Corneal (Ver Figura 5)[16].

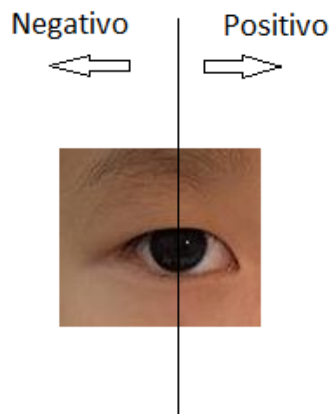


Figura 5. Representación de Ángulo Kappa de magnitud positiva[16].

2.1.6 Ambliopía

Es un déficit de tipo funcional de la agudeza visual, es reversible si se trata precozmente. El cual busca prevenir o tratar la ambliopía funcional siendo esta la prioridad del tratamiento. La ambliopía ocurre principalmente cuando el estrabismo se presenta en la infancia (Ver Figura 5)[18].



Figura 6. Ambliopía estrábica[18].

2.1.7 Tratamiento Quirúrgico

Es una opción de tratamiento para la corrección del estrabismo indiferente de qué tipo sea, esta cirugía busca corregir el des-alineamiento de los ejes oculares. Para la realización de este tratamiento se debe tener en cuenta otros procedimientos aparte de la sola medida en dioptrías prismáticas del estrabismo como lo son: Sus antecedentes clínicos, examen de agudeza visual en cada ojo (fijador y no fijador), biomicroscopia con lámpara de hendidura, tonometría, oftalmoscopia indirecta con fundoscopios, movimientos oculares y el objeto de estudio de este proyecto la clasificación de la desviación[19].

2.2 Estado del Arte

Actualmente los profesionales de la salud realizan el diagnóstico de estrabismo de forma manual y sin la ayuda de ningún equipo biomédico especializado. Dentro de los métodos más comúnmente utilizados se encuentran el Test de Hirschberg, el cual consiste en observar el reflejo de una luz centrado en ambas Pupilas de forma simétrica, para ello se iluminan los dos ojos a una distancia de 30-40cm y mediante el cálculo de dioptrías prismáticas permite cuantificar el grado de desviación ocular. Otro método usado es el Cover Test, el cual busca detectar forias o estrabismos intermitentes, este se realiza fijando la vista en un objeto pequeño a una distancia de 3 metros del examinador, posteriormente se ocluye un ojo y se observa el contrario, de tal modo que si existe una tropia(estrabismo continuo) el ojo desviado se moverá al ocluir el ojo normal, para permitir fijar la vista, por caso contrario si existe una foria(estrabismo intermitente), el ojo con mal alineamiento se moverá a su posición normal cuando se descubra. entre otros procesos de diagnóstico [20].

Dado que la mayoría de estos procesos se realizan manual y empíricamente, se ve la necesidad de desarrollar un software el cual permita la detección de estrabismo mediante el uso de fotografías y que a su vez permita clasificar según el grado de desviación del ojo. En consecuencia, como un primer enfoque en la investigación presente en este documento, se indagará acerca de las técnicas existentes para la medición de estrabismos y demás técnicas innovadoras que están en desarrollo y proyectos de desarrollo similares.

En el 2016 Daniela Cioplean y Lăcrămioara Nitescu Raluca, investigación como la sarcopenia causada por la edad afecta los músculos de los ojos, causando lo que se conoce como ojo caído, siendo este un tipo de estrabismo asociado a la edad. Lo cual da una visión más amplia de que la desviación ocular no solo ocurre de manera congénita sino también se puede adquirir con el paso del tiempo [21].

Actualmente se han desarrollado diferentes métodos(aunque poco comunes) para diagnosticar si una persona padece algún tipo de desviación ocular, como lo es el caso de Lluvia Grant Bosworth y Eileen e. Abedul quienes en el 2008 desarrollaron un método de detección de dirección de movimiento y asimetrías de

movimiento VEP en niños normales y niños con esotropía infantil, el cual consiste en colocar un electrodo de referencia en un infante, posteriormente este se sienta en el regazo de uno de sus padres a una distancia de visualización de 50cm del monitor, se ocluye un ojo con un parche ocular opaco con el fin de obtener una visualización monocular se coloca un juguete pequeño transparente frente al monitor para atraer la mirada del niño hacia el centro del monitor durante la presentación del estímulo. La respuesta mVEP para cada individuo contiene una respuesta más fuerte al movimiento del estímulo en una dirección que en la otra, estas medidas son el índice de asimetría [22].

Uno de los métodos de identificación del paralelismo ocular fue realizado por Keli Mao, Yahan Yang, Chong Guo en el 2021, en el cual desarrollan una IA, la cual consta de tres sistemas de aprendizaje profundo (DL) para el diagnóstico de estrabismo, la evaluación del ángulo y la planificación de operaciones basada en fotografías de reflexión de la luz corneal. Con el fin de entrenar la IA se tomaron dos fotos de cada paciente con estrabismo alterno, una con el ojo derecho mirando fijamente y la otra con el izquierdo [4]. Este proyecto describe de buena manera lo que se desea realizar en este proyecto el uso de inteligencia artificial para detectar estrabismo mediante el uso de fotografía usando una red neuronal convolucional(CNN) InceptionResNetV2(CNN) InceptionResNetV2 [23].

Durante el año 2003, el estrabismo llega a definirse como un trastorno ocular relativamente frecuente en la población infantil. Es por esto que se realiza un estudio para evaluar la prevalencia del estrabismo y los factores asociados en una muestra representativa en una población de mil setecientos treinta y nueve niños de niños de 6 años de Australia. Dentro de los resultados se diagnostica 48 niños (2.8% de la población) con estrabismo, de los cuales 5 habían sido sometidos previamente a corrección quirúrgica, 26 niños (54%) tenían esotropía, 14 (29%) exotropía, 7 (15%) microestrabismo y 1 niño tenía parálisis del nervio craneal VI [24].

Es importante tener en cuenta los posibles predictores del resultado motor tras la cirugía correctiva de la esotropía comitante adquirida. Por ende Rehab R Kassem, realiza un estudio donde incluyó a 40 pacientes con esotropía comitante adquirida, de los cuales tuvo en cuenta una evaluación del estado visual, sensorial y estado motor; refracción ciclopléjica, examen del fondo de ojo, los cuales fueron sometidos a cirugía correctiva del estrabismo. Llegando a la conclusión que las funciones sensoriales y motoras preoperatorias y postoperatorias son predictores del resultado motor postoperatorio a los 6 meses[25].

Con respecto a softwares, uno de los desarrollados actualmente es el software llamado ‘strabochek’, el cual busca realizar una evaluación de la desviación ocular de forma horizontal. Desarrollado por FNRO, patrocinado por FNRO, AFSOP, SFO, INSERM y LISSAC OPTIC 2000, del departamento de Oftalmología, Hospital Universitario de Nantes, Nantes, Francia. A tener en cuenta, este software requiere de fotografías tomadas con un flash colocado sobre la lente, desde una distancia de aproximadamente 1 metro, ampliadas en la cara de un oído al otro, centradas en la parte inferior de la nariz, reflejos claros de las córneas, siendo necesarias tres fotos, una monocular del ojo derecho, una monocular del ojo izquierdo y prismáticos. Este software nos permite obtener la evaluación sobre el paciente, sin la necesidad de almacenar ni recopilar datos personales de los pacientes. [26].

Con respecto a equipos para el estudio del estrabismo o desviación ocular, actualmente existen dos muy relevantes. Comenzando por el sinoptóforo (ver figura 7), el cual es un haploscopio usado para la evaluación y el diagnóstico del estrabismo de forma objetiva y subjetiva, estudio de la visión binocular y la realización de ejercicios que incrementen la amplitud de fusión en convergencia, este sistema utiliza prismas, que permiten colocar el objeto en línea recta con respecto al ojo del paciente, de tal forma que se posicione alineado a cada uno de los ojos, en vez de ser el ojo el que busque el objeto. Estos procesos se dan por medio de tres tipos de test, primero la percepción simultánea, siendo esta parejas de imágenes distintas que se sitúan frente a cada

ojo, segundo la fusión, donde las parejas de figuras similares con una parte diferente se sitúan cada una a un ojo, buscando que las figuras de ambos ojos se fusionen en el cerebro y se perciba la figura con ambas partes, existen 3 grados de visión: periférica, macular y foveal y estas se utilizan para realizar ejercicios de incremento de amplitud de fusión en convergencia y como tercer test la estereopsis, donde se muestra un columpio y el paciente debe decir si se balancea ya sea hacia el o hacia el examinador[27].



Figura 7. Paciente mientras se le realiza la prueba con un Sinoptóforo en consulta. [27].

Al paciente se le realizan los tres test en el orden dado anteriormente y si realiza de forma satisfactoria los test, se puede decir que su visión binocular es normal. Este aparato calibra en dioptrías prismáticas la posición relativa del objeto, por lo cual, si el objeto se mueve este puede medir cuánta desviación habrá respecto a la posición recta. El sinoptóforo es útil para medir los distintos componentes de una desviación: horizontal (endo o exo-desviación), vertical (hiper o hipo-desviación) y torsional (inciclo o exciclo-desviación). También para estudiar la visión binocular y realizar ejercicios de incremento de amplitud de fusión en convergencia [27]. A tener en cuenta el sinoptóforo permite la medición del ángulo de desviación en pacientes que tienen percepción simultánea, es decir pacientes que se presentan con estrabismo en la edad adulta, ya que el estrabismo que se presenta desde la infancia produce supresión y por tanto alteración de la visión binocular, lo cual no permite realizar este test.

Como segundo equipo se encuentra la videooculografía Gazelab, siendo este uno de los métodos más modernos para el estudio y el diagnóstico del tipo de la desviación ocular, este sistema permite medir y diagnosticar con más precisión las desviaciones complejas de la mirada en diferentes direcciones. Este equipo consta de dos cámaras con visión infrarroja, las cuáles permiten capturar los movimientos oculares en todo momento y cuenta con un láser que proyecta un patrón en retícula cuadrada.. Con respecto al funcionamiento del Gazelab, este se basa en el registro de imágenes en video de las pupilas y el procesamiento posterior por medio de un sistema de visión infrarrojo dado por las dos cámaras nombradas anteriormente, mediado por un ordenador que permite detectar la pupila y todas las características de los puntos del ojo. Los componentes del sistema Gazelad son, un proyector láser, dos cámaras de visión infrarroja, luz LED IR y un vidrio infrarrojo/espejo[28].

Este equipo realiza diferentes exámenes, que se dan por medio de tres módulos. Primero la videooculografía para estudio del estrabismo, dentro de este módulo, se realizan varios tipos de exámenes, ayudándose de la proyección de la retícula cuadrada en una superficie clara para evaluar, captando la posición de ambos ojos en el mismo punto en el espacio. Para lograr esto, el paciente debe mirar a cada una de las esquinas del cuadro mientras el equipo recoge la información de la posición a la que mira cada ojo y así calcula su desviación. De esta forma, se calcula el valor del estrabismo en las nueve posiciones de forma objetiva y precisa[28].

Como segundo módulo encontramos el video-pupilometría, mediante el sistema GazeLab Pupil Dynamic se graba videos y datos de ambos diámetros de pupilas mientras uno de ellos se estimula con luz y movimientos (Ver Figura 8). Este módulo es una buena herramienta para evaluar las pupilas con precisión, para esta prueba se graba las pupilas mientras el examinador realiza el examen de éstas y como resultado se muestra la diferencia entre tamaños que pueda guiar hacia el diagnóstico de un problema neurológico.

Y como tercer módulo, el video-nistagmometría, en este módulo el sistema graba videos y datos de la motilidad ocular en pacientes con enfermedades por nistagmo, músculos paréticos, estrabismo. Además de

evaluar el nistagmus en todas las posiciones de mirada, y su magnitud, puede conocer la posición de bloqueo de ambos ojos[28].

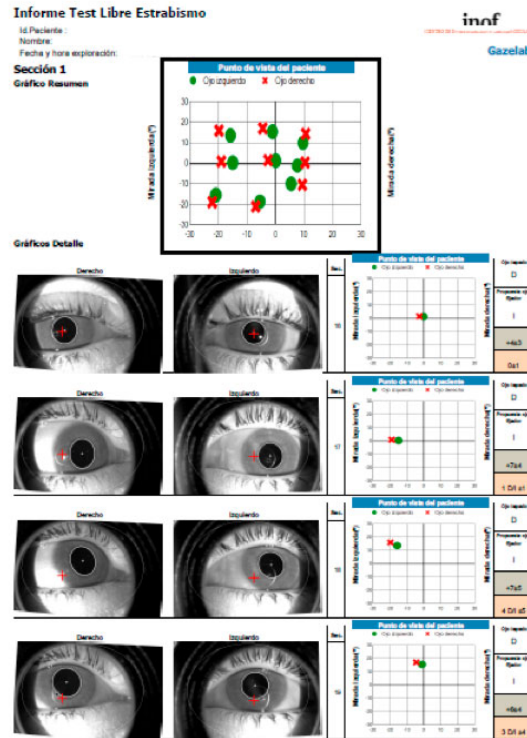


Figura 8. Ejemplo de informe que realiza el Gazelab. [28].

Capítulo III

Metodología

El presente capítulo evidencia la estructura de trabajo establecida para desarrollar el proyecto planteado, tal como se muestra en la figura 8. El desarrollo de un software para la clasificación del paralelismo y desviación ocular mediante el reconocimiento del Reflejo Corneal, se debe abordar con diferentes actividades que permitan el desarrollo paso a paso de los objetivos propuestos en el proyecto.

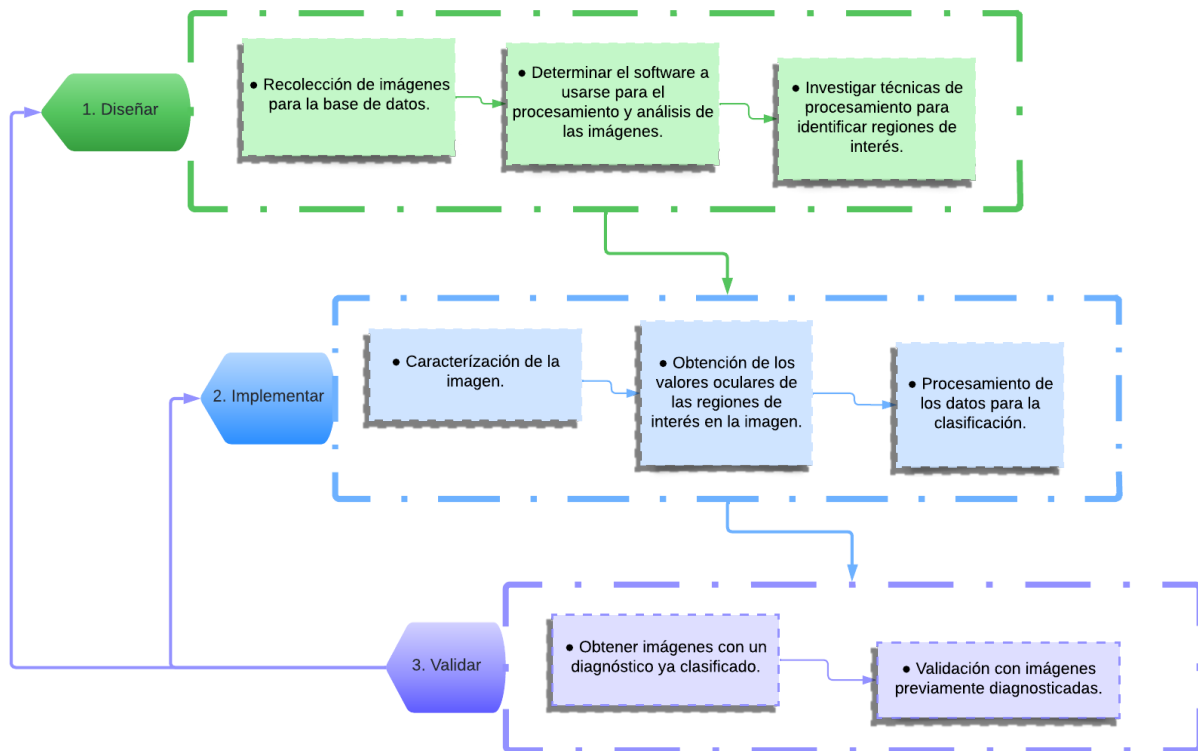


Figura 9. Diagrama de bloques de la metodología aplicada

La figura 9 ilustra los procesos que se plantearon para culminar con cada uno de los objetivos específicos del proyecto. A continuación se mostrará con más detalle cada una de las etapas.

3.1 Diseño

3.1.1 Recolección de imágenes para la base de datos

Se realiza una búsqueda por medio de la plataforma PubMed usando algoritmos de búsqueda con la terminología encontrada en Mesh, obteniendo artículos relacionados con los tipos de Estrabismo, de los cuales se extrae el número de imágenes necesarias para la realización de la base de datos, teniendo en cuenta también el uso de imágenes de libros de soporte bibliográfico universitario del tema.

3.1.2 Determinar el Software a usarse para el procesamiento y análisis de las imágenes

Se realiza un análisis del posible software a usar que permita el procesamiento y análisis de las imágenes, bajo criterios de compatibilidad de programación, capacidad computacional y permisos o licencias de uso, cumpliendo con los parámetros que permitan obtener un mejor resultado. Al final se emplea una matriz de evaluación para la selección de la mejor alternativa.

3.1.3 Investigar técnicas de procesamiento para identificar regiones de interés

Se realiza el análisis de diferentes técnicas de procesamiento de imágenes, para el reconocimiento de regiones de interés y patrones de imágenes, así como su consumo computacional para poder ser usado en la identificación y clasificación de las desviaciones oculares.

3.2 Implementar

3.2.1 Caracterización de la imagen

Se debe cargar al Software la imagen desde la base de datos para su adecuación, con el fin de poder identificar las estructuras del Iris, Pupila y Reflejo Corneal para la obtención de los valores a ser analizados.

3.2.2 Obtención de los valores oculares de las regiones de interés en la imagen.

Con la identificación de la estructura de la Pupila, podemos medir el Arco Ocular, el perímetro de la Pupila, distancia entre las Pupilas y estimar el tamaño de la órbita, así mismo con el Reflejo Corneal se calcula el Ángulo Kappa y distancias entre los Reflejos Corneales

3.2.3 Procesamiento de los datos para la clasificación

Con los valores oculares obtenidos de la identificación de las regiones de interés se realiza un análisis vectorial en planos cartesianos para la clasificación de nuestra base de datos de desviación ocular.

3.3 Validar

3.3.1 Obtener imágenes con un diagnóstico ya clasificado

Se realizará una búsqueda de imágenes con diagnósticos ya realizados a personas con desviaciones oculares.

3.3.2 Validación con imágenes previamente diagnosticadas

Una vez se obtengan los resultados del Software que se desarrolló se debe realizar una comparación con los valores obtenidos de la base de datos, con el fin de corroborar la precisión del Software a la hora de realizar la clasificación.

Capítulo IV

Resultados y Análisis de Resultados

Al llevar a cabo la metodología propuesta, se obtienen resultados que por medio de un análisis permiten validar y evaluar el cumplimiento de cada uno de los objetivos planteados en el proyecto, se exponen en el siguiente orden: i) Diseño, ii) Implementación, iii) Validación.

4.1 Diseño

4.1.1. Recolección de imágenes para la base de datos

Se lleva a cabo la búsqueda de bancos de imágenes de pacientes con desviación ocular buscando que cumplieran los criterios establecidos en la metodología, los cuales se seleccionaron de diferentes artículos y bases de datos(Eye disease dataset,siblings, Columbia)[29-30-31].

Para el adecuado tratamiento de las imágenes se debe realizar un tamizaje de las mismas buscando que cuenten con las características o rasgos necesarios y que contengan los parámetros propuestos para la realización de este proyecto (que cuente con los 3 elementos Iris, Pupila, y Reflejo Corneal). En el proceso de selección de imágenes muchas de estas no eran aptas para su uso debido a su calidad deficiente y que no contaban con las 3 regiones de interés buscadas, por ende se seleccionaron solo un pequeño grupo de imágenes para realizar las pruebas al código en sus primeras etapas para identificar el Iris y la Pupila. Al final la base de datos cuenta con 11 imágenes de pacientes nombrados como Paciente 1 al Paciente 11 (Ver Figura 10).

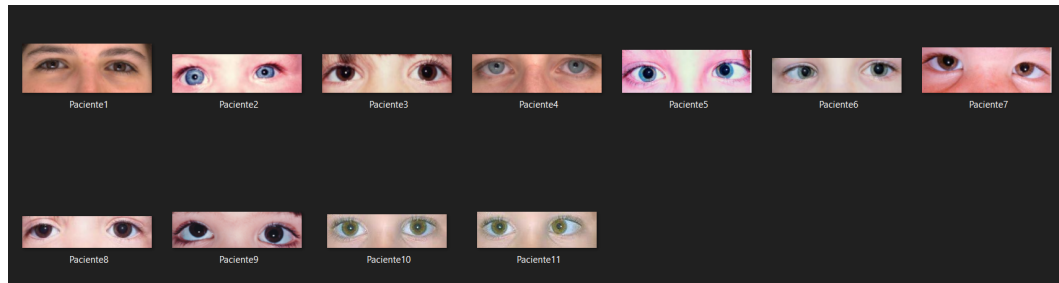


Figura 10. Base de datos de imágenes a usar para pruebas en el Software.

4.1.2 Selección del software a utilizar

Para determinar el software a usarse para el desarrollo de este proyecto, se tuvo en cuenta dos posibles opciones, Matlab y Python. Para escogerlo se realizó una Matriz de Pugh donde se tuvo en cuenta cinco criterios: comparación de los lenguajes, base de usuarios, soporte y documentación, indexación y coste del software.

Conceptos (Software a escoger)			
Software		Matlab	Phyton
CRITERIOS	Comparación de los lenguajes	1	-1
	Base de usuarios	-1	1
	Soporte y documentación	1	-1
	Indexacion	1	-1
	Coste del software	-1	1
		1	-1

Tabla 1. Matriz de Pugh para comparar dos software, Matlab y Python.

Con respecto a la comparación de los lenguajes, Python y Matlab se pueden usar tanto de forma interactiva como para desarrollar aplicaciones a gran escala. El lenguaje de Matlab es el primer lenguaje de programación de muchos profesionales de ingeniería y ciencia, debido a que usa matemática matricial orientada de Arrays que facilitan su aprendizaje y aplicación a la resolución de problemas tanto científicos como de ingeniería. Python tiene un lenguaje central y bibliotecas que se ejecutan de la misma forma que en todas las plataformas, por ende puede programar en entornos mixtos mientras que Matlab tiene un lenguaje de programación

comercial que es orientado a las matemáticas y las matrices. Este tiene cajas de herramientas especializadas para ingeniería y ciencia puesto que tiene una funcionalidad numérica más completa y especializada que Python.

Ambos software cuentan con una gran base de usuarios, a fecha de mayo de 2022 según encuestas de LinkedIn, Python cuenta con un aproximado de 7,6 millones de usuarios, mientras que Matlab cuenta con 4,1 millones de usuarios.

Por parte de la documentación y soporte, se puede resaltar que Matlab cuenta con licencias que incluyen soporte en vivo gratuito para los usuarios, estos pueden realizar llamadas por teléfono o enviar E-mails para obtener ayuda por parte del equipo de Matlab sobre el proyecto que están llevando a cabo. Además brinda el soporte gratuito de MathWorks y Matlab Answers el cual es similar a Stack Overflow, siendo este el que ofrece Python.

Con respecto a la indexación, Python utiliza una basada en 0, siendo esto, el uso de matrices que se indexan desde 0 para proporcionar una coherencia con otra familia de lenguajes, como lo son C,C++ y java, mientras que Matlab usa la indexación basada en 1.

Una ventaja de Python es que sus librerías se pueden descargar y usar de forma gratuita, teniendo también la opción de usar Python a través de un servicio de pago. Matlab por lo contrario, no es gratis para personas comúnmente independientes, aunque es más accesible y su costo no es elevado. Este también ofrece la opción de acceso educativo para el personal de instituciones universitarias, aproximadamente ocho millones de personas lo usan de esta manera, muchas instituciones deciden invertir en Matlab porque buscan mayor productividad cuando usan herramientas creadas para fines específicos.

Luego de llevar a cabo la matriz se obtiene como resultado que Matlab es el software más adecuado para el desarrollo de nuestro proyecto, puesto que cuenta con características que buscamos para el desarrollo eficiente de nuestro software.

4.2 Implementación

4.2.1. Adecuación de la imagen e identificación del Iris

En esta fase, primero se selecciona la imagen a trabajar, en este caso una paciente con ojos de color oscuro (Ver Figura 11), al tener la imagen se carga al software y realizando un preprocesamiento inicial se recorta la imagen obteniendo solo la sección de interés necesaria para trabajar.



Figura 11. Paciente uno, imagen original para pruebas iniciales.

Como segundo paso, se realiza el histograma de la imagen y como tercero se aplica un filtro gaussiano, donde se realiza el mismo proceso expresado anteriormente, para así obtener la imagen en el tercer plano, pasando la imagen a escala de grises y se modifica su brillo (Ver Figura 12), buscando así resaltar la ROI (región de interés, por sus siglas en inglés), en este caso el Iris.



Figura 12. Paciente uno, imagen original en tercer plano, escala de grises modificando brillo.

Luego se procede a binarizar teniendo en cuenta los valores entre los cuales puede variar dicha binarización, buscando el porcentaje más adecuado, para así obtener una imagen con la mejor distribución (Ver Figura 13). La binarización varía entre los valores de 0,1 y 0,9.

Al obtener la imagen ya binarizada es necesario realizar un relleno de área a las partes blancas que quedan dentro de la ROI al realizar la binarización, quedando seleccionada de forma completa la region que se necesita.



Figura 13. Paciente uno, imagen binarizada.

Cuando se obtiene completamente el Iris visible en la imagen, se procede a invertir la binarización (Ver Figura 14), eliminar las áreas sobrantes y aplicar una erosión tipo disco para tener como resultado en toda la imagen solo el Iris.

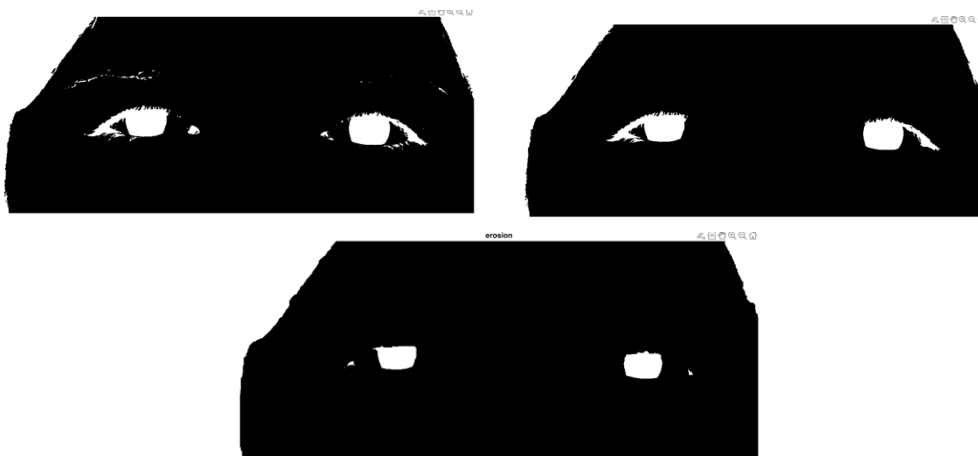


Figura 14. Paciente uno, imagen con binarización invertida y eliminación de áreas sobrantes para obtener el Iris.

4.2.2. Adecuación de la imagen e identificación de la pupila

Una vez teniendo la imagen seleccionada que se trabajó para encontrar el iris, se realiza el histograma de la imagen para mostrar la distribución de una única variable de escala, en este caso los colores. Como tercera parte se aplica un filtro Gaussiano, de tipo pasa altas y uno pasa bajas (Ver Figura 15), tomando la imagen en el plano tres, siendo este el plano rojo porque tiene una mayor gama de brillo que permite una mejor forma de modificar la imagen.

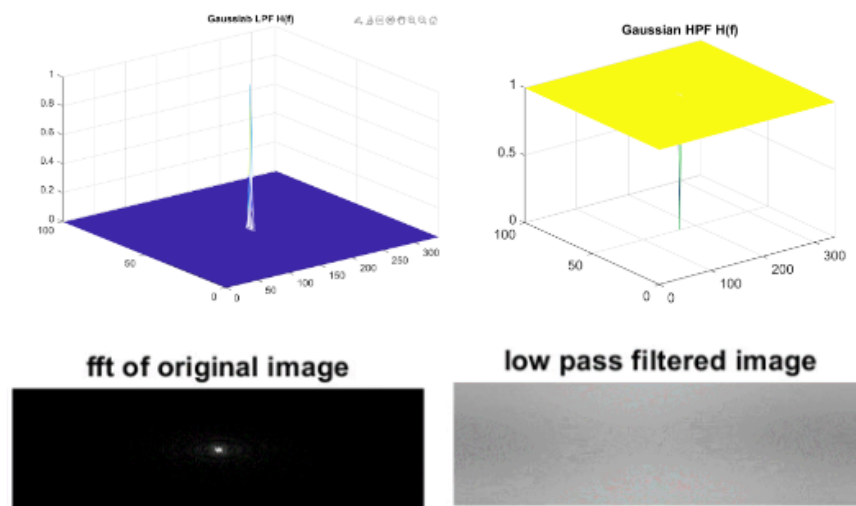


Figura 15. Filtro Gaussiano, donde se aplica un filtro pasa altas y un filtro pasa bajas.

Al tener la imagen en el tercer plano, se pasa la imagen a escala de grises y se comienza a modificar su brillo (Ver Figura 16), buscando así resaltar la nueva ROI, en este caso la Pupila.



Figura 16. Paciente uno, Imagen original en escala de grises.

Una vez obtenida la imagen, se aplica un binarizado, la cual puede variar dependiendo de la imagen y sus características, el color de sus ojos, el brillo de la imagen (Ver Figura 17). La binarización varía entre los valores de 0,1 y 0,9.



Figura 17. Paciente uno, Imagen en escala de grises binarizada.

Cuando obtenemos el valor de la binarización, la invertimos para obtener solo el contorno, y se elimina el área sobrante con la función `bwareaopen` para luego superponer la imagen original y así obtener el ROI buscado (Ver Figura 18), siendo este las Pupilas.

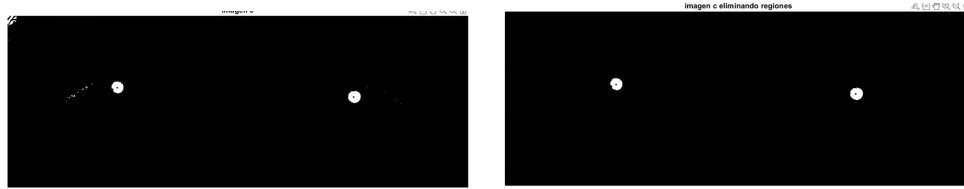


Figura 18. Paciente uno, imagen con binarización invertida y eliminación de áreas sobrantes para obtener la Pupila.

4.2.3. Adecuación de la imagen e identificación del Reflejo Corneal

Para la identificación del Reflejo Corneal se toma la imagen binarizada que se usó para encontrar la Pupila y se invierte el binarizado para resaltar el Reflejo Corneal (Ver Figura 19).



Figura 19. Paciente uno, imagen con binarización invertida para resaltar Reflejo Corneal.

4.2.4 Obtención de los valores oculares posibles con la identificación de las estructuras de los ojos en la imagen

Para obtener los valores necesarios, estos se basan en la búsqueda de los Centroides, los cuales buscan las partes blancas de la imagen para obtener con mayor claridad la información de interés. Para esto, se forma un círculo.

4.2.5 Iris

En el caso del Iris, se hallan los Centroides (Ver Figura 20) generando círculos específicos alrededor de las partes blancas del Iris de la imagen (Ver Figura 21).

```
s1 = 2x1 struct
```

Fields	Centroid
1	[267.6940,184.9440]
2	[861.5350,204.9151]


```
statsIRIS = 2x3 table
```

	Centroid		MajorAxisLength	MinorAxisLength
1	267.6940	184.9440	101.5007	61.0987
2	861.5350	204.9151	97.9692	67.0338

Figura 20. Paciente uno, centroides para resaltar el iris.



Figura 21. Paciente uno, círculos resaltando el Iris.

Al obtener los Centroides, se genera el contorno de las estructuras para obtener la región de interés de forma más exacta. Por último se superpone el contorno en la imagen original, y se generan círculos usando la función

“viscircles” (Ver Figura 22) a partir de los centroides antes encontrados y un radio que se acomoda a la estructura.



Figura 22. Paciente uno, círculos resaltando el Iris en la imagen inicial.

4.2.6 Pupila

En el caso de la Pupila, se hallan los Centroides (Ver Figura 23), generando círculos específicos alrededor de las partes blancas de la Pupila de la imagen (Ver Figura 24).

s1 = 2x1 struct

Fields	Centroid
1	[272.4632,177.3865]
2	[857.5722,201.1778]

statspupila = 2x3 table

	Centroid		MajorAxisLength	MinorAxisLength
1	272.4632	177.3865	30.7240	28.2542
2	857.5722	201.1778	31.6561	30.0564

Figura 23. Paciente uno, Centroides para resaltar la Pupila.



Figura 24. Paciente uno, círculos resaltando la Pupila.

Luego usando los centroides se genera un círculo usando la función “viscircles” que se acomoda al radio de la estructura, en este caso las Pupilas y se superpone a la imagen original (Ver Figura 25)..



Figura 25. Paciente uno, círculos resaltando la Pupila en la imagen inicial.

4.2.7 Reflejo Corneal

En el caso del Reflejo Corneal, se hallan los Centroides (Ver Figura 26) generando un punto en la coordenada correspondiente (Ver Figura 27), para determinar el tipo de desviación que tiene el paciente.

s2 = 4x1 struct

Fields	Centroid
1	[569.4706,213.0670]
2	[271.5000,178.5000]
3	[855.5000,201.5000]
4	[857,188.5000]

statsREFLEJ0 = 4x3 table

	Centroid		MajorAxisLength	MinorAxisLength
1	569.4706	213.0670	1.3144e+03	491.4240
2	271.5000	178.5000	4.0000	4
3	855.5000	201.5000	4.0000	4
4	857	188.5000	2.3094	1.1547

Figura 26. Paciente uno, centroides para resaltar el Reflejo Corneal.



Figura 27. Paciente uno, puntos blancos resaltando el Reflejo Corneal en la imagen inicial.

4.2.8 Análisis y procesamiento de imágenes para la clasificación de la desviación ocular.

Al obtener el ROI de cada una de las estructuras trabajadas anteriormente, como lo son Iris, Pupila, Reflejo Corneal, se lleva a cabo la suma de regiones en la imagen original, queriendo decir la sobreposición de estos ROI en la imagen inicial (Ver Figura 28), para así llevar a cabo la determinación de la desviación del paciente.



Figura 28. Paciente uno, suma de ROIs en la imagen inicial.

4.2.9 Diseño de la interfaz

La interfaz del software está diseñada para brindar facilidad al personal al momento de utilizarlo, siendo una herramienta específica para determinar qué tipo de desviación tiene el paciente, teniendo en cuenta las coordenadas de los ojos, con respecto al Iris, la Pupila y el Reflejo Corneal.

La aplicación consta de tres ventanas:

- Ventana de introducción al programa
- Ventana de introducción de datos del paciente
- Ventana de introducción al programa

Cada ventana contiene botones que poseen funciones específicas, permitiendo al profesional hacer un uso adecuado de esta herramienta. Al acceder a la aplicación la primera ventana en aparecer es la introducción con toda la información de los autores y colaboradores del programa.

Luego muestra la ventana de ingreso de información del paciente como el nombre, número de documento e imágenes del paciente.

La página principal del software es la que permite llevar a cabo todas las acciones (Ver Figura 29), gracias a los diez componentes con las funciones que desarrollan cada uno de ellos y que se describen a continuación:

1. **Información del paciente:** se encuentra en la parte superior izquierda del software, mostrando la información correspondiente a el nombre y género del paciente y la fecha.
2. **Load IMG:** este botón se encuentra en la parte superior izquierda, el cual nos permite cargar la imagen a editar.
3. **Clipping IMG:** este botón se encuentra en la parte superior izquierda, el cual nos permite recortar la imagen a editar, obteniendo solo la región de interés
4. **Botones de edición:** Esta sección de la interfaz contiene tres partes, dependiendo del ROI que se busca trabajar en la imagen inicial, con respecto a la Pupila, teniendo las opciones de Binarizado, Eliminar Área, Radio Pupila y Eliminar Área de la imagen binarizada. Con respecto al Iris, la opción de Tamaño y del Reflejo Corneal, las opciones de Binarizado y Eliminar Área
5. **Sum regiones:** Este botón, permite superponer los datos obtenidos en las diferentes regiones en la imagen inicial.
6. **Quadrant:** este botón, muestra un plano sobre el ojo y las coordenadas del los Reflejos Corneales, al mismo tiempo genera un plano cartesiano con los 2 puntos (el ojo de referencia ubicado en 0.0 y el Reflejo Corneal del ojo opuesto) y finalmente indica en la parte inferior derecha el tipo de desviación que posee.
7. **Visualización de imágenes:** en esta sección encontramos la información obtenida en la imagen dependiendo del ROI trabajado, en la parte inferior la imagen original, en la parte inferior izquierda la imagen resultante de Pupila, Iris y Reflejo Corneal.
8. **Espacio de edición:** en esta sección, centrada del software, encontramos la imagen que se está editando.

9. **Visualización de plano:** En esta sección de la interfaz se observa el plano cartesiano con los valores que nos permiten clasificar el tipo de desviación del paciente. En este se observa durante la edición un punto rojo que hace referencia al centroide del ojo derecho y un punto azul que hace referencia al Reflejo Corneal del ojo izquierdo
10. **Botones de Reset:** En esta sección de la interfaz se encuentran dos botones, los cuales permiten realizar acciones de reset en la aplicación.
11. **Tipo de desviación:** En esta sección de la interfaz se expone el tipo de desviación que presenta el paciente.

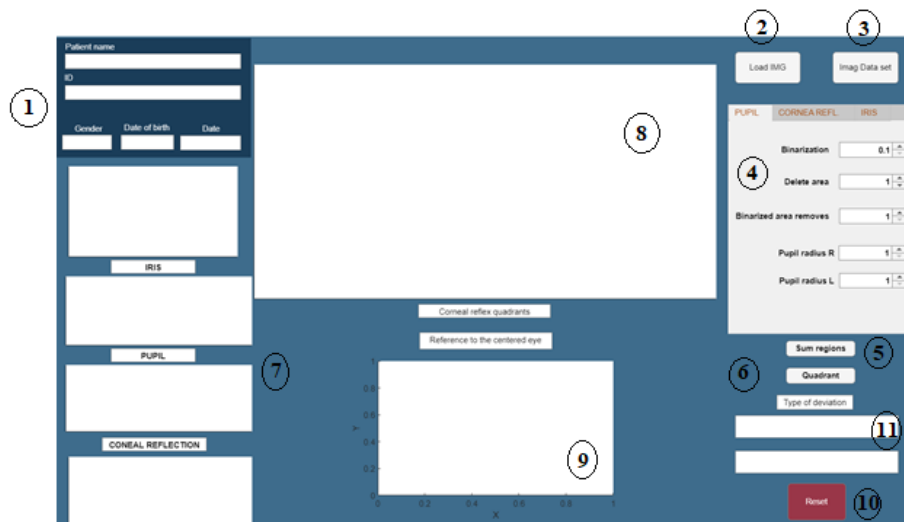


Figura 29. Interfaz principal del software.

Para tener una explicación paso a paso sobre el uso del software, se presenta la guía de manejo rápido como anexo #1 del presente documento.

4.3. Validar

4.3.1 Funcionamiento del software

La aplicación se obtiene como un ejecutable que genera Matlab, que permite ser instalado y usado en cualquier computador Windows, incluso si este no cuenta con el software Matlab instalado.

Al abrir el ejecutable se genera una primera ventana en el interfaz que corresponde a la información sobre los autores y colaboradores del proyecto, con su respectivo logo y el logo de la institución (Ver Figura 30).



Figura 30. Primera ventana de la interfaz, que incluye datos de autores y colaboradores.

Al hacer clic en next se abre la siguiente ventana de la interfaz en la cual se digita los datos del paciente a registrar (Ver Figura 31).

The image shows a user interface for 'STRAB VISION'. It features a dark blue background with white text and input fields. At the top left, there is a logo for 'unab' (Universidad Nacional Autónoma de Bogotá) with the text 'VIGILADA MINEDUCACIÓN' below it. To the right of the logo is a circular graphic consisting of several concentric, slightly offset lines. Below this graphic, the title 'STRAB VISION' is displayed in a large, bold, white font. Underneath the title, there are five input fields for patient information: 'Patient name', 'ID', 'Gender' (a dropdown menu with a small downward arrow), 'Date of birth', and 'Date'. At the bottom center of the form, there is a white button with the text 'NEXT' in blue.

Figura 31. Segunda ventana de la interfaz, permite ingresar datos del paciente.

Al ingresar los datos del paciente, proseguimos a abrir la ventana principal de la interfaz en la cual se carga la imagen y se modifican los parámetros necesarios para identificar cada una de las estructuras. Al ingresar a la ventana, se carga la imagen del paciente y se realiza el recorte (solo si es necesario) con el fin de mostrar únicamente la zona de los ojos. Luego se procede a identificar cada una de las estructuras usando los parámetros de edición de la aplicación, en el orden en que se muestra en la pestaña, Pupila, Reflejo Corneal e Iris (NOTA: se debe identificar correctamente cada una de las estructuras para el correcto funcionamiento de la aplicación y una correcta clasificación). Como siguiente paso, se suman las regiones identificadas anteriormente en la imagen original. Como paso final, se usa el botón llamado *quadrant* para generar un plano en los ojos con las coordenadas de los Reflejos Corneales, adicionalmente el software toma de referencia la Pupila de mayor radio, como la Pupila más centrada (ojo sano) y esta se usa para identificar en el ojo opuesto el tipo de desviación que este presenta.

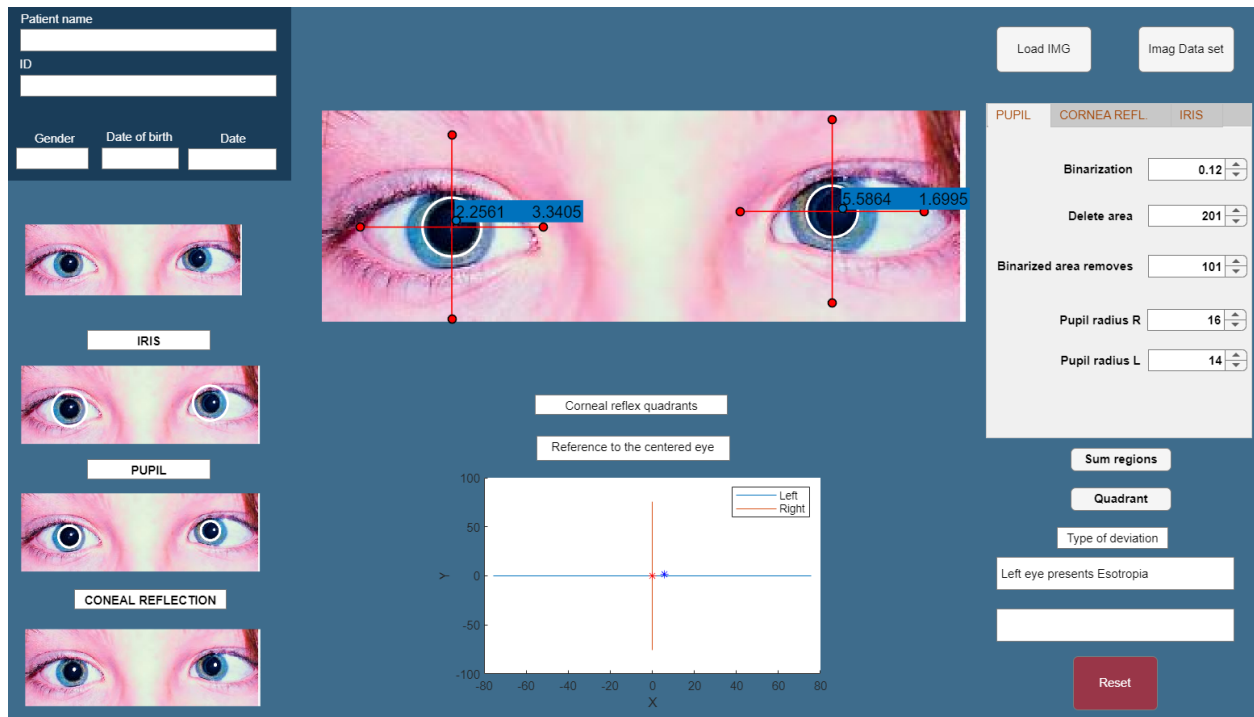


Figura 32. Primera prueba, paciente cinco.

En este caso, para el paciente cinco al cual se le hizo la prueba (Ver Figura 32), se tomó como ojo de referencia el ojo derecho ya que el radio de este es de 16 mientras que el del izquierdo es de 14, lo cual indica que el ojo derecho está más centrado que el izquierdo. En la parte inferior de la imagen se observa un plano cartesiano donde el punto rojo hace referencia al Centroides del ojo derecho y el punto azul es el Reflejo Corneal del ojo izquierdo, se puede observar que el Reflejo Corneal se encuentra desplazado hacia la derecha, por lo tanto el ojo se desplaza en el sentido contrario, hacia la parte central del rostro. obteniendo como resultado que el ojo izquierdo presenta Endotropia.

Si se desea trabajar con otro paciente, se da clic en el botón reset, y esto nos lleva a la segunda ventana del interfaz, donde ingresamos los datos del paciente.

4.3.2 Pruebas y Análisis de resultados

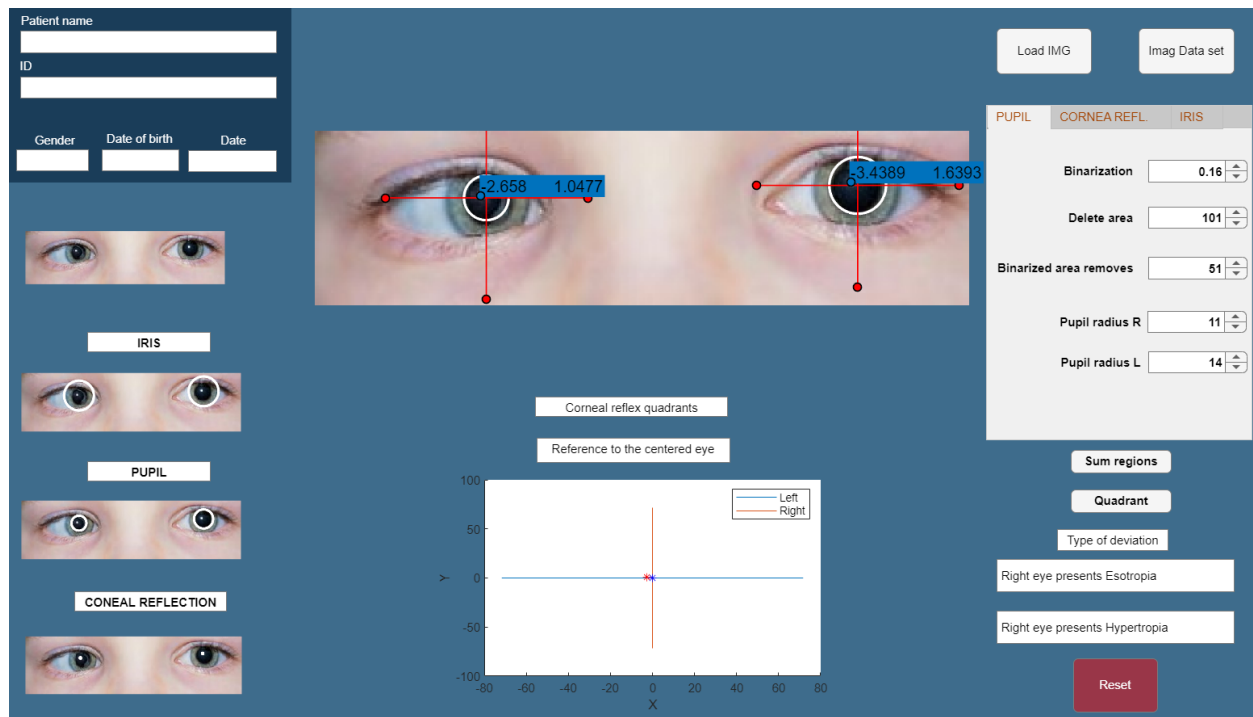


Figura 33. Segunda prueba, paciente seis.

Para el paciente seis se realizó el mismo proceso anterior y se generó su clasificación (Ver Figura 33), en este caso el ojo de referencia es el izquierdo y se observa que el ojo derecho presenta una Esotropia ya que el ojo se desvía hacia la zona central del rostro. Adicionalmente el software identifica que el paciente presenta también una pequeña Hipertropia, que no se aprecia a simple vista en la imagen, esto se debe a que los parámetros de software no se encuentran debidamente calibrados, puesto que la base de datos que se usó cuenta con imágenes que no se ajustan a un mismo protocolo, por ende las imágenes son de diferentes tamaños (resolución) y al realizar la programación para poderlas clasificar, no se puede estandarizar un código que clasifique cada una de las imágenes con precisión.

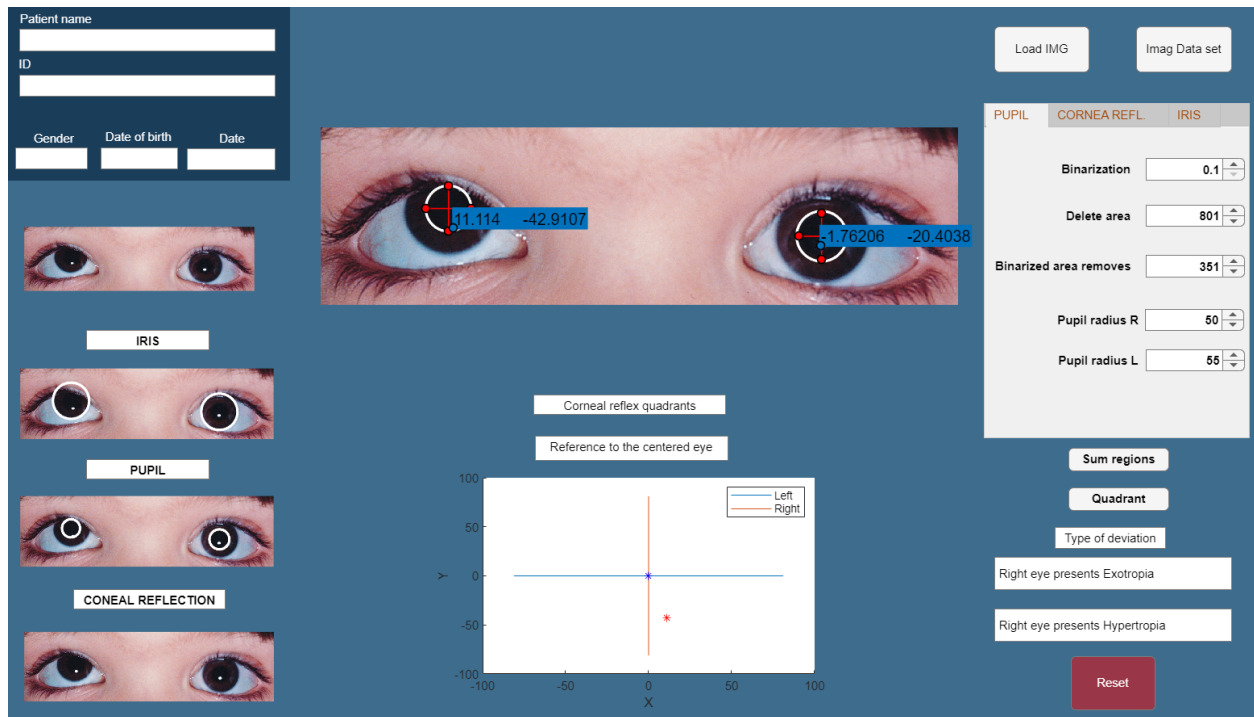


Figura 34. Tercera prueba, paciente nueve.

Para este paciente nueve se realizó el mismo proceso anterior y se generó su clasificación, en este caso el ojo de referencia es el izquierdo y se observa que el ojo derecho presenta una evidente Hipertropia ya que el Reflejo Corneal, como se observa en la gráfica, se desplaza hacia la parte negativa del eje y, por lo tanto el ojo se desplaza en el sentido contrario, por tanto el ojo tiene una desviación hacia la parte superior del rostro (Ver Figura 34)). Para este caso la Exotropia que se muestra en la imagen, se debe a la misma situación expuesta en el caso anterior.

Para la siguiente prueba se tiene en cuenta el paciente diez. Sus imágenes son tomadas de una base de datos donde se encuentra la foto del paciente antes y después de la cirugía para la corrección de su desviación ocular.

Primero se realiza la prueba teniendo la imagen del paciente diez donde no ha sido corregida su desviación ocular y cuenta con esotropía (Ver Figura 35). (al introducirla al software fue necesario realizar un pequeño recorte para eliminar el nombre de esotropía ya que este dañaba la binarización y de igual forma se realizó el recorte para el paciente después de cirugía)

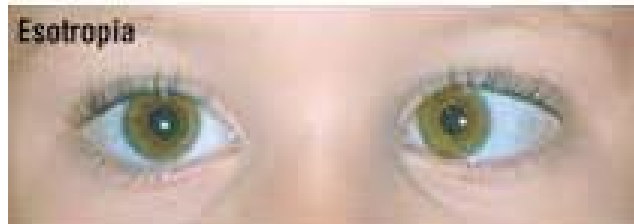


Figura 35. Cuarta prueba, paciente diez con Esotropía [28]

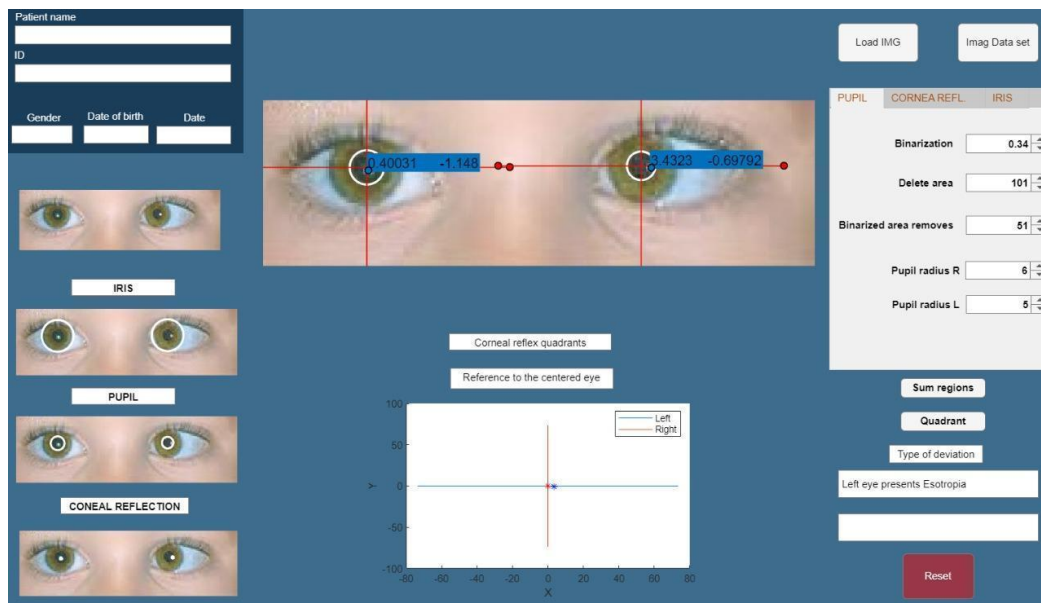


Figura 36. Cuarta prueba, paciente diez con esotropía.

Para este paciente diez se realizó el mismo proceso que con los pacientes anteriores y se generó su clasificación, en este caso el ojo de referencia es el derecho puesto que presenta un radio mayor al estar más centrado y se observa que el ojo izquierdo presenta una Esotropía ya que el ojo se desvía

hacia la zona central del rostro, dando así como resultado el mismo que fue expuesto por el profesional antes de realizar la cirugía para la corrección de la desviación del paciente (Ver Figura 36).

Como última prueba, se realiza con el mismo paciente, luego de realizarle la cirugía de corrección (Ver Figura 37).

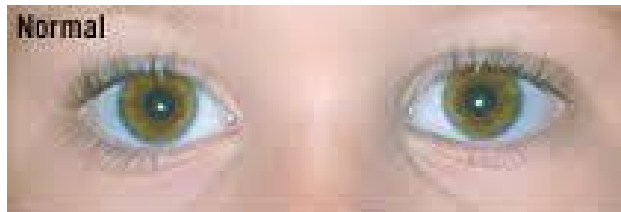


Figura 37. Quinta prueba, paciente diez, luego de cirugía de corrección [28]

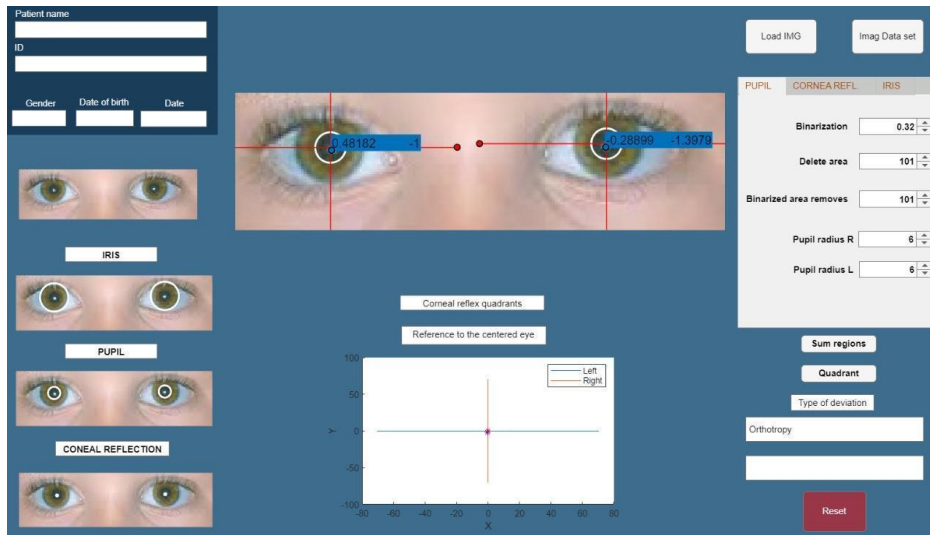


Figura 38. Cuarta prueba, paciente diez, Orthotropia.

Luego de la cirugía de corrección se realiza la misma prueba al paciente diez, y en este caso ambos ojos presentan un mismo radio, queriendo decir que ambos se encuentran centrados, dando como resultado un paciente ortotrópico, sin alteraciones o alguna desviación ocular (Ver Figura 38).

4.3.3 Tabla comparativa de diagnóstico definido vs clasificación del software

Al comparar los resultados obtenidos por el software con las imágenes de la base de datos previamente diagnosticadas, se observa que el software tiene asertividad del 100%. Se puede evidenciar que en algunos casos el software identifica dos tipos de desviación puesto que maneja valores muy pequeños e identifica desviaciones leves que no se diagnostican a plena vista.

Pacientes	Tipo de desviación	Clasificación por software	Asertividad
Paciente1	Normal	Normal	Si
Paciente2	Hipertropia	Hipertropia/Exotropia	Si
Paciente3	Exotropia	Exotropia	Si
Paciente4	Normal	Normal	Si
Paciente5	Esotropia	Esotropia	Si
Paciente6	Esotropia	Esotropia/Hipertropia	Si
Paciente7	Esotropia	Esotropia	Si
Paciente8	Exotropia	Exotropia/Hipertropia	Si
Paciente9	Hipertropia	Hipertropia/Exotropia	Si
Paciente10	Normal	Normal	Si
Paciente11	Esotropia	Esotropia	Si

Tabla 2. Comparativa diagnóstico Vs clasificación.

Capítulo V

Conclusiones y Recomendaciones

El presente capítulo expone las conclusiones obtenidas en el proceso de planteamiento, ejecución y análisis del proyecto. Además de las recomendaciones para mejoras y trabajos futuros en esta área.

5.1 Conclusiones

Siguiendo los lineamientos establecidos en la metodología al inicio del proyecto, se realizó un proceso de investigación con respecto a los posibles software para su desarrollo, teniendo en cuenta las posibles opciones se llevó a cabo una matriz de Pugh siendo Matlab el software elegido, ya que se acoplaban a las necesidades del proyecto. El código se fue desarrollando a lo largo de todo el semestre, este se fue refinando teniendo en cuenta cada una de las estructuras a trabajar (Pupila, Reflejo Corneal e Iris) hasta llegar a obtener el código final funcional, para luego implementarlo en una interfaz compuesta por tres ventanas, la cual clasifica e identifica correctamente la desviación que presenta cada paciente estudiado. Sin embargo se tienen en cuenta las limitaciones presentadas anteriormente.

Dentro de los beneficios que ofrece la interfaz es que brinda una ayuda al profesional para que pueda realizar una clasificación de forma más rápida y exacta, buscando automatizar y reducir el tiempo de diagnóstico. Siendo una interfaz que permite realizar una identificación de la desviación del paciente, permitiendo calcular las regiones de interés de forma manual. Otro de sus beneficios es que Matlab permite compilar el software como un ejecutable .exe con todas las herramientas necesarias para ser

instalado en cualquier computador Windows sin necesidad de tener instalado el programa Matlab en la computadora que se va a ejecutar el software STRAB-VISION.

5.2 Recomendaciones

Se presenta como recomendación, si se desea continuar el proyecto en un futuro, el desarrollo de una base de datos propia, con un protocolo de toma de imágenes que permita implementar en el código parámetros específicos, para así obtener una clasificación más estandarizada y así poder agregar a futuro una cuantificación de la desviación de los resultados.

También se recomienda a un futuro tener en cuenta la implementación de técnicas de inteligencia artificial, que permitan la eliminación de pasos dentro de la aplicación, para que sea más rápida y exacta.

Capítulo VI

Referencias Bibliográficas

- [1] Gunter, K. Von. et al. “Strabismus”. 5ta edición, editorial Mosby St. Louis Missouri 1995.
- [2] José Perea. (2015, septiembre). DEFINICIONES DE ESTRABISMO. <http://www.doctorjoseperea.com/libros/images/libros/pdf/definiciones.pdf>
- [3] Jampolsky A. Norcia. “Basic and clinical of strabismus”. American Academy of Ophthalmology, section 6. San Francisco 1995.
- [4] Bmotik Group. (2019, 27 febrero). ESTRABISMO. Acope Colombia. <https://acopecolombia.org/estrabismo/#:%7E:text=Se%20calcula%20que%20aproximadamente%20se,tener%20alg%C3%BAn%20tipo%20de%20desviaci%C3%B3n>.
- [5] Eficacia de la sutura ajustable para la corrección del estrabismo en niños y adultos. (2017). Geraadpleegd op 2 mei2022, van <https://eprints.ucm.es/id/eprint/47121/1/T39812.pdf>
- [6] Avanzada, Á. O. (2020, julio 13). Fijación visual. Área Oftalmológica Avanzada. <https://areaoftalmologica.com/terminos-de-ofthalmologia/fijacion-visual/>
- [7](S/f). Studocu.com. Recuperado el 17 de marzo de 2022, de <https://www.studocu.com/cl/document/universidad-austral-de-chile/metodos-de-exploracion/test-de-hitscher-y-angulokappatest-de-hitscher-y-angulo-de-kappa/8750645>
- [8]María, L., & Negrete, C. (s/f). Estandarización de la prueba Cover Test mediante prueba piloto Estandarización de la prueba Cover Test mediante prueba piloto en niños de 5 a 15 años con el protocolo Reisvo en niños de 5 a 15 años con el protocolo Reisvo. Edu.co. Recuperado el 7 de diciembre de 2022, de https://ciencia.lasalle.edu.co/cgi/viewcontent.cgi?article=1049&context=maest_ciencias_vision
- [9]Shenny, L. [UCrIm2P4DPY0Csq1-fdeR_Mg]. (2014, agosto 17). TEST DE HIRSCHBERG Y Ángulo Kappa. Youtube.<https://www.youtube.com/watch?v=fCIRodlmVER> PROPUESTA DE PROYECTO DE GRADO
- [10]André Roth. (1999). Le strabisme concomitant: du strabisme aux syndromes strabiques
- [11] Dana Robaei,. (2022). Factores asociados con Estrabismo Infantil.
- [12]Avanzada, Á. O. (2020, 5 junio). Visión binocular. Área Oftalmológica Avanzada. <https://areaoftalmologica.com/neurooftalmologia/vision-binocular>

[13]Oftalvist. (2019, April 26). El sentido de la vista: Definición, partes y funcionamiento. Blog Oftalvist: Lo Último en Salud Ocular; Clínicas Oftalvist. <https://www.oftalvist.es/blog/sentido-de-la-vista/>

[14]Óptica por la Cara. (2011, February 26). Óptica por la Cara.<http://opticaporlacara.com/2011/02/el-ojo-humano-en-numeros/>

[15] O. (2019, 7 noviembre). Ver de cerca: la acomodación y la vista cansada. Ocularis. <https://ocularis.es/ver-de-cerca-la-acomodacion-y-la-vista-cansada/>

[16] Agarwal, A. (2011). Ángulo kappa podría jugar un rol importante en el éxito de los LIOs multifocales. Occular Surgery News. <https://www.healio.com/news/ophthalmology/20120424/ngulo-kappa-podra-jugar-un-rol-importante-en-el-xito-de-los-lios-multifocales>

[17] Unknown, & Perfil, V. T. mi. (s/f). Test empleados en optometría. Blogspot.com. Recuperado el 29 de abril de 2022, de <http://optometriaclinica05.blogspot.com/2017/10/angulo-kappa.html>

[18] Bosworth RG, Birch EE. Direction-of-motion detection and motion VEP asymmetries in normal children and children with infantile esotropia. Invest Ophthalmol Vis Sci. 2007 Dec;48(12):5523-31. doi: 10.1167/iovs.07-0666. PMID: 18055801.

[19]Rehab R Kassem. (2018). Factors affecting the motor outcome after corrective surgery for acquired comitant esotropia.

[20]Cl, I.-W. I. (2017, julio 31). EXAMEN OFTALMOLÓGICO INFANTIL. Escuela de Medicina. <https://medicina.uc.cl/publicacion/examen-oftalmologico-infantil/>

[21](S/f). Studocu.com. Recuperado el 17 de marzo de 2022, de <https://www.studocu.com/cl/document/universidad-austral-de-chile/metodos-de-exploracion/test-de-hitscher-y-angulokappatest-de-hitscher-y-angulo-de-kappa/8750645>

[22]Ley 23 de 1982 - Gestor Normativo - FunciÃ3n PÃoblica. (2015, 1 december).Geraadpleegd op 2 mei 2022, van <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=3431#:~:text=Esta%20Ley%20protege%20exclusivamente%20la,obras%20literarias%2C%20cient%3%ADficas%20y%20art%3%ADsticas.>

[23]Shenny, L. [UCrIm2P4DPY0Csq1-fdeR_Mg]. (2014, agosto 17). TEST DE HIRSCHBERG Y Ángulo Kappa. Youtube. <https://www.youtube.com/watch?v=fCIRodlmVEE>

[24]Avanzada, Á. O. (2020, julio 13). Fijación visual. Área Oftalmológica Avanzada.

<https://areaoftalmologica.com/terminos-de-oftalmologia/fijacion-visual/>

[25] Estándar IEC 62304 para el desarrollo de software de dispositivos médicos. (z.d.). MATLAB & Simulink. Geraadpleegd op 2 mei 2022, van <https://la.mathworks.com/discovery/iec-62304.html>

[26] Strabocheck. (s. f.). <http://strabocheck.com/>

[27] Sinoptóforo. (2021, 26 abril). Clínica Oftalmológica San Bernardo. <https://www.sanbernardoclinica.com/dictionary/sinoptoforo/>

[28] ¿Qué es y cómo funciona la videooculografía Gazelab? (2022, 22 mayo). Inof | Centro de investigación y cirugía ocular. <https://inof.es/que-es-y-como-funciona-la-videooculografia-gazelab/>

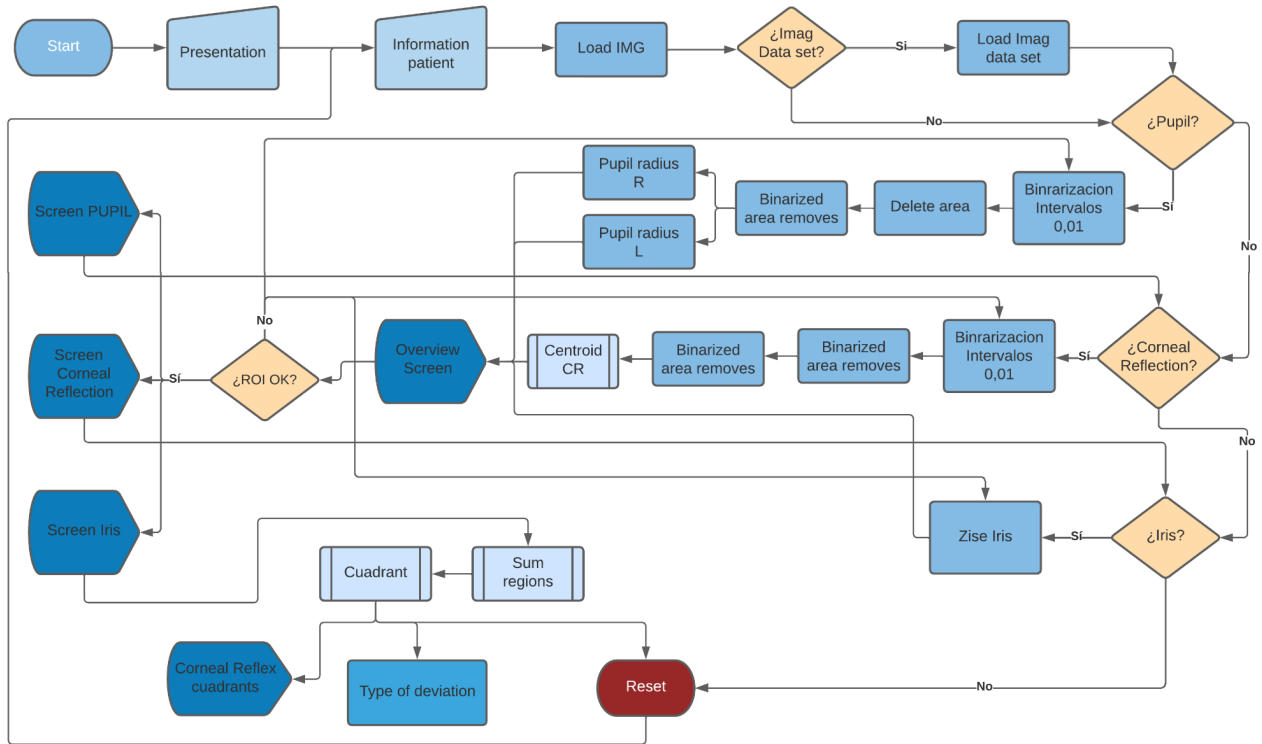
[29] Larxel. (2020). Ocular Disease Recognition [Data set].

[30] *CG&VG projects*. (s/f). Polito.It. Recuperado el 29 de julio de 2022, de <https://areaweb.polito.it/ricerca/cvg/siblingsDB.html>

[31] *CAVE*. (s/f). Columbia.edu. Recuperado el 29 de julio de 2022, de https://www.cs.columbia.edu/CAVE/databases/columbia_gaze/

Anexo 1

Diagrama de flujo software Strap-Vision



Anexo 2

GUIA RAPIDA PARA EL USO Y MANEJO DEL SOFTWARE STRAB-VISION PARA LA CLASIFICACIÓN DEL PARALELISMO Y DESVIACIÓN OCULAR MEDIANTE EL RECONOCIMIENTO DEL REFLEJO CORNEAL

La presente guía elaborada, ofrece a sus clientes una forma rápida para el buen uso y manejo del software la clasificación del paralelismo y desviación ocular mediante el reconocimiento del Reflejo Corneal.

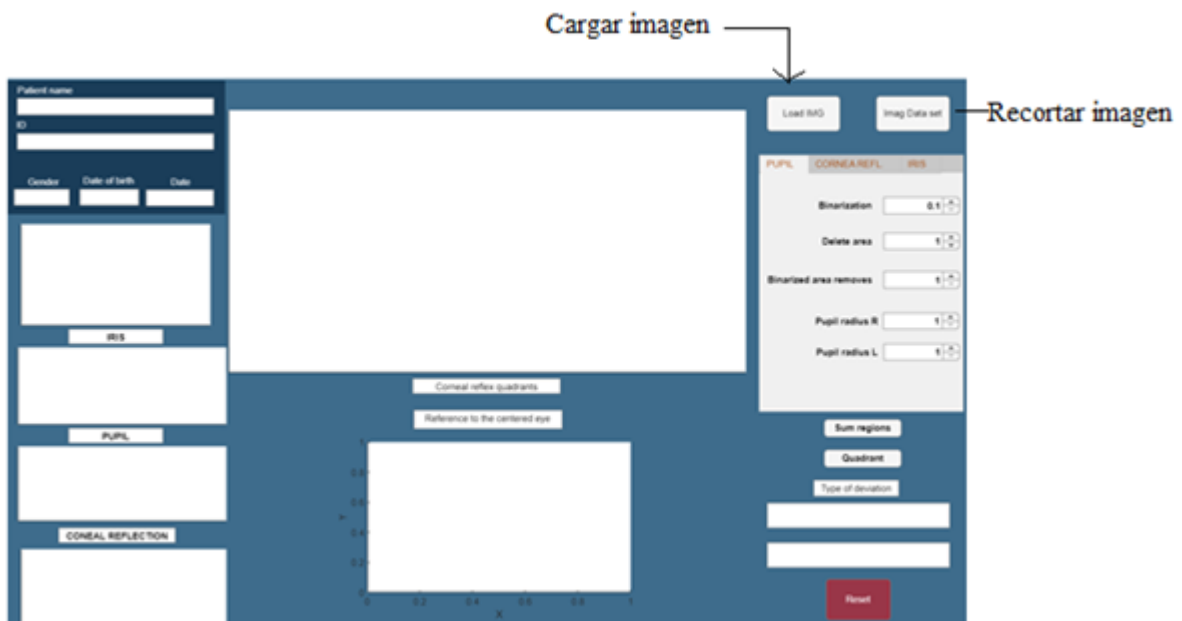
Al abrir el programa se genera una primera ventana en el interfaz que corresponde a la información sobre los autores y colaboradores del proyecto, con su respectivo logo y el logo de la institución.



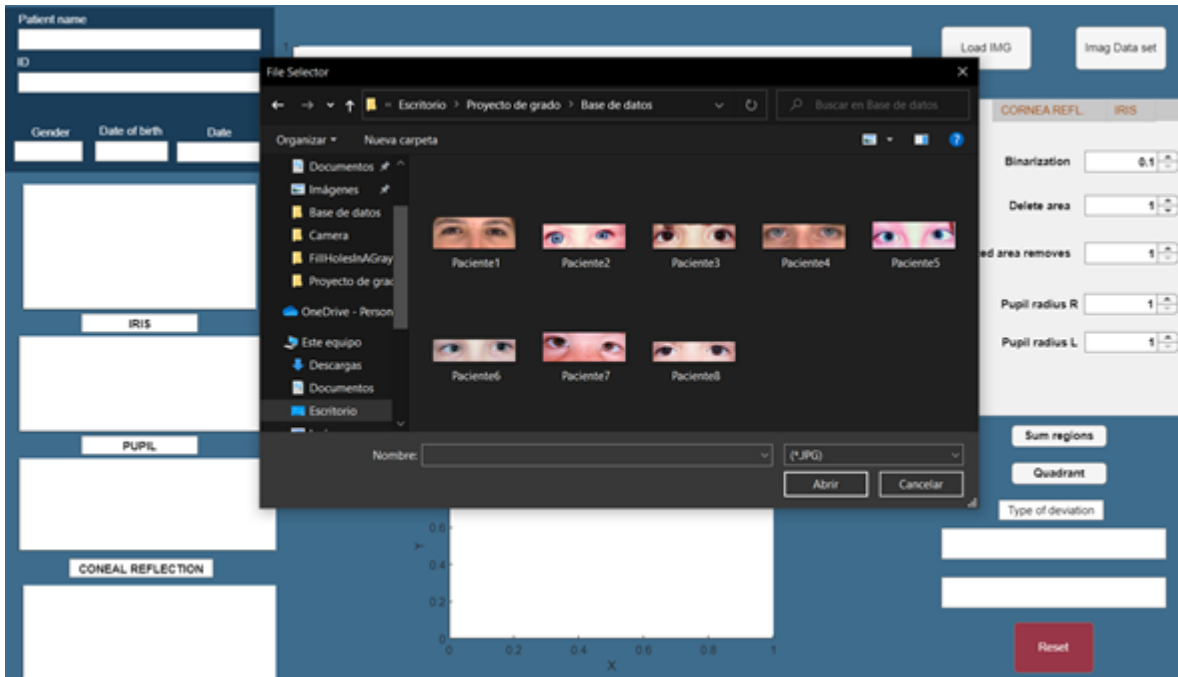
Al darle next, proceda a abrir la segunda ventana del programa STRAB-VISION. En esta llene los datos correspondientes al paciente a estudiar, siendo estos su nombre, cédula, género, fecha de nacimiento y fecha de registro.



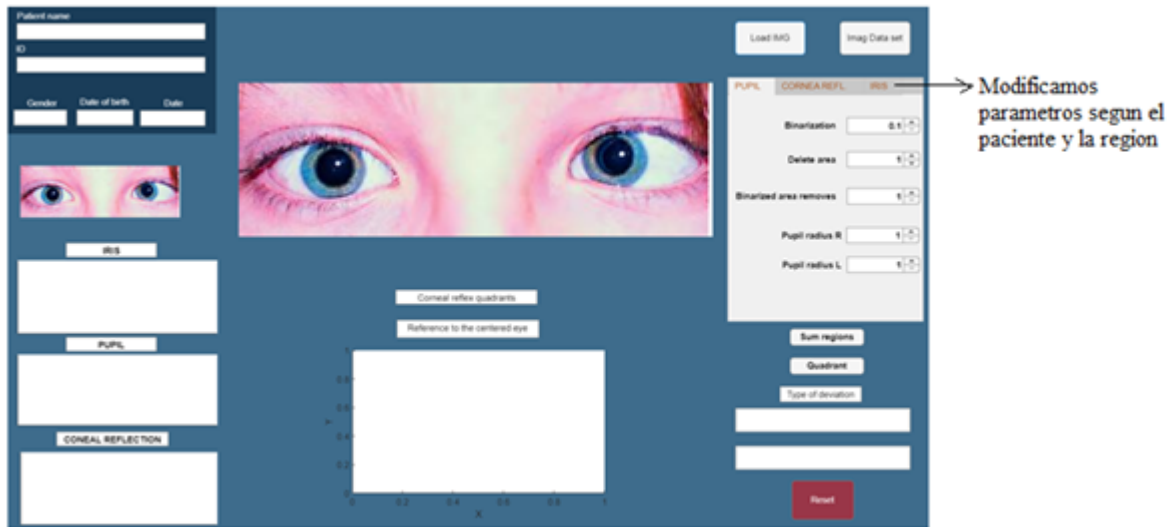
Una vez llene los datos del paciente proceda a darle *next*, para abrir la siguiente ventana principal del programa.



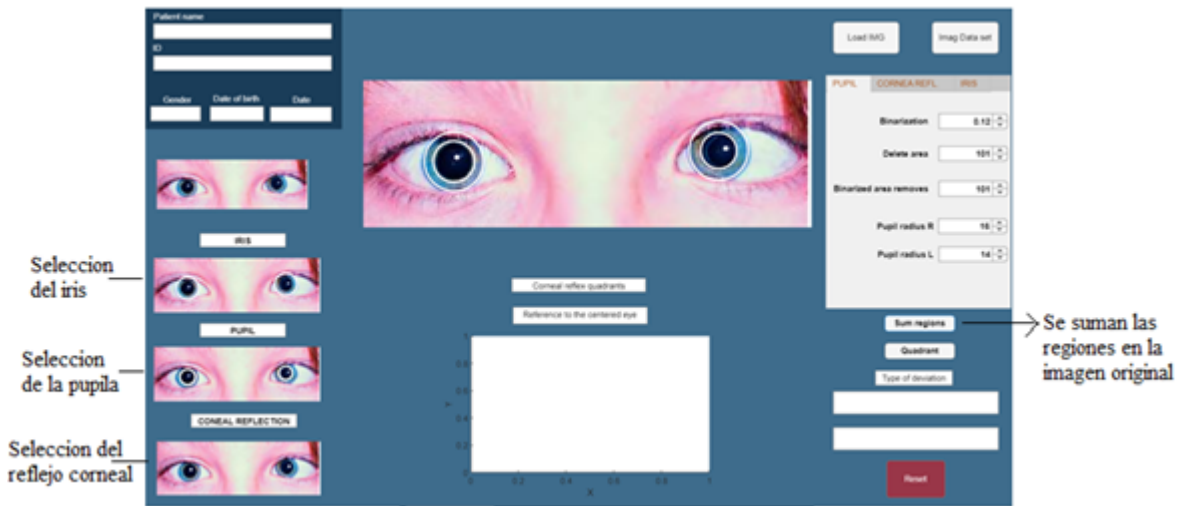
Una vez se abre la venta, proceda a cargar la imagen del paciente dando clic en el botón *Load IMG* y recortarla (solo si es necesario) dando clic en el botón *Imag Data set* con el fin de mostrar únicamente la región de interés a trabajar.



Una vez se carga la imagen, proceda a modificar los parámetros necesarios para identificar cada una de las estructuras (Pupila, Reflejo Corneal e Iris). (NOTA: se debe identificar correctamente cada una de las estructuras para el correcto funcionamiento de la aplicación y una correcta clasificación).

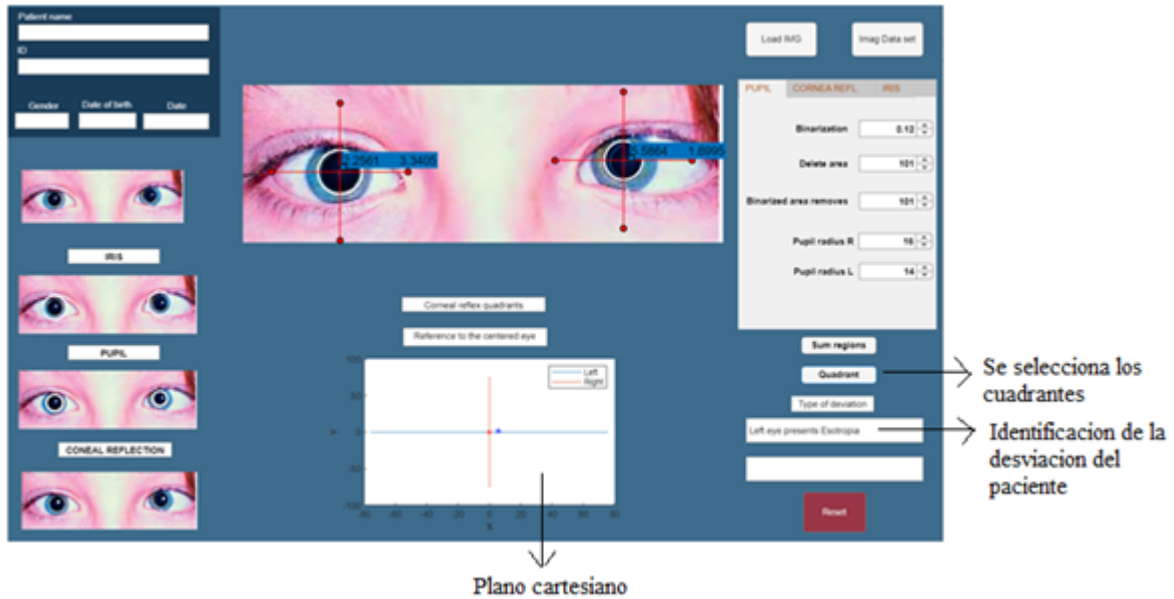


Como siguiente paso, da clic en el botón *sum regions* para sumar las regiones identificadas anteriormente en la imagen original.



Como paso final, se usa el botón llamado *Quadrant* para generar un plano en los ojos con las coordenadas de los Reflejos Corneales, adicionalmente el software toma de referencia la Pupila de mayor radio, como la Pupila más centrada (ojo sano) y esta se usa para identificar en el ojo opuesto el tipo de desviación que este presenta.

Una vez se da clic en el botón *Quadrant*, se puede observar un plano cartesiano con los valores que nos permiten clasificar el tipo de desviación del paciente. En este se observa durante la edición un punto rojo que hace referencia al centroide del ojo derecho y un punto azul que hace referencia al Reflejo Corneal del ojo izquierdo. Por último, se obtiene la clasificación del paciente y sus valores.



Por último, paso puede cerrar la ventana para finalizar, o dar clic en el botón reset para comenzar un nuevo estudio en otro paciente.

Anexo 3

Código interfaz de presentación

```
classdef Presentacion < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        UIFigure    matlab.ui.Figure
        Image2      matlab.ui.control.Image
        Image       matlab.ui.control.Image
        NEXTButton  matlab.ui.control.Button
        TextArea_2  matlab.ui.control.TextArea
    end

    % Callbacks that handle component events
    methods (Access = private)

        % Callback function
        function TextArea_3ValueChanged(app, event)

            end

        % Button pushed function: NEXTButton
        function NEXTButtonPushed(app, event)
            INTERFAZPRIMARIA
            delete(app)

            end

        % Close request function: UIFigure
        function UIFigureCloseRequest(app, event)

            end
        end

    % Component initialization
    methods (Access = private)
```



```

% Create UIFigure and components
function createComponents(app)

    % Get the file path for locating images
    pathToMLAPP = fileparts(mfilename('fullpath'));

    % Create UIFigure and hide until all components are created
    app.UIFigure = uifigure('Visible', 'off');
    app.UIFigure.Color = [0.102 0.2392 0.349];
    app.UIFigure.Position = [100 100 640 480];
    app.UIFigure.Name = 'MATLAB App';
    app.UIFigure.CloseRequestFcn = createCallbackFcn(app, @UIFigureCloseRequest, true);

    % Create TextArea_2
    app.TextArea_2 = uitextarea(app.UIFigure);
    app.TextArea_2.FontName = 'Dubai Light';
    app.TextArea_2.FontSize = 18;
    app.TextArea_2.FontWeight = 'bold';
    app.TextArea_2.FontColor = [1 1 1];
    app.TextArea_2.BackgroundColor = [0.102 0.2392 0.349];
    app.TextArea_2.Position = [10 96 622 127];
    app.TextArea_2.Value = {'Unab - Biomedical Engineering'; 'Elaborated by: John Rojas - Jose
Camargo - Maria Blanco'; 'Coordinated by: Ing. Manuel Franco - Dr. Angélica Prada - Ing. Edwing
Ayala'; ''};

    % Create NEXTButton
    app.NEXTButton = uibutton(app.UIFigure, 'push');
    app.NEXTButton.ButtonPushedFcn = createCallbackFcn(app, @NEXTButtonPushed, true);
    app.NEXTButton.Position = [271 54 100 23];
    app.NEXTButton.Text = 'NEXT';

    % Create Image
    app.Image = uiimage(app.UIFigure);
    app.Image.Position = [499 392 142 85];
    app.Image.ImageSource = fullfile(pathToMLAPP, 'logo_unabvirtual.png');

    % Create Image2
    app.Image2 = uiimage(app.UIFigure);
    app.Image2.Position = [92 231 439 235];
    app.Image2.ImageSource = fullfile(pathToMLAPP, 'logo.png');

    % Show the figure after all components are created
    app.UIFigure.Visible = 'on';

```

```

    end
end

% App creation and deletion
methods (Access = public)

    % Construct app
    function app = Presentacion

        % Create UIFigure and components
        createComponents(app)

        % Register the app with App Designer
        registerApp(app, app.UIFigure)

        if nargin == 0
            clear app
        end
    end
end

% Code that executes before app deletion
function delete(app)

    % Delete UIFigure when app is deleted
    delete(app.UIFigure)
end
end
end

```

Anexo 4

Código interfaz datos del paciente

```
classdef INTERFAZPRIMARIA < matlab.apps.AppBase

% Properties that correspond to app components
properties (Access = public)
    UIFigure          matlab.ui.Figure
    Image2            matlab.ui.control.Image
    NEXTButton       matlab.ui.control.Button
    DateEditField     matlab.ui.control.EditField
    DateEditFieldLabel  matlab.ui.control.Label
    DateofbirthEditField_2  matlab.ui.control.EditField
    DateofbirthEditField_2Label  matlab.ui.control.Label
    GenderDropDown   matlab.ui.control.DropDown
    GenderDropDownLabel  matlab.ui.control.Label
    IEditField       matlab.ui.control.EditField
    IEditFieldLabel  matlab.ui.control.Label
    PatientnameEditField  matlab.ui.control.EditField
    PatientnameEditFieldLabel  matlab.ui.control.Label
    Image            matlab.ui.control.Image
end

properties (Access = private)
    app2 % Description
end

% Callbacks that handle component events
methods (Access = private)

% Value changed function: PatientnameEditField
function PatientnameEditFieldValueChanged(app, event)
    value = app.PatientnameEditField.Value;

end

% Value changed function: IEditField
function IEditFieldValueChanged(app, event)
```

```

        value = app.IDEditField.Value;

    end

    % Value changed function: GenderDropDown
    function GenderDropDownValueChanged(app, event)
        value = app.GenderDropDown.Value;

    end

    % Value changed function: DateofbirthEditField_2
    function DateofbirthEditField_2ValueChanged(app, event)
        value = app.DateofbirthEditField_2.Value;

    end

    % Value changed function: DateEditField
    function DateEditFieldValueChanged(app, event)
        value = app.DateEditField.Value;

    end

    % Button pushed function: NEXTButton
    function NEXTButtonPushed(app, event)

app.app2=app2(app,app.PatientnameEditField.Value,app.IDEditField.Value,app.GenderDropDown.V
alue,app.DateofbirthEditField_2.Value,app.DateEditField.Value);
        delete(app)

    end

    % Close request function: UIFigure
    function UIFigureCloseRequest(app, event)

    end
end

% Component initialization
methods (Access = private)

    % Create UIFigure and components
    function createComponents(app)

```

```

% Get the file path for locating images
pathToMLAPP = fileparts(mfilename('fullpath'));

% Create UIFigure and hide until all components are created
app.UIFigure = uifigure('Visible', 'off');
app.UIFigure.Color = [0.102 0.2392 0.349];
app.UIFigure.Position = [100 100 640 480];
app.UIFigure.Name = 'MATLAB App';
app.UIFigure.CloseRequestFcn = createCallbackFcn(app, @UIFigureCloseRequest, true);

% Create Image
app.Image = uiimage(app.UIFigure);
app.Image.Position = [1 386 151 95];
app.Image.ImageSource = fullfile(pathToMLAPP, 'logo_unabvirtual.png');

% Create PatientnameEditFieldLabel
app.PatientnameEditFieldLabel = uilabel(app.UIFigure);
app.PatientnameEditFieldLabel.FontColor = [1 1 1];
app.PatientnameEditFieldLabel.Position = [118 243 76 22];
app.PatientnameEditFieldLabel.Text = 'Patient name';

% Create PatientnameEditField
app.PatientnameEditField = uieditfield(app.UIFigure, 'text');
app.PatientnameEditField.ValueChangedFcn = createCallbackFcn(app,
@PatientnameEditFieldValueChanged, true);
app.PatientnameEditField.Position = [205 243 319 22];

% Create IEditFieldLabel
app.IEditFieldLabel = uilabel(app.UIFigure);
app.IEditFieldLabel.FontColor = [1 1 1];
app.IEditFieldLabel.Position = [118 203 74 22];
app.IEditFieldLabel.Text = 'ID';

% Create IEditField
app.IEditField = uieditfield(app.UIFigure, 'text');
app.IEditField.ValueChangedFcn = createCallbackFcn(app, @IEditFieldValueChanged,
true);
app.IEditField.Position = [205 203 319 22];

% Create GenderDropDownLabel
app.GenderDropDownLabel = uilabel(app.UIFigure);
app.GenderDropDownLabel.FontColor = [1 1 1];
app.GenderDropDownLabel.Position = [118 156 74 22];

```

```

app.GenderDropDownLabel.Text = 'Gender';

% Create GenderDropDown
app.GenderDropDown = uidropdown(app.UIFigure);
app.GenderDropDown.Items = {'', 'Male', 'Female', 'Other'};
app.GenderDropDown.ValueChangedFcn = createCallbackFcn(app,
@GenderDropDownValueChanged, true);
app.GenderDropDown.Position = [205 156 320 22];
app.GenderDropDown.Value = '';

% Create DateofbirthEditField_2Label
app.DateofbirthEditField_2Label = uilabel(app.UIFigure);
app.DateofbirthEditField_2Label.FontColor = [1 1 1];
app.DateofbirthEditField_2Label.Position = [118 104 71 22];
app.DateofbirthEditField_2Label.Text = 'Date of birth';

% Create DateofbirthEditField_2
app.DateofbirthEditField_2 = uieditfield(app.UIFigure, 'text');
app.DateofbirthEditField_2.ValueChangedFcn = createCallbackFcn(app,
@DateofbirthEditField_2ValueChanged, true);
app.DateofbirthEditField_2.Position = [205 104 320 22];

% Create DateEditFieldLabel
app.DateEditFieldLabel = uilabel(app.UIFigure);
app.DateEditFieldLabel.FontColor = [1 1 1];
app.DateEditFieldLabel.Position = [118 59 31 22];
app.DateEditFieldLabel.Text = 'Date';

% Create DateEditField
app.DateEditField = uieditfield(app.UIFigure, 'text');
app.DateEditField.ValueChangedFcn = createCallbackFcn(app,
@DateEditFieldValueChanged, true);
app.DateEditField.Position = [205 59 320 22];

% Create NEXTButton
app.NEXTButton = uibutton(app.UIFigure, 'push');
app.NEXTButton.ButtonPushedFcn = createCallbackFcn(app, @NEXTButtonPushed, true);
app.NEXTButton.Position = [272 12 100 23];
app.NEXTButton.Text = 'NEXT';

% Create Image2
app.Image2 = uiimage(app.UIFigure);
app.Image2.Position = [171 272 338 201];

```

```

    app.Image2.ImageSource = fullfile(pathToMLAPP, 'logo.png');

    % Show the figure after all components are created
    app.UIFigure.Visible = 'on';
end
end

% App creation and deletion
methods (Access = public)

    % Construct app
    function app = INTERFAZPRIMARIA

        % Create UIFigure and components
        createComponents(app)

        % Register the app with App Designer
        registerApp(app, app.UIFigure)

        if nargin == 0
            clear app
        end
    end

    % Code that executes before app deletion
    function delete(app)

        % Delete UIFigure when app is deleted
        delete(app.UIFigure)
    end
end
end
end

```

Anexo 5

Código interfaz principal

```
classdef app2 < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public)
        UIFigure          matlab.ui.Figure
        ResetButton       matlab.ui.control.Button
        Desviaciontext2   matlab.ui.control.EditField
        TextArea_7        matlab.ui.control.TextArea
        Desviaciontext    matlab.ui.control.EditField
        TextArea_6        matlab.ui.control.TextArea
        TextArea_4        matlab.ui.control.TextArea
        DateEditField     matlab.ui.control.EditField
        DateEditFieldLabel matlab.ui.control.Label
        DateofbirthEditField_2 matlab.ui.control.EditField
        DateofbirthEditField_2Label matlab.ui.control.Label
        GenderEditField   matlab.ui.control.EditField
        GenderEditFieldLabel matlab.ui.control.Label
        IEditField        matlab.ui.control.EditField
        IEditFieldLabel   matlab.ui.control.Label
        PatientnameEditField matlab.ui.control.EditField
        PatientnameEditFieldLabel matlab.ui.control.Label
        ButtLoading_2     matlab.ui.control.Button
        QuadrantButton    matlab.ui.control.Button
        TextArea_3        matlab.ui.control.TextArea
        TextArea_2        matlab.ui.control.TextArea
        TextArea          matlab.ui.control.TextArea
        SumregionsButton  matlab.ui.control.Button
        ButtLoading       matlab.ui.control.Button
        TabGroup          matlab.ui.container.TabGroup
        PUPILTab          matlab.ui.container.Tab
        PupilradiusLSpinner matlab.ui.control.Spinner
        PupilradiusLSpinnerLabel matlab.ui.control.Label
        PupilradiusRSpinner matlab.ui.control.Spinner
        PupilradiusRSpinnerLabel matlab.ui.control.Label
        BinarizedarearemovesSpinner matlab.ui.control.Spinner
        BinarizedarearemovesSpinnerLabel matlab.ui.control.Label
        DeleteareaSpinner matlab.ui.control.Spinner
        DeleteareaSpinnerLabel matlab.ui.control.Label
    end
end
```



```

BinarizationSpinner      matlab.ui.control.Spinner
BinarizationSpinnerLabel  matlab.ui.control.Label
CORNEAREFLTab           matlab.ui.container.Tab
CentroidCRButton         matlab.ui.control.Button
BinarizedarearemovesSpinner_3  matlab.ui.control.Spinner
BinarizedarearemovesSpinner_3Label  matlab.ui.control.Label
BinarizationSpinner_2    matlab.ui.control.Spinner
BinarizationSpinner_2Label  matlab.ui.control.Label
BinarizedarearemovesSpinner_2  matlab.ui.control.Spinner
BinarizedarearemovesSpinner_2Label  matlab.ui.control.Label
IRISTab                  matlab.ui.container.Tab
TamaoIrisSlider          matlab.ui.control.Slider
TamaoIrisSliderLabel     matlab.ui.control.Label
Label_3                  matlab.ui.control.Label
Cuadrantes2              matlab.ui.control.UIAxes
Paciente                  matlab.ui.control.UIAxes
previw1                  matlab.ui.control.UIAxes
previw2                  matlab.ui.control.UIAxes
previw3                  matlab.ui.control.UIAxes
Img2                     matlab.ui.control.UIAxes
end

```

```

properties (Access = private)
    callingapp % Description
end

```

```

% Callbacks that handle component events
methods (Access = private)

```

```

% Code that executes after component creation
function startupFcn(app, INTERFAZPRIMARIA, Patientname, ID, Gender, Age, Date)
    app.callingapp=INTERFAZPRIMARIA;
    app.PatientnameEditField.Value=Patientname;
    app.IDEditField.Value=ID;
    app.GenderEditField.Value=Gender;
    app.DateofbirthEditField_2.Value=Age;
    app.DateEditField.Value=Date;

```

```

end

% Callback function
function UploadIMGButtonValueChanged(app, event)

end

% Callback function
function ImageClicked(app, event)

end

% Button pushed function: ButtLoading
function ButtLoadingPushed(app, event)
    global a
    global I
    global Fotopaciente
    global REc

    cla(app.Img2,'reset');

    [filename pathname] = uigetfile({'*.JPG'},'File Selector');
    fullpathname = strcat(pathname,filename);

    a = imread(fullpathname);
    imshow(a,'Parent',app.Img2);
    REc=(a);
    I=rgb2gray(a);
    Fotopaciente=(a);
    imshow(Fotopaciente,'Parent',app.Paciente);

end

% Button down function: Img2
function Img2ButtonDown(app, event)

end

```

```

% Callback function
function PatientEditFieldValueChanged(app, event)
    global Nombrepaciente
    PatientEditFieldValueChanged.value=Nombrepaciente;

```

```
end
```

```

% Callback function
function img3ButtonDown(app, event)

```

```
end
```

```

% Value changed function: BinarizationSpinner
function BinarizationSpinnerValueChanged(app, event)
    global bpupila
    global bpupilar
    global I

```

```
    bpupilar = app.BinarizationSpinner.Value;
```

```
    bpupila=im2bw(I,bpupilar);
```

```
    imshow(bpupila,'Parent',app.Img2);
```

```
end
```

```

% Callback function
function BinarizadoSpinner_2ValueChanged(app, event)
    global biris
    global binairis
    global I

```

```
    binairis = app.BinarizadoSpinner_2.Value;
```

```
    biris=im2bw(I,binairis);
```

```
    imshow(biris,'Parent',app.Img2)
```

```
end
```

```

% Value changed function: DeleteareaSpinner

```

```

function DeleteareaSpinnerValueChanged(app, event)
    global bpupila
    global bpupilararea

    bpupilararea= app.DeleteareaSpinner.Value;

    bpupila=bwareaopen(bpupila,bpupilararea);
    imshow(bpupila,'Parent',app.Img2);

end

% Value changed function: BinarizedarearemovesSpinner
function BinarizedarearemovesSpinnerValueChanged(app, event)
    global areabin
    global c
    global bpupila
    global c1

    c=zeros(size(bpupila));
    c(bpupila==0)=1;
    c=logical(c);

    areabin = app.BinarizedarearemovesSpinner.Value;

    c1=bwareaopen(c,areabin);
    imshow(c1,'Parent',app.Img2);

end

% Value changed function: PupilradiusRSpinner
function PupilradiusRSpinnerValueChanged(app, event)
    global s1
    global c1
    global centroidspupila

```

```

global statspupila
global centersPUPILA
global diameters
global radii
global r21pupila
global r22pupila
global REc
global pup
global radiopupiR

s1 = regionprops(c1,'centroid')

centroidsPUPILA = cat(1,s1.Centroid);
statspupila = regionprops('table',c1,'Centroid','MajorAxisLength','MinorAxisLength')

centersPUPILA = statspupila.Centroid;
diameters = mean([statspupila.MajorAxisLength statspupila.MinorAxisLength],2);
radii = diameters/2;
r21pupila=radii(1,1);
r22pupila=radii(2,1);

radiopupiR = app.PupilradiusRSpinner.Value;

pup=(REc);

imshow(pup,'Parent',app.Img2)
hold on
viscircles(app.Img2,centersPUPILA(1,:),radiopupiR,'Color','w','LineWidth',1);
hold off

imshow(pup,'Parent',app.previw2)
hold on
viscircles(app.previw2,centersPUPILA(1,:),radiopupiR,'Color','w','LineWidth',1);
hold off

end

% Callback function
function EliminarareaSpinner_2ValueChanged(app, event)
    global birisarea
    global biris
    global BW1

```

```

birisarea= app.EliminarareaSpinner_2.Value;

BW1=bwareaopen(biris,birisarea);
imshow(BW1,'Parent',app.Img2);

end

% Callback function
function EliminarareainvSpinnerValueChanged(app, event)
    global invarea
    global BW2
    global BW1
    global BWr

    invarea = app.EliminarareainvSpinner.Value;

    BW2=imadjust(double(BW1),[0 1],[1 0]);

    BW2=logical(BW2);

    BWr=bwareaopen(BW2,invarea);
    imshow(BWr,'Parent',app.Img2);

end

% Callback function
function ErosionSpinnerValueChanged(app, event)
    global se
    global disco
    global erosion
    global BWr

    disco = app.ErosionSpinner.Value;

    se = strel("disk",disco)
    erosion = imerode(BWr,se);
    imshow(erosion,'Parent',app.Img2)

```

```
end
```

```
% Callback function
```

```
function ElimnarareaerosionSpinnerValueChanged(app, event)
```

```
    global iris  
    global elimarea  
    global erosion
```

```
    elimarea = app.ElimnarareaerosionSpinner.Value;
```

```
    iris=bwareaopen(erosion,elimarea);  
    imshow(iris,'Parent',app.Img2);
```

```
end
```

```
% Callback function
```

```
function RadioirisSpinnerValueChanged(app, event)
```

```
    global s1  
    global radioiris  
    global centroidsIRIS  
    global statsIRIS  
    global centersIRIS  
    global diameters  
    global radii  
    global r21IRIS  
    global r22IRIS  
    global iri  
    global a  
    global iris
```

```
    radioiris = app.RadioirisSpinner.Value;
```

```
    s1 = regionprops(iris,'centroid')
```

```
    centroidsIRIS = cat(1,s1.Centroid);  
    statsIRIS = regionprops('table',iris,'Centroid','MajorAxisLength','MinorAxisLength')
```

```
    centersIRIS = statsIRIS.Centroid;  
    diameters = mean([statsIRIS.MajorAxisLength statsIRIS.MinorAxisLength],2);  
    radii = diameters/2;  
    r21IRIS=radii(1,1);
```

```

r22IRIS=radii(2,1);

iri=(a);
imshow(iri,'Parent',app.Img2);
hold on
viscircles(app.Img2,centersIRIS(1,:),radioiris,'Color','w','LineWidth',1);
viscircles(app.Img2,centersIRIS(2,:),radioiris,'Color','w','LineWidth',1);
hold off

imshow(iri,'Parent',app.previw1)
hold on
viscircles(app.previw1,centersIRIS(1,:),radioiris,'Color','w','LineWidth',1);
viscircles(app.previw1,centersIRIS(2,:),radioiris,'Color','w','LineWidth',1);
hold off

end

% Callback function
function BinarizadoSpinner_3ValueChanged(app, event)
    global c2
    global c1
    global s2
    global centroidsREFLEJO
    global statsREFLEJO
    global centersREFLEJO
    global diametersR
    global radiiR
    global r21R
    global r22R

    c2=zeros(size(c1));
    c2(c1==0)=1;
    c2=logical(c2);

    s2 = regionprops(c2,'centroid')

    centroidsREFLEJO = cat(1,s2.Centroid);
    statsREFLEJO = regionprops('table',c2,'Centroid','MajorAxisLength','MinorAxisLength')

    centersREFLEJO = statsREFLEJO.Centroid;
    diametersR = mean([statsREFLEJO.MajorAxisLength statsREFLEJO.MinorAxisLength],2);

```



```

radiiR = diametersR/2;
r21R=radii(1,1);
r22R=radii(2,1);

reflejocorneal=(a);
imshow(reflejocorneal,'Parent',app.Img2);
hold on
viscircles(app.Img2,centersREFLEJO(2,:),1,'Color','w','LineWidth',1);
viscircles(app.Img2,centersREFLEJO(3,:),1,'Color','w','LineWidth',1);
hold off

```

end

% Callback function

```
function MostrarRCButtonPushed(app, event)
```

```

global c2
global c1
global s2
global centroidsREFLEJO
global statsREFLEJO
global centersREFLEJO
global diametersR
global radiiRC
global a
global reflejocorneal

```

```

c2=zeros(size(c1));
c2(c1==0)=1;
c2=logical(c2);

```

```
s2 = regionprops(c2,'centroid')
```

```

centroidsREFLEJO = cat(1,s2.Centroid);
statsREFLEJO = regionprops('table',c2,'Centroid','MajorAxisLength','MinorAxisLength')

```

```

centersREFLEJO = statsREFLEJO.Centroid;
diametersR = mean([statsREFLEJO.MajorAxisLength statsREFLEJO.MinorAxisLength],2);
radiiRC = diametersR/2;

```

```

reflejocorneal=(a);
imshow(reflejocorneal,'Parent',app.Img2);

```

```

    hold on
    viscircles(app.Img2,centersREFLEJO(2,:),1,'Color','w','LineWidth',1);
    viscircles(app.Img2,centersREFLEJO(3,:),1,'Color','w','LineWidth',1);
    hold off
end

```

```

% Callback function
function BinarizarButtonPushed(app, event)

```

```

    global c2
    global c

```

```

    c2=zeros(size(c));
    c2(c==0)=1;
    c2=logical(c2);

```

```

    imshow(c2,'Parent',app.Img2)

```

```

end

```

```

% Value changed function: BinarizedarearemovesSpinner_2
function BinarizedarearemovesSpinner_2ValueChanged(app, event)

```

```

    global c2
    global EliminarRC
    global bRc
    global cRC

```

```

    cRC=zeros(size(bRc));
    cRC(bRc==0)=1;
    cRC=logical(cRC);

```

```

    EliminarRC= app.BinarizedarearemovesSpinner_2.Value;
    c2=bwareaopen(cRC,EliminarRC);

```

```

    imshow(c2,'Parent',app.Img2)

```

```

end

```

```

% Value changed function: BinarizationSpinner_2
function BinarizationSpinner_2ValueChanged2(app, event)

```

```

    global I
    global bRc
    global BinarizadoRC

```

```

BinarizadoRC = app.BinarizationSpinner_2.Value;
bRc=im2bw(I,BinarizadoRC);

imshow(bRc,'Parent',app.Img2)

end

% Value changed function: BinarizedarearemovesSpinner_3
function BinarizedarearemovesSpinner_3ValueChanged(app, event)
    global c2
    global invertiRC
    global c2RC

    c2RC=zeros(size(c2));
    c2RC(c2==0)=1;
    c2RC=logical(c2RC);

    invertiRC = app.BinarizedarearemovesSpinner_3.Value;

    c2RC=bwareaopen(c2RC,invertiRC);

    imshow(c2RC,'Parent',app.Img2);
end

% Button pushed function: CentroidCRButton
function CentroidCRButtonPushed(app, event)
    global s2
    global c2RC
    global REc
    global centroidsREFLEJOC
    global statsREFLEJOC
    global radiiR
    global REflejoC
    global centersREFLEJO

    s2 = regionprops(c2RC,'centroid')

```

```

centroidsREFLEJOC = cat(1,s2.Centroid);
statsREFLEJOC = regionprops('table',c2RC,'Centroid','MajorAxisLength','MinorAxisLength')

centersREFLEJO = statsREFLEJOC.Centroid;
diametersR = mean([statsREFLEJOC.MajorAxisLength
statsREFLEJOC.MinorAxisLength],2);
radiiR = diametersR/2;

ReflejoC=(REc);
imshow(ReflejoC,'Parent',app.Img2);
hold on
viscircles(app.Img2,centersREFLEJO(2,:),1,'Color','w','LineWidth',1);
viscircles(app.Img2,centersREFLEJO(3,:),1,'Color','w','LineWidth',1);
hold off

imshow(ReflejoC,'Parent',app.prewiw3)
hold on
viscircles(app.prewiw3,centersREFLEJO(2,:),1,'Color','w','LineWidth',1);
viscircles(app.prewiw3,centersREFLEJO(3,:),1,'Color','w','LineWidth',1);
hold off

end

% Button pushed function: SumregionsButton
function SumregionsButtonPushed(app, event)
    global iri
    global Radiodeiris
    global radiopupiR
    global radiopupiL
    global centersPUPILA
    global centersREFLEJO

    imshow(iri,'Parent',app.Img2);
    hold on
    viscircles(app.Img2,centersPUPILA(1,:),Radiodeiris,'Color','w','LineWidth',1);
    viscircles(app.Img2,centersPUPILA(2,:),Radiodeiris,'Color','w','LineWidth',1);
    viscircles(app.Img2,centersPUPILA(1,:),radiopupiR,'Color','w','LineWidth',1);
    viscircles(app.Img2,centersPUPILA(2,:),radiopupiL,'Color','w','LineWidth',1);
    viscircles(app.Img2,centersREFLEJO(2,:),1,'Color','w','LineWidth',1);
    viscircles(app.Img2,centersREFLEJO(3,:),1,'Color','w','LineWidth',1);
    hold off

```

end

% Button pushed function: QuadrantButton
function QuadrantButtonPushed(app, event)

global sumatoria
global radiopupiR
global radiopupiL
global centersPUPILA
global centersREFLEJO
global ER
global EL
global ELXY
global ERXY
global REc
global dere
global izqu
global aMatrix
global bMatrix
global Valormax
global nuevo
global Valorcuadrante
global nuemeronuevo

sumatoria=(REc);

imshow(sumatoria,'Parent',app.Img2);

hold on

viscircles(app.Img2,centersPUPILA(1,:),radiopupiR,'Color','w','LineWidth',1);

viscircles(app.Img2,centersPUPILA(2,:),radiopupiL,'Color','w','LineWidth',1);

viscircles(app.Img2,centersREFLEJO(2,:),1,'Color','w','LineWidth',1);

viscircles(app.Img2,centersREFLEJO(3,:),1,'Color','w','LineWidth',1);

drawline(app.Img2,'Position',[centersPUPILA(1,1)

(centersPUPILA(1,2)-50);centersPUPILA(1,1) (centersPUPILA(1,2)+50)],'Color','r','LineWidth',1);

```
drawline(app.Img2,'Position',[centersPUPILA(1,1)-50
(centersPUPILA(1,2));(centersPUPILA(1,1)+50) centersPUPILA(1,2)],'Color','r','LineWidth',1);
```

```
drawline(app.Img2,'Position',[centersPUPILA(2,1)
(centersPUPILA(2,2)-50);centersPUPILA(2,1) (centersPUPILA(2,2)+50)],'Color','r','LineWidth',1);
drawline(app.Img2,'Position',[(centersPUPILA(2,1)-50)
(centersPUPILA(2,2));(centersPUPILA(2,1)+50) centersPUPILA(2,2)],'Color','r','LineWidth',1);
```

```
ER = drawpoint(app.Img2,'Position',[centersREFLEJO(2,1) centersREFLEJO(2,2)]);
EL = drawpoint(app.Img2,'Position',[centersREFLEJO(3,1) centersREFLEJO(3,2)]);
```

```
centersPUPILA(1,:);
centersREFLEJO(2,:);
ERXY = centersREFLEJO(2,:)-centersPUPILA(1,:);
ERXY = ERXY.*[1,-1]
```

```
centersPUPILA(2,:);
centersREFLEJO(3,:);
ELXY = centersREFLEJO(3,:)-centersPUPILA(2,:);
ELXY = ELXY.*[1,-1]
```

```
ER.Label = num2str(ERXY)
EL.Label = num2str(ELXY)
```

```
hold off
```

```
dere = ERXY;
izqu = ELXY;
```

```
Valorcuadrante=max(ERXY,ELXY);
```

```
Valormax=max(radiopupiR,radiopupiL);
```

```
nuemeronuevo=Valorcuadrante+70;
```

```

if Valormax==radiopupiR

    aMatrix = [-nuemeronuevo nuemeronuevo];
    bMatrix = aMatrix-aMatrix;

    % centrar 0,0 en el ojo bueno

    plot(app.Cuadrantes2,aMatrix,bMatrix,bMatrix,aMatrix);
    hold(app.Cuadrantes2,"on")
    plot(app.Cuadrantes2,izqu(1),izqu(2),'K*', 'color','b');
    plot(app.Cuadrantes2,0,0,'K*', 'color','r');
    legend(app.Cuadrantes2,["Left" "Right"])

    if izqu(1)<1 && izqu(1)>-1
        app.Desviaciontext.Value='Orthotropy';
    end

    if izqu(1)<-2
        app.Desviaciontext.Value='Left eye presents Exotropia';
    end

    if izqu(1)>2
        app.Desviaciontext.Value='Left eye presents Esotropia';
    end

    if izqu(2)<3 && dere(1)<-1
        app.Desviaciontext2.Value='Left eye presents Hypertropia';
    end

    if izqu(2)>3 && dere(1)<-1
        app.Desviaciontext2.Value='Left eye presents hypotropia';
    end

else

    aMatrix = [-nuemeronuevo nuemeronuevo];
    bMatrix = aMatrix-aMatrix;

```

```

% centrar 0,0 en el ojo bueno

plot(app.Cuadrantes2,aMatrix,bMatrix,bMatrix,aMatrix);
hold(app.Cuadrantes2,"on")
plot(app.Cuadrantes2,dere(1),dere(2),'K*', 'color','r');
plot(app.Cuadrantes2,0,0,'K*', 'color','b');
legend(app.Cuadrantes2,["Left" "Right"])

if dere(1)<1 && dere(1)>=-1
    app.Desviaciontext.Value='Orthotropy';
end

if dere(1)>2
    app.Desviaciontext.Value='Right eye presents Exotropia';
end

if dere(1)<-2
    app.Desviaciontext.Value='Right eye presents Esotropia';

end

if dere(2)<3
    app.Desviaciontext2.Value='Right eye presents Hypertropia';

end

if dere(2)>3 && dere(1)<1
    app.Desviaciontext2.Value='Right eye presents hypotropia';

end

end

end

% Value changed function: TamaoIrisSlider
function TamaoIrisSliderValueChanged(app, event)

```



```

global Radiodeiris
global iriz
global REc
global centersPUPILA

Radiodeiris = app.TamaoIrisSlider.Value;
iriz=(REc);

imshow(iriz,'Parent',app.Img2)
hold on
viscircles(app.Img2,centersPUPILA(1,:),Radiodeiris,'Color','w','LineWidth',1);
viscircles(app.Img2,centersPUPILA(2,:),Radiodeiris,'Color','w','LineWidth',1);
hold off

imshow(iriz,'Parent',app.previw1)
hold on
viscircles(app.previw1,centersPUPILA(1,:),Radiodeiris,'Color','w','LineWidth',1);
viscircles(app.previw1,centersPUPILA(2,:),Radiodeiris,'Color','w','LineWidth',1);
hold off
end

% Button pushed function: ButtLoading_2
function ButtLoading_2ButtonPushed(app, event)
    global Escalagrisrecorte
    global a
    global Erorecorte
    global Recorteback
    global Binarirecorte
    global Filtrorecorte
    global Centroiderecorte
    global size1
    global record1
    global record2
    global I
    global centroidsrecord
    global REc

    Escalagrisrecorte=rgb2gray(a);

    Erorecorte = strel('disk',10);

```

```

Recorteback = imopen(Escalagrisrecorte,Erorecorte);

Binarirecorte = imbinarize(Recorteback);

Filtrorecorte = bwareafilt(Binarirecorte,1);

Centroiderecorte = regionprops(Filtrorecorte,'centroid');

centroidsrecord = cat(1,Centroiderecorte.Centroid);

size1 = size(a);
record1 = size1(1)/4;
record2 = size1(2)/4;

REc = imcrop(a,[(centroidsrecord(1)-record1) (centroidsrecord(2)-record2) (record1*2)
(record2)]);

imshow(REc,'Parent',app.Img2);

I=rgb2gray(REc);

end

% Callback function
function REButtonPushed(app, event)
    global sumatoria
    global Recorteojoderecho
    global centersPUPILA

    Recorteojoderecho = imcrop(sumatoria,[(centersPUPILA(1,1)-100) (centersPUPILA(1,2)-50)
(centersPUPILA(1,1)+10) (centersPUPILA(1,2)+30)]);
    imshow(Recorteojoderecho,'Parent',app.ojoizquierdo)
end

% Callback function
function ojoderechoButtonDown(app, event)

end

```

```

% Callback function
function ojoizquierdoButtonDown(app, event)

end

% Callback function
function LEButtonPushed(app, event)
    global sumatoria
    global Recortejoizquierdo
    global centersPUPILA

    Recortejoizquierdo = imcrop(sumatoria,[(centersPUPILA(2,1)-100)
(centersPUPILA(2,2)-50) (centersPUPILA(2,1)+10) (centersPUPILA(2,2)+30)]);
    imshow(Recortejoizquierdo,'Parent',app.Img2)

end

% Callback function
function PatientnameEditFieldValueChanged(app, event)

end

% Value changed function: PatientnameEditField
function PatientnameEditFieldValueChanged2(app, event)
    value = app.PatientnameEditField.Value;

end

% Value changed function: IDEditField
function IDEditFieldValueChanged(app, event)
    value = app.IDEditField.Value;

end

% Value changed function: GenderEditField
function GenderEditFieldValueChanged(app, event)
    value = app.GenderEditField.Value;

end

```

```

% Value changed function: DateofbirthEditField_2
function DateofbirthEditField_2ValueChanged(app, event)
    value = app.DateofbirthEditField_2.Value;

end

% Value changed function: DateEditField
function DateEditFieldValueChanged(app, event)
    value = app.DateEditField.Value;

end

% Value changing function: PatientnameEditField
function PatientnameEditFieldValueChanging(app, event)

end

% Callback function
function SaveimagButtonPushed(app, event)
    global iri
    imsave(iri,'C:\Users\USUARIO\Desktop\Proyecto de grado\imagen
guardad\Imageninterfaz.jpg')
end

% Value changed function: Desviaciontext
function DesviaciontextValueChanged(app, event)
%     global tipodesvi

    value = app.Desviaciontext.Value;

end

% Button down function: Cuadrantes2
function Cuadrantes2ButtonDown(app, event)

end

% Value changed function: PupilradiusLSpinner
function PupilradiusLSpinnerValueChanged(app, event)
    global s1
    global c1
    global centroidspupila

```

```
global statspupila
global centersPUPILA
global diameters
global radii
global r21pupila
global r22pupila
global REc
global pup
global radiopupiL
global radiopupiR
```

```
s1 = regionprops(c1,'centroid')
```

```
centroidspupila = cat(1,s1.Centroid);
statspupila = regionprops('table',c1,'Centroid','MajorAxisLength','MinorAxisLength')
```

```
centersPUPILA = statspupila.Centroid;
diameters = mean([statspupila.MajorAxisLength statspupila.MinorAxisLength],2);
radii = diameters/2;
r21pupila=radii(1,1);
r22pupila=radii(2,1);
```

```
radiopupiL = app.PupilradiusLSpinner.Value;
```

```
pup=(REc);
```

```
imshow(pup,'Parent',app.Img2)
hold on
viscircles(app.Img2,centersPUPILA(1,:),radiopupiR,'Color','w','LineWidth',1);
viscircles(app.Img2,centersPUPILA(2,:),radiopupiL,'Color','w','LineWidth',1);
hold off
```

```
imshow(pup,'Parent',app.previw2)
hold on
viscircles(app.previw2,centersPUPILA(1,:),radiopupiR,'Color','w','LineWidth',1);
viscircles(app.previw2,centersPUPILA(2,:),radiopupiL,'Color','w','LineWidth',1);
hold off
```

```
end
```

```

% Button pushed function: ResetButton
function ResetButtonPushed(app, event)
    delete(app2)
    INTERFAZPRIMARIA
end
end
end

```

```

% Component initialization
methods (Access = private)

```

```

% Create UIFigure and components
function createComponents(app)

```

```

% Create UIFigure and hide until all components are created
app.UIFigure = uifigure('Visible', 'off');
app.UIFigure.AutoResizeChildren = 'off';
app.UIFigure.Color = [0.2431 0.4235 0.549];
app.UIFigure.Position = [100 100 1244 712];
app.UIFigure.Name = 'MATLAB App';
app.UIFigure.Resize = 'off';

```

```

% Create Img2
app.Img2 = uiaxes(app.UIFigure);
app.Img2.XTick = [];
app.Img2.YTick = [];
app.Img2.YTickLabel = "";
app.Img2.ButtonDownFcn = createCallbackFcn(app, @Img2ButtonDown, true);
app.Img2.Position = [283 328 678 349];

```

```

% Create prewiw3
app.prewiw3 = uiaxes(app.UIFigure);
app.prewiw3.XTick = [];
app.prewiw3.YTick = [];
app.prewiw3.YTickLabel = "";
app.prewiw3.Position = [15 1 276 109];

```

```

% Create prewiw2
app.prewiw2 = uiaxes(app.UIFigure);
app.prewiw2.XTick = [];
app.prewiw2.YTick = [];
app.prewiw2.YTickLabel = "";
app.prewiw2.Position = [11 136 280 107];

```

```

% Create previ1
app.previ1 = uiaxes(app.UIFigure);
app.previ1.FontWeight = 'bold';
app.previ1.XTick = [];
app.previ1.YTick = [];
app.previ1.YTickLabel = "";
app.previ1.Position = [11 261 280 110];

% Create Paciente
app.Paciente = uiaxes(app.UIFigure);
app.Paciente.XTick = [];
app.Paciente.YTick = [];
app.Paciente.YTickLabel = "";
app.Paciente.Position = [15 389 255 141];

% Create Cuadrantes2
app.Cuadrantes2 = uiaxes(app.UIFigure);
xlabel(app.Cuadrantes2, 'X')
ylabel(app.Cuadrantes2, 'Y')
zlabel(app.Cuadrantes2, 'Z')
app.Cuadrantes2.ButtonDownFcn = createCallbackFcn(app, @Cuadrantes2ButtonDown,
true);
app.Cuadrantes2.Position = [428 13 388 238];

% Create Label_3
app.Label_3 = uilabel(app.UIFigure);
app.Label_3.BackgroundColor = [0.102 0.2392 0.349];
app.Label_3.Position = [1 538 283 173];
app.Label_3.Text = "";

% Create TabGroup
app.TabGroup = uitabgroup(app.UIFigure);
app.TabGroup.AutoSizeChildren = 'off';
app.TabGroup.Position = [973 283 264 332];

% Create PUPILTab
app.PUPILTab = uitab(app.TabGroup);
app.PUPILTab.AutoSizeChildren = 'off';
app.PUPILTab.Title = 'PUPIL';
app.PUPILTab.ForegroundColor = [0.6392 0.2588 0];

% Create BinarizationSpinnerLabel
app.BinarizationSpinnerLabel = uilabel(app.PUPILTab);

```

```

app.BinarizationSpinnerLabel.HorizontalAlignment = 'right';
app.BinarizationSpinnerLabel.FontWeight = 'bold';
app.BinarizationSpinnerLabel.Position = [72 255 74 22];
app.BinarizationSpinnerLabel.Text = 'Binarization';

% Create BinarizationSpinner
app.BinarizationSpinner = uispinner(app.PUPILTab);
app.BinarizationSpinner.Step = 0.01;
app.BinarizationSpinner.Limits = [0.1 0.5];
app.BinarizationSpinner.ValueChangedFcn = createCallbackFcn(app,
@BinarizationSpinnerValueChanged, true);
app.BinarizationSpinner.FontWeight = 'bold';
app.BinarizationSpinner.Position = [161 255 100 22];
app.BinarizationSpinner.Value = 0.1;

% Create DeleteareaSpinnerLabel
app.DeleteareaSpinnerLabel = uilabel(app.PUPILTab);
app.DeleteareaSpinnerLabel.HorizontalAlignment = 'right';
app.DeleteareaSpinnerLabel.FontWeight = 'bold';
app.DeleteareaSpinnerLabel.Position = [76 209 70 22];
app.DeleteareaSpinnerLabel.Text = 'Delete area';

% Create DeleteareaSpinner
app.DeleteareaSpinner = uispinner(app.PUPILTab);
app.DeleteareaSpinner.Step = 100;
app.DeleteareaSpinner.Limits = [0 3500];
app.DeleteareaSpinner.ValueChangedFcn = createCallbackFcn(app,
@DeleteareaSpinnerValueChanged, true);
app.DeleteareaSpinner.FontWeight = 'bold';
app.DeleteareaSpinner.Position = [161 209 100 22];
app.DeleteareaSpinner.Value = 1;

% Create BinarizedarearemovesSpinnerLabel
app.BinarizedarearemovesSpinnerLabel = uilabel(app.PUPILTab);
app.BinarizedarearemovesSpinnerLabel.HorizontalAlignment = 'right';
app.BinarizedarearemovesSpinnerLabel.FontWeight = 'bold';
app.BinarizedarearemovesSpinnerLabel.Position = [6 159 140 22];
app.BinarizedarearemovesSpinnerLabel.Text = 'Binarized area removes';

% Create BinarizedarearemovesSpinner
app.BinarizedarearemovesSpinner = uispinner(app.PUPILTab);
app.BinarizedarearemovesSpinner.Step = 50;
app.BinarizedarearemovesSpinner.Limits = [1 3500];

```



```

    app.BinarizedarearemovesSpinner.ValueChangedFcn = createCallbackFcn(app,
    @BinarizedarearemovesSpinnerValueChanged, true);
    app.BinarizedarearemovesSpinner.FontWeight = 'bold';
    app.BinarizedarearemovesSpinner.Position = [161 159 100 22];
    app.BinarizedarearemovesSpinner.Value = 1;

    % Create PupilradiusRSpinnerLabel
    app.PupilradiusRSpinnerLabel = uilabel(app.PUPILTab);
    app.PupilradiusRSpinnerLabel.FontWeight = 'bold';
    app.PupilradiusRSpinnerLabel.Position = [73 106 86 22];
    app.PupilradiusRSpinnerLabel.Text = 'Pupil radius R';

    % Create PupilradiusRSpinner
    app.PupilradiusRSpinner = uispinner(app.PUPILTab);
    app.PupilradiusRSpinner.Limits = [1 60];
    app.PupilradiusRSpinner.ValueChangedFcn = createCallbackFcn(app,
    @PupilradiusRSpinnerValueChanged, true);
    app.PupilradiusRSpinner.FontWeight = 'bold';
    app.PupilradiusRSpinner.Position = [160 106 100 22];
    app.PupilradiusRSpinner.Value = 1;

    % Create PupilradiusLSpinnerLabel
    app.PupilradiusLSpinnerLabel = uilabel(app.PUPILTab);
    app.PupilradiusLSpinnerLabel.FontWeight = 'bold';
    app.PupilradiusLSpinnerLabel.Position = [74 66 85 22];
    app.PupilradiusLSpinnerLabel.Text = 'Pupil radius L';

    % Create PupilradiusLSpinner
    app.PupilradiusLSpinner = uispinner(app.PUPILTab);
    app.PupilradiusLSpinner.Limits = [1 60];
    app.PupilradiusLSpinner.ValueChangedFcn = createCallbackFcn(app,
    @PupilradiusLSpinnerValueChanged, true);
    app.PupilradiusLSpinner.FontWeight = 'bold';
    app.PupilradiusLSpinner.Position = [161 66 100 22];
    app.PupilradiusLSpinner.Value = 1;

    % Create CORNEAREFLTab
    app.CORNEAREFLTab = uitab(app.TabGroup);
    app.CORNEAREFLTab.AutoResizeChildren = 'off';
    app.CORNEAREFLTab.Title = 'CORNEA REFL. ';
    app.CORNEAREFLTab.ForegroundColor = [0.6392 0.2588 0];

    % Create BinarizedarearemovesSpinner_2Label

```

```

app.BinarizedarearemovesSpinner_2Label = uilabel(app.CORNEAREFLTab);
app.BinarizedarearemovesSpinner_2Label.HorizontalAlignment = 'right';
app.BinarizedarearemovesSpinner_2Label.FontWeight = 'bold';
app.BinarizedarearemovesSpinner_2Label.Position = [4 218 140 22];
app.BinarizedarearemovesSpinner_2Label.Text = 'Binarized area removes';

% Create BinarizedarearemovesSpinner_2
app.BinarizedarearemovesSpinner_2 = uispinner(app.CORNEAREFLTab);
app.BinarizedarearemovesSpinner_2.Step = 50;
app.BinarizedarearemovesSpinner_2.Limits = [0 2000];
app.BinarizedarearemovesSpinner_2.ValueChangedFcn = createCallbackFcn(app,
@BinarizedarearemovesSpinner_2ValueChanged, true);
app.BinarizedarearemovesSpinner_2.FontWeight = 'bold';
app.BinarizedarearemovesSpinner_2.Position = [159 218 100 22];

% Create BinarizationSpinner_2Label
app.BinarizationSpinner_2Label = uilabel(app.CORNEAREFLTab);
app.BinarizationSpinner_2Label.HorizontalAlignment = 'right';
app.BinarizationSpinner_2Label.FontWeight = 'bold';
app.BinarizationSpinner_2Label.Position = [70 255 74 22];
app.BinarizationSpinner_2Label.Text = 'Binarization';

% Create BinarizationSpinner_2
app.BinarizationSpinner_2 = uispinner(app.CORNEAREFLTab);
app.BinarizationSpinner_2.Step = 0.01;
app.BinarizationSpinner_2.Limits = [0.1 0.5];
app.BinarizationSpinner_2.ValueChangedFcn = createCallbackFcn(app,
@BinarizationSpinner_2ValueChanged2, true);
app.BinarizationSpinner_2.FontWeight = 'bold';
app.BinarizationSpinner_2.Position = [159 255 100 22];
app.BinarizationSpinner_2.Value = 0.1;

% Create BinarizedarearemovesSpinner_3Label
app.BinarizedarearemovesSpinner_3Label = uilabel(app.CORNEAREFLTab);
app.BinarizedarearemovesSpinner_3Label.HorizontalAlignment = 'right';
app.BinarizedarearemovesSpinner_3Label.FontWeight = 'bold';
app.BinarizedarearemovesSpinner_3Label.Position = [4 180 140 22];
app.BinarizedarearemovesSpinner_3Label.Text = 'Binarized area removes';

% Create BinarizedarearemovesSpinner_3
app.BinarizedarearemovesSpinner_3 = uispinner(app.CORNEAREFLTab);
app.BinarizedarearemovesSpinner_3.Step = 5;
app.BinarizedarearemovesSpinner_3.Limits = [0 2000];

```

```

    app.BinarizedarearemovesSpinner_3.ValueChangedFcn = createCallbackFcn(app,
    @BinarizedarearemovesSpinner_3ValueChanged, true);
    app.BinarizedarearemovesSpinner_3.FontWeight = 'bold';
    app.BinarizedarearemovesSpinner_3.Position = [159 180 100 22];

    % Create CentroidCRButton
    app.CentroidCRButton = uibutton(app.CORNEAREFLTab, 'push');
    app.CentroidCRButton.ButtonPushedFcn = createCallbackFcn(app,
    @CentroidCRButtonPushed, true);
    app.CentroidCRButton.FontWeight = 'bold';
    app.CentroidCRButton.Position = [70 138 100 23];
    app.CentroidCRButton.Text = 'Centroid CR';

    % Create IRISTab
    app.IRISTab = uitab(app.TabGroup);
    app.IRISTab.AutoResizeChildren = 'off';
    app.IRISTab.Title = 'IRIS';
    app.IRISTab.BackgroundColor = [0.9412 0.9412 0.9412];
    app.IRISTab.ForegroundColor = [0.6392 0.2588 0];

    % Create TamaoIrisSliderLabel
    app.TamaoIrisSliderLabel = uilabel(app.IRISTab);
    app.TamaoIrisSliderLabel.HorizontalAlignment = 'right';
    app.TamaoIrisSliderLabel.FontWeight = 'bold';
    app.TamaoIrisSliderLabel.Position = [3 180 72 22];
    app.TamaoIrisSliderLabel.Text = 'Size Iris';

    % Create TamaoIrisSlider
    app.TamaoIrisSlider = uislider(app.IRISTab);
    app.TamaoIrisSlider.ValueChangedFcn = createCallbackFcn(app,
    @TamaoIrisSliderValueChanged, true);
    app.TamaoIrisSlider.FontWeight = 'bold';
    app.TamaoIrisSlider.Position = [96 189 150 3];

    % Create ButtLoading
    app.ButtLoading = uibutton(app.UIFigure, 'push');
    app.ButtLoading.ButtonPushedFcn = createCallbackFcn(app, @ButtLoadingPushed, true);
    app.ButtLoading.Position = [983 645 95 46];
    app.ButtLoading.Text = 'Load IMG';

    % Create SumregionsButton
    app.SumregionsButton = uibutton(app.UIFigure, 'push');

```

```

    app.SumregionsButton.ButtonPushedFcn = createCallbackFcn(app,
@SumregionsButtonPushed, true);
    app.SumregionsButton.FontWeight = 'bold';
    app.SumregionsButton.Position = [1057 250 100 23];
    app.SumregionsButton.Text = 'Sum regions';

% Create TextArea
app.TextArea = uitextarea(app.UIFigure);
app.TextArea.HorizontalAlignment = 'center';
app.TextArea.FontWeight = 'bold';
app.TextArea.Position = [82 370 122 20];
app.TextArea.Value = {'IRIS'};

% Create TextArea_2
app.TextArea_2 = uitextarea(app.UIFigure);
app.TextArea_2.HorizontalAlignment = 'center';
app.TextArea_2.FontWeight = 'bold';
app.TextArea_2.Position = [82 242 122 20];
app.TextArea_2.Value = {'PUPIL'};

% Create TextArea_3
app.TextArea_3 = uitextarea(app.UIFigure);
app.TextArea_3.HorizontalAlignment = 'center';
app.TextArea_3.FontWeight = 'bold';
app.TextArea_3.Position = [69 113 151 20];
app.TextArea_3.Value = {'CONEAL REFLECTION'};

% Create QuadrantButton
app.QuadrantButton = uibutton(app.UIFigure, 'push');
app.QuadrantButton.ButtonPushedFcn = createCallbackFcn(app, @QuadrantButtonPushed,
true);
app.QuadrantButton.FontWeight = 'bold';
app.QuadrantButton.Position = [1057 211 100 23];
app.QuadrantButton.Text = 'Quadrant';

% Create ButtLoading_2
app.ButtLoading_2 = uibutton(app.UIFigure, 'push');
app.ButtLoading_2.ButtonPushedFcn = createCallbackFcn(app,
@ButtLoading_2ButtonPushed, true);
app.ButtLoading_2.Position = [1124 645 95 46];
app.ButtLoading_2.Text = 'Imag Data set';

% Create PatientnameEditFieldLabel

```

```

app.PatientnameEditFieldLabel = uilabel(app.UIFigure);
app.PatientnameEditFieldLabel.HorizontalAlignment = 'right';
app.PatientnameEditFieldLabel.FontColor = [1 1 1];
app.PatientnameEditFieldLabel.Position = [11 687 76 22];
app.PatientnameEditFieldLabel.Text = 'Patient name';

% Create PatientnameEditField
app.PatientnameEditField = uieditfield(app.UIFigure, 'text');
app.PatientnameEditField.ValueChangedFcn = createCallbackFcn(app,
@PatientnameEditFieldValueChanged2, true);
app.PatientnameEditField.ValueChangingFcn = createCallbackFcn(app,
@PatientnameEditFieldValueChanging, true);
app.PatientnameEditField.Position = [15 666 255 22];

% Create IEditFieldLabel
app.IEditFieldLabel = uilabel(app.UIFigure);
app.IEditFieldLabel.FontColor = [1 1 1];
app.IEditFieldLabel.Position = [15 642 25 22];
app.IEditFieldLabel.Text = 'ID';

% Create IEditField
app.IEditField = uieditfield(app.UIFigure, 'text');
app.IEditField.ValueChangedFcn = createCallbackFcn(app, @IEditFieldValueChanged,
true);
app.IEditField.Position = [15 621 255 22];

% Create GenderEditFieldLabel
app.GenderEditFieldLabel = uilabel(app.UIFigure);
app.GenderEditFieldLabel.HorizontalAlignment = 'right';
app.GenderEditFieldLabel.FontColor = [1 1 1];
app.GenderEditFieldLabel.Position = [24 569 46 22];
app.GenderEditFieldLabel.Text = 'Gender';

% Create GenderEditField
app.GenderEditField = uieditfield(app.UIFigure, 'text');
app.GenderEditField.ValueChangedFcn = createCallbackFcn(app,
@GenderEditFieldValueChanged, true);
app.GenderEditField.Position = [11 549 72 22];

% Create DateofbirthEditField_2Label
app.DateofbirthEditField_2Label = uilabel(app.UIFigure);
app.DateofbirthEditField_2Label.HorizontalAlignment = 'right';
app.DateofbirthEditField_2Label.FontColor = [1 1 1];

```

```

app.DateofbirthEditField_2Label.Position = [96 570 71 22];
app.DateofbirthEditField_2Label.Text = 'Date of birth';

% Create DateofbirthEditField_2
app.DateofbirthEditField_2 = uieditfield(app.UIFigure, 'text');
app.DateofbirthEditField_2.ValueChangedFcn = createCallbackFcn(app,
@DateofbirthEditField_2ValueChanged, true);
app.DateofbirthEditField_2.Position = [96 549 77 22];

% Create DateEditFieldLabel
app.DateEditFieldLabel = uilabel(app.UIFigure);
app.DateEditFieldLabel.HorizontalAlignment = 'right';
app.DateEditFieldLabel.FontColor = [1 1 1];
app.DateEditFieldLabel.Position = [209 569 34 22];
app.DateEditFieldLabel.Text = 'Date ';

% Create DateEditField
app.DateEditField = uieditfield(app.UIFigure, 'text');
app.DateEditField.ValueChangedFcn = createCallbackFcn(app,
@DateEditFieldValueChanged, true);
app.DateEditField.Position = [182 548 88 22];

% Create TextArea_4
app.TextArea_4 = uitextarea(app.UIFigure);
app.TextArea_4.HorizontalAlignment = 'center';
app.TextArea_4.Position = [526 306 191 20];
app.TextArea_4.Value = {'Corneal reflex quadrants'};

% Create TextArea_6
app.TextArea_6 = uitextarea(app.UIFigure);
app.TextArea_6.HorizontalAlignment = 'center';
app.TextArea_6.Position = [527 260 192 25];
app.TextArea_6.Value = {'Reference to the centered eye'};

% Create Desviaciontext
app.Desviaciontext = uieditfield(app.UIFigure, 'text');
app.Desviaciontext.ValueChangedFcn = createCallbackFcn(app,
@DesviaciontextValueChanged, true);
app.Desviaciontext.Position = [983 132 236 33];

% Create TextArea_7
app.TextArea_7 = uitextarea(app.UIFigure);
app.TextArea_7.HorizontalAlignment = 'center';

```

```

app.TextArea_7.Position = [1043 173 111 22];
app.TextArea_7.Value = {'Type of deviation'};

% Create Desviaciontext2
app.Desviaciontext2 = uieditfield(app.UIFigure, 'text');
app.Desviaciontext2.Position = [983 81 236 33];

% Create ResetButton
app.ResetButton = uibutton(app.UIFigure, 'push');
app.ResetButton.ButtonPushedFcn = createCallbackFcn(app, @ResetButtonPushed, true);
app.ResetButton.BackgroundColor = [0.6 0.2118 0.2824];
app.ResetButton.FontColor = [1 1 1];
app.ResetButton.Position = [1059 13 84 54];
app.ResetButton.Text = 'Reset';

% Show the figure after all components are created
app.UIFigure.Visible = 'on';
end
end

% App creation and deletion
methods (Access = public)

% Construct app
function app = app2(varargin)

    runningApp = getRunningApp(app);

% Check for running singleton app
if isempty(runningApp)

    % Create UIFigure and components
    createComponents(app)

    % Register the app with App Designer
    registerApp(app, app.UIFigure)

    % Execute the startup function
    runStartupFcn(app, @(app)startupFcn(app, varargin{:}))
else

    % Focus the running singleton app
    figure(runningApp.UIFigure)

```

```
        app = runningApp;
    end

    if nargin == 0
        clear app
    end
end

% Code that executes before app deletion
function delete(app)

    % Delete UIFigure when app is deleted
    delete(app.UIFigure)
end
end
end
```