



Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE GRADO

Determinación de la altura del Sol mediante técnicas de visión artificial

Grado en Ingeniería Mecánica

ALUMNO: David Muñoz Yébenes

DIRECTORES: José González Coma

Francisco Troncoso Pastoriza

CURSO ACADÉMICO: 2023-2024

Universida_{de}Vigo



Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE GRADO

Determinación de la altura del Sol mediante técnicas de visión artificial

Grado en Ingeniería Mecánica
Intensificación en Tecnología Naval
Cuerpo General

UniversidadeVigo

RESUMEN

Este Trabajo de Fin de Grado se centra en la aplicación de técnicas de visión artificial para calcular con precisión la altura del Sol en el cielo. Desde la invención del astrolabio en la antigua Grecia alrededor del siglo II a.C, la determinación de la altura de los astros ha sido utilizada en diversos ámbitos, desde la astronomía y la navegación hasta la meteorología o energía solar.

La metodología empleada implica el uso de algoritmos de procesamiento de imágenes para analizar fotografías del cielo y determinar la altura angular del Sol. Se exploran diferentes técnicas para la determinación de la altura del Sol, como la calibración de la cámara o el escalado de elementos. Además, se estudia la precisión con la que el sistema de visión artificial detecta el Sol y el horizonte, lo que ofrece ventajas significativas sobre los métodos tradicionales, ya que permite mediciones automáticas y continuas sin depender de instrumentos costosos.

Los resultados obtenidos demuestran la viabilidad y eficacia de las técnicas de visión artificial en la determinación de la altura solar, abriendo nuevas posibilidades para investigaciones futuras y aplicaciones prácticas en campos como la navegación, la agricultura, la gestión de energía o la meteorología.

PALABRAS CLAVE

Sextante, Visión artificial, Algoritmos de procesamiento, Calibración de lente, Altura del Sol

AGRADECIMIENTOS

A mi familia y, más si cabe, a mi hermano.

A mis profesores y oficiales durante mi periodo de formación y en particular a mis tutores D. José González Coma y D. Francisco Troncoso Pastoriza.

Y, por último, a mi promoción. Honor y Gloria a la 424-154.

CONTENIDO

Contenido	1
Índice de Figuras	3
Índice de Tablas.....	4
1 Introducción y objetivos	5
1.1 Contextualización y motivación.....	5
1.2 Objetivos	6
1.3 Estudios relacionados.....	6
1.3.1 Sistema automático de geolocalización basado en los astros: módulo de adquisición de datos.....	6
1.3.2 Sistema de posicionamiento solar para buques	6
1.4 Organización de la memoria	7
2 Estado del arte	8
2.1 Introducción a la visión artificial	8
2.1.1 Evolución de la visión artificial	8
2.1.2 Procesos de la visión artificial	9
2.1.3 Componentes de un sistema de visión artificial.....	10
2.2 Imagen digital	11
2.2.1 Modelo de color RGB.....	12
2.2.2 Modelo de color TRUE COLOR.....	12
2.2.3 Almacenamiento de imágenes digitales.....	12
2.2.4 Procesamiento de imágenes	12
2.2.4.1 Imagen digital monocromática	12
2.2.4.2 Extracción de canales RGB	13
2.3 Algoritmos de procesamiento de imágenes	14
2.3.1 Algoritmo Canny	14
2.3.2 Transformada de Hough	16
2.4 Software utilizado en el desarrollo del proyecto.....	17
2.4.1 Lenguaje de programación: Python	17
2.4.2 PyCharm	17
2.4.3 OpenCV	18
2.4.4 NumPy	18
2.4.5 Tkinter.....	18
3 Desarrollo del TFG.....	19

3.1 Relación entre píxeles y grados	19
3.1.1 Relación a partir del diámetro del Sol	20
3.1.2 Relación a partir de la calibración de la cámara	21
3.2 Captura de la imagen.....	22
3.2.1 Dispositivo de Captura	22
3.2.2 Proceso de Captura	23
3.2.3 Manejo de Condiciones Ambientales	23
3.3 Procesamiento de la imagen.....	24
3.4 Detección de bordes	24
3.4.1 Algoritmo Canny	24
3.5 Localización del Sol y del horizonte	25
3.5.1 Transformada de Hough	25
3.6 Coordenadas y dimensiones de los elementos	26
3.7 Cálculo de la altura	27
3.8 Interfaz de usuario.....	28
3.9 Repositorio <i>GitHub</i>	29
4 Validación y resultados	30
4.1 Validación de los Procesos	30
4.1.1 Elección del método de escalado de la imagen.....	30
4.1.2 Evaluación de la Precisión de la Detección de Bordes.....	30
4.1.3 Verificación de la Exactitud en la Detección de Líneas y Círculos.....	32
4.1.4 Análisis de la Estabilidad bajo Variaciones.....	33
4.2 Resultados	34
4.2.1 Comparación con la altura obtenida con el sextante	34
4.2.2 Interpretación de los Resultados	35
5 Conclusiones y líneas futuras	36
5.1 Conclusiones	36
5.2 Líneas futuras	37
6 Bibliografía.....	38
Anexo I: Implicaciones Sociales, Económicas y Ambientales	40
Anexo II: Reflexiones éticas y sociales.....	41

ÍNDICE DE FIGURAS

Figura 2-1 Componentes de un sistema de visión artificial.....	10
Figura 2-2 Matriz monocromática.....	13
Figura 2-3 Extracción de canales RGB de una imagen.....	14
Figura 2-4 Gradiente y ángulo del gradiente.....	15
Figura 2-5 Escaneo de los píxeles	15
Figura 2-6 Representación nivel de intensidad de píxel.....	16
Figura 3-1 Diagrama de bloques del funcionamiento del programa	19
Figura 3-2 (a) Sextante, (b) Uso del sextante	20
Figura 3-3 Esquema de determinación del diámetro angular del Sol	20
Figura 3-4 Imagen utilizada para la calibración	21
Figura 3-5 Esquema de situación de la cámara	22
Figura 3-6 (a) Imagen original RGB, (b) imagen escala de grises	24
Figura 3-7 Imágenes obtenidas al ejecutar <i>Canny</i> (a) RGB, (b) Escala de grises.....	25
Figura 3-8 Detección de la circunferencia alrededor del Sol y del horizonte	25
Figura 3-9 Obtención de las coordenadas del Sol y la línea del horizonte.....	27
Figura 3-10 Extracto del código de Python para el cálculo de la altura.....	27
Figura 3-11 Captura de consola de comandos <i>Python</i>	28
Figura 3-12 Interfaz de usuario	28
Figura 3-13 Selección de la región de interés (ROI).....	28
Figura 3-14 Visualización de resultados	29
Figura 4-1 Ejemplo de detección de bordes con cielo parcialmente cubierto.....	31
Figura 4-2 Ejemplo de detección de bordes con cielo despejado.....	31
Figura 4-3 Ejemplo de detección de bordes próximo al ocaso.....	32
Figura 4-4 Gráfica de evolución de alturas en grados frente al tiempo del 18 de marzo de 2024 ...	34
Figura 4-5 Gráfica de evolución de alturas en grados frente al tiempo del 21 de marzo de 2024	35

ÍNDICE DE TABLAS

Tabla 2-1 Procesos de la visión artificial	9
Tabla 3-1 Valores intensidad de luz	13
Tabla 3-1 Comparativa <i>GoPro Hero 5</i> con <i>iPhone SE 2ª generación</i>	22

1 INTRODUCCIÓN Y OBJETIVOS

1.1 Contextualización y motivación

La exploración de los fenómenos celestes ha cautivado a la humanidad a lo largo de los siglos, siendo el Sol un objeto de especial interés debido a su papel fundamental en disciplinas como la astronomía, la meteorología y la navegación astronómica. En este contexto, este Trabajo de Fin de Grado se enfoca en la determinación precisa de la altura solar, un problema históricamente desafiante que requiere de instrumentación costosa y conocimientos especializados. La propuesta de este estudio consiste en la implementación de un sistema de visión artificial, una tecnología emergente que ha demostrado su eficacia en la interpretación y análisis de imágenes. Al aprovechar algoritmos computacionales y técnicas de procesamiento de imágenes, se busca desarrollar un método innovador y eficiente para calcular la altura del Sol a partir de su posición aparente en imágenes capturadas. Esto no solo ofrece una solución técnica al desafío planteado, sino que también representa un avance significativo en la comprensión de fenómenos naturales por inteligencia artificial. En esta introducción, se presenta el contexto y la motivación detrás de este proyecto, así como un esbozo de los objetivos y metodología propuestos para su realización.

Actualmente, es inconcebible hablar de tecnología sin mencionar la Inteligencia Artificial. En los últimos años, esta disciplina ha adquirido una enorme importancia en multitud de campos, y la Armada [1], comprometida con la evolución a una armada 4.0, no ha permanecido ajena a esta tendencia.

El Almirante General Jefe de Estado Mayor de la Armada destaca en las líneas generales para la Armada la importancia de la transformación digital como un medio para mantener una superioridad tecnológica sobre los adversarios, y de invertir en I+D+i con el fin de adaptar la flota a los tiempos actuales, incorporando equipos, técnicas y tecnologías de última generación para garantizar la eficiencia de sus unidades.

A través del uso de algoritmos computacionales, la visión artificial capacita a las máquinas para interpretar y comprender el contenido visual de las imágenes, lo que abre nuevas posibilidades para la automatización y la precisión en la medición de fenómenos naturales.

Por ello, la visión artificial es una herramienta poderosa para abordar el desafío de calcular la altura del Sol, un problema que desde hace cientos de años requiere de instrumentación costosa y compleja, así como de conocimientos especializados.

En este proyecto, la idea inicial consiste en crear un nuevo sistema para determinar la altura del Sol, buscando desarrollar un método preciso y eficiente para determinar esta altura basándose en la posición aparente del Sol en la imagen capturada mediante el uso de algoritmos de procesamiento de imágenes y

técnicas de análisis. Por tanto, se explora el potencial de la visión artificial, su uso práctico en diferentes ámbitos y concluye con el desarrollo de un programa con el que obtener los valores de la altura del Sol de forma autónoma.

1.2 Objetivos

El objetivo principal de este trabajo de fin de grado se concentra en la determinación de la altura del Sol instrumental, es decir, la distancia angular entre el horizonte del mar y el limbo inferior del Sol, mediante un sistema de visión artificial de bajo coste mediante la captura de imágenes en las que aparecen el Sol y el horizonte del mar con un dispositivo *iPhone*.

Para ello, se busca desarrollar un programa capaz de determinar de forma automática la altura del Sol con una simple imagen con los siguientes objetivos secundarios:

- Detección de bordes en imágenes del Sol. Se explorará el empleo de algoritmos, como *Canny*, y su efectividad en la detección de bordes en imágenes con alta iluminación.
- Localización del Sol y del horizonte en imágenes. Se estudiará el empleo de algoritmos, como la *Transformada de Hough*, para la correcta detección de los elementos mesurables en la imagen.
- Calibración y validación del sistema. Se llevará a cabo un estudio experimental sobre la calibración de lentes con la que se tomarán medidas. Estas se compararán con instrumentos especializados en la medición de la altura solar determinando así la precisión del sistema de visión artificial contrastada con métodos tradicionales.

1.3 Estudios relacionados

1.3.1 Sistema automático de geolocalización basado en los astros: módulo de adquisición de datos

Este Trabajo de Fin de Grado [2] desarrolla un sistema automático para la obtención de la posición mediante observación de astros con una aplicación para ejecutar en dispositivos *Android* creada por el autor en lenguaje de programación *Java*.

Después de completar la implementación de esta aplicación, lleva a cabo pruebas exhaustivas para evaluar su funcionamiento. Los resultados obtenidos evidencian la factibilidad y eficacia del enfoque adoptado, lo que promueve la perspectiva de una evolución continua en el futuro y como objetivo último propone la creación de un sistema completamente autónomo capaz de determinar su posición de manera precisa.

A diferencia de un sistema puro de visión artificial, este sistema basa su funcionamiento en el uso de los sensores internos del dispositivo utilizado, como acelerómetros, giróscopos, magnetómetros y brújulas.

1.3.2 Sistema de posicionamiento solar para buques

Este Trabajo de Fin de Grado [3] se centra en el análisis de los métodos matemáticos basados en la trigonometría esférica para la determinación de la posición a través de la observación solar. Se desarrolla una aplicación que presenta estos métodos de forma clara y educativa. El resultado final consiste en una interfaz que, utilizando la altura solar calcula automáticamente la posición aplicable en embarcaciones civiles y militares en navegación oceánica, donde las referencias costeras no están disponibles.

Es interesante resaltar que la visión artificial supone un gran avance para el sistema de posicionamiento solar en buques ya que de forma automática puede actualizar la información sin necesidad de un operador.

1.4 Organización de la memoria

En este primer capítulo, luego de proporcionar contexto sobre el trabajo, clarificar sus objetivos y exponer trabajos del mismo ámbito, se procede a detallar la estructura de este y a desglosar sus secciones.

En el capítulo 2, se presenta el estado del arte de la visión artificial. Inicialmente en el apartado 2.1, se pretende introducir al lector dando una visión general de lo que consiste un sistema de visión artificial. A continuación, en el apartado 2.2, se abordan las nociones básicas sobre la imagen digital que facilitan la comprensión de este proyecto. En el apartado 2.3, se explican los principales algoritmos de procesamiento de imágenes utilizados en el desarrollo. Y finalmente en el apartado 2.4 se exponen las herramientas de Software utilizadas en el desarrollo.

El capítulo 3 constituye el desarrollo de este TFG. Comienza con el apartado 3.1, en el cual se profundiza en la relación píxel-grado. Seguidamente, en el apartado 3.2, se aborda con la metodología de captura de imágenes desde el dispositivo seleccionado, junto con las consideraciones técnicas pertinentes. Posteriormente, en el apartado 3.3, se expone con detalle el proceso de procesamiento de imágenes. En los apartados 3.4 y 3.5 se hace uso de los principales algoritmos empleados. A continuación, en el apartado 3.6, se realiza la determinación de las coordenadas y dimensiones de los elementos detectados en las imágenes. El apartado 3.7 se dedica al cálculo de la altura del Sol, detallando los métodos y procedimientos utilizados para esta tarea fundamental en el contexto del proyecto. Finalmente, en el apartado 3.8, se describe la interfaz de usuario desarrollada y en el apartado 3.9 se facilita el repositorio donde se encuentran los códigos utilizados.

En el capítulo 4 se presentan los resultados obtenidos tras la realización de diferentes pruebas. En el apartado 4.1 se presenta la validación de los procesos llevados a cabo, comenzando con la elección del método de relación píxel-grado para el cálculo de la altura y finalizando, comentando los resultados obtenidos tras la aplicación de los diferentes algoritmos en las imágenes. Finalmente, en el apartado 4.2 se muestran los resultados obtenidos. Se comparan con métodos verificados se lleva a cabo una interpretación de estos.

En el capítulo 5, se exponen las conclusiones respecto a los objetivos establecidos al comienzo del proyecto, así como las líneas de mejora y futuras encontradas tras la finalización del Trabajo de Fin de Grado.

El capítulo 6 consta con la bibliografía utilizada.

Finalmente, se pueden consultar el Anexo I, sobre las implicaciones sociales, económicas y ambientales, y el Anexo II sobre las reflexiones éticas y sociales.

2 ESTADO DEL ARTE

En esta sección se describen los principios teóricos de la visión artificial, así como los últimos avances e investigaciones llevadas a cabo en relación con la determinación de la altura solar mediante técnicas de visión artificial en ámbitos, como la navegación astronómica, la meteorología, la agricultura o la astronomía.

En primer lugar, es necesario conocer cómo funciona la visión humana. La retina funciona como un dispositivo fotosensible para la captación de información del entorno a través de un sistema de visión estereoscópico. Dentro del globo ocular, los receptores conocidos como bastones y conos son responsables de la detección de los colores básicos. Posteriormente, esta información es transmitida al cerebro por medio del nervio óptico, donde es sometida a un complejo proceso de análisis por millones de neuronas distribuidas en múltiples capas. Estas capas neuronales cumplen diversas funciones, desde la detección de los límites y bordes en las imágenes hasta la identificación de contornos y otros aspectos como la profundidad y el movimiento [4].

2.1 Introducción a la visión artificial

La visión artificial es una rama de la inteligencia artificial (IA) que otorga a los ordenadores y sistemas la capacidad de ver, observar y comprender su entorno con el objetivo de extraer información relevante a de imágenes digitales, vídeos u otras entradas visuales con la que pueden llevar a cabo acciones o recomendaciones [5].

Según la *Automated Imaging Association* [6], la visión artificial implica una combinación de *hardware* y *software* que proporciona guía operativa a un dispositivo para realizar funciones basadas en la captura y procesamiento de imágenes, produciendo resultados numéricos y símbolos.

La visión artificial funciona de manera similar a la visión humana. El objetivo es almacenar información para que pueda ser procesada al igual que los humanos usamos nuestros ojos y nuestro cerebro para comprender el mundo que nos rodea, con la diferencia que no poseen un contexto y un aprendizaje que les permite obtener información a través de la visión. Por ello, es necesario entrenar a las máquinas para su funcionamiento.

2.1.1 Evolución de la visión artificial

La noción de visión artificial nació en la década de los sesenta con el objetivo primordial de establecer una conexión entre una cámara de vídeo y una computadora. La experimentación inicial se remonta a 1959, cuando un grupo de neurofisiólogos expuso a un gato a una serie de imágenes con la intención de comprender cómo el animal las percibía y procesaba. Durante este estudio, observaron que las respuestas iniciales del gato estaban dirigidas principalmente a los bordes o líneas sólidas. Este

hallazgo llevó a la conclusión de que el procesamiento de imágenes en el animal comenzaba con formas simples, como los bordes [5].

Simultáneamente, se inició el desarrollo de la primera tecnología de escaneo artificial de imágenes, posibilitando que las computadoras pudieran digitalizar y capturar imágenes. Logrando convertir imágenes bidimensionales en representaciones tridimensionales mediante el empleo de ordenadores.

Uno de los hitos que marcó los inicios de la visión artificial fue el desarrollo, por parte de Larry Roberts de un programa llamado *Mundo de Microbloques*, en el que un robot tenía la capacidad de capturar una estructura de bloques sobre una mesa, analizar su contenido y reproducirla desde otra perspectiva. Se logró que la imagen capturada por la cámara fuese procesada de manera adecuada por el ordenador [7].

Al mismo tiempo, surgió la inteligencia artificial como un ámbito de estudio académico, y esta empezó a ser aplicada en el ámbito de la visión artificial. A comienzos de siglo, la investigación se centraba en el reconocimiento de objetos, y para el año 2001 ya se estaban desarrollando aplicaciones específicas para el reconocimiento facial [5].

En la actualidad, la visión artificial abarca la adquisición, caracterización e interpretación de objetos presentes en una imagen, siendo esencial para potenciar la autonomía en robótica. Además, los métodos empleados en visión artificial son considerablemente más asequibles en comparación con otras alternativas que proporcionan resultados similares [8].

2.1.2 Procesos de la visión artificial

La visión artificial es el proceso de adquirir, identificar e interpretar información de imágenes del mundo tridimensional a partir de imágenes bidimensionales [9]. Estos métodos se pueden dividir en seis subprocesos mostrados en la Tabla 2-1.

Procesos	Nivel de visión	Entrada	Salida	Área
Captura	Bajo	Imagen	Imagen	Procesamiento de imágenes
Preprocesamiento				
Segmentación	Medio	Imagen	Grupo de píxeles (objetos o regiones)	Análisis de imágenes
Descripción		Objeto o regiones	Información cuantitativa	
Reconocimiento		Información cuantitativa	Objetos clasificados en categorías	
Interpretación	Alto	Objetos clasificados en categorías	Comprensión de la escena	Visión por computadora

Tabla 2-1 Procesos de la visión artificial [9], [10]

La captura es el proceso básico de capturar imágenes digitales utilizando un dispositivo adecuado, como una cámara digital, un escáner, un satélite, etc. Implica la conversión de un formato analógico a un formato digital de la imagen.

El preprocesamiento tiene como objetivo mejorar la calidad de las imágenes capturadas. Mediante técnicas como la corrección de color, el filtrado y la reducción de ruido, facilitando así el análisis y la extracción de información en aplicaciones como el reconocimiento de objetos o la detección de patrones en imágenes.

La segmentación consiste en la división de una imagen en porciones de interés para su estudio.

La descripción consiste en identificar y extraer características relevantes de una imagen que permitan diferenciar entre diferentes elementos presentes en ella. Estas características pueden ser externas (forma, color, perímetro...), o internas (textura, distribución de píxeles, áreas...)

El reconocimiento, actual línea de investigación, consiste en la clasificación de los objetos utilizando la información del proceso previo. Las detecciones se agrupan de forma automática en grupos semejantes utilizando métodos estadísticos basados en la apariencia, redes neuronales artificiales, etc.

La interpretación es el proceso de asignar significado o importancia a una categoría de objetos generada en el proceso anterior. Para comprender lo que está sucediendo, se simula la percepción humana para tomar decisiones utilizando métodos inteligentes.

2.1.3 Componentes de un sistema de visión artificial

Previo al conocimiento de los procesos llevados a cabo en un sistema de Visión artificial, es preciso conocer los componentes implicados [8], [11]:

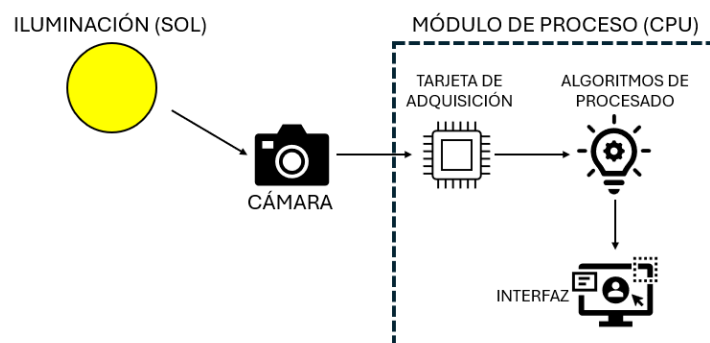


Figura 2-1 Componentes de un sistema de visión artificial

- i) Iluminación. Los objetivos fundamentales de la iluminación son optimizar el contraste, normalizar posibles variaciones en la iluminación ambiental y facilitar el procesamiento de la imagen. En este proyecto en particular, la iluminación no es relevante ya que se va a fotografiar el Sol.
- ii) Cámara. Es un dispositivo que, mediante un conjunto de lentes y un sensor, captura la luz reflejada y la utiliza para generar imágenes. En un sistema de visión artificial, la cámara actúa como sensor, y la elección del tipo de cámara y óptica adecuados se realiza en función de la aplicación y los requisitos específicos.
 - (1) Lentes. Capturan la imagen y la dirigen hacia el sensor en forma de luz. La mayoría de las cámaras de visión ofrecen dos tipos de lentes: intercambiables y fijos. Los lentes fijos, como los presentes en la cámara utilizada en este trabajo suelen contar con enfoque automático, manteniendo un campo de visión fijo a una distancia específica.

- iii) Módulo de proceso (CPU): utilizando algoritmos especializados, controla la imagen, extrae información relevante, lleva a cabo la inspección y toma decisiones.
- (1) Tarjeta de Adquisición. Permite la transferencia de la imagen desde la cámara a la memoria de la computadora, permitiendo que esta realice el procesamiento necesario. Aspectos cruciales que considerar incluyen la velocidad de transmisión, el formato de los datos, la capacidad de preprocesamiento de la imagen, la velocidad de transferencia hacia la memoria de la computadora y la capacidad de controlar la cámara en tiempo real.
 - (2) Algoritmos de Procesado. Representa la inteligencia del sistema, ya que implica la aplicación de transformaciones y extracciones de información de las imágenes capturadas. El propósito es obtener los resultados deseados según el diseño del sistema.
 - (3) Interfaz. Después de llevar a cabo el procesamiento de la imagen mediante los algoritmos, es útil presentar los resultados de manera accesible al usuario final. Por lo tanto, se crea una interfaz para mostrar los resultados de acuerdo con las necesidades específicas.
- iv) Comunicaciones: por lo general, se efectúa a través de señales de entrada/salida discreta o mediante una conexión serial con un dispositivo encargado de registrar o utilizar la información.

2.2 Imagen digital

Previo al desarrollo de programas para el procesamiento de imágenes, es esencial comprender cómo se representan estas imágenes dentro de un ordenador.

Los sistemas informáticos operan exclusivamente con códigos binarios, es decir, con secuencias de ceros y unos que representan información. Por ello, podemos conceptualizar una imagen, en términos iniciales, como un conjunto de datos numéricos dispuestos en una matriz bidimensional, organizados según coordenadas (x, y) [12].

Cada celda de esta matriz registra un valor que indica el color del punto correspondiente. En una imagen en escala de grises, cada una de estas celdas representa el nivel de luminancia o la intensidad de gris. Un valor de 0 representa negro y un valor de 255 representa blanco. De esta forma, cada celda de la matriz almacena un byte, dando acceso a los 256 valores mencionados anteriormente aptos para representar imágenes en escala de grises.

Por otro lado, para la imagen digital a color existen dos métodos de almacenamiento en la matriz:

- **Paleta de color.** Se establece una tabla que contiene los colores presentes en la imagen, representados por sus componentes R, G y B. En la matriz de la imagen, se guarda el índice de la entrada en la paleta que corresponde al color del punto que se está describiendo. Por ejemplo, si la entrada número 25 de la paleta tiene los valores (255, 0, 0) para el rojo puro, todos los puntos de la imagen que sean rojos saturados se referenciarán con el valor 25.
- **TRUE COLOR.** En este esquema, cada punto de la imagen almacena las tres componentes de color que lo definen. Con una profundidad de color de 8 bits, cada punto se representa con 3 bytes, lo que equivale a 24 bits de color.

El formato de paleta de color resulta útil para la representación básica de imágenes, pero cuando se requiere procesarlas, es necesario convertirlas al formato *TRUE-COLOR*. Esta conversión se lleva a cabo de manera inmediata, reemplazando cada índice de la matriz de la imagen por el color correspondiente en la paleta. La limitación del formato de paleta de color para el procesamiento radica en que la aplicación de cualquier tipo de procesamiento a la imagen genera nuevos colores que no están predefinidos, lo que imposibilita su representación. Además, implica que los puntos no son independientes entre sí, es decir, todos los puntos con el mismo valor comparten el mismo color. Por lo

tanto, si se necesita modificar un único punto, sería necesario generar una nueva entrada en la paleta para ese color específico o localizar ese color entre los ya definidos [12].

2.2.1 Modelo de color RGB

El modelo de color RGB (acrónimo de Rojo, Verde, Azul) se basa en la síntesis aditiva de los tres colores primarios y permite representar un color mediante la combinación de estos componentes. Aunque el nombre del modelo proviene de los colores primarios, no define específicamente qué tonalidades corresponden a cada uno de ellos. Por lo tanto, los mismos valores RGB pueden producir colores muy diferentes en distintos dispositivos [9].

La codificación de intensidad de color en el modelo RGB es fundamental para representar de manera precisa los colores en dispositivos digitales. Cada color primario se representa mediante un valor numérico desde 0 hasta 255, lo que permite una amplia gama de tonalidades. Por ejemplo, el color azul se representa como (0, 0, 255), los valores 0 indican la ausencia total de rojo y verde, mientras que el valor 255 indica la máxima intensidad de azul. Esta notación proporciona un sistema estandarizado para especificar cualquier color dentro del espacio de color RGB, lo que facilita la comunicación y la reproducción precisa de los colores en diferentes dispositivos y plataformas [13], [14].

2.2.2 Modelo de color TRUE COLOR

Se denomina *TRUE COLOR* debido a que se aproxima al número de colores capaces de detectar por el ojo humano. Cada celda RGB tiene 8 bits asociados, obteniéndose 256 valores de cada color y dando lugar hasta 16.777.216 configuraciones de color para cada píxel [15].

2.2.3 Almacenamiento de imágenes digitales

El almacenamiento y gestión de imágenes digitales es fundamental en múltiples campos, desde la fotografía y el diseño gráfico hasta la medicina y la ingeniería. Con el avance tecnológico y la proliferación de dispositivos digitales, la cantidad de imágenes generadas y almacenadas ha experimentado un crecimiento exponencial.

El almacenamiento de imágenes digitales abarca una variedad de aspectos, desde los diferentes formatos de archivo que se utilizan para almacenar imágenes hasta las opciones de almacenamiento local en dispositivos físicos y el almacenamiento en la nube. La elección del formato de imagen adecuado y del método de almacenamiento influye en la cantidad de información almacenada, en la calidad de la imagen, el espacio requerido y la facilidad de acceso y manipulación.

2.2.4 Procesamiento de imágenes

Previo al desarrollo del programa, se han implementado una serie de programas con el propósito de adquirir conocimientos fundamentales en el uso de librerías, procesamiento de imágenes y aplicación de algoritmos relacionados con la visión artificial.

En visión artificial, una imagen se define como una matriz numérica bidimensional que representa visualmente la información capturada por un dispositivo de adquisición de imágenes, como una cámara. Esta matriz está compuesta por valores numéricos que representan la intensidad lumínica o el color de cada píxel en la imagen.

2.2.4.1 Imagen digital monocromática

Una imagen digital monocromática se define como una representación bidimensional de la intensidad lumínica, donde a cada píxel se le asocia un valor que indica la luminosidad o nivel de gris en ese punto específico. Esta definición es comúnmente aplicada a las imágenes en escala de grises. En términos prácticos, una imagen monocromática puede ser visualizada como una matriz de dimensiones $m \times n$, donde cada elemento de la matriz representa un píxel dentro de la imagen. La intensidad de cada píxel está determinada por el valor almacenado en esa posición de la matriz, y esta intensidad es proporcional a la luminosidad del punto correspondiente en la imagen [9].

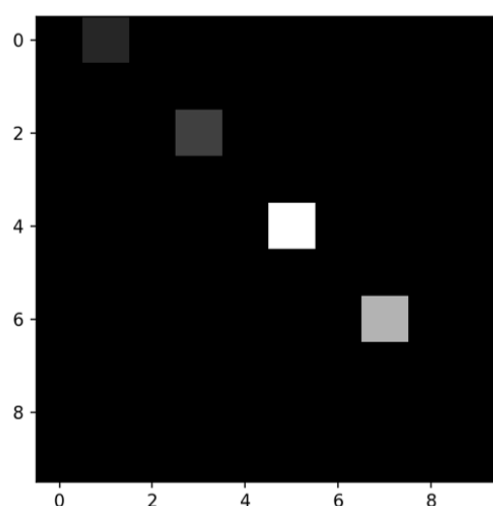


Figura 2-2 Matriz monocromática

La Figura 2-2 muestra una matriz de 10x10 creada en *Python* inicialmente en color negro, es decir, todos los píxeles en valor 0. Se han alterado el valor de intensidad de algunos píxeles reflejado en la Tabla 2-2

Coordenadas píxel	Intensidad de luz (0-255)
(0,1)	30
(2,3)	50
(4,5)	200
(6,7)	140

Tabla 2-2 Valores intensidad de luz

2.2.4.2 Extracción de canales RGB

Una imagen en color RGB se compone de tres imágenes monocromáticas independientes correspondientes a las escalas de rojo (R), verde (G) y azul (B). El método de reproducción del color se basa en la mezcla aditiva de acuerdo con la primera ley de Grassmann. Se comienza con el negro y se añade mayor o menor cantidad de luz a los tres colores básicos (rojo, verde y azul) que conforman el espacio de color RGB [13].

En la Figura 2-3 se muestran los tres canales extraídos y la imagen original. En cada imagen monocromática, el color extraído tiene un valor de intensidad igual a 255 (blanco), mientras que los otros colores no tienen ninguna información y su valor es 0 (negro).

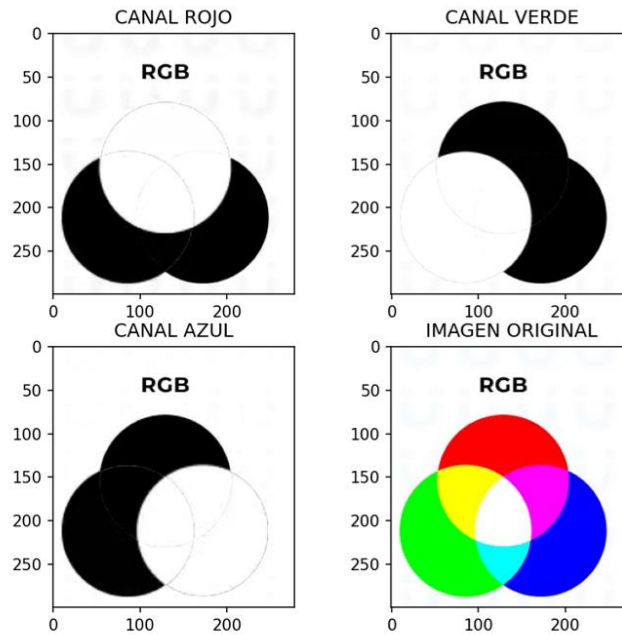


Figura 2-3 Extracción de canales RGB de una imagen

En cuanto a la aplicación en el programa final, inicialmente se lleva a cabo una conversión de RGB a escala de grises con la finalidad de reducir la complejidad computacional, simplificar el procesamiento para la aplicación de los algoritmos y destacar la información relevante para la detección de bordes.

2.3 Algoritmos de procesamiento de imágenes

2.3.1 Algoritmo Canny

El Algoritmo *Canny* [16] es un algoritmo de múltiples etapas que detecta una gran variedad de bordes sobre imágenes. Se basa en el uso de la primera derivada porque toma el valor de cero en todas las regiones donde no varía la intensidad y tiene un valor constante a lo largo de la transición de intensidad. Por lo tanto, un cambio brusco en la intensidad se manifiesta como una variación abrupta en la primera derivada, una característica que se emplea para detectar bordes.

El algoritmo consta de diferentes etapas:

1) Reducción de ruido

Dado que la detección de bordes es susceptible al ruido en la imagen, el primer paso es eliminar el ruido de la imagen con un filtro gaussiano de 5x5.

2) Encontrar el gradiente de intensidad de la imagen

La imagen suavizada se filtra con un núcleo Sobel en ambas direcciones, horizontal y vertical, para obtener la primera derivada en la dirección horizontal (G_x) y vertical (G_y). A partir de estas dos imágenes, podemos encontrar el gradiente de borde (G) y el ángulo (θ) con la dirección para cada píxel de la siguiente manera:

$$(G) = \sqrt{G_x^2 + G_y^2}$$

$$(\theta) = \arctan\left(\frac{G_y}{G_x}\right)$$

La dirección del gradiente siempre es perpendicular a los bordes. Se redondea a uno de cuatro ángulos que representan la dirección vertical horizontal y dos diagonales principales.

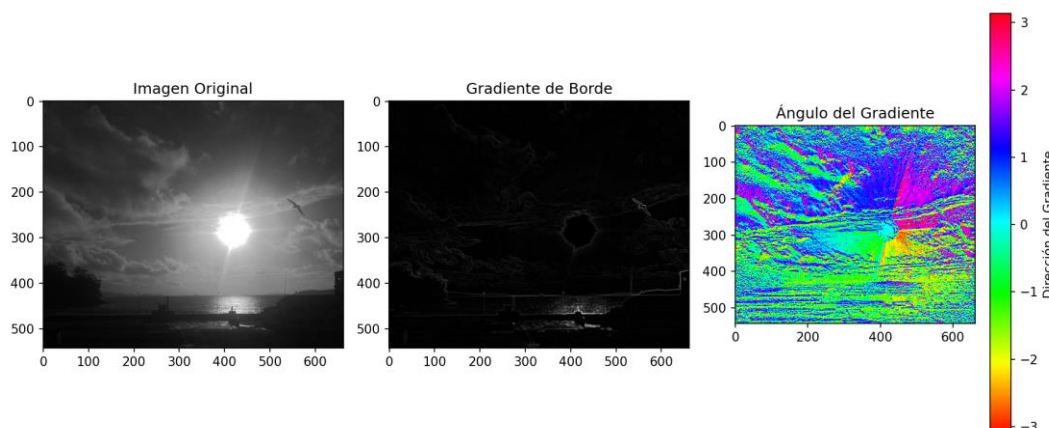


Figura 2-4 Gradiente y ángulo del gradiente

3) Supresión no máxima al resultado del gradiente

Después de obtener la magnitud y dirección del gradiente, se realiza un escaneo completo de la imagen. Como se observa en la Figura 2-5.

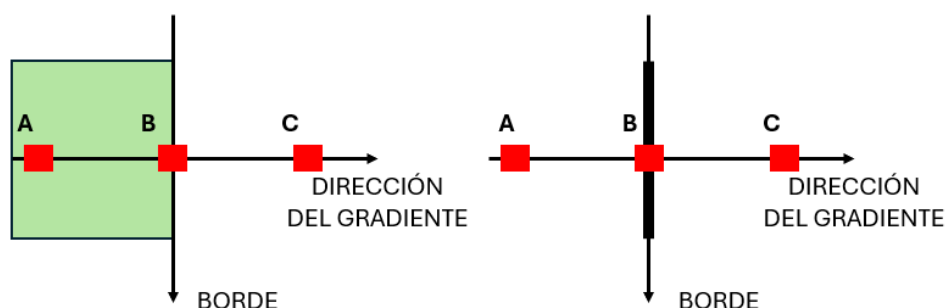


Figura 2-5 Escaneo de los píxeles [16]

A los píxeles A, B y C se encuentran en la dirección del gradiente y el píxel B se encuentra en el punto en la dirección vertical del borde normal a la dirección del gradiente [17], [18].

En el escaneo se comprueba si el píxel es un máximo local en su vecindario en la dirección del gradiente. Si el valor del píxel es mayor o igual que los valores de los píxeles vecinos, se considera un máximo local y se mantiene intacto. De lo contrario, se suprime poniéndose a cero.

Del mismo modo, se verifica si el valor de la magnitud del gradiente en el píxel actual es menor que al menos uno de sus dos vecinos en la dirección del ángulo obtenido en el paso anterior. Si esto ocurre, significa que el píxel actual no es un máximo local en la dirección del gradiente, por lo que se le asigna el valor cero. Si no ocurre, el píxel mantiene su valor de magnitud de gradiente.

Finalmente se obtiene una imagen binaria con bordes delgados.

4) Histéresis de umbral a la supresión no máxima

Esta etapa del algoritmo determina cuáles son realmente bordes y cuáles no lo son. Para ello, se necesitan dos valores de umbral (mínimo y máximo). Cualquier borde con un gradiente de intensidad

mayor que el máximo se considera “borde seguro”, mientras que aquellos por debajo del mínimo se consideran que no son “bordes seguros” y se descartan. Aquellos que están entre estos dos umbrales se clasifican como bordes o no bordes según su conectividad. Si están conectados a píxeles de “borde seguro”, se consideran parte de los bordes. De lo contrario, también se descartan [18].

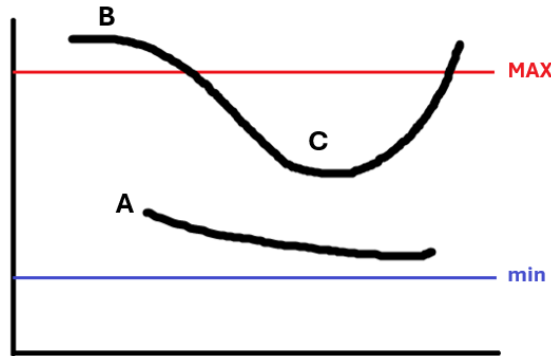


Figura 2-6 Representación nivel de intensidad de píxel [16]

El borde B está por encima del valor máximo, por lo que se considera un "borde seguro". Aunque el borde C está por debajo del máximo, está conectado al borde B, por lo que también se considera un borde válido y obtenemos esa curva completa. Sin embargo, el borde A, aunque está por encima del mínimo y está en la misma región que el borde C, no está conectado a ningún "borde seguro", por lo que se descarta. Por lo tanto, es muy importante seleccionar adecuadamente los umbrales para obtener el resultado correcto.

Esta etapa también elimina pequeños ruidos de píxeles bajo la suposición de que los bordes son líneas largas. Por lo tanto, lo que finalmente obtenemos son bordes fuertes en la imagen.

2.3.2 Transformada de Hough

La *Transformada de Hough* es una técnica para la detección de formas en imágenes digitales. Con este algoritmo es posible encontrar todo tipo de figuras que puedan ser expresadas matemáticamente, tales como rectas o circunferencias.

Aunque este algoritmo es eficiente, requiere un mapa de bordes de la imagen de entrada, del que se obtienen ecuaciones de las figuras a encontrar y puede resultar susceptible al ruido presente en la imagen. Por ello, se recurre al algoritmo *Canny* para aumentar la robustez de los bordes ayudando a mejorar la precisión y confiabilidad en la detección de formas en la imagen original [19].

Hough permite agrupar puntos de borde en posibles candidatos a objetos. Este método funciona mediante un proceso de votación sobre un conjunto de objetos de imagen parametrizados [20]. Implemente una inferencia aproximada de Bayes, partiendo de una distribución uniforme en el espacio de la forma y enfocándose únicamente en la evidencia positiva, lo que permite detectar formas incluso cuando están parcialmente ocultas.

En este proceso, una línea en el espacio de la imagen puede expresarse con dos variables: coordenadas cartesianas como (m, b) y en coordenadas polares como (r, θ) . Las líneas se expresan en el sistema polar, quedando la ecuación de línea como:

Para la *Transformada de Hough*, expresaremos las líneas en el sistema Polar. Por lo tanto, una ecuación de línea puede escribirse como:

$$r = x \cos \theta + y \sin \theta$$

$$y = \left(-\frac{\cos \theta}{\sin \theta}\right)x + \left(\frac{r}{\sin \theta}\right)$$

Al trazar estas ecuaciones para cada punto en la imagen, se genera una "curva" en el espacio (r, θ) correspondiente a cada punto. Es importante señalar que estas "curvas" no tienen la forma física de curvas tradicionales, sino que más bien representan las posibles líneas que podrían pasar por los puntos de la imagen en este espacio paramétrico. La presencia de evidencia de una línea en la imagen se refleja cuando múltiples de estas representaciones se cruzan o se intersecan en el espacio de parámetros.

Para cada punto (x_0, y_0) , podemos definir la familia de líneas que pasa por ese punto como:

$$r_\theta = x_0 \cdot \cos \theta + y_0 \cdot \sin \theta$$

Significando que cada par (r, θ) representa cada línea que pasa por (x_0, y_0) .

La detección de una línea implica encontrar el número de intersecciones entre estas 'curvas'. Cuantas más curvas se crucen, más puntos tiene la línea representada por esa intersección. Se define un umbral del número mínimo de intersecciones necesarias para detectar una línea.

Esto es lo que hace la *Transformada de Hough*. Realiza un seguimiento de la intersección entre curvas de cada punto en la imagen. Si el número de intersecciones está por encima de cierto umbral, entonces la declara una línea con los parámetros (r, θ) del punto de intersección.

Presenta dos variantes utilizadas en el procesamiento de imágenes para detectar líneas:

- Estándar. Trabaja en el espacio de parámetros (r, θ) , donde cada punto en la imagen se mapea a una curva en este espacio. La intersección de estas curvas indica la presencia de una línea en la imagen. Proporciona una precisión muy alta en la detección de líneas.
- Probabilística. Es una versión más eficiente que solo considera un subconjunto aleatorio de puntos en la imagen. Proporciona los extremos de las líneas detectadas en lugar de los parámetros de la línea completa, lo que la hace más rápida pero menos precisa.

Ambas transformadas son implementadas en la biblioteca *OpenCV*, donde la función *HoughLines* se utiliza para la transformada estándar y la función *HoughLinesP* se utiliza para la transformada probabilística.

2.4 Software utilizado en el desarrollo del proyecto

2.4.1 Lenguaje de programación: Python

Python [21] se trata de un lenguaje de programación de propósito general interpretado, con tipado dinámico, fuertemente tipado, multiplataforma y orientado a objetos [22].

Es el lenguaje de programación más utilizado hoy en día, destaca por su similitud con el lenguaje humano y su flexibilidad sintáctica, que permite al usuario una comprensión más accesible y una gran facilidad en la declaración de atributos. Su popularidad se debe a una comunidad activa y a su naturaleza de código abierto, siendo compatible con sistemas operativos como UNIX, Linux, Mac OS, Windows... Esta versatilidad se extiende al ámbito del aprendizaje automático e inteligencia artificial, donde *Python* sobresale gracias a sus librerías especializadas, proporcionando las características necesarias para proyectos diversos [23]. La existencia de estas librerías hace que *Python* sea el lenguaje idóneo para el desarrollo de este proyecto en particular.

2.4.2 PyCharm

PyCharm [24] es un entorno de desarrollo integrado (IDE) diseñado para la programación informática, especialmente orientado al lenguaje de programación *Python*. Desarrollado por *JetBrains*, ofrece características destacadas como análisis de código, un depurador integrado y ejecutor de pruebas;

perfilador *Python*; una terminal integrada e integración con las principales bases de datos integradas y VCS.

Es una herramienta multiplataforma con versiones disponibles para *Windows*, *macOS* y *Linux*. Se presenta en diversas ediciones, entre las cuales se encuentran la versión profesional, educativa, y la seleccionada para este proyecto: *PyCharm Community Edition*, que se distribuye bajo la licencia *Apache* [25].

2.4.3 OpenCV

OpenCV [26] (*Open Computer Vision*) es una biblioteca de visión artificial de código abierto que constituye un conjunto de funciones y algoritmos destinados a la visión por computadora, procesamiento de imágenes y propósitos generales [27].

Esta biblioteca, desarrollada inicialmente en el Centro de Desarrollo de Software de Intel está escrita en un lenguaje de alto nivel e incorpora algoritmos esenciales para tareas presentes en este trabajo, tales como la interpretación de imágenes, la calibración de cámaras, la corrección de distorsiones ópticas, la determinación de la forma y seguimiento de objetos o la reconstrucción tridimensional [27].

La elección de esta biblioteca se fundamenta en su integración eficiente con *Python*. La combinación de *OpenCV* y este lenguaje no solo amplía considerablemente las oportunidades de desarrollo en el ámbito de la visión artificial, sino que también se aprovecha de una interfaz versátil y fácil de utilizar para desarrolladores y la extensa comunidad que *Python* ofrece.

2.4.4 NumPy

NumPy [28] (*Numerical Python*) es una biblioteca de código abierto ampliamente utilizada en los campos de la ciencia y la ingeniería. Reconocida como el estándar por excelencia en el procesamiento de datos numéricos en *Python*, se ha vuelto esencial para la realización de investigaciones y avances científicos e industriales de vanguardia.

Esta biblioteca es especialmente conocida por sus estructuras de datos de matrices. Proporciona un objeto de matriz *n*-dimensional homogéneo que ofrece métodos eficientes para operar en él. Permite realizar una amplia variedad de operaciones matemáticas en matrices, lo que lo convierte en una herramienta poderosa en el ámbito de la visión artificial para el análisis y computación de imágenes.

En el núcleo de *NumPy* se encuentra el objeto *ndarray*, que encapsula matrices *n*-dimensionales con tipos de datos homogéneos. Este objeto permite realizar operaciones de manera eficiente gracias a que la mayoría se ejecutan en código compilado. Además, presenta diferencias importantes con respecto a las secuencias estándar de *Python*, como la fijación del tamaño al crearse y la uniformidad en el tipo de datos.

2.4.5 Tkinter

Tkinter [29] es una biblioteca estándar de *Python* para la creación de interfaces gráficas de usuario (GUI). Conocida por su simplicidad y facilidad de uso. Proporciona herramientas para diseñar ventanas, botones, menús y otros elementos interactivos.

Esta biblioteca se basa en la biblioteca gráfica *Tcl/Tk* y ofrece una integración fluida con *Python*, lo que la convierte en una opción popular para desarrolladores que buscan crear interfaces visuales intuitivas y funcionales. La versatilidad de *Tkinter* permite la creación de aplicaciones con una interacción amigable para el usuario por lo que es idónea para proyectos que requieren una interfaz gráfica eficiente.

3 DESARROLLO DEL TFG

En esta sección se describe el proceso llevado a cabo en la realización del Trabajo de Fin de Grado, comenzando con la captura de imágenes y procesamiento de imagen, continuando con la programación del sistema y la relación de píxeles con unidades angulares para calcular la altura y finalizando con una interfaz básica de usuario.

En cuanto a la programación, se ha seguido el diagrama de bloques de la Figura 3-1. En primer lugar, consta de la captura de la imagen. A continuación, el procesamiento de esta. Después de ello, se lleva a cabo la detección de bordes, la detección de líneas y círculos. Se representan los elementos reconocidos en la imagen (Sol y horizonte) y finalmente se lleva un escalado de distancias y el cálculo de la altura del Sol.

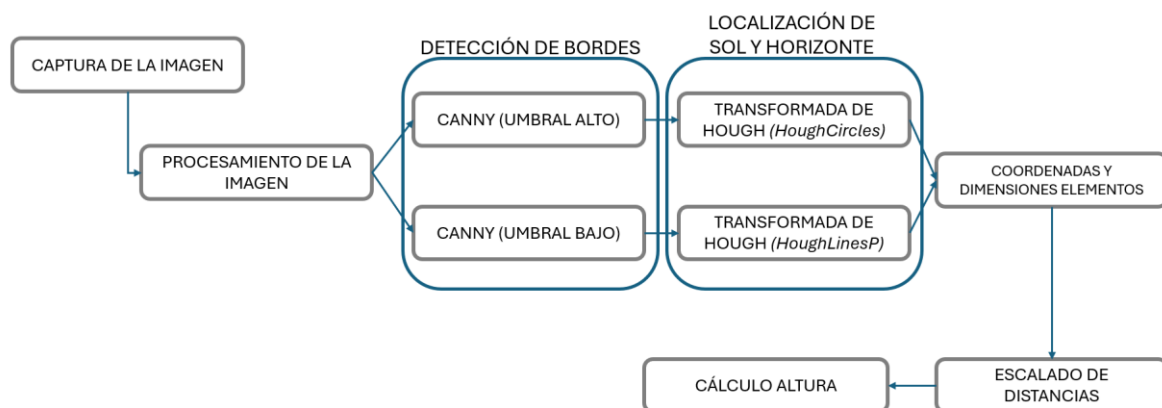


Figura 3-1 Diagrama de bloques del funcionamiento del programa

3.1 Relación entre píxeles y grados

Como se ha descrito anteriormente en el apartado 2.2.4, una imagen se representa como una cuadrícula de píxeles, donde cada píxel posee una ubicación única definida por sus coordenadas. En este código, las coordenadas representan la ubicación de los píxeles dentro de la imagen. Durante la detección de bordes u objetos, se determinan las coordenadas de los píxeles involucrados en dichos bordes u objetos.

Es importante destacar que el sistema de coordenadas tiene su origen en la esquina superior izquierda de la imagen. Esto significa que el píxel en la esquina superior izquierda de la imagen tiene coordenadas (0, 0), mientras que las coordenadas aumentan hacia la derecha en el *eje x* y hacia abajo en el *eje y*. De

esta manera, cada píxel en la imagen se puede ubicar de manera única mediante un par de coordenadas (x, y) .

Para la obtención de la relación de píxeles en la imagen y grados en el mundo real, se llevaron a cabo dos experimentos.

3.1.1 Relación a partir del diámetro del Sol

En primer lugar, se llevó a cabo una relación a partir del diámetro del Sol.

Para realizar la conversión de píxeles a unidades angulares, se llevaron a cabo mediciones utilizando un sextante proporcionado por la Escuela Naval Militar, como se muestra en la Figura 3-2.



(a)



(b)

Figura 3-2 (a) Sextante, (b) Uso del sextante

Para determinar el diámetro angular del Sol, es necesario medir en primer lugar la altura del Sol desde el limbo superior y posteriormente la altura desde el limbo inferior. Una vez se toman estas medidas, se realiza la diferencia entre estas alturas obteniéndose una medición precisa del diámetro angular del Sol. En la Figura 3-3 se puede ver de forma gráfica.

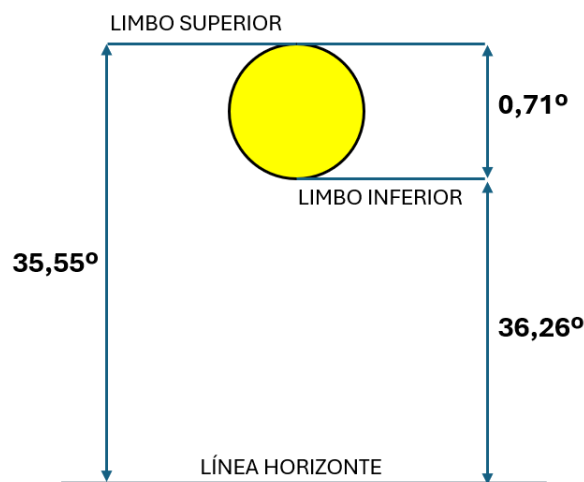


Figura 3-3 Esquema determinación del diámetro angular del Sol

Finalmente, el diámetro angular del Sol se compara con su tamaño en la imagen expresado en píxeles. Esta comparación permite calcular un factor de conversión que transforma las mediciones en píxeles a unidades angulares reales.

Este método resulta interesante ya que presenta la posibilidad de calibrar la cámara realizando unas medidas sencillas con el sextante. Sin embargo, el tamaño del Sol aparente desde la Tierra varía debido al paso de los rayos solares por la atmósfera y particularmente en las imágenes debido a la cantidad de luz presente en ellas y a las variaciones de la luminosidad para diferentes alturas.

Por estas razones, este método no es el más eficiente para la determinación de una relación de medidas.

3.1.2 Relación a partir de la calibración de la cámara

Este método no precisa de información presente en la imagen, si no que previamente a la captura ya está calibrado. Esto otorga una gran flexibilidad para la toma de medidas.

Para la calibración de la cámara, se ha empleado un procedimiento comúnmente utilizado en fotografía. Este procedimiento implica un experimento que utiliza fórmulas trigonométricas para calcular el ángulo de visión y determinar la relación entre píxeles y ángulos en diversas áreas de la imagen.

En primer lugar, es fundamental contar con una imagen cuyas dimensiones sean conocidas. En este procedimiento, se han trazado líneas verticales separadas por una distancia de 4 cm entre sí, colocando la línea central en el centro de la lente, tal como se muestra en la Figura 3-4. Es crucial asegurar una correcta alineación de la lente tanto con las líneas verticales como con la línea central para evitar la introducción de errores.

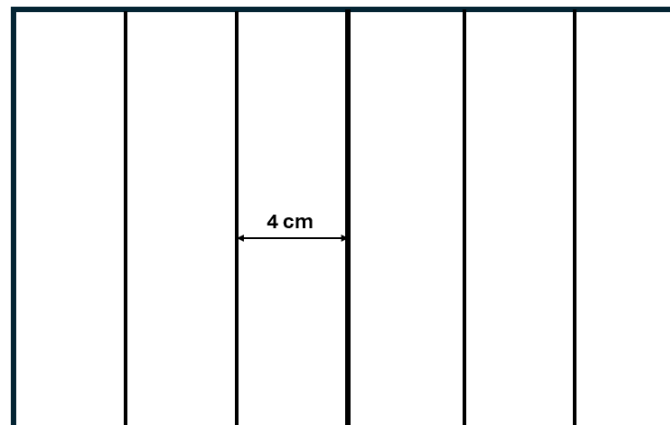


Figura 3-4 Imagen utilizada para la calibración

En segundo lugar, la cámara se posiciona a una distancia predefinida, y utilizando principios trigonométricos, se procede a calcular el ángulo de visión total de la cámara, junto con una relación píxel-gradus. Esta relación se emplea posteriormente para determinar las medidas reales en la imagen capturada. La disposición se esquematiza en la Figura 3-5.

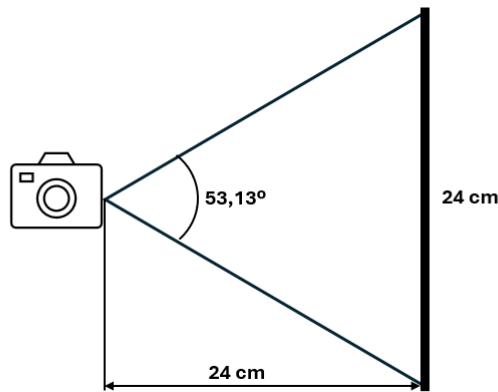


Figura 3-5 Esquema situación de la cámara

Finalmente, se determina que 1 píxel en la imagen equivale a $0,016^\circ$ en la realidad. Esta relación permanece constante en la región central de la imagen, donde las distorsiones son prácticamente imperceptibles extendiéndose hasta aproximadamente una apertura de 50° . Más allá de este umbral, las distorsiones son mínimas y apenas afectan a la calidad de la imagen, lo que destaca la fiabilidad de este método para la captura de imágenes precisas.

3.2 Captura de la imagen

La captura de imágenes constituye un paso esencial en el proceso, pues provee los datos iniciales para el procesamiento y análisis. En esta sección, se abordará en detalle el procedimiento de captura de imágenes.

3.2.1 Dispositivo de Captura

Para el desarrollo de este trabajo se ha realizado una selección de la cámara utilizada previo a la implementación del código. Por un lado, la cámara *GoPro Hero 5*, y por otro el *iPhone SE 2ª generación*.

En la Tabla 3-1 se presentan las principales características de las cámaras utilizadas en el desarrollo del proyecto.

	<i>GoPro Hero 5</i>	<i>iPhone SE 2ª generación</i>
Resolución	12 MPx	12 MPx
Apertura de lente	$f/2,8$	$f/1,8$
Tecnología de captura	WDR	HDR
Enfoque	Automático	Automático
Geoetiquetado	Sí	Sí
Formato de imagen	RAW, JPEG	JPEG, HEIF
Estabilizador de imagen	Sí	Sí

Tabla 3-1 Comparativa *GoPro Hero 5* con *iPhone SE 2ª generación*

Por un lado, la apertura en fotografía desempeña un papel fundamental en la captura de imágenes en condiciones de mucha luz en entornos muy iluminados o como sucede en este proyecto la fotografía del Sol.

Una apertura más amplia permite una mayor entrada de luz en la cámara, lo que facilita la captura de detalles finos y mantiene una exposición equilibrada en condiciones de gran luminosidad. Esto es

especialmente útil para evitar que las áreas de la imagen se sobreexpongan, lo que resultaría en áreas blancas quemadas sin detalles. El valor de la apertura es inversamente proporcional a esta, es decir, cuanto menor es el valor de f , mayor es la apertura. La cámara *GoPro* tiene una apertura de 2,8 mientras que la cámara del *iPhone SE* tiene una apertura de 1,8.

Por otro lado, cada una de las cámaras evaluadas poseen WDR (*GoPro*) y HDR (*iPhone*), dos tecnologías relacionadas con la captura de imágenes que se utilizan en diferentes aplicaciones.

Tanto el WDR (*Wide Dynamic Range*) como el HDR (*High Dynamic Range*) han sido diseñados para escenas con alto contraste de luz y capturar detalles tanto en áreas brillantes como en áreas oscuras de una imagen. Aunque comparten similitudes en su objetivo general, existen diferencias en sus mecanismos de funcionamiento y en las aplicaciones para las que son más adecuados.

El WDR se enfoca principalmente en el procesamiento de hardware en tiempo real y es comúnmente utilizado en dispositivos como cámaras de acción y videocámaras para resaltar detalles en escenas con alto rango dinámico. Mientras que el HDR es una técnica de procesamiento de software utilizada en la fotografía estática para combinar múltiples exposiciones y obtener una imagen final con una gama más amplia de tonos y detalles.

Tras evaluar las características de ambas cámaras y considerar la importancia de la apertura, así como las tecnologías de procesamiento de imágenes (WDR y HDR) que poseen, se ha determinado que la cámara del *iPhone*, con su apertura más amplia y la capacidad de HDR, es más adecuada para la captura de imágenes estáticas en entornos muy iluminados, como la fotografía del Sol. Aunque se han obtenido resultados similares con ambas cámaras, se ha observado que el *iPhone* ofrece una mayor calidad en este tipo de situaciones.

3.2.2 Proceso de Captura

Dado que la altura del Sol se mide desde el nivel del mar, es crucial que el dispositivo esté situado lo más bajo posible. En este caso se coloca en el suelo, en un área abierta y despejada donde se tiene una vista clara del cielo y el horizonte.

Se utiliza la aplicación de cámara del *iPhone*. Se garantiza que la captura de una imagen que posea la información suficiente para su posterior procesamiento.

La tecnología de enfoque automático que posee el dispositivo facilita la captura de la imagen, ya que con un toque en la zona iluminada en la pantalla se obtiene un enfoque óptimo en la imagen capturada, con las zonas sin interés oscurecidas y los bordes del Sol y el horizonte resaltados debido a su luminosidad.

Finalmente, la imagen capturada se guarda en el dispositivo para su posterior procesamiento y análisis.

3.2.3 Manejo de Condiciones Ambientales

El éxito de la determinación de la altura del Sol mediante visión artificial está influenciado por las condiciones ambientales en las que se capturan las imágenes. Variaciones en la iluminación, la presencia de sombras, fenómenos atmosféricos y otros factores pueden influir significativamente en la calidad y la precisión de las imágenes capturadas.

La luz solar directa, la nubosidad, la hora del día, o la presencia de reflexiones y brillos son factores que pueden provocar cambios en la intensidad y distribución de la luz en la escena. Estas diferencias tienen un impacto significativo en el contraste, la saturación del color y el brillo de la imagen capturada. Como resultado, la capacidad de identificar con precisión bordes y elementos relevantes puede verse comprometida, afectando a la precisión de las mediciones realizadas.

Teniendo en cuenta estas condiciones ambientales y afrontándolas, queremos confirmar la calidad y fiabilidad de los resultados de este estudio para la determinación de la altitud solar mediante observaciones artificiales.

3.3 Procesamiento de la imagen

En el procesamiento de imágenes utilizando la conversión de una imagen a escala de grises es un proceso fundamental. Aunque *OpenCV* utiliza el formato BGR (azul, verde, rojo) en lugar de RGB, el proceso de conversión es el mismo, es simplemente la forma de representar en pantalla [30], [31].

La conversión de una imagen de RGB a escala de grises implica calcular el valor de intensidad de cada píxel en la imagen resultante. Este proceso se ha realizado utilizando la función `cv2.cvtColor` de *OpenCV*. Específicamente, se utiliza el código `cv2.COLOR_BGR2GRAY` como argumento de esta función. Obteniéndose la Figura 3-6.



Figura 3-6 (a) Imagen original RGB, (b) imagen escala de grises

El algoritmo utilizado para calcular los valores de los píxeles en la imagen en escala de grises sigue una fórmula estándar. Esta fórmula pondera los valores de los canales de color según su contribución a la luminosidad percibida en el ojo humano [32]. La fórmula generalmente aceptada es:

$$Y=0.299\times R+0.587\times G+0.114\times B$$

Donde R, G y B representan los valores de los canales de color rojo, verde y azul respectivamente, y la Y representa el valor del píxel en la imagen en escala de grises.

Este proceso de conversión simplifica la representación de la imagen al eliminar la información de color, lo que facilita el análisis de características como el contraste y la intensidad de la luz. Además, la imagen en escala de grises resalta las transiciones de intensidad en la imagen, lo que resulta útil en aplicaciones como la detección de bordes y la extracción de características.

3.4 Detección de bordes

La detección de bordes es esencial para la posterior detección de formas. Para llevarse a cabo, previamente se aplica el *Algoritmo Canny* y posterior la *Transformada de Hough*.

3.4.1 Algoritmo Canny

Al ejecutar el algoritmo se obtiene la Figura 3-7. Los resultados obtenidos proporcionan una representación clara de los bordes presentes en la imagen original, lo que facilita la identificación y el análisis de características importantes. Esto es esencial para la posterior detección del Sol y el horizonte.

Además, se puede comparar la diferencia de resultados entre la imagen original de la que previamente ha sido convertida a escala de grises. Esta conversión ayuda a reducir el ruido obtenido debido a la pérdida de información y detalle de la imagen.

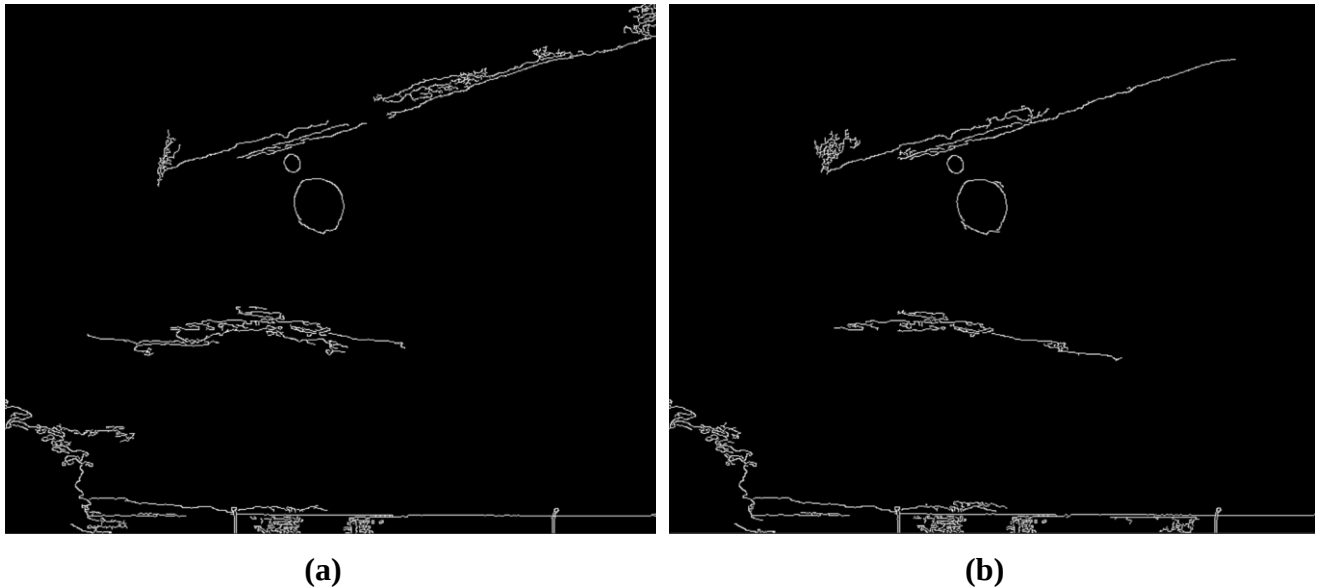


Figura 3-7 Imágenes obtenidas al ejecutar *Canny* (a) RGB, (b) Escala de grises

Se observa ciertas diferencias en las imágenes tras aplicar el algoritmo. Por un lado, el resultado obtenido desde la imagen original, muestra mayor ruido en la zona inferior de la imagen mientras que el resultado obtenido desde la imagen en escala de grises muestra con más claridad esta zona. Sin embargo, en esta última el detalle de los bordes de las nubes es mayor, por lo que existe una mejor detección de esos bordes.

A pesar de ese ruido presente en las imágenes, al no presentar características relevantes, no presentan un impedimento para la posterior detección del Sol y el horizonte.

3.5 Localización del Sol y del horizonte

3.5.1 Transformada de Hough

Mediante el ajuste de los umbrales y parámetros en la *Transformada de Hough*, se ha logrado detectar con éxito la línea del horizonte y los círculos presentes en la imagen, proporcionando así una comprensión detallada de la estructura geométrica de la escena representada en la imagen.

Al ejecutar la transformada se obtiene la Figura 3-8.



Figura 3-8 Detección de la circunferencia alrededor del Sol y del horizonte

En el análisis de la imagen para la detección de la línea del horizonte, se ha aplicado la *Transformada de Hough* utilizando la función `cv2.HoughLinesP` (transformada probabilística). Se establecen umbrales específicos para determinar qué líneas son consideradas como válidas. Del mismo modo, se aplican restricciones en la longitud mínima de las líneas en el atributo *minLineLength* y en la distancia máxima entre segmentos de línea que pueden unirse para formar una sola línea en el atributo *maxLineGap*, lo que ayuda a evitar falsos positivos y mejorar la precisión de la detección.

Además, se filtran las líneas detectadas basándose en su pendiente, descartando aquellas que no se ajustan a una orientación horizontal. Esto se realiza estableciendo un límite para la diferencia en las coordenadas verticales. En este caso, se ha definido un umbral de 10 para la diferencia, lo que significa que solo las líneas con una pendiente que no supere este valor son consideradas como líneas horizontales.

Se ha definido un umbral de confianza en el atributo *param1* de 1400, lo que indica que solo los círculos que reciben al menos 1400 votos son considerados como círculos válidos. Además, se ha especificado un segundo umbral en el atributo *param2* de 55, que determina la sensibilidad del detector de círculos. Un valor más alto reduce la sensibilidad, lo que resulta en la detección de los círculos más prominentes en la imagen. Estos parámetros, junto con los rangos mínimo y máximo de radio de los círculos, *minRadius* y *maxRadius* respectivamente, permiten identificar los círculos presentes en la imagen, como se ha observado en los resultados.

Estos parámetros ayudan a eliminar líneas no deseadas y a garantizar la detección de la línea del horizonte principal en la imagen y se reducen las posibles interferencias de otras líneas presentes en la imagen que no sean horizontales. Es importante destacar que la captura de la imagen desempeña un papel crucial en la detección del Sol, ya que este será el principal círculo presente en la mayoría de los casos. Por lo tanto, la calidad y la composición de la imagen pueden influir significativamente en los resultados de la detección.

3.6 Coordenadas y dimensiones de los elementos

El código almacena la información del círculo y de la línea del horizonte con las siguientes estructuras de datos:

- i) **Círculo.** Las coordenadas del centro del círculo se almacenan en la variable *center*, que es una tupla de la forma (x, y) . Esta variable se actualiza en cada iteración del bucle que recorre los círculos detectados.
El radio del círculo se almacena en la variable *radius*, que se calcula a partir de la información proporcionada por la detección de círculos.
- ii) **Línea.** Las líneas detectadas se almacenan en la lista *lines*, donde cada elemento es una lista de cuatro elementos $[x1, y1, x2, y2]$. Estos cuatro elementos representan las coordenadas de dos puntos que definen la línea: $(x1, y1)$ y $(x2, y2)$.
Después de filtrar las líneas para obtener solo la línea del horizonte, las coordenadas de esta línea se almacenan en la lista *horizon_lines*. Cada elemento de esta lista es similar a los elementos de *lines*, es decir, una lista de cuatro elementos representando los puntos de inicio y fin de la línea horizontal.

Es importante tener en cuenta que el origen de coordenadas se encuentra en la esquina superior izquierda de la imagen para la comprensión de la representación gráfica de estos elementos.

Al ejecutar el código, se genera la Figura 3-9. El programa utiliza las coordenadas almacenadas para dibujar los círculos y la línea en la imagen de salida utilizando las funciones `cv2.circle` y `cv2.line` respectivamente, proporcionadas por la biblioteca *OpenCV*. También se utiliza la función `cv2.putText` para agregar texto que muestra información sobre el círculo y la línea en la imagen.

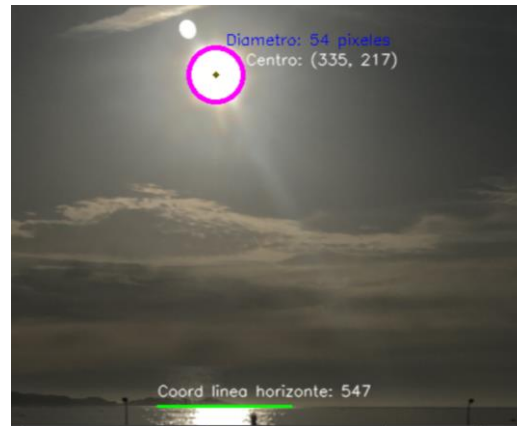


Figura 3-9 Obtención de las coordenadas del Sol y la línea del horizonte

Para la posterior determinación de altura, es de interés la distancia vertical y que corta perpendicularmente la línea del horizonte. Se ha observado en las diferentes imágenes tomadas la detección del horizonte coincide en la vertical del Sol debido al brillo reflejado en la superficie del agua.

3.7 Cálculo de la altura

En este apartado se va a calcular la altura del Sol, concretamente la altura instrumental. Esta altura es la distancia que existe entre el limbo inferior del Sol y la línea del horizonte en el mar.

En primer lugar, se lleva a cabo la determinación de la posición vertical del centro del Sol. Esta distancia se obtiene con la diferencia entre las coordenadas y de los puntos que definen el centro del Sol y la línea del horizonte.

Habiendo obtenido esta distancia, se calcula la distancia entre el limbo inferior del Sol y la línea del horizonte en términos de píxeles mediante la siguiente expresión:

$$\text{Altura (píxeles)} = (\text{centro del Sol} - \text{línea horizonte}) - \text{radio del Sol}$$

Para obtener la distancia desde el borde inferior del Sol hasta la línea del horizonte, simplemente se resta el radio del círculo al valor calculado. Esto asegura que se considere la posición del borde inferior del Sol en el cálculo.

En el código se emplean las líneas de la Figura 3-10.

```
# Calcula la distancia desde el borde inferior del sol hasta la línea horizontal
if circle_center is not None:
    circle_center_y = circle_center[1]
    distance_to_horizon_pixels = abs(circle_center_y - average_horizon_y) - radius
    distance_to_horizon_real = distance_to_horizon_pixels * 0.016 * 5

    measurements["Distancia al horizonte (píxeles)"] = distance_to_horizon_pixels
    measurements["Distancia al horizonte (unidades del mundo real)"] = distance_to_horizon_real
```

Figura 3-10 Extracto del código de Python para el cálculo de la altura

Obteniéndose los resultados de la Figura 3-11.

```
C:\Users\david\PycharmProjects\pythonProject1\.venv\Scripts\Python.exe C:\Users\david\PycharmProjects
Altura desde el borde inferior del sol hasta la línea del horizonte: 303.0 píxeles
Altura desde el borde inferior del sol hasta la línea del horizonte: 4.848 grados
|
```

Figura 3-11 Captura consola comandos *Python*

3.8 Interfaz de usuario

Con el objetivo de facilitar el uso de este sistema, se ha desarrollado una interfaz gráfica para el usuario utilizando la biblioteca *Tkinter*. La interfaz permite cargar una imagen desde el sistema de archivos del usuario y realizar las siguientes operaciones:

- i) **Cargar Imagen.** Al hacer clic en este botón, se muestra un cuadro de diálogo que permite al usuario seleccionar una imagen desde su sistema de archivos.

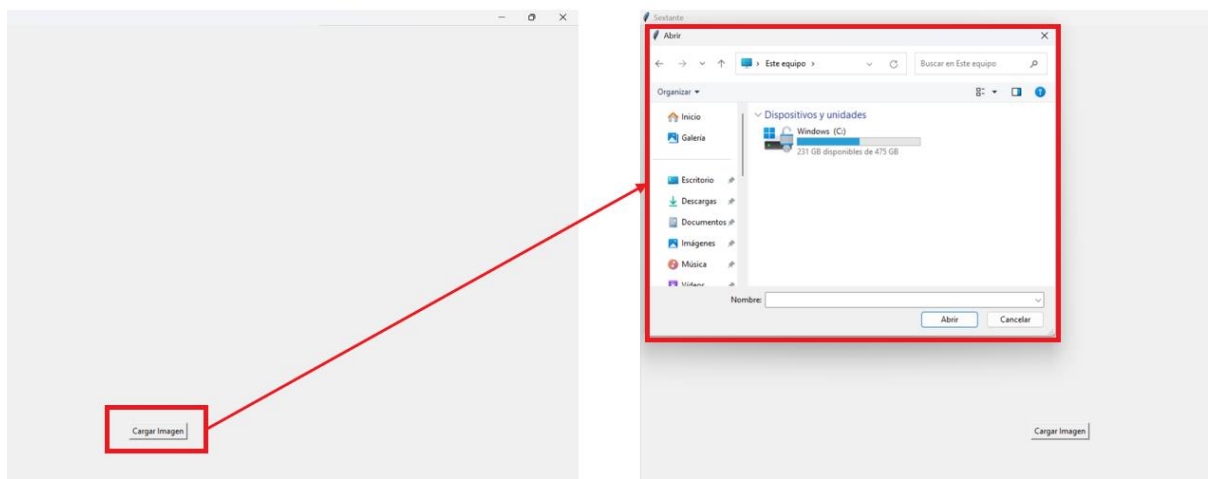


Figura 3-12 Interfaz de usuario

- ii) **Selección de Región de Interés (ROI).** Después de cargar la imagen, el usuario puede seleccionar una región de interés dentro de la imagen utilizando un cuadro de selección mediante la función `cv2.selectROI` de la biblioteca *OpenCV*. Una vez seleccionada, se procesa y se utiliza esta imagen recortada.

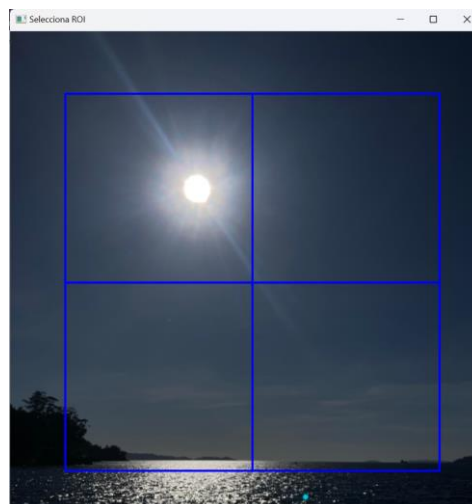


Figura 3-13 Selección de la región de interés (ROI)

- iii) **Visualización de resultados.** Los resultados de la detección de la línea de horizonte y el círculo del Sol se muestran en la interfaz gráfica en un lienzo utilizando la biblioteca *PIL* (*Python Imaging Library*) para la visualización de imágenes en *Tkinter*. Además, se muestran las coordenadas sobre la imagen y las medidas calculadas en una etiqueta informativa.

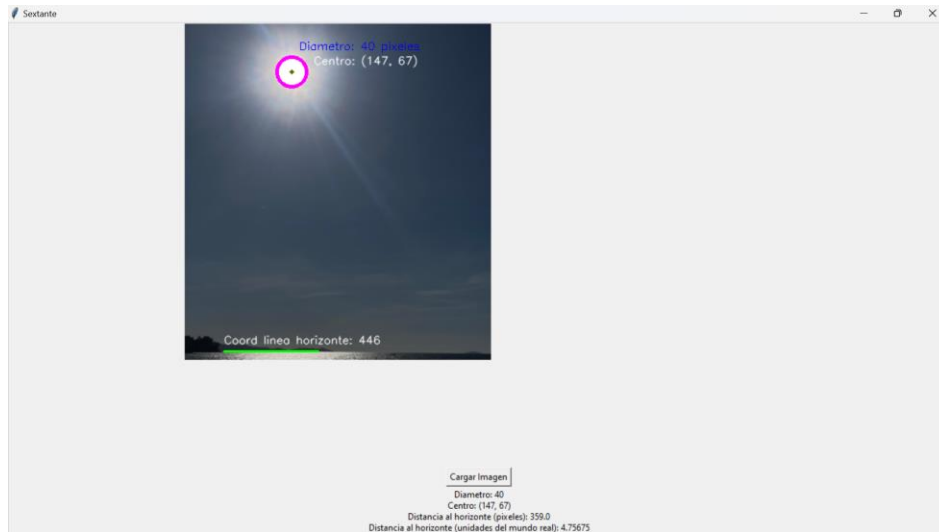


Figura 3-14 Visualización de resultados

- iv) **Ejecución de la aplicación.** La aplicación se ejecuta mediante la llamada al método *run* de la clase *HorizonSunDetector*, lo que permite al usuario interactuar con la interfaz gráfica y realizar las operaciones mencionadas anteriormente.

3.9 Repositorio *GitHub*

Finalmente, se ha concluido con la creación de un repositorio online en la plataforma *GitHub* ampliamente utilizada para divulgación de códigos. [33]

4 VALIDACIÓN Y RESULTADOS

4.1 Validación de los Procesos

4.1.1 Elección del método de escalado de la imagen

Para validar los procesos de escalado de la imagen, se han explorado y comparado dos métodos distintos: la relación entre píxeles y grados derivada del diámetro del Sol detallado en el apartado 3.7 y la calibración de la cámara, detallada en el apartado 3.6. Tras un análisis exhaustivo, se ha determinado que el método de calibración de la cámara es la opción más idónea para garantizar la precisión y fiabilidad en la conversión de medidas de la imagen a unidades angulares reales.

La calibración de la cámara ofrece una serie de ventajas significativas sobre el método basado en el diámetro del Sol. En primer lugar, este método establece una relación, mediante principios trigonométricos, constante entre los píxeles de la imagen y los ángulos en el mundo físico, lo que asegura mediciones consistentes y confiables añadiendo una capa adicional de precisión matemática al proceso.

Además, la calibración de la cámara se lleva a cabo antes de la captura de la imagen, lo que proporciona flexibilidad y control sobre el proceso de medición. Al tener la cámara ya calibrada, se elimina la necesidad de depender de condiciones atmosféricas variables o del comportamiento del Sol, lo que garantiza mediciones precisas independientemente de las circunstancias externas.

Por el contrario, el método basado en el diámetro del Sol presenta ciertas limitaciones que afectan a la precisión. La variabilidad en el tamaño aparente del Sol debido a la atmósfera y las condiciones de iluminación introduce incertidumbre en las mediciones, lo que puede conducir a resultados inconsistentes. Aunque este método ofrece una perspectiva interesante para la calibración de la cámara, su utilidad se ve comprometida por su falta de consistencia y confiabilidad en entornos variables.

4.1.2 Evaluación de la Precisión de la Detección de Bordes

La evaluación de la precisión en la detección de bordes es crucial para comprender el rendimiento del algoritmo en la identificación de los contornos en imágenes. A continuación, se presentan ejemplos visuales para ilustrar la comparación entre los bordes detectados por el algoritmo y los bordes reales en las imágenes de prueba.



Figura 4-1 Ejemplo de detección de bordes con cielo parcialmente cubierto (a) Imagen original (b) Bordes detectados con Canny

En la Figura 4-1 se observa una óptima detección de bordes. A pesar del reflejo generado por el Sol en la lente, no es reconocido como borde por el algoritmo. Por otro lado, las nubes se detectan en la zona más próxima al Sol debido a la luminosidad en esa parte de la foto. Además, cabe destacar que los elementos visibles en la parte inferior de la imagen son fácilmente detectables por el algoritmo introduciendo ruido en la imagen. No son de interés para el proyecto, por lo que es importante evitar elementos en la escena.

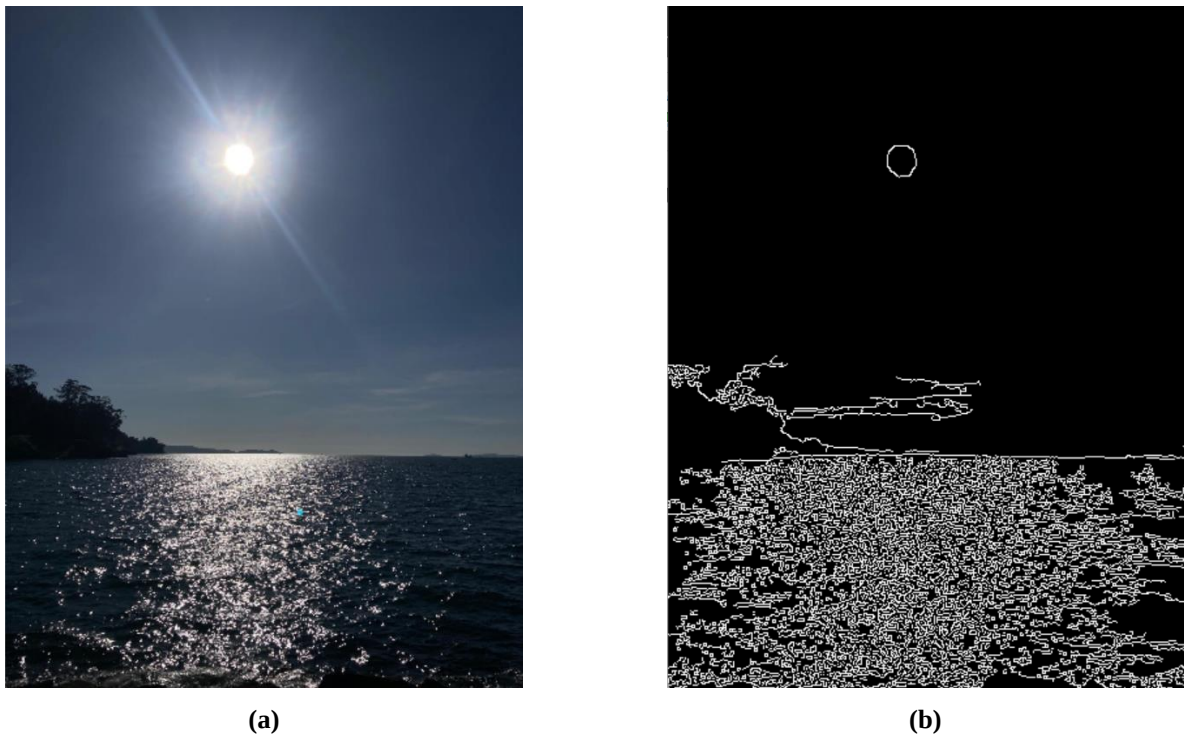


Figura 4-2 Ejemplo de detección de bordes con cielo despejado (a) Imagen original (b) Bordes detectados con Canny

En la Figura 4-2 se observa una correcta detección de bordes. En esta imagen con nubosidad mínima, la detección del Sol se realiza de forma precisa quedando un círculo completo despreciando los artefactos presentes en la imagen y los reflejos generados por la lente de la cámara. Además, se observa una línea

del horizonte bien definida y diferenciada del resto de bordes detectados en la superficie del agua. Aunque estos bordes introducidos no son de interés, posteriormente en la detección de líneas con la *Transformada de Hough* se filtrarán determinando los umbrales correctos.

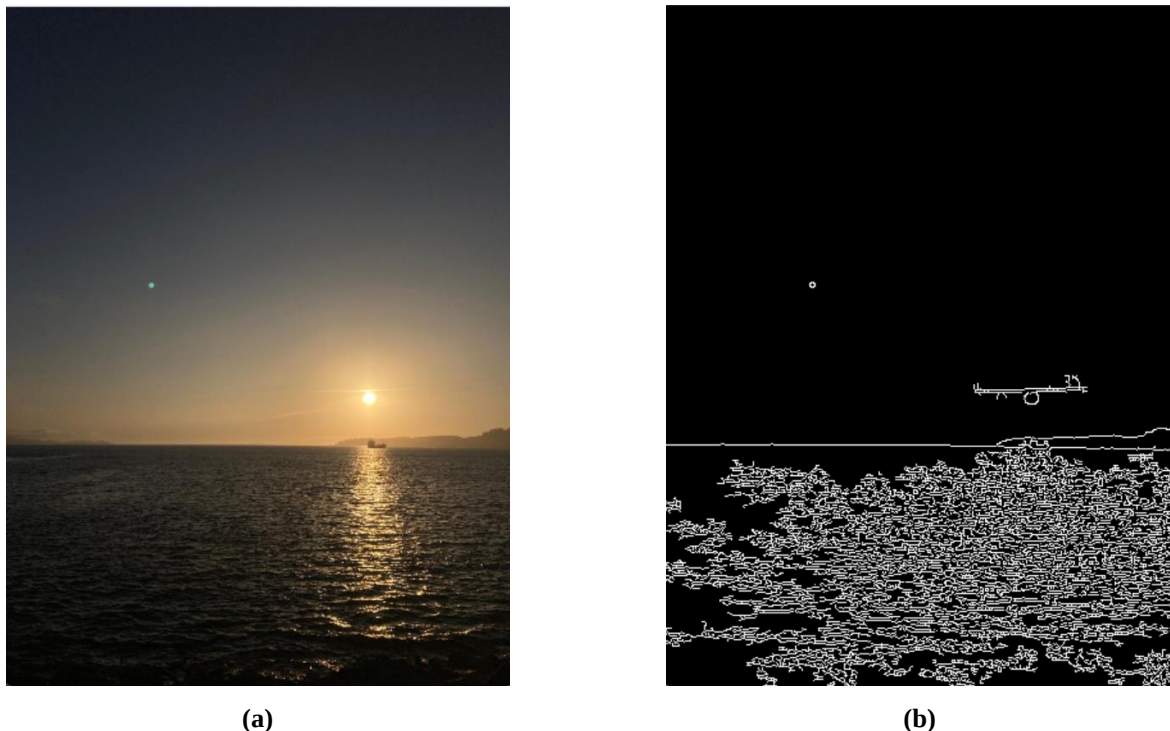


Figura 4-3 Ejemplo de detección de bordes próximo al ocaso (a) Imagen original (b) Bordes detectados con Canny

En la Figura 4-3, se aprecia una mejora notable en la detección del horizonte, atribuible a la ausencia de reflejos en la superficie del agua. Sin embargo, la posición baja del Sol cerca del ocaso introduce cierto ruido en la imagen debido a la baja iluminación y los reflejos generados. A pesar de estas dificultades, se observa una capacidad del algoritmo para adaptarse y detectar los contornos con relativa precisión.

Estos ejemplos visuales destacan la importancia de evaluar la precisión en la detección de bordes para comprender el desempeño del algoritmo en diferentes condiciones y escenarios. Además, subrayan la necesidad de considerar variables como la iluminación, la presencia de reflejos y la relevancia de los elementos presentes en la escena para mejorar la calidad de los resultados obtenidos.

4.1.3 Verificación de la Exactitud en la Detección de Líneas y Círculos

El reconocimiento del Sol y del horizonte es otro de los pilares del sistema de visión artificial, ya que establece la base para la posterior determinación de la altura del Sol. Encontramos dos procesos principales:

- i) **Reconocimiento del horizonte.** Inicialmente supuso un reto la determinación del método a utilizar y el filtrado de las líneas para detectar la línea del horizonte. Para ello, se definieron umbrales, tanto para la longitud, como para la pendiente de estas rectas. El reconocimiento del horizonte es generalmente preciso, se ha notado una ligera variabilidad en la precisión en algunas imágenes. Esta variabilidad se atribuye a condiciones ambientales como la iluminación irregular o la presencia de obstrucciones en el horizonte. A pesar de que estas situaciones pueden influir en la precisión del reconocimiento del horizonte, en la mayoría de los casos, el algoritmo logra identificar esta característica con una precisión aceptable.

- ii) **Reconocimiento del Sol.** Se ha observado que el algoritmo de detección del Sol logra resultados notablemente precisos en la mayoría de las imágenes procesadas. Se lleva a cabo de manera consistente y confiable, lo que indica una alta precisión en la identificación de este elemento clave en el proceso de determinación de la altura solar.

Es importante destacar que, a pesar de algunas variaciones en la precisión del reconocimiento del horizonte, el algoritmo general demuestra una capacidad satisfactoria para identificar tanto el Sol como el horizonte en una variedad de condiciones ambientales y situaciones de imagen.

4.1.4 Análisis de la Estabilidad bajo Variaciones.

Por un lado, la estabilidad del algoritmo bajo diversas condiciones ambientales es crucial para su aplicabilidad y su capacidad para producir resultados precisos de manera consistente. A continuación, se presenta un análisis detallado de la estabilidad del sistema bajo diferentes variaciones ambientales:

- i) **Variaciones en la Iluminación.** Se han realizado pruebas bajo condiciones de iluminación variable, desde días soleados hasta días nublados, con el fin de evaluar la robustez del algoritmo ante cambios en la luminosidad. Se ha observado que, en general, el algoritmo mantiene un rendimiento estable y puede identificar tanto el Sol como el horizonte de manera confiable. Sin embargo, se ha notado que la detección del Sol se ve afectada cuando está cerca del ocaso, lo cual se atribuye al cambio de color del cielo hacia tonos cercanos al del Sol.
- ii) **Presencia de Nubes.** Las imágenes tomadas en condiciones de nubosidad variable se utilizaron para evaluar la capacidad del algoritmo para discernir los contornos relevantes del Sol y del horizonte en entornos más desafiantes. Si bien la presencia de nubes puede tener un ligero impacto en la precisión del reconocimiento de los elementos, el algoritmo muestra una notable resistencia y sigue ofreciendo resultados satisfactorios en la mayoría de los casos. Es importante destacar que el algoritmo funciona correctamente cuando los cielos están parcialmente cubiertos y permiten la visibilidad de los elementos. Sin embargo, si los cielos están completamente cubiertos o los elementos no son visibles la determinación de la altura del Sol no es posible.
- iii) **Impacto de la Reflexión y el Brillo.** Durante las pruebas, la presencia de reflexiones y brillos en la superficie del agua y la lente de la cámara afecta a la precisión de la detección de los elementos. Aunque estas condiciones pueden introducir cierto nivel de ruido en las imágenes, el algoritmo ha demostrado su capacidad para mitigar estos efectos. Es importante resaltar que, en ciertas ocasiones, como cuando la diferencia de tonalidad entre el cielo y el agua del mar es mínima, la presencia de reflejos en el horizonte ayuda a su detección, lo que contribuye de forma positiva al proceso. Además, se ha observado que los reflejos generados en la lente son residuales y no interfieren de manera significativa en el proceso de detección.

Por otro lado, la presencia de elementos externos afecta a la detección del Sol y del horizonte. Durante las pruebas se han detectado los siguientes elementos que pueden inferir en la precisión del algoritmo:

- i) **Presencia de Tierra en el Fondo de la Imagen.** La presencia formaciones terrestres en el fondo de la imagen puede interferir en la detección del horizonte, ya que puede ser confundida con la línea del horizonte real. En las pruebas realizadas, la detección del horizonte no ha presentado problemas, pero si estas formaciones terrestres ocupan gran parte de la imagen, puede verse comprometido.
- ii) **Presencia de objetos en el área de interés.** Puede alterar los resultados de detección, afectando la precisión del algoritmo la presencia de barcos navegando o la presencia de obstáculos en la cercanías alrededor de la imagen.

4.2 Resultados

4.2.1 Comparación con la altura obtenida con el sextante

Los resultados de altura obtenidos a través del sistema de visión artificial se contrastan con las mediciones realizadas mediante el uso de un sextante.

Cuando las condiciones meteorológicas y la posición del Sol lo permitieron, se llevaron a cabo una serie de mediciones de la altura del Sol tanto con sextante como con el sistema de visión artificial en dirección hacia la boca de la Ría de Pontevedra. Cada medición se realizó simultáneamente para garantizar la coincidencia en el momento exacto, asegurando así la uniformidad en la altura registrada.

Dado que la altura obtenida mediante el sistema de visión artificial se corresponde con la altura instrumental del sextante, se considera la corrección de índice (desviación intrínseca en el instrumento), mientras que no se incluyen las correcciones por altura de observación (la altura es la misma para ambas mediciones) ni la corrección por semidiámetro del Sol (se tiene en cuenta en el cálculo de altura). Tampoco se corrigió por paralaje ni refracción, ya que afectan tanto a la cámara como al ojo humano de manera similar [2].

En las gráficas presentadas en la Figura 4-4 y la Figura 4-5 se muestran los datos recopilados en dos días distintos y a diferentes horas. Se observa que los errores no siguen un patrón sistemático y no son predecibles en relación con las mediciones realizadas. Estos errores pueden atribuirse a la precisión de la cámara y a la presencia de ruido, especialmente en la detección del Sol. En las imágenes, la luminosidad del Sol puede hacer que la circunferencia solar no sea tan definida como la mostrada por un sextante utilizando lentes.

Además, se ha notado que la variación en los errores es mayor a medida que aumenta la altura del Sol, lo que indica que no es posible aplicar un único factor de corrección para todas las situaciones.

Para el día 18 de marzo, el mayor desfase observado es de $0,27^\circ$ o 0,16 minutos de arco, mientras que para el día 24 de marzo, el mayor desfase es de $0,79^\circ$ o 47,4 minutos de arco. Estos errores pueden ser introducidos por diversos factores, como el ruido presente en la imagen, errores humanos ocasionales o condiciones variables del entorno.

Los errores observados en las mediciones no muestran un patrón sistemático y pueden ser influenciados por múltiples factores, lo que resalta la importancia de considerar la precisión y las condiciones ambientales al interpretar los resultados obtenidos.

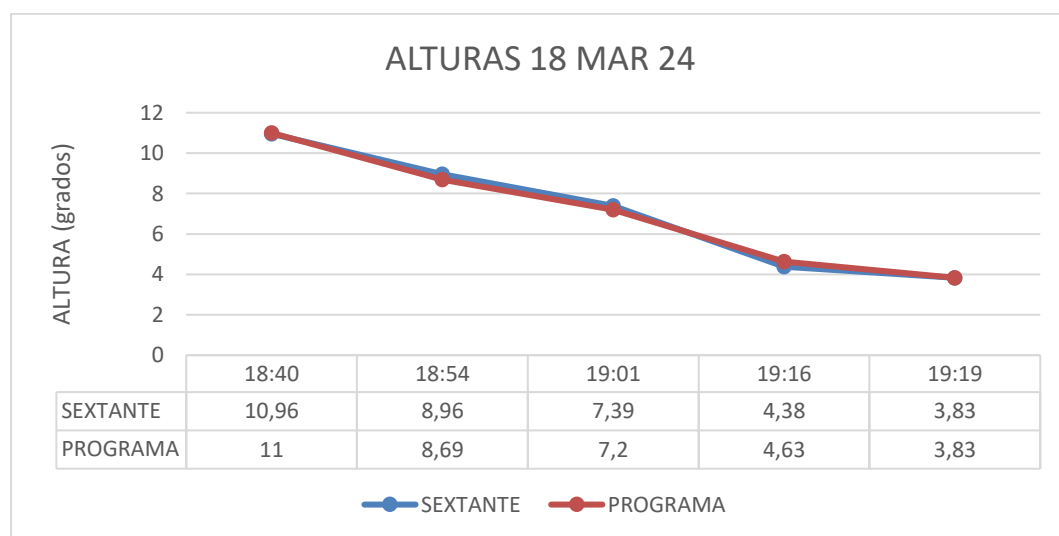


Figura 4-4 Gráfica de evolución de alturas en grados frente al tiempo del 18 de marzo de 2024

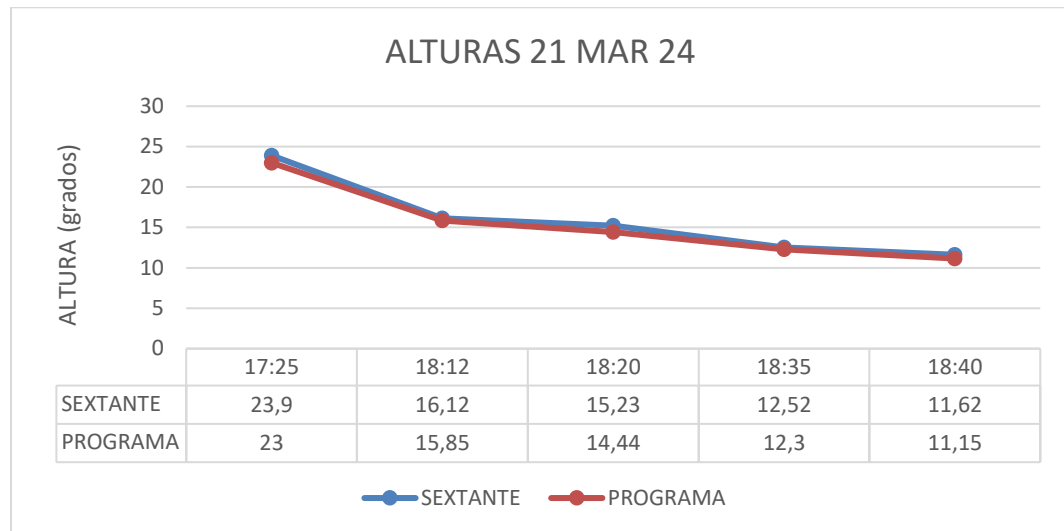


Figura 4-5 Gráfica de evolución de alturas en grados frente al tiempo del 21 de marzo de 2024

4.2.2 Interpretación de los Resultados

En la interpretación de los resultados, se identifica una tendencia de disminución en la altura del Sol a medida que transcurre el tiempo. Sin embargo, surge una discrepancia de minutos entre las mediciones. Este desajuste sugiere la falta de precisión y puede ser atribuido a diversos factores.

En primer lugar, el error humano es una consideración importante. A pesar de los esfuerzos por garantizar la uniformidad y precisión en las mediciones, siempre existe la posibilidad de error humano. La variabilidad en la ejecución de las mediciones podría contribuir al desfase observado, tanto en la toma de alturas con el sextante, como en las imágenes tomadas con el dispositivo móvil. Como error introducido por el observador encontramos los siguientes:

- **Procedimientos de medición.** Aunque se hayan establecido procedimientos claros, la ejecución exacta de las mediciones puede variar entre las mediciones. La forma en que se sostiene el sextante o se capturan las imágenes con el dispositivo móvil podría diferir ligeramente entre unas mediciones y otras, lo que podría introducir discrepancias en los datos.
- **Interpretación de los datos.** La interpretación de las lecturas obtenidas del sextante y de las imágenes capturadas puede ser subjetiva hasta cierto punto. En cada medición, existe variabilidad en la identificación del punto exacto del borde del Sol, lo que podría influir en las mediciones de altura.

En segundo lugar, la calidad de la imagen tomada con el dispositivo móvil puede influir significativamente en la precisión de las mediciones. Imperfecciones en la resolución, enfoque o iluminación podrían introducir errores en la estimación de la altura del Sol.

En tercer lugar, aunque se haya realizado una calibración meticulosa del sistema, siempre existe la posibilidad de que pequeños errores en la configuración inicial o durante la operación afecten los resultados. Por ejemplo, ajustes inadecuados en el dispositivo móvil y en el sextante podrían influir en la precisión de las mediciones.

5 CONCLUSIONES Y LÍNEAS FUTURAS

5.1 Conclusiones

Durante el desarrollo de este Trabajo de Fin de Grado, se ha avanzado considerablemente en la consecución de los objetivos establecidos.

En cuanto al objetivo principal, se ha obtenido un programa semiautomático que determina la altura del Sol. El programa requiere de un preprocesado de la imagen que consiste en la selección de una región de interés para evitar la introducción de ruido en la imagen, como elementos indeseables en la escena (tierra, embarcaciones, reflejos en la superficie del mar...). Es por ello por lo que se ha cumplido parcialmente el objetivo obteniendo un sistema económico y accesible mediante el uso de un dispositivo móvil que toma mediciones de forma robusta.

Por otro lado, se ha logrado un sólido progreso en la consecución de los objetivos secundarios, destacando:

- **Detección de bordes en las imágenes.** Se exploró y se evaluó en algoritmo *Canny*, para la detección de bordes en imágenes con alta iluminación. Los resultados obtenidos mostraron una efectividad notable en la detección de los bordes haciendo uso de los umbrales apropiados para la detección de objetos muy brillantes, como es el Sol, y para la detección de los bordes menos marcados, como es el caso de la línea del horizonte que separa el mar del cielo.
- **Localización del Sol y del horizonte en imágenes.** Se implementó con éxito la Transformada de *Hough* para la localización precisa del Sol y del horizonte en las imágenes capturadas. Esta fase del proceso demostró una alta fiabilidad en la identificación de estos elementos. Inicialmente la detección del horizonte supuso un reto ya que la dificultad de diferir entre el cielo y el mar no presentaba la suficiente variación para detectarlo, pero finalmente se ha obtenido con bastante robustez ajustando los parámetros de la transformada.
- **Calibración del sistema.** Se llevó a cabo un estudio experimental para calibrar las lentes utilizadas en la captura de imágenes, comparando los resultados con instrumentos especializados en la medición de la altura solar. Los resultados respaldaron la precisión del sistema de visión artificial desarrollado.

Es importante destacar que, si bien se han alcanzado resultados satisfactorios, existen áreas identificadas para futuras mejoras. La optimización del programa para lograr una mayor autonomía y precisión en la determinación de la altura solar será un objetivo prioritario para trabajos futuros.

5.2 Líneas futuras

En este apartado se exponen las siguientes mejoras con el fin de evolucionar hacia un sistema de visión artificial más preciso y automatizado, capaz de ser implementado como herramienta principal o de apoyo a otros sistemas sin la necesidad de operadores.

1. Desarrollo continuo del sistema para automatización total

Se propone continuar el desarrollo del sistema con el objetivo de automatizar por completo el proceso. Esto implica la implementación de una cámara en constante grabación para permitir cálculos en tiempo real. Esta mejora facilitaría la obtención continua y sin interrupciones de medidas de altura.

2. Utilización de lentes y filtros especializados

El uso de lentes y filtros especializados para la visión solar, similares a los empleados en los sextantes, puede ser de gran ayuda para el sistema. Estos dispositivos mejoran la precisión al proporcionar una mayor definición de la forma circular del Sol al evitar la aparición de reflejos no deseados.

3. Incorporación de cámaras infrarrojas

La integración de cámaras termográficas permitiría al sistema operar incluso en situaciones donde la visión del Sol esté obstaculizada por la nubosidad. Esta mejora aumenta la versatilidad y la fiabilidad del sistema en diversas condiciones climáticas.

4. Implementación de *Machine Learning*

La creación de un conjunto de datos de imágenes etiquetadas con los elementos a detectar podría facilitar la detección automatizada en las imágenes procesadas por el sistema. La aplicación de estas técnicas mejoraría la precisión y la eficiencia del sistema, reduciendo así la necesidad de intervención humana.

5. Uso de sensores

El aprovechamiento de acelerómetros y giróscopos presentes en el dispositivo móvil sería de gran ayuda para determinar ciertos parámetros, como la posición o la inclinación del dispositivo, aumentando la precisión en la toma de medidas.

6. Integración con otros sistemas

La integración de un sistema de visión artificial con otros sistemas, como los mencionados en el apartado 1.3, capaces de procesar los datos recopilados posibilita la automatización del proceso. Esta unión de sistemas permite una transferencia fluida de información, lo que conlleva a una operación eficiente y precisa.

6 BIBLIOGRAFÍA

- [1] Armada - Ministerio de Defensa - Gobierno de España. Accedido: 8 de febrero de 2024. [En línea]. Disponible en: <https://armada.defensa.gob.es/ArmadaPortal/page/Portal/ArmadaEspannola/iniciohome/prefLang-es/>
- [2] B. Barragáns Martínez (advisor), N. Fernández García (advisor), y S. González-Aller Rodríguez, «Sistema automático de geolocalización basado en los astros: módulo de adquisición de datos», jun. 2015, Accedido: 19 de marzo de 2024. [En línea]. Disponible en: <http://calderon.cud.uvigo.es/handle/123456789/154>
- [3] P. de Marcos González, J. González Coma (advisor), y J. I. Valles Cancela (advisor), «Sistema de posicionamiento solar para buques», abr. 2022, Accedido: 19 de marzo de 2024. [En línea]. Disponible en: <http://calderon.cud.uvigo.es/handle/123456789/631>
- [4] Luis Enrique Sucar Succar, «Visión: Natural y Artificial», octubre de 2008.
- [5] What is Computer Vision? | IBM. Accedido: 25 de enero de 2024. [En línea]. Disponible en: <https://www.ibm.com/es-es/topics/computer-vision>
- [6] «A3 Vision & Imaging | Association for Advancing Automation». Accedido: 5 de marzo de 2024. [En línea]. Disponible en: <https://www.automate.org/vision>
- [7] R.A Aquino Castro, «Reconocimiento E Interpretación Del Alfabeto Dactilológico De La Lengua De Señas Mediante Tecnología Móvil Y Redes Neuronales Artificiales.» Universidad Mayor de San Andrés.
- [8] «Sistemas de Visión Artificial, Historia, Componentes y Procesamiento de Imágenes.» Accedido: 25 de enero de 2024. [En línea]. Disponible en: <https://1library.co/document/y9r89vry-sistemas-de-vision-artificial-historia-componentes-y-procesamiento-de-imagenes.html>
- [9] R. C. Gonzalez y R. E. Woods, «Digital image processing.» New York, NY: Pearson, 2018.
- [10] I. G. S y V. C. S, «La visión artificial y los campos de aplicación», Tierra Infin., vol. 1, n.º 1, Art. n.º 1, dic. 2015.
- [11] «Procesador de visión | Cognex». Accedido: 22 de febrero de 2024. [En línea]. Disponible en: <https://www.cognex.com/es-es/what-is/machine-vision/components/vision-processing>
- [12] «Curso de Procesado de Imagen GPI». Accedido: 21 de febrero de 2024. [En línea]. Disponible en: https://www.uv.es/gpoei/eng/Pfc_web/generalidades/como.htm
- [13] J. D. Foley y A. V. Dam, «Fundamentals of Interactive Computer Graphics.» Addison-Wesley Publishing Company, 1982.

- [14] «Modelo de color RGB | Guías Prácticas». Accedido: 27 de febrero de 2024. [En línea]. Disponible en: <https://www.guiaspracticas.com/impresoras/modelo-de-color-rgb>
- [15] M. O. Al-Dwairi, Z. A. Alqadi, A. A. AbuJazar, y R. A. Zneit, «Optimized True-Color Image Processing», 2010.
- [16] «OpenCV: Canny Edge Detection». Accedido: 14 de febrero de 2024. [En línea]. Disponible en: https://docs.opencv.org/3.4/da/d22/tutorial_py_canny.html
- [17] J. Tejedor Gómez, «Análisis comparativo de algoritmos en segmentación de iris», jun. 2009, Accedido: 7 de marzo de 2024. [En línea]. Disponible en: <http://hdl.handle.net/10016/7601>
- [18] J. Valverde-Rebaza, «Detección de bordes mediante el algoritmo de Canny», oct. 2007.
- [19] F. Escolano Ruiz, M. Cazorla Quevedo, M. I. Alonso Galipienso, O. Colomina Pardo, y M. Á. Lozano Ortega, «Inteligencia artificial: Modelos, técnicas y áreas de aplicación.» Thomson Reuters, 2003. Accedido: 7 de marzo de 2024. [En línea]. Disponible en: <https://dialnet.unirioja.es/servlet/libro?codigo=655446>
- [20] L. G. Shapiro y G. C. Stockman, «Computer Vision.» Prentice Hall, 2001.
- [21] «Welcome to Python.org», Python.org. Accedido: 5 de febrero de 2024. [En línea]. Disponible en: <https://www.python.org/>
- [22] «Python para todos». Accedido: 5 de febrero de 2024. [En línea]. Disponible en: https://repositorio.uci.cu/bitstream/123456789/10206/1/Python_para_todos.pdf
- [23] «El tutorial de Python», Python documentation. Accedido: 5 de febrero de 2024. [En línea]. Disponible en: <https://docs.python.org/3/tutorial/index.html>
- [24] «PyCharm: el IDE de Python para desarrolladores profesionales, por JetBrains». Accedido: 5 de febrero de 2024. [En línea]. Disponible en: <https://www.jetbrains.com/es-es/pycharm/>
- [25] «Welcome to The Apache Software Foundation!» Accedido: 5 de febrero de 2024. [En línea]. Disponible en: <https://www.apache.org/>
- [26] «Home», OpenCV. Accedido: 5 de febrero de 2024. [En línea]. Disponible en: <https://opencv.org/>
- [27] «Modern Computer Vision Tools: An Overview by Svitla Systems». Accedido: 5 de febrero de 2024. [En línea]. Disponible en: <https://svitla.com/blog/overview-of-modern-computer-vision-tools>
- [28] «NumPy documentation — NumPy v1.26 Manual». Accedido: 27 de febrero de 2024. [En línea]. Disponible en: <https://numpy.org/doc/stable/index.html>
- [29] «tkinter — Interface de Python para Tcl/Tk», Python documentation. Accedido: 18 de marzo de 2024. [En línea]. Disponible en: <https://docs.python.org/3/library/tkinter.html>
- [30] «Python | Grayscale of Images using OpenCV - GeeksforGeeks». Accedido: 18 de marzo de 2024. [En línea]. Disponible en: <https://www.geeksforgeeks.org/python-grayscale-of-images-using-opencv/>
- [31] J. Igual, «Todo sobre el Histograma en Fotografía», foto igual. Accedido: 18 de marzo de 2024. [En línea]. Disponible en: <https://fotoigual.com/tutoriales/todo-sobre-el-histograma-en-fotografia/>
- [32] V. Campa, *Análisis de Imágenes de Microscopía con ImageJ*. 2017.
- [33] DavidMuYe, «DavidMuYe/Sextante-Vision-Artificial». 3 de abril de 2024. Accedido: 3 de abril de 2024. [En línea]. Disponible en: <https://github.com/DavidMuYe/Sextante-Vision-Artificial>

ANEXO I: IMPLICACIONES SOCIALES, ECONÓMICAS Y AMBIENTALES

La implementación de sistemas de determinación de la altura del Sol mediante visión artificial tiene un impacto en varios aspectos sociales, económicos y ambientales:

1. Accesibilidad universal

Al democratizar el acceso a herramientas precisas y confiables para la determinación de la altura solar, esta tecnología simplifica un proceso tradicionalmente costoso y complejo, haciéndolo accesible para cualquier persona con tecnología básica. Esto no solo facilita la adopción generalizada de sistemas de medición solar, sino que también promueve la inclusión en el acceso a métodos eficaces de determinación de la altura del Sol.

2. Compromiso social

La disponibilidad de herramientas de determinación de la altura del Sol refleja un compromiso con la sociedad al acercar un cálculo complejo a un nivel accesible para el público en general. Esto facilita la adopción de métodos de medición solar por parte de una variedad de sectores, y promueve una mayor conciencia sobre el potencial de la determinación de la altura solar en diversas áreas, desde la agricultura hasta la navegación.

3. Bajo coste

Esta tecnología es de bajo coste y accesible para cualquier persona que posea un dispositivo móvil, ampliando su alcance e impacto. No se requiere una infraestructura costosa o especializada para su implementación, lo que facilita su adopción generalizada e integración en la vida cotidiana.

4. Reducción de materiales y residuos

Al utilizar un dispositivo móvil común para medir la altura del Sol a través de algoritmos de visión artificial, se elimina la necesidad de dispositivos especializados y materiales costosos. Esto reduce la demanda de recursos naturales y la generación de residuos asociados con la producción, el mantenimiento y la eliminación de equipos tradicionales de medición solar.

ANEXO II: REFLEXIONES ÉTICAS Y SOCIALES

En este anexo, se profundizará en las reflexiones éticas y sociales asociadas con la implementación de sistemas de determinación de la altura del Sol mediante técnicas de visión artificial.

1. Equidad y accesibilidad

Si bien la tecnología puede facilitar el acceso a herramientas de determinación de la altura del Sol, es crucial considerar la accesibilidad. Se deben abordar posibles disparidades en el acceso a la tecnología, asegurando que todos los sectores de la sociedad tengan la oportunidad de beneficiarse de estas herramientas. Además, se deben tener en cuenta las necesidades de personas con discapacidades para garantizar la accesibilidad universal.

2. Impacto en el empleo y la economía

La adopción de sistemas de visión artificial puede tener implicaciones en el mercado laboral y la economía. Es importante evaluar cómo estas tecnologías pueden afectar a los trabajadores que realizan tareas relacionadas con la medición solar, así como identificar oportunidades para la creación de empleo en sectores emergentes asociados con la tecnología.

3. Consideraciones éticas en la aplicación práctica

Al implementar sistemas de determinación de la altura del Sol, se deben tener en cuenta consideraciones éticas, como la transparencia en la recopilación y procesamiento de datos, y la responsabilidad en la toma de decisiones basadas en los resultados obtenidos.

4. Educación y concienciación

Para maximizar los beneficios sociales de la tecnología de visión artificial, es fundamental promover la educación y la concienciación sobre su uso adecuado y sus implicaciones éticas. Esto incluye la capacitación de usuarios en el manejo responsable de los sistemas de determinación de la altura del Sol y la promoción de un debate público informado sobre los aspectos éticos y sociales asociados con estas tecnologías.