# PORTADA

Tabla de contenido

[PORTADA 1](#_Toc180675494)

[DEDICATORIA 2](#_Toc180675495)

[AGRADECIMIENTOS 3](#_Toc180675496)

[RESUMEN 4](#_Toc180675497)

[INDICE DE FIGURAS 6](#_Toc180675498)

[1. Introducción 7](#_Toc180675499)

[1.1. Objetivos 8](#_Toc180675500)

[1.2. Hipótesis 8](#_Toc180675501)

[2. Revisión de la literatura 9](#_Toc180675502)

[2.1. Marco teórico 9](#_Toc180675503)

[2.2. Antecedentes 11](#_Toc180675504)

[2.3. Estado del arte 11](#_Toc180675505)

[3. Materiales y métodos 13](#_Toc180675506)

[3.1. Lista de materiales 13](#_Toc180675507)

[3.2. Descripción de los materiales 14](#_Toc180675508)

[3.3. Diseño Metodológico 16](#_Toc180675509)

[Calibración 18](#_Toc180675510)

[Prueba para la detección del disco óptico 21](#_Toc180675511)

[4. Resultados 25](#_Toc180675512)

[4.1. Presentación de Datos 25](#_Toc180675513)

[4.2. Análisis e interpretación 25](#_Toc180675514)

[5. Discusión 25](#_Toc180675515)

[6. Conclusiones 25](#_Toc180675516)

[7. Bibliografía 25](#_Toc180675517)

[8. Referencias y Trabajos citados 26](#_Toc180675518)

[9. Anexos 26](#_Toc180675519)

# INDICE DE FIGURAS

[Ilustración 1. Campimetría. 7](#_Toc180675520)

[Ilustración 2. Perimetría 7](#_Toc180675521)

[Ilustración 3. Trazado campimétrico que representa el campo visual del ojo izquierdo. El círculo rojo muestra el punto ciego 9](#_Toc180675522)

[Ilustración 4. Representación de visión normal y con escotoma. 10](#_Toc180675523)

[Ilustración 5. Captura de pantalla del código Eyetracker.py, junto con la imagen y la detección de la pupila. 14](#_Toc180675524)

[Ilustración 6. Imagen ROI binarizada. 14](#_Toc180675525)

[Ilustración 7. Gráfico de coordenadas x e y obtenidas mediante la función de eyetracking. 15](#_Toc180675526)

[Ilustración 8. Imagen de la montura de la cámara infrarroja utilizada en el sistema de eyetracking. 15](#_Toc180675527)

[Ilustración 9. Representación del sistema de ejes (X,Y) en pantalla. 16](#_Toc180675528)

[Ilustración 10. Relación píxel- milímetro en ejes X (naranja) e Y (azul). 16](#_Toc180675529)

[Ilustración 11. Representación gráfica para obtener los grados de rotación de la mirada. 16](#_Toc180675530)

[Ilustración 12. Trazado campimétrico para ojo derecho en monitor de prueba. 17](#_Toc180675531)

[Ilustración 13. Ventana de Operación. 18](#_Toc180675532)

[Ilustración 14. Datos obtenidos durante la calibración. 18](#_Toc180675533)

[Ilustración 15. Datos obtenidos de la validación. 19](#_Toc180675534)

[Ilustración 16. Errores y desviaciones obtenidos de la validación. 19](#_Toc180675535)

[Ilustración 17. Diagrama de flujo del código de calibración. 20](#_Toc180675536)

[Ilustración 18. 21](#_Toc180675537)

[Ilustración 19. 22](#_Toc180675538)

[Ilustración 20. 23](#_Toc180675539)

INDICE DE TABLAS

# Introducción

En el ámbito de la oftalmología, uno de los estudios más importantes para evaluar la salud visual de los pacientes es la campimetría y la perimetría. Estos procedimientos tienen como objetivo principal detectar la ceguera en diferentes zonas de la retina. Aunque ambos estudios comparten el mismo propósito, difieren en su metodología y en el alcance del área del campo visual que evalúan.

La campimetría mide la sensibilidad del campo visual central, trabajando dentro de los 30 grados centrales alrededor del punto de fijación. Esta prueba se realiza generalmente utilizando una pantalla plana (campímetro) donde se proyectan estímulos luminosos en diferentes puntos del campo visual central. El paciente debe responder cuándo ve estos estímulos.

Por otro lado, la perimetría es una prueba más completa y detallada que evalúa el campo visual total, tanto central como periférico, pudiendo evaluar hasta 90 grados o más en cada dirección desde el punto de fijación. Esta prueba es esencial para detectar y monitorear enfermedades como el glaucoma, donde los defectos pueden aparecer en el campo visual periférico. Para realizar esta evaluación, se utiliza un dispositivo llamado perímetro, que puede ser manual (como el perímetro de Goldmann) o automatizado (como el perímetro Humphrey). El paciente fija su mirada en un punto central y debe indicar cuándo percibe luces en varias ubicaciones del campo visual.



Ilustración . Campimetría.



Ilustración . Perimetría

A pesar de las diferencias en complejidad y alcance de detección entre ambos métodos, tanto la campimetría como la perimetría requieren de la participación verbal del paciente. La consistencia y reproducibilidad de estos estudios dependen de la atención del sujeto a explorar y de la habilidad del explorador. Estos métodos no son prácticos en niños pequeños o personas de avanzada edad y resultan poco fiables en adolescentes. Diversas circunstancias, como la duración de la prueba y el estado de ánimo del paciente, pueden modificar los resultados.

## Objetivos

Objetivo General

El presente trabajo plantea el desarrollo de una metodología que elimine la necesidad de la participación verbal del paciente. Se propone la implementación de un dispositivo basado en un eye-tracker de laboratorio para registrar los movimientos oculares e inferir la respuesta del sujeto (si detectó o no el estímulo) a partir de la reacción del ojo al estímulo.

Objetivos Particulares

1. Obtener el patrón de movimientos oculares más informativo sobre la detección del estímulo presentado.
2. Generar un algoritmo que permita visualizar estímulos aleatorios y las coordenadas del ojo durante la prueba.
3. Elaborar e implementar una interfaz para conectar el dispositivo eye-tracker y la computadora para la obtención de datos.

## Hipótesis

*Los sujetos que pueden percibir un estímulo visual podrán fijar su vista en él o en la posición aproximada donde se presentó el estímulo. Por el contrario, los sujetos con escotomas no podrán fijar su vista en la posición del estímulo debido a su incapacidad para percibirlo.*

# Revisión de la literatura

## Marco teórico

Campo visual y disco óptico

El campo visual es la zona de visión observada por un ojo en un instante dado. La región percibida por el lado nasal se llama campo visual nasal y la que llega al lado lateral campo visual temporal. (Hall, 2016)

Las dimensiones monoculares del campo visual de una persona normal se extienden hasta los 60 grados a nivel superior y hasta los 75 en nivel inferior. En sentido horizontal se extiende nasalmente hasta los 60 grados y en sentido temporal hasta los 100º. Los huesos maxilar, frontal, nasal y cigomático limitan el campo visual, por tanto, los valores descritos anteriormente son aproximados.

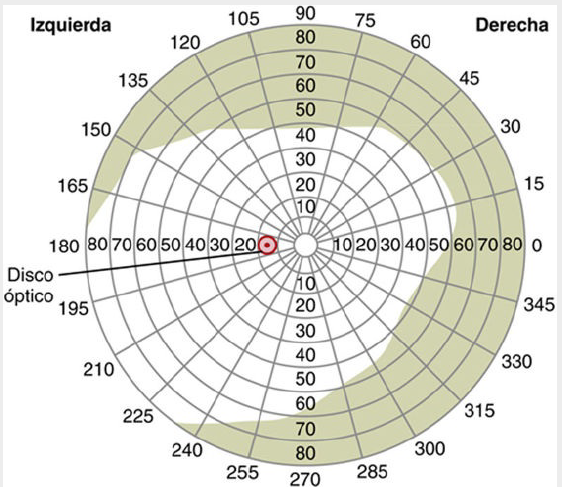


Ilustración . Trazado campimétrico que representa el campo visual del ojo izquierdo. El círculo rojo muestra el punto ciego

El disco óptico es la zona de la retina de donde surge el nervio óptico. Esta zona del polo posterior del ojo carece de células sensibles a la luz, tanto de conos como de bastones, perdiendo así toda la sensibilidad óptica. Normalmente no se percibe su existencia debido a que el punto ciego de un ojo es suplido por la información visual que proporciona el movimiento ocular y la visión bifocal. También es difícil percibirlo con un solo ojo, ya que, ante la falta de información visual en la zona del punto ciego, el cerebro recrea virtualmente y rellena esa pequeña área en relación con el entorno visual que la rodea. Esta mancha ciega se encuentra a 15 grados laterales desde el punto de fijación central.



Ilustración . Representación de visión normal y con escotoma.

Patologías y condiciones que afectan el campo visual[[1]](https://medlineplus.gov/)

El campo visual puede verse afectado por diversas patologías y condiciones, que pueden casuar pérdida parcial o completa de la visión en ciertas áreas. Algunas de las patologías más comunes son:

* **Glaucoma**. Es una neuropatía óptica progresiva que daña el nervio óptico, generalmente debido a un aumento de la presión intraocular. Con el tiempo, el glaucoma puede provocar una pérdida de visión periférica que avanza hacia el centro, lo que puede resultar en una pérdida de visión en túnel.
* **Degeneración macular relacionada con la edad (DMAE).** Es una enfermedad degenerativa que afecta la mácula, la parte central de la retina responsable de la visión central detallada. Es más común en personas mayores de 50 años.
* **Retinopatía diabética**. Es una complicación de la diabetes que daña los vasos sanguíneos de la retina. Puede causar áreas de visión borrosa o ceguera en partes del campo visual, debido a la acumulación de líquidos, exudados, hemorragias, o el crecimiento anormal de vasos sanguíneos.
* **Desprendimiento de retina**. Ocurre cuando la retina se separa de la capa subyacente de tejido de soporte, lo que impide que funcione correctamente. Sin tratamiento rápido, puede llevar a una pérdida completa de visión en el área afectada.
* **Neuropatía óptica isquémica anterior (NOIA)**. Es una afección que ocurre cuando el flujo sanguíneo al nervio óptico se interrumpe, causando daño al nervio y pérdida de visión. Suele afectar a personas mayores y se asocia con factores de riesgo como hipertensión, diabetes y arteriosclerosis.
* **Retinosis pigmentaria**. Es un grupo de enfermedades hereditarias que causar la degeneración progresiva de la retina. Comienza con la pérdida de la visión nocturna y avanza a la pérdida de la visión periférica, lo que puede llevar a una visión en túnel.
* **Cataratas.** Opacificación del cristalino del ojo, que puede dispersar la luz que entra en el ojo y causar visión borrosa, deslumbramiento y una pérdida de la sensibilidad al contraste. Aunque afecta principalmente la visión central, en etapas avanzadas puede afectar el campo visual.

## Antecedentes

La evaluación del campo visual es crucial para el diagnóstico y seguimiento de diversas enfermedades oculares. La campimetría y la perimetría son dos de las técnicas más utilizadas para este propósito. La campimetría, desarrollada a principios del siglo XX, se centra en la medición de la sensibilidad del campo visual central, mientras que la perimetría, que surgió más tarde, ofrece una evaluación más completa del campo visual periférico.

Estudios previos sobre la campimetría y la perimetría

La campimetría ha sido utilizada históricamente para detectar defectos en el campo visual central, particularmente en condiciones como la degeneración macular. Investigaciones tempranas, como las realizadas por Goldmann y sus contemporáneos, establecieron las bases para la campimetría moderna. Por otro lado, la perimetría ha sido fundamental en el diagnóstico de glaucoma y otras enfermedades que afectan la visión periférica. Estudios recientes han demostrado que la perimetría automatizada, como la que utiliza el perímetro Humphrey, proporciona una mayor precisión y reproducibilidad en la detección de defectos del campo visual.

## Estado del arte

Métodos tradicionales y sus limitaciones

A pesar de sus ventajas, los métodos tradicionales de campimetría y perimetría presentan limitaciones significativas. La necesidad de participación verbal por parte del paciente introduce variabilidad en los resultados, especialmente en niños pequeños, personas mayores y adolescentes. Además, la duración de las pruebas y el estado emocional del paciente pueden afectar la precisión de los resultados.

Para mitigar estos inconvenientes, se han implementado algunas estrategias en los métodos tradicionales, como los "falsos positivos":

***"La confiabilidad del paciente es una consideración inicial importante al interpretar los campos visuales. Esto puede evaluarse mediante las tasas de respuestas falsamente positivas y falsas negativas; la evaluación de la confiabilidad por parte del operador; la fluctuación de mediciones de umbral repetidas; el número de pérdidas de fijación; y el número de presentaciones de estímulos necesarias para completar un examen del campo visual. El operador debe explicar adecuadamente el procedimiento de prueba y alentar la cooperación y el estado de alerta del paciente durante el examen si se quieren obtener resultados confiables."*** (Joseph Caprioli, 1991)[***[2]***](#_Referencias_y_Trabajos)

***"Una evaluación adicional de la confiabilidad del paciente se obtiene cuando el perímetro produce periódicamente solo el sonido asociado con el estímulo luminoso sin presentar realmente la luz (llamado falso positivo si el paciente responde) y presentando periódicamente un estímulo en un área que el paciente puede ver (llamado falso negativo si el paciente no responde)"*** (Roy W. Beck, 1985)[***[3]***](#_Referencias_y_Trabajos)

Dado el contexto histórico y los avances recientes en tecnología de seguimiento ocular, se justifica la necesidad de desarrollar una metodología que elimine la dependencia de la participación verbal del paciente. Este proyecto propone la implementación de un dispositivo basado en un eye-tracker para registrar los movimientos oculares e inferir la respuesta del sujeto a partir de la reacción del ojo al estímulo.

Alternativas actuales: Specvis y otros softwares

Actualmente, existen softwares capaces de ofrecer al sujeto la facilidad de realizar pruebas de perimetría a través de sus computadoras personales, reduciendo así el costo de la intervención temprana y permitiendo realizar la prueba prácticamente en cualquier lugar. Algunos de estos softwares son:

* **NovaVision Inc.**[[4]](#_Referencias_y_Trabajos): Ofrece una funcionalidad que permite a los usuarios realizar una prueba de detección muy básica en línea. El paciente se fija en un punto central y utiliza el teclado para responder a los puntos blancos que se muestran estáticamente (solo una vez) en ubicaciones predefinidas sobre un fondo negro.
* **Ceeable Visual Field Analyzer (CVFA)**[[5]](#_Referencias_y_Trabajos): Proporcionada por Ceeable Inc., esta es una prueba de campo visual que puede detectar, clasificar y controlar las enfermedades oculares degenerativas utilizando solo una tableta. Esta solución no es gratuita y requiere una conexión a Internet.
* **Specvis**: Una aplicación gratuita y de código abierto desarrollada por nuestro grupo para el examen del campo visual escrita en el lenguaje de programación Java, que puede ejecutarse en cualquier computadora personal:
  + ***"Al diseñar Specvis, nuestro objetivo era cumplir con tres requisitos principales para nuestro software con respecto a llenar el vacío entre las pruebas de confrontación y perimetría. En primer lugar, queríamos ofrecer un programa gratuito y de código abierto. En segundo lugar, el software debía haber podido examinar todo el campo visual del paciente, su sensibilidad a la luz de diferentes longitudes de onda y luminancia, y debía haber sido preciso y fiable para el diagnóstico de alteraciones del campo visual. Por último, el software debe haber sido lo suficientemente fácil de usar para permitir que cualquiera lo utilice sin una formación intensiva."*** (Dzwiniel P, 2017)[***[6]***](#_Referencias_y_Trabajos)

# Materiales y métodos

## Lista de materiales

A continuación, se presenta una lista detallada de los materiales y equipos utilizados en la prueba:

1. Computadora con dos monitores.
   1. Descripción: Un monitor se utiliza para leer los datos y ejecutar el programa. El segundo monitor será utilizado para la presentación de estímulos visuales.
2. Un conjunto de cámara de alta velocidad que incorpora:
   1. Un espejo infrarrojo
   2. Una cámara infrarroja con iris automático
   3. Fuentes de iluminación infrarrojas
3. Fuente de Alimentación
   1. Descripción: Fuente de alimentación para iluminación infrarroja
4. Software Utilizado
   1. Código para eyetracking: Programa de demostración visualizar la cámara, la detección de la pupila y las coordenadas de esta en tiempo real.
   2. Código para calibración.
   3. Código personalizado: Script para la prueba y análisis de los movimientos oculares. El código completo estará disponible en una carpeta comprimida entregada junto con este informe.
5. Otros Materiales
   1. Cables de conexión.
   2. Soporte para el eye-tracker y el monitor.

## Descripción de los materiales

Eyetracking con OpenCV

Utilizando la librería de OpenCV en Python, se desarrolló un código capaz de detectar la pupila y determinar su posición en tiempo real para las pruebas (Ilustración 5)

Interfaz de usuario gráfica

Descripción generada automáticamente

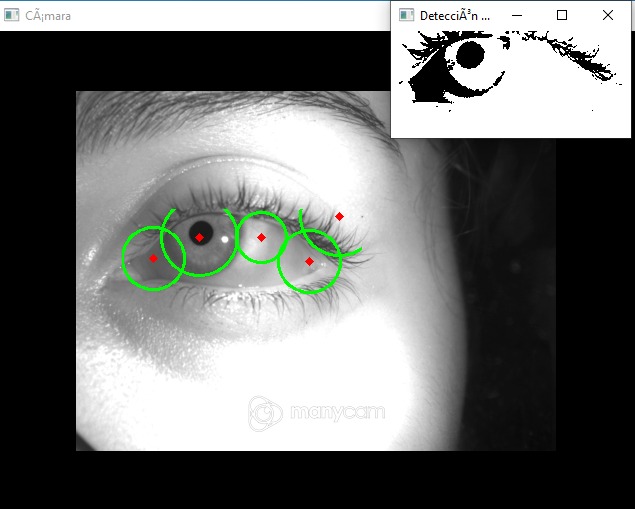
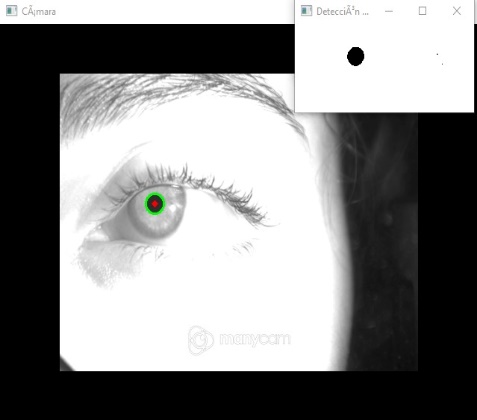
Ilustración 5. Captura de pantalla del código Eyetracker.py, junto con la imagen y la detección de la pupila.

*OpenCV* es una biblioteca que facilita el desarrollo de aplicaciones de visión por computadora. Se utiliza en tareas como la detección y reconocimiento de objetos, análisis de imágenes, procesamiento de videos, entre otros. Es eficiente y optimizada para el procesamiento en tiempo real.

Mediante una cámara infrarroja conectada a la computadora, se empleó *OpenCV* para el reconocimiento de la pupila utilizando la función ‘*Hough Circles*’, que utiliza la **Transformada de Hough** para detectar **círculos** en una imagen. Esta técnica matemática se emplea para identificar formas geométricas, como líneas, círculos o elipses, incluso cuando están parcialmente ocultas o distorsionadas.

Para mejorar la precisión de la detección de la pupila, el código permite definir una Región de Interés (ROI) en la imagen binarizada, lo que ayuda a limitar el área de búsqueda de la pupila (Ilustración 6) y optimiza el uso de *HoughCircles*. Además, se utilizó la biblioteca *‘matplotlib’* para graficar en tiempo real las coordenadas de la pupila en términos de píxeles (Ilustración 7).

Ilustración . Imagen ROI binarizada.



Gráfico, Gráfico de líneas

Descripción generada automáticamente

Ilustración 7. Gráfico de coordenadas x e y obtenidas mediante la función de eyetracking.

Montura de la cámara infrarroja

La cámara infrarroja, utilizada para capturar la imagen del ojo y realizar la detección de la pupila, está montada en un soporte específico que asegura su posición y estabilidad durante las pruebas. La correcta alineación de la cámara es fundamental para el rendimiento del sistema de eyetracking, ya que garantiza una captura precisa de las imágenes en tiempo real (Ilustración 8).

Imagen que contiene cuarto, escritorio, computadora, tabla

Descripción generada automáticamente

Ilustración 8. Imagen de la montura de la cámara infrarroja utilizada en el sistema de eyetracking.

Funcionamiento  
El seguimiento ocular funciona rastreando la posición y los movimientos de los ojos de manera no intrusiva. Una fuente de luz infrarroja o cercana al infrarrojo, invisible para el ojo humano, ilumina la pupila. Una cámara infrarroja captura la imagen, y *OpenCV* utiliza la función *Hough Circles* para detectar y delimitar el centro de la pupila.

## Diseño Metodológico

La presentación de estímulos se llevará a cabo con un monitor de resolución 1366x768 pixeles. La ubicación de los estímulos se medirá en milímetros de la pantalla. Por lo tanto, se tienen en cuenta las relaciones píxel-milímetro para cada eje, considerando el centro de la pantalla como el origen de coordenadas.

Calendario

Descripción generada automáticamente con confianza media

Ilustración 9. Representación del sistema de ejes (X,Y) en pantalla.

Ilustración . Relación píxel- milímetro en ejes X (naranja) e Y (azul).

|  |  |
| --- | --- |
| Transformaciones en eje X | |
| Píxel | Valor en mm |
| 0 | -200 |
| 1360 | 200 |
|  |  |
| Transformaciones en eje Y | |
| Píxel | Valor en mm |
| 0 | 110 |
| 768 | -110 |

Teniendo en cuenta el trazado campimétrico de la [Ilustración](#_Campo_visual_y) 3, los resultados esperados durante la prueba se deben expresar en grados de rotación del ojo/pupila. Para comprender la relación entre los milímetros en pantalla y los grados de desplazamiento del ojo, se tiene en cuenta la siguiente ilustración:

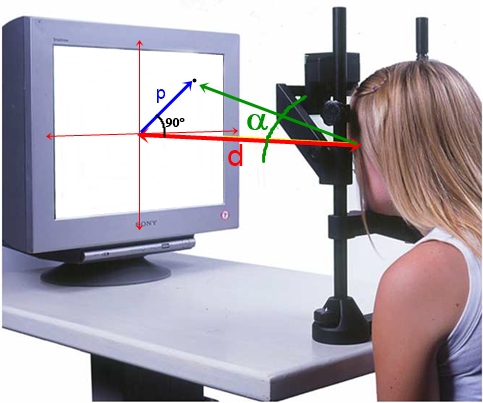


Ilustración 11. Representación gráfica para obtener los grados de rotación de la mirada.

Para un determinado punto en la pantalla, el desplazamiento de la mirada en grados se puede medir utilizando como datos la distancia entre el ojo del sujeto y la pantalla (d) y el vector desplazamiento del punto (p):

Durante las pruebas se buscará detectar el disco óptico, el cual se encuentra aproximadamente a 15° laterales desde el punto de fijación central. Por lo tanto, tomando una distancia de 300 mm, se evaluará el campo central en 50 grados latitudinales y 30 grados longitudinales.

Diagrama

Descripción generada automáticamente

Ilustración 12. Trazado campimétrico para ojo derecho en monitor de prueba.

### Calibración

La primera etapa del procedimiento metodológico consiste en obtener las rectas de calibración para los ejes X e Y, con el fin de transformar los valores obtenidos del eyetracker (en pixeles) a la unidad de medición en pantalla (milímetros). La relación se describe mediante la ecuación:

Donde **m** es la pendiente de la recta y **b** la ordenada al origen.

Para obtener estas rectas de calibración, el código ‘Calibración.py’ tiene como objetivo mostrar 16 puntos en pantalla (grilla de 4x4 puntos). A partir de los datos obtenidos, el algoritmo de regresión lineal de Python ([*‘LinearRegression’ [7]*](https://scikit-learn.org/dev/modules/generated/sklearn.linear_model.LinearRegression.html)) se utiliza para ajustar la recta que mejor representa la relación entre los valores en píxeles y milímetros, minimizando la diferencia entre las predicciones y los valores reales.

El proceso de calibración incluye una interfaz en pantalla que permite al operario iniciar la calibración y guardar los datos generados durante el proceso.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Ilustración 13. Ventana de Operación.

En la siguiente ilustración se puede observar la terminal de Python durante el proceso de calibración. Durante la ejecución de la función, se muestran las coordenadas **x** e **y** tanto de los puntos presentados en pantalla como de los datos obtenidos por el eyetracker.

Patrón de fondo

Descripción generada automáticamente

Ilustración .

Texto

Descripción generada automáticamente

Ilustración 15. Datos obtenidos durante la calibración.

Validación

Una vez finalizada la calibración, se ejecuta una función de validación utilizando el mismo código. Esta validación consiste en presentar nuevos puntos en la pantalla para evaluar el error o la distancia entre los valores reales (en pantalla) y las lecturas obtenidas por el eyetracker.

Durante el proceso de validación, se imprimen en la terminal las coordenadas **x** e **y** de cada punto mostrado en pantalla, junto con las correspondientes lecturas del eyetracker. Al finalizar, se calcula y devuelve el error promedio y la desviación estándar de cada eje.

En la siguiente figura se muestra la terminal de Python durante el proceso de validación.

Texto

Descripción generada automáticamente

Ilustración 16. Datos obtenidos de la validación.

Texto

Descripción generada automáticamente

Ilustración 17. Errores y desviaciones obtenidos de la validación.

Diagrama de flujo

En la Ilustración 16 se presenta el diagrama de flujo del proceso de calibración. El proceso comienza con la ejecución simultánea de dos funciones: la presentación de puntos en pantalla y la obtención de los datos de la posición de la pupila para cada intervalo de tiempo y cada punto mostrado.

Una vez completadas ambas funciones, se calculan las rectas de calibración, y a continuación se inicia el proceso de validación. Este proceso funciona de manera similar a la calibración, con la excepción de que los datos obtenidos por el eyetracker se transforman utilizando las rectas calculadas previamente.

Al finalizar, el programa devuelve el error promedio y la desviación estándar para cada eje. Dependiendo del criterio del operario, se pueden guardar las rectas de calibración o repetir el procedimiento.

Diagrama

Descripción generada automáticamente

Ilustración 18. Diagrama de flujo del código de calibración.

### Prueba para la detección del disco óptico

El objetivo de este trabajo es desarrollar una metodología de detección de escotomas que elimine la necesidad de participación verbal o motora del paciente, evitando así las respuestas subjetivas a los estímulos luminosos. Para ello, se diseñaron dos pruebas que mapean el campo visual en un rango de 50° laterales y 30° longitudinales, con el fin de detectar el disco óptico, ubicado a 15° laterales del punto de fijación central (ver [Ilustración 3](#_Marco_teórico)).

En la primera prueba, los resultados dependen de la respuesta subjetiva del paciente, quien presionará un botón por cada estímulo percibido. En la segunda prueba, en cambio, los resultados dependen de los movimientos de la pupila durante la presentación de los estímulos, lo que constituye una respuesta objetiva. Para esta segunda prueba, se utiliza un dispositivo basado en un eyetracker que registra los movimientos oculares e infiere la respuesta del paciente (si detectó o no el estímulo) a partir de la reacción del ojo ante el mismo.

Ambas pruebas siguen el mismo funcionamiento. Se comienza con una ronda de 46 estímulos luminosos presentados de manera aleatoria en la pantalla. Dependiendo de la respuesta del sujeto, se crea un nuevo patrón basado en las respuestas de 'no percepción' de la primera ronda. Al finalizar cada prueba genera un mapa de calor que muestra los estímulos no detectados y su frecuencia de fallo, además de un archivo CSV con el siguiente formato de tabla:

|  |  |  |
| --- | --- | --- |
| coordenada x [mm] | coordenada y [mm] | frecuencia de fallo |
| X1 | Y1 | 0 |
| X2 | Y2 | 1 |
| X3 | Y3 | 2 |
| … | … | … |

El patrón de estímulos se obtiene de la función ‘*patron\_vectores’*, ubicada en el archivo Funciones.py (Ilustración 17).

Diagrama

Descripción generada automáticamente

Ilustración . Patrón de estímulos

El archivo Prueba.py genera una interfase donde el operario puede elegir entre realizar la prueba con Eyetracker o con teclado y guardar los resultados de la prueba elegida.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Ilustración . Interfase en ventana de Operario

Presentación Aleatoria de Estímulos

Los 46 estímulos están definidos a partir de una cruz de fijación en el origen de coordenadas. Dado que los puntos se presentan de forma aleatoria, es necesario cambiar constantemente las coordenadas de la cruz de fijación.

El proceso de presentación de estímulos sigue el diagrama de flujo que se muestra en la Ilustración 20. El procedimiento comienza con la aparición de una cruz de fijación en el origen de coordenadas, que se muestra en la pantalla durante 1 segundo. A continuación, se presenta un punto luminoso en una posición aleatoria (x₁, y₁) durante 200 milisegundos.

Luego, la pantalla se queda completamente negra durante 800 milisegundos. En este intervalo, si el sujeto ha percibido el estímulo, su mirada estará dirigida hacia la posición (x₁, y₁). En caso contrario, estará mirando en otra parte de la pantalla.

Durante este periodo, el sujeto debe presionar un botón (en la primera prueba), o simplemente dirigir la mirada hacia la posición (x₁, y₁), mientras el eyetracker registra los movimientos oculares en la segunda prueba.

Después de este tiempo, se presenta una nueva cruz de fijación en la posición (x₁, y₁), que ahora actúa como el nuevo origen de coordenadas. El próximo estímulo aleatorio se mostrará a una distancia (x₂, y₂) desde esta nueva posición.

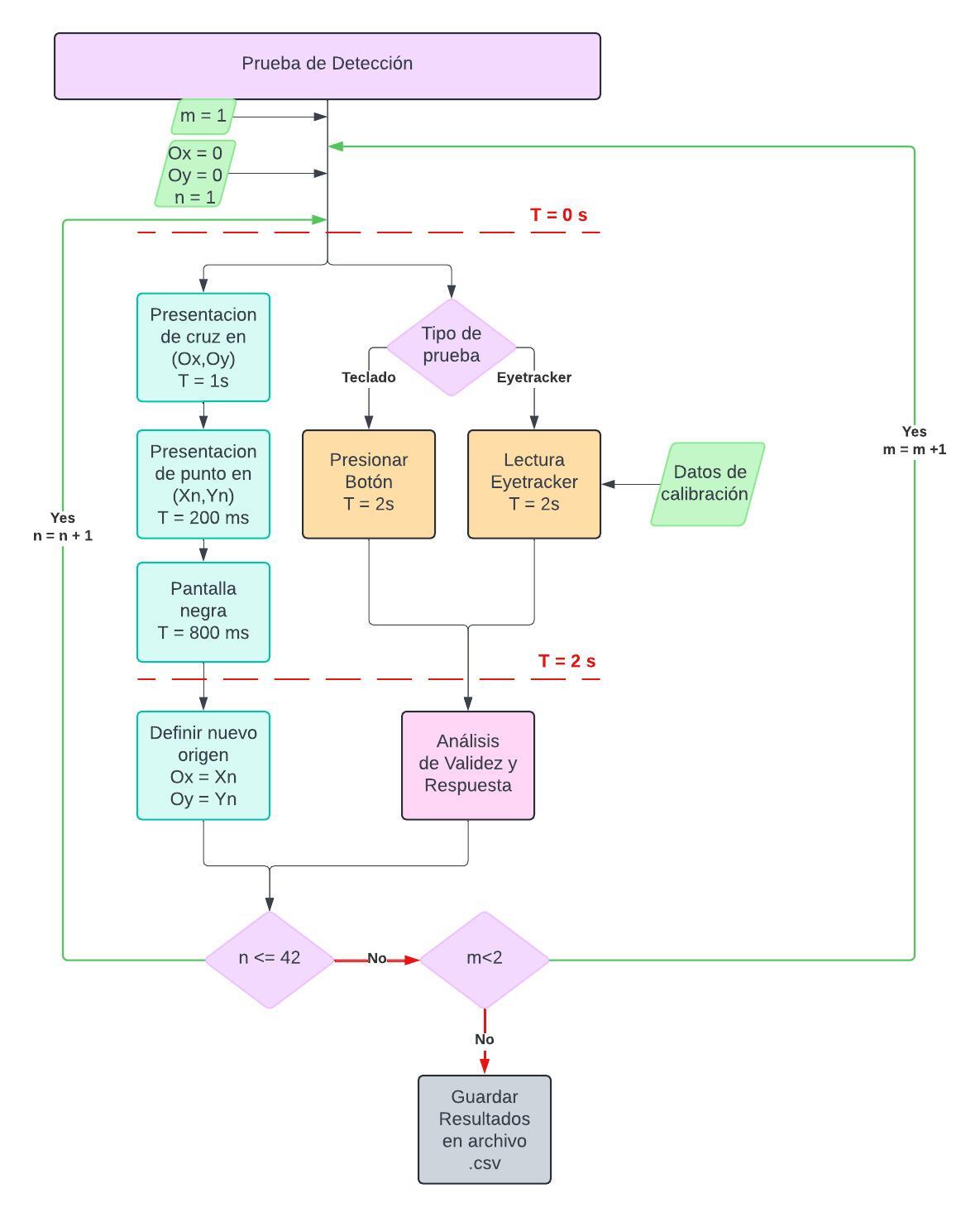


Ilustración . Diagrama de flujo de la prueba de detección

Interfaz de Operario y Sujeto

La ejecución del código Pruebas.py utiliza ambas pantallas de la computadora. En la pantalla del sujeto, se abre una ventana con fondo blanco donde se muestran los puntos y cruces necesarios para la prueba.

En la pantalla del operario, se despliega una interfaz que permite seleccionar entre realizar la prueba con el Eyetracker o con el teclado, y guardar los resultados de la prueba elegida (Ilustración 19). Si se elige trabajar con el Eyetracker, el programa habilita la cámara y muestra la imagen binarizada del ROI con la detección de la pupila (Ilustración 6), además de una ventana blanca donde se presentan las cruces y puntos, junto con un punto rojo que indica la posición de la mirada del sujeto durante la prueba (Ilustración 21).

De este modo, el operario puede optar por visualizar tanto la ventana de prueba como la posición de la mirada del sujeto, o simplemente observar los resultados impresos en la terminal para cada estímulo presentado (Ilustración 23).

Gráfico

Descripción generada automáticamenteGráfico, Gráfico de cajas y bigotes

Descripción generada automáticamente

Ilustración . Pantalla Operario.

Ilustración . Pantalla Sujeto

Texto

Descripción generada automáticamente

Ilustración . Terminal de Python en prueba de detección.

Análisis de Respuesta con Eyetracker

La prueba de detección de estímulos utilizando el Eyetracker presenta, a diferencia de la prueba con la implementación del botón, dos etapas de análisis para determinar la respuesta a la percepción de cada estímulo.

El objetivo de esta prueba es evaluar si los estímulos presentados a una distancia específica de una cruz de fijación fueron percibidos o no por el sujeto. Para lograrlo, es crucial verificar que el sujeto comience la prueba dirigiendo su mirada hacia la cruz de fijación. Si no se cumpliera esta condición inicial, el sujeto podría estar mirando cualquier parte de la pantalla al inicio, lo que alteraría la distancia entre su mirada y el estímulo posteriormente presentado, afectando así los resultados de la prueba.

Considerando esta condición inicial, los datos registrados por el Eyetracker también se emplean para validar la prueba. Si el código determina, mediante la lectura de los datos del Eyetracker, que el sujeto ha posicionado correctamente su mirada en la cruz de fijación al inicio, la prueba se considera válida. En caso contrario, la prueba se clasifica como "No Válida" y no se evalúa la detección del estímulo.

Si la prueba es considerada válida, el código analiza si el sujeto percibió el estímulo basado en la posición de su mirada una vez que el estímulo ha sido presentado, comparándola con la posición exacta del estímulo y tomando en cuenta un rango de error previamente establecido.

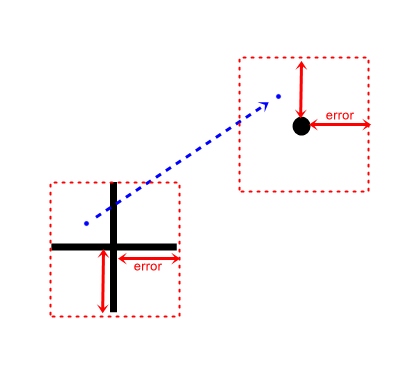


Ilustración .

Determinación de errores

Cuando se presenta un estímulo en pantalla, el ojo se encuentra en constante movimiento alrededor de la zona donde se presentó el estímulo. Puesto que durante la prueba se presentan puntos y cruces, es necesario determinar un área para cada caso en el que se considere que el ojo del sujeto se encuentra cerca de la posición de estos estímulos (para así determinar la validez de la prueba). Para ello, se realizaron 20 pruebas con un patrón de puntos y cruces presentados en el campo central (entre los 2° y 12°) para determinar, por medio de cálculos estadísticos y gráficas, la posición del ojo en ambos casos (cuando mira fijamente a la cruz y cuando mira fijamente la zona donde se presentó el punto) y determinar de esta manera el rango de error aceptable en futuras mediciones.

Diagrama

Descripción generada automáticamente

# Resultados

## 4.1. Presentación de Datos

## 4.2. Análisis e interpretación

# Discusión

# Conclusiones

# Bibliografía

Dzwiniel P, G. M.-G. (2017). Specvis: Free and open-source software for visual field examination. *Plos One*.

Hall, G. y. (2016). *Tratado de fisiología Médica.* Barcelona, España: Elsevier.

Joseph Caprioli, M. (1991). Automated Perimetry in Glaucoma. *American Journal of Ophthalmology*, 236.

Roy W. Beck, ,. T. (1985). A Clinical Comparison of Visual Field Testing With a New Automated Perimeter, the Humphrey Field Analyzer, and the Goldmann Perimeter. *Ophtalmology*, 78.

# Referencias y Trabajos citados

* 1. <https://medlineplus.gov/>
  2. "Automated Perimetry in Glaucoma," Joseph Caprioli, M.D. - *American Journal of Ophthalmology*.
  3. "A Clinical Comparison of Visual Field Testing With a New Automated Perimeter, the Humphrey Field Analyzer, and the Goldmann Perimeter," Roy W. Beck, M.D., Terry J. Bergstrom, M.D., Paul R. Lichter, M.D. - *Departments of Ophthalmology, Neurology, and Neurosurgery, and the Kellogg Eye Center, University of Michigan School of Medicine, Ann Arbo*
  4. <http://www.novavision.com/>
  5. <http://www.ceeable.com/>
  6. Dzwiniel P, Gola M, WoÂjcik-Gryciuk A, Waleszczyk WJ (2017) Specvis: Free and open-source software for visual field examination. PLoS ONE 12(10): e0186224. <https://doi.org/10.1371/journal.pone.0186224>
  7. <https://scikit-learn.org/dev/modules/generated/sklearn.linear_model.LinearRegression.html>

# Anexos