

UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE

ESCUELA POLITÉCNICA SUPERIOR DE ELCHE

GRADO EN INGENIERÍA INFORMÁTICA EN TECNOLOGÍAS
DE LA INFORMACIÓN



UNIVERSITAS
Miguel Hernández



Biblioteca
UNIVERSITAS Miguel Hernández

"PUESTA EN MARCHA Y PROGRAMACIÓN
DE CÁMARAS DE POLARIZACIÓN
MONOCROMA Y EN COLOR"

TRABAJO FIN DE GRADO

Junio - 2026

AUTOR: Iván Ruiz Boix

DIRECTORES: María del Mar Sánchez López

Ignacio Moreno Soriano

AGRADECIMIENTOS

Tras la finalización de este trabajo, me gustaría dedicar unas palabras de agradecimiento a todas aquellas personas que me dieron apoyo e hicieron posible la realización de este mismo proyecto, ya sea de forma directa o indirecta.

En primer lugar, agradecer a mis tutores María del Mar Sánchez López e Ignacio Moreno Soriano así como a todo el grupo TecnOpto de la UMH por darme la oportunidad de realizar el trabajo que aquí expongo. Sin su ayuda no dispondría de los componentes ni los conocimientos físicos necesarios para el desarrollo del proyecto, por lo que ha sido gracias a su guía y apoyo por los que este ha sido posible de realizar.

También quiero agradecer a todo el cuerpo docente de la UMH del grado en ingeniería informática en tecnologías de la información por todas las clases impartidas, las cuales fueron sin duda uno de los aspectos clave de mi formación.

Además, agradecer a todos mis compañeros de clase por todos los buenos momentos que pasamos juntos. Desde horas de estudio en conjunto y proyectos grupales a pequeñas charlas y fotos navideñas.

Por último, pero no por ello menos importante, agradecer a mi familia por haber estado siempre apoyándome y aconsejándome en momentos de duda.

RESUMEN

En este proyecto se han desarrollado dos softwares para el control de dos cámaras de polarización diferentes, una monocromática y otra a color RGB, para el posterior análisis en tiempo real de las imágenes capturadas. También se describen los complementos desarrollados para mejorar la calidad de vida de estos, como los programas de calibración. En este documento se detalla toda la información relevante referente a la realización y finalización del trabajo.

En el primer capítulo, “Introducción”, se detallan todos los conceptos y características clave de las áreas y métodos que engloban al proyecto en sí, además de describir los objetivos del mismo.

En el segundo capítulo, “Materiales y Métodos”, se contextualiza el proyecto detallando el uso y características actuales de las cámaras de polarización, además de presentar los componentes utilizados.

En el siguiente capítulo, “Requerimientos y Diseño”, se detalla el proceso de ingeniería del software llevado a cabo para la creación de los diferentes programas.

A continuación, en el capítulo de “Implementación”, se detallan las prestaciones del proyecto en su etapa final, mostrando características, funcionalidades y formas de uso.

Para concluir el documento, en el capítulo de “Conclusiones y Trabajo Futuro”, se expone una breve reflexión de los logros alcanzados tras la finalización del proyecto junto a posibles áreas a mejorar más adelante.

INDICE

1. INTRODUCCIÓN	7
1.1 LUZ POLARIZADA.....	7
1.2 CÁMARAS DE POLARIZACIÓN	9
1.3 POLARIMETRÍA DE IMÁGENES.....	10
1.4 RETARDADORES LCR.....	12
1.5 OBJETIVOS.....	13
2. MATERIALES Y MÉTODOS.....	14
2.1 ESTADO DEL ARTE DE LAS CÁMARAS DE POLARIZACIÓN.....	14
2.2 CÁMARA MONOCROMÁTICA “ <i>THORLABS KIRALUX CS505MUP</i> ”	16
2.2.1 ESPECIFICACIONES DEL SENSOR	17
2.2.2 ESPECIFICACIONES PARA TOMA DE IMÁGENES	19
2.2.3 COMPONENTES Y ACCESORIOS.....	20
2.3 CÁMARA RGB “ <i>IMAGINGSOURCE DYK 33UX250</i> ”	22
2.3.1 ESPECIFICACIONES DEL SENSOR	23
2.3.2 ESPECIFICACIONES PARA LA TOMA DE IMÁGENES	24
2.3.3 COMPONENTES Y ACCESORIOS.....	25
2.4 RETARDADOR LCR Y CONTROLADOR “ <i>THORLABS KLC101</i> ”	26
2.4.1 ESPECIFICACIONES Y CARACTERÍSTICAS	27
2.5 SELECCIÓN DE HERRAMIENTAS DE DESARROLLO	29
3. REQUERIMIENTOS Y DISEÑO	31
3.1 SOFTWARE PARA CÁMARA MONOCROMÁTICA DE POLARIZACIÓN	31
3.1.1 REQUISITOS	31
3.1.2 DIAGRAMA DE CASOS DE USO.....	31
3.1.3 DIAGRAMA DE GANTT	32
3.1.4 DISEÑO	33
3.2 SOFTWARE PARA CALIBRACIÓN DEL LCR	35

3.2.1 REQUISITOS	35
3.2.2 DIAGRAMA DE CASOS DE USO.....	35
3.2.3 DIAGRAMA DE GANTT	36
3.2.4 DISEÑO	36
3.3 SOFTWARE PARA CALIBRACIÓN CON MONITOR.....	37
3.3.1 REQUISITOS	37
3.3.2 DIAGRAMA DE CASOS DE USO.....	37
3.3.3 DIAGRAMA DE GANTT	38
3.3.4 DISEÑO	38
3.4 SOFTWARE PARA CÁMARA RGB DE POLARIZACIÓN.....	39
3.4.1 REQUISITOS	39
3.4.2 DIAGRAMA DE CASOS DE USO.....	39
3.4.3 DIAGRAMA DE GANTT	40
3.4.4 DISEÑO	40
4. IMPLEMENTACIÓN.....	41
4.1. SOFTWARE PARA CÁMARA MONOCROMÁTICA DE POLARIZACIÓN	41
4.1.1 FUNCIONALIDAD BÁSICA	41
4.1.2 COMBINACIÓN CON LCR.....	48
4.1.3 COMBINACIÓN CON MONITOR.....	52
4.1.4 COMBINACIÓN CON BOTÓN FÍSICO	55
4.2 SOFTWARE PARA CALIBRACIÓN DEL LCR	57
4.3 SOFTWARE PARA CALIBRACIÓN CON MONITOR.....	63
4.4 SOFTWARE PARA CÁMARA RGB DE POLARIZACIÓN.....	67
4.4.1 FUNCIONALIDAD BÁSICA	67
4.4.2 COMBINACIÓN CON MONITOR.....	71
5. CONCLUSIONES Y TRABAJO FUTURO.....	73
5.1 CONCLUSIONES.....	73

5.2 TRABAJO FUTURO	74
6. BIBLIOGRAFÍA	75
ANEXO 1: DEFINICIÓN DE CASOS DE USO	77
ANEXO 1.1: SOFTWARE PARA CÁMARA MONOCROMÁTICA DE POLARIZACIÓN	77
ANEXO 1.2: SOFTWARE PARA CALIBRACIÓN DEL LCR	81
ANEXO 1.3: SOFTWARE PARA CALIBRACIÓN CON MONITOR.....	85
ANEXO 1.4: SOFTWARE PARA CÁMARA RGB DE POLARIZACIÓN.....	88



1. INTRODUCCIÓN

1.1 LUZ POLARIZADA

La luz es una onda electromagnética transversal formada por un campo eléctrico y un campo magnético que oscilan de forma perpendicular entre ellos y al eje de propagación. Los diferentes estados de polarización de la luz están definidos según la dirección del campo eléctrico que la compone.

La luz se considera polarizada linealmente cuando el campo eléctrico oscila dentro de un plano específico. Dependiendo de su orientación se distinguen los estados de polarización vertical (90°), horizontal (0°), diagonal (45°) y antidiagonal (135°). También existen dos estados de polarización circulares, que se forman cuando el campo eléctrico se propaga describiendo un círculo en el plano transversal. Si el círculo barre en sentido horario se obtiene el estado de polarización circular a derechas (CD), mientras que si lo hace en sentido antihorario se obtiene el estado de polarización circular a izquierdas (CI) [Figura 1.1].

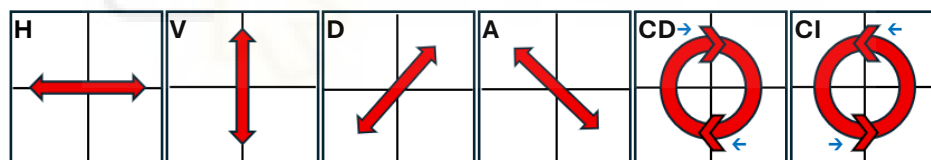


Figura 1.1: Dirección de los estados de polarización lineal Horizontal (H), Vertical (V), Diagonal (D), Antidiagonal (A), Circular a Derechas (CD) y Circular a Izquierdas (CI)

El estado de polarización de la luz también se puede representar de forma matemática usando el vector de Stokes. Este es un vector columna formado por cuatro componentes cuyos valores se pueden obtener midiendo la intensidad de la luz filtrada por polarizadores [Ecuación 1.1]. Los parámetros que forman el vector de Stokes son:

- S_0 : Representa la intensidad total de la luz.
- S_1 : Representa la diferencia de intensidad entre las componentes horizontal y vertical.

- S_2 : Representa la diferencia de intensidad entre las componentes diagonal y antidiagonal.
- S_3 : Representa la diferencia de intensidad entre las componentes circular a derechas y circular a izquierdas.

$$S = \begin{bmatrix} S_0 \\ S_1 \\ S_2 \\ S_3 \end{bmatrix} = \begin{bmatrix} A + D \\ H - V \\ D - A \\ CD - CI \end{bmatrix} \quad (1.1)$$

Cuando nos encontramos ante luz totalmente polarizada, el parámetro de Stokes S_3 también se puede obtener midiendo solo las componentes de polarización lineales mediante los parámetros S_1 y S_2 **[Ecuación 1.2]**.

$$S_3 = \sqrt{S_0^2 - S_1^2 - S_2^2} \quad (1.2)$$

Sin embargo, en general la luz no está polarizada completamente, sino que se encuentra de forma parcial. En ese caso podemos usar el grado de polarización (DoP) para indicar la fracción de la intensidad que se encuentra totalmente polarizada. **[Ecuación 1.3]**. Si su valor es igual a uno indica que la luz está completamente polarizada, si es igual a cero indica que es no polarizada y si es un valor intermedio indica luz parcialmente polarizada.

$$DoP = \frac{\sqrt{S_1^2 + S_2^2 + S_3^2}}{S_0} \quad (1.3)$$

En ocasiones, en especial cuando solamente se tienen filtros de polarizador lineales como ocurre en las cámaras de polarización, se utiliza también el grado de polarización lineal **[Ecuación 1.4]**.

$$DoPL = \frac{\sqrt{S_1^2 + S_2^2}}{S_0} \quad (1.4)$$

1.2 CÁMARAS DE POLARIZACIÓN

Tanto el ojo humano como las cámaras comunes son capaces de detectar la intensidad y color de la luz del espectro visible, dando lugar a una imagen. Sin embargo, no son sensibles a la polarización de la luz. Las llamadas cámaras de polarización cuentan con un array de polarizadores integrado en el sensor que filtran las componentes de la luz que oscilan en la dirección de un eje de transmisión. Esto permite generar imágenes capaces de mostrar aspectos que se presentarían indetectables ante la luz natural (no polarizada) [Figura 1.2].

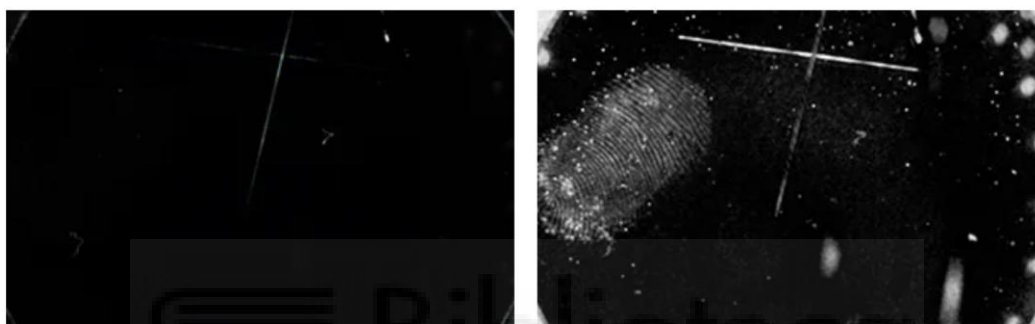


Figura 1.2: Comparación entre una imagen normal (izquierda) y una polarizada (derecha) de la [web de Sony](#)

Para poder usar este tipo de cámaras de manera eficiente debemos tener en cuenta algunas de las características que las definen. Una de la más importantes en todos los sensores de las cámaras es la eficiencia cuántica (QE), que mide la probabilidad de que un fotón sea detectado por el sensor de la cámara. Como no existe ningún material que sea completamente transparente ante todas las longitudes de onda, la eficiencia de estas cámaras se suele concentrar en un rango determinado, decayendo a medida que nos alejamos de este. En la práctica, se debe elegir una cámara con buena eficiencia cuántica para las longitudes de onda con las que formaremos las imágenes.

Otra característica, específica de las cámaras de polarización, es la ratio de extinción (ER), que indica la calidad de los polarizadores utilizados. Este se expresa como el cociente $T_{\max} / T_{\min}$, dónde $T_{\max}$ es la máxima transmisión del polarizador, idealmente del 100%, cuando coincide con la luz incidente polarizada en la dirección del polarizador. $T_{\min}$ es la mínima transmisión, idealmente igual a cero, correspondiente a la luz incidente polarizada perpendicular al polarizador.

1.3 POLARIMETRÍA DE IMÁGENES

La polarimetría de imágenes es una técnica para captar detalles invisibles a simple vista en imágenes usando técnicas de polarización. Este tipo de imágenes se pueden obtener usando tanto cámaras de polarización como cámaras comunes con ayuda de filtros polarizadores externos.

Si decidimos utilizar una cámara de polarización, debemos tener en cuenta que para poder obtener una imagen coherente deberemos de procesar la información del archivo RAW, comúnmente usando un software. El array de polarizadores del sensor no capta la información de los píxeles de forma ordenada, en su lugar se presentan en forma de mosaico, intercalando la información de los cuatro filtros de polarización lineales (horizontal, diagonal, antidiagonal y vertical) en macropíxeles de 2x2 píxeles que se repiten a lo largo del sensor **[Figura 1.4]**.

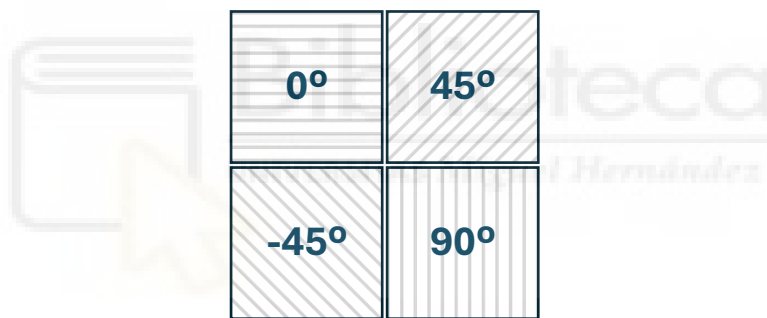


Figura 1.4: Ejemplo de mosaico de distribución de los polarizadores de una cámara

Si la cámara además captura imágenes a color, se suele utilizar un filtro adicional con estructura de mosaico para guardar esta información de forma más eficiente. Un filtro de este tipo es el de Bayer, donde cada píxel guarda información de una única componente de color RGB, distribuido en un patrón repetitivo de 2x2 (un píxel rojo, uno azul y dos verdes) **[Figura 1.5]**. Para obtener los colores reales de la imagen hay que realizar un proceso de “*demosaicing*”, que en el caso del filtro de Bayer consiste en obtener el valor de los componentes RGB ausentes en el píxel calculando la media de sus vecinos.

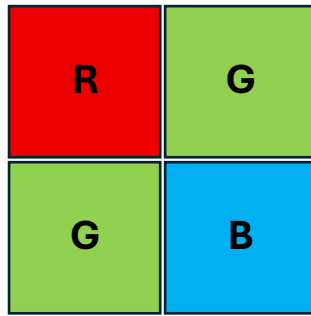


Figura 1.5: Ejemplo de patrón del filtro de Bayer RGGB (Red, Green, Green, Blue)

En este trabajo se han programado dos métodos para el proceso de “*demosaicing*”, uno para la cámara de polarización monocromática y otro para la cámara de polarización RGB.



1.4 RETARDADORES LCR

Los retardadores LCR (Liquid Crystal Retarder) son dispositivos ópticos que utilizan una celda de cristal líquido para modificar la fase o polarización de la luz que los atraviesa dependiendo de la orientación de sus moléculas.

Dos caras internas paralelas de la celda están recubiertas con materiales conductores para permitir aplicar un voltaje al dispositivo. Cuando se aplica una corriente alterna las moléculas del cristal líquido se reorientan, cambiando su alineación predeterminada según el nivel del voltaje cuadrático medio aplicado [Figura 1.3]. Esta característica permite modificar con precisión el estado de polarización de la luz al tener control sobre el índice de refracción extraordinario del cristal líquido.



Figura 1.3: Esquema sobre el principio de funcionamiento de los LCR de la [web de Thorlabs](#)

Dado que las cámaras de polarización solamente contienen filtros polarizadores lineales, estos dispositivos LCR son necesarios en polarimetría de imagen para poder medir las componentes circulares de la luz y así obtener la información completa del vector de Stokes, en particular el parámetro S_3 y el grado de polarización.

1.5 OBJETIVOS

El objetivo principal del proyecto es la creación de softwares que permitan la obtención y análisis de imágenes de polarización en tiempo real centralizando el manejo del equipamiento de laboratorio necesario, como cámaras de polarización o retardadores de cristal líquido. Durante la evolución del proyecto se han desarrollado cuatro programas diferentes, cada uno con una finalidad concreta.

El software desarrollado para la cámara de polarización monocromática tiene como objetivo el análisis de imágenes de polarización en blanco y negro obteniendo los parámetros de Stokes. Como complementos a este se ha desarrollado un software independiente para ayudar a calibrar el LCR (necesario para la correcta obtención del parámetro S_3) y otro para facilitar la calibración del brillo de la cámara usando un monitor.

De la misma forma, también se ha creado otro programa para poder obtener y analizar las imágenes de polarización a color RGB. En este caso se ha usado una cámara de polarización a color, y ha sido necesario realizar un doble proceso de “*demosaijing*”, uno para los filtros polarizados y otro para los filtros RGB.

Todo el proyecto ha sido concebido con uso en laboratorios en mente, por lo que se le ha dado más énfasis a la funcionalidad que a pulir el diseño para ser “atractivo”, ya que este no va a ser distribuido al público general.

2. MATERIALES Y MÉTODOS

2.1 ESTADO DEL ARTE DE LAS CÁMARAS DE POLARIZACIÓN

Los filtros de polarización tienen utilidades diversas en fotografía y, a nivel científico, en una amplia gama de áreas, debido a sus propiedades de filtrado de la luz. Un uso común es la reducción de reflejos. Cuando estos se generan en superficies no metálicas la luz reflejada queda polarizada de modo que puede ser eliminada utilizando un filtro polarizador si se orienta adecuadamente, eliminando el efecto deslumbrante. Ejemplos de aplicación de esta propiedad se pueden ver en cámaras de seguridad, por ejemplo para evitar reflejos en los parabrisas de los coches que impidan ver a los conductores, o en láminas “antirreflectantes” para pantallas de ordenador.

Actualmente las imágenes obtenidas con filtros polarizadores también son muy usadas en el ámbito de la fotoelasticidad y el análisis de materiales. Muchos materiales presentan esta propiedad de fotoelasticidad, que hace visible defectos o deformaciones al ser iluminados con luz polarizada y visualizados con un filtro polarizador. Muchos tejidos también presentan esta propiedad, por lo que en los últimos años estas imágenes están encontrando mucho uso en ámbitos biomédicos. Sin embargo, los sensores de las cámaras corrientes no son capaces de registrar este fenómeno, ya que no son sensibles a la polarización.

Tradicionalmente para las aplicaciones industriales de estos procesos se necesitaban realizar cuatro tomas diferentes utilizando un filtro de polarización situado delante de la cámara en distintas orientaciones para conseguir los resultados deseados. Hoy en día se ha podido simplificar este proceso a una única captura gracias al desarrollo de la tecnología “*Polarsens*” de Sony en 2018.

“*Polarsens*” es una tecnología a nivel de píxel de imágenes formadas con sensores CMOS que integra un array de polarizadores con ángulos distintos en el chip durante el proceso de fabricación. Estos polarizadores se encuentran entre la lente

y el fotodiodo, permitiendo así obtener datos sobre las cuatro componentes de polarización lineales (vertical, horizontal, diagonal y antidiagonal) al mismo tiempo.

Actualmente existen múltiples empresas que desarrollan cámaras con estas características con fines comerciales. Ejemplos de algunas de ellas pueden ser *Thorlabs*, *The Imaging Source* o *IDS*. A continuación se describen las dos cámaras de polarización usadas en este trabajo.



2.2 CÁMARA MONOCROMÁTICA “*THORLABS KIRALUX*”

CS505MUP”

La primera cámara de polarización utilizada es una *Thorlabs Kiralux* modelo *CS505MUP* [Figura 2.1]. Esta forma parte de la serie CS de la colección de cámaras científicas compactas digitales de *Thorlabs*. Sin embargo, el 16 de diciembre de 2021 fue descontinuada y reemplazada por su versión superior *CS505MUP1*.

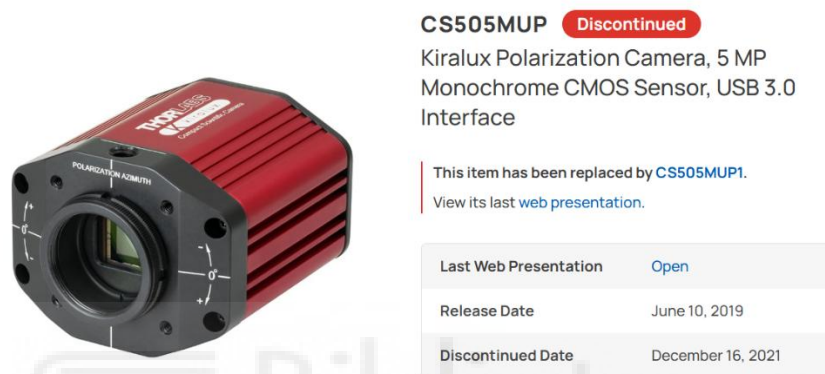


Figura 2.1: Página del producto de la [web oficial de Thorlabs](#)

Las cámaras de la serie *Kiralux* utilizan sensores CMOS y siguen el esquema de nomenclatura CSxxy{M/C}U{P}, donde: “xxy” es el número de píxeles, siendo “xx” el multiplicador, “y” el exponente de 10, “M/C” especifica si las imágenes son monocromáticas (M) o a color (C), “U” indica el uso de una interfaz USB 3.0 y “P” indica que la cámara es sensible a la polarización.

Por ende, el nombre del modelo usado indica que tiene una resolución de 5 Megapíxeles ($505 = 50 * 10^5$), captura imágenes de en blanco y negro (M), hace uso de los puertos USB 3.0 (U) y es capaz de capturar información de los diferentes grados de polarización (P).

Las dimensiones del dispositivo son 47.6 mm de alto, 70.4 mm de largo y 30.3 mm de ancho [Figura 2.2].

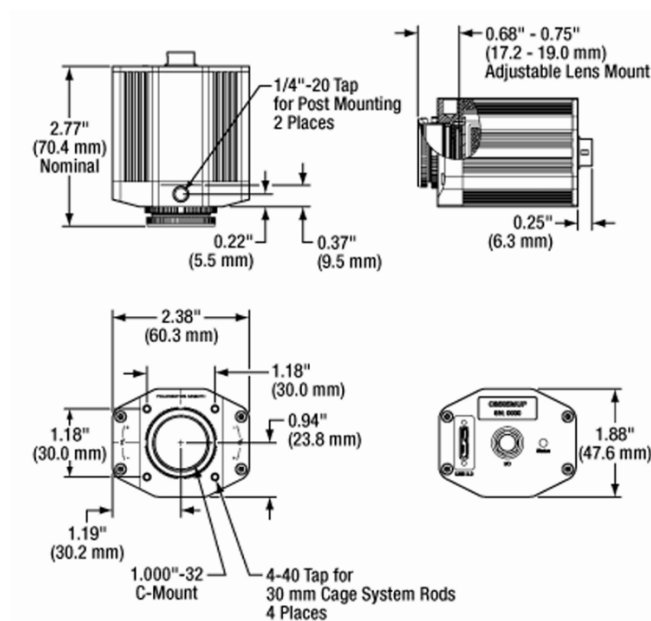


Figura 2.2: Dimensiones de la cámara *Kiralux CS505MUP*, según [documentación de Thorlabs](#)

2.2.1 ESPECIFICACIONES DEL SENSOR

Los atributos y características del sensor utilizado por la cámara se encuentran recogidos en la siguiente tabla [Tabla 2.1]:

Características del Sensor	
Tipo	CMOS Monocroma con array de micropolarización
Nº de píxeles	2448 (H) x 2048 (V) (~5.01 MP)
Tamaño de píxel	3.45 μm x 3.45 μm
Tamaño del sensor	8445.6 μm x 7065.6 μm
Máxima eficiencia cuántica	72% en 525 a 580 nm
Rango dinámico	Hasta 71 dB
Tipo de obturador	Global

Tabla 2.1: Características del sensor de la cámara *Kiralux CS505MUP*

El sensor lleva incorporado un array de micropolarizadores para detectar los estados de polarización lineales de la imagen. Cada píxel está cubierto con uno de estos polarizadores orientados en ángulos de -45° , 0° , 45° o 90° para posteriormente ser procesados y obtener los tres parámetros de polarización de la luz incidente: intensidad, ángulo y grado de polarización. El array de polarización usado sigue un patrón de mosaico, dónde cada polarizador afecta a un único píxel.

Cada cuatro píxeles (2x2) forman un macropíxel con la información de los cuatro canales de polarización [Figura 2.3].

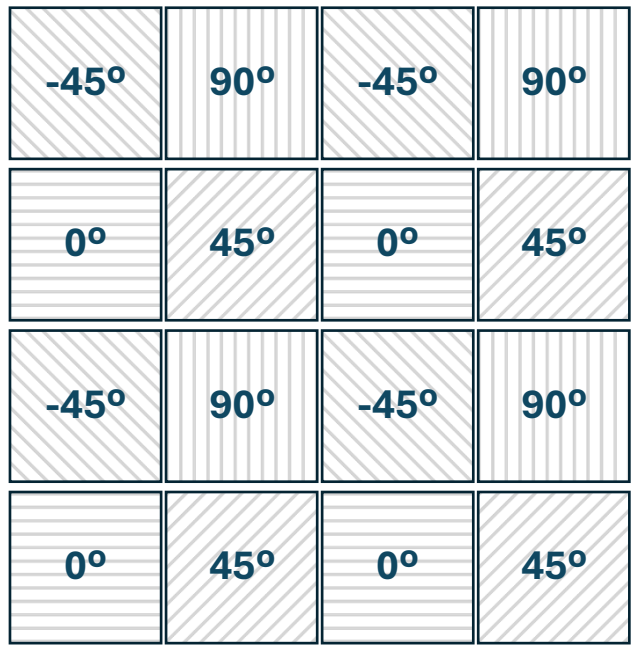


Figura 2.3: Mosaico del array de micropolarizadores

En cuanto a la eficiencia cuántica, su valor varía dependiendo de la longitud de la onda de la luz y viene definida por la gráfica que se muestra a continuación [Figura 2.4]. Como se observa, la eficiencia supera el 70% en el rango entre 470 – 600 nm, que cubre gran parte de la zona central del espectro visible, y decae tanto en la zona azul – violeta (<450 nm) como en el rojo (>600 nm), alcanzando valores muy bajos en infra-rojo (>750nm).

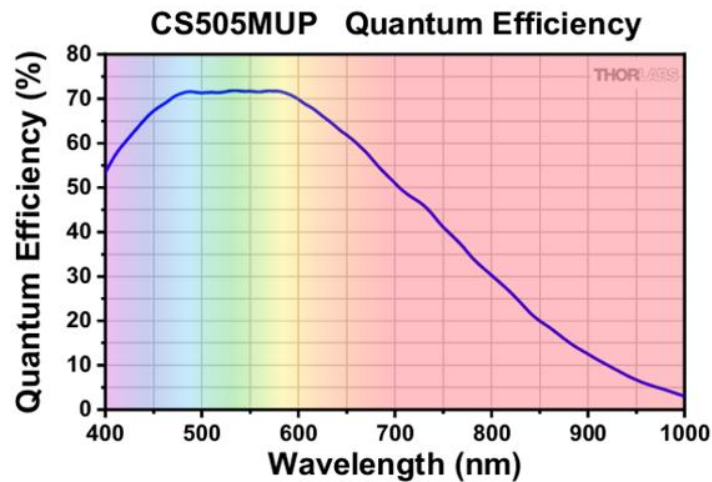


Figura 2.4: Eficiencia Cuántica de la cámara Kiralux CS505MUP, según [documentación de Thorlabs](#)

2.2.2 ESPECIFICACIONES PARA TOMA DE IMÁGENES

Los atributos y características para la formación de imágenes de este modelo de cámara Kiralux vienen recogidos en la tabla siguiente [Tabla 2.2]:

Características de Formación de Imágenes	
Tiempo de exposición	De 0.027 a 14235 ms en incrementos de 0.013 ms
Resolución ADC	12 Bits
Agrupamiento digital de píxeles (binning)	1 x 1 a 16 x 16
Región de Interés (ROI)	De 260 x 4 a 2448 x 2048 Píxeles, Rectangular
Lectura de Ruido	<2.5 e ⁻ RMS

Tabla 2.2: Características de imágenes de la cámara Kiralux CS505MUP

La tasa de fotogramas del vídeo depende del ROI utilizado en la grabación según los detalles de la siguiente tabla [Tabla 2.3].

ROI	Tasa de Fotogramas
Sensor Completo (2448 x 2045)	35 fps
Medio Sensor (1224 x 1024)	68 fps
~1/10 del Sensor (260 x 208)	290 fps
ROI Mínimo (260 x 4)	887.6 fps

Tabla 2.3: Tasa de fotogramas de la cámara Kiralux CS505MUP

La ratio de extinción de los polarizadores en esta cámara alcanza su máximo en longitudes de onda de 400 nm a 500 nm y evoluciona según se puede apreciar en la gráfica [Figura 2.5].

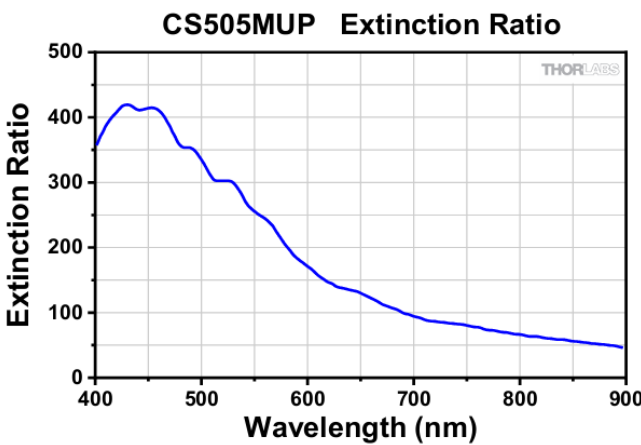


Figura 2.5: Ratio de Extinción de la cámara Kiralux CS505MUP, según [documentación de Thorlabs](#)

2.2.3 COMPONENTES Y ACCESORIOS

La cámara viene equipada con su propio cable USB 3.0 de tres metros de longitud para conectarla al ordenador y una llave de gancho para poder desmontar el foco y poder realizar mantenimiento [Figura 2.6].

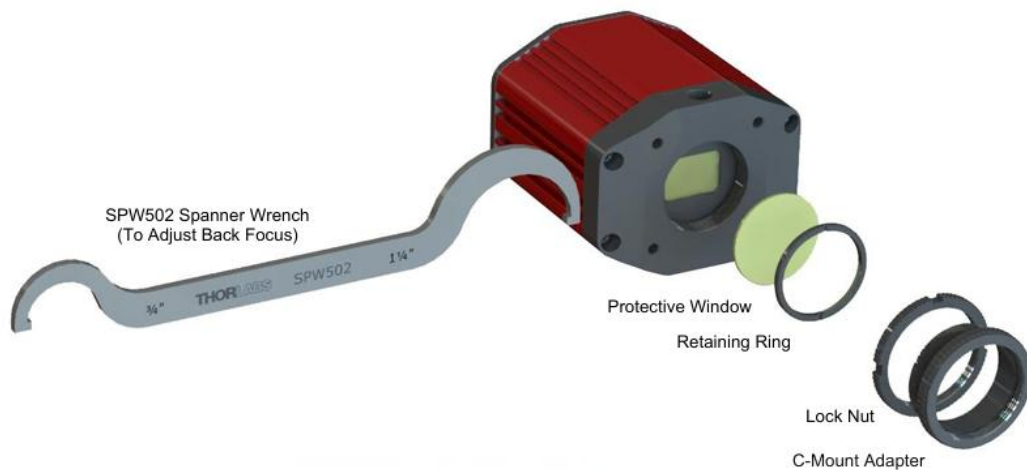


Figura 2.6: Foto de la llave de gancho del [manual de Thorlabs](#)

Además, para usuarios finales, *Thorlabs* ofrece el software *ThorImageCAM* que proporciona control completo de la cámara a través de una interfaz de usuario, permitiéndole capturar, ampliar, analizar y guardar tanto las imágenes como los metadatos [Figura 2.7].

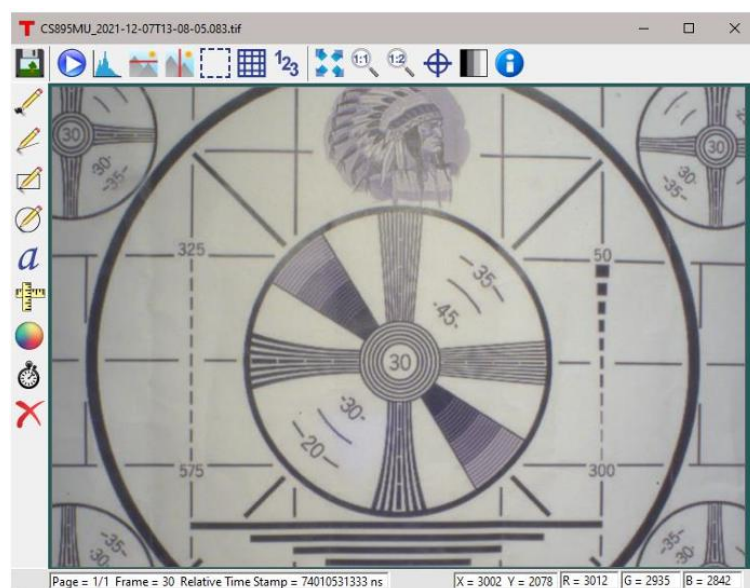


Figura 2.7: Imagen de la Interfaz gráfica de *ThorImageCAM* del [manual de Thorlabs](#)

Por otra parte, para desarrolladores, *Thorlabs* también ofrece su propio SDK (*Software Developer's Kit*) para C, C++, C#, Python y Visual Basic .NET que facilita la integración de cualquiera de sus cámaras en el sistema. Este SDK viene incluido en la instalación de *ThorImageCAM* y se puede encontrar en el subdirectorio *C:\Program Files\Thorlabs*,



2.3 CÁMARA RGB “IMAGINGSOURCE DYK 33UX250”

La segunda cámara utilizada durante el desarrollo del proyecto es el modelo *DYK 33UX250* de *ImagingSource*, la cual es capaz de captar las componentes de polarización a color [Figura 2.8]. El modelo pertenece a la serie 33U y funciona mediante USB 3.0.

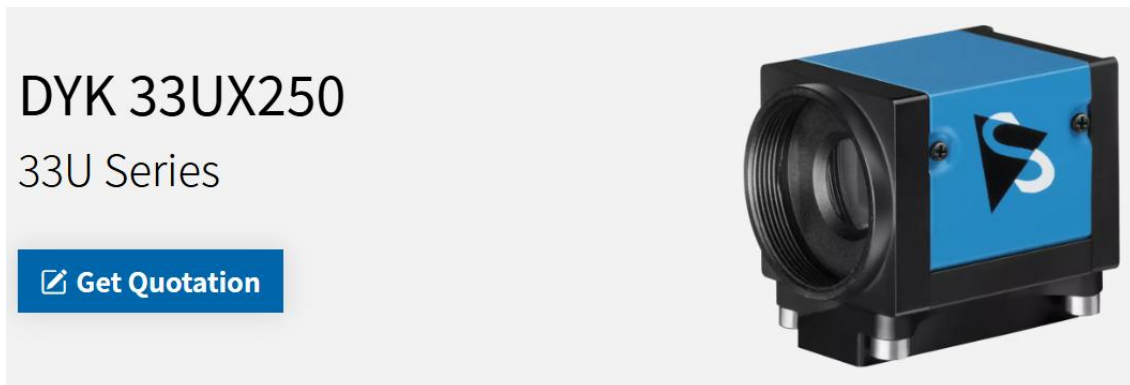


Figura 2.8: Página del producto de la [web oficial de ImagingSource](#)

Las dimensiones del dispositivo son solamente 29 mm de ancho, 29 mm de alto y 43 mm de largo, por lo que es bastante compacto [Figura 2.9].

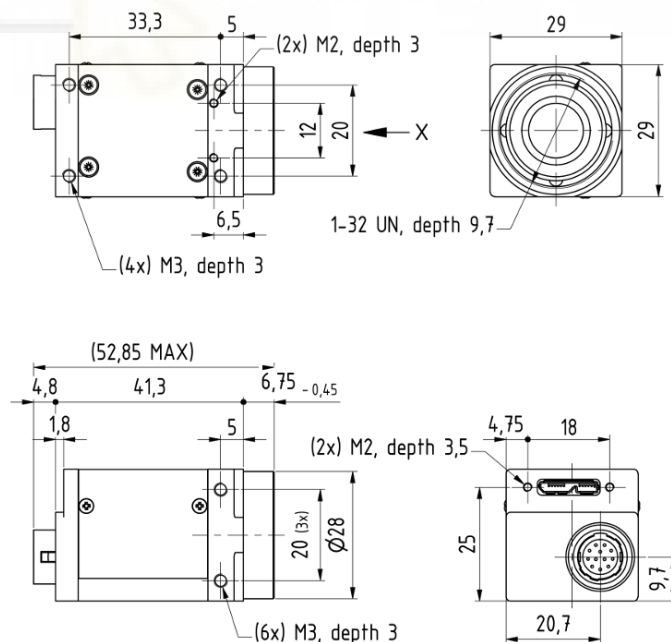


Figura 2.9: Dimensiones de la cámara *DYK 33UX250*, según el [datasheet de ImagingSource](#)

2.3.1 ESPECIFICACIONES DEL SENSOR

Las características principales del sensor de la cámara están recogidas en la siguiente tabla [Tabla 2.4]:

Características del Sensor	
Tipo	CMOS policroma con array de micropolarizadores y filtro de Bayer
Nº de píxeles	2448 (H) x 2048 (V) (~5.01 MP)
Tamaño de pixel	3.45 μm x 3.45 μm
Tamaño del sensor	8.4 mm x 7.1 mm
Rango dinámico	Hasta 72 dB
Tipo de obturador	Global

Tabla 2.4: Características del sensor de la cámara DYK 33UX250

El sensor está cubierto por un array de micropolarizadores en ángulos de 90°, 45°, -45° y 0° que afectan a cada píxel individualmente para conseguir la información necesaria de polarización de la misma forma que la cámara *Kiralux* de *Thorlabs*. No obstante, este además lleva un filtro de Bayer dónde cada macropíxel guarda únicamente el valor de la componente roja, verde o azul acorde al patrón utilizado. Concretamente el sensor utiliza un mosaico Bayer BGGR (*Blue, Green, Green, Red*) [Figura 2.10].

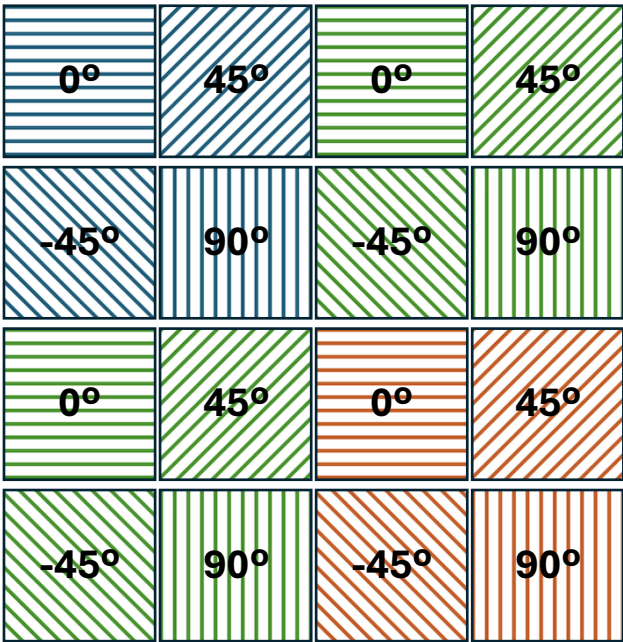


Figura 2.10: Mosaico del array de micropolarizadores y filtro de Bayer

El sensor utilizado por la cámara es un *Sony IMX250MYR-C* cuya sensibilidad espectral (variación de la transmisión y la eficiencia cuántica según la longitud de onda) viene definida por la siguiente gráfica [Figura 2.11]:

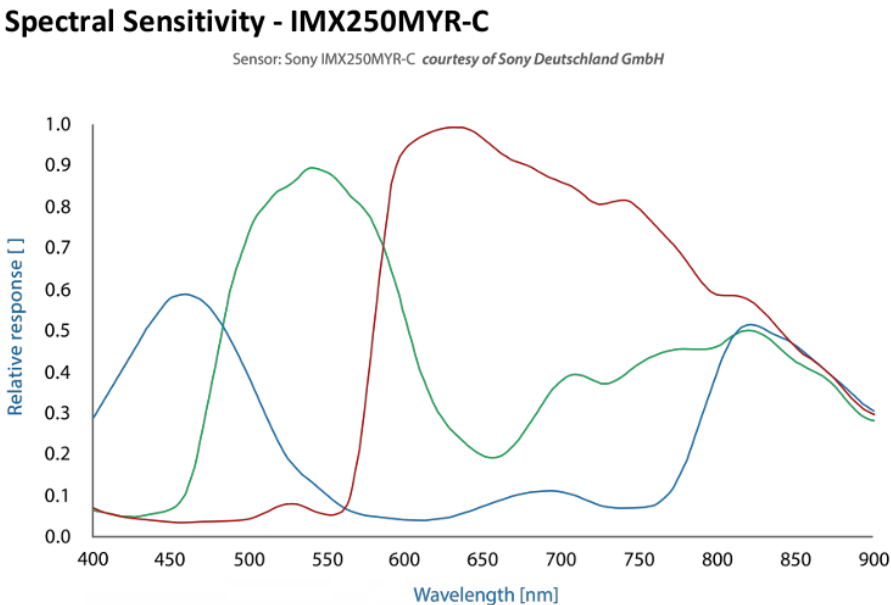


Figura 2.11: Sensibilidad espectral del sensor, según [datasheet de ImagingSource](#)

2.3.2 ESPECIFICACIONES PARA LA TOMA DE IMÁGENES

Las características del modelo *DYK 33UX250* más relevantes a la hora de formar imágenes vienen recogidas en la siguiente tabla [Tabla 2.5]:

Características de Formación de Imágenes	
Tiempo de Exposición	De 1µs a 4 s
Ganancia	De 0 dB a 48 dB
Región de Interés (ROI)	De 256 x 8 a 2448 x 2048 Píxeles, Rectangular

Tabla 2.5: Características de imágenes de la cámara *DYK 33UX250*

La tasa de fotogramas del vídeo producido depende tanto de la región de interés (ROI) como del formato de píxel utilizado. Los valores correspondientes dependiendo de la combinación se encuentran detallados en la tabla mostrada a continuación [Tabla 2.6].

ROI	Tasa de Fotogramas
2448 x 2048	75 fps
2048 x 2048	89 fps
1920 x 1080	181 fps
640 x 480	608 fps

Tabla 2.6: Tasas de fotogramas de la cámara *DYK 33UX250*

2.3.3 COMPONENTES Y ACCESORIOS

La empresa *ImagingSource* ofrece varias facilidades para poder manejar el dispositivo una vez conectado al ordenador con un cable USB 3.0.

Para los usuarios finales podemos encontrar el software “*IC Capture*” disponible en la página de la propia *ImagingSource*. Este es un programa que nos permite capturar imágenes, aplicar filtros, facilitar tareas de calibración o ver medidas en tiempo real con una interfaz amigable y sencilla de comprender **[Figura 2.12]**.

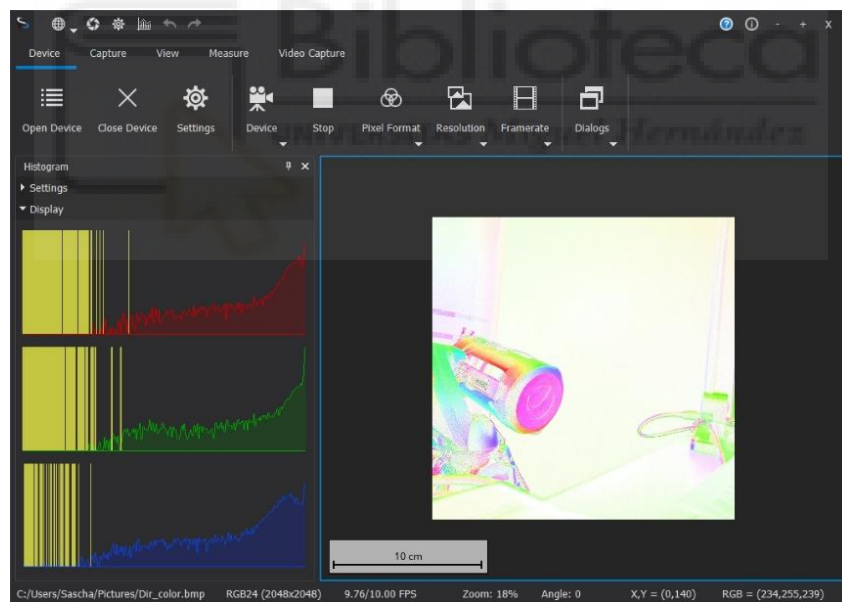


Figura 12: Imagen de la interfaz del software *IC Capture* del [catálogo de ImagingSource](#)

Adicionalmente, para desarrolladores, podemos descargar en la misma página los SDK para poder hacer uso de las cámaras *ImagingSource* en nuestros programas. Concretamente la empresa ofrece kits de desarrollo para C, C++, .NET y Python.

2.4 RETARDADOR LCR Y CONTROLADOR “*THORLABS*

KLC101”

Para la realización del proyecto se ha utilizado un retardador de cristal líquido de dos pulgadas de diámetro haciendo uso de una montura *LCRM2/M* de *Thorlabs* [Figura 2.13]. Este dispositivo permite variar la polarización de la luz mediante la aplicación de un voltaje.



Figura 2.13: Retardador de cristal líquido utilizado

Para el control voltaje aplicado a la celda de cristal líquido se ha utilizado un “*KLC101 K-Cube Liquid Crystal Controller*” de la gama de componentes de *Thorlabs* [Figura 2.14].



KLC101
Liquid Crystal Controller
(Power Supply Sold Separately)

Figura 2.14: Página del producto de la [web oficial de Thorlabs](https://www.thorlabs.com/)

Este dispositivo permite suministrar un voltaje de onda cuadrada con una frecuencia y amplitud definida por el usuario. Cualquier desplazamiento es detectado automáticamente y corregido en un rango de $\pm 10\text{mV}$ en tiempo real, lo que ayuda a aumentar el tiempo de vida del dispositivo. Las opciones se pueden ajustar usando la rueda del panel superior o desde un ordenador conectado por puerto USB.

Las dimensiones del dispositivo son 60mm de ancho, 60mm de largo y 46.7mm de alto (sin contar botones, base y conectores) [Figura 2.15].

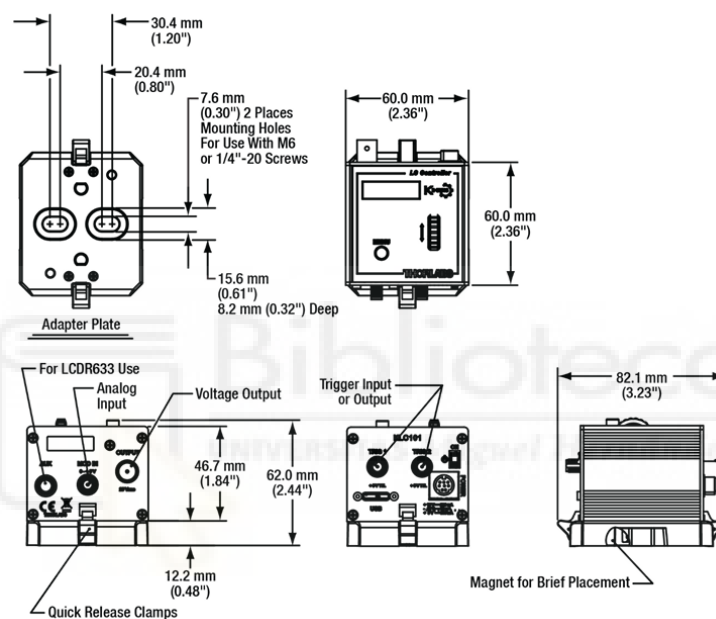


Figura 2.15: Dimensiones de *Thorlabs KLC101*, según [documentación de Thorlabs](#)

2.4.1 ESPECIFICACIONES Y CARACTERÍSTICAS

Las principales características a tener en cuenta del controlador aparecen recopiladas en la siguiente tabla [Tabla 2.7]:

Características Eléctricas	
Rango de voltaje	0 a ± 25 V RMS
Resolución del voltaje	1.0 mV
Rango de frecuencia ajustable	500 Hz a 10 KHz
Tasa de respuesta	10 V/ μ s
Desplazamiento DC	± 10 mV
Tiempo de calentamiento	30 minutos
Puerto USB	USB 3.0 Micro B
Valores Límite	
Máximo voltaje externo	5 VDC
Modulación externa máxima	150 Hz
Rango de temperatura	10 a 40 °C
Máximo nivel de humedad	85%

Tabla 2.7: Características eléctricas y valores límite de *KLC101*

El dispositivo viene con su propio cable USB 3.0 para poder conectarlo al PC. Este es de 1.5 m de longitud y el fabricante recomienda no usar un cable que supere los 3 m.

Además, para poder controlar el dispositivo con mayor facilidad usando el ordenador, *Thorlabs* ofrece el software “*Kinesis LC Controller*”, el cual permite controlar uno o varios dispositivos de este mismo fabricante usando una interfaz visual fácil de comprender [Figura 2.16].

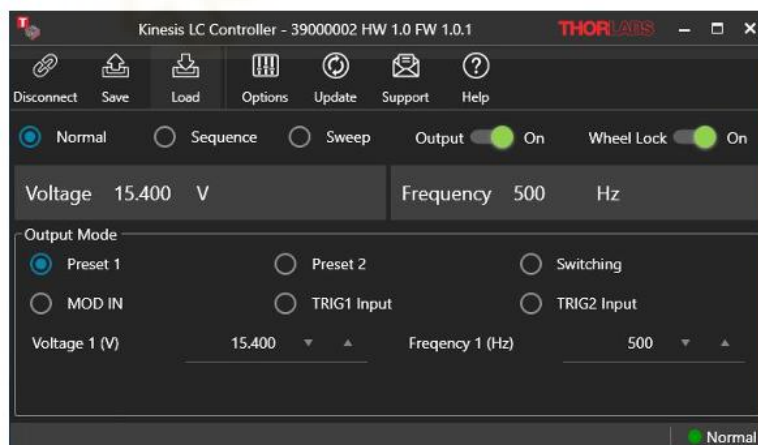


Figura 2.16: Imagen de la interfaz de “*Kinesis LC Controller*” de la [documentación de Thorlabs](#)

La descarga del software también incluye el SDK para desarrolladores con soporte para C++, Labview y Python.

2.5 SELECCIÓN DE HERRAMIENTAS DE DESARROLLO

A la hora de desarrollar el proyecto se han tenido en cuenta tres alternativas diferentes de desarrollo: C++, Java y Python. Para tomar la decisión final de entre estas opciones se ha realizado una tabla comparativa de sus características para facilitar la decisión, teniendo en cuenta que la compatibilidad con todos los SDK de los diferentes componentes a utilizar es un requisito que cumplir [Tabla 2.8].

	C++	Java	Python
Tipo	Compilado	Compilado	Interpretado
Aprendizaje	Medio	Lento	Rápido
Facilidad de Uso	Media	Media	Muy alta
Disponible	Sí	Sí	Sí
Compatible SDK cámara Monocroma	Sí	No	Sí
Compatible SDK cámara RGB	Sí	No	Sí
Compatible SDK dispositivo LCR	Sí	No	Sí

Tabla 2.8: Comparativa de características de lenguajes de programación

Gracias a la comparación anterior podemos descartar el uso de Java como lenguaje de programación principal, al no ser soportado por los SDK proporcionados por los fabricantes de los dispositivos a integrar en el programa. Comparando las características de C++ y Python expuestas anteriormente se ha decidido realizar la implementación del programa haciendo uso de Python.

El factor determinante en esta decisión es la facilidad de uso y de aprendizaje que este proporciona frente a sus competidores. Como el proyecto a desarrollar es un software de laboratorio es muy probable que en el futuro se deban hacer modificaciones o actualizaciones para adaptarlo a nuevas necesidades de análisis o experimentación. Realizar el desarrollo en Python permite que dichos cambios puedan ser llevados a cabo por personas no necesariamente especializadas en ingeniería informática, lo cual es una gran ventaja al no estar garantizada la presencia de informáticos en el laboratorio.

Por otra parte, el hecho de que Python sea un lenguaje interpretado en lugar de compilado le confiere una pequeña desventaja frente a C++. Como se verá con más

detalle en el apartado “3.1 Requisitos”, uno de los requisitos del programa es poder calcular los parámetros de Stokes en tiempo real. Un lenguaje compilado sería capaz de tratar una mayor tasa de fotogramas por segundo al tener una velocidad de ejecución mayor. Sin embargo, el rendimiento que proporciona el uso de Python es suficiente para la tarea a realizar, por lo que no es motivo suficiente para ignorar las ventajas presentadas anteriormente.



3. REQUERIMIENTOS Y DISEÑO

3.1 SOFTWARE PARA CÁMARA MONOCROMÁTICA DE POLARIZACIÓN

3.1.1 REQUISITOS

En la siguiente tabla tenemos los principales requerimientos que debe cumplir el software [Tabla 3.1].

Requisito	Tipo
Cálculo en tiempo real parámetros de Stokes	Funcional, Técnico
Captura de vídeo con cámara	Funcional
Control sobre el vídeo	Funcional, Técnico
Control sobre el LCR	Funcional, Técnico
Visualización de resultados clara	Usabilidad
Guardar resultados	Funcional
Crear muestra de color	Funcional, Usabilidad

Tabla 3.1: Requisitos principales del software para la cámara monocromática

3.1.2 DIAGRAMA DE CASOS DE USO

El sistema encargado del control y gestión de la cámara de polarización monocromática cuenta con un único actor que aparece en todos los sistemas [Tabla 3.2] y un total de 10 casos de usos diferentes [Figura 3.1]. Las tablas de detalle de cada caso de uso se encuentran agrupadas en el anexo [Anexo 1.1].

Actor	Usuario
Descripción	Persona que usa la aplicación sin ningún tipo de privilegio o identificación.
Casos de Uso relacionados	• Sistema de la cámara monocroma de polarización: C1.1, C1.2, C1.3, C1.4, C1.5, C1.6, C1.7, C1.8, C1.9, C1.10 • Sistema de calibración del LCR: C2.1, C2.2, C2.3, C2.4, C2.5, C2.6, C2.7, C2.8, C2.9 • Sistema de calibración del monitor: C3.1, C3.2, C3.3, C3.4, C3.5, C3.6, C3.7 • Sistema de la cámara RGB de polarización: C4.1, C4.2, C4.3, C4.4, C4.5, C4.6

Tabla 3.2: Definición de actores del sistema

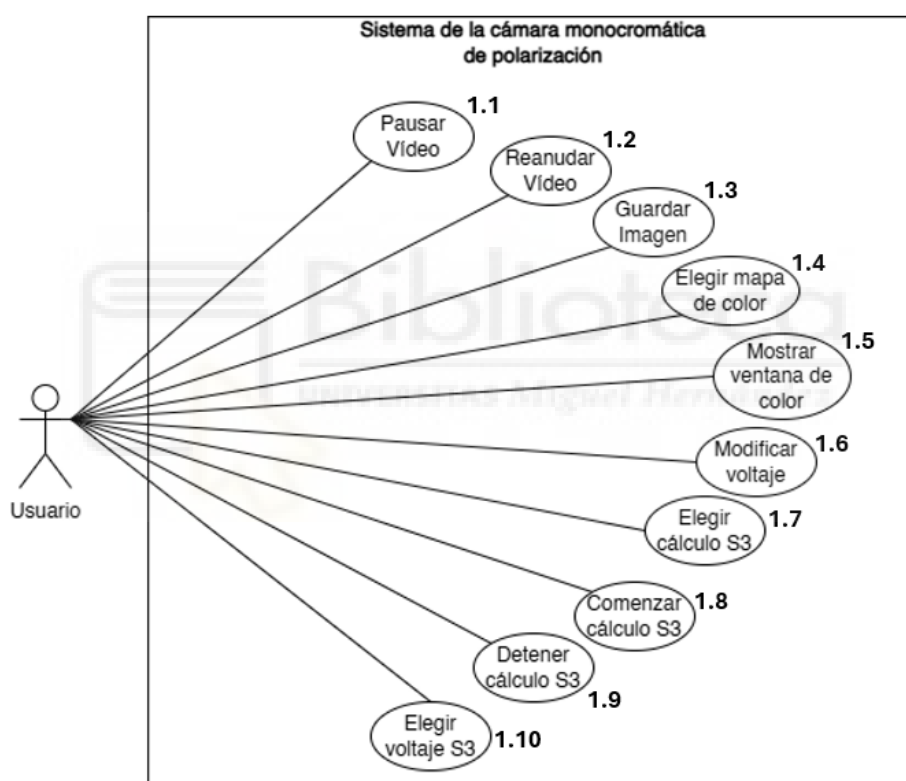


Figura 3.1: Diagrama de casos de uso del sistema para cámara monocromática, realizado con Draw.io

3.1.3 DIAGRAMA DE GANTT

El desarrollo de todo el proyecto en conjunto estaba planificado para durar cuatro meses. Debido a que solo se contaba con una única persona el avance de los programas se ralentiza al realizar diferentes proyectos de forma concurrente, como es el caso para el desarrollo de este software **[Figura 3.2]**.

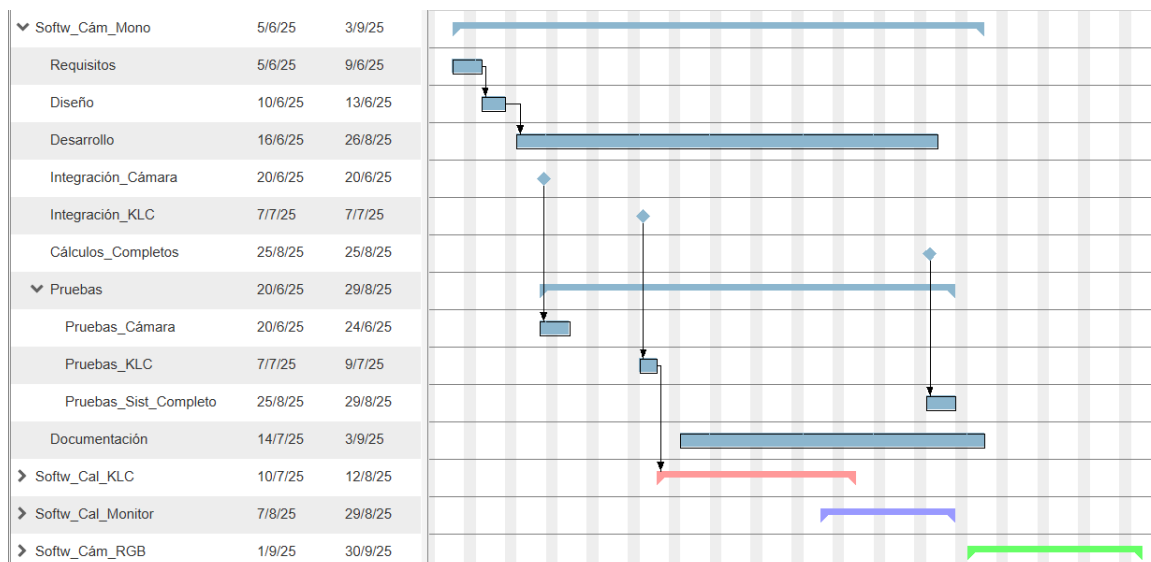


Figura 3.2: Diagrama de Gantt del desarrollo del software para cámara monocromática, realizado con GanttProject

3.1.4 DISEÑO

En el diseño original el programa estaba distribuido con la toma monocromática a la izquierda, seguida de los cálculos de los parámetros de Stokes y de los controles [Figura 3.3].

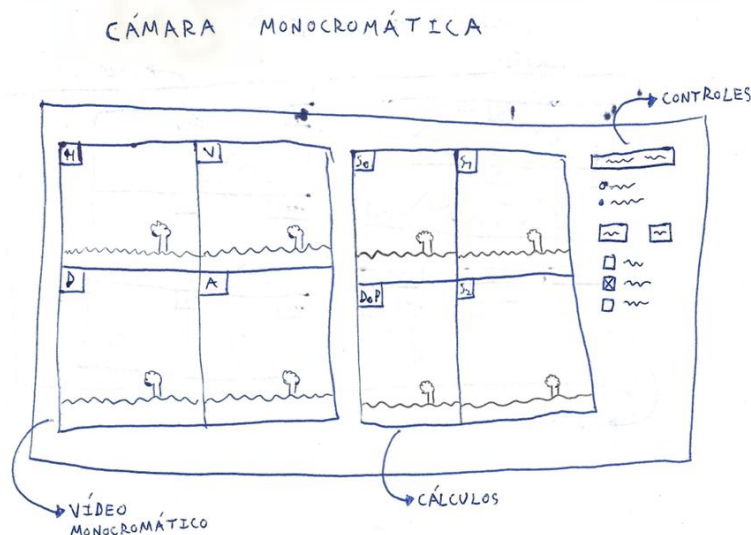


Figura 3.3: Primer boceto a papel de la interfaz del software para cámara monocromática

No obstante, como los cálculos ocupan más espacio que el vídeo, posteriormente se rediseñó la interfaz con el área de control en la zona inferior izquierda para

conseguir un diseño más compacto [Figura 3.4]. Esta segunda iteración es la utilizada en la implementación.

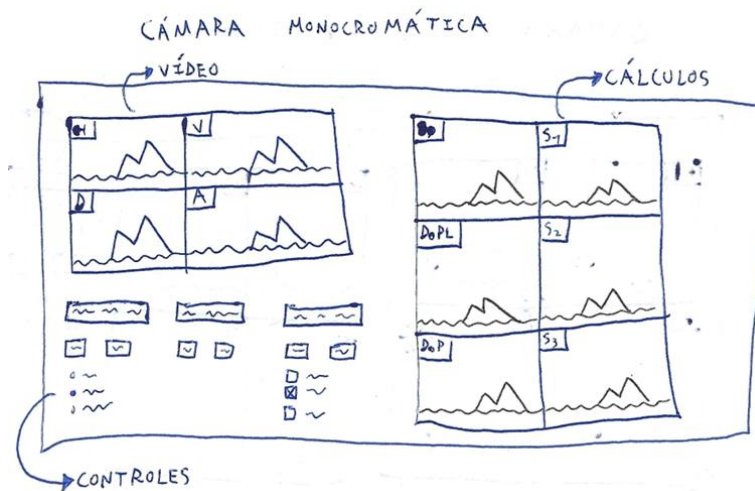


Figura 3.4: Segundo boceto a papel de la interfaz del software para cámara monocromática



3.2 SOFTWARE PARA CALIBRACIÓN DEL LCR

3.2.1 REQUISITOS

Los requisitos que este software debe cumplir se encuentran recogidos en la siguiente tabla [Tabla 3.3].

Requisito	Tipo
Selección zona de análisis	Funcional, Usabilidad
Control sobre LCR	Funcional, Técnico
Modificar características del análisis	Funcional
Muestra gráfica de resultados	Usabilidad
Guardar resultados	Funcional
Captura de vídeo en vivo	Funcional, Técnico

Tabla 3.3: Requisitos principales del software para calibración del LCR

3.2.2 DIAGRAMA DE CASOS DE USO

El sistema para calibración del LCR presenta nueve casos de usos diferentes [Figura 3.5] [Anexo 1.2] realizados por un único actor [Tabla 3.2].

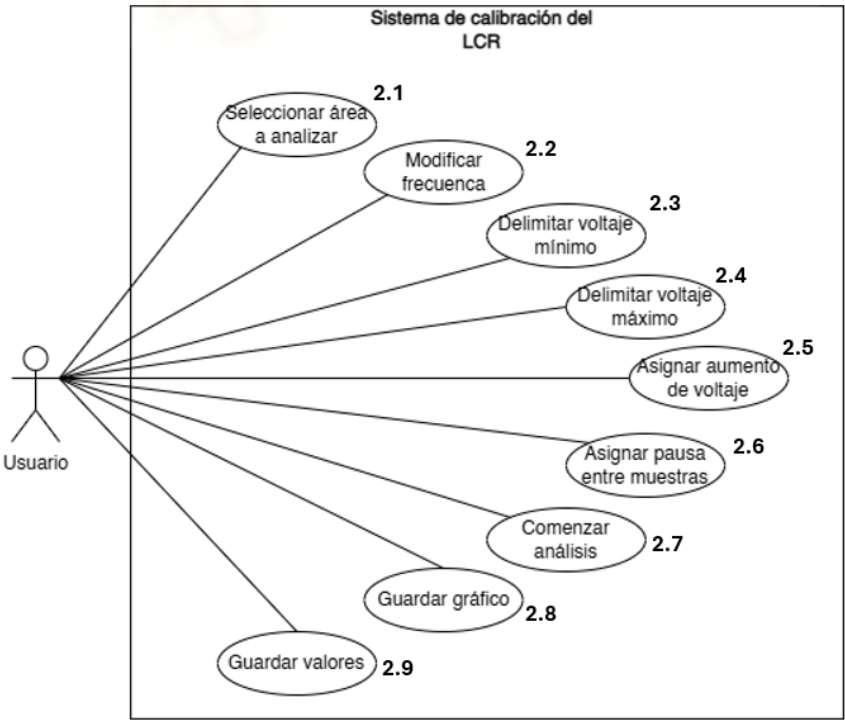


Figura 3.5: Diagrama de casos de uso del sistema para calibración del LCR, realizado con Draw.io

3.2.3 DIAGRAMA DE GANTT

En la siguiente imagen se encuentra la planificación temporal del proyecto **[Figura 3.6]**, el desarrollo del cual depende del hito de integrar el dispositivo “*Thorlabs KLC101*” al programa de la cámara monocromática.

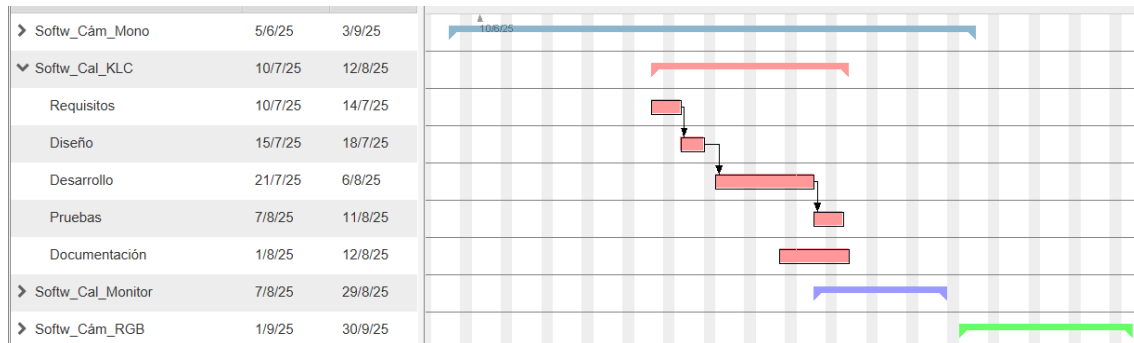


Figura 3.6: Diagrama de Gantt del desarrollo del software para calibración del LCR, realizado con GanttProject

3.2.4 DISEÑO

La distribución del diseño de la interfaz tiene la toma de video en la zona izquierda, seguida de los controles y los resultados **[Figura 3.7]**.



Figura 3.7: Boceto a papel de la interfaz del software para calibración del LCR

3.3 SOFTWARE PARA CALIBRACIÓN CON MONITOR

3.3.1 REQUISITOS

El software ha sido desarrollado acorde con los siguientes requisitos [Tabla 3.4].

Requisito	Tipo
Captura de vídeo en vivo	Funcional, Técnico
Selección de área de análisis	Funcional, Usabilidad
Modificar parámetros del análisis	Funcional
Muestra de resultados visual	Usabilidad
Guardar resultados	Funcional
Creación de muestra de color	Funcional, Usabilidad

Tabla 3.4: Requisitos principales del software para calibración con monitor

3.3.2 DIAGRAMA DE CASOS DE USO

Una vez más nos encontramos con el usuario como único actor [Tabla 3.2], el cual realiza los siete casos de uso presentes en el sistema [Figura 3.8] [Anexo 1.3].

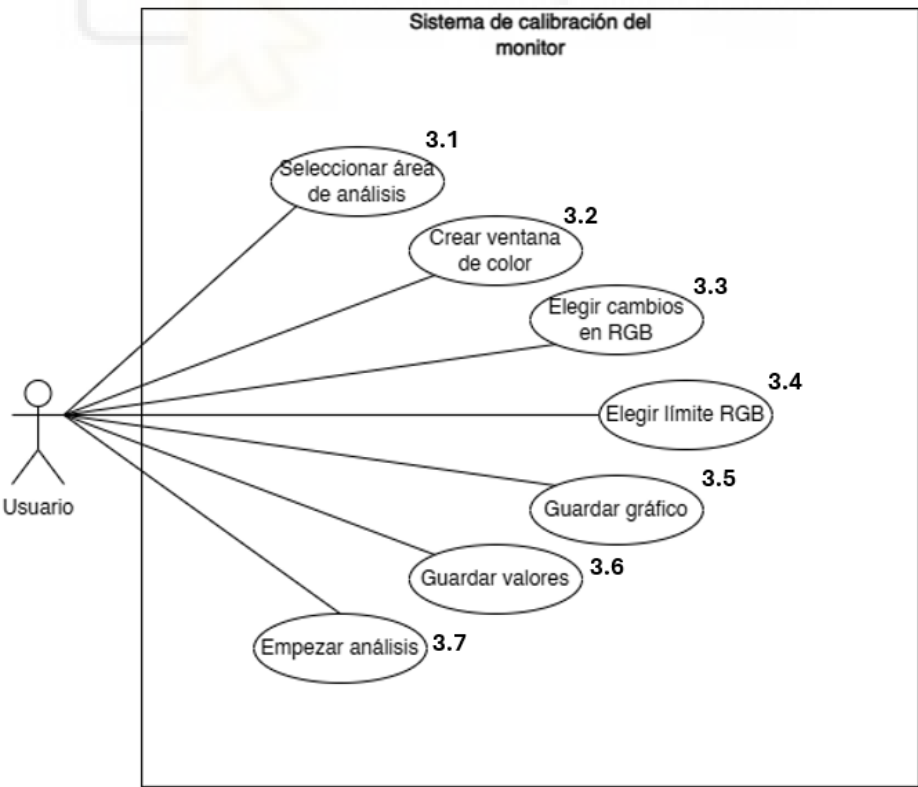


Figura 3.8: Diagrama de casos de uso del sistema para calibración del monitor, realizado con Draw.io

3.3.3 DIAGRAMA DE GANTT

La inclusión de este software en el diagrama de Gantt se hizo una vez detectado el problema de saturación de los colores debido a un mal ajuste del brillo. A continuación, se muestra la distribución temporal de sus tareas [Figura 3.9].

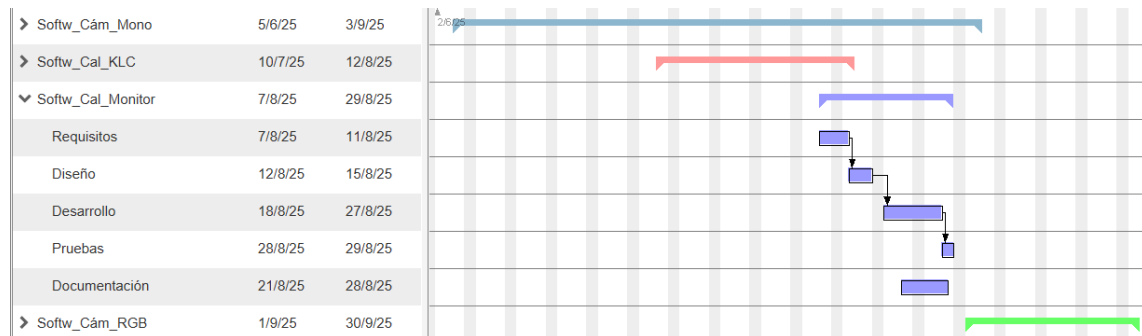


Figura 3.9: Diagrama de Gantt del desarrollo del software para calibración con monitor, realizado con GanttProject

3.3.4 DISEÑO

Para que el uso del programa resulte familiar a los usuarios, se ha utilizado la misma distribución de elementos en ambos softwares de calibración [Figura 3.10].

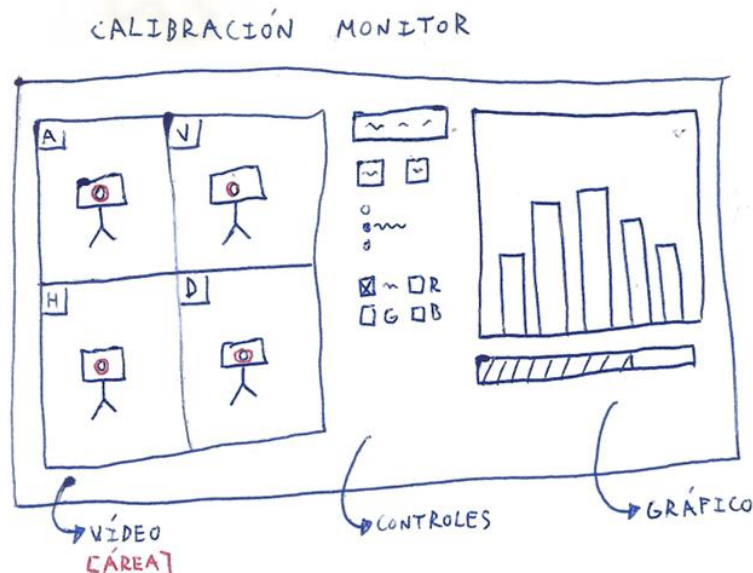


Figura 3.10: Boceto a papel de la interfaz del software para calibración con monitor

3.4 SOFTWARE PARA CÁMARA RGB DE POLARIZACIÓN

3.4.1 REQUISITOS

Los requerimientos de este programa se encuentran recogidos en la tabla siguiente [Tabla 3.5].

Requisitos	Tipo
Visualización RGB en tiempo real	Funcional, Técnico
Control sobre el vídeo	Funcional
Múltiples modos de visión	Funcional, Técnico
Guardar imágenes	Funcional

Tabla 3.5: Requisitos principales del software para la cámara RGB

3.4.2 DIAGRAMA DE CASOS DE USO

En el siguiente diagrama se muestran los casos de uso [Figura 3.11] [Anexo 1.4] correspondientes a cada actor del sistema [Tabla 3.2].

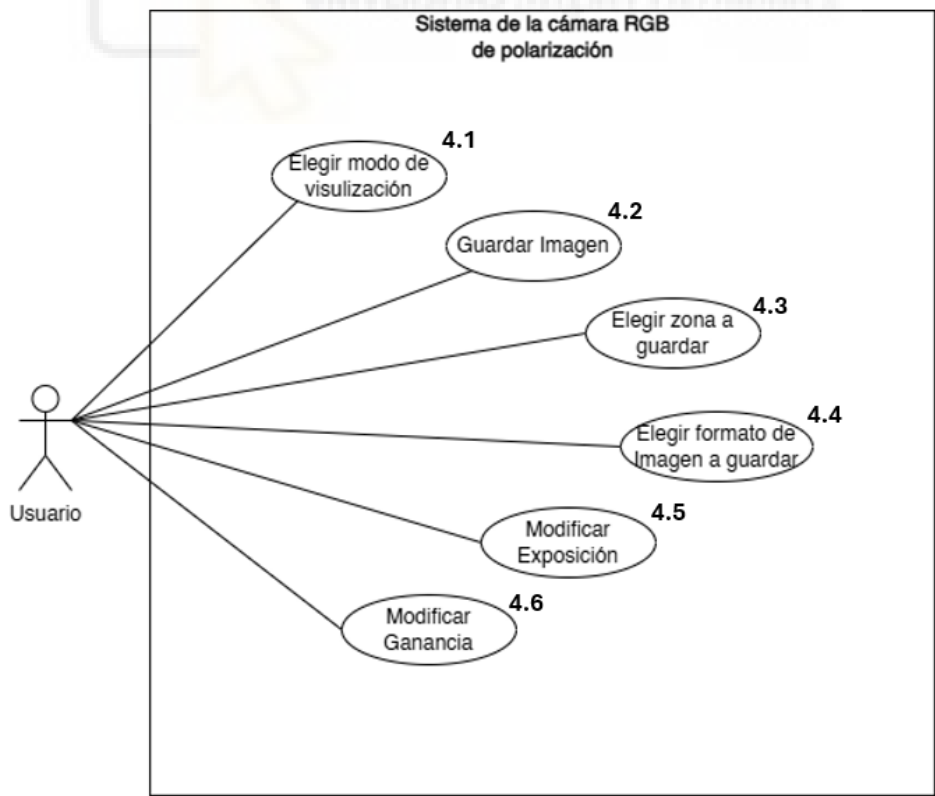


Figura 3.11: Diagrama de casos de uso del sistema para la cámara RGB, realizado con Draw.io

3.4.3 DIAGRAMA DE GANTT

La distribución temporal de las tareas necesarias para la realización de este software se encuentra representada en el siguiente diagrama de Gantt [Figura 3.12].

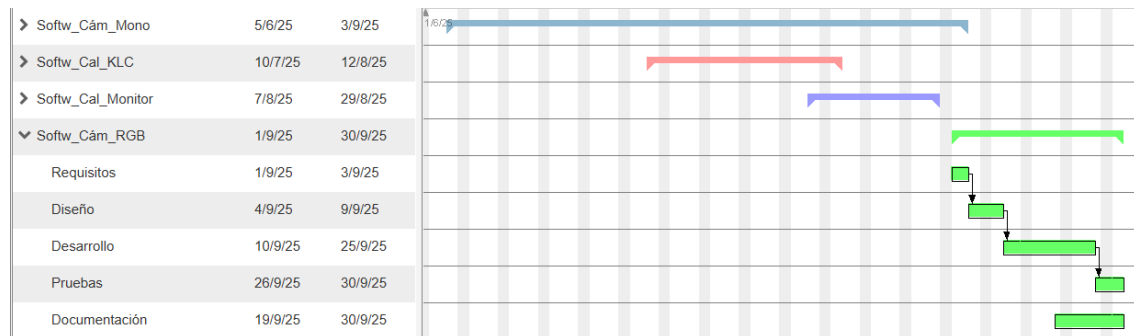


Figura 3.12: Diagrama de Gantt del desarrollo del software para la cámara RGB, realizado con GanttProject

3.4.4 DISEÑO

La interfaz de este programa es la más sencilla de todas, teniendo solo dos áreas diferenciadas: la toma de vídeo a la izquierda y los controles a la derecha [Figura 3.13].

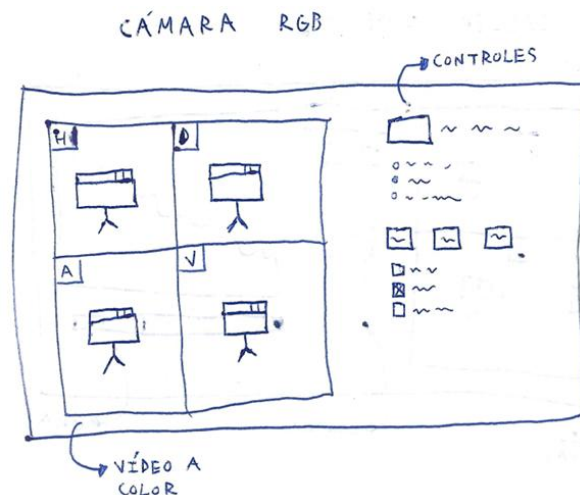


Figura 3.13: Boceto a papel de la interfaz del software para la cámara RGB

4. IMPLEMENTACIÓN

4.1. SOFTWARE PARA CÁMARA MONOCROMÁTICA DE POLARIZACIÓN

4.1.1 FUNCIONALIDAD BÁSICA

Al iniciarse el programa se crea una ventana haciendo uso de la librería “*tkinter*” con el nombre de la cámara utilizada que sirve como interfaz para manejar el software [Figura 4.1]. La aplicación ha sido diseñada con los dispositivos utilizados en el laboratorio en mente, siendo ordenadores de torre y portátiles, por lo que el tamaño de la ventana se adapta para que ocupe alrededor del 70% del ancho de la pantalla al iniciarse, garantizando un uso correcto en diferentes dispositivos.

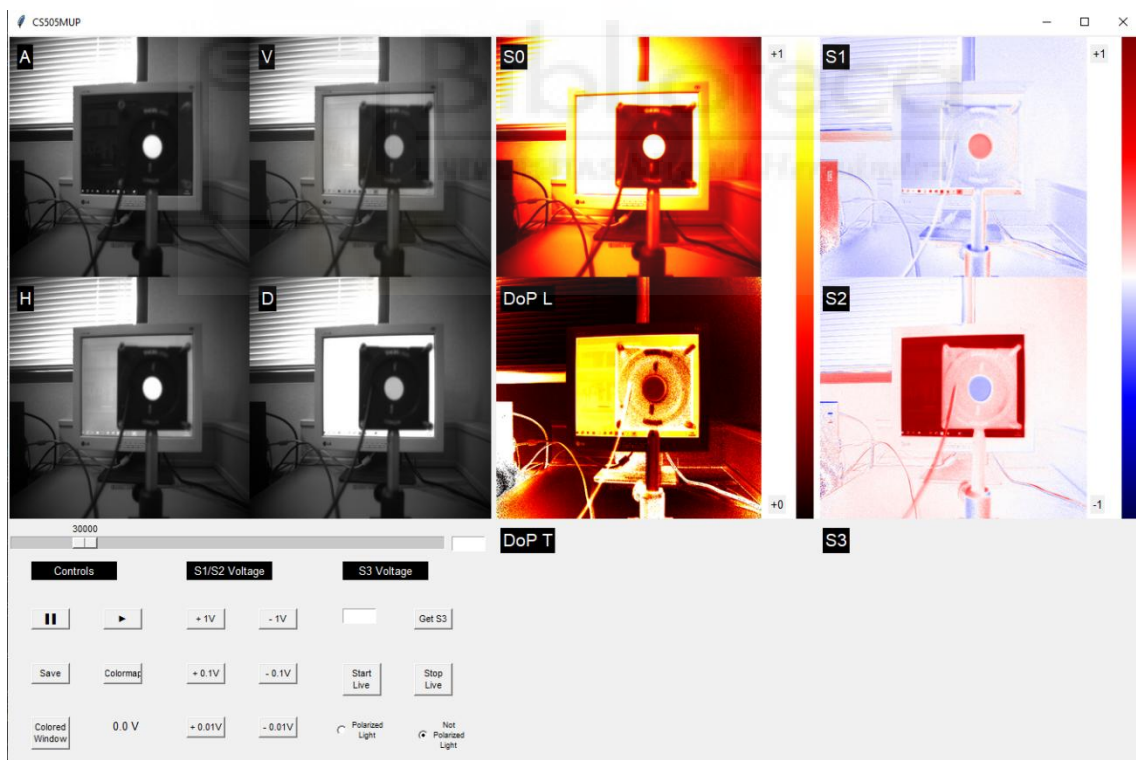


Figura 4.1: Ventana creada al iniciar el programa.
Solamente se muestran S_0 , S_1 , S_2 y DoP Lineal

Durante la realización de esta parte del proyecto se ha usado la cámara de polarización monocromática comercial “*Thorlabs Kiralux CS505MUP*”. Para poder

utilizar el componente simplemente debemos tener su SDK instalado y conectar la cámara al ordenador mediante una conexión USB 3.0.

Todas las funciones que hacen uso de la cámara, como la captura de vídeo, cálculos o detalles de control (brillo, pausa, etc.) se han agrupado dentro de una misma clase. La finalidad de esto es facilitar la futura modificación y ampliación de este *software* y la reutilización de estas funciones en proyectos futuros.

Como podemos observar, en la zona superior izquierda de la pantalla se muestra la toma de vídeo monocromático en tiempo real capturado por la cámara, separado en las cuatro orientaciones del ángulo de polarización: Horizontal (H), Vertical (V), Diagonal (D) y Antidiagonal (A).

Para generar este vídeo primero se obtienen los fotogramas capturados con la cámara en modo “*Quadview*”, que captura información en los cuatro ángulos de polarización lineales. No obstante, este modo tiene los píxeles de cada polarización mezclados en forma de mosaico siguiendo la distribución de micropolarizadores del sensor representada en el apartado 2.2 **[Figura 2.3]**, por lo que hay que reordenar la información de cada *frame* para distinguir de forma clara cada imagen.

Como esta reordenación de información se debe hacer de forma constante en el tiempo, la captura de vídeo y su respectivo tratamiento se realizan en un hilo de ejecución diferente. Esto permite al usuario mantener el control sobre el programa y la cámara desde el hilo de ejecución principal al no necesitar esperar a que dicho tratamiento finalice para realizar acciones. Para pasar las imágenes al resto de hilos se han usado colas con una capacidad máxima de dos elementos.

A la hora de mostrar el vídeo por pantalla se ha hecho uso del componente “*canva*”. Este se encarga de recibir las imágenes del hilo de la cámara usando la cola. La función que se encarga de esto es recursiva y se llama a sí misma cada diez milisegundos y, una vez se ha ejecutado, llama a otra función que se encarga de ajustar el tamaño de la imagen al lienzo y mostrarla por pantalla. Como este comportamiento se va a integrar en diferentes “*canva*”, estas funciones se han agrupado en una clase para posterior reutilización.

Al lado de la toma de video monocromática, en el área de la derecha, se muestran los cálculos de los parámetros de Stokes y del grado de polarización (DoP). Los parámetros S_0 , S_1 , S_2 y el grado de polarización lineal (DoPL) se calculan constantemente en todo momento, siempre y cuando la cámara se encuentre conectada. Como estos cálculos añaden un tiempo de cómputo adicional, se ha decidido formar otro hilo de ejecución diferente para estos y así no comprometer la tasa de fotogramas del video monocromático, la cual es superior a la de los parámetros.

Este hilo de cálculos toma la última imagen procesada con los diferentes grados de polarización separados del hilo anterior, por lo es necesario que ambos se encuentren en ejecución para un uso correcto. Una vez recibida la imagen se calculan los parámetros de Stokes S_0 , S_1 y S_2 [Ecuación 1.1] junto a DoPL (grado de polarización lineal) [Ecuación 1.4].

Los cálculos de S_3 y del DoP no se realizan de forma constante al necesitar que el usuario realice la configuración óptica necesaria para garantizar un cálculo completo correcto incluyendo las polarizaciones circulares, tras lo cual comenzarán a mostrarse [Figura 4.2].

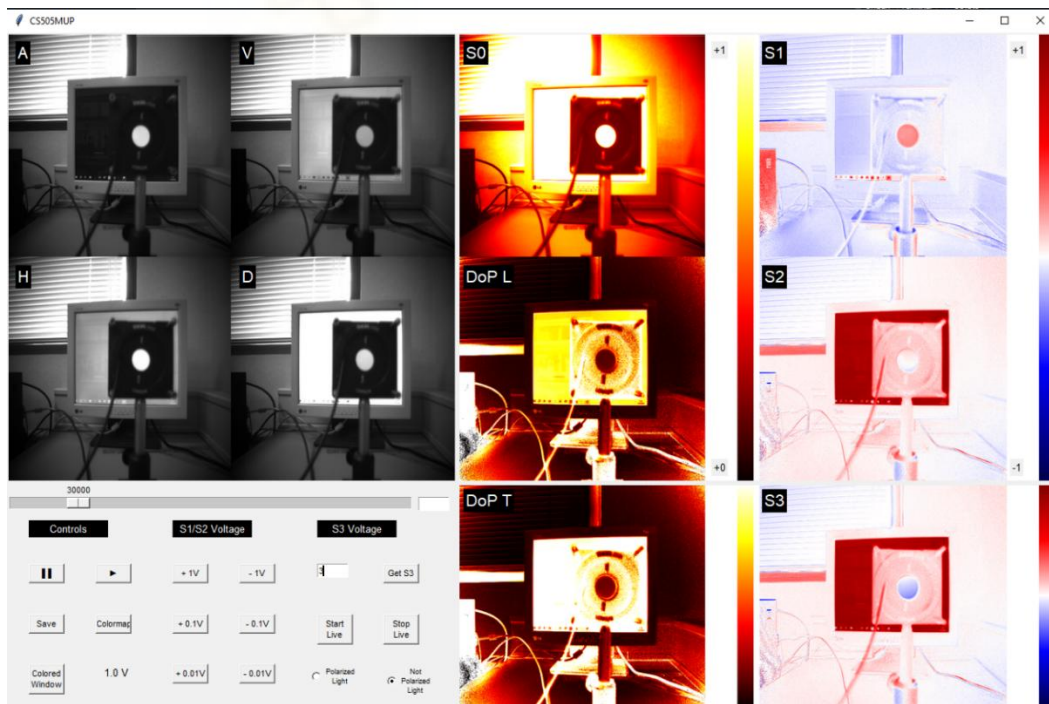


Figura 4.2: Muestra de resultados incorporando S_3 y DoP Total

El usuario puede elegir entre dos formas diferentes de calcular el parámetro S_3 , necesario para la obtención de DoP **[Ecuación 1.3]**. La primera forma solo funciona con luz totalmente polarizada y se puede utilizar para conseguir una toma de video en tiempo real. En esta, S_3 se calcula a partir de S_1 y S_2 **[Ecuación 1.2]**, obteniendo el resultado en valor absoluto, por lo que si nos interesa analizar el parámetro teniendo en cuenta el signo tendremos que realizar el cálculo usando el método alternativo.

La segunda forma de realizar los cálculos sirve para todo tipo de luz, pero requiere el uso de un LCR para alterar el ángulo de polarización de una celda de cristal líquido si queremos conseguir un resultado correcto que mida las componentes circulares CD y CI. Para poder obtener los valores de los parámetros, el usuario debe especificar primero dos valores de voltaje que se aplicarán de forma alterna. El valor de los parámetros S_0 , S_1 y S_2 se calcularán utilizando el voltaje especificado en la zona “S1/S2 Voltage”, mientras que S_3 será equivalente al parámetro S_2 **[Ecuación 1.1]** con el nuevo voltaje especificado en la zona “S3 Voltage”.

De esta forma se obtiene el valor real del componente S_3 , sin embargo, debido al retardo introducido por el LCR, el resultado final son capturas instantáneas en lugar de una toma de vídeo.

Una vez tenemos los resultados, aplicamos el mapa de color correspondiente para poder diferenciar con mayor facilidad las zonas con diferentes valores en la imagen. Para los parámetros S_0 y ambos DoP se ha utilizado el mapa “hot” de la librería “matplotlib”, mientras que S_1 , S_2 y S_3 utilizan una representación de colores diferente, ya que al contener tanto valores positivos como negativos es importante que las diferentes regiones sean fácilmente diferenciables. Por este motivo se han implementado múltiples visualizaciones en forma de lista cíclica que el usuario puede ir modificando, aplicándose por defecto el mapa “seismic” con la posibilidad de cambiar a “jet” y “hsv” **[Figura 4.3]**. Como es muy probable que en el futuro se requiera utilizar una representación de color nueva para facilitar la visualización y estudio de algún caso concreto, se ha abstraído la implementación de estos de tal forma que para poder añadirlos al programa simplemente hay que modificar los

nombres incluidos en la lista, los cuales deben ser mapas de color de la librería “matplotlib”.

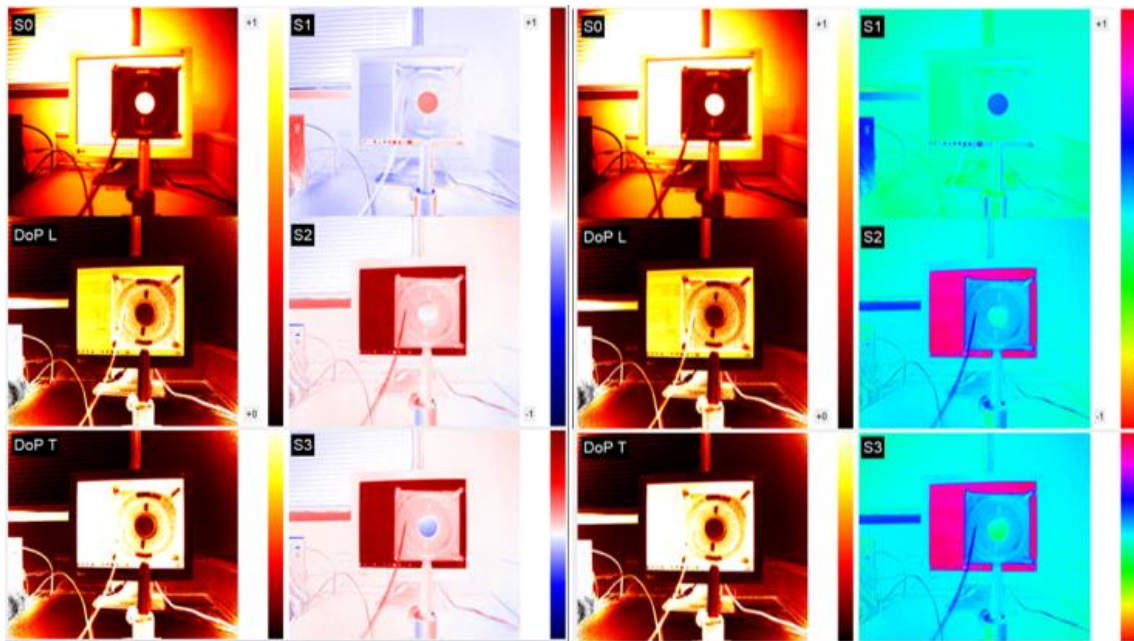


Figura 4.3: Comparación de resultados con filtro “hot” (izquierda) y filtro “hsv” (derecha)

Para visualizar el resultado de estos cálculos por pantalla se ha hecho uso del elemento “canva” nuevamente, el cual también recoge las imágenes desde una cola. Como este elemento necesita realizar las mismas funciones que el anterior, se ha creado como otra instancia de la misma clase. Además, se ha añadido una leyenda del rango de valores al lado de las imágenes para facilitar la comprensión del significado de cada color utilizado.

Debajo de la toma de vídeo, en la esquina inferior izquierda, se encuentran todos los controles del programa disponibles al usuario. Estos se dividen en 3 secciones según su función **[Figura 4.4]**.

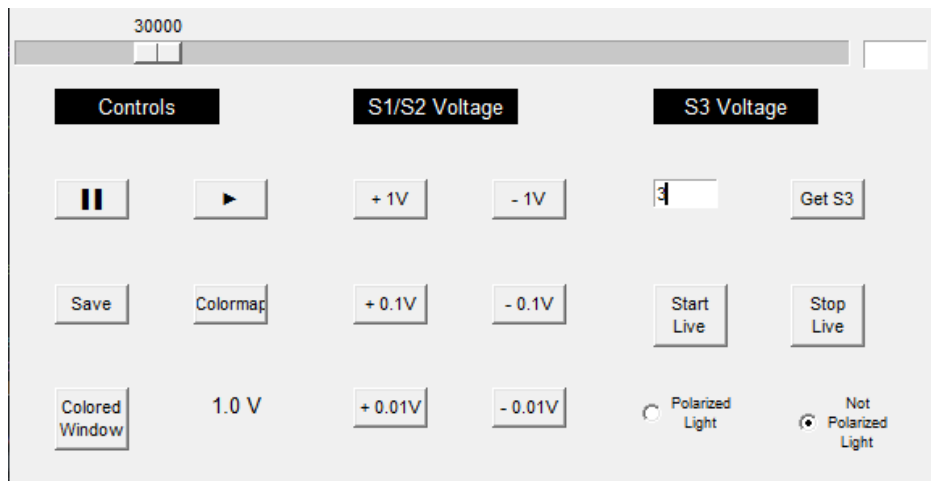


Figura 4.4: Opciones de la sección de controles del programa

Bajo la etiqueta “Controls” se agrupan los controles generales de la toma y visualización del vídeo. Concretamente se disponen de las siguientes opciones:

- **Botón Pause (“||”):** Permite detener la captura de imágenes del vídeo.
- **Botón Play (“▶”):** Permite reanudar la captura de imágenes para el vídeo.
- **Save:** Guarda como imagen PNG el estado actual del programa para su futuro análisis. La imagen incluye el fotograma actual del vídeo monocromático, los cálculos S_0 , S_1 , S_2 , S_3 , ambos DoP y los parámetros y opciones elegidos en la sección de controles.
- **Colormap:** Cambia el mapa de color aplicado durante la visualización de los parámetros S_1 , S_2 y S_3 .
- **Colored Window:** Crea una ventana auxiliar con un color de fondo sólido elegido por el usuario mediante una ventana modal [Figura 4.5]. Esta es útil para ver el efecto de dicho color en cada ángulo de polarización.

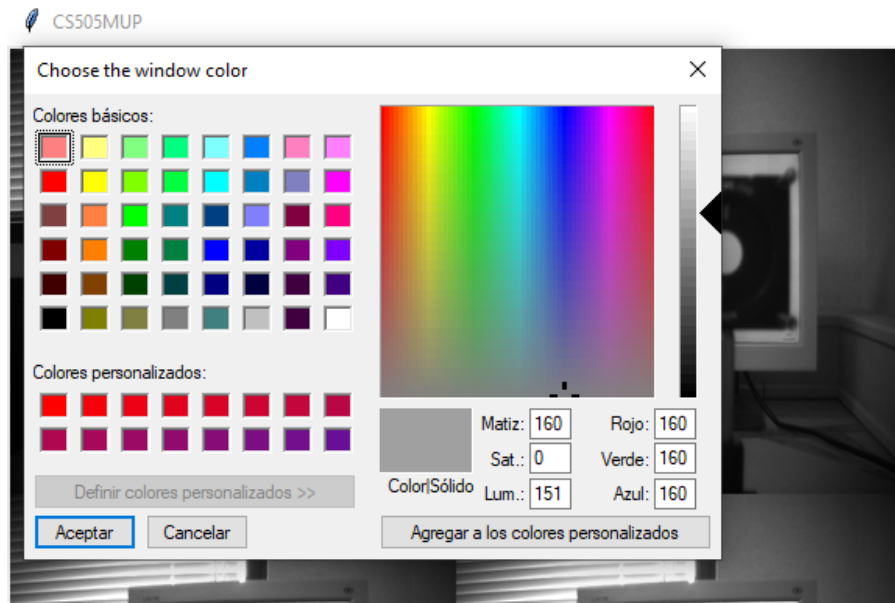


Figura 4.5: Selector de color para la creación de la ventana auxiliar

Bajo la etiqueta “S1/S2 Voltage” se encuentran las opciones que dispone el usuario para modificar las características de la señal eléctrica aplicada por el controlador *KLC101* a la celda de cristal líquido LCR. Concretamente, los valores especificados en esta sección corresponden a el voltaje que se aplica para calcular los valores de S_0 , S_1 , S_2 y DoPL. Para ello disponemos de diferentes botones que nos permiten aumentar o disminuir el voltaje de esta en diferentes cantidades.

En la última sección, agrupados bajo la etiqueta “S3 Voltage”, tenemos las funcionalidades necesarias asociadas al cálculo de los parámetros S_3 y DoP. Antes de poder empezar con los cálculos, deberemos elegir el método a utilizar seleccionando la opción correspondiente:

- **Polarized Light:** Esta opción sirve solamente usando luz polarizada y obtiene el valor de S_3 en valor absoluto. El resultado de S_3 y DoP es un vídeo en tiempo real debido a la rapidez de cálculo del método utilizado.
- **Not Polarized Light:** Esta opción sirve para todo tipo de luz y obtiene el valor de S_3 distinguiendo valores positivos de los negativos. El resultado de S_3 y DoP es una captura instantánea debido al retardo introducido por el método de cálculo utilizado. Requiere el uso de un LCR y una celda de cristal líquido para obtener los valores correctos y es la opción seleccionada por defecto.

Si elegimos la opción “Polarized Light”, podremos ver los resultados inmediatamente. Sin embargo, si elegimos la opción “Not Polarized Light” tendremos que ajustar algunos parámetros adicionales usando los botones correspondientes:

- **Valor Segundo Voltaje:** El usuario puede introducir por teclado el valor deseado del segundo voltaje a aplicar para los cálculos de S_3 y DoP mediante un cuadro de texto.
- **Get S3:** Calcula el valor de los parámetros S_3 y DoP de un instante determinado.
- **Start S3:** Calcula el valor de los parámetros S_3 y DoP de forma continua.
- **Stop S3:** Detiene el cálculo continuo de los parámetros S_3 y DoP.

Además de estos controles, debajo del video capturado hay una barra deslizable para controlar el nivel de exposición, cuyo valor por defecto al iniciar el programa es 30000. Este parámetro indica la cantidad de luz que recibe el sensor de la cámara para formar la imagen, por lo que es importante definir un valor correcto para el nivel de luminosidad presente. Como la cámara utilizada admite un gran rango de valores para definir esta variable, asignar un valor concreto usando solamente la barra deslizable puede ser una tarea ardua. Por este motivo también se ha implementado un cuadro de texto para poder definir este valor por teclado, el cual se actualiza al presionar la tecla “Intro”.

4.1.2 COMBINACIÓN CON LCR

Además del uso de la cámara de polarización y del tratamiento de sus imágenes, se ha desarrollado el programa con la posibilidad de controlar el retardador óptico variable de cristal líquido (LCR). La incorporación de este tipo de dispositivo tiene interés ya que permite modificar la polarización de la luz incidente a estados elípticos o circulares, de modo que en combinación con la cámara permite el análisis polarimétrico completo. Para lograr esto es necesario conectar el LCR al controlador electrónico “Thorlabs KLC101” [Figura 4.6].

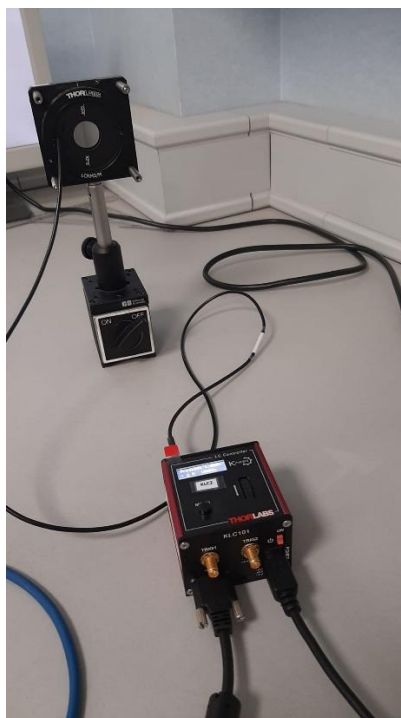


Figura 4.6: Fotografía del controlador *KLC101* conectado al modulador de cristal líquido

Para poder controlar el LCR mediante nuestro ordenador es necesario descargar el SDK desde la sección de “*Software*” de la página del producto en la tienda oficial de la empresa *Thorlabs*. Una vez descargado, debemos conectar los componentes para poder utilizarlos. Para conectar el *KLC101* con el modulador de cristal líquido lo hacemos mediante el puerto “*OUTPUT*”, situado a la derecha del panel trasero. Para integrarlo con el PC se necesita conectar un cable USB 3.0 entre ambos dispositivos, mientras que la toma de corriente se debe conectar al puerto “*POWER*” [Figura 4.7].

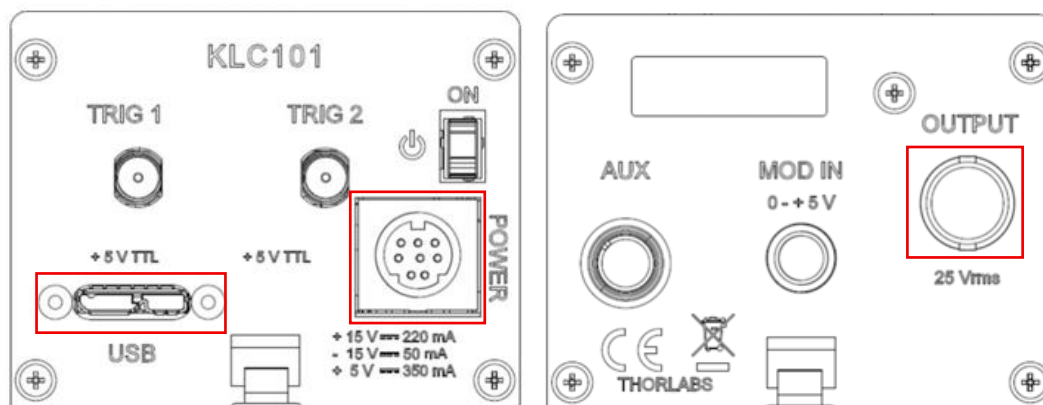


Figura 4.7: Esquema del Panel delantero (izquierda) y trasero (derecha) de Thorlabs KLC101, según [documentación de Thorlabs](#)

Para simplificar el tratamiento del código y sus futuras actualizaciones, se han recogido todas las funciones que controlan el LCR en una misma clase llamada *KLCcontroller()*. Con la misma finalidad se ha abstraído el proceso de apertura y cierre del dispositivo, estando en archivos independientes llamados *initialKLCfor1.py* y *closeKLCfor1.py* respectivamente para ser fácilmente reutilizables en el futuro y conseguir un diseño más modular. Concretamente, estos archivos contienen el código necesario para garantizar un inicio y cierre correctos, tratando los posibles errores que puedan ocurrir durante estos procesos.

El LCR tiene asignado por defecto una onda de 0V de amplitud con 1000HZ de frecuencia al iniciar el programa, pudiendo modificar el valor del voltaje mediante los botones correspondientes bajo la etiqueta “S1/S2 Voltage”. Esta onda es generada durante toda la ejecución del programa para ver los efectos en tiempo real del cristal líquido, tanto en la toma de vídeo monocromático como en la medida de los parámetros de Stokes **[Figura 4.8]**.

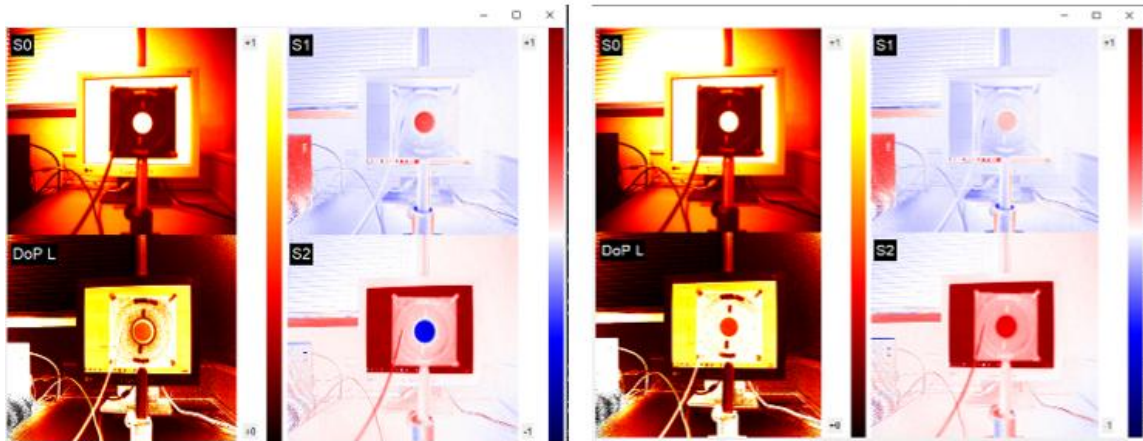


Figura 4.8: Visualización del LCR a 2V (izquierda) y a 6V (derecha).
Se observa como el parámetro S_2 cambia de valores negativos a positivos al cambiar el voltaje

Para llevar un seguimiento de las características de la onda dentro del propio programa, existe una etiqueta indicando el voltaje actual aplicado por el LCR [Figura 4.9]. El valor de esta etiqueta se actualiza dinámicamente cada vez que el usuario modifica la amplitud del voltaje usando los controles de la sección “S1/S2 Voltage”.

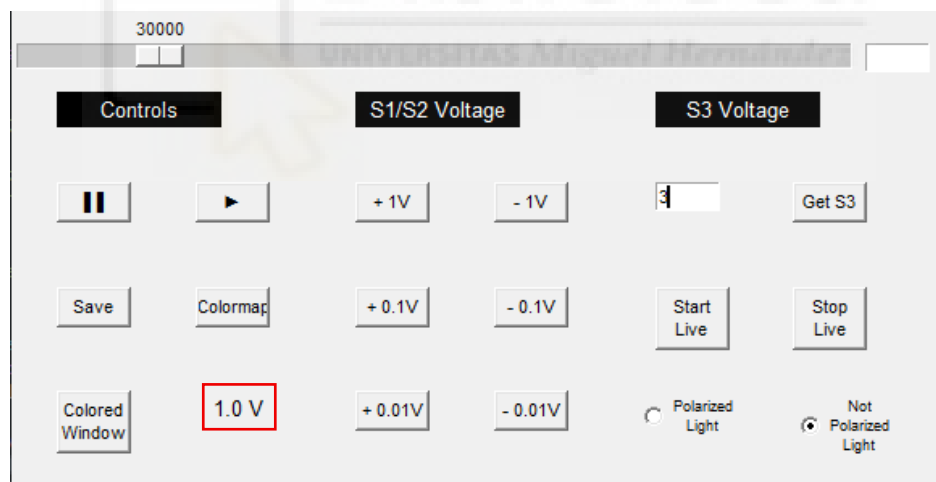


Figura 4.9: Posición de la etiqueta de voltaje en la interfaz

El usuario puede asignar también un segundo valor de amplitud del voltaje en la sección “S3 Voltage” [Figura 4.10] para la obtención de S_3 y el cálculo de DoP. Una vez iniciado el proceso, el LCR alternará entre ambos voltajes a intervalos constantes para recopilar los datos necesarios para el cálculo de DoP (S_0 , S_1 , S_2 y S_3).

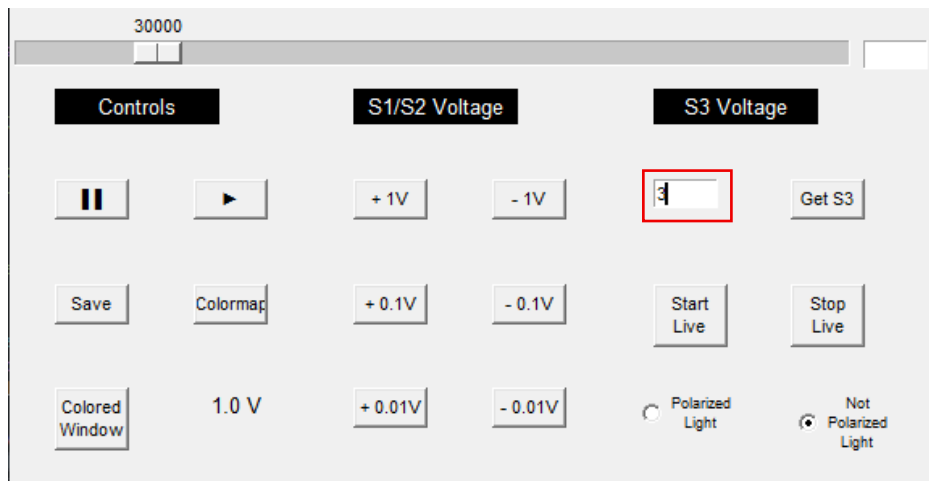


Figura 4.10: Entrada para especificar el voltaje de S_3 en la interfaz

Como el rango de salida del LCR utilizado es de 0-25V, se ha implementado la lógica necesaria para que el usuario nunca pueda asignar un valor no compatible con el dispositivo, indicando las causas del error en un mensaje en los casos necesarios [Figura 4.11].



Figura 4.11: Ejemplos de mensajes de error controlados

4.1.3 COMBINACIÓN CON MONITOR

Este proyecto es especialmente útil para filtrar las diferentes componentes de polarización de la luz y poder así apreciar aspectos de la escena que serían imperceptibles en una imagen estándar de intensidad. Por este motivo se ha utilizado un monitor LCD (*Liquid Crystal Display*) carente del polarizador externo para probar y mostrar el funcionamiento de la cámara. Para comprender esta parte del proyecto, debemos recordar que los monitores LCD tienen dos filtros polarizadores, uno en cada extremo de la capa de cristal líquido, orientados perpendicularmente entre sí. El polarizador de salida es un elemento esencial para

visualizar imágenes, pues bloquea la luz de fondo no deseada, aumenta el contraste y reduce reflejos y brillos ambientales. Al retirar el polarizador externo de la pantalla, ésta toma un aspecto totalmente blanco al perder la capacidad de filtrar la luz emitida **[Figura 4.12]**. Sin embargo, aunque el contenido quede oculto, la pantalla sigue siendo completamente funcional.



Figura 4.12: Monitor sin polarizador utilizado

Para poder operar con mayor facilidad se ha hecho uso de una configuración de monitores dual, siendo uno de ellos el mencionado anteriormente y el otro uno estándar sin manipular. Como ambos se encuentran conectados al mismo *PC*, se puede navegar entre estos usando el ratón y compartir pestañas.

La finalidad del monitor sin polarizador es servir de fuente de iluminación para la cámara de polarización, mientras que la finalidad del segundo es iniciar el programa y analizar los resultados que este nos proporciona. La siguiente figura muestra una fotografía del montaje realizado **[Figura 4.13]**. La cámara de polarización se enfoca hacia el monitor sin polarizador.

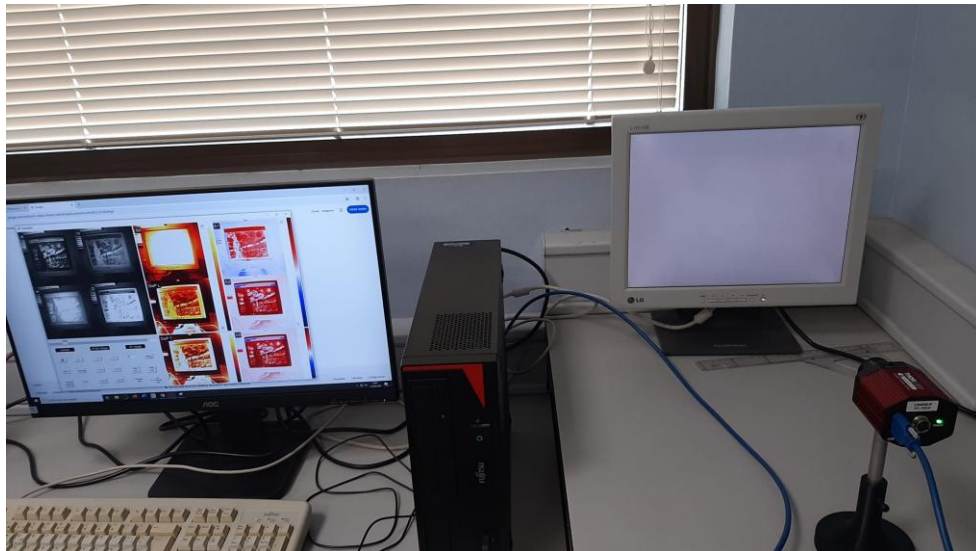


Figura 4.13: Ejemplo de uso conjunto de ambos monitores para capturar el cuadro “Noche Estrellada” de Vincent van Gogh

Como se puede apreciar, la cámara de polarización capta los contenidos de la pantalla sin polarizador y lo muestra de forma monocromática en cada estado de polarización de forma independiente [Figura 4.14]. Así, podemos observar que la imagen se visualiza con intensidades diferentes dependiendo de la orientación del analizador de polarización de modo que las parejas de polarizadores ortogonales (Horizontal/Vertical y Diagonal/Antidiagonal) muestran imágenes complementarias.

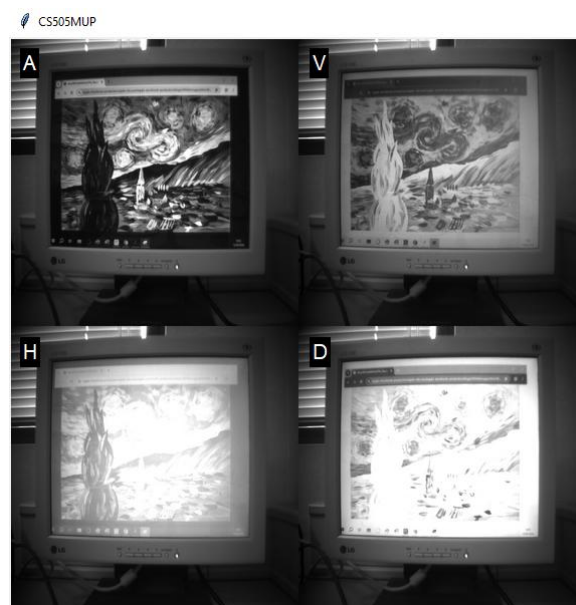


Figura 4.14: Comparación de visualizaciones del cuadro “Noche Estrellada” de Vincent van Gogh

4.1.4 COMBINACIÓN CON BOTÓN FÍSICO

El programa cuenta con todas las funcionalidades necesarias para controlar la toma de vídeo, sin embargo, esto requiere que el usuario se encuentre cerca de su PC. Aunque la cámara esté conectada mediante cable USB, sigue siendo un dispositivo portátil, por lo que la indisponibilidad de alejarse demasiado de la pantalla para tratar las imágenes capturadas puede ser una molestia.

Para solucionar esto, se ha implementado un botón físico auxiliar **[Figura 4.15, izquierda]** que nos permite pausar o continuar la toma de vídeo. La idea es que el usuario pueda tomar la cámara con una mano y, una vez estemos enfocando al área de interés, pausar el vídeo con el botón para volver al ordenador a analizar la captura, dándole así una mayor flexibilidad y libertad de maniobra al usuario.

El botón cuenta con una base de plástico realizada con una impresora 3D que le permite ser utilizado de forma más ergonómica **[Figura 4.15, derecha]**. Esta base cuenta con un soporte en el cual se puede acomodar la cámara, combinando ambos como si fuera un accesorio de esta.

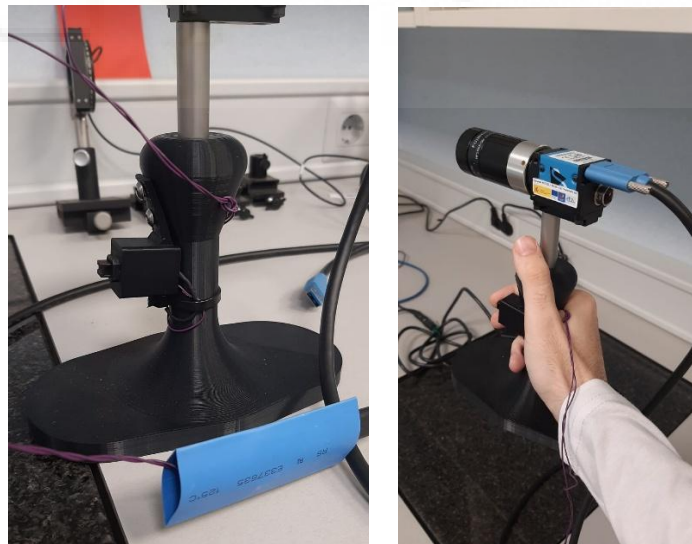


Figura 4.15: Uso de la cámara con la base usando un Arduino (envoltura azul)

Para poder integrar el botón con el programa, este cuenta con un Arduino que debe ser conectado al PC mediante un cable USB. Al iniciarse el programa, se utiliza una función para buscar si el Arduino ha sido conectado en alguno de los puertos del dispositivo. Si ha sido conectado, se crea un hilo de ejecución adicional encargado

de esperar a recibir las señales que este envía. Cada vez que el botón es pulsado envía la señal “*TRIGGER*” al programa, la cual es capturada para dar paso a la función que cambia el estado del vídeo de *Play* a *Pause* o viceversa, dependiendo del estado actual.

Si no se detecta que el dispositivo está conectado, este hilo adicional no se llega a crear y se continúa con la ejecución normal del programa. El uso de este componente es completamente opcional, no sustituye ni añade ninguna funcionalidad adicional al programa, por lo que actúa solo como complemento.



4.2 SOFTWARE PARA CALIBRACIÓN DEL LCR

El uso correcto del LCR cuando se requiere medir el parámetro S_3 implica una correcta calibración del retardo introducido por el dispositivo, que debe sintonizarse a un retardo de un cuarto de la longitud de onda de la luz. La técnica clásica para la calibración de retardadores implica situarlo entre dos polarizadores paralelos o cruzados, con una orientación de 45° relativa al eje director del LCR. En esta situación, la intensidad del haz de luz modulado sigue una función sinusoidal o cosenoidal al cuadrado con respecto a la mitad del retardo. Por ello, se ha programado un método de calibración del LCR, suponiendo que el dispositivo se ilumina con luz polarizada y se detecta con la cámara de polarización. Utilizada en modo “*Quadview*”, la cámara ofrece directamente (en dos de sus cuadrantes) las imágenes de la transmitancia del LCR para polarizadores paralelos y cruzados.

El programa analiza estas imágenes cuando se aplica a la celda una señal cuadrada de voltaje de frecuencia determinada, generada por el controlador KLC, para luego mostrar las curvas de transmitancia versus voltaje de forma gráfica. Al iniciar el programa nos encontramos con la toma de vídeo de la cámara junto a la zona de control [Figura 4.16].

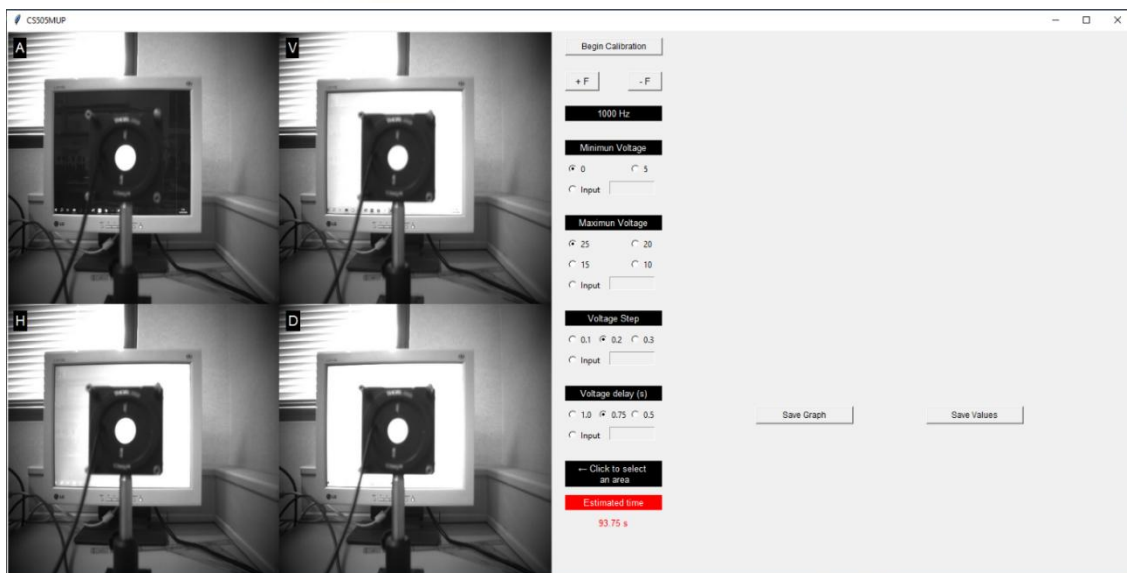


Figura 4.16: Ventana principal del programa

A la izquierda, el programa muestra la toma de video en blanco y negro separada en los cuatro ángulos de polarización lineales. Los cálculos y los métodos utilizados son los mismos que los usados en el programa que controla la cámara de polarización monocromática.

Como solamente nos interesa la zona de la imagen que ha atravesado el área activa del dispositivo de cristal líquido, en el vídeo se ha añadido interacción con el ratón para seleccionar el área de interés. Al hacer *click* izquierdo sobre el vídeo, se crea un círculo verde con posiciones homólogas en todas las imágenes indicando la zona seleccionada [Figura 4.17]. Solo puede existir un único círculo en cada toma de vídeo, por lo que volver a pulsar el ratón tiene como resultado su desplazamiento. Para ajustar el tamaño de la selección se puede hacer uso de la rueda del ratón, deslizando hacia arriba para aumentar el tamaño y hacia abajo para disminuirlo.

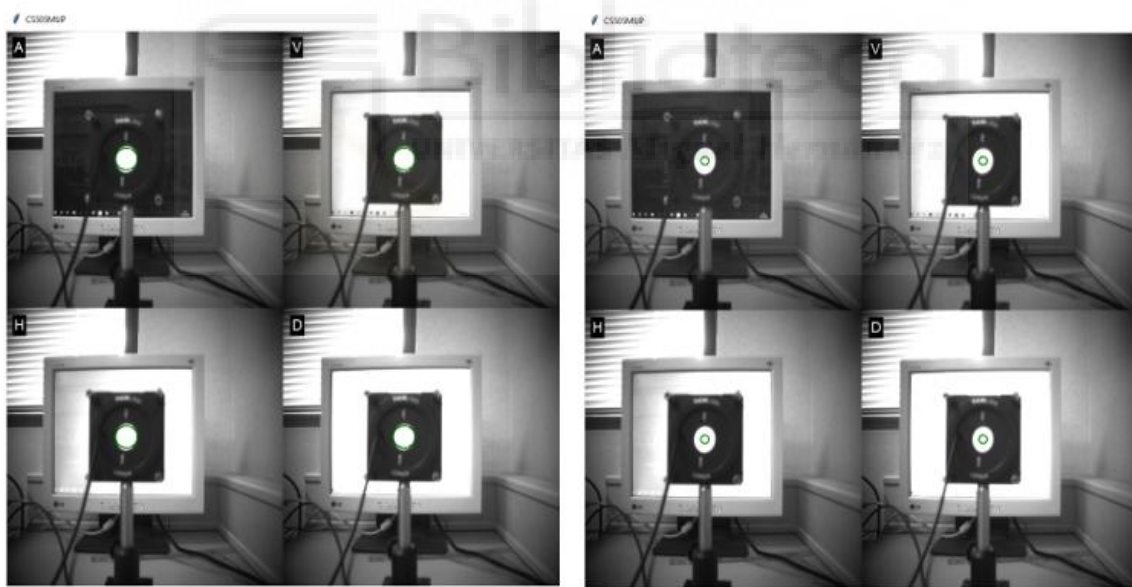


Figura 4.17: Representación visual del área de calibración seleccionada a diferentes tamaños

Una vez elegida la porción de la imagen a analizar, podemos comenzar el proceso dándole al botón “Begin Calibration”. Una vez hecho esto en la zona de la derecha del programa aparecerá una barra de progreso y, una vez completada, se mostrará un gráfico mostrando la cantidad de luz que atraviesa el sistema en el área seleccionada [Figura 4.18].



Figura 4.18: Muestra gráfica de resultados

Este tiene representado en el eje X los valores del voltaje aplicado al LCR, mientras que en el eje Y se representa la media de intensidad del área seleccionada en la imagen. Para diferenciar las capturas de cada cuadrante se ha añadido una leyenda indicando la polarización lineal correspondiente: Diagonal (D), Horizontal (H), Vertical (V) o Antidiagonal (A). Si el conjunto cristal líquido junto al cuadrante polarizador de la cámara dejan pasar la luz, se registrarán valores elevados, mientras que se registrarán valores bajos si la luz no es capaz de atravesar el conjunto.

Para formar el gráfico se ha hecho uso de la librería “*matplotlib*” y del elemento “*canva*” para mostrarlo por pantalla. De forma nativa los gráficos creados por esta librería no son compatibles con los “*canva*” de la librería “*tkinter*”. Para solucionar esto se ha hecho uso de la función de conversión “*FigureCanvasTkAgg*” importada desde “*matplotlib.backends.backend_tkagg*”, la cual recibe como parámetro el gráfico para devolver un equivalente compatible.

En la zona central del programa se encuentran las opciones y configuración. La configuración necesaria para la ejecución suele comenzar con valores predefinidos, con el fin de que el programa se encuentre listo para ser usado rápidamente. No obstante, antes de poder empezar el análisis el usuario debe seleccionar la zona deseada del vídeo.

Para empezar, podemos ajustar la frecuencia de la señal cuadrada generada por el KLC de forma rápida pulsando en “+ F” o “- F”. El valor predefinido de la misma es de 1000HZ.

Bajo la etiqueta “Minimun Voltage” tenemos opciones para cambiar el valor del voltaje pico-pico inicial del análisis. Concretamente contamos con dos opciones predefinidas para una configuración rápida y otra para asignar un valor personalizado:

- **0:** El valor inicial de la amplitud es 0V. Marcada por defecto
- **5:** El valor inicial de la amplitud es 5V.
- **Input:** Utiliza el valor introducido en el cuadro de texto como valor inicial. Dicho valor debe ser un número positivo dentro del rango de voltaje del KLC. Si no se cumplen estas condiciones aparecerá un mensaje de error.

A continuación, englobadas en la sección “Maximun Voltage”, podemos especificar cual es el voltaje máximo que alcanzará la señal durante el análisis. Una vez alcanzado este valor finaliza la recogida de resultados y se procede a formas y mostrar el gráfico. Las opciones disponibles son:

- **25:** El valor máximo de la amplitud es 25V. Marcada por defecto.
- **20:** El valor máximo de la amplitud es 20V.
- **15:** El valor máximo de la amplitud es 15V.
- **10:** El valor máximo de la amplitud es 10V.
- **Input:** Utiliza el valor introducido en el cuadro de texto como valor final. Este debe ser un número positivo dentro del rango del KLC y superior al valor inicial (especificado en “Minimun Voltage”). Si dichas condiciones no se cumplen se informará al usuario con un mensaje de error.

En el apartado “Voltage Step” podemos especificar en cuántos voltios variará el salto de la amplitud de la señal en cada recogida de datos. Cuanto menor es el paso en voltios, más datos son recogidos para elaborar el resultado. Las opciones son:

- **0.1:** Se recogerán datos cada vez que la amplitud varíe 0.1V. Marcada por defecto.

- **0.2:** Se recogerán datos cada vez que la amplitud varíe 0.2V.
- **0.3:** Se recogerán datos cada vez que la amplitud varíe 0.3V.
- **Input:** Se analizarán los datos cada vez que la amplitud varíe en la cantidad especificada. El valor introducido debe ser un número positivo mayor que 0, si no se mostrará el mensaje de error correspondiente.

El apartado “Voltage delay (s)” permite especificar el intervalo de tiempo (en segundos) en que se mantiene el voltaje aplicado durante cada paso. La celda de cristal líquido tarda un tiempo en responder al voltaje aplicado por el KLC, por lo que se recomienda usar valores de entre 0.75 y 1 segundo para análisis detallados. Sin embargo, esto aumenta bastante el tiempo requerido, por lo que en estimaciones rápidas dónde solo nos interesa obtener la forma general de las curvas de transmitancia podemos usar un tiempo más reducido.

Si decidimos dejar menos de 0.75 segundos entre muestras hay que tener en cuenta que es muy probable recoger valores erróneos o duplicados, resultando en una gráfica con menos detalle y con escalones [Figura 4.19]. Para facilitar este tipo de decisiones se ha añadido una etiqueta con una estimación del tiempo requerido para la obtención de resultados.

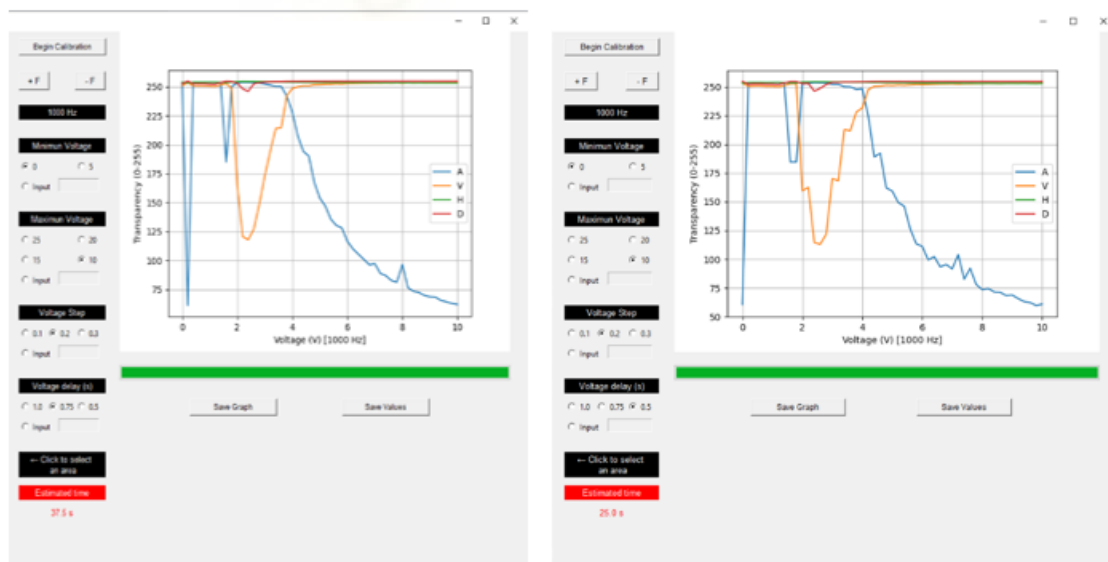


Figura 4.19: Grafico detallado de 37.5s (izquierda) y gráfico rápido de 25s (derecha)

Las opciones englobadas en esta sección son:

- **1.0:** El tiempo entre pasos se establece a 1 segundo.
- **0.75:** El tiempo entre pasos se establece a 0.75 segundos. Marcada por defecto.
- **0.5:** El tiempo entre pasos se establece a 0.5 segundos.
- **Input:** El tiempo entre pasos se establece al valor introducido por el usuario. Este debe ser un número positivo mayor que 0, apareciendo un mensaje de error si no se cumple estas condiciones.

The figure displays three identical calibration interface panels side-by-side, each showing a different error state. The interface includes a 'Begin Calibration' button at the top, followed by '+ F' and '- F' buttons. Below these are three main sections: '1000 Hz', 'Minimum Voltage', and 'Maximum Voltage'. Each section has radio buttons for '0', '5', '25', '20', '15', and '10', and an 'Input' field. The 'Voltage Step' section has radio buttons for '0.1', '0.2', and '0.3', and an 'Input' field. The 'Voltage delay (s)' section has radio buttons for '1.0', '0.75', and '0.5', and an 'Input' field. At the bottom of each panel is a red 'Estimated time' button showing '37.5 s' and a 'Click to select an area' button. The error messages are as follows:

- Panel 1 (Left): 'Minimum is not a number!!!' (red text).
- Panel 2 (Middle): 'Maximum must be greater than Minimum!!!' (red text).
- Panel 3 (Right): 'Step value must be above 0!!!' (red text).

Figura 4.20: Ejemplos de errores controlados

Una vez obtenidos los resultados con los criterios y especificaciones introducidas, podemos guardarlos para consultas posteriores. El comando “Save Graph” permite guardar el gráfico como una imagen PNG. Si en lugar de una curva gráfica queremos el detalle de los valores experimentales obtenidos durante el análisis podemos hacer uso de la opción “Save values”, la cual nos crea un archivo CSV con los valores obtenidos redondeados al cuarto decimal.

4.3 SOFTWARE PARA CALIBRACIÓN CON MONITOR

Para evitar saturar accidentalmente las imágenes capturadas por la cámara, se ha creado un software que nos permite analizar el nivel de intensidad captado en un rango de colores para detectar errores y solucionarlos ajustando el nivel de exposición de la cámara. Para ello el programa crea una ventana auxiliar con fondo de color sólido que variará según los parámetros especificados, analizando la intensidad capturada por la cámara en cada color, para luego mostrar los resultados mediante un gráfico.

Al iniciar el programa se crea una ventana mostrándonos la toma de vídeo capturada por la cámara, separada en los ángulos de polarización lineales junto a una zona de control con los ajustes del programa **[Figura 4.21]**.

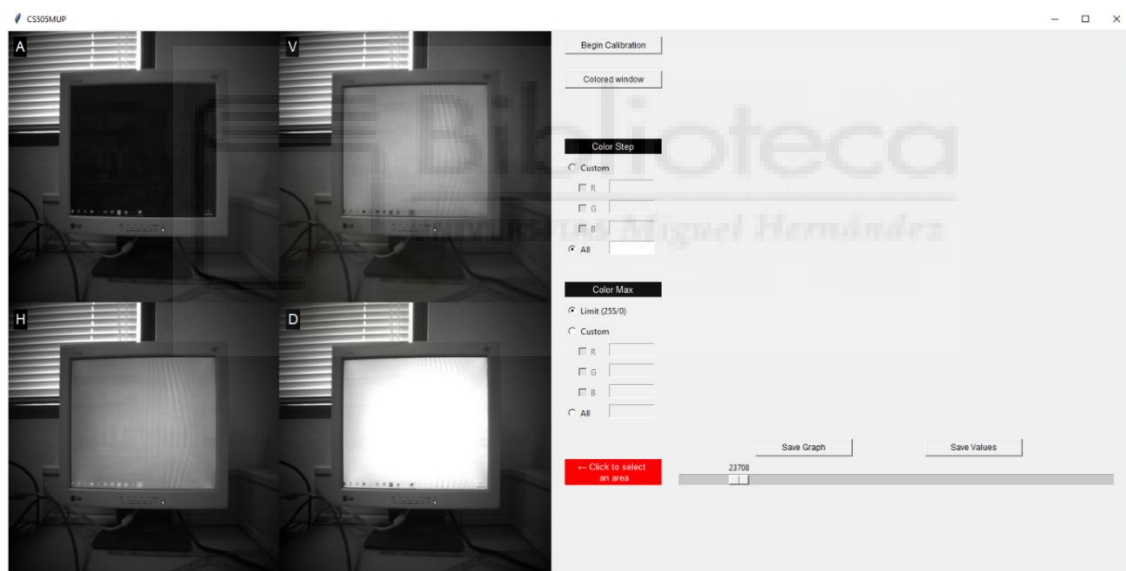


Figura 4.21: Ventana principal del programa

El vídeo se calcula de forma análoga a los programas anteriores, usando un hilo de ejecución independiente para procesar las imágenes y usando un elemento “*canva*” para mostrarlo por pantalla. Al igual que en la calibración del LCR, antes de comenzar el proceso deberemos elegir la zona a analizar usando el ratón. Para mejorar la consistencia y el aprendizaje de los softwares se ha utilizado el mismo esquema de controles para ambos, eligiendo el área deseada en todos los cuadrantes haciendo *click* izquierdo y aumentando o disminuyendo su tamaño con la rueda del ratón. Adicionalmente, podemos ajustar el nivel de luminosidad de la

imagen usando la barra deslizante que hay en la zona inferior, la cual controla el nivel de exposición de la cámara.

Una vez elegida el área necesitamos crear una ventana de color utilizando el botón “Colored Window” para empezar el análisis. Una vez comenzado, se mostrará en la zona de la derecha una barra de progreso indicando el tiempo restante y, una vez esta llegue al máximo, aparecerán los resultados representados en un gráfico de barras [Figura 4.22].

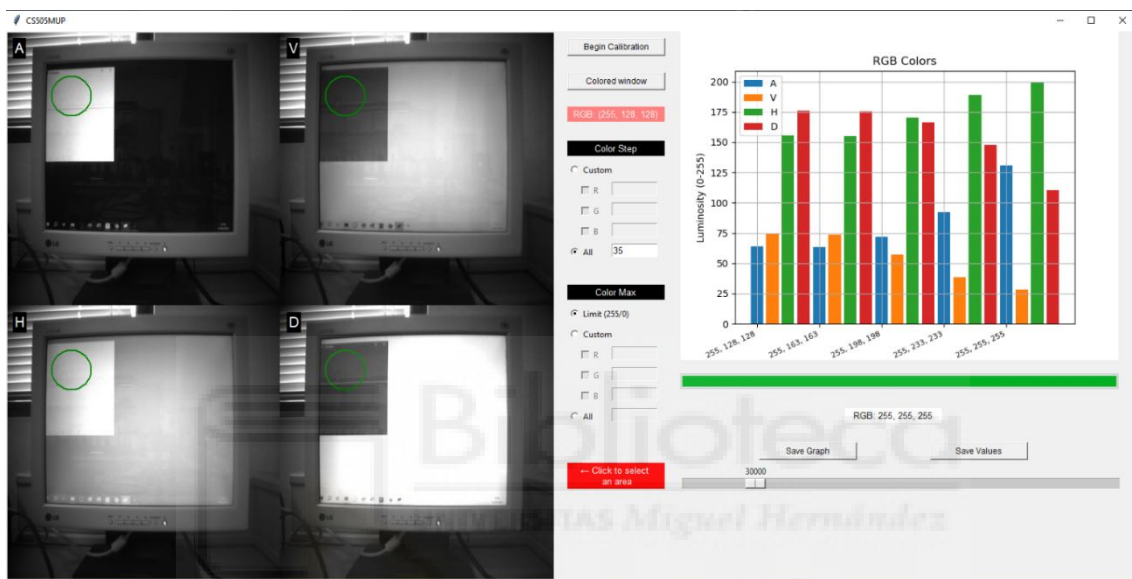


Figura 4.22: Muestra gráfica de resultados

Para la creación del gráfico se ha utilizado una vez más la librería “*matplotlib*”. Para mostrar la información de los colores utilizados se utiliza el eje X, mientras que los valores de intensidad capturados se representan en el eje Y. Como para cada color utilizado se corresponden cuatro valores de intensidad, se ha decidido aplicar un pequeño desplazamiento a las barras para que no queden superpuestas, facilitando así la visualización. Además, se ha incluido una pequeña leyenda para diferenciar qué color corresponde a cada imagen. Para mostrarlo por pantalla se ha vuelto a utilizar un componente “*canva*” utilizando la función “*FigureCanvasTkagg()*” por compatibilidad con la interfaz gráfica de “*tkinter*”.

En la zona central tenemos agrupadas todas las opciones pertinentes a la formación del gráfico. El botón “Begin calibration” comienza el análisis de las regiones para cada cambio de color de la ventana emergente. Si aún no se ha

especificado la zona o creado la ventana, no se realiza ningún cálculo y aparece un error por pantalla informando al usuario.

Al pulsar en “Colored Window” podemos crear una ventana auxiliar con el fondo sólido que será objetivo del análisis utilizando una ventana modal con el mismo formato que en el programa para la cámara monocromática [Figura 4.5]. La existencia de esta ventana es un requisito para obtener resultados.

A continuación, se encuentran las opciones necesarias para modificar el color utilizado por la ventana a analizar. Bajo la etiqueta “Color step” tenemos englobadas las opciones correspondientes al aumento y disminución de los componentes RGB. Las opciones disponibles son las siguientes:

- **Custom:** Permite al usuario elegir el cambio de los componentes RGB por separado. Para ello se seleccionan los componentes a modificar marcando las casillas correspondientes:
 - **R:** Indica el cambio del componente Rojo en cada iteración.
 - **G:** Indica el cambio del componente Verde en cada iteración.
 - **B:** Indica el cambio del componente Azul en cada iteración.
- **All:** Aplica el cambio introducido a los tres componentes de color al mismo tiempo. Marcada por defecto.

Los valores positivos introducidos aumentan la intensidad del componente, mientras que los negativos lo disminuyen. Si no se especifica ningún valor se conserva el del color original.

En la sección “Color Max” tenemos las opciones necesarias para definir el valor máximo de los componentes RGB durante los cambios. Las opciones existentes son las siguientes:

- **Limit (0/255):** Los cambios se aplican hasta llegar al límite correspondiente de cada componente, ya sea 0 o 255. Marcada por defecto.
- **Custom:** Permite elegir el límite de cada componente RGB de forma independiente. Para ello es necesario marcar en las casillas correspondientes:

- **R:** Especifica el máximo del componente Rojo.
- **G:** Especifica el máximo del componente Verde.
- **B:** Especifica el máximo del componente Azul.
- **All:** Aplica un máximo personalizado a todos los componentes.

Si el usuario no especifica valor para los componentes se llega al límite por defecto (0/255). Si el valor introducido por el usuario no concuerda con los cambios aplicados al componente, este se ignora y se llega al límite por defecto.

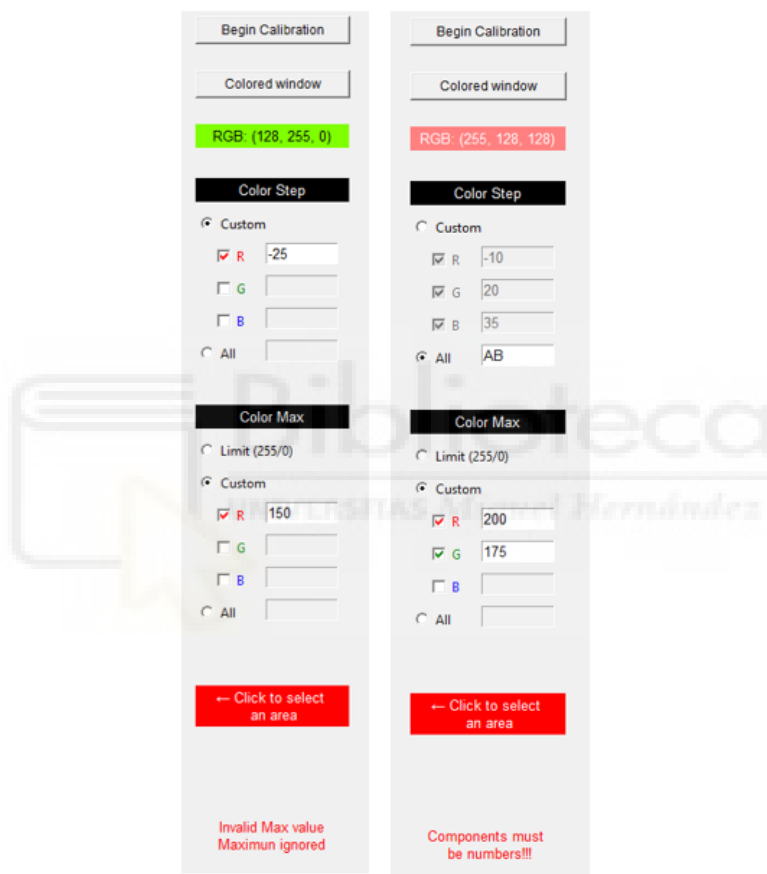


Figura 4.23: Ejemplos de errores controlados

Una vez formado el gráfico tenemos varias opciones para guardarlo y poder consultarlo en otro momento. Si nos interesa ver los resultados de forma visual con el botón “Save Graph” podemos obtener una imagen en formato PNG de la totalidad del gráfico. Si lo que nos interesa es conocer los valores obtenidos durante el análisis con un mayor nivel de precisión podemos usar la opción “Save values”, con la cual obtenemos un archivo CSV con los valores aproximados al cuarto decimal para cada color analizado.

4.4 SOFTWARE PARA CÁMARA RGB DE POLARIZACIÓN

4.4.1 FUNCIONALIDAD BÁSICA

Adicionalmente al programa para el análisis de imágenes en los diferentes ángulos de polarización de forma monocromática, se ha desarrollado otro complementario que hace uso de una cámara de polarización en color RGB con el objetivo de estudiar imágenes a color. Concretamente se ha utilizado la cámara comercial “*ImagingSource DYK 33UX250*”, la cual debe conectarse al ordenador mediante un cable USB 3.0.

Al iniciar el programa aparece una ventana mostrándonos la toma de vídeo captada por la cámara en tiempo real a la izquierda, junto a los controles y opciones disponibles desplegadas en el lateral opuesto [Figura 4.24].

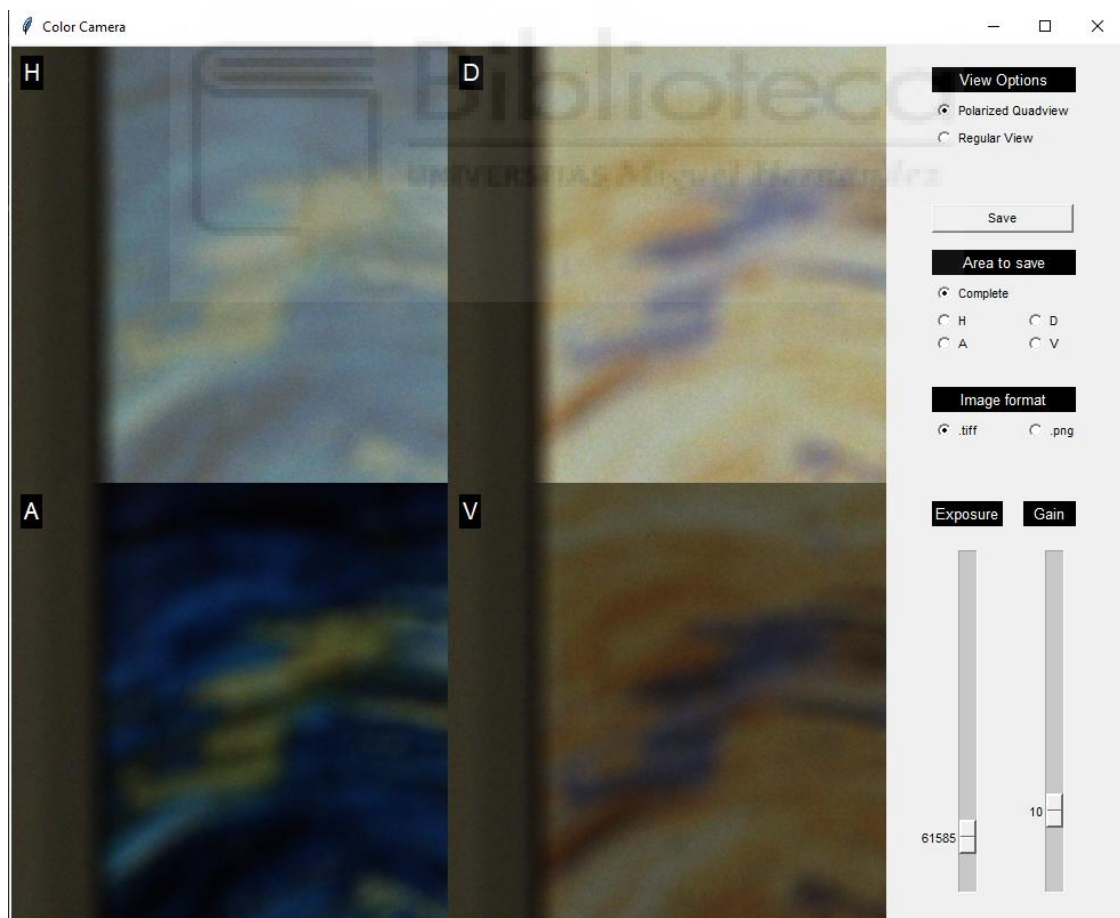


Figura 4.24: Ventana principal del programa

Por defecto el vídeo muestra las imágenes capturadas separadas en los cuatro grados de polarización lineal: Vertical (V), Horizontal (H), Diagonal (D) y Antidiagonal (A). Aunque esta división de los diferentes canales ya se hizo en la cámara monocromática, implementarla en formato RGB presenta un nivel de complejidad adicional.

Esto es debido a que los píxeles de la cámara llevan también incorporados un filtro de Bayer para distinguir los canales de color. Con el objetivo de obtener la información correspondiente a cada píxel, además de tener que reorganizar cada imagen separando los canales de polarización, hay que descomponer el mosaico del filtro ("*debayering*") para obtener el color de las imágenes.

Para que el usuario pueda seguir teniendo el control sobre el sistema mientras se realiza el procesamiento de las imágenes se ha decidido realizar estas operaciones en un hilo de ejecución diferente al principal.

Al iniciar el programa, una vez conectada la cámara, se crea el hilo cuya función es la adquisición de fotogramas y su tratamiento. Para comenzar, se guarda la información cruda de la imagen capturada en una variable. A continuación, se reordenan los píxeles de esta para separar los canales de polarización en zonas bien diferenciadas. Una vez hecho todo esto, necesitamos deshacer el mosaico de Bayer para obtener los colores originales de la imagen en RGB. Como cada píxel del sensor solamente registra un color (R, G o B), este algoritmo estima los dos colores restantes basándose en los píxeles vecinos.

Para ello se ha decidido utilizar las funciones de "*demosaiicing*" incluidas en la librería de tratamiento de imágenes "*OpenCV*". Concretamente, la función "*cvtColor*" de la misma librería nos permite realizar una gran variedad de estas conversiones, incluidos procesos de "*debayering*", dependiendo del código pasado como parámetro. En nuestro caso, como el filtro utilizado por la cámara es un Bayer BGGR [Figura 2.10], pasamos como argumentos la matriz con la información de la imagen y el código "*COLOR_BAYER_RG2RGB*" para realizar el algoritmo de interpolación cromática necesario para convertir la imagen en RGB [Figura 4.25].

Figura 4.25: [Documentación de OpenCV](#)
sobre el código de conversión Bayer BGGR

Además de la visualización del vídeo en los canales de polarización lineales, también existe la opción de obtener un vídeo corriente como el que tomaría una cámara común. Esto es debido a que se puede calcular a partir de los ángulos de polarización, concretamente es equivalente al parámetro de Stokes S_0 [Ecuación 1.1].

Una vez obtenido el resultado, simplemente tenemos que aplicar el algoritmo de “*debayering*” usando las funciones de “*OpenCV*” para obtener la imagen carente de ángulo de polarización [Figura 4.26].

Una vez obtenida la imagen a mostrar, la guardamos en una cola compartida con el hilo principal. Este se encarga de recoger el resultado, eliminarlo de la cola y mostrarlo por pantalla usando el elemento “*canva*”.



Figura 4.26: Captura del monitor sin
ángulos de polarización

Al lado de la toma de vídeo nos encontramos con los controles básicos del programa, agrupados bajo etiquetas según su funcionalidad.

En la primera sección, bajo la etiqueta “View Options”, tenemos de un botón radial de selección única para escoger el modo de visualización del vídeo. Las opciones para elegir son las siguientes:

- **Polarized Quadview:** El programa procesa las imágenes separando la información de cada estado de polarización lineal, resultando en un vídeo cuádruple. Opción marcada por defecto.
- **Regular View:** El programa calcula el valor del parámetro S_0 a partir de las componentes diagonal y antidiagonal captadas por la cámara. El resultado es una única toma de vídeo sin información sobre el estado de polarización de la escena, equivalente a una capturada con una cámara RGB estándar.

Debajo tenemos un botón que nos permite guardar capturas de las imágenes del vídeo que nos interesen tanto en formato TIFF para su futuro tratamiento como en formato PNG para visualización. Cuando se muestran múltiples tomas de vídeo, podemos elegir guardar la imagen de ángulos específicos eligiendo la opción correspondiente en el apartado “Area to Save”. Las opciones disponibles son:

- **Complete:** Contiene información de las cuatro componentes lineales de polarización, indicadas mediante los respectivos ángulos del eje de transmisión del analizador. Marcada por defecto.
- **H:** Contiene únicamente la imagen de la componente de polarización horizontal.
- **D:** Contiene únicamente la imagen de la componente de polarización diagonal.
- **A:** Contiene únicamente la imagen de la componente de polarización antidiagonal.
- **V:** Contiene únicamente la imagen de la componente de polarización vertical.

Si la forma de visualizar el vídeo es en *Regular View* las opciones del área a guardar no aparecerán como marcables y por defecto se guardará la imagen completa [Figura 4.27].



Figura 4.27: Opciones en modo polarizado (izquierda) y normal (derecha)

Para cambiar el formato de la imagen guardada se usa la opción elegida bajo la etiqueta “*Image Format*”. Las opciones presentadas son:

- **.tiff:** Las imágenes se guardarán en formato TIFF. Marcada por defecto.
- **.png:** Las imágenes se guardarán en formato PNG.

Además, el programa cuenta con algunas opciones para controlar parámetros de visualización de la cámara y ajustar la toma de vídeo en directo. Concretamente el software cuenta con dos barras deslizantes para adaptar la captura de fotogramas al nivel de luminosidad presente durante la toma. Los parámetros ajustables mediante dichas barras son:

- **Exposure:** Modifica el nivel de exposición de la cámara. Esto indica la cantidad de luz que recibe el sensor para crear la imagen.
- **Gain:** Modifica el nivel de ganancia de la cámara. Esto permite modificar la señal electrónica del sensor y cambiar el nivel de luminosidad de la imagen a falta de luz natural. A niveles altos añade ruido y pérdidas de calidad.

4.4.2 COMBINACIÓN CON MONITOR

Al igual que con la cámara monocromática, se ha usado la misma organización de pantallas duales, teniendo una común para el despliegue y uso del programa y otra carente de polarizador para capturar vídeos e imágenes a color con la cámara de polarización policromática [Figura 4.28].

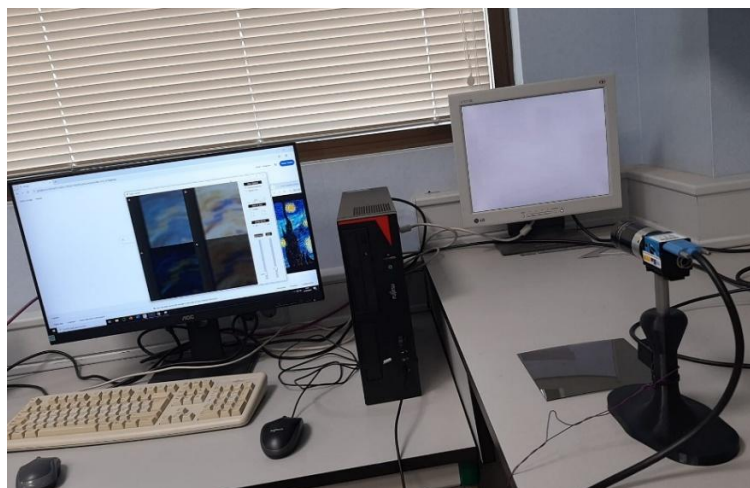


Figura 4.28: Ejemplo de uso de ambos monitores para capturar el cuadro “Noche Estrellada” de Vincent van Gogh

Como se puede observar, los colores de la imagen y la intensidad de estos difieren en cada ángulo de polarización. Si usamos el modo de visualización de intensidad, equivalente a una cámara convencional, no podremos visualizar el contenido del monitor.

El modo de visualización de intensidad muestra el parámetro de Stokes S_0 y resulta muy útil para definir los parámetros adecuados de exposición y ganancia de la cámara, al poder comparar las características de las imágenes captadas con su equivalente real.

El ajuste del nivel de exposición se puede realizar fácilmente comparando el nivel de luminosidad captado con la cámara con el nivel deseado. Para el nivel de ganancia, hay que comprobar que la imagen obtenida sea lo más nítida posible, ya que si se emplean valores mal ajustados a la luz ambiente se distorsiona la imagen con ruido [Figura 4.29].

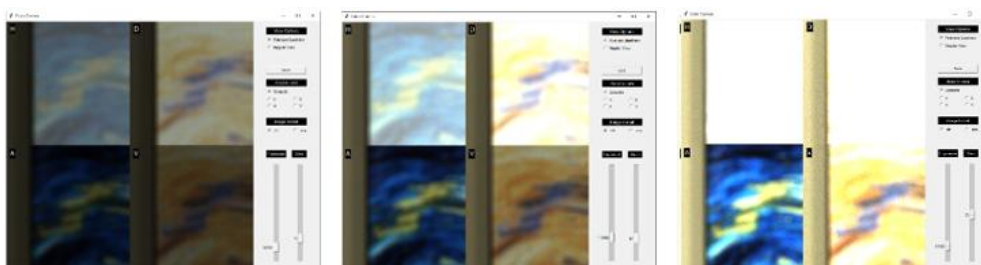


Figura 4.29: Capturas de un fragmento del cuadro “Noche Estrellada” de Vincent van Gogh a diferentes valores de exposición y ganancia

5. CONCLUSIONES Y TRABAJO FUTURO

5.1 CONCLUSIONES

Durante el desarrollo de este proyecto se ha conseguido diseñar y producir dos softwares diferentes para el control de dos cámaras de polarización diferentes (una monocromática y otra en color) y el análisis de sus resultados en tiempo real de forma satisfactoria.

El software para la cámara monocromática recoge la información de los ángulos de polarización y de los parámetros de Stokes S_0 , S_1 y S_2 constantemente para formar resultados con forma de video en tiempo real. Aunque los parámetros S_3 y DoP no se han podido llegar a calcular a esta velocidad para todo tipo de luz debido al retardo introducido por el cambio de fase del LCR, se ha implementado una alternativa que solo funciona con luz polarizada que sí es capaz de proporcionar resultados en forma de vídeo con algunas características especiales (resultados en valor absoluto), dejando al usuario elegir el método que mejor se adapte a cada caso. Además, se ha conseguido centralizar satisfactoriamente el control de todos los componentes necesarios para la captura de estos datos.

Para mejorar la experiencia de uso de este programa se han creado tanto elementos auxiliares software como hardware. Estos son los programas para calibración del LCR para ayudar a encontrar los valores correctos para el cálculo de S_3 , el programa para calibrar el brillo usando el monitor para no saturar los resultados y el botón físico que mejora la operabilidad de la cámara permitiéndonos un control básico sobre la captura de vídeo de forma remota.

También se ha logrado cumplir los objetivos relacionados con el uso de la cámara de polarización a color RGB, permitiendo cambiar entre una toma de vídeo común y uno polarizado mediante controles integrados dentro del propio programa.

5.2 TRABAJO FUTURO

Aunque el programa cumple la funcionalidad esperada, aún tiene aspectos mejorables en posibles futuras actualizaciones o versiones. El que más margen de mejora tiene es la interfaz gráfica, la cual no fue una prioridad del desarrollo.

Como las imágenes de polarización son usadas en ámbitos muy diversos, como la biomedicina o la fotoelasticidad, es posible que en el futuro se requieran nuevas funcionalidades de análisis de datos o nuevas formas de visualización de resultados.

Además, a medida que pase el tiempo puede ser apropiado usar cámaras diferentes para la captura de imágenes. Ahora mismo los softwares solamente son compatibles con las cámaras de los fabricantes presentados en la memoria (*Thorlabs* y *ImagingSource*) al solamente hacer uso de estos SDK, por lo que hay una alta probabilidad de que una actualización de compatibilidad sea necesaria más adelante.



6. BIBLIOGRAFÍA

- [REF 1] Nabadda, Esther (2024). *Liquid-Crystal modulators in optical phaseshifting interferometry and polarimetric imaging*. [Tesis doctoral, Universidad Miguel Hernández]. Repositorio RediUMH. <https://hdl.handle.net/11000/35752>
- [REF 2] A. Vargas, R. Donoso, M. Ramírez, J. Carrión, M. M. Sánchez-López and I. Moreno (2013). *Liquid crystal retarder spectral retardance characterization based on a Cauchy dispersion relation and a voltage transfer function*. (*Opt. Rev.* 20, 378–384). <https://doi.org/10.1007/s10043-013-0068-4>
- [REF 3] IDS (25 de agosto de 2020). *Polarización en tiempo real*. <https://es.ids-imaging.com/techtipp-details/items/techtip-on-camera-polarization.html>
- [REF 4] Sony, Sony Semiconductor Solutions Group (S.F.). *Polarization Image Sensor Technology Poloarsens*. Recuperado el 17 de junio de 2026. <https://www.sony-semicon.com/en/technology/industry/polarsens.html>
- [REF 5] Tucsen (30 de septiembre de 2025). *Eficiencia cuántica en cámaras científicas: una guía para principiantes*. <https://www.tucsen.com/es/learning/quantum-efficiency-in-scientific-cameras/>, www.tucsen.com
- [REF 6] Thorlabs (S.F.). *Half-Wave Liquid Crystal Variable Retarders / Wave Plates*. Recuperado el 22 de junio de 2026. <https://www.thorlabs.com/half-wave-liquid-crystal-variable-retarders-wave-plates?tabName=Overview>
- [REF 7] Thorlabs (S.F.). *K-Cube Liquid Crystal Controller*. Recuperado el 10 de junio de 2026. <https://www.thorlabs.com/k-cube-r-liquid-crystal-controller?tabName=Overview>
- [REF 8] Thorlabs (5 de noviembre de 205). *KLC k-Cube Liquid Crystal Controller User Guide*. (Rev. B). <https://media.thorlabs.com/globalassets/items/k/kl/klc/klc101/ctn014867-d02.pdf?v=0116115931>

[REF 9] Thorlabs (16 de diciembre de 2021). *CS505MUP, Polarization Camera With 5.0 MP Monochrome CMOS Sensor*.

<https://media.thorlabs.com/globalassets/items/c/cs/cs5/cs505mup/cs505mup.pdf?v=0116110527>

[REF 10] ImagingSource (abril 2025). *DYK 33UX250 Technical Reference Manual*.

https://s1-dl.theimagingsource.com/api/2.5/packages/documentation/manual-trm/trmdyk33ux250/0d172353-dece-501a-a915-447bd2fc981b/trmdyk33ux250.en_US.pdf

[REF 11] OpenCV, Open-Source Computer Vision (17 de junio de 2025). *Color Space Conversions*.

https://docs.opencv.org/3.4/d8/d01/group_imgproc_color_conversions.html



ANEXO 1: DEFINICIÓN DE CASOS DE USO

ANEXO 1.1: SOFTWARE PARA CÁMARA MONOCROMÁTICA DE POLARIZACIÓN

CU 1.1	<i>Pausar Vídeo</i>
Actores	Usuario
Descripción	Pausa la toma de vídeo, manteniendo visible el último fotograma capturado
Precondición	Tener una toma de vídeo en reproducción
Secuencia Normal	1. Iniciar la toma de vídeo con el programa 2. Pulsar el botón de Pausa 3. Fin
Postcondición	Ninguna
Excepciones	• Si en el paso 2 el vídeo no existe o ya se encuentra en pausa, abandonar
Frecuencia	Media
Importancia	Media

Tabla A1.1: Definición del caso de uso "Pausar Vídeo"

CU 1.2	<i>Reanudar Vídeo</i>
Actores	Usuario
Descripción	Retoma la toma de vídeo tras la detención de este
Precondición	CU 1.1 (Tener una toma de vídeo en pausa)
Secuencia Normal	1. Iniciar la toma de vídeo con el programa 2. Detener la toma de vídeo 3. Pulsar el botón de Reanudar 4. Fin
Postcondición	Ninguna
Excepciones	• Si en el paso 3 el vídeo no existe o ya se encuentra en reproducción, abandonar
Frecuencia	Media
Importancia	Media

Tabla A1.2: Definición del caso de uso "Reanudar Vídeo"

CU 1.3	<i>Guardar Imagen</i>
Actores	Usuario
Descripción	Guarda la imagen actual del video mostrada por pantalla
Precondición	Ninguna
Secuencia Normal	1. Iniciar la toma de vídeo con el programa 2. Pulsar el botón para guardar la imagen 3. Elegir nombre de la imagen y ruta de almacenamiento 4. Fin
Postcondición	Mensaje imagen guardada con éxito
Excepciones	• Si en el paso 3 ruta no existe, reintentar
Frecuencia	Media
Importancia	Baja

Tabla A1.3: Definición del caso de uso “Guardar Imagen”

CU 1.4	<i>Elegir Mapa de Color</i>
Actores	Usuario
Descripción	Cambia la forma de visualización de los parámetros
Precondición	Ninguna
Secuencia Normal	1. Iniciar la toma de vídeo con el programa 2. Pulsar el botón para cambiar el mapa de color 3. Repetir hasta elegir el deseado 4. Fin
Postcondición	Ninguna
Excepciones	Ninguna
Frecuencia	Alta
Importancia	Media

Tabla A1.4: Definición del caso de uso “Elegir Mapa de Color”

CU 1.5	<i>Mostrar Ventana de Color</i>
Actores	Usuario
Descripción	Crea una ventana auxiliar con un color sólido a elegir
Precondición	Ninguna
Secuencia Normal	1. Pulsar el botón para crear la ventana 2. Elegir el color para utilizar como fondo 3. Confirmar la creación con las características introducidas 4. Fin
Postcondición	Muestra de la ventana creada
Excepciones	Ninguna
Frecuencia	Baja
Importancia	Baja

Tabla A1.5: Definición del caso de uso “Mostrar Ventana de Color”

CU 1.6	<i>Modificar Voltaje</i>
Actores	Usuario
Descripción	Modifica el valor de la amplitud de la onda aplicada al dispositivo LCR
Precondición	Tener el LCR conectado al controlador
Secuencia Normal	1. Elegir el valor deseado utilizando los botones 2. Fin
Postcondición	Cambio de polaridad del LCR
Excepciones	• Si en el paso 1 el valor deseado no es válido, abandonar
Frecuencia	Muy alta
Importancia	Muy alta

Tabla A1.6: Definición del caso de uso “Modificar Voltaje”

CU 1.7	<i>Elegir Cálculo S3</i>
Actores	Usuario
Descripción	Permite seleccionar el método de cálculo del parámetro S_3
Precondición	Ninguna
Secuencia Normal	1. Elegir la opción deseada 2. Fin
Postcondición	Ninguna
Excepciones	Ninguna
Frecuencia	Media
Importancia	Muy alta

Tabla A1.7: Definición del caso de uso “Elegir Cálculo S3”

CU 1.8	<i>Comenzar Cálculo S3</i>
Actores	Usuario
Descripción	Da comienzo al cálculo del parámetro S_3
Precondición	CU 1.7 (Elegir la forma de cálculo) CU 1.10 (Definir el voltaje)
Secuencia Normal	1. Elegir la forma de cálculo deseada 2. Asignar el voltaje correspondiente 3. Pulsar el botón de comenzar cálculo 4. Fin
Postcondición	Muestra del resultado
Excepciones	Ninguna
Frecuencia	Muy alta
Importancia	Muy alta

Tabla A1.8: Definición del caso de uso “Comenzar Cálculo S3”

CU 1.9	<i>Detener Cálculo S3</i>
Actores	Usuario
Descripción	Detiene al cálculo del parámetro S_3
Precondición	CU 1.7 (Elegir la forma de cálculo) CU 1.10 (Definir el voltaje) CU 1.8 (Comenzar el cálculo)
Secuencia Normal	1. Elegir la forma de cálculo deseada 2. Asignar el voltaje correspondiente 3. Pulsar el botón de comenzar cálculo 4. Pulsar el botón de detener el cálculo 5. Fin
Postcondición	Ninguna
Excepciones	Ninguna
Frecuencia	Alta
Importancia	Alta

Tabla A1.9: Definición del caso de uso “Detener Cálculo S3”

CU 1.10	<i>Elegir Voltaje S3</i>
Actores	Usuario
Descripción	Define el voltaje a usar para el cálculo del parámetro S_3
Precondición	Ninguna
Secuencia Normal	1. Elegir el valor del voltaje deseado 2. Fin
Postcondición	Ninguna
Excepciones	• Si en el paso 1 el valor introducido no es válido, abandonar
Frecuencia	Muy Alta
Importancia	Muy Alta

Tabla A1.10: Definición del caso de uso “Elegir Voltaje S3”

ANEXO 1.2: SOFTWARE PARA CALIBRACIÓN DEL LCR

CU 2.1	<i>Seleccionar Área a Analizar</i>
Actores	Usuario
Descripción	Define el área que se tendrá en cuenta durante la calibración
Precondición	Ninguna
Secuencia Normal	1. Pulsar en la zona deseada 2. Modificar el radio del área 3. Fin
Postcondición	Visualización del área en el vídeo
Excepciones	Ninguna
Frecuencia	Alta
Importancia	Alta

Tabla A1.11: Definición del caso de uso “Seleccionar Área a Analizar”

CU 2.2	<i>Modificar Frecuencia</i>
Actores	Usuario
Descripción	Cambia la frecuencia de la señal utilizada durante el análisis
Precondición	Ninguna
Secuencia Normal	1. Definir el valor deseado usando los botones 2. Fin
Postcondición	Muestra del valor actual por pantalla
Excepciones	Ninguna
Frecuencia	Media
Importancia	Media

Tabla A1.12: Definición del caso de uso “Modificar Frecuencia”

CU 2.3	<i>Delimitar Voltaje Mínimo</i>
Actores	Usuario
Descripción	Indica la amplitud de onda inicial del análisis
Precondición	Ninguna
Secuencia Normal	1. Definir el valor deseado usando los elementos de la interfaz 2. Fin
Postcondición	Actualización del tiempo estimado
Excepciones	• Si en el paso 1 el valor introducido no es válido, mostrar mensaje de error
Frecuencia	Alta
Importancia	Alta

Tabla A1.13: Definición del caso de uso “Delimitar Voltaje Mínimo”

CU 2.4	<i>Delimitar Voltaje Máximo</i>
Actores	Usuario
Descripción	Indica la amplitud de señal final del análisis
Precondición	Ninguna
Secuencia Normal	1. Definir el valor deseado usando los elementos de la interfaz 2. Fin
Postcondición	Actualización del tiempo estimado
Excepciones	• Si en el paso 1 el valor introducido no es válido, mostrar mensaje de error
Frecuencia	Alta
Importancia	Alta

Tabla A1.14: Definición del caso de uso “Delimitar Voltaje Máximo”

CU 2.5	<i>Asignar Aumento de Voltaje</i>
Actores	Usuario
Descripción	Indica la variación de la amplitud de la señal entre los pasos del análisis
Precondición	Ninguna
Secuencia Normal	1. Definir el valor deseado usando los elementos de la interfaz 2. Fin
Postcondición	Actualización del tiempo estimado
Excepciones	• Si en el paso 1 el valor introducido no es válido, mostrar mensaje de error
Frecuencia	Alta
Importancia	Alta

Tabla A1.15: Definición del caso de uso “Asignar Aumento de Voltaje”

CU 2.6	<i>Asignar Pausa Entre Muestras</i>
Actores	Usuario
Descripción	Indica el tiempo que se mantiene cada muestra durante el análisis
Precondición	Ninguna
Secuencia Normal	1. Definir el valor deseado usando los elementos de la interfaz 2. Fin
Postcondición	Actualización del tiempo estimado
Excepciones	• Si en el paso 1 el valor introducido no es válido, mostrar mensaje de error
Frecuencia	Alta
Importancia	Alta

Tabla A1.16: Definición del caso de uso “Asignar Pausa Entre Muestras”

CU 2.7	Comenzar Análisis
Actores	Usuario
Descripción	Comienza la calibración con las características definidas
Precondición	CU 2.1 (Selección de área) CU 2.2 (Selección de frecuencia) CU 2.3 (Seleccionar voltaje mínimo) CU 2.4 (Seleccionar voltaje máximo) CU 2.5 (Seleccionar cambio entre muestras) CU 2.6 (Seleccionar tiempo entre muestras)
Secuencia Normal	1. Definir el área de análisis 2. Especificar las características del análisis usando la interfaz 3. Pulsar en el botón de iniciar análisis 4. Fin
Postcondición	Muestra de barra de progreso y gráfico resultado
Excepciones	• Si en el paso 3 no se ha seleccionado área de análisis o existen valores incorrectos, abandonar
Frecuencia	Muy Alta
Importancia	Muy Alta

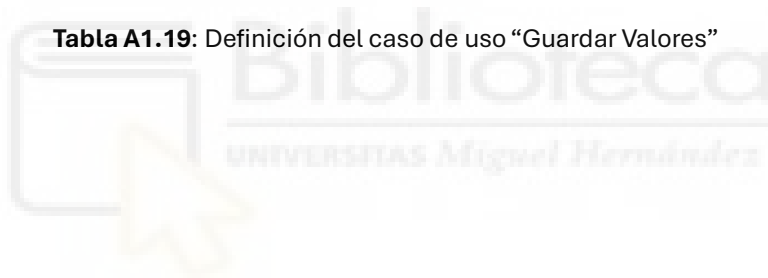
Tabla A1.17: Definición del caso de uso “Comenzar Análisis”

CU 2.8	Guardar Gráfico
Actores	Usuario
Descripción	Guarda los resultados de forma visual como imagen
Precondición	CU 2.1 (Selección de área) CU 2.2 (Selección de frecuencia) CU 2.3 (Seleccionar voltaje mínimo) CU 2.4 (Seleccionar voltaje máximo) CU 2.5 (Seleccionar cambio entre muestras) CU 2.6 (Seleccionar tiempo entre muestras) CU 2.7 (Realizar el análisis)
Secuencia Normal	1. Definir el área de análisis 2. Especificar las características del análisis usando la interfaz 3. Pulsar en el botón para realizar el análisis 4. Pulsar el botón para guardar los resultados como imagen 5. Seleccionar nombre de imagen y ruta de guardado 6. Fin
Postcondición	Mensaje guardado realizado con éxito
Excepciones	• Si en el paso 3 no se ha seleccionado área de análisis o existen valores incorrectos, abandonar • Si en el paso 4 ruta no existe, reintentar
Frecuencia	Media
Importancia	Media

Tabla A1.18: Definición del caso de uso “Guardar Gráfico”

CU 2.9	<i>Guardar Valores</i>
Actores	Usuario
Descripción	Guarda los resultados de forma numérica como archivo CSV
Precondición	CU 2.1 (Selección de área) CU 2.2 (Selección de frecuencia) CU 2.3 (Seleccionar voltaje mínimo) CU 2.4 (Seleccionar voltaje máximo) CU 2.5 (Seleccionar cambio entre muestras) CU 2.6 (Seleccionar tiempo entre muestras) CU 2.7 (Realizar el análisis)
Secuencia Normal	1. Definir el área de análisis 2. Especificar las características del análisis usando la interfaz 3. Pulsar en el botón para realizar el análisis 4. Pulsar el botón para guardar los resultados como archivo 5. Seleccionar nombre del archivo y ruta de guardado 6. Fin
Postcondición	Mensaje guardado realizado con éxito
Excepciones	• Si en el paso 3 no se ha seleccionado área de análisis o existen valores incorrectos, abandonar • Si en el paso 4 ruta no existe, reintentar
Frecuencia	Media
Importancia	Media

Tabla A1.19: Definición del caso de uso “Guardar Valores”



ANEXO 1.3: SOFTWARE PARA CALIBRACIÓN CON MONITOR

CU 3.1	<i>Seleccionar Área de Análisis</i>
Actores	Usuario
Descripción	Define el área que se tendrá en cuenta durante la calibración
Precondición	Ninguna
Secuencia Normal	1. Pulsar en la zona deseada 2. Modificar el radio del área 3. Fin
Postcondición	Visualización del área en el vídeo
Excepciones	Ninguna
Frecuencia	Alta
Importancia	Alta

Tabla A1.20: Definición del caso de uso “Seleccionar Área de Análisis”

CU 3.2	<i>Crear Ventana de Color</i>
Actores	Usuario
Descripción	Crea una ventana con fondo de color sólido variable
Precondición	Ninguna
Secuencia Normal	1. Pulsar en el botón de creación de ventana 2. Elegir el color inicial a usar 3. Fin
Postcondición	Despliegue por pantalla de la ventana creada
Excepciones	Ninguna
Frecuencia	Alta
Importancia	Alta

Tabla A1.21: Definición del caso de uso “Crear Ventana de Color”

CU 3.3	<i>Elegir Cambios en RGB</i>
Actores	Usuario
Descripción	Determina la variación de los componentes Rojo, Verde y Azul en cada muestra del análisis
Precondición	Ninguna
Secuencia Normal	1. Definir el valor deseado usando los elementos de la interfaz 2. Fin
Postcondición	Ninguna
Excepciones	• Si en el paso 1 el valor introducido no es válido, mostrar mensaje de error
Frecuencia	Alta
Importancia	Alta

Tabla A1.22: Definición del caso de uso “Elegir Cambios en RGB”

CU 3.4	<i>Elegir Límite RGB</i>
Actores	Usuario
Descripción	Determina el valor de los componentes Rojo, Verde y Azul de la última muestra del análisis
Precondición	Ninguna
Secuencia Normal	1. Definir el valor deseado usando los elementos de la interfaz 2. Fin
Postcondición	Ninguna
Excepciones	• Si en el paso 1 el valor introducido no es válido, mostrar mensaje de error
Frecuencia	Alta
Importancia	Alta

Tabla A1.23: Definición del caso de uso “Elegir Límite RGB”

CU 3.5	<i>Guardar Gráfico</i>
Actores	Usuario
Descripción	Guarda el gráfico resultado como una imagen en el dispositivo
Precondición	CU 3.1 (Elegir área de análisis) CU 3.2 (Crear ventana de color) CU 3.3 (Especificar cambios de color) CU 3.4 (Especificar valores límite) CU 3.7 (Realizar el análisis)
Secuencia Normal	1. Definir el área de análisis 2. Especificar las características del análisis usando la interfaz 3. Pulsar en el botón para realizar el análisis 4. Pulsar el botón para guardar los resultados como imagen 5. Seleccionar nombre de imagen y ruta de guardado 6. Fin
Postcondición	Mensaje guardado realizado con éxito
Excepciones	• Si en el paso 2 los valores introducidos no son válidos, abandonar • Si en el paso 5 la ruta introducida no existe, reintentar
Frecuencia	Media
Importancia	Media

Tabla A1.24: Definición del caso de uso “Guardar Gráfico”

CU 3.6	<i>Guardar Valores</i>
Actores	Usuario
Descripción	Guarda los valores de los resultados como un archivo CSV
Precondición	CU 3.1 (Elegir área de análisis) CU 3.2 (Crear ventana de color) CU 3.3 (Especificar cambios de color) CU 3.4 (Especificar valores límite) CU 3.7 (Realizar el análisis)
Secuencia Normal	1. Definir el área de análisis 2. Especificar las características del análisis usando la interfaz 3. Pulsar en el botón para realizar el análisis 4. Pulsar el botón para guardar los resultados como archivo 5. Seleccionar nombre del archivo y ruta de guardado 6. Fin
Postcondición	Mensaje guardado realizado con éxito
Excepciones	• Si en el paso 2 los valores introducidos no son válidos, abandonar • Si en el paso 5 la ruta introducida no existe, reintentar
Frecuencia	Media
Importancia	Media

Tabla A1.25: Definición del caso de uso “Guardar Valores”

CU 3.7	<i>Empezar Análisis</i>
Actores	Usuario
Descripción	Comienza el proceso de análisis
Precondición	CU 3.1 (Elegir área de análisis) CU 3.2 (Crear ventana de color) CU 3.3 (Especificar cambios de color) CU 3.4 (Especificar valores límite)
Secuencia Normal	1. Definir el área de análisis 2. Especificar las características del análisis usando la interfaz 3. Pulsar en el botón para realizar el análisis 4. Fin
Postcondición	Muestra de la barra de progreso y gráfico resultado
Excepciones	• Si en el paso 2 los valores introducidos no son válidos, abandonar
Frecuencia	Muy alta
Importancia	Muy alta

Tabla A1.26: Definición del caso de uso “Empezar Análisis”

ANEXO 1.4: SOFTWARE PARA CÁMARA RGB DE POLARIZACIÓN

CU 4.1	<i>Elegir Modo de Visualización</i>
Actores	Usuario
Descripción	Permite elegir si mostrar la captura de vídeo de forma polarizada o normal
Precondición	Ninguna
Secuencia Normal	1. Elegir la forma de visualización usando la interfaz 2. Fin
Postcondición	Actualización de la muestra de vídeo
Excepciones	Ninguna
Frecuencia	Media
Importancia	Alta

Tabla A1.27: Definición del caso de uso “Elegir Modo de Visualización”

CU 4.2	<i>Guardar Imagen</i>
Actores	Usuario
Descripción	Permite guardar el fotograma actual del vídeo como una imagen
Precondición	CU 4.3 (Elegir región a guardar) CU 4.4 (Elegir formato de la imagen)
Secuencia Normal	1. Elegir la zona a guardar usando la interfaz 2. Elegir el formato de imagen usando la interfaz 3. Pulsar en el botón para guardar imagen 4. Elegir nombre de la imagen y ruta de guardado 5. Fin
Postcondición	Mensaje guardado realizado con éxito
Excepciones	• Si en el paso 5 la ruta no existe, reintentar
Frecuencia	Media
Importancia	Media

Tabla A1.28: Definición del caso de uso “Guardar Imagen”

CU 4.3	<i>Elegir Zona a Guardar</i>
Actores	Usuario
Descripción	Permite elegir guardar la información de alguna componente de polarización específica o de todas
Precondición	Ninguna
Secuencia Normal	1. Elegir la zona a guardar usando la interfaz 2. Fin
Postcondición	Ninguna
Excepciones	Ninguna
Frecuencia	Media
Importancia	Baja

Tabla A1.29: Definición del caso de uso “Elegir Zona a Guardar”

CU 4.4	<i>Elegir Formato de Imagen a Guardar</i>
Actores	Usuario
Descripción	Permite elegir el formato de imagen en la que se guardarán los datos
Precondición	Ninguna
Secuencia Normal	1. Elegir el formato de la imagen usando la interfaz 2. Fin
Postcondición	Ninguna
Excepciones	Ninguna
Frecuencia	Media
Importancia	Baja

Tabla A1.30: Definición del caso de uso “Elegir Formato de Imagen a Guardar”

CU 4.5	<i>Modificar Exposición</i>
Actores	Usuario
Descripción	Permite modificar el valor de la exposición de la cámara
Precondición	Ninguna
Secuencia Normal	1. Elegir el valor deseado usando la interfaz 2. Fin
Postcondición	Actualización de la captura de vídeo
Excepciones	Ninguna
Frecuencia	Alta
Importancia	Alta

Tabla A1.31: Definición del caso de uso “Modificar Exposición”

CU 4.6	<i>Modificar Ganancia</i>
Actores	Usuario
Descripción	Permite modificar el valor de la ganancia de la cámara
Precondición	Ninguna
Secuencia Normal	1. Elegir el valor deseado usando la interfaz 2. Fin
Postcondición	Actualización de la captura de vídeo
Excepciones	Ninguna
Frecuencia	Media
Importancia	Alta

Tabla A1.32: Definición del caso de uso “Modificar Ganancia”

