

UNIVERSIDAD DE ALMERÍA

Departamento de Informática

**Procesado eficiente de
imagen mediante técnicas
multi-espectrales y
computación cuántica**

**(Efficient image processing through
multi-spectral techniques and quantum
computing)**

Almería, Diciembre 2023

Autor:

Luis Ortega López

Directoras:

Gracia Ester Martín Garzón

Gloria Ortega López

Tesis Doctoral

Procesado eficiente de imagen mediante técnicas multi-espectrales y computación cuántica



Universidad de Almería
Departamento de Informática
Doctorado en Informática (RD99/11)

Autor: Luis Ortega López
Directoras: Gracia Ester Martín Garzón
Gloria Ortega López

Almería, Diciembre 2023

Ph.D. Thesis

Efficient image processing through multi-spectral techniques and quantum computing



University of Almería
Informatics Department
Ph.D. in Informatics (RD99/11)

Author: Luis Ortega López
Supervisors: Gracia Ester Martín Garzón
Gloria Ortega López

Almería, December 2023

He de darle gracias al universo por regalarme una vida llena de cosas por descubrir. Gracias por darme una familia acogedora, su cariño, su respaldo y su apoyo. Gracias por darme buenos amigos que han motivado mi ilusión y mi curiosidad por el mundo. Dedico este libro a todas las personas que han hecho posible esta tesis.

Agradecimientos

Quiero expresar mi más sincero agradecimiento a mis tutoras, Ester Martín y Gloria Ortega, por su inestimable orientación, su apoyo constante y dedicación durante todo el proceso de desarrollo de esta tesis. Su experiencia y conocimiento han sido fundamentales para el éxito de este trabajo.

También me gustaría agradecer a Francisco Agüera-Vega, Fernando Carvajal y Patricio Carricondo (del Departamento de Ingeniería de la UAL) por su colaboración y valiosas aportaciones que enriquecieron significativamente mi investigación. Su apoyo ha sido esencial para llevar a cabo este proyecto.

Finalmente, me gustaría extender mi agradecimiento al grupo de investigación “TIC-146: Supercomputación y Algoritmos”, en cuyo seno se ha desarrollado la presente tesis. Especialmente, me gustaría destacar a Marcos Lupión Lorente, cuya profunda comprensión de las redes neuronales y su generosa colaboración han sido determinantes para definir una estrategia eficaz de comparación y evaluación del algoritmo de segmentación y clasificación desarrollado en esta tesis. Su conocimiento y asesoramiento fueron fundamentales en la consecución de los resultados alcanzados.

A todos vosotros os estoy profundamente agradecido por vuestra contribución, orientación y apoyo a lo largo de este viaje.

Contribuciones científicas

Artículos

1. L. O. López, G. Ortega, F. Agüera-Vega, F. Carvajal-Ramírez, P. Martínez-Carricondo, and E. M. Garzón. **Multi-Spectral Imaging for Weed Identification in Herbicides Testing**. *Informatica*, Volume 33, Issue 4, pp. 771–793 (2022). <https://doi.org/10.15388/22-INFOR498> JCR (2022): 2.9. Subject categories: MATHEMATICS, APPLIED 29/267 (Q1); COMPUTER SCIENCE, INFORMATION SYSTEMS 99/158 (Q3)
2. L. O. López, F. Orts, G. Ortega, V. Ruíz, and E. M. Garzón. **Fault-tolerant quantum algorithm for dual-threshold image segmentation**. *The Journal of Supercomputing*, Volume 79, pp. 12549–12562 (2023). <https://link.springer.com/article/10.1007/s11227-023-05148-9>. JCR (2022): 3.3. Subject categories: COMPUTER SCIENCE, THEORY & METHODS 37/111 (Q2); COMPUTER SCIENCE, HARDWARE & ARCHITECTURE 22/54 (Q2); ENGINEERING, ELECTRICAL & ELECTRONIC 114/275 (Q2).

Conferencias

1. G. Ortega, L. O. López, I. García, A. Plaza, E. M. Garzón. **Accelerating the LRASR anomaly detection method on heterogeneous platforms**. En *8th International Workshop Data Analysis Methods for Software Systems (DAMSS)*, Druskininkai, Lithuania, 1-3 de diciembre de 2016.
2. L. O. López. **Procesado eficiente de imágenes en el campo de la agricultura de precisión. Corrección de imágenes obtenidas con cámaras multi-espectrales**. En *Jornadas de seguimiento de la investigación (JSIDI)*, 15 de febrero de 2018, Universidad de Almería, Almería.
3. L. O. López. **Análisis de imágenes multi-espectrales aplicadas al campo de la agricultura**. En *II Jornadas de Doctorado en Informática (JDI'2019)*, 14 de febrero de 2019, Universidad de Almería, Almería.
4. L. O. López, G. Ortega, F. Agüera-Vega, F. Carvajal-Ramírez, P. Martínez Carricondo, E. M. Garzón. **Corrección de imágenes multi-espectrales para agricultura de precisión**. En *II Congreso de Jóvenes Investigadores en Ciencias Agroalimentarias*, 17 de octubre de 2019, Universidad de Almería, Almería.
5. L. O. López, G. Ortega, F. Agüera-Vega, F. Carvajal-Ramírez, P. Martínez Carricondo, E.M. Garzón. **Preprocesado de imágenes multi-espectrales aplicado al campo de la agricultura**. En *VIII Simposio de Investigación en Ciencias Experimentales*, 14 de noviembre de 2019, Universidad de Almería, Almería.
6. L. O. López. **Clasificación de cultivos utilizando imágenes multi-espectrales y software eCognition**. En *III Jornadas de Doctorado en Informática (JDI'2020)*, 21 de febrero de 2020, Universidad de Almería, Almería.
7. L. O. López. **Algoritmo de clasificación de cultivos utilizando imágenes multi-espectrales y software eCognition**. En *IV Jornadas de Doctorado en Informática (JDI'2021)*, 26 de febrero de 2021, Universidad de Almería, Almería.
8. L. O. López. **Algoritmo cuántico de segmentación de doble umbral para agricultura de precisión**. En *V Jornadas de Doctorado en Informática de la Universidad de Almería*, 25 de febrero de 2022, Universidad de Almería, Almería. ISBN: 978-84-1351-142-9.

9. L. O. López, F. Orts, E. M. Garzón, V. González-Ruiz, M. Lupión, G. Ortega. **Efficient quantum comparators for dual-threshold image segmentation**. En *Computational and Mathematical Methods in Science and Engineering & International Conference in HPC (CMMSE 2022)*, 3-7 de julio de 2022, Rota, Cádiz, España.
10. L. O. López, F. Orts, L. M. Donaire, G. Ortega, E. M. Garzón. **Improving quantum comparators for image segmentation**. En *XI Simposio de investigación en ciencias experimentales*, 15 de noviembre de 2022, Universidad de Almería, Almería.

Educación

1. G. Ortega, L.O. López, F. Orts, C. Díaz-Carrasco, J. J. Moreno, E. M. Garzón. **M-Learning based initiative to enhance student motivation in Computer Science**. En *The 17th International Conference on Frontiers in Education: Computer Science & Computer Engineering (FECS 2021)*, Las Vegas, Nevada, USA, 26-29 de julio de 2021.
2. G. Ortega, J. J. Moreno, E. M. Garzón, F. Orts, L. M. Donaire, L.O. López, C. Díaz-Carrasco, M. Lupión, A. Redondo. **Domotic-based setups using a Raspberry Pi to increase interest in computer engineering**. En *16th Annual International Conference of Education, Research and Innovation (ICERI)*, Seville (Spain), 13-15 de noviembre de 2023.

Resumen

La investigación científica y tecnológica avanza a un ritmo vertiginoso. En este continuo progreso, la capacidad de procesamiento de imágenes desempeña un papel crucial en una amplia variedad de campos de aplicación (medicina, teledetección, seguridad, agricultura, etc.), donde un análisis exhaustivo de los datos permite una mejor toma de decisiones.

Actualmente, existen cámaras y sensores que posibilitan la captura de una gran cantidad de imágenes de alta resolución. El procesamiento de estas imágenes representa un desafío computacional significativo en la tarea de extraer eficientemente la información que contienen. Este aspecto es aún más evidente al trabajar con imágenes que incorporan información en distintas longitudes de onda o bandas espectrales, como ocurre en el caso de las imágenes multi-espectrales.

Por otro lado, dentro del procesado de imagen también es de interés destacar el procesamiento cuántico de imágenes que, aunque actualmente es un área emergente, promete un gran potencial en la resolución de problemas complejos de procesamiento de imágenes en el futuro. Este procesamiento aprovecha las propiedades de la mecánica cuántica para resolver problemas complejos, abriendo nuevas posibilidades en el campo del procesamiento de imágenes. Sin embargo, la computación cuántica se encuentra aún en una fase preliminar, debido principalmente a los retos que suponen el incremento en el número de cúbits de las plataformas, y mantener la fiabilidad de los dispositivos con una gestión adecuada del ruido inherente a su funcionamiento. A medida que esta tecnología continúe desarrollándose es probable que se alcancen avances significativos en la capacidad de procesamiento de imágenes. Por ello, resulta esencial que el diseño de circuitos cuánticos esté optimizado, permitiendo así que los algoritmos de procesado de imágenes puedan utilizarlos cuando los desafíos anteriores estén resueltos.

A lo largo de esta tesis se exploran los fundamentos teóricos y las aplicaciones prácticas de dos problemas relacionados con el procesamiento de imagen.

En primer lugar, se investiga un problema de agricultura de precisión que trabaja con imágenes multi-espectrales. Concretamente, se realiza la implementación de una herramienta automática para comprobar el funcionamiento de herbicidas en un cultivo de girasol. Este problema se aborda utilizando técnicas de visión artificial como es la clasificación basada en objetos y, adicionalmente, se compara con dos aproximaciones basadas en Aprendizaje Profundo. Uno de los principales retos al trabajar con imágenes multi-espectrales son las distorsiones tanto geométricas como de perspectiva que estas presentan. Por lo tanto, en esta tesis se lleva a cabo un exhaustivo proceso de calibración y co-registro de las imágenes de una cámara multi-espectral Sequoia. Este calibrado puede considerarse también una aportación de la tesis, puesto que la metodología definida podría extrapolarse para calibrar otras cámaras multi-espectrales.

El segundo problema abordado está relacionado con la computación cuántica. Concretamente, en esta tesis se implementa un algoritmo tolerante a fallos para la segmentación de imágenes sobre plataformas cuánticas. El algoritmo se denomina “dual threshold”, o algoritmo de segmentación de doble umbral. Debido a que este algoritmo involucra dos operaciones de comparación, se diseñan dos circuitos cuánticos comparadores tolerantes a fallos. La bondad tanto de los circuitos cuánticos comparadores como del algoritmo implementado se mide utilizando un conjunto de

métricas contrastadas entre las que destaca el coste de puertas T, que son las puertas relacionadas con la tolerancia a fallos.

Finalmente, los desarrollos e implementaciones que se han propuesto para solucionar los dos problemas anteriormente citados se comparan con sus análogos del estado del arte, mostrando las bondades de las aportaciones realizadas durante la presente disertación.

Abstract

Scientific and technological research is advancing at a rapid pace. In this era of continuous progress, image processing capability plays a crucial role in various application domains such as medicine, remote sensing, security, agriculture, etc., facilitating comprehensive data analysis for informed decision-making.

Currently, there are cameras and sensors that facilitate the capture of a large number of high-resolution images. The processing of these images poses a significant computational challenge in efficiently extracting the information they contain. The difficulty intensifies when working with images that incorporate information at different wavelengths or spectral bands, as is the case with multi-spectral images.

On another note, quantum image processing is also of interest in the image processing domain. Although currently an emerging field, quantum image processing holds great potential for addressing complex image processing problems in the future. This processing leverages the properties of quantum mechanics to address intricate problems, opening new possibilities in the field of image processing. However, quantum computing is still in a preliminary phase, primarily due to the challenges associated with increasing the number of qubits on platforms and maintaining device reliability with proper management of inherent operational noise. As this technology continues to develop, significant advancements in image processing capability are likely to be achieved. Hence, optimizing the design of quantum circuits is essential to enable image processing algorithms to utilize them once the aforementioned challenges are resolved.

Throughout this thesis, the theoretical foundations and practical applications of two image processing-related problems are explored.

Firstly, a precision agriculture problem dealing with multi-spectral images is investigated. Specifically, an automated tool is implemented to assess the performance of herbicides in a sunflower crop. This problem is addressed using computer vision techniques, such as object-based classification, and is additionally compared with two Deep Learning-based approaches. One of the main challenges when working with multi-spectral images is the geometric and perspective distortions they present. Therefore, this thesis undertakes a thorough calibration and co-registration process for images from a Sequoia multi-spectral camera. This calibration can also be considered a contribution of the thesis, as the defined methodology could be extrapolated to calibrate other multi-spectral cameras.

The second problem addressed is related to quantum computing. Specifically, this thesis implements a fault-tolerant algorithm for image segmentation on quantum platforms. The algorithm is referred to as the "dual threshold" or dual-threshold segmentation algorithm. Since this algorithm involves two comparison operations, two fault-tolerant quantum comparator circuits are designed. The goodness of both quantum comparator circuits and the implemented algorithm is assessed using a set of established metrics, among which the T-gate cost stands out. T-gates are gates associated with fault tolerance.

Finally, the developments and implementations proposed to address the two aforementioned problems are compared to their state-of-the-art counterparts, demonstrating the merits of the contributions made during this dissertation.



Contenidos

Lista de Abreviaturas	xxiii
1 Introducción	1
1.1 Motivación de la tesis	1
1.2 Objetivos de la tesis	2
1.3 Estructura de la tesis	5
2 Tecnologías y arquitecturas	7
2.1 Recursos software	7
2.1.1 Matlab	7
2.1.2 Software Trimble eCognition	9
2.1.3 Librería TensorFlow	10
2.1.4 IBM Quantum	10
2.2 Recursos Hardware	11
2.2.1 Arquitecturas multicore	11
2.2.2 Computación cuántica	12
3 Preprocesado de imagen	23
3.1 Importancia del preprocesado de imagen	24
3.2 Campos de aplicación	24
3.2.1 Agricultura de precisión e imágenes multi-espectrales	25
3.3 Técnicas de mejora de imagen	26
3.3.1 Ecualización de histograma	26
3.3.2 Ajuste del contraste	28
Procesado eficiente de imagen mediante técnicas multi-espectrales y computación cuántica	xi

3.3.3	Ajuste del brillo	30
3.3.4	Saturación y realce de color	31
3.3.5	Otras técnicas de mejora de la imagen	33
3.4	Técnicas de eliminación de ruido	34
3.4.1	Tipos de ruido	34
3.4.2	Filtros lineales para la eliminación del ruido	36
3.4.3	Filtros no lineales para la eliminación del ruido	37
3.5	Técnicas de calibrado y eliminación de la distorsión	38
3.5.1	Modelo de cámara	38
3.5.2	Aberraciones primarias y cromáticas	42
3.5.3	Calibración geométrica de la cámara	45
3.5.4	Corrección de las distorsiones geométricas (radial y tangencial)	46
3.5.5	Corrección de la perspectiva (rectificación al plano principal)	49
3.6	Resumen de las técnicas de preprocesado utilizadas en la tesis	50
4	Técnicas de segmentación	51
4.1	Segmentación basada en umbralización	52
4.2	Segmentación basada en detección de bordes	53
4.3	Segmentación basada en regiones	54
4.3.1	Segmentación basada en clusterización	55
4.3.2	Segmentación basada en crecimiento de regiones	56
4.3.3	Segmentación Quadtree	57
4.3.4	Segmentación multi-resolución	58
4.4	Técnicas de segmentación basadas en aprendizaje profundo (Deep Learning)	60
4.4.1	Segmentación basada en U-Net	62
4.4.2	Segmentación basada en redes FPN	62
4.5	Resumen de las técnicas de segmentación utilizadas en la tesis	64
5	Caso de estudio de agricultura de precisión	65
5.1	Definición del caso de estudio en agricultura de precisión	66
5.1.1	Estado del arte de la segmentación de imágenes multi-espectrales en agricultura de precisión	68
5.1.2	Sistema de imagen	68
5.1.3	Experimento de campo	70
5.2	Preprocesado de las imágenes del caso de estudio	70
5.2.1	Proceso de calibración geométrica de las lentes de la cámara	71
5.2.2	Corrección de la distorsión radial y tangencial de las imágenes del caso de estudio	73
5.2.3	Rectificación al plano principal y escalado	74
5.3	Composición de imágenes multi-espectrales (co-registro)	74

5.4	Proceso de clasificación de las imágenes del caso de estudio	75
5.4.1	Proceso de clasificación basado en técnicas de visión artificial con objetos (ClasOb)	76
5.4.2	Enfoques de Aprendizaje Profundo	81
5.4.3	Evaluación del proceso de clasificación	82
5.5	Resultados del caso práctico de agricultura de precisión	83
5.5.1	Discusión de la calibración de la lente	83
5.5.2	Discusión del proceso de clasificación	84
5.6	Conclusiones y trabajo futuro del caso de estudio de agricultura de precisión	88
6	Caso de estudio de procesamiento cuántico de imagen	91
6.1	Segmentación de doble umbral con computación cuántica	92
6.2	Codificación de imágenes utilizando la representación NEQR	93
6.3	Optimización de puertas cuánticas en términos de la métrica T	94
6.4	Implementaciones tolerantes a fallos de circuitos comparadores	95
6.4.1	Estado del arte de los circuitos comparadores para la segmentación de doble umbral	95
6.4.2	Circuitos comparadores propuestos para la segmentación de doble umbral	96
6.4.3	Operador reset en IBM Quantum	98
6.5	Algoritmo propuesto de segmentación de imágenes con doble umbral	99
6.6	Evaluación de los circuitos cuánticos comparadores y de los algoritmos cuánticos de segmentación de doble umbral	100
6.7	Conclusiones y trabajo futuro del caso de estudio de segmentación cuántica de doble umbral	102
7	Conclusiones y trabajo futuro	105
7.1	Español	105
7.2	English	108
	Bibliografía	111



Lista de Figuras

1.1	Esquema general de la tesis donde se identifican los dos casos de estudio abordados, el tipo de arquitecturas sobre las que se han realizado las implementaciones, las tecnologías utilizadas y los resultados obtenidos.	3
2.1	Imagen típica utilizada para la calibración de una lente. Esta imagen está tomada por una cámara multi-espectral Sequoia en la banda REG.	9
2.2	Esfera de Bloch donde cada punto de la superficie corresponde a un vector de estado normalizado que representa un estado cuántico válido.	14
2.3	Símbolo y matriz de la puerta Pauli-X.	15
2.4	Símbolo y matriz de la puerta Pauli-Y.	15
2.5	Símbolo y matriz de la puerta Pauli-Z.	15
2.6	Símbolo y matriz de la puerta identidad.	15
2.7	Símbolo y matriz de la puerta CNOT.	16
2.8	Símbolos y matrices de las puertas V y V^+	16
2.9	Símbolos y matrices de las puertas V y V^+ Controladas, que aplican la operación de V y V^+ al primer cúbit si el cúbit de control tiene el valor de $ 1\rangle$	17
2.10	Símbolo y matriz de la puerta Toffoli, la cual niega el cúbit objetivo si ambos cúbit de control se encuentran en estado $ 1\rangle$	17
2.11	Símbolo y matriz de la puerta Hadamard, que se aplica a un cúbit y crea un estado de superposición.	17
2.12	Símbolo y matriz de la puerta Swap la cual se aplica sobre dos cúbits e intercambia sus estados.	18
2.13	Implementación y símbolo de la puerta Peres.	18
2.14	Puerta Temporary logical-AND y su implementación (33) (arriba) e implementación de la puerta de descomputación de la puerta Temporary Logical-AND, donde no hay ninguna puerta T involucrada (abajo).	19

3.1	Imagen original con su histograma sin modificar. Se observa una ligera sobreexposición en la imagen. En el gráfico, el eje x indica cada uno de los 256 grados de nivel de gris posibles en la imagen, mientras que el eje y representa la distribución de probabilidad de ese color en términos del número de píxeles.	29
3.2	Imagen modificada con su histograma ecualizado en 10 bins planos. El eje x de la gráfica indica cada uno de los 10 grados de nivel de gris destino (bins). El eje y representa la distribución de probabilidad de ese color en términos del número de píxeles.	29
3.3	Imagen modificada con su histograma ecualizado en 64 bins relativamente planos. El eje x de la gráfica indica cada uno de los 64 grados de nivel de gris destino (bins). El eje y representa la distribución de probabilidad de ese color en términos del número de píxeles.	29
3.4	Imagen modificada con su histograma estirado. El contraste se ha aumentado al máximo posible sin perder información.	30
3.5	Imagen modificada con su histograma desplazado a la izquierda 50 niveles de color, resultando en una disminución del brillo.	31
3.6	Imagen original sin modificar su saturación. Esta imagen corresponde a una fotografía de cultivos de girasol en fase temprana.	32
3.7	Imagen tras editar la Figura 3.6 aumentando la saturación en un 50%.	32
3.8	Imagen original del videojuego, input del algoritmo DLSS (75).	34
3.9	Imagen reescalada y mejorada, output del algoritmo DLSS (75).	34
3.10	Ejemplo de deconvolución ciega extraído de la referencia (57). (a) Input de imagen borrosa para algoritmos de deconvolución ciega. (b) Mapa de verdad de la imagen para evaluar la calidad de las técnicas empleadas. (c) Imagen restaurada mediante un algoritmo de deconvolución ciega basado en la técnica propuesta en (27). (d) Imagen similar restaurada mediante la técnica propuesta en (99).	35
3.11	(a) Imagen original. (b) Imagen con ruido gaussiano. (c) Imagen con ruido "sal y pimienta".	36
3.12	Ejemplo de cámara estenopeica. Todos los puntos del objeto pasan por una única apertura a a través de la barrera definida (40).	39
3.13	Ejemplos de nitidez de una imagen capturada por una cámara estenopeica. Tamaños más pequeños de apertura aportan nitidez, pero eliminan cantidad de luz (19).	40
3.14	Las lentes enfocan los rayos de luz paralelos al eje óptico en el punto focal. Además, esta configuración ilustra el modelo de refracción paraxial, que nos ayuda a encontrar la relación entre los puntos en el plano de la imagen y el mundo tridimensional en cámaras con lentes, asumiendo una lente totalmente plana (40).	41
3.15	Imagen de la Università Ca' Foscari Venezia que muestra una aberración cromática o distorsión del tipo "fringing".	42
3.16	Dada una imagen original, ejemplos de distorsiones de tipo cojín y barril provocadas por las imperfecciones de la lente.	43
3.17	Aberración del sistema óptico del tipo "coma". El ángulo θ indica la desviación respecto al eje paraxial.	44
3.18	Fotografías de ejemplo de diferentes posiciones de un damero obtenidos de la web de la librería Matlab Calibration Toolbox. El orden de las esquinas se mantiene constante al variar la posición relativa para tener una referencia estable.	46

3.19	Distorsión radial y parámetros intrínsecos k_{1-3} después de la calibración de la cámara.	47
3.20	Distorsión tangencial y parámetros intrínsecos p_{1-2} obtenidos tras la calibración de la cámara.	48
3.21	Cálculo de las posiciones relativas del damero respecto a la cámara durante el análisis, que reflejan los parámetros extrínsecos.	48
3.22	Imagen original que presenta distorsiones geométricas.	49
3.23	Imagen sin distorsión tras realizar una transformación homográfica.	49
3.24	Imagen de entrada, donde la región de interés está delimitada por las cuatro esquinas del marco.	50
3.25	Imagen rectificada al plano principal, resultado de la transformación homográfica.	50
4.1	Imagen original.	52
4.2	Imagen binarizada en blanco y negro.	52
4.3	(a) Imagen original; (b) Imagen después de aplicar el operador Sobel; (c) Imagen después de aplicar el filtro Canny; (d) Imagen después de aplicar el filtro Roberts.	54
4.4	Imagen original.	56
4.5	Imagen segmentada en tres regiones tras aplicar el algoritmo K-means para $k = 3$	56
4.6	Imagen original y segmentada en regiones tras la aplicación de tres nodos semilla (2).	57
4.7	Imagen original.	59
4.8	Imagen segmentada en regiones tras aplicar el algoritmo Quadtree.	59
4.9	Imagen original.	60
4.10	Imagen segmentada en regiones mediante multi-resolución.	60
4.11	Esquema de la operación de una capa de convolución en una CNN, consistente en tres pasos. Paso 1: kernels definidos mediante aprendizaje esperando ser ejecutados. Paso 2: los kernels se aplican uno a uno sobre la imagen input utilizando una convolución. Paso 3: la salida de cada convolución produce un mapa de activación.	61
4.12	Esquema funcional de una red de segmentación convolucional U-Net, en forma de "U" (97).	63
4.13	Esquema funcional de una red de segmentación convolucional FPN, en forma de pirámide.(18)	63
5.1	Esquema general que indica los principales pasos a seguir en el proceso de identificación de malas hierbas utilizando imágenes multi-espectrales.	67
5.2	Cámara Parrot Sequoia donde se observan las diferentes cámaras para las bandas de interés, así como el sensor de luz solar.	69
5.3	Fotografía del sensor GRE de una cuadrícula tomada con la cámara Sequoia. Se puede percibir a simple vista la fuerte distorsión de barril.	69
5.4	Fotografía en bruto del espectro NIR donde se muestra el marco utilizado en el experimento.	70
5.5	Diagrama del proceso de calibración de las lentes.	72
5.6	Diagrama del proceso de evaluación de la corrección de las lentes.	72
5.7	Imágenes de prueba del damero con distorsión de barril corregida en la banda GRE.	73

5.8	Imágenes de prueba del damero rectificadas en la banda GRE. Las cruces rojas marcan las intersecciones que se almacenarán en la matriz $A = (a_{ij})$. La matriz $B = (b_{ij})$ se calculará utilizando las posiciones teóricas de las intersecciones mostradas.	73
5.9	Esquema del proceso de corrección de distorsión geométrica.	74
5.10	Esquema del proceso de rectificación al plano principal de las imágenes y su escalado.	75
5.11	(a) Fotografía en crudo en la banda NIR para la muestra 38; (b) Banda NIR de la muestra 38 después de corregir la distorsión de la lente; (c) Banda NIR después de corregir la perspectiva tomando como referencia las cuatro esquinas internas del marco; y (d) Representación RGB de tres bandas superpuestas (GRE, NIR y RED).	76
5.12	Diagrama del proceso de clasificación.	77
5.13	(a) Imagen de salida para la muestra 38 después de ejecutar el bloque de clasificación de materia vegetal-suelo. La materia vegetal y el suelo están marcados en color verde y negro, respectivamente; (b) Imagen después de agrupar la materia vegetal en nodos adyacentes y eliminar clústeres con menos del 0.03% del tamaño total de la imagen; (c) Imagen después de ejecutar la segmentación Quadtree multi-espectral; (d) Imagen con los clústeres en forma de cuadrados marcados (semillas); (e) Imagen después de la primera etapa del algoritmo de crecimiento; (f) Imagen de la segunda etapa del algoritmo de crecimiento. Las áreas que potencialmente corresponden a girasol están marcadas en color amarillo y las áreas potenciales de maleza en color rojo; (g) Imagen de la tercera etapa del algoritmo de crecimiento de regiones; (h) Imagen de salida del proceso de clasificación desde el visor de eCognition. Las áreas amarillas representan materia vegetal clasificada como girasol, las áreas rojas representan materia vegetal clasificada como maleza y las áreas negras identifican el suelo; e (i) Imagen de salida del proceso de clasificación.	81
5.14	Tres imágenes de muestra (34, 25 y 15) segmentadas por el algoritmo clásico propuesto (una por fila). (a), (d) y (g) muestran una representación RGB de la imagen multi-espectral utilizando las bandas RED, NIR y GRE; (b), (e) y (h) muestran el resultado del algoritmo; y (c), (f) y (i) muestran la verdad terreno (ground truth) determinada por humanos.	87
6.1	Símbolos y matrices del conjunto de puertas Clifford+T utilizadas en el algoritmo de segmentación de imágenes de doble umbral propuesto.	95
6.2	Circuito comparador propuesto por Yuan et al. (127), para el caso $N = 3$. Está optimizado en términos de cúbits auxiliares. Sin embargo, implica el uso de puertas no Clifford+T para realizar la operación de reinicio.	96
6.3	Circuito comparador obtenido al reemplazar las operaciones de reinicio en el circuito de Yuan et al. (127). La parte de descomputación consiste en aplicar el mismo circuito, pero en orden inverso. Este circuito solo contiene puertas Clifford+T, por lo que es tolerante a fallos. Sin embargo, su T-depth se duplica y tiene el doble de cúbits auxiliares que el circuito original.	97
6.4	Primera propuesta de circuito comparador cuántico centrado en reducir el número de puertas T, para el caso $N = 3$. a_i y b_i son los cúbits de los dos números A y B que se van a comparar. Las salidas marcadas como x e y son las salidas de la comparación.	97

6.5	Segunda propuesta de circuito comparador cuántico centrado en reducir el número de puertas T, para el caso $N = 3$. a_i y b_i son los bits de los dos números A y B que se van a comparar. Las salidas marcadas como x e y son las salidas de la comparación.	98
6.6	Circuito en IBM Quantum que representa la realidad de una operación reset.	99
6.7	Circuito del algoritmo de segmentación de doble umbral propuesto por Yuan et al. (127). En este ejemplo, el umbral bajo se establece en 2 y el umbral alto en 6. El uso de la operación de reinicio hace que el circuito no sea reversible. . .	100
6.8	Implementación propuesta del algoritmo de segmentación de doble umbral. En este ejemplo, el umbral bajo se establece en 2 y el umbral alto en 6. Se implementa utilizando solo puertas Clifford+T, y reduce el T-count en 6 con respecto al circuito mostrado en la Figura 6.7. Los comparadores requerirán entradas auxiliares adicionales, como se ilustró en las Figuras 6.4 y 6.5. . . .	101



Lista de Tablas

- 5.1 Número de imágenes utilizadas para cada sensor en el proceso de calibración. La columna “Total de imágenes” identifica el número total de imágenes consideradas, mientras que “Imágenes para conjunto de calibración” e “Imágenes para conjunto de prueba” muestran cuántas de ellas se utilizaron para el conjunto de calibración y el conjunto de prueba, respectivamente. La columna “Media de c_{ij} (error en píxeles)” muestra el error promedio en píxeles después del proceso de calibración considerando las imágenes del conjunto de prueba. 83
- 5.2 Estudio preliminar de las redes U-Net y FPN entrenadas en tres conjuntos de datos diferentes, y validadas con 16 representaciones RGB obtenidas de la Sección 5.3. Los mejores valores en términos de IoU promedio para cultivos (Girasol) y maleza están marcados en negrita. 84
- 5.3 Resultados de IoU obtenidos para 16 imágenes de muestra multi-espectrales utilizando tres enfoques de clasificación: la que se ha propuesto en este capítulo (denominada ClasOb) y dos modelos de DL (U-Net_Multi y FPN_I_Jes_Multi). Además, se muestra el porcentaje de áreas de materia vegetal, girasol y maleza. Los mejores valores de IoU han sido resaltados en negrita. 86
- 5.4 IoU promedio para la estrategia propuesta ClasOb y dos enfoques de DL, para cada clase en la clasificación de 16 imágenes de muestra. Los mejores valores de IoU promedio se han resaltado en negrita. 87
- 6.1 Comparación de T-count, T-depth y cúbits auxiliares entre los comparadores propuestos y los mejores diseños encontrados en la literatura. También se indica si los circuitos utilizan solo puertas Clifford+T y si son completos o medios comparadores. 102

- 6.2 Comparación, en términos de T-count, T-depth y número de cúbits auxiliares, entre el circuito original de segmentación de imágenes de doble umbral presentado por Yuan et al. (127), Wang et al. (114) y el propuesto en este trabajo, que incluye tolerancia a fallos mediante el uso en exclusiva del subconjunto de puertas Clifford+T. La métrica T-count no incluye el coste de los comparadores, por lo que a estos valores se debe agregar el del comparador elegido de los que se muestran en la Tabla 6.1 (teniendo en cuenta que el comparador se aplica dos veces). Los circuitos no requieren nuevos cúbits auxiliares, ya que pueden reutilizar los cúbits auxiliares de los comparadores. 103



Lista de Abreviaturas

CNN *Convolutional Neural Networks* (Redes Neuronales Convolucionales)

DA *Data Augmentation* (Aumento de Datos)

DBSCAN *Density-Based Spatial Clustering of Applications with Noise* (Agrupamiento Espacial Basado en Densidad de Aplicaciones con Ruido)

DL *Deep Learning* (Aprendizaje Profundo)

DLSS *Deep Learning Super Sampling* (Super Muestreo de Aprendizaje Profundo)

FN *False Negatives* (Falsos Negativos)

FP *False Positives* (Falsos Positivos)

FPN *Feature Pyramid Network* (Red de Pirámide de Características)

GAN *Generative Adversarial Networks* (Redes Generativas Adversariales)

GRE *Green* (Verde)

IA Inteligencia Artificial

IFRQI *Improved Flexible Representation of Quantum Images* (Mejora de la Representación Flexible de Imágenes Cuánticas)

INCQI *Improved Color Digital Image Quantum Representation* (Mejora de la Representación Cuántica de Imágenes Digitales a Color)

INEQR *Improved Novel Enhanced Quantum Representation* (Mejora de la Nueva Representación Cuántica Mejorada)

IoU *Intersection Over Union* (Intersección sobre la Unión)

MCRQI *Multi-Channel Representation of Quantum Images* (Representación Multicanal de Imágenes Cuánticas)

mIoU *Mean Intersection Over Union* (Media de la Intersección sobre la Unión)

NIR *Near Infrared* (Infrarrojo Cercano)

NDVI *Normalized Difference Vegetation Index* (Índice de Vegetación de Diferencia Normalizada)

NEQR *Novel Enhanced Quantum Representation* (Nueva Representación Cuántica Mejorada)

OCR *Optical Character Recognition* (Reconocimiento Óptico de Caracteres)

QIMP *Quantum Image Processing* (Procesamiento de Imagen Cuántico)

QPIXL *Quantum Pixel Representation* (Representación Cuántica de Píxeles)

REG *Red Edge* (Banda Roja)

RGB *Red, Green, Blue* (Rojo, Verde, Azul)

UAV *Unmanned Aerial Vehicles* (Vehículos Aéreos No Tripulados)

USB *Universal Serial Bus* (Bus Universal en Serie)

VP Verdaderos Positivos



1. Introducción

En este capítulo introductorio se muestra una visión completa del trabajo de tesis realizado. En primer lugar, en la Sección 1.1 se motiva esta tesis y se justifica el interés de utilizar la computación en general y la cuántica en particular para resolver los problemas planteados. A continuación, en la Sección 1.2 se describen los casos de estudio que se han considerado y se establecen los principales objetivos que se han alcanzado con el desarrollo de esta tesis. Por último, la Sección 1.3 indica la estructura que tiene el presente documento.

1.1 Motivación de la tesis

El procesado de imagen se utiliza para extraer información útil a partir de imágenes digitales y la velocidad y precisión del procesamiento tienen un impacto significativo en la calidad y utilidad de la información que se obtenga.

En la actualidad, existen diversas áreas de aplicación relacionadas con el procesamiento de imágenes digitales, cuyo propósito radica en mejorar la calidad de las imágenes para una interpretación precisa por parte de los seres humanos. Como resultado, esta disciplina ha experimentado un notable crecimiento en los últimos años. Algunas de las áreas donde el procesamiento eficiente de imagen es un aspecto clave son: la medicina [89], la robótica [93], la automatización industrial [112], la seguridad [62], la fotografía digital [55], la teledetección [132] y la agricultura de precisión [105].

En general, los algoritmos utilizados para preprocesar una imagen tienen un alto coste computacional, lo que puede limitar la velocidad y la eficiencia del procesamiento. Este hecho es incluso más evidente cuando se trabaja con imágenes que contienen información en diferentes longitudes de onda o bandas espectrales, como es el caso de las imágenes multi-espectrales

o hiper-espectrales. Por lo tanto, la optimización del procesamiento de imágenes es fundamental para maximizar el rendimiento y la precisión de las aplicaciones que dependen de ellas [34]. Existen numerosas estrategias para optimizar el procesamiento de imagen. Entre ellas destacan la implementación de algoritmos eficientes, así como el hecho de utilizar arquitecturas de alto rendimiento y tecnologías emergentes para implementar tales algoritmos. Con respecto al uso de arquitecturas de alto rendimiento, destaca el uso de las tarjetas gráficas, arquitecturas multicore y clústeres de procesadores, mientras que entre las tecnologías emergentes destaca el uso de la computación cuántica para este fin.

La computación cuántica se caracteriza por el principio de superposición, que permite un procesamiento paralelo de la información, lo que tiene el potencial de acelerar significativamente el procesamiento de imágenes así como una amplia gama de otros tipos de cálculos. El procesamiento cuántico de imágenes (Quantum Image Processing, QIMP) es un área emergente que combina el uso de la computación cuántica y el procesamiento de información cuántica para diseñar y trabajar con imágenes. Este campo busca responder a preguntas tales como: cuál es la mejor manera de representar imágenes en computadores cuánticos, y cómo se deben preparar, procesar y recuperar dichas imágenes [121, 122]. La idea principal es aprovechar las propiedades de la computación cuántica, como el entrelazamiento y el paralelismo, para lograr avances en el campo del procesamiento de imágenes [34].

Sin embargo, QIMP todavía se encuentra en una etapa muy temprana de desarrollo y su uso en aplicaciones prácticas sigue siendo limitado. Por lo tanto, el realizar avances a la hora de diseñar circuitos cuánticos para el preprocesado de imágenes es de gran interés para la comunidad científica.

Esta tesis se ha centrado en dos casos de estudio relacionados con el procesamiento de imagen utilizando enfoques muy diferentes. En el primero se han seleccionado y aplicado algoritmos en el contexto de agricultura de precisión. En el segundo caso, se ha optimizado el diseño de circuitos para procesamiento de imagen eficiente en el contexto de la computación cuántica. Principalmente, el foco de estudio ha sido la segmentación de imágenes, un problema bien conocido de visión artificial en el que aún quedan retos por abordar, ya que conseguir la segmentación y posterior clasificación de imágenes es una tarea compleja.

1.2 Objetivos de la tesis

En esta tesis, el procesamiento de imagen es un aspecto clave y se utiliza sobre dos casos de estudio, uno centrado en la agricultura de precisión y otro relacionado con un algoritmo de segmentación de doble umbral, tal y como se muestra en la Figura 1.1.

En primer lugar, se investiga el desarrollo de una herramienta computacional que permita la localización automática de plantas de interés (girasoles) y de maleza en imágenes de cultivos de girasol. El fin de esta herramienta es testear la eficacia de herbicidas en un cultivo. Las malas hierbas son un problema relevante para los cultivos porque pueden causar daños de varias formas. Compiten con los cultivos por espacio, luz, agua y nutrientes, y pueden albergar enfermedades

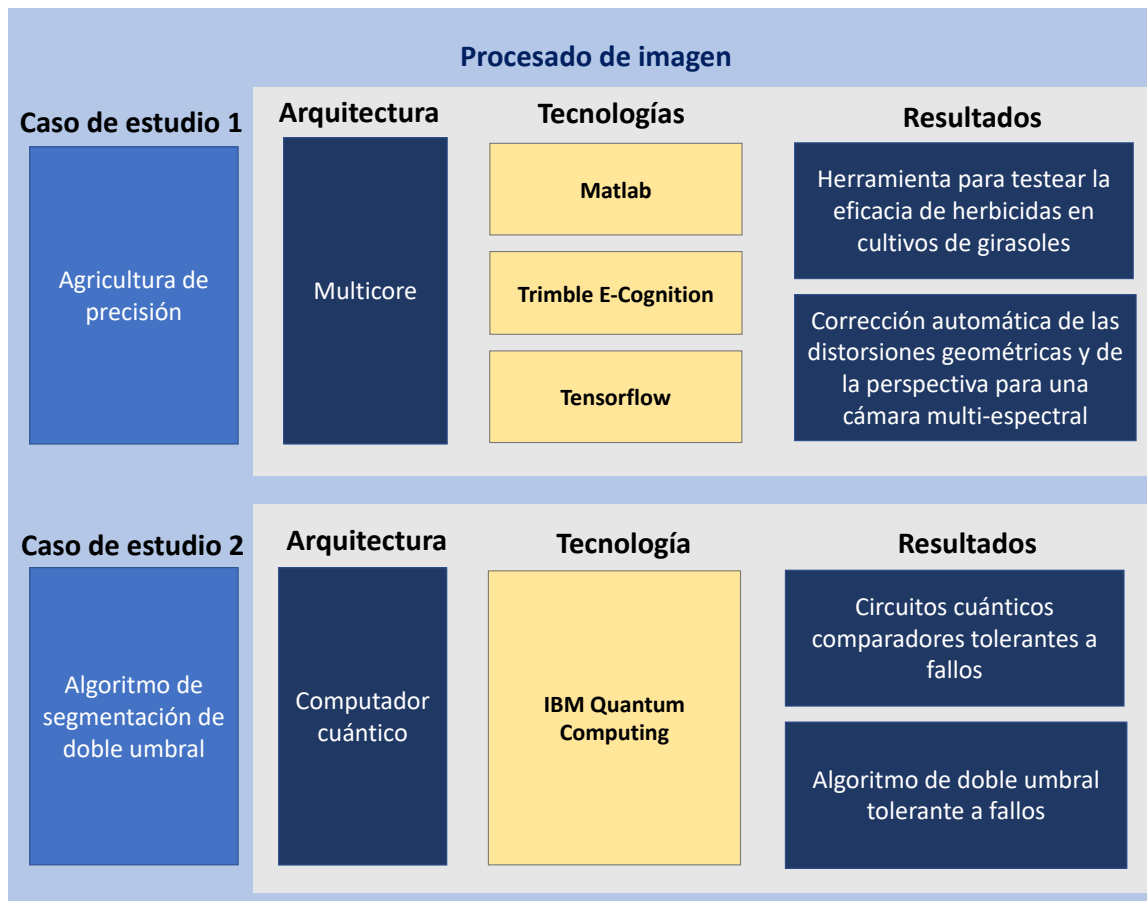


Figura 1.1: Esquema general de la tesis donde se identifican los dos casos de estudio abordados, el tipo de arquitecturas sobre las que se han realizado las implementaciones, las tecnologías utilizadas y los resultados obtenidos.

y plagas. Los herbicidas intentan minimizar la presencia de estas malas hierbas. Sin embargo, se debe realizar una evaluación del rendimiento de estos herbicidas en los cultivos antes de que se comercialicen. En cuanto a la medición de la eficacia de los herbicidas recién desarrollados, los métodos cualitativos, también conocidos como métodos de evaluación visual, siguen siendo los más utilizados a pesar de sus limitaciones [108]. Existe una normativa específica para la autorización de herbicidas en forma comercial y su puesta en el mercado [21] que exige la realización de un informe que incluya información relacionada con la eficacia, la aparición o posible aparición de resistencias, efectos adversos en los cultivos tratados y otros efectos secundarios no deseados. En consecuencia, actualmente se llevan a cabo ensayos para investigar todos estos problemas específicos. Los datos derivados de estos ensayos deben estar relacionados con cada una de las especies en estudio y, por lo tanto, se debe discriminar previamente la especie. Avanzar en esta línea utilizando métodos cuantitativos basados en la teledetección para la prueba de herbicidas es de gran interés para la comunidad científica.

Las imágenes de entrada de este caso de estudio han sido capturadas con una cámara multi-espectral. Estas imágenes, que contienen información en diferentes longitudes de onda, son de gran interés ya que permiten realizar análisis exhaustivos de información de interés, por ejemplo, determinar la salud de los cultivos, identificar problemas de fertilidad, detectar enfermedades o

estrés en las plantas, así como monitorizar el crecimiento y desarrollo de los cultivos a lo largo del tiempo.

Para conseguir el desarrollo de una herramienta automática de testeo de herbicidas, en primer lugar, estas imágenes han sido preprocesadas para mejorar su calidad. Posteriormente, se han empleado diversas técnicas de segmentación y de clasificación con el fin de identificar automáticamente el porcentaje de planta principal (girasol) y de maleza presente en las imágenes. Los resultados obtenidos con nuestra herramienta han sido comparados con dos enfoques basados en Deep Learning, concluyendo que nuestra propuesta supera a las versiones Deep Learning consideradas en la clasificación de materia vegetal y maleza. Además, la herramienta propuesta no necesita ni la generación de conjuntos de datos ni el proceso de entrenamiento del modelo, como es el caso de las estrategias basadas en aprendizaje profundo.

Este problema se ha abordado de forma conjunta con el Departamento de Ingeniería de la Universidad de Almería. Este departamento disponía de decenas de fotos de cultivos reales tomadas con una cámara multi-espectral, sobre las cuales existía la necesidad de realizar un análisis automático de la cantidad de maleza y de cultivo en las diferentes tomas. Gracias a esta tesis, se ha implementado una herramienta automática para analizar y procesar tales imágenes y así simplificar el proceso de evaluación del desempeño de herbicidas específicos en cultivos de girasoles. La herramienta resultante de este estudio podría extenderse para considerar otros tipos de cultivos. Además, debido a que fue necesario el calibrado de la cámara multi-espectral y la correcciones de aberraciones geométricas y de perspectiva que presentaban las imágenes originales, también se ha obtenido como parte de la presente tesis una metodología automatizada para corregir las aberraciones comentadas anteriormente.

El segundo caso de estudio se ha centrado en la implementación eficiente de un algoritmo para la segmentación de imágenes, conocido como “algoritmo de segmentación por doble umbral” o “dual threshold” sobre plataformas cuánticas, utilizando el paradigma basado en circuitos. Este método se basa en identificar si el valor de una imagen se encuentra dentro del rango de valores establecido por dos umbrales. Los métodos de segmentación basados en umbrales son computacionalmente eficientes, directos y ampliamente utilizados en campos tan diversos como la medicina [124] o la agricultura de precisión. Por ejemplo, la segmentación basada en doble umbral se ha utilizado de manera efectiva en la segmentación de imágenes de enfermedades de hojas de maíz [16] y en la estimación de cultivos de arroz maduros [111].

El algoritmo de segmentación de doble umbral se ha implementado utilizando circuitos cuánticos tolerantes a fallos. En concreto, se han propuesto varios circuitos cuánticos comparadores que son también tolerantes a fallos. En la actualidad, el desarrollo de circuitos y algoritmos sobre plataformas cuánticas presenta un desafío significativo debido a la escasez de recursos, especialmente en términos de número de cúbits disponibles en las plataformas cuánticas. Además, se agrega una complicación adicional, ya que los dispositivos cuánticos actuales son altamente susceptibles al ruido, tanto interno como externo. Esto requiere que los diseñadores utilicen una parte de los recursos disponibles para implementar procedimientos de detección y corrección de errores con el propósito de mitigar estos efectos no deseados. Como resultado, los recursos, que ya son limitados, se ven aún más reducidos. Por lo tanto, los circuitos deben ser tolerantes a fallos para mitigar los efectos del ruido, mientras se optimizan los recursos disponibles.

En este segundo estudio se ha realizado una comparación tanto del algoritmo como de los circuitos cuánticos comparadores implementados con respecto a sus homólogos presentes en la literatura, demostrando que las propuestas llevadas a cabo en esta tesis son tolerantes a fallos y, además, optimizan el coste de las puertas asociadas a la tolerancia a fallos, mejorando así la eficiencia del conjunto.

En base a estos dos casos de estudio, los principales objetivos de la presente tesis han sido los siguientes:

- Desarrollar una herramienta para corregir automáticamente las distorsiones geométricas y de perspectiva de las imágenes capturadas a través de una cámara multi-espectral situada a corta distancia. El uso de una cámara multi-espectral a pocos metros del objeto de interés conlleva una serie de desafíos, entre los que destacan dos. En primer lugar, la posición relativa de las lentes resulta en la percepción de la escena de manera ligeramente diferente desde cada cámara, lo que se conoce como error de perspectiva. En segundo lugar, las características intrínsecas de cada lente de la cámara introducen diversas distorsiones geométricas. Por lo tanto, el disponer de una herramienta que automáticamente corrija estas distorsiones y componga una imagen multi-espectral de mejor calidad es de interés para distintos campos de aplicación.
- Implementar una herramienta automática para comprobar la eficacia de herbicidas en cultivos de girasoles. Concretamente, esta herramienta permitirá cuantificar el porcentaje de maleza y plantas de interés (girasoles) que se encuentren en un área determinada, que es una tarea que actualmente se está llevando a cabo mayoritariamente mediante métodos de evaluación visual, a pesar de sus limitaciones. Esta herramienta deberá poder extenderse para poder ser aplicada a otros cultivos de interés.
- Realizar avances significativos en la implementación de algoritmos sobre plataformas cuánticas basadas en circuitos. En concreto, se pretende diseñar circuitos cuánticos tolerantes a fallos, que optimicen sus recursos y procesen el principal algoritmo de segmentación de imágenes mediante doble umbral.
- Implementar circuitos cuánticos comparadores tolerantes a fallos, siendo de especial interés que puedan integrarse en el algoritmo de segmentación de doble umbral mencionado anteriormente.

1.3 Estructura de la tesis

En el presente capítulo se ha descrito tanto la motivación como los objetivos de la tesis y la estructura del documento. En el Capítulo 2 se detallan los distintos recursos software y hardware que se han considerado para el desarrollo de esta tesis. En el Capítulo 3 se explican las principales técnicas de preprocesado sobre imágenes, que es un aspecto clave en la presente tesis. En el capítulo 4 se detallan las técnicas de segmentación más utilizadas en el estado del arte. El Capítulo 5 está dedicado al caso práctico de agricultura de precisión que se ha abordado en esta tesis. En el Capítulo 6 se presentan de manera detallada las contribuciones asociadas a la implementación de un algoritmo de segmentación de doble umbral en una plataforma cuántica

basada en circuitos. Finalmente, en el Capítulo 7, se ilustran las principales conclusiones y trabajos futuros de la tesis.



2. Tecnologías y arquitecturas

En este capítulo se describen en detalle tanto las tecnologías (software específico) como las arquitecturas y paradigmas computacionales que se han utilizado para el desarrollo de la presente tesis.

2.1 Recursos software

En esta sección se describen las principales tecnologías y recursos software que han sido considerados para el desarrollo de esta tesis. Concretamente, para el primer caso de estudio se ha utilizado la interfaz Matlab, el software Trimble eCognition y la librería Tensorflow, mientras que para el segundo caso de estudio, se ha utilizado la plataforma IBM Quantum.

2.1.1 Matlab

Matlab (MATrix LABoratory)¹ es un entorno software, así como un lenguaje de programación que fue desarrollado por MathWorks. Matlab se utiliza ampliamente en la academia, la investigación y la industria para realizar una gran variedad de tareas relacionadas con el análisis numérico, la modelización, la simulación y la visualización de datos. Entre las funciones más destacadas de Matlab se encuentran el análisis numérico, la optimización, la estadística y el procesamiento de señales, entre otras. Asimismo, este entorno brinda una gran cantidad de herramientas para la visualización y creación de gráficos que permiten a los usuarios comprender mejor los datos y resultados obtenidos.

¹<https://www.mathworks.com/products/matlab.html>

Por su capacidad para resolver problemas complejos en campos como la ingeniería, la física, las matemáticas, la economía y la biología, así como en la investigación académica y educación, Matlab se ha posicionado como una herramienta de referencia. Además, es una herramienta muy accesible ya que está disponible para distintas plataformas, tales como Windows, Mac y Linux.

Matlab pone a disposición de sus usuarios múltiples librerías temáticas. Un Toolbox es un software que se utiliza para resolver un conjunto específico de problemas en una disciplina o aplicación determinada. Los Toolbox de Matlab contienen múltiples funciones y herramientas que pueden ser utilizadas para una amplia gama de aplicaciones, desde análisis de datos y cálculo numérico hasta procesamiento de señales, control y diseño de sistemas. En esta tesis, se han utilizado tanto Computer Vision Toolbox como Camera Calibration Toolbox.

Computer Vision Toolbox y Camera Calibration Toolbox

Computer Vision Toolbox de Matlab es una colección de herramientas y funciones que están diseñadas para facilitar la implementación de algoritmos de visión artificial. Esta librería está destinada a ayudar en el procesamiento de imágenes y vídeos, permitiendo extraer información visual de manera efectiva.

Este conjunto de herramientas ofrece una amplia gama de funciones predefinidas que abarcan tareas comunes en visión artificial, como la detección y seguimiento de objetos, el reconocimiento de patrones, la segmentación de imágenes y la calibración de cámaras. Además, se puede combinar con otras funcionalidades de Matlab, lo que simplifica el flujo de trabajo al combinar el procesamiento de imágenes con análisis de datos y otros cálculos.

En esta tesis, Computer Vision Toolbox se ha utilizado en la etapa de preprocesamiento de las imágenes del problema de agricultura de precisión.

Camera Calibration Toolbox de Matlab es una extensión especializada de Computer Vision Toolbox, centrada en la calibración precisa de cámaras, que se utiliza para calibrar cámaras y eliminar la distorsión de la imagen. En óptica, la calibración de una cámara se refiere a la estimación de los parámetros intrínsecos y extrínsecos de la propia lente que afectan la proyección de la imagen en el plano de la imagen. Los parámetros intrínsecos incluyen la distancia focal, el punto principal, la relación de aspecto y los parámetros de distorsión, mientras que los parámetros extrínsecos describen la posición y orientación de la cámara en relación con la escena.

Camera Calibration Toolbox de Matlab ofrece una serie de funciones que permiten la medición precisa de los parámetros de la cámara y la eliminación de la distorsión de la imagen. Esto es útil en aplicaciones de visión artificial donde se necesita una medición precisa de las dimensiones y posiciones de los objetos en la imagen, y especialmente útil en la calibración de las diferentes imágenes que componen una imagen multi-espectral completa.

En esta tesis se ha utilizado Matlab 2019b en el primer caso de estudio, y su licencia ha sido proporcionada por parte de la Universidad de Almería. Concretamente, se ha utilizado Matlab para resolver el problema de agricultura de precisión, donde ha sido necesaria la calibración

de las lentes y sensores, la aplicación de los parámetros de distorsión para la corrección de la misma, la selección del área de interés y la rectificación al plano principal.

La calibración de la cámara se realiza generalmente utilizando un objeto de calibración como referencia, por ejemplo, un damero. Esta librería permite calcular los parámetros intrínsecos de cada lente utilizando un conjunto de imágenes de este tipo. En la Figura 2.1 se observa una imagen con un damero que ha sido capturada en la banda REG (Red Edge). En la imagen se puede observar una fuerte distorsión en la curvatura del tablero, que debería ser recto.

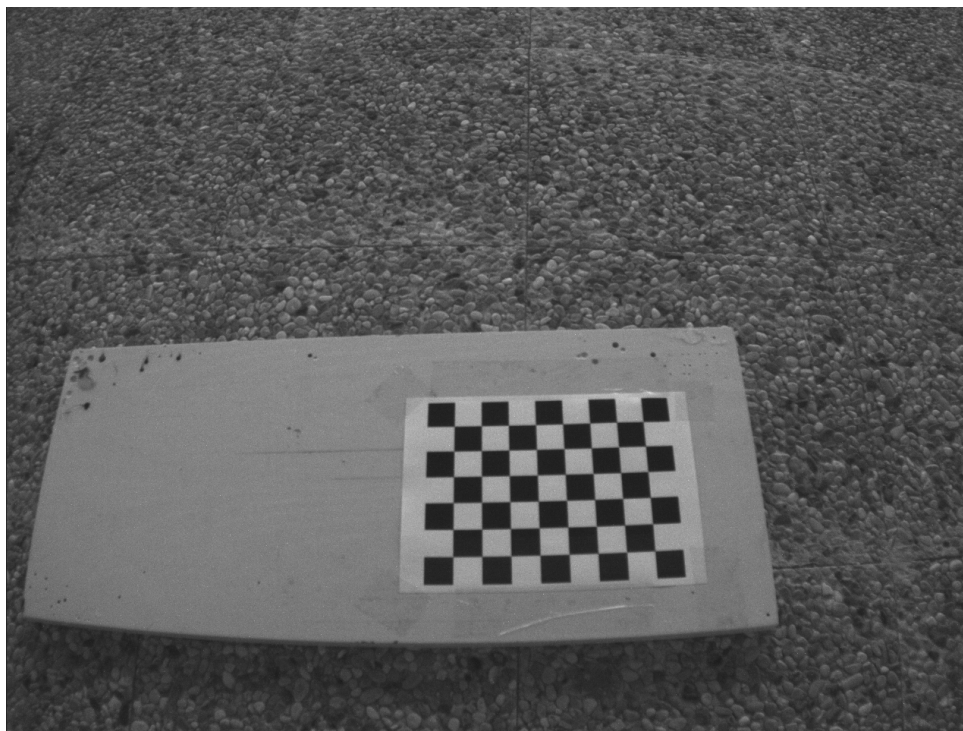


Figura 2.1: Imagen típica utilizada para la calibración de una lente. Esta imagen está tomada por una cámara multi-espectral Sequoia en la banda REG.

2.1.2 Software Trimble eCognition

Trimble eCognition Developer² es un entorno de desarrollo para el análisis de imágenes basado en objetos. Se utiliza ampliamente en varios contextos para el análisis automático de datos de teledetección. El software Trimble eCognition es utilizado por profesionales de sistemas de información geográfica, expertos en teledetección y científicos de datos para automatizar el análisis de datos geospaciales. Este software clasifica y analiza imágenes, vectores y nubes de puntos utilizando toda la información semántica requerida para interpretarla correctamente. En lugar de examinar píxeles o puntos independientes, extrae el significado de las connotaciones y las relaciones mutuas de los objetos, no solo con objetos vecinos, sino en todo el conjunto de datos de entrada. Este software utiliza el color, la forma, la textura y el tamaño de los objetos, así como su contexto y relaciones para sacar extraer conclusiones de las imágenes [117].

²<https://geospatial.trimble.com/products-and-solutions/trimble-ecognition>

En esta tesis se ha utilizado la versión 9.5.0 del software eCognition Developer para implementar un clasificador de los píxeles de la imagen, utilizando técnicas de procesamiento de imágenes basadas en objetos. Esto ha sido posible porque permite programar algoritmos complejos gracias a las rutinas de procesamiento de imágenes de las que consta. Este software es propietario y su licencia de uso ha sido proporcionada por parte del departamento de Ingeniería de la Universidad de Almería.

2.1.3 Librería TensorFlow

TensorFlow es una librería de código abierto para el aprendizaje automático y la inteligencia artificial desarrollada por Google. Proporciona una plataforma para construir, entrenar y desplegar modelos de aprendizaje automático y redes neuronales profundas. TensorFlow se utiliza ampliamente en una variedad de aplicaciones, incluyendo reconocimiento de imágenes, procesamiento de lenguaje natural, traducción automática, análisis de datos y mucho más. Esta librería es muy utilizada en la comunidad científica debido a su flexibilidad, escalabilidad y capacidad de ejecutarse sobre múltiples plataformas y dispositivos. TensorFlow permite a los desarrolladores trabajar con un alto nivel de abstracción para construir y entrenar modelos, y también les permite personalizar la red neuronal y las operaciones de aprendizaje a través de su estructura de gráficos.

TensorFlow permite la creación de modelos de aprendizaje automático personalizados que pueden segmentar imágenes automáticamente. Los modelos de segmentación de imágenes de TensorFlow utilizan técnicas de redes neuronales convolucionales (CNN) para analizar las características de las imágenes y separarlas en regiones distintas. TensorFlow ofrece una amplia variedad de modelos pre-entrenados para la segmentación de imágenes, así como herramientas para personalizar y entrenar modelos específicos para tareas de segmentación específicas.

En esta tesis se ha utilizado TensorFlow para implementar modelos de segmentación mediante Deep Learning, basados en U-Net y en FPN [1]. Estos modelos han permitido realizar una comparativa entre los algoritmos de visión artificial clásicos basados en objetos y dos modelos del estado del arte de segmentación mediante redes neuronales.

2.1.4 IBM Quantum

IBM Quantum es una plataforma en línea proporcionada por IBM desde el año 2016, que ofrece acceso a computadoras cuánticas reales y recursos relacionados con la computación cuántica. Es una iniciativa para fomentar la investigación, el desarrollo y la educación en el campo de la computación cuántica y brindar a los usuarios la oportunidad de experimentar y aprender sobre esta tecnología emergente.

La plataforma IBM Quantum ha sido un punto de referencia en la industria de la computación cuántica desde su lanzamiento. Concretamente, la plataforma cuántica que se ofrece es de propósito general y se basa en el modelo de circuitos. Esta interfaz permite a los usuarios ejecutar circuitos cuánticos tanto en simuladores como en plataformas cuánticas reales a través

de una interfaz en línea. Esto proporciona una oportunidad única para explorar y comprender el funcionamiento de las computadoras cuánticas y sus aplicaciones potenciales.

Una de las características más destacadas de IBM Quantum es el acceso a procesadores cuánticos reales, que incluyen cúbits superconductores de alta calidad. A medida que la computación cuántica avanza, IBM continúa desarrollando y mejorando sus sistemas cuánticos, lo que brinda a los usuarios la posibilidad de trabajar con hardware cada vez más avanzado y poderoso.

Además de proporcionar acceso a hardware cuántico, la plataforma IBM Quantum ofrece una amplia gama de recursos educativos y herramientas para ayudar a los usuarios a adentrarse en el mundo de la computación cuántica y también ofrece una serie de librerías cuánticas avanzadas como Qiskit Terra, Qiskit Aer, Qiskit Ignis y Qiskit Aqua para el desarrollo de algoritmos cuánticos más sofisticados [47].

En esta tesis, se ha utilizado IBM Quantum Composer que es conjunto personalizable de herramientas que permiten construir, visualizar y ejecutar circuitos cuánticos en hardware cuántico o simuladores. En concreto, esta herramienta se ha utilizado en el segundo caso de estudio, para el diseño de circuitos cuánticos implicados en algoritmos de segmentación de imágenes.

2.2 Recursos Hardware

En esta sección se detallan los componentes físicos que se han utilizado para el desarrollo de esta tesis.

2.2.1 Arquitecturas multicore

En el primer caso de estudio abordado en esta tesis, relacionado con la agricultura de precisión, se utilizan los paradigmas computacionales Matlab y Trimble eCognition. Ambos paradigmas están diseñados para aprovechar de manera eficiente algunos recursos hardware, concretamente los múltiples núcleos de procesamiento del procesador. De este modo, el programador se abstrae de tareas relacionadas con la programación paralela, pero se aprovecha la capacidad de ejecutar de forma paralela en arquitecturas multicore, en lugar de programar directamente con hilos (threads).

Las implementaciones que se han llevado en el primer caso de estudio han sido realizadas sobre procesadores de memoria compartida con varios núcleos, también conocidos como procesadores multicore. En estos procesadores, las mejoras de rendimiento se han conseguido gracias a añadir más procesadores (cores) al mismo chip, en lugar de enfocarse en aumentar la frecuencia del reloj o la velocidad del procesador. Los cores están conectados al mismo bus, compartiendo el mismo dispositivo de memoria y otros recursos del chip, así como todos los recursos externos.

2.2.2 Computación cuántica

En la actualidad, los computadores cuánticos están experimentando un auge impulsado por el desarrollo de la computación cuántica. A pesar de este progreso, se enfrentan a importantes desafíos, como el elevado coste de fabricación, la preservación de la información y su limitada tolerancia a fallos. Estos desafíos son relevantes en el contexto del segundo caso de estudio de esta tesis, que se centra en el diseño de circuitos cuánticos, evidenciando la necesidad de abordar y superar estos obstáculos para avanzar en el campo de la computación cuántica.

La computación cuántica aplica las propiedades de los sistemas cuánticos (como la superposición, el entrelazamiento y la interferencia) para realizar tareas computacionales. Las propiedades cuánticas no se manifiestan en nuestro mundo macroscópico y, aunque están presentes a nivel subatómico en los dispositivos informáticos, no se utilizan explícitamente en los modelos de computación tradicionales que empleamos para construir los microprocesadores, ni en el diseño de nuestros algoritmos. Por esta razón, los computadores cuánticos se comportan de manera radicalmente diferente a los computadores clásicos, lo que hace posible resolver algunas tareas de manera mucho más eficiente que con dispositivos informáticos tradicionales [24].

Como se ha comentado anteriormente, las principales propiedades de los sistemas cuánticos son la superposición, el entrelazamiento y la interferencia, por lo que resulta de interés describir estos conceptos.

La superposición se refiere a la capacidad de un sistema cuántico de estar en múltiples estados al mismo tiempo. En contraste con la física clásica, donde un sistema se encuentra en un estado bien definido, en la superposición cuántica un sistema puede existir en una combinación lineal de varios estados posibles. Por ejemplo, en el famoso experimento del gato de Schrödinger, se plantea un gato en una caja que está en un estado de superposición cuántica de estar vivo y muerto al mismo tiempo [96]. Según la mecánica cuántica, hasta que se observe el estado del gato, existe una superposición de ambos estados posibles.

El entrelazamiento cuántico se refiere a la correlación o conexión intrínseca que puede existir entre dos o más partículas cuánticas, de manera que el estado de una partícula no se pueda describir independientemente del estado de las demás [10, 45]. Cuando dos partículas cuánticas están entrelazadas, el estado cuántico conjunto del sistema no puede ser descompuesto en los estados individuales de cada partícula de forma separada. Esto significa que, al medir o cambiar el estado de una de las partículas entrelazadas, instantáneamente afectará al estado de la otra, incluso si están separadas por grandes distancias. Este fenómeno se conoce como “acción fantasmal a distancia”. El entrelazamiento cuántico tiene aplicaciones importantes en diversos campos, como la criptografía cuántica o la teleportación cuántica.

La interferencia cuántica hace alusión al fenómeno en el cual dos o más estados cuánticos superpuestos se combinan y se interfieren entre sí, dando lugar a patrones de interferencia característicos [83]. En la interferencia cuántica, los estados cuánticos superpuestos se combinan y pueden interferir constructivamente (reforzarse) o destructivamente (anularse) dependiendo de la fase relativa entre ellos. Este fenómeno se observa, por ejemplo, en el experimento de la doble rendija, donde se envían partículas individuales a través de dos rendijas y se obtiene un patrón

de interferencia en la pantalla de detección, lo cual indica que las partículas interfieren consigo mismas [28].

Existen dos modelos computacionales que incluyen máquinas cuánticas de Turing, la computación cuántica basada en circuitos y la computación cuántica adiabática. Estos modelos son fundamentales para comprender cómo se pueden aprovechar los principios de la mecánica cuántica en el procesamiento de información.

El paradigma de computación adiabática se basa en el teorema adiabático. En este tipo de computación se lleva a cabo una evolución lenta de un estado inicial utilizando un Hamiltoniano que depende del tiempo hasta alcanzar un estado final. Si esta evolución ocurre de forma gradual y los niveles de energía no se cruzan, el estado fundamental del Hamiltoniano inicial se transformará en el estado de energía mínima después de un tiempo lo suficientemente largo [24, 37]. La computación cuántica adiabática se fundamenta en inicializar el sistema en un estado de baja energía e ir ajustando gradualmente los parámetros del problema que se desea resolver. Al realizar cambios lentos en estos parámetros, es más probable que el sistema alcance un estado de baja energía, que correspondería a una solución óptima. Este enfoque se utiliza principalmente para resolver problemas de optimización combinatoria, como la partición de grafos, la gestión del tráfico y la programación de tareas. En todos estos casos, el problema se puede expresar como una optimización binaria cuadrática no restringida, que luego puede resolverse utilizando plataformas de computación cuántica adiabática [51, 66]. En la actualidad, la empresa D-Wave Systems es referente en plataformas de computación cuántica adiabática y ofrece acceso a ellas a la comunidad científica a través de un servicio en la nube.

El modelo computacional más general y el que se ha considerado en la presente tesis, es el modelo basado en circuitos cuánticos. Dicho modelo se describe en detalle en las siguientes subsecciones de este capítulo.

Es de interés resaltar que cada modelo computacional tiene tres elementos comunes: los datos, las operaciones y los resultados. En el modelo de circuitos cuánticos, estos elementos se corresponden con los cúbits, las puertas cuánticas y las medidas [24].

Cúbit

Un cúbit (quantum bit o bit cuántico) es la unidad básica de información cuántica. Mientras que un bit clásico solo puede tener dos estados posibles, 0 y 1, un cúbit puede estar en una superposición de estos dos estados y puede representar información en una combinación lineal de ambos [74]. Por convención, la forma matricial de los estados de un cúbit es la siguiente:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (2.1)$$

De forma general, podemos representar un sistema de un cúbit con un estado arbitrario $|\psi\rangle$ como la combinación lineal de los vectores de base, $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$, siendo α y β números complejos y, además, con la condición de normalización $|\alpha|^2 + |\beta|^2 = 1$.

La esfera de Bloch es una esfera unitaria que proporciona una forma intuitiva de visualizar los estados de un cúbit. En dicha esfera, cada punto en la superficie de una esfera unitaria representa un estado cuántico posible del cúbit. El polo norte de la esfera representa el estado $|0\rangle$, el polo sur representa el estado $|1\rangle$, y los puntos en la superficie de la esfera representan superposiciones de ambos estados. La Figura 2.2 muestra la esfera de Bloch, donde la coordenada x representa la probabilidad de encontrar el cúbit en el estado $|0\rangle$ (spin arriba) o en el estado $|1\rangle$ (spin abajo); la coordenada y es la probabilidad de encontrar el cúbit en un estado superpuesto; y la coordenada z representa la fase relativa entre los estados $|0\rangle$ y $|1\rangle$.

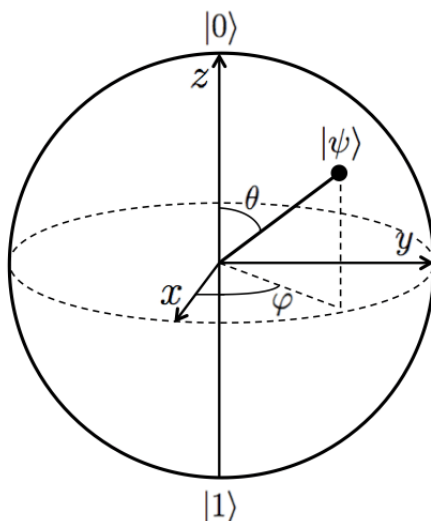


Figura 2.2: Esfera de Bloch donde cada punto de la superficie corresponde a un vector de estado normalizado que representa un estado cuántico válido.

Puertas cuánticas

Las puertas cuánticas son operaciones unitarias para manipular y transformar los estados de los cúbits. Se pueden expresar como matrices reversibles que actúan sobre uno o varios cúbits, transformando sus estados iniciales en otros [22]. En esta sección se van a describir brevemente las principales puertas cuánticas.

Las puertas Pauli son un conjunto de operadores unitarios que actúan sobre un único cúbit y que son fundamentales para el procesamiento y manipulación de información cuántica. Las puertas Pauli son hermitianas y autoduales, lo que significa que su inversa es igual a su adjunta, y que cada puerta es su propia autodual (es decir, su matriz es igual a su adjunta). Estas propiedades son cruciales para el funcionamiento de las computaciones cuánticas y su capacidad para formar una base universal para la manipulación de información cuántica. Existen tres tipos de puertas Pauli: X, Y y Z. Todas ellas están indicadas por matrices 2×2 que permutan el estado de un cúbit.

La puerta Pauli-X es el equivalente cuántico a la puerta NOT clásica [118], aunque negar un estado cuántico es más complicado que negar un bit. De hecho, esta analogía solo es válida si nos limitamos a los estados $|0\rangle$ y $|1\rangle$. Esta puerta realiza una rotación de π radianes alrededor del

eje x en la esfera de Bloch. El símbolo y la forma matricial de la puerta mencionada se pueden observar en la Figura 2.3.

$$\text{---} \boxed{X} \text{---} \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Figura 2.3: Símbolo y matriz de la puerta Pauli-X.

La puerta Pauli-Y realiza una rotación de π radianes alrededor del eje y en la esfera de Bloch. El símbolo y la forma matricial de esta puerta se pueden observar en la Figura 2.4.

$$\text{---} \boxed{Y} \text{---} \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

Figura 2.4: Símbolo y matriz de la puerta Pauli-Y.

La puerta Pauli-Z aplica una fase global de -1 al estado del cúbit, sin realizar una rotación física en la esfera de Bloch. El símbolo y la forma matricial de esta puerta se pueden observar en la Figura 2.5

$$\text{---} \boxed{Z} \text{---} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Figura 2.5: Símbolo y matriz de la puerta Pauli-Z.

La puerta identidad (I), que en algunas fuentes se incluye dentro de las puertas Pauli, es una puerta cuántica trivial que no realiza ninguna rotación o transformación en el estado de un cúbit. Se representa por la matriz de identidad de 2×2 que se muestra en la Figura 2.6. Es importante destacar que la puerta identidad es una de las puertas más simples en computación cuántica y que a menudo se utiliza como parte de circuitos más complejos para preservar el estado de ciertos cúbits en momentos específicos del proceso computacional.

$$\text{---} \boxed{I} \text{---} \quad I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Figura 2.6: Símbolo y matriz de la puerta identidad.

La puerta cuántica CNOT o NOT-Controlada fue propuesta originalmente en la década de 1980 por Charles H. Bennett y David P. DiVincenzo. Es una puerta de dos cúbits ampliamente

utilizada para aplicar una operación condicional en el segundo cúbit (objetivo) controlado por el estado del primer cúbit (control). Si el cúbit de control está en el estado $|0\rangle$, no hace ninguna rotación en el cúbit objetivo. Sin embargo, si su estado es $|1\rangle$, realiza una rotación de π alrededor del eje x en la esfera de Bloch del cúbit objetivo. El símbolo y la forma matricial de esta puerta se pueden observar en la Figura 2.7.

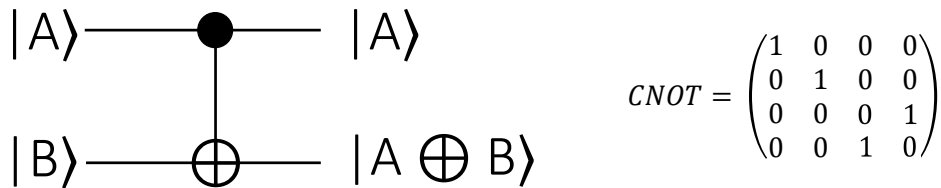


Figura 2.7: Símbolo y matriz de la puerta CNOT.

La puerta cuántica V [46], también conocida como puerta de raíz cuadrada de NOT (V-gate), es una puerta de dimension 1×1 , al igual que la puerta V^+ . La Figura 2.8 muestra tanto el símbolo como la forma matricial de las puertas V y V^+ .

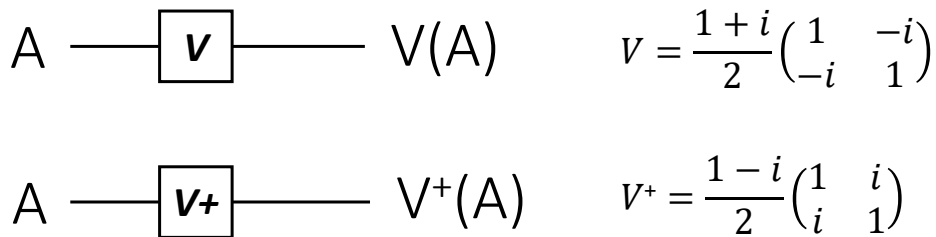


Figura 2.8: Símbolos y matrices de las puertas V y V^+ .

Las puertas controladas V y V^+ (también llamadas Controlled- V y Controlled- V^+ , respectivamente) son similares a las anteriores puertas V y V^+ , pero añadiendo un cúbit de control, como se comentó anteriormente en la puerta CNOT [46]. Estas puertas tienen una dimensión de 2×2 y pueden ser activadas o desactivadas utilizando el cúbit de control. La Figura 2.9 muestra los símbolos y la forma de sus matrices.

La puerta Toffoli [79] es una puerta similar a la CNOT, pero en lugar de tener un solo cúbit de control tiene dos, es decir, solo opera en el cúbit objetivo si los dos cúbits de control valen $|1\rangle$. Si se consideran las bases estándares $|1\rangle$ y $|0\rangle$ y el cúbit objetivo tiene el valor $|0\rangle$, esta puerta puede tener un uso similar a la puerta clásica NAND. El símbolo y la forma matricial de esta puerta se pueden observar en la Figura 2.10.

La puerta Hadamard se aplica sobre un solo cúbit cambiando el estado $|0\rangle$ a $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$ y el estado $|1\rangle$ a $\frac{|0\rangle-|1\rangle}{\sqrt{2}}$ creando así una superposición igualitaria de los dos estados básicos. El símbolo y la forma matricial de esta puerta se pueden observar en la Figura 2.11

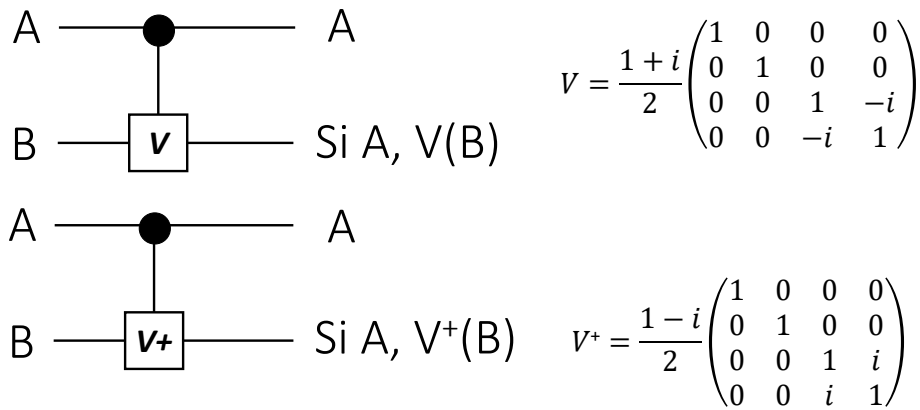


Figura 2.9: Símbolos y matrices de las puertas V y V^+ Controladas, que aplican la operación de V y V^+ al primer cúbit si el cúbit de control tiene el valor de $|1\rangle$.

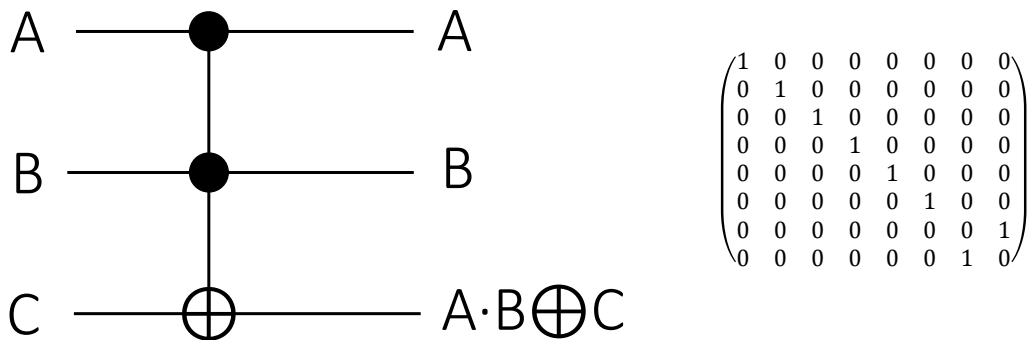


Figura 2.10: Símbolo y matriz de la puerta Toffoli, la cual niega el cúbit objetivo si ambos cúbit de control se encuentran en estado $|1\rangle$.

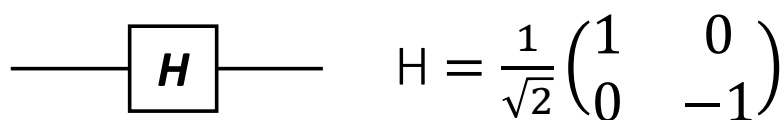


Figura 2.11: Símbolo y matriz de la puerta Hadamard, que se aplica a un cúbit y crea un estado de superposición.

La puerta Swap opera sobre dos cúbits, intercambiado los estados de ambos cúbits haciendo que el primero tome el valor del segundo y viceversa [118]. El símbolo y la forma matricial de esta puerta se pueden observar en la Figura 2.12.

La puerta Peres es reversible y posee tres entradas y tres salidas. Para las entradas A , B y C , las salidas son: $P = A$, $Q = A \oplus B$, y $R = A \cdot B \oplus C$, respectivamente. En la Figura 2.13 se muestra tanto la implementación como el símbolo de la puerta Peres.

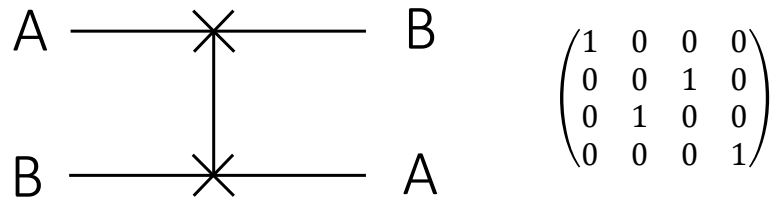


Figura 2.12: Símbolo y matriz de la puerta Swap la cual se aplica sobre dos cúbits e intercambia sus estados.

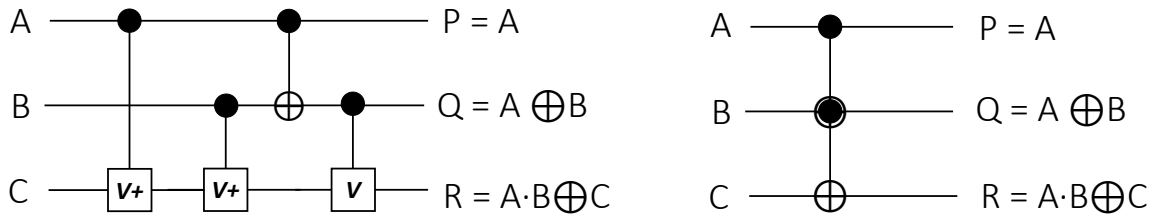


Figura 2.13: Implementación y símbolo de la puerta Peres.

También existen algunas puertas que aplican un giro de fase en la esfera de Bloch al cúbit sobre el que actúan, entre ellas destacan la puerta T y la S. Por un lado, las puertas T realizan una rotación de 45 grados alrededor del eje z en la esfera de Bloch. Por lo tanto, realiza una fase $\pi/4$. Puede verse su matriz equivalente en la Ecuación 2.2. La puerta T^+ es la inversa de la puerta T.

$$T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix} \quad (2.2)$$

Por otro lado, las puertas S realizan una rotación de 90 grados alrededor del eje z. De este modo, realizan una fase de $\pi/2$. Puede verse su matriz equivalente en la Ecuación 2.3. La puerta S^+ es la inversa de la puerta S.

$$S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \quad (2.3)$$

La puerta Temporary Logical-AND es una puerta alternativa a la Toffoli que se centra en reducir el coste de las puertas T [33], mencionadas anteriormente. La puerta Temporary Logical-AND tiene un coste cuántico mayor que el de la puerta Toffoli presentada en [74], pero ofrece mejoras significativas en las implementaciones orientadas a la tolerancia a fallos. La implementación de la puerta Temporary logical-AND se puede observar en la Figura 2.14 (arriba).

Métricas

En computación cuántica no existe una métrica estándar que indique la calidad de cada uno de los circuitos diseñados. Por lo tanto, resulta de interés el uso de un conjunto de métricas que proporcionen la mayor cantidad de información posible sobre cada circuito, como el propuesto por Mohammadi y otros en [70]. Las métricas más relevantes incluidas en dicho trabajo son:

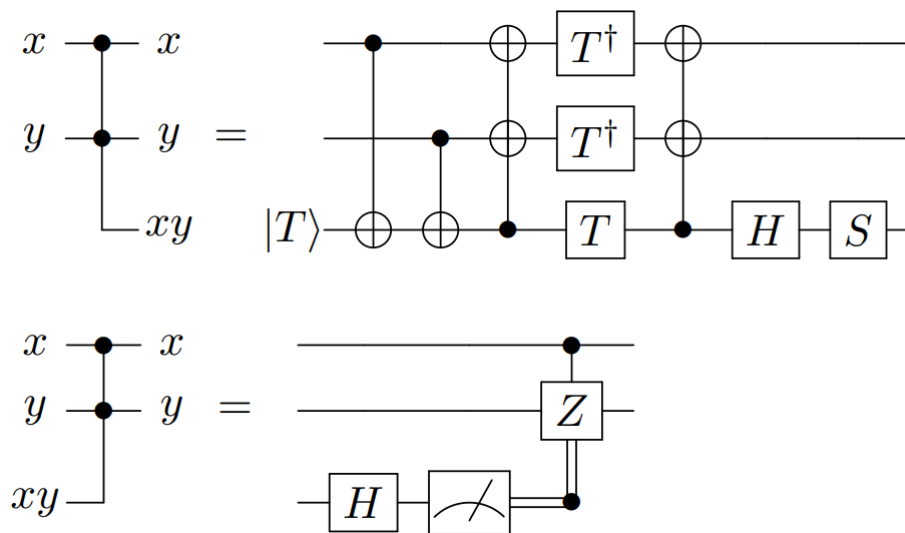


Figura 2.14: Puerta Temporary logical-AND y su implementación [33] (arriba) e implementación de la puerta de descomputación de la puerta Temporary Logical-AND, donde no hay ninguna puerta T involucrada (abajo).

- **Coste cuántico**, que se refiere a la cantidad total de puertas cuánticas necesarias para implementar un circuito. El coste cuántico de las puertas que actúan sobre uno o dos cúbits se define como 1. El coste cuántico de cualquier otra puerta es la suma de los costes cuánticos de las puertas que la componen.
- **Retardo cuántico**, que representa el número de etapas necesarias para ejecutar un circuito. El retardo de una puerta cuántica que actúa sobre uno o dos cúbits se establece a 1Δ .
- **Número de entradas ancilla (cúbits ancilla o cúbits auxiliares)**, que identifican los cúbits adicionales que se utilizan en un circuito cuántico para realizar operaciones auxiliares, pero que no contienen información útil al final del circuito.
- **Salidas basura**, que son cúbits adicionales que contienen valores desconocidos y no son relevantes para la solución final. Es necesario descomputar las salidas basura y devolverlas a su estado original para mantener la reversibilidad y coherencia de un circuito cuántico y, de este modo, garantizar un uso eficiente de los recursos cuánticos disponibles [54].
- **Puertas T**, que son las puertas que se utilizan para implementar códigos de corrección de errores y así garantizar la tolerancia a fallos en circuitos cuánticos. El principal inconvenientes que tienen es que son más costosas que el resto en términos de espacio y tiempo [72, 129]. En relación a las puertas T, las métricas interesantes son el **T-count** y el **T-depth**. T-count, es el número de puertas T; mientras que T-depth identifica el número de puertas T que deben actuar de forma secuencial.

Circuitos y algoritmos cuánticos

Un circuito cuántico es una secuencia de puertas cuánticas y operaciones aplicadas a cúbits para realizar una tarea específica en un sistema cuántico. Por tanto, se trata de una representación abstracta de las transformaciones cuánticas que se llevan a cabo durante la computación cuántica. Los circuitos cuánticos deben ser reversibles. La reversibilidad es una propiedad que significa

que todas las operaciones realizadas en un sistema cuántico pueden deshacerse completamente. De este modo, a partir de un resultado obtenido, es posible recuperar los valores originales [101, 125].

Los circuitos cuánticos son componentes fundamentales en la construcción de algoritmos cuánticos. En la actualidad, el diseño de circuitos y algoritmos cuánticos se enfrenta a diversos desafíos para hacer que la computación cuántica sea práctica y efectiva, especialmente en campos como el procesamiento de imágenes. Estos desafíos incluyen la limitación en la cantidad de cúbits disponibles en las plataformas cuánticas, la falta de compiladores que optimicen los circuitos para las particularidades de cada plataforma, la necesidad de librerías cuánticas optimizadas para algoritmos, la carencia de una metodología estándar para el desarrollo de algoritmos cuánticos y la complejidad de programar algoritmos cuánticos, que requieren una reescritura completa debido a las reglas de la mecánica cuántica. Además, los dispositivos cuánticos actuales son altamente sensibles al ruido (tanto interno como externo), lo que impulsa a los diseñadores a dedicar recursos a la detección y corrección de errores, aumentando aún más la demanda de recursos limitados. Por lo tanto, es esencial diseñar circuitos de tamaño reducido para que sean funcionales, especialmente cuando se prevé su integración en algoritmos más complejos.

Uno de los algoritmos cuánticos más importantes es el algoritmo de Shor, que puede factorizar números enteros grandes en sus componentes primos de manera significativamente más rápida que los algoritmos clásicos [103]. Otro algoritmo cuántico de gran interés es el algoritmo de Grover [38], que se utiliza para realizar búsquedas no estructuradas en bases de datos no ordenadas. Este algoritmo presenta una mejora cuadrática en la eficiencia frente a su homólogo en computación clásica. El algoritmo de Grover se basa en la amplificación cuántica, a través de un oráculo. Al aplicar iteraciones del algoritmo, el oráculo marca las entradas que son soluciones válidas. Esto se realiza mediante transformaciones cuánticas que aumentan la amplitud de las soluciones correctas, permitiendo que el algoritmo converja más rápido hacia la solución.

Como se comentó en el Capítulo 1.1, la implementación de algoritmos cuánticos utilizando el paradigma basado en circuitos es el que permite abordar problemas de computación general y, por tanto, es el más utilizado en la actualidad y en el que vamos a realizar avances en la presente tesis, concretamente en el segundo caso de estudio.

Procesado eficiente de imagen mediante computación cuántica

La integración de la computación cuántica en el procesamiento de imágenes ofrece nuevas perspectivas para abordar desafíos en el campo de QIMP, brindando métodos innovadores y eficientes para el análisis y procesamiento de imágenes.

En [48], se emplea el algoritmo de Grover para la búsqueda eficiente de píxeles con valores específicos en imágenes. Como se comentó anteriormente, este algoritmo posibilita búsquedas más rápidas en bases de datos no estructuradas. En el estudio mencionado, se aplicó el algoritmo de Grover para identificar los píxeles más oscuros en una imagen en escala de grises de tamaño 2×2 , primero en una imagen clásica generada mediante Python y luego en una imagen cuántica. La investigación demostró que la búsqueda no estructurada de Grover presenta una complejidad inferior en comparación con los esquemas clásicos. En concreto, la complejidad de la búsqueda

no estructurada de Grover es $O(2^n)$, mientras que los enfoques clásicos tienen una complejidad de $O(2^{2n+2m})$, donde m y n representan las dimensiones de la imagen. Este resultado sugiere un gran potencial para la utilización de la computación cuántica en el procesamiento y análisis de imágenes.

Otros ejemplos de aplicación de la computación cuántica en el procesamiento eficiente de imágenes incluyen la compresión eficiente de imágenes [133], la segmentación de imágenes mediante histogramas y umbralización utilizando la aceleración de los algoritmos de Grover y la transformada de Fourier [13], transformaciones homográficas de imágenes [56], operaciones morfológicas como erosión y crecimiento [128], y la comparativa cuántica entre imágenes [123].

3. Preprocesado de imagen

En este capítulo se exploran varias técnicas de preprocesado de imagen ampliamente utilizadas en la literatura. Estas técnicas nos permitirán mejorar la calidad visual, resaltar características relevantes y eliminar perturbaciones no deseadas en las imágenes. De este modo, las imágenes resultantes tendrán una mayor calidad para su posterior procesamiento y análisis en función del campo de aplicación.

A continuación, se detalla la estructura de este capítulo. En la Sección 3.1 se realiza una breve introducción al concepto de imagen y se explica la importancia del preprocesado de imágenes. En la Sección 3.2 se describen una serie de campos de aplicación donde el preprocesado de la imagen juega un papel fundamental. A continuación, en la Sección 3.3, se abordan técnicas generales de mejora de imagen, como, por ejemplo, la ecualización de histograma, que permite ajustar la distribución de los niveles de intensidad para mejorar el contraste. También se exploran algunos métodos de mejora del contraste y de realce de bordes, que permiten resaltar detalles y estructuras importantes en la imagen. Estas técnicas proporcionan herramientas valiosas para el preprocesamiento de imágenes, optimizando la calidad visual, resaltando características relevantes y preparando las imágenes para tareas más avanzadas de análisis y procesamiento. En la Sección 3.4 se detallan los distintos tipos de ruidos y algunos filtros lineales y no lineales para eliminarlos. En la Sección 3.5 se exploran varios métodos de calibrado y eliminación de distorsión en las imágenes. Se indica cómo corregir la distorsión geométrica causada por factores como la deformación de la lente o la perspectiva. También se discuten modelos de distorsión y métodos de calibración de cámara para rectificar y eliminar la distorsión en las imágenes. Finalmente, la última sección está dedicada a indicar cuáles han sido los métodos de preprocesamiento utilizados en la presente tesis.

3.1 Importancia del preprocesado de imagen

El término “imagen” se refiere a una función bidimensional de intensidad de color $I(x,y)$ donde x e y denotan las coordenadas espaciales. En el contexto de la computación y el procesamiento de imágenes, una “imagen” se refiere a un tipo específico de dato que ofrece una representación visual de objetos, escenas o información gráfica. Una imagen digital se compone de una cuadrícula de píxeles, y cada píxel tiene un valor numérico que define su color y su intensidad. Las imágenes digitales se utilizan para representar gráficos, fotografías, ilustraciones y otros tipos de contenido visual de manera que los computadores puedan procesarlo y mostrarlo [34, 86].

Las imágenes digitales pueden experimentar diversas distorsiones durante su adquisición, procesamiento, compresión, almacenamiento, transmisión y reproducción. Cualquiera de estos pasos puede provocar una degradación en la calidad visual de la imagen. Por ello, las técnicas de preprocesado, que ayudan a reducir o eliminar el ruido y otras distorsiones, son clave para obtener imágenes de mayor calidad.

El preprocesamiento de imágenes tiene como objetivo mejorar el aspecto de las imágenes y hacer más evidentes ciertos detalles que se desean resaltar. Las operaciones realizadas en el preprocesamiento de imágenes implican la alteración del histograma de la imagen o el procesamiento de los valores de sus píxeles. Las etapas de preprocesamiento y extracción de propiedades son fundamentales para el éxito de cualquier sistema de reconocimiento de modelos.

El preprocesado de imagen no se aplica de manera universal, ya que su implementación depende del tipo de imágenes y de la aplicación específica. Por ejemplo, en una aplicación de telemetría, la corrección de la perspectiva y la distorsión son críticas, mientras que en una aplicación de reconocimiento de patrones como un OCR, se pretende un preprocesado enfocado en el realce de bordes o esquinas. Sin embargo, existen algunas técnicas de propósito general que pueden ser aplicadas para mejorar la calidad de la imagen.

3.2 Campos de aplicación

En esta sección se describen algunas de las principales áreas de aplicación donde el preprocesado de imágenes resulta de gran interés para aumentar la fiabilidad de las imágenes analizadas: la medicina, la industria, la astronomía, la seguridad y la agricultura de precisión, entre otras.

En el ámbito de la medicina, el preprocesado de imagen se centra fundamentalmente en eliminar ruido, mejorar el contraste y resaltar regiones de interés para facilitar el diagnóstico médico. De este modo, es posible la mejora de la calidad en imágenes médicas como pueden ser: las tomografías computarizadas, las resonancias magnéticas y las imágenes de ultrasonido [32].

En la industria, el procesamiento de imágenes se utiliza para inspeccionar productos, detectar defectos y controlar la calidad en líneas de producción. El preprocesado de la información es clave, ya que de forma previa se aplican técnicas de mejora de imagen y eliminación de ruido para garantizar una visualización clara de los objetos y facilitar el análisis automatizado

de características específicas [106]. Son de interés también en la industria, las aplicaciones relacionadas con la realidad aumentada, la robótica y la visión por computador donde se utilizan técnicas avanzadas de preprocesado de imagen [95].

En astronomía, el preprocesado de imágenes juega un papel crucial en la mejora de la calidad de las imágenes capturadas por telescopios. Estas técnicas permiten eliminar el ruido generado por la atmósfera terrestre y resaltar detalles en objetos celestes, lo que ayuda a los astrónomos a realizar mediciones precisas y estudiar fenómenos cósmicos [109].

En el campo de la seguridad, el preprocesamiento de imágenes se utiliza para mejorar la calidad de las imágenes de vigilancia y facilitar la detección de objetos o personas de interés. Se aplican técnicas de mejora de contraste, eliminación de ruido y detección de bordes para mejorar la visualización de detalles relevantes y facilitar la identificación de posibles amenazas [76].

En todos los campos citados anteriormente y en cada vez más disciplinas, el preprocesamiento de imagen desempeña un papel fundamental para las posteriores etapas de procesamiento y de análisis de las imágenes. En esta tesis, nos hemos centrado en un problema relacionado con la agricultura de precisión, donde ha sido necesario realizar un preprocesado para optimizar la calidad visual de las imágenes y facilitar el posterior proceso de clasificación.

3.2.1 Agricultura de precisión e imágenes multi-espectrales

Un campo clave en esta tesis y donde el procesamiento de imagen es de especial interés es la agricultura de precisión [44]. La agricultura de precisión es un enfoque agrícola que utiliza tecnología avanzada, como sistemas de información geográfica, sensores, drones y software especializado, para optimizar la gestión de recursos en la agricultura.

Para que esta optimización se realice con éxito, es necesaria una monitorización del cultivo, que se lleva a cabo, en muchos casos, utilizando imágenes tomadas con cámaras multi-espectrales. Estas cámaras capturan información en diferentes bandas espectrales. Por tanto, una imagen multi-espectral consiste en una superposición espacial de diferentes fotografías en las múltiples bandas de interés dentro del espectro radioeléctrico. Debido a que cada banda espectral capta información diferente, las imágenes multi-espectrales pueden proporcionar datos precisos sobre la vegetación, la humedad del suelo y otros parámetros relevantes para la agricultura. Sin embargo, la calidad de las imágenes se ve comprometida debido a diversos factores: las condiciones de iluminación no controladas en entornos agrícolas, las diferencias en alturas y volúmenes de las plantas, etc. Por lo tanto, el preprocesado de estas imágenes multi-espectrales es de interés para después computar imágenes fiables y obtener un buen diagnóstico y una posterior toma de decisiones correcta. Este preprocesado consiste en la eliminación de distorsiones y errores de perspectiva en las imágenes, así como la supresión del ruido debido a factores como la mala iluminación, entre otros. En la agricultura de precisión, las imágenes multi-espectrales se pueden utilizar para monitorizar el crecimiento de los cultivos, la salud de las plantas, la calidad del suelo, o discernir entre cultivo principal y maleza, como se pretende realizar en el primer caso de estudio de esta tesis.

3.3 Técnicas de mejora de imagen

En esta sección se presentan las principales técnicas de mejora de la imagen. Estas técnicas tienen un rol fundamental a la hora de resaltar características relevantes y de mejorar la calidad visual de las imágenes.

3.3.1 Ecuación de histograma

La técnica de ecualización de histograma tiene como objetivo la mejora del contraste y de la distribución de los niveles de intensidad en una imagen.

El histograma de una imagen representa la frecuencia de aparición de los diferentes valores de intensidad presentes en la imagen. En un contexto discreto, se puede definir como una función $H(r_k)$ donde r_k representa un valor de intensidad específico y $H(r_k)$ es el número de píxeles en la imagen que tienen esa intensidad. La expresión matemática para definir el histograma se muestra en la Ecuación 3.1.

$$H(r_k) = \text{Número de píxeles con intensidad } r_k \quad (3.1)$$

Mediante la ecualización de histograma, buscamos redistribuir estos valores de intensidad de manera uniforme en todo el rango, lo que resulta en una imagen con un mejor contraste y una apariencia visual más equilibrada, al tener todos los niveles de gris la misma probabilidad de aparición.

El proceso de ecualización de histograma consta de dos etapas principales. En primer lugar, se calcula el histograma de la imagen original, lo que nos proporciona información sobre la distribución de intensidades. A continuación, se realiza una transformación en la imagen original utilizando el histograma acumulativo. Durante esta transformación, los valores de intensidad de la imagen se asignan a nuevos valores según la función acumulativa del histograma. Este proceso permite redistribuir los valores de intensidad de forma que los píxeles más oscuros se iluminen y los píxeles más claros se oscurezcan, lo que resulta en una imagen con un mejor contraste y una mayor cantidad de detalles visibles [34, 86].

La transformación de ecualización se calcula mediante la función de distribución acumulativa del histograma original. Si el histograma original es $H(r_k)$ y su función de distribución acumulativa correspondiente es $C(r_k)$, la transformación de ecualización T se calcula como:

$$T(r_k) = \frac{C(r_k) - \min(C)}{m \cdot n - \min(C)} \cdot (L - 1) \quad (3.2)$$

donde r_k es el valor de intensidad original, $C(r_k)$ es la función de distribución acumulativa del histograma original, $\min(C)$ es el valor mínimo de $C(r_k)$, L es el número de niveles de intensidad

en la imagen (por lo general 256 para imágenes en escala de grises) y m y n son las dimensiones de la imagen.

La transformación $T(r_k)$ se aplica a cada píxel en la imagen original para obtener la imagen ecualizada. La ecualización del histograma mejora el contraste y la visibilidad de detalles en la imagen al redistribuir las intensidades de manera uniforme.

Los "bins" generalmente se refieren a los intervalos en los que se divide el rango de intensidades de la imagen, L . Es posible expresar la función de distribución acumulativa $C(r_k)$ en términos de bins:

$$C(r_k) = \sum_{i=0}^{r_k} \text{hist}(i) \quad (3.3)$$

donde $C(r_k)$ se expresa como la suma de las frecuencias de los "bins" desde 0 hasta r_k , representando la función de distribución acumulativa del histograma original en términos de "bins".

La ecualización de histograma es particularmente efectiva en imágenes con un rango de intensidad limitado o en aquellas en las que los niveles de intensidad están sesgados hacia un extremo. Al aplicar esta técnica, se pueden resaltar detalles ocultos en áreas subexpuestas o sobreexpuestas, mejorando así la calidad visual y facilitando el procesamiento posterior de la imagen. Es importante tener en cuenta que la ecualización de histograma puede tener impacto en la apariencia global de la imagen. En algunos casos, puede generar un contraste excesivo o introducir artefactos no deseados. Por tanto, se recomienda utilizar técnicas adicionales de mejora de imagen y evaluar los resultados obtenidos para lograr el efecto deseado [34].

El Listado 3.1 muestra un código Matlab para llevar a cabo la operación de ecualización de histograma. Es posible realizar una ecualización de histograma traduciendo la imagen a un menor número de niveles de color discretizados, conocidos como bins.

```
1 I = imread("hist.jpg") ;  
  % Imagen original  
3 figure  
  subplot(1,2,1)  
5 imshow(I)  
  subplot(1,2,2)  
7 imhist(I)  
  % Imagen ecualizada en 10 bins  
9 nbins = 10;  
  J = histeq(I,nbins);  
11 figure  
  subplot(1,2,1)  
13 imshow(J)  
  subplot(1,2,2)  
15 imhist(J)  
  % Imagen ecualizada en 64 bins  
17 K = histeq(I); % Por defecto la funcion histeq() utiliza 64 bins  
19 figure  
  subplot(1,2,1)
```

```

imshow(K)
subplot(1,2,2)
imhist(K)

```

Listado 3.1: Código Matlab para el preprocesado basado en la ecualización de histograma

Las Figuras 3.1, 3.2 y 3.3 muestran dos ejemplos de ecualización de histograma sobre una imagen sobreexpuesta. En estas figuras se ha realizado un preprocesado sobre una imagen en escala de grises con una profundidad de color de 256 niveles de gris (equivalente a 8 bits por píxel). La Figura 3.1 representa la imagen original y su histograma sin modificar. La Figura 3.2 representa una imagen modificada donde el histograma se ha ecualizado utilizando 10 bloques de nivel de gris (bins). La Figura 3.3 representa la imagen original modificada utilizando 64 bins.

3.3.2 Ajuste del contraste

Es posible realizar manipulaciones más sofisticadas del histograma para aumentar o reducir la diferencia de intensidad entre los colores de una imagen. Dicha operación se conoce como ajuste del contraste y consiste en mapear el histograma a un rango mayor (aumento del contraste) o menor de valores (reducción del contraste).

Esta técnica es útil en imágenes que ya tienen una distribución relativamente uniforme de valores de histograma, pero que se encuentra concentrada en una región específica. Puede observarse un ejemplo de esta operación en la Figura 3.4, con su correspondiente código en el Listado 3.2. Para realizar un estiramiento del histograma (aumento del contraste) se ha mapeado el histograma original de tal manera que el 1% inferior se mapea a los valores más bajos, y el 1% superior se mapea a los valores más altos de color. De forma similar, es posible reducir el contraste comprimiendo el histograma entre valores más cercanos.

```

I = imread("hist.jpg");
% Imagen con histograma "estirado"
L = imadjust(I);
figure
subplot(1,2,1)
imshow(L)
subplot(1,2,2)
imhist(L)

```

Listado 3.2: Código Matlab para el ajuste de contraste.

Las fórmulas para realizar compresiones (reducción del contraste) y estiramientos del histograma (aumento del contraste) son las siguientes, respectivamente:

$$Comprimida(x,y) = \frac{L_{max} - L_{min}}{I_{max} - I_{min}} \times (I(x,y) - I_{min}) + L_{min} \quad (3.4)$$

$$Estirada(x,y) = \frac{I(x,y) - I_{min}}{I_{max} - I_{min}} \times (L_{max} - L_{min}) + L_{min} \quad (3.5)$$

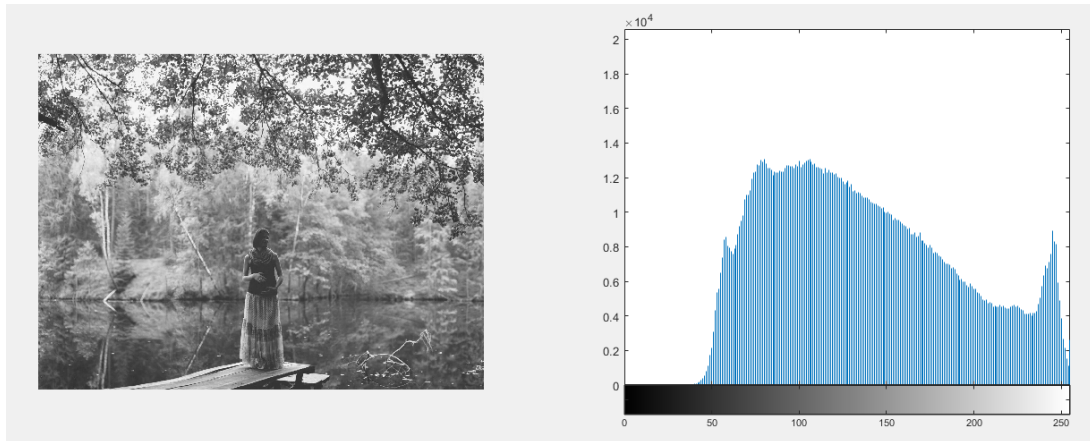


Figura 3.1: Imagen original con su histograma sin modificar. Se observa una ligera sobreexposición en la imagen. En el gráfico, el eje x indica cada uno de los 256 grados de nivel de gris posibles en la imagen, mientras que el eje y representa la distribución de probabilidad de ese color en términos del número de píxeles.

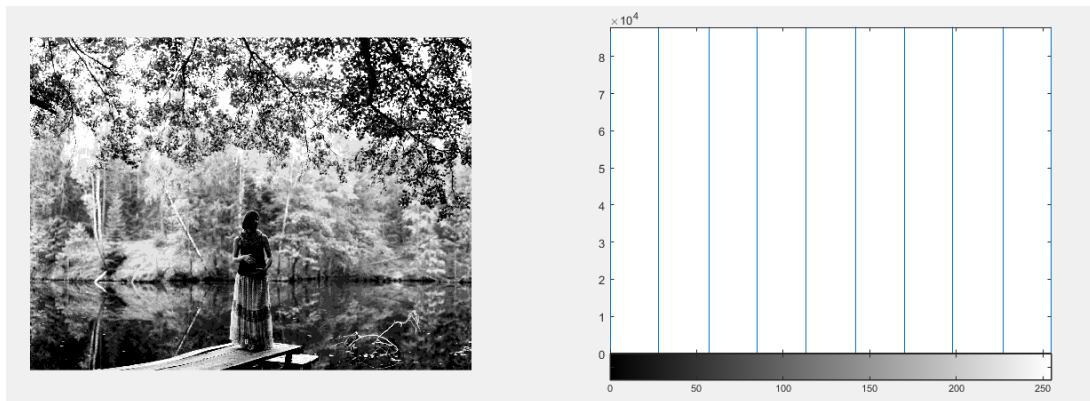


Figura 3.2: Imagen modificada con su histograma ecualizado en 10 bins planos. El eje x de la gráfica indica cada uno de los 10 grados de nivel de gris destino (bins). El eje y representa la distribución de probabilidad de ese color en términos del número de píxeles.

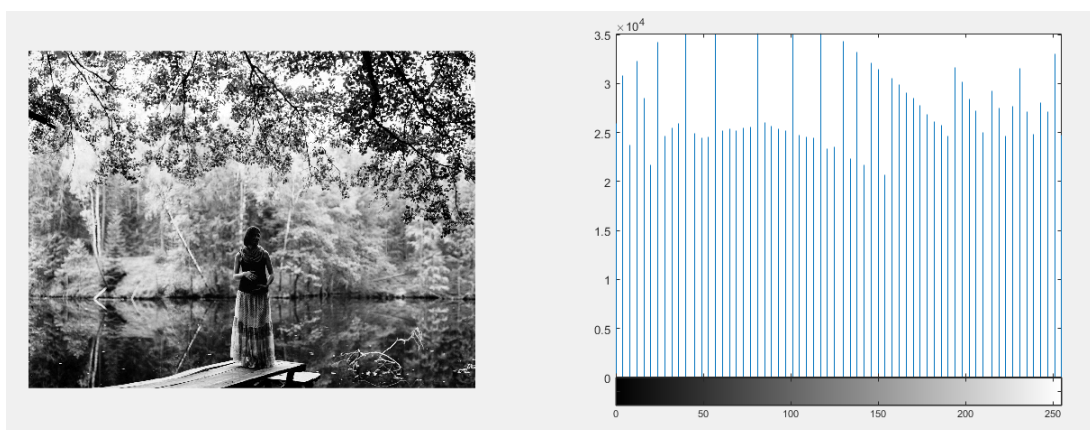


Figura 3.3: Imagen modificada con su histograma ecualizado en 64 bins relativamente planos. El eje x de la gráfica indica cada uno de los 64 grados de nivel de gris destino (bins). El eje y representa la distribución de probabilidad de ese color en términos del número de píxeles.

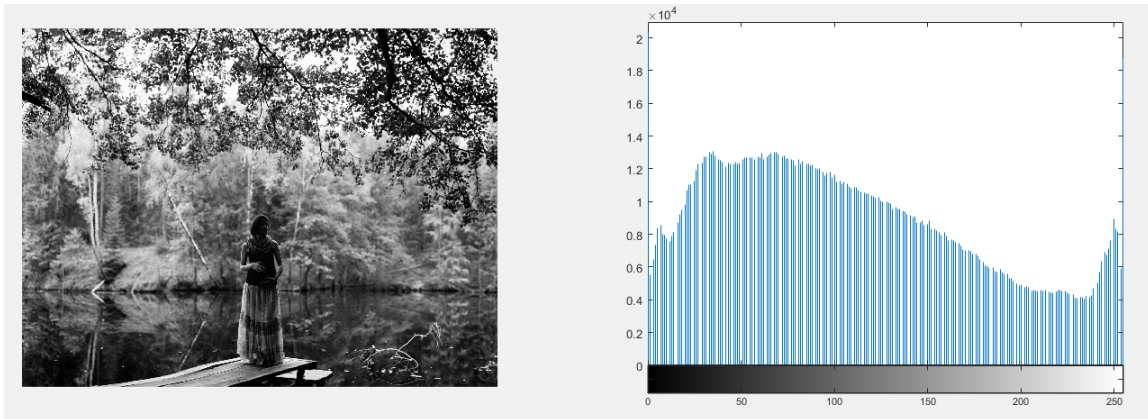


Figura 3.4: Imagen modificada con su histograma estirado. El contraste se ha aumentado al máximo posible sin perder información.

donde I_{\min} e I_{\max} representan los valores de intensidad mínimo y máximo de la imagen, y L_{\min} y L_{\max} representan los nuevos límites de valores máximos y mínimos del histograma de la imagen modificada. Nótese que $I(x,y)$ es el valor de la imagen original [34, 84].

Mejora del contraste local

A diferencia del ajuste global del contraste mediante técnicas globales de manipulación del histograma, como son la ecualización de histograma y el ajuste del contraste, la mejora del contraste local se enfoca en resaltar regiones específicas de la imagen en función de su contenido local. Esta técnica es especialmente útil cuando se tienen áreas con diferentes niveles de iluminación en la imagen, ya que permite mejorar la visibilidad de los detalles en esas áreas. Esta operación requiere definir la región de interés y aplicar correcciones a su histograma particular [34, 84].

3.3.3 Ajuste del brillo

Mediante la manipulación del histograma también es posible aumentar y reducir el brillo de una imagen. Para ello, se desplaza el histograma de la imagen a la izquierda para oscurecerla, y hacia la derecha para aclararla, ajustando la luminosidad de toda la imagen. Esto puede lograrse reduciendo o aumentando los niveles de intensidad del color originales.

La Figura 3.5 muestra un ejemplo donde se ha oscurecido la Figura 3.1 utilizando esta técnica. Además, el código Matlab asociado a esta reducción del brillo en 50 grados de color se muestra en el Listado 3.3.

```

1 I = imread("hist.jpg");
3 % Imagen con histograma oscurecido (reduccion del brillo)
M = I - 50;
5 figure

```

```

subplot(1,2,1)
7 imshow(M)
subplot(1,2,2)
9 imhist(M)

```

Listado 3.3: Código Matlab para el preprocesado basado en el desplazamiento del histograma. El código disminuye el brillo en 50 niveles de color.

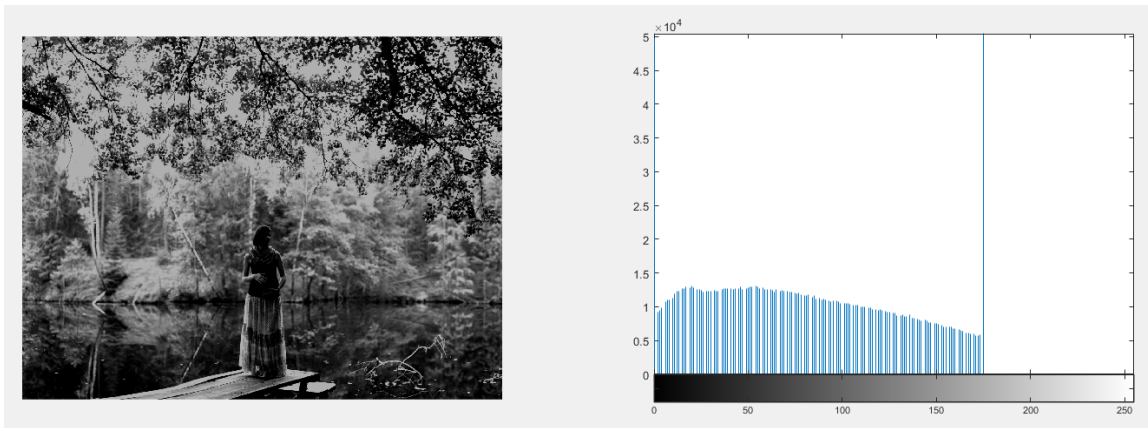


Figura 3.5: Imagen modificada con su histograma desplazado a la izquierda 50 niveles de color, resultando en una disminución del brillo.

Las fórmulas para realizar desplazamientos del histograma resultantes de aclarar y oscurecer la imagen son las siguientes, respectivamente:

$$Aclarada(x,y) = I(x,y) + valor \quad (3.6)$$

$$Oscurecida(x,y) = I(x,y) - valor \quad (3.7)$$

donde *valor* es el nivel de brillo que se desea aclarar u oscurecer e $I(x,y)$ identifica el valor de la imagen original [34, 84].

3.3.4 Saturación y realce de color

Es frecuente que las imágenes a procesar no contengan una sola banda, sino que se presenten en formato RGB (Rojo, Verde, Azul), donde existen tres canales diferentes que pueden ser manipulados. Esto permite el uso de técnicas de realce de color, con el objetivo de resaltar uno de los canales por encima de los demás.

El realce de color tiene como objetivo mejorar la saturación y la viveza de los colores en una imagen. Esto se logra mediante la manipulación selectiva de los canales de color para resaltar tonalidades específicas. El realce de color puede ser útil en aplicaciones como la fotografía, donde se busca resaltar los detalles y la calidad visual de la imagen [34, 84].

La saturación se puede interpretar como la medida de la dispersión de los canales de color en el espacio RGB. Un valor de saturación alto indicaría una mayor variabilidad y, por lo tanto, una mayor intensidad o vivacidad en el color. Mientras que un valor de saturación bajo indicaría una menor variabilidad y, por lo tanto, colores más suaves o menos intensos.

La saturación en una imagen RGB se calcula como la desviación estándar entre las coordenadas R (roja), G (verde) y B (azul). El punto más alto de saturación del color se alcanza cuando un canal está al 100% de intensidad, y los otros dos al 0%. Si definimos μ como la luminosidad, la fórmula para calcular la saturación sería la siguiente:

$$\sigma(R, G, B) = \sqrt{\frac{(R - \mu)^2 + (G - \mu)^2 + (B - \mu)^2}{3}} \quad (3.8)$$

En esta fórmula, las expresiones $(R - \mu)$, $(G - \mu)$ y $(B - \mu)$ representan las diferencias entre los valores de cada canal de color (rojo, verde y azul) y la luminosidad μ . La luminosidad μ se calcula como la media de los valores de los tres canales para un píxel específico, considerando los valores de cada canal en ese píxel individualmente. Es decir, la fórmula considera las diferencias con respecto a la luminosidad promedio de cada píxel, no el promedio de los canales en la imagen en su totalidad.

De esta manera, la saturación medida indica cuánto difieren los valores de los canales de color de su promedio. Un valor de saturación alto indica una mayor diferencia entre los canales de color y una mayor intensidad de ese color. Mientras que un valor de saturación bajo indica una menor diferencia y, por lo tanto, colores más suaves o menos intensos.

La Figura 3.7 muestra un ejemplo de aumento del 50% de la saturación sobre una imagen RGB original (Figura 3.6). Concretamente, se ha aumentado un 50% la desviación estándar de la luminosidad en los tres canales respecto a la imagen original, utilizando Adobe Photoshop.



Figura 3.6: Imagen original sin modificar su saturación. Esta imagen corresponde a una fotografía de cultivos de girasol en fase temprana.

Figura 3.7: Imagen tras editar la Figura 3.6 aumentando la saturación en un 50%.

3.3.5 Otras técnicas de mejora de la imagen

El procesado de imagen y la visión artificial cuentan con innumerables aplicaciones, siendo algunas de ellas extremadamente específicas. Además de las mejoras generales anteriormente mencionadas, existen diferentes técnicas para mejorar la imagen en situaciones muy concretas. Algunos ejemplos específicos pueden ser la mejora de imágenes mediante Inteligencia Artificial (IA), la reducción del desenfoque mediante técnicas de restauración de imagen como convolución ciega o restauración basada en desenfoque inverso, la optimización de la paleta de colores para mejorar la percepción humana, etc. A continuación se describen algunas de estas técnicas.

Mejora de imagen utilizando Inteligencia Artificial (DLSS)

Un método para mejorar las imágenes utilizando IA es el empleo de Deep Learning Super Sampling (DLSS), una técnica desarrollada por NVIDIA con el propósito de mejorar la calidad y resolución de las imágenes en los gráficos de videojuegos en tiempo real [75].

Gracias al uso de redes neuronales de aprendizaje profundo, DLSS aprovecha el poder del aprendizaje automático para generar imágenes más detalladas y nítidas a partir de imágenes de menor resolución. En lugar de renderizar directamente a la resolución final deseada, DLSS procesa la imagen a una resolución más baja y, posteriormente, utiliza una red neuronal para reconstruirla y mejorar su calidad. DLSS emplea una técnica conocida como supermuestreo, la cual genera una imagen de mayor resolución utilizando información de imágenes de menor resolución. Esto se logra mediante el entrenamiento de una red neuronal con un amplio conjunto de imágenes de alta resolución y sus versiones correspondientes de baja resolución. La red neuronal aprende a inferir detalles y características ausentes en la imagen de baja resolución, generando una imagen final de mayor calidad. De este modo, en el contexto de los videojuegos DLSS permite disfrutar de gráficos de alta calidad y resolución, incluso en hardware menos potente, al reducir la carga de procesamiento requerida para renderizar imágenes a resoluciones nativas más altas. Esta técnica disminuye la carga computacional del juego y mejora su rendimiento, lo que se traduce en una experiencia de juego mejorada para los usuarios.

La Figura 3.8 muestra una imagen original de un videojuego. Sobre ella se ha aplicado un reescalado y una mejora de la imagen mediante la técnica DLSS, obteniéndose como resultado la imagen que se muestra en la Figura 3.9.

Restauración de imagen mediante deconvolución ciega

A menudo nos enfrentamos a la necesidad de mejorar imágenes que presentan desenfoque o movimientos no deseados. En este contexto, se recurre a técnicas de reconstrucción, incluyendo diversos algoritmos de deconvolución ciega. En [57], se profundiza en el tema de la deconvolución ciega y su aplicación en la corrección de imágenes borrosas. El objetivo es recuperar la imagen original a partir de una imagen borrosa sin requerir conocimiento previo sobre el proceso de desenfoque.



Figura 3.8: Imagen original del videojuego, input del algoritmo DLSS [75].



Figura 3.9: Imagen reescalada y mejorada, output del algoritmo DLSS [75].

La Figura 3.10 (extraída de [57]) muestra dos ejemplos de aplicación de convolución ciega. En la Figura 3.10 (a) se observa la imagen de entrada para los algoritmos de deconvolución, y en la Figura 3.10 (b) su mapa de verdad (la imagen que realmente se desea obtener). En las imágenes de la Figura 3.10 (c) y (d) se observan los resultados de dos técnicas de deconvolución ciega, propuestas por los artículos [27, 99].

3.4 Técnicas de eliminación de ruido

En esta sección se proporciona una descripción de los diversos tipos de ruido que pueden estar presentes en imágenes. Posteriormente, se presenta un listado de filtros, tanto lineales como no lineales, que se pueden emplear para llevar a cabo su corrección.

3.4.1 Tipos de ruido

El ruido en una imagen representa una variación aleatoria en la información de brillo o color, y suele estar relacionado con interferencias electrónicas. Este tipo de ruido puede ser ocasionado por el sensor y la electrónica presente en dispositivos como escáneres o cámaras digitales. En el ámbito de la fotografía analógica, el ruido también puede surgir debido al grano de la película, mientras que en la fotografía digital, el ruido de disparo es inherente a los detectores de fotones. El ruido en una imagen se considera un subproducto no deseado en el proceso de captura de imágenes, ya que puede oscurecer la información relevante. En la fotografía digital, los tipos de ruido más comunes incluyen el ruido gaussiano, el ruido de disparo, el ruido periódico, el ruido speckle y el ruido de lectura del sensor [15]. A continuación, se proporciona una descripción detallada de estos tipos, así como de sus causas subyacentes.

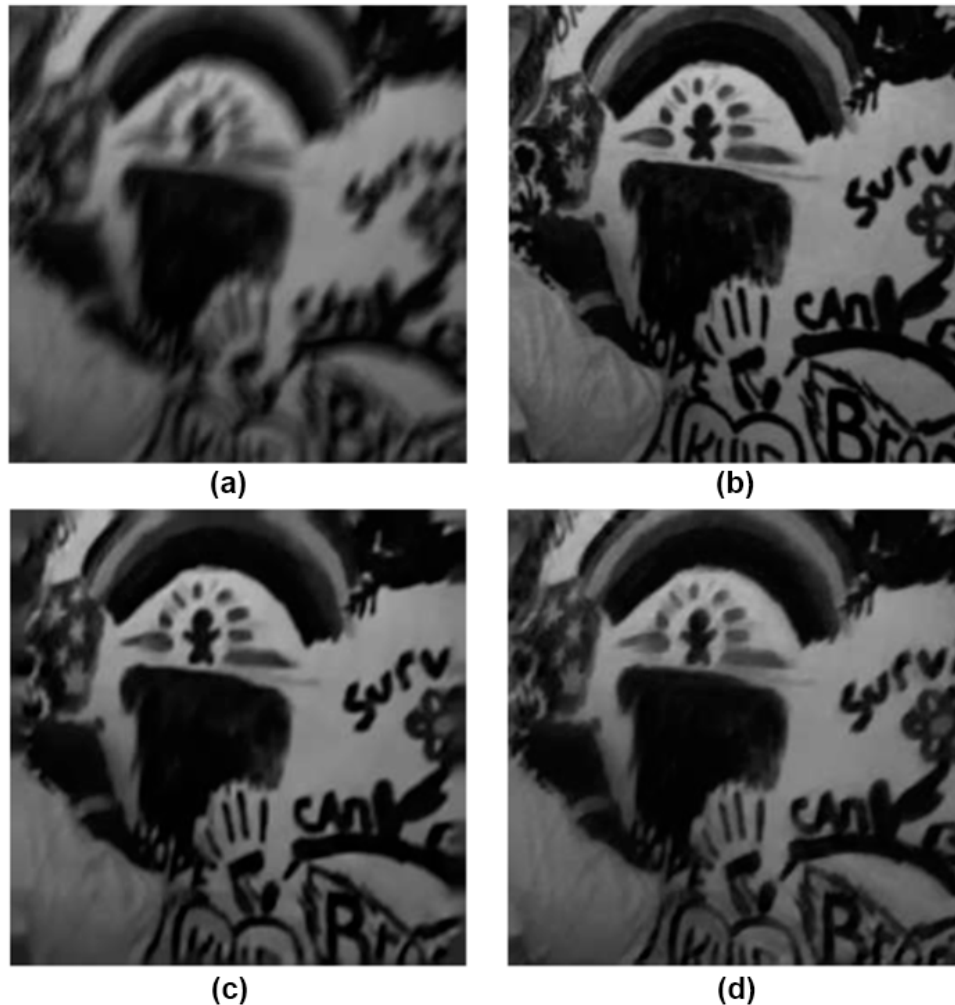


Figura 3.10: Ejemplo de deconvolución ciega extraído de la referencia [57]. (a) Input de imagen borrosa para algoritmos de deconvolución ciega. (b) Mapa de verdad de la imagen para evaluar la calidad de las técnicas empleadas. (c) Imagen restaurada mediante un algoritmo de deconvolución ciega basado en la técnica propuesta en [27]. (d) Imagen similar restaurada mediante la técnica propuesta en [99].

El ruido gaussiano sigue una distribución normal y es similar a las fluctuaciones aleatorias en la imagen. Se asemeja a puntos aleatorios de brillo en la imagen y suele ser causado por la interferencia electrónica en los dispositivos de captura de imágenes. Las principales fuentes de ruido gaussiano en las imágenes digitales surgen durante la adquisición. El sensor tiene un ruido inherente debido al nivel de iluminación y su propia temperatura, y los circuitos electrónicos conectados al sensor introducen su propia parte de ruido electrónico del circuito [15].

El ruido dominante en las partes más brillantes de una imagen capturada por un sensor de imagen suele ser causado por fluctuaciones cuánticas estadísticas, es decir, la variación en el número de fotones detectados a un nivel de exposición dado. Este ruido se conoce como ruido de disparo de fotones o ruido “shot”. El ruido de disparo de fotones sigue una distribución de Poisson, que puede aproximarse mediante una distribución gaussiana para intensidades de imagen grandes. El ruido de disparo de fotones tiene una desviación estándar proporcional a la

raíz cuadrada de la intensidad de la imagen, y el ruido en diferentes píxeles es independiente entre sí [67].

El ruido periódico exhibe patrones repetitivos en la imagen. Puede deberse a interferencias eléctricas u otras fuentes y se manifiesta como bandas o estructuras repetitivas en la imagen. En el dominio de la frecuencia, este tipo de ruido aparece como picos discretos. Se puede lograr una reducción significativa de este ruido aplicando filtros de rechazo de frecuencia en el dominio de Fourier [34].

El ruido speckle se caracteriza por tener una apariencia granulada que se asemeja a pequeñas manchas o motas en la imagen. Es común en imágenes médicas y de radar y es causado por la interferencia constructiva y destructiva de las ondas. El ruido speckle es un ruido multiplicativo que resulta de las fluctuaciones aleatorias de las señales reflejadas en una superficie [87].

Otros tipos de ruido son inherentes al propio dispositivo, como es el caso del ruido de lectura, que se debe a la mala interpretación de la señal eléctrica sobre la captación de fotones por parte del sensor. El ruido de cuantificación es muy similar al ruido de lectura y aparece por el redondeo de los valores de los fotones detectados. Finalmente, existen otros tipos de ruido, como el ruido térmico por el calentamiento de la propia cámara, ruido por la sensibilidad no uniforme de los píxeles del sensor, o errores de transmisión o conversión de datos internos, como el ruido de “sal y pimienta” [34].

En la Figura 3.11 se ilustran dos ejemplos visuales de tipos de ruido (gaussiano y “sal y pimienta”).

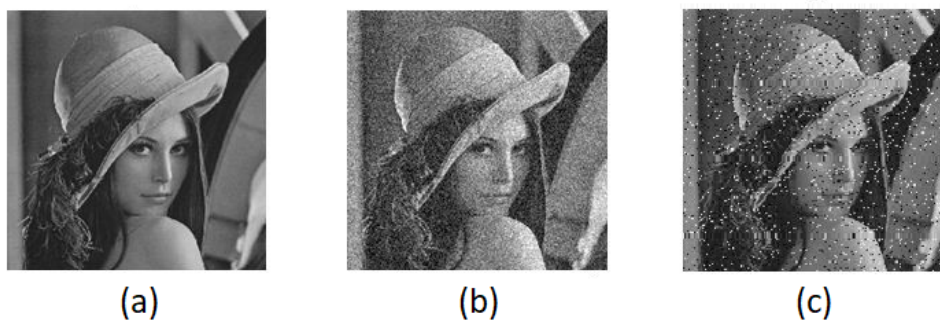


Figura 3.11: (a) Imagen original. (b) Imagen con ruido gaussiano. (c) Imagen con ruido "sal y pimienta".

3.4.2 Filtros lineales para la eliminación del ruido

Filtro de la media

El filtro de la media se utiliza para suavizar una imagen y reducir el ruido promediando los valores de píxeles en una vecindad. Se aplica una máscara de cierto tamaño que promedia el valor de cada píxel según sus vecinos. La ecuación del filtro de la media en una vecindad de

$n \times n$ píxeles es:

$$G(x,y) = \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n I(x+i-1, y+j-1) \quad (3.9)$$

donde $G(x,y)$ representa el valor del píxel filtrado en la posición (x,y) , n es el tamaño de la vecindad (por ejemplo, 3×3 o 5×5) e $I(x+i-1, y+j-1)$ es el valor del píxel original de una posición de la vecindad [34].

Filtro gaussiano

El filtro gaussiano es una técnica de procesamiento de imágenes que se utiliza para suavizar o difuminar una imagen al aplicar una convolución con una función de distribución gaussiana. Su propósito principal es reducir el ruido y las pequeñas variaciones en la intensidad de los píxeles, lo que resulta en una imagen más suave y menos rugosa. Al aplicar una máscara de distribución gaussiana, los píxeles más cercanos tienen más peso sobre el valor final aplicado. La ecuación general del filtro gaussiano es:

$$G(x,y) = \frac{1}{2\pi\sigma^2} \sum_{i=-k}^k \sum_{j=-k}^k e^{-\frac{i^2+j^2}{2\sigma^2}} \cdot I(x+i, y+j) \quad (3.10)$$

donde $G(x,y)$ representa el valor del píxel filtrado en la posición (x,y) , σ corresponde a la desviación estándar del filtro gaussiano, que controla el grado de suavizado o difuminación aplicado a la imagen. Esta desviación estándar está relacionada con la función gaussiana utilizada para ponderar los píxeles en la vecindad cuando se aplica el filtro, y determina el tamaño y la forma de la máscara de filtrado. El valor de k es el tamaño de la vecindad (por ejemplo, 3×3 o 5×5), e $I(x+i, y+j)$ son los valores de píxeles originales en las posiciones $(x+i, y+j)$ [34].

3.4.3 Filtros no lineales para la eliminación del ruido

Filtro de máximos y filtro de mínimos

El filtro de máximos reemplaza el valor del píxel en el centro de la máscara por el valor máximo en esa vecindad definida por la máscara aplicada. De forma similar, el filtro de mínimos busca el valor mínimo dentro de la máscara y aplica ese valor al píxel objetivo. Este tipo de filtros es muy útil para eliminar el ruido de tipo “sal y pimienta” en una imagen. Las ecuaciones que describen estos filtros son las siguientes:

$$G(x,y) = \max_{i,j}[I(x+i, y+j)] \quad (3.11)$$

$$G(x,y) = \min_{i,j}[I(x+i, y+j)] \quad (3.12)$$

donde $G(x,y)$ es el valor del píxel filtrado en la posición (x,y) , $I(x+i, y+j)$ representa los valores de píxeles en la vecindad definida por la máscara aplicada, con (i, j) recorriendo todos los píxeles dentro de la máscara y \max (o \min) denota la función que busca el valor máximo (o mínimo) entre todos los valores de píxeles dentro de la máscara.

Filtro de la mediana

El filtro de la mediana se utiliza para reemplazar el valor del píxel en el centro de la máscara por el valor de la mediana de esa vecindad definida por la máscara. Es muy útil para el ruido tipo “sal y pimienta” y, normalmente, se implementa de forma nativa en multitud de aplicaciones de fotografía. La ecuación que describe este filtro es la siguiente:

$$G(x,y) = med_{i,j}[I(x+i,y+j)] \quad (3.13)$$

donde *med* representa la función que busca el valor mediano entre todos los valores de píxeles dentro de la máscara y el resto de la notación utilizada ya fue descrita anteriormente.

3.5 Técnicas de calibrado y eliminación de la distorsión

La corrección de la distorsión de imágenes en cámaras ha sido un tema de gran importancia desde que los usuarios desean reproducir o utilizar de manera precisa la información recogida por estos dispositivos. Inicialmente, esta corrección se utilizaba principalmente en aplicaciones de cartografía. Aunque esta tarea continúa siendo relevante en la actualidad, también se requiere una calibración precisa de las cámaras en otras aplicaciones, como los diferentes métodos de medición bidimensional y tridimensional. En el pasado, las cámaras utilizadas eran escasas y costosas, pero hoy en día es común que cualquier dispositivo de uso común o proceso industrial cuente con múltiples cámaras (pudiendo ser estas de bajo coste) para llevar a cabo tareas de medición o de captura de datos [17]. Estos avances tecnológicos a nivel de cámaras han permitido una mayor disponibilidad y accesibilidad para los usuarios, lo que ha incrementado el número de aplicaciones que requieren una calibración precisa de la cámara.

La corrección de la distorsión de imágenes es fundamental para garantizar la precisión y la fiabilidad de los resultados obtenidos a partir de las imágenes capturadas. Al eliminar las distorsiones geométricas y de lente, se puede lograr una representación más precisa del mundo real y mejorar la calidad de las mediciones y las interpretaciones realizadas a partir de las imágenes.

A lo largo de esta sección se describirán diversas técnicas utilizadas para la identificación, modelado y corrección de la distorsión de imágenes, con el objetivo de garantizar la precisión y fiabilidad de los datos obtenidos.

3.5.1 Modelo de cámara

El modelo de cámara más elemental consiste en la cámara estenopeica, el cual se basa en un diseño simple de una cámara. En este modelo, la cámara consiste en una caja oscura con un pequeño agujero o apertura en un extremo y una superficie fotosensible (como una película o un sensor) en el otro extremo. El agujero actúa como una especie de lente, permitiendo que la luz entre a la cámara en forma de rayos.

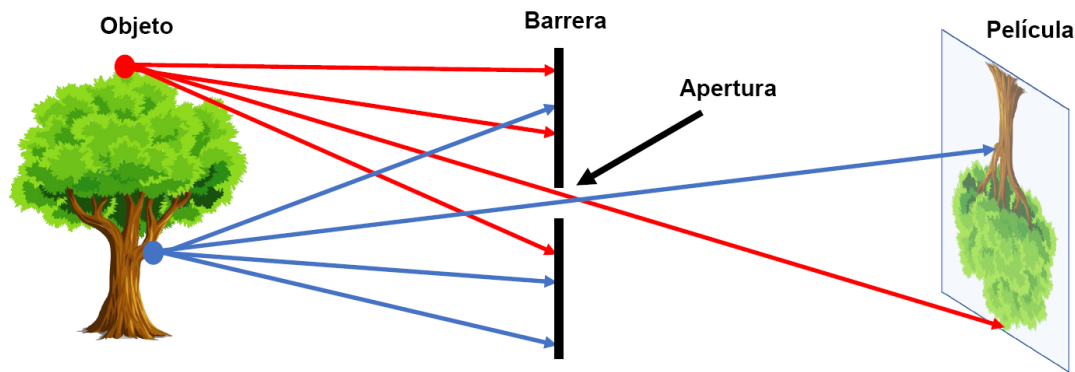


Figura 3.12: Ejemplo de cámara estenopeica. Todos los puntos del objeto pasan por una única apertura a través de la barrera definida [40].

Cuando un objeto se encuentra frente al agujero, los rayos de luz emitidos por cada punto del objeto pasan a través del agujero y forman una imagen invertida en la superficie fotosensible. Debido al tamaño diminuto del agujero, los rayos de luz se dispersan y se intersecan en la superficie fotosensible, creando una imagen nítida en el plano focal. Este proceso puede observarse en la Figura 3.12.

Los parámetros de la cámara estenopeica son representados en una matriz llamada matriz de la cámara, denotada por P . Esta matriz mapea la escena tridimensional del mundo en el plano de la imagen.

$$w \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = P \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (3.14)$$

Los puntos de la imagen (vector columna $[x, y, 1]$) escalados con el factor w se obtienen como el producto de la matriz de la cámara por los puntos reales tridimensionales representados por el vector columna $[x, y, z, 1]$. La matriz de la cámara es igual a K (la matriz intrínseca) multiplicada por el vector fila $[R, t]$ (los parámetros extrínsecos, rotación y traslación). Los parámetros extrínsecos son los parámetros de posición y orientación de la cámara en el espacio tridimensional. El vector fila consta de la matriz de rotación R y el vector de traslación t . Estos parámetros describen la ubicación y orientación de la cámara en relación con el sistema de coordenadas del mundo [131]:

$$P = K [R \ t] \quad (3.15)$$

Este modelo de cámara carece de lentes convencionales y sistemas de enfoque, lo que resulta en imágenes con una profundidad de campo amplia. Sin embargo, también tiene algunas limitaciones, como la necesidad de exposiciones más largas debido a la cantidad limitada de luz que pasa a través del agujero. En el modelo de cámara estenopeica, uno de los aspectos clave a considerar es el tamaño de la apertura. A medida que aumenta el tamaño de la apertura, más rayos de luz logran pasar a través de la barrera. Sin embargo, esta mayor cantidad de luz puede tener un efecto negativo en la imagen capturada, ya que cada punto en la película puede verse

afectado por rayos de luz provenientes de múltiples puntos en el espacio tridimensional. Esto resulta en una imagen difuminada, como puede observarse en la Figura 3.13.

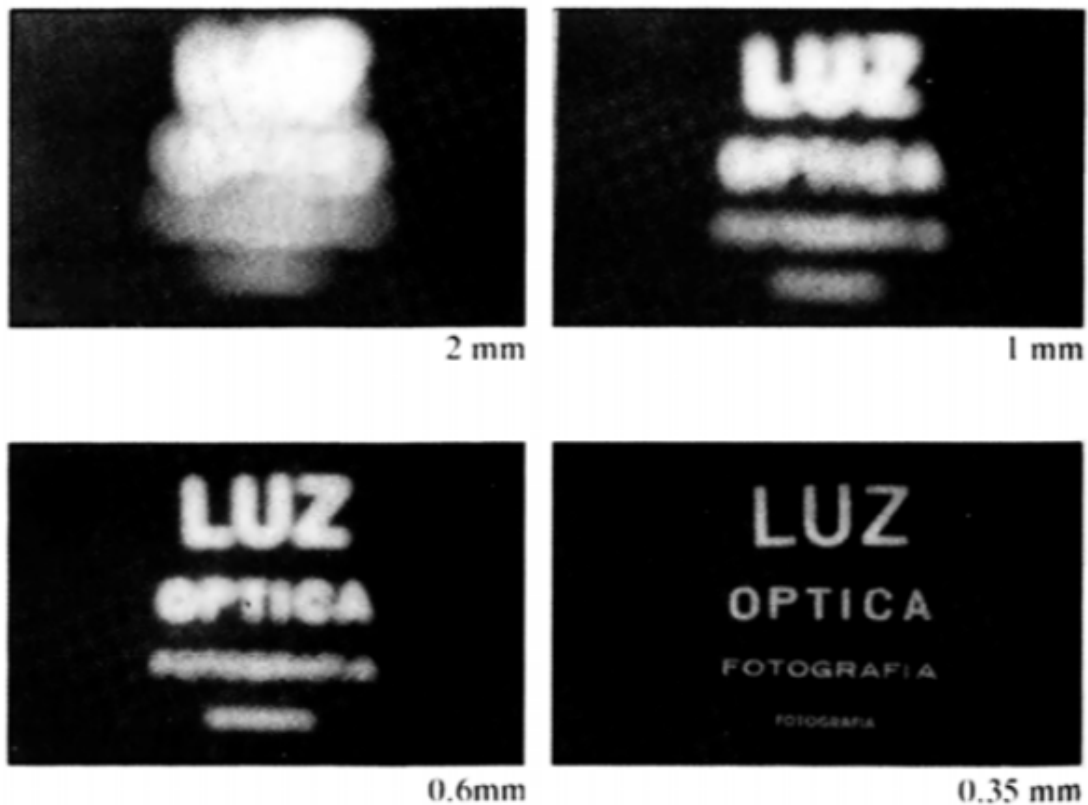


Figura 3.13: Ejemplos de nitidez de una imagen capturada por una cámara estenopeica. Tamaños más pequeños de apertura aportan nitidez, pero eliminan cantidad de luz [19].

Por otro lado, si se considerase la opción de reducir al mínimo el tamaño de la apertura para obtener imágenes más nítidas, es importante tener en cuenta que un tamaño de apertura más pequeño permite el paso de menos rayos de luz, lo que resulta en imágenes más nítidas pero también más oscuras.

Por lo tanto, encontrar el equilibrio adecuado entre nitidez y brillo en las imágenes capturadas con una cámara estenopeica es un desafío. Este conflicto puede resolverse mediante el uso de lentes, dispositivos que pueden enfocar o dispersar la luz. Si reemplazamos el agujero estenopeico con una lente colocada y dimensionada correctamente, se cumple la siguiente propiedad: todos los rayos de luz emitidos por un punto P son refractados por la lente de manera que convergen hacia un punto único, el punto focal. Esto puede observarse en la Figura 3.14, donde cada rayo de luz paralelo al eje óptico converge en el punto focal por acción de la lente, y, posteriormente, se traduce en el punto P' del receptor óptico o película de la cámara. La distancia f hasta el punto focal se denomina distancia focal [31].

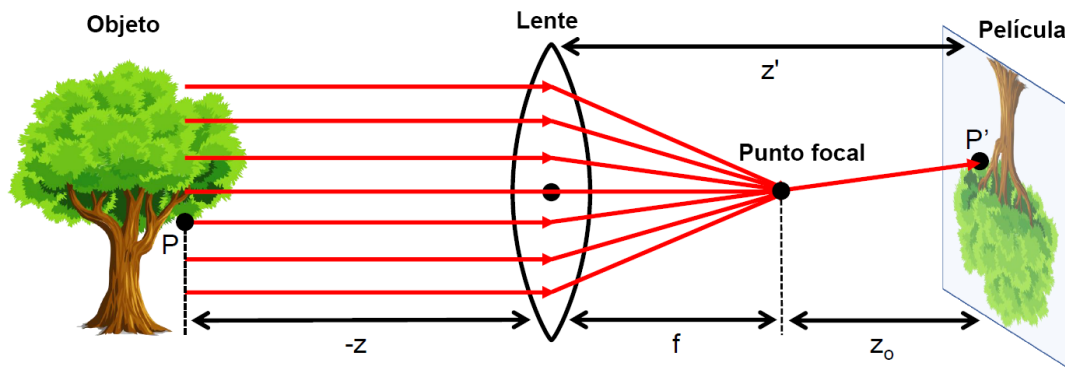


Figura 3.14: Las lentes enfocan los rayos de luz paralelos al eje óptico en el punto focal. Además, esta configuración ilustra el modelo de refracción paraxial, que nos ayuda a encontrar la relación entre los puntos en el plano de la imagen y el mundo tridimensional en cámaras con lentes, asumiendo una lente totalmente plana [40].

Para poder entender mejor el modelo matricial de cámara con lente es importante destacar que la información de la cámara se puede traducir en parámetros extrínsecos e intrínsecos de la propia cámara.

Como se comentó anteriormente, los parámetros extrínsecos definen una rotación, R , y una traslación, t . El origen del sistema de coordenadas de la cámara se encuentra en su centro óptico y sus ejes x e y definen el plano de la imagen [69].

Los parámetros intrínsecos representan una transformación proyectiva desde las coordenadas tridimensionales de la cámara a las coordenadas bidimensionales de la imagen. Las coordenadas de la cámara se mapean en el plano de la imagen utilizando los parámetros intrínsecos. Estos parámetros para un modelo de cámara completo incluyen la longitud focal, el punto principal y el coeficiente de inclinación. $[c_x, c_y]$ representa el centro óptico (el punto principal) en píxeles. Es el punto en el plano de la imagen donde intersecan los ejes de proyección de la cámara. Alrededor de este punto la imagen se hace simétrica y generalmente se encuentra en el centro del plano de la imagen. f_x y f_y representan la distancia focal en píxeles, donde $f_x = \frac{F}{p_x}$ y $f_y = \frac{F}{p_y}$, siendo F la distancia focal en unidades métricas y (p_x, p_y) representa el tamaño del píxel en unidades métricas. Finalmente, s es el coeficiente de inclinación, que es distinto de cero si los ejes de la imagen no son perpendiculares. Utilizando esta nomenclatura, la matriz intrínseca de la cámara, K , se define según [69]:

$$K = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (3.16)$$

Mientras que la inclinación de píxeles o “skewing” se define tal y como se muestra en la siguiente ecuación:

$$s = f_x \tan(\alpha) \quad (3.17)$$

3.5.2 Aberraciones primarias y cromáticas

El uso de lentes introduce otra serie de problemas, denominadas aberraciones o distorsiones ópticas. Existen varios tipos de aberraciones, como las aberraciones cromáticas y las aberraciones primarias (esférica, coma, astigmatismo y distorsiones geométricas).

Por un lado, el índice de refracción de la lente no es el mismo para todas las frecuencias, lo que ocasiona aberraciones del tipo “fringing”, o aberraciones cromáticas. El fringing ocurre cuando hay una distorsión o aberración en los bordes o áreas de alto contraste de una imagen capturada por una cámara o lente. Este fenómeno puede manifestarse como franjas de colores no deseadas, generalmente en los bordes de objetos, donde los colores se desvían o se superponen de manera incorrecta. Este efecto puede apreciarse en la Figura 3.15.

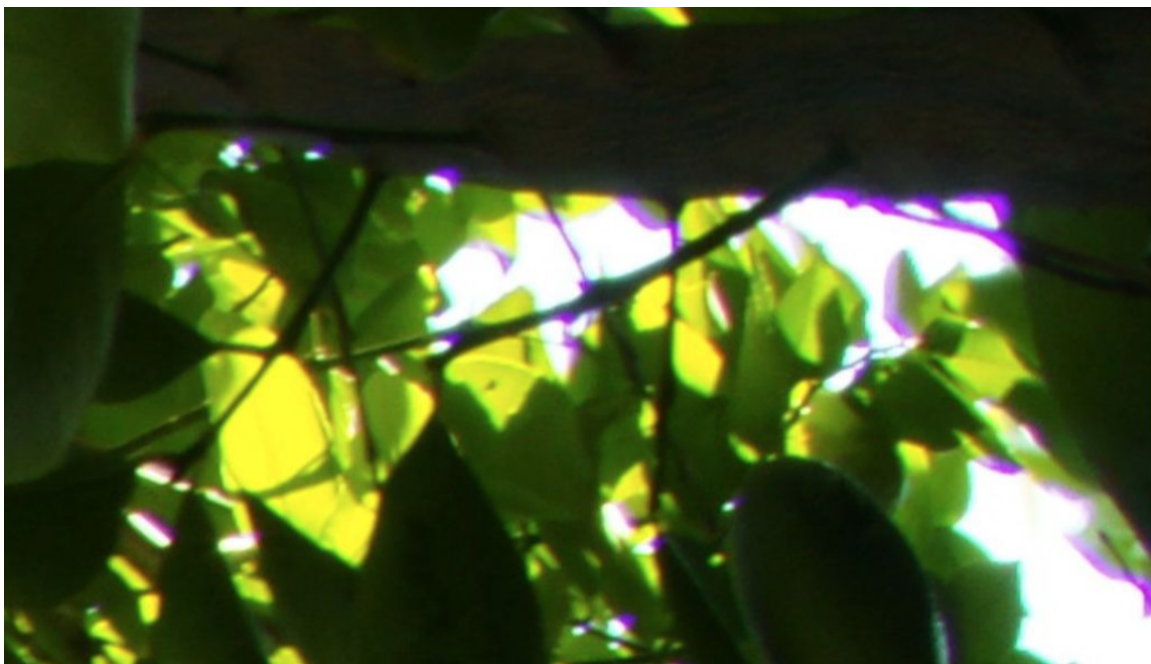


Figura 3.15: Imagen de la Università Ca' Foscari Venezia que muestra una aberración cromática o distorsión del tipo “fringing”.

Existen dos tipos de distorsión geométrica, la distorsión radial y la distorsión tangencial. Las propias imperfecciones de la lente ocasionan distorsión radial, donde la imagen se ve deformada en sus extremos. Dependiendo del tipo de imperfecciones en la lente, estas aberraciones pueden deformar la imagen en forma de cojín o barril, tal y como se observa en la Figura 3.16.

La distorsión radial es causada por el hecho de que diferentes porciones de la lente tienen longitudes focales distintas, aplicando un aumento (distorsión de barril) o disminución (distorsión de cojín) a la imagen proyectada cuanto más alejada se encuentre del eje paraxial. La desviación

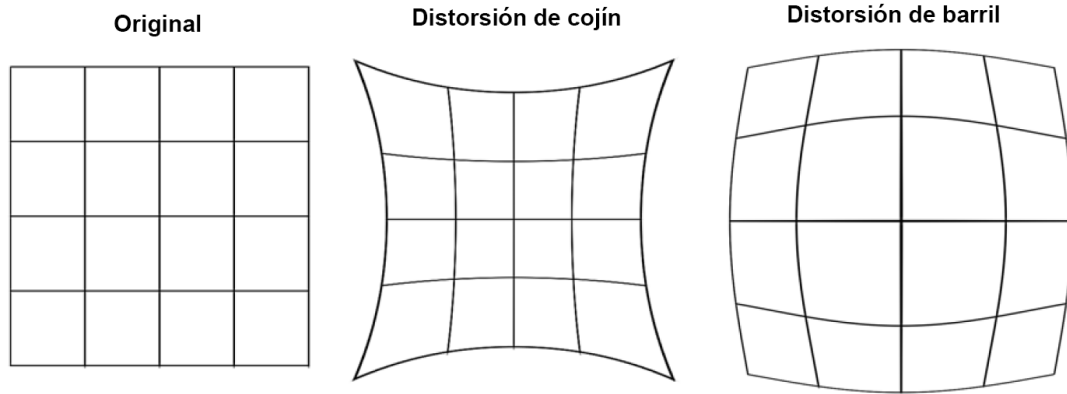


Figura 3.16: Dada una imagen original, ejemplos de distorsiones de tipo cojín y barril provocadas por las imperfecciones de la lente.

más importante en cámaras con lentes es generalmente la distorsión radial. En la práctica, este error se vuelve más significativo a medida que la longitud focal (y el precio) de la lente disminuye [39]. La distorsión radial provoca que los puntos de impacto de los rayos de luz se desvíen cada vez más a medida que se alejan del centro de la imagen. Como resultado, se produce una imagen tipo “ojo de pez”, donde los píxeles cercanos al centro experimentan pequeñas modificaciones y los más alejados sufren aberraciones más pronunciadas. Por lo tanto, el error acumulado en los píxeles de la imagen es proporcional a su distancia al centro, y la distorsión es similar para dos puntos a la misma distancia del centro. Para representar estas aberraciones, se utiliza un modelo de circunferencias. Una vez que entendemos la naturaleza de esta aberración, podemos intentar corregirla utilizando las siguientes ecuaciones [71]:

$$G(x_{\text{corregida}}, y_{\text{corregida}}) = I(x, y) \quad (3.18)$$

donde:

$$x_{\text{corregida}} = x \left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6 \right) \quad (3.19)$$

$$y_{\text{corregida}} = y \left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6 \right) \quad (3.20)$$

En las anteriores ecuaciones, r representa la distancia desde el centro hasta la circunferencia de aberración donde se encuentra el punto a corregir, es decir, el radio de dicha circunferencia. Los coeficientes k_1 , k_2 y k_3 son valores de corrección específicos que deben ser determinados para cada cámara, ya que varían según la naturaleza de la aberración. El coeficiente k_3 generalmente se utiliza solo en cámaras con una gran distorsión radial, mientras que en el resto se le asigna un valor de 0 [71].

El segundo tipo de distorsión geométrica es la distorsión tangencial. Se denomina distorsión tangencial a la distorsión que ocurre debido a la falta de paralelismo entre la lente de la cámara y el sensor de la misma, lo cual resulta en imágenes con forma trapezoidal. Esta distorsión se puede corregir utilizando las siguientes fórmulas [71]:

$$x_{\text{corregida}} = x + 2p_1 y + p_2 (r^2 + 2x^2) \quad (3.21)$$

$$y_{\text{corregida}} = y + 2p_1(r^2 + 2y^2) + 2p_2x \quad (3.22)$$

Los parámetros p_1 y p_2 representan los coeficientes que deben ser utilizados en las fórmulas. Al igual que en el caso de la distorsión radial, estos coeficientes varían de una cámara a otra, por lo que es necesario encontrar los coeficientes adecuados para corregir la distorsión específica de nuestra cámara.

El resto de aberraciones primarias que podemos encontrar son: la aberración astigmática, la coma y la aberración esférica. A continuación se detallan brevemente sus características.

En óptica, el astigmatismo es la definición de la precisión en la formación de imágenes. En un astigmatismo ideal, todos los rayos que provienen de un mismo punto del objeto convergen en un solo punto de la imagen. Sin embargo, en la realidad, esta situación nunca ocurre. La aberración astigmática ocurre cuando los rayos que se propagan de forma perpendicular tienen distancias focales diferentes. Si se utiliza un sistema óptico con astigmatismo para formar una imagen de una cruz, las líneas verticales y horizontales estarán enfocadas nítidamente a dos distancias focales diferentes [52].

Cuando un objeto se encuentra alejado del eje óptico de la lente, ocurre el fenómeno de la aberración cromática, o coma. La coma es una aberración óptica que ocurre en ciertos sistemas ópticos, como los telescopios con espejos parabólicos cuando una fuente de luz se encuentra fuera del centro del campo. Esto provoca que las partes del espejo enfoquen la luz hacia puntos diferentes, generando puntos de luz descentrados en forma de comas. Este tipo de aberración puede observarse en la Figura 3.17.

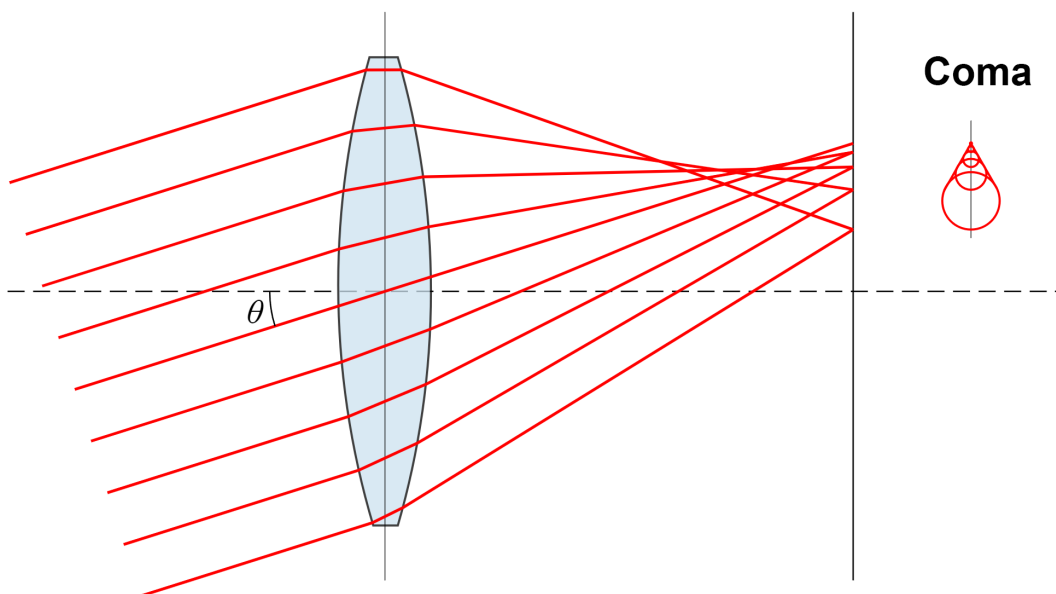


Figura 3.17: Aberración del sistema óptico del tipo "coma". El ángulo θ indica la desviación respecto al eje paraxial.

Por último, existe la aberración esférica. Este tipo de aberración no depende de los defectos de la propia lente, sino que depende de que los rayos no convergen en el punto focal debido a que no se cumple la aproximación paraxial. La aberración esférica se refiere a una desviación de los rayos que inciden paralelamente al eje óptico, resultando en un enfoque diferente para los rayos que se encuentran alejados del eje en comparación con los rayos cercanos a él. El efecto de la aberración esférica es proporcional a la cuarta potencia del diámetro de la lente o espejo, e inversamente proporcional al cubo de la longitud focal. Este fenómeno es más pronunciado en sistemas ópticos de corta distancia focal, como en las lentes utilizadas en microscopios.

3.5.3 Calibración geométrica de la cámara

Llamamos calibración geométrica de una cámara al proceso de determinar los parámetros intrínsecos y extrínsecos de la cámara para poder relacionar las coordenadas en el mundo real con las coordenadas en la imagen capturada por la cámara. Los parámetros intrínsecos se refieren a las características internas de la cámara, como la distancia focal, el centro óptico, la distorsión radial y la distorsión tangencial. Estos parámetros son específicos de cada cámara y afectan a la forma en que se proyecta la imagen en el plano de la imagen. Los parámetros extrínsecos describen la posición y orientación relativa de la cámara con respecto al mundo tridimensional. Determinan cómo se mapean los puntos en el espacio al plano de la imagen. La calibración geométrica se realiza utilizando patrones de calibración, que pueden ser objetos planos con marcas conocidas o patrones tridimensionales. Al capturar imágenes del patrón desde diferentes ángulos y posiciones, se pueden encontrar las correspondencias entre los puntos en el mundo real y los puntos en la imagen. Utilizando estas correspondencias, se puede estimar la matriz de proyección de la cámara, así como los parámetros intrínsecos y extrínsecos [11, 12, 43, 131].

La calibración geométrica es fundamental en muchas aplicaciones de visión por computador, como la reconstrucción 3D, la detección de objetos y la navegación robótica, ya que permite una correspondencia precisa entre el mundo real y las imágenes capturadas. En general, el principal problema que intenta resolver la calibración es la distorsión radial, ya que es el tipo de distorsión más frecuente en fotografía.

A lo largo de esta tesis se ha utilizado principalmente el método de calibración de cámaras propuesto en [131], cuya implementación en Matlab puede encontrarse en la librería Camera Calibration Toolbox [69]. Este método de calibración de la cámara puede resumirse en los siguientes pasos [131]:

1. Se imprime un patrón, como un damero, y se adhiere a una superficie plana.
2. Se toman algunas imágenes del plano modelo en diferentes orientaciones, moviendo el plano o la cámara (Figura 3.18).
3. Se detectan los puntos característicos de las imágenes, por ejemplo, las intersecciones de un damero.
4. Se realiza una estimación de los cinco parámetros intrínsecos (k_{1-3} referentes a las Ecuaciones 3.19 y 3.20, p_{1-2} referentes a las Ecuaciones 3.21 y 3.22) y todos los parámetros extrínsecos (f_x , f_y , c_x y c_y referentes a la Ecuación 3.16).

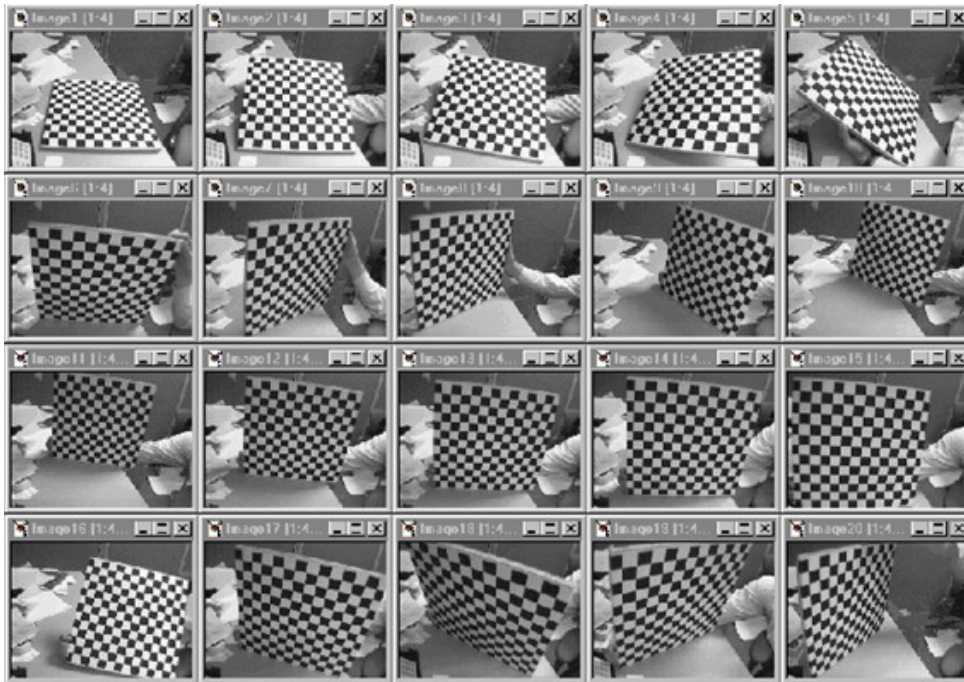


Figura 3.18: Fotografías de ejemplo de diferentes posiciones de un damero obtenidos de la web de la librería Matlab Calibration Toolbox. El orden de las esquinas se mantiene constante al variar la posición relativa para tener una referencia estable.

5. Se realiza un ajuste de los parámetros a través de una función de minimización que tenga en cuenta múltiples imágenes de calibración.

Es posible obtener un mayor nivel de detalle sobre este proceso al observar las siguientes imágenes extraídas de la documentación del Toolbox de Camera Calibration de Matlab.

En la Figura 3.18, se aprecia un patrón en forma de damero en diversas posiciones y áreas del sensor, lo que permite lograr una calibración lo más precisa posible. En las Figuras 3.19 y 3.20, se pueden observar dos representaciones de las distorsiones radiales y tangenciales, así como los parámetros intrínsecos que describen la distorsión de la cámara a calibrar, mostrados como anillos distorsionados. Finalmente, en la Figura 3.21, se presenta el resultado de la calibración de los parámetros extrínsecos, que indican la posición relativa de la cámara y el damero.

3.5.4 Corrección de las distorsiones geométricas (radial y tangencial)

Una vez obtenidos los parámetros de la cámara después de la calibración, es factible realizar una transformación homográfica de las fotografías adquiridas para eliminar los factores de distorsión radial y tangencial.

Existen diversas estrategias para llevar a cabo este proceso; por ejemplo, la librería OpenCV utiliza la función *cv.initUndistortRectifyMap* para calcular la matriz de transforma-

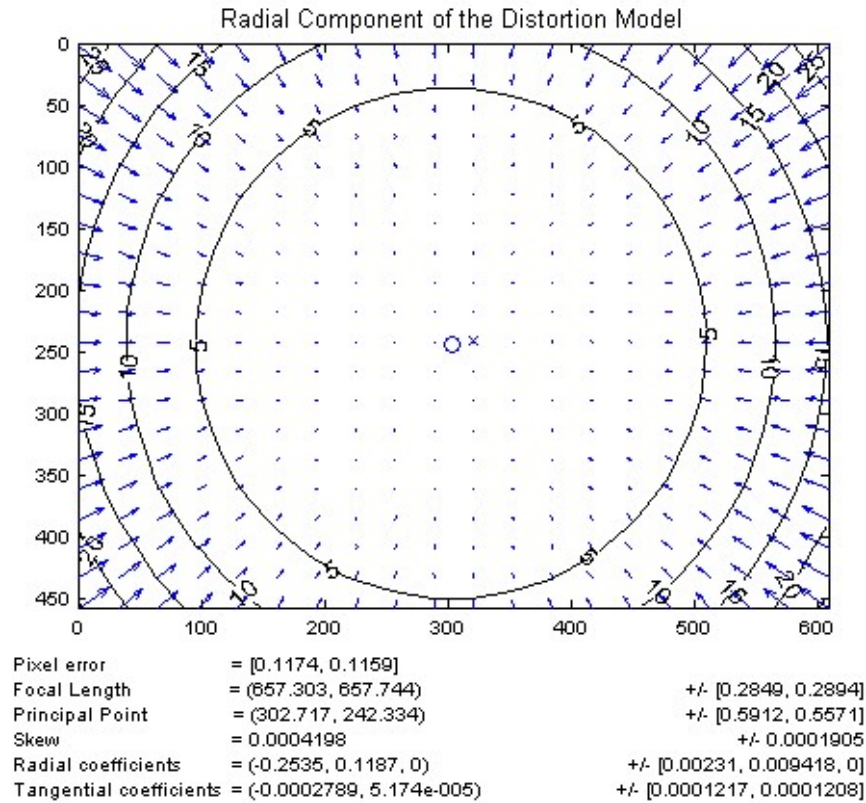


Figura 3.19: Distorsión radial y parámetros intrínsecos k_{1-3} después de la calibración de la cámara.

ción de la imagen a partir de los parámetros intrínsecos definidos en la matriz intrínseca K descrita en la Ecuación 3.16. En esta tesis se ha utilizado una función propia de Matlab que se encuentra en la librería Camera Calibration Toolbox, llamada *undistortImage*, que calcula la matriz de transformación y aplica una homografía sobre la imagen a partir de dicha matriz para eliminar la distorsión geométrica. A continuación, se presenta un ejemplo sencillo del cálculo de las nuevas coordenadas a partir de las posiciones normalizadas con solo cinco parámetros de distorsión:

$$x_{\text{corregida}} = x \left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6 \right) + 2p_1 xy + p_2 (r^2 + 2x^2) \quad (3.23)$$

$$y_{\text{corregida}} = y \left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6 \right) + p_1 (r^2 + 2y^2) + 2p_2 xy \quad (3.24)$$

Estas ecuaciones representan el cálculo de las coordenadas distorsionadas corregidas ($x_{\text{corregida}}$, $y_{\text{corregida}}$) a partir de las coordenadas de la imagen original (x , y), los coeficientes de distorsión (k_1 , k_2 , k_3), y la distancia radial r . Posteriormente, las coordenadas corregidas se escalan multiplicando por las distancias focales en píxeles (f_x , f_y) y se desplazan agregando las coordenadas del punto principal (c_x , c_y). Esto se realiza para transformar las coordenadas corregidas de nuevo a las coordenadas de píxeles de la imagen de salida. El resultado son las coordenadas mapeadas (x_{mapeada} , y_{mapeada}) en el sistema de coordenadas de píxeles de la imagen de salida:

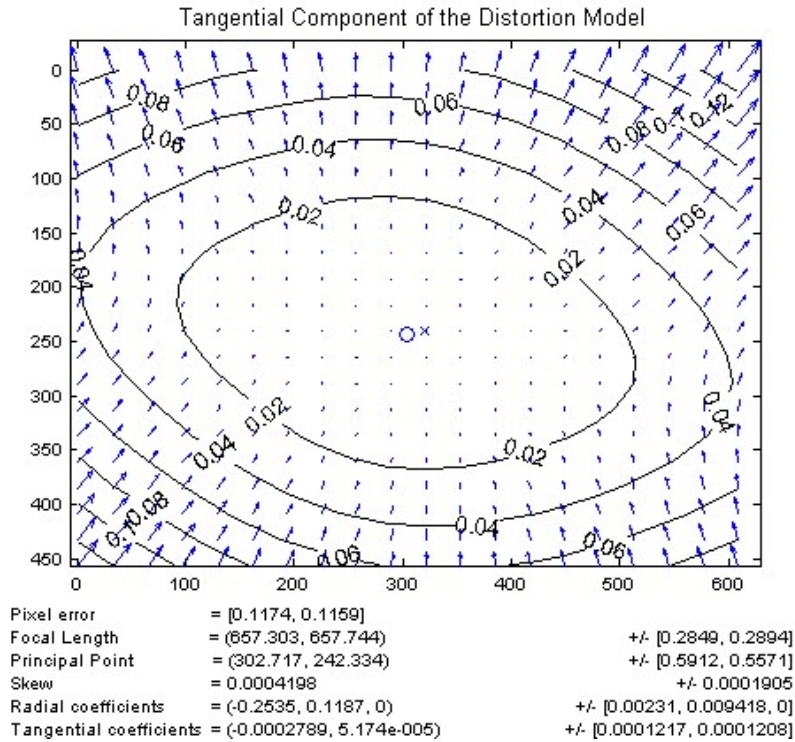


Figura 3.20: Distorsión tangencial y parámetros intrínsecos p_{1-2} obtenidos tras la calibración de la cámara.

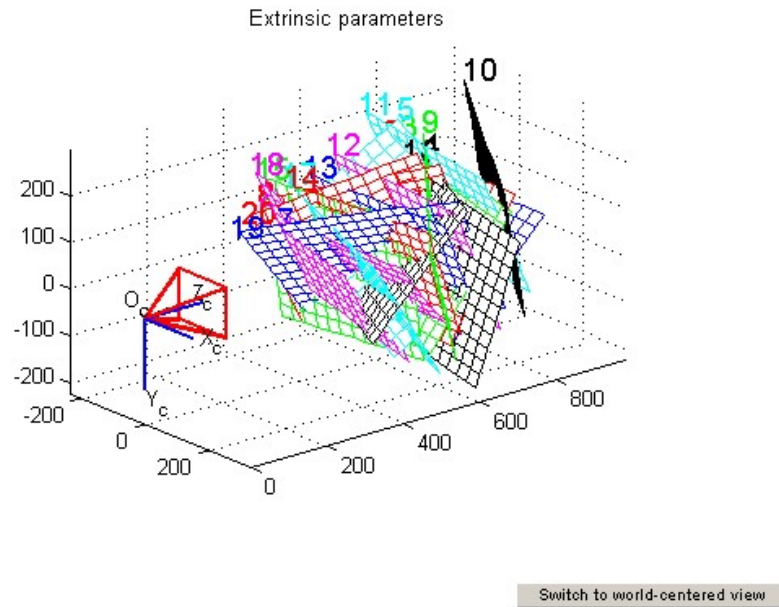


Figura 3.21: Cálculo de las posiciones relativas del damero respecto a la cámara durante el análisis, que reflejan los parámetros extrínsecos.

$$x_{\text{mapeada}} = x_{\text{corregida}} \cdot f_x + c_x \quad (3.25)$$

$$y_{\text{mapeada}} = y_{\text{corregida}} \cdot f_y + c_y \quad (3.26)$$

En función de la complejidad del modelo, es posible utilizar más de cinco valores para describir los parámetros intrínsecos. No obstante, con tan solo cinco valores, se logra una precisión satisfactoria para la mayoría de aplicaciones con cámaras. En las Figuras 3.22 y 3.23 se observa un ejemplo de corrección de la distorsión utilizando la estrategia descrita anteriormente.



Figura 3.22: Imagen original que presenta distorsiones geométricas.



Figura 3.23: Imagen sin distorsión tras realizar una transformación homográfica.

3.5.5 Corrección de la perspectiva (rectificación al plano principal)

Es posible que la región de interés de nuestra imagen esté delimitada por patrones dentro de la propia imagen. Una técnica clásica para llevar esta región de interés al plano principal es la rectificación. En este proceso, se detectan las cuatro esquinas de la región de interés y se aplica una homografía para crear una nueva imagen que incluye únicamente la región de interés.

La descripción formal de la operación homográfica que representa la rectificación al plano principal se puede definir de la siguiente manera: Supongamos que (x, y) representa la posición de un punto en la imagen original, y (x', y') es la posición correspondiente en la imagen rectificada. La homografía se representa como una matriz 3×3 , denotada por H , tal que:

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = H \cdot \begin{bmatrix} x \\ y \\ w \end{bmatrix} \quad (3.27)$$

donde (x, y) son las coordenadas del punto en la imagen original, (x', y') identifican las coordenadas del punto correspondiente en la imagen rectificada y w y w' representan los factores de escala homogéneos que se utilizan para normalizar las coordenadas. La matriz de la homografía H se calcula utilizando las coordenadas de las cuatro esquinas de la región de interés.

Las Figuras 3.24 y 3.25 ilustran un ejemplo del proceso de rectificación. En la Figura 3.24 se muestra una imagen de entrada después de un preprocesado inicial en la que se ha eliminado la distorsión de la lente. Esta imagen contiene una región de interés definida por las cuatro esquinas del marco de referencia. En la Figura 3.25 se presenta el resultado de la rectificación al plano principal mediante la homografía descrita.



Figura 3.24: Imagen de entrada, donde la región de interés está delimitada por las cuatro esquinas del marco.

Figura 3.25: Imagen rectificada al plano principal, resultado de la transformación homográfica.

3.6 Resumen de las técnicas de preprocesado utilizadas en la tesis

Una vez que se han enumerado las principales técnicas de mejora de imagen, de eliminación de ruido y de calibrado y eliminación de la distorsión, es de interés particularizar en las que se han utilizado en la presente tesis.

Estas técnicas solo se han considerado en el caso de estudio de agricultura de precisión, donde se pretendía el desarrollo de una herramienta automática para cuantificar el porcentaje de maleza y plantas de interés (girasoles) que se encontraban en un área determinada. Para ello, las técnicas de preprocesado que se han utilizado directamente han sido: la calibración geométrica de las lentes de la cámara y la corrección de la distorsión radial y tangencial, y la corrección de la perspectiva de las imágenes del estudio.

Sin embargo, también es importante conocer varios de los conceptos introducidos en este capítulo (como la relación entre una imagen, su histograma y sus niveles de color) para entender las técnicas de segmentación del Capítulo 4.

Aunque en esta tesis no aparece directamente ninguno de los tipos de ruido mencionados, sí que surgen ciertas problemáticas de generación de ruido al utilizar imágenes multi-espectrales y haber cierto error en la superposición de las bandas. Las estrategias para solventar este tipo de error se verán con detalle en el capítulo del caso de estudio de agricultura de precisión (Capítulo 5).

4. Técnicas de segmentación

En el procesamiento digital de imágenes y la visión artificial, la segmentación es el proceso de dividir una imagen digital en múltiples segmentos de imagen, también conocidos como regiones de imagen u objetos de imagen (conjuntos de píxeles). El objetivo de la segmentación es simplificar y/o cambiar la representación de una imagen de forma que sea más significativa y más fácil de analizar. Es frecuente realizar un proceso de clasificación para asignar una etiqueta o categoría a cada uno de los segmentos resultantes. De este modo, la segmentación divide la imagen en regiones o segmentos basados en ciertas características comunes, y luego la clasificación asigna una etiqueta específica a cada uno de esos segmentos [7, 100].

La segmentación de imágenes se utiliza típicamente para localizar objetos y límites (líneas, curvas, etc.) en ellas. Debido a que la segmentación es una de las transformaciones más estudiadas en la literatura, en el estado del arte se han definido una gran variedad de algoritmos específicos para la segmentación de diversos tipos de imágenes [2, 25, 34, 63, 68, 107]. Además, en los últimos años, gracias a los avances en el aprendizaje profundo (conocido en inglés como “Deep Learning” o DL), se han empleado técnicas de DL para la segmentación y posterior clasificación de los píxeles en las imágenes. Con estas técnicas, es posible entrenar modelos que sean capaces de identificar y segmentar automáticamente objetos en imágenes complejas.

Existe una amplia variedad de algoritmos y estrategias de segmentación. El uso de uno u otro depende de diversos factores, como la complejidad de la tarea, la cantidad de datos a procesar, los recursos computacionales disponibles, etc.

Este capítulo se organiza en las siguientes secciones. En la Sección 4.1 se describe en qué consiste la segmentación basada en clusterización. La Sección 4.2 está dedicada a describir la segmentación basada en la detección de bordes. En la Sección 4.3 se detallan las técnicas más utilizadas para la segmentación basada en regiones: la clusterización, el crecimiento de regiones, la segmentación Quadtree y la segmentación multi-resolución. En la Sección 4.4 se

analizan las técnicas más empleadas para la segmentación y clasificación de píxeles utilizando DL. Finalmente, en la última sección, se realiza una síntesis de las técnicas de segmentación que se han utilizado en la presente tesis.

4.1 Segmentación basada en umbralización

La umbralización es una técnica básica en el procesamiento de imágenes que se utiliza para segmentar una imagen en regiones u objetos de interés, al separar los píxeles que tienen intensidades superiores o inferiores a un valor umbral específico [34]. Existen diferentes métodos para aplicar la umbralización dependiendo de la imagen y de los objetivos de la segmentación.

Una de las formas más básicas y ampliamente utilizadas de la umbralización es la binarización. La binarización es una técnica fundamental en el procesamiento de imágenes que tiene como objetivo segmentar una imagen en dos regiones distintas, utilizando un valor umbral específico. Esta técnica convierte la imagen en una representación binaria, donde los píxeles se clasifican como negro (valor 0) o blanco (valor 1) en función de si su intensidad es mayor o menor al umbral establecido. La binarización se emplea con frecuencia en aplicaciones que involucran la detección de bordes, el reconocimiento de objetos, así como el análisis de texturas. Puede observarse un ejemplo de binarización en las Figuras 4.1 y 4.2.



Figura 4.1: Imagen original.



Figura 4.2: Imagen binarizada en blanco y negro.

Debido a que la umbralización depende del valor escogido como umbral, existen numerosos métodos para calcular este umbral según la aplicación deseada, por ejemplo, el método de Otsu [78]. El método de Otsu es una técnica de umbralización automática utilizada en procesamiento de imágenes para segmentar una imagen en dos clases (objeto y fondo) mediante la búsqueda del valor óptimo de umbral. Este método se basa en el histograma de intensidades de la imagen y busca un valor de umbral, T , que maximice la separación entre las clases, minimizando la varianza intra-clase y maximizando la varianza inter-clase. De este modo, itera a través de todos los posibles valores de umbral T y calcula la varianza inter-clase para cada uno de ellos, con el fin de encontrar el valor de umbral que cumpla con esta condición de maximización. Este

valor de umbral se utiliza finalmente para la binarización de la imagen:

$$T_{\text{opt}} = \operatorname{argmax}_T (\sigma_{\text{inter}}^2(T)) \quad (4.1)$$

donde T_{opt} es el valor óptimo del umbral que maximiza la varianza inter-clase $\sigma_{\text{inter}}^2(T)$ y se utiliza para segmentar la imagen en dos clases: objeto y fondo. Entre las aplicaciones del método de Otsu destacan: el reconocimiento de caracteres, la detección de bordes o el análisis de imágenes médicas, donde se requiere una segmentación automática y precisa.

Otra de las técnicas de umbralización de más importancia es la segmentación de doble umbral. A diferencia de la umbralización simple, en este caso se determinan dos valores de umbral, T_{min} y T_{max} . Esta técnica nos permite establecer con precisión los valores que definen la región de interés con píxeles cuya intensidad de color se encuentra entre esos dos umbrales. La fórmula para la segmentación por doble umbral se puede expresar como se indica a continuación:

$$I(x,y) = \begin{cases} 0 & \text{si } I(x,y) < T_{\text{min}} \\ 1 & \text{si } T_{\text{min}} \leq I(x,y) \leq T_{\text{max}} \\ 0 & \text{si } I(x,y) > T_{\text{max}} \end{cases} \quad (4.2)$$

donde $I(x,y)$ representa la intensidad del píxel en la posición (x,y) de la imagen original. Los píxeles con intensidades menores a T_{min} se asignan al fondo (valor 0), los píxeles con intensidades entre T_{min} y T_{max} se asignan a la región de interés (valor 1), y los píxeles con intensidades mayores a T_{max} se asignan nuevamente al fondo (valor 0).

La segmentación por doble umbral es útil para resaltar con precisión regiones de interés específicas en una imagen, lo que permite un análisis más detallado de esas áreas en aplicaciones como detección de bordes, reconocimiento de objetos o identificación de patrones complejos.

4.2 Segmentación basada en detección de bordes

La detección de bordes es una técnica ampliamente utilizada en el procesamiento de imágenes para identificar y resaltar las regiones donde ocurren cambios bruscos en la intensidad, lo que indica la presencia de límites entre diferentes objetos o regiones en la imagen. Los bordes son características de gran importancia en una imagen, ya que proporcionan información relevante sobre la forma y la estructura de los objetos contenidos.

Existen diversos métodos ampliamente utilizados en la práctica para la detección de bordes. Los operadores de gradiente, como el operador Sobel y el operador Prewitt, se basan en el gradiente de intensidad de los píxeles y sirven para resaltar áreas con cambios bruscos en la imagen. Los filtros de convolución, como el filtro de Canny, suavizan la imagen y buscan áreas donde el gradiente de intensidad sea máximo, formando contornos claros. Además, los operadores basados en diferencia de intensidades, como el operador de diferencia de intensidades de Roberts, buscan píxeles con diferencias significativas de intensidad respecto a sus vecinos, lo que ayuda a identificar bordes en la imagen [34, 107]. En la Figura 4.3 se han aplicado algunos de los métodos de detección de bordes anteriormente citados.

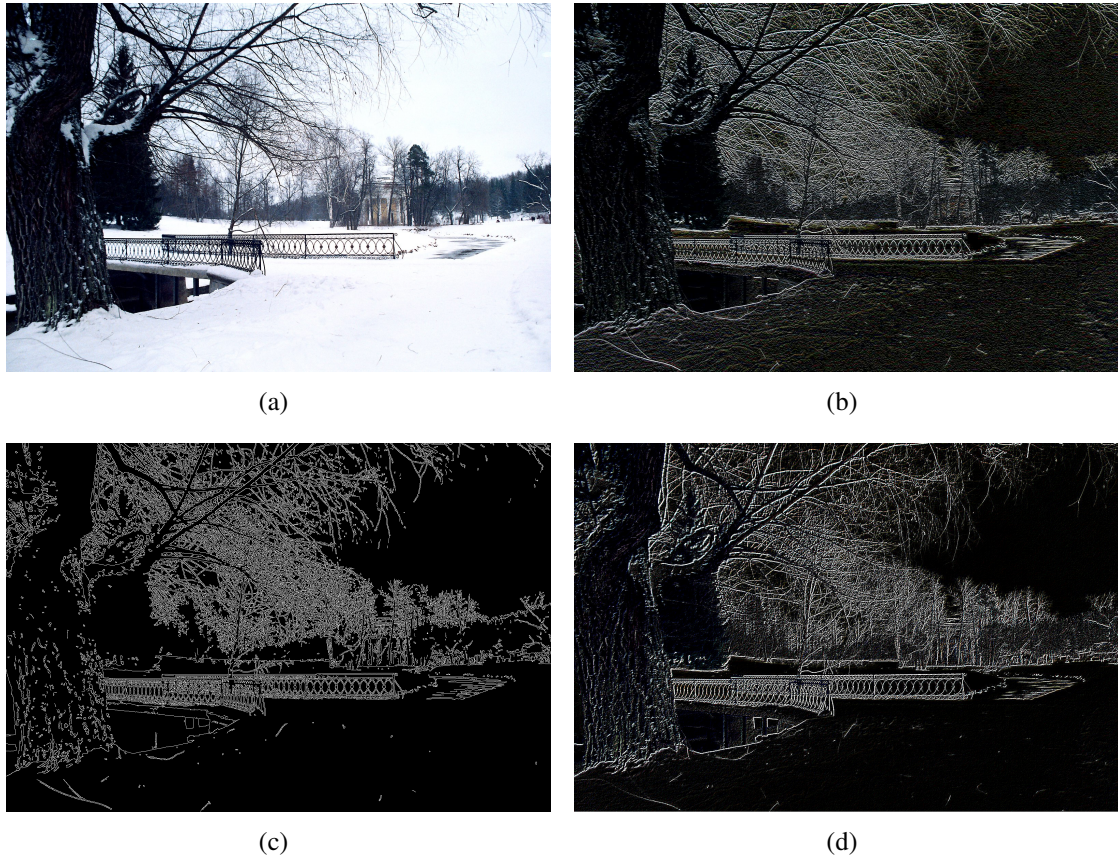


Figura 4.3: (a) Imagen original; (b) Imagen después de aplicar el operador Sobel; (c) Imagen después de aplicar el filtro Canny; (d) Imagen después de aplicar el filtro Roberts.

Aunque la detección de bordes no es por sí misma una técnica de segmentación completa, su aplicación es un paso esencial previo a muchas estrategias de segmentación posteriores como la umbralización o el crecimiento de regiones. Antes de aplicar los métodos de detección de bordes, es común realizar un preprocesado de la imagen para reducir el ruido y mejorar la calidad de la misma. Algunas técnicas como el suavizado y la mejora del contraste son utilizadas previamente sobre la imagen para obtener mejores resultados en la detección de bordes.

Después de detectar los bordes, puede ser necesario aplicar técnicas de postprocesamiento para eliminar bordes falsos o fragmentados. Operaciones morfológicas, como la erosión y la dilatación, son comúnmente utilizadas para refinar y conectar los contornos detectados, mejorando la precisión del proceso de segmentación.

4.3 Segmentación basada en regiones

La segmentación basada en regiones es una técnica de procesamiento de imágenes que agrupa píxeles o puntos similares en regiones contiguas con características homogéneas. Esta aproximación se centra en la similitud entre los píxeles y se utiliza para dividir una imagen en regiones significativas. Existen numerosas técnicas para realizar este tipo de segmentación en base a las características de agrupaciones de píxeles, como su forma, color o las características de las

regiones anexas. La clave para las segmentaciones basadas en regiones es la selección del mejor criterio para la combinación o la partición. A lo largo de esta sección se describen las principales técnicas de segmentación basadas en regiones que se han utilizado en la presente tesis.

4.3.1 Segmentación basada en clusterización

La segmentación basada en clusterización es un enfoque que agrupa píxeles o puntos de datos en clústeres, basándose en sus similitudes. Se utilizan algoritmos de agrupamiento, como K-means o DBSCAN, para dividir la imagen en regiones coherentes según la similitud de los valores de los píxeles.

El algoritmo de agrupación K-means [68] pretende dividir un conjunto de datos en un número predeterminado de clústeres. Este algoritmo K-means, cuyo pseudocódigo se encuentra ilustrado en el Algoritmo 1, consta de los pasos que se detallan a continuación.

Algoritmo 1 Clusterización K-Means

Input: Puntos de datos X , Número de clústeres K
 Inicializa K centroides de clúster de forma aleatoria
repeat

for cada punto de datos $x \in X$ **do**
 Asigna x al centroide más cercano

for cada cluster k **do**
 Actualiza el centroide del cluster k como la media de sus puntos de datos asignados

until Convergencia

Output: Asignación final de clústeres y centroides

En primer lugar, se eligen k centroides iniciales, donde k es el número de clústeres que se desea obtener. Estos centroides pueden ser seleccionados al azar o mediante algún otro método de inicialización. Cada centroide representa el punto central de un clúster. Para cada dato en el conjunto, se asigna al clúster cuyo centroide es el más cercano utilizando la distancia euclidiana. La distancia euclidiana se calcula como la distancia entre dos puntos en un espacio multidimensional, tal y como se describe en la Ecuación 4.3.

$$\text{dist}(p, q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (4.3)$$

donde n es la dimensión del espacio, p_i es la coordenada i -ésima del punto p y q_i es la coordenada i -ésima del punto q . Una vez asignados todos los datos a sus respectivos clústeres, se calculan nuevos centroides para cada clúster tomando la media de todas las coordenadas de los datos que pertenecen a ese clúster. Este paso puede verse reflejado en la Ecuación 4.4:

$$c_i = \frac{1}{|C|} \sum_{p \in C} p_i \quad (4.4)$$

donde c_i es la coordenada i -ésima del centroide, $|C|$ es el número de puntos en el clúster y p_i es la coordenada i -ésima del punto p en el clúster C . Los pasos de asignación y actualización se

repiten en iteraciones hasta que los centroides convergen y ya no haya cambios significativos en la asignación de los datos a los clústeres o hasta alcanzar un número máximo de iteraciones. La convergencia significa que los centroides han alcanzado una posición estable y los clústeres son coherentes con los datos asignados a ellos. El resultado final del algoritmo K-means es la división del conjunto de datos en k clústeres, donde cada dato pertenece al clúster cuyo centroide es más cercano.

Las Figuras 4.4 y 4.5 muestran un ejemplo de aplicación del algoritmo K-means para $k = 3$.



Figura 4.4: Imagen original.



Figura 4.5: Imagen segmentada en tres regiones tras aplicar el algoritmo K-means para $k = 3$.

El algoritmo K-means es muy utilizado por su simplicidad para clasificar conjuntos de datos e imágenes. Sin embargo, existen otros algoritmos más recientes que resuelven el problema de definir a priori el número óptimo de clústeres, como es el algoritmo de segmentación basado en densidad DBSCAN [25]. DBSCAN es un algoritmo que agrupa puntos en regiones densas y separa regiones de baja densidad. No requiere especificar el número de clústeres de antemano, lo que lo hace más flexible en ciertos escenarios.

4.3.2 Segmentación basada en crecimiento de regiones

La segmentación basada en crecimiento de regiones es una técnica utilizada en el procesamiento de imágenes y análisis de datos para dividir una imagen o conjunto de datos en regiones o clústeres coherentes y homogéneos [2]. A diferencia de otras técnicas de segmentación que se centran en criterios globales de similitud, la segmentación basada en crecimiento se basa en criterios locales y se construye iterativamente.

El proceso de segmentación basada en crecimiento comienza con la selección de uno o más puntos de semilla en la imagen o conjunto de datos. Estos puntos de semilla actúan como

inicio de las regiones y pueden ser elegidos manualmente o mediante la aplicación de métodos automáticos. A continuación, se aplica un criterio de crecimiento para expandir cada región a partir de los puntos de semilla. El criterio de crecimiento se basa en la similitud entre los píxeles o datos vecinos y los puntos de semilla. Por lo general, se establece una condición que compara la similitud entre el valor del píxel (o dato) vecino y el valor de la semilla. Si se cumple la condición, el píxel vecino se agrega a la región en crecimiento y se convierte en un nuevo punto de semilla para expandir la región aún más. Este proceso de crecimiento se repite iterativamente hasta que ya no es posible agregar nuevos píxeles a la región o se alcanza un criterio de parada específico.

Una variante de este método consiste en utilizar objetos (grupos de píxeles o clústeres obtenidos mediante otro tipo de segmentación) en lugar de utilizar píxeles aislados. Esto permite añadir capas de complejidad a los criterios de selección de semillas o a la condición de crecimiento, tales como factores de forma del objeto, intensidad media o cantidad de bordes en común respecto a la región semilla.

La segmentación basada en crecimiento es útil cuando se desean segmentar regiones de formas irregulares o cuando la imagen o conjunto de datos no presenta un contraste claro entre las diferentes regiones. Sin embargo, esta técnica suele ser sensible a la elección de los puntos de semilla y al criterio de crecimiento, lo que puede afectar al resultado de la segmentación [34, 107]. La Figura 4.6 ilustra un ejemplo de segmentación utilizando el método de crecimiento de regiones, que ha sido extraído de [2].



Figura 4.6: Imagen original y segmentada en regiones tras la aplicación de tres nodos semilla [2].

4.3.3 Segmentación Quadtree

El término Quadtree, o árbol cuaternario, se utiliza para describir clases de estructuras de datos jerárquicas cuya propiedad común es que están basados en el principio de descomposición recursiva del espacio. En un Quadtree de puntos, el centro de una subdivisión está siempre en un punto. Al insertar un nuevo elemento, el espacio queda dividido en cuatro. Al repetir el proceso, el cuadrante se divide de nuevo en cuatro, y así sucesivamente [29].

La segmentación Quadtree es una técnica que divide una imagen en regiones recursivamente mediante la subdivisión de cuadrantes. Comienza con un cuadrante que cubre toda la imagen y, si la región no es homogénea, se divide en cuatro cuadrantes más pequeños. Este proceso de subdivisión continúa hasta que se cumplan ciertos criterios de parada, como alcanzar un tamaño mínimo de cuadrante o cuando todos los cuadrantes resultantes sean homogéneos. La subdivisión se realiza de manera recursiva, creando una estructura jerárquica de cuadrantes, donde cada nivel representa una resolución diferente de la imagen. Esta estructura puede representarse como un árbol cuaternario, donde el cuadrante original es el nodo raíz, y los cuadrantes más pequeños son sus hijos, y así sucesivamente.

Esta técnica es especialmente útil para segmentar imágenes con regiones de diferentes tamaños y formas, ya que se adapta automáticamente a la estructura de la imagen.

La selección del criterio de homogeneidad es un factor clave para el resultado final de la segmentación. Para determinar si una región es homogénea o no, se pueden utilizar diferentes criterios y métricas. Por ejemplo, en imágenes en escala de grises, la homogeneidad puede basarse en la varianza de intensidades de los píxeles en el cuadrante. En imágenes a color o con varias bandas, se pueden considerar métricas basadas en la diferencia de color entre píxeles.

A continuación, se detalla cómo calcular la homogeneidad para el caso sencillo de una imagen en escala de grises. Sea R un cuadrante con una matriz de píxeles I de tamaño $n \times n$. La métrica de homogeneidad H para el cuadrante R se puede calcular como la varianza de las intensidades de los píxeles según la Ecuación 4.5:

$$H(R) = \frac{1}{n^2} \sum_{x=1}^n \sum_{y=1}^n (I(x,y) - \mu)^2 \quad (4.5)$$

donde μ es la media de las intensidades de los píxeles en el cuadrante R .

Como se ha comentado anteriormente, el algoritmo Quadtree puede aplicarse en imágenes RGB o multi-espectrales teniendo en cuenta toda la información de homogeneidad de las diferentes bandas. Las Figuras 4.7 y 4.8 muestran un ejemplo de aplicación para una imagen de tres bandas (RGB).

4.3.4 Segmentación multi-resolución

La segmentación multi-resolución (también conocida como multi-escala) es una técnica ampliamente utilizada en el procesamiento de imágenes y análisis de datos, que implica analizar una imagen a diferentes escalas o resoluciones [63]. Esta estrategia es muy eficaz cuando se necesita detectar objetos o regiones a diferentes niveles de detalle en la imagen, o cuando se desea acelerar el proceso de segmentación en áreas menos relevantes.

Para llevar a cabo la segmentación multi-resolución se utilizan pirámides de imágenes, que son representaciones jerárquicas de la misma imagen en diferentes niveles de detalle. La pirámide de imágenes se construye mediante la generación de múltiples versiones de la imagen original,



Figura 4.7: Imagen original.

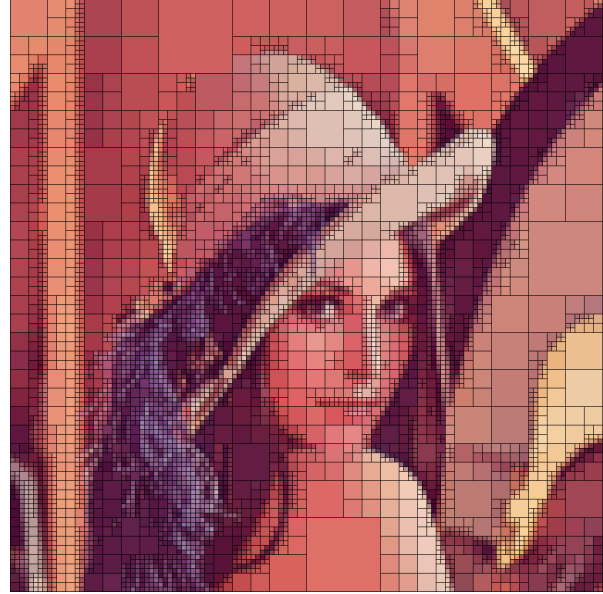


Figura 4.8: Imagen segmentada en regiones tras aplicar el algoritmo Quadtree.

donde cada versión tiene una resolución más baja que la anterior. Esto se logra mediante procesos de submuestreo o suavizado de la imagen.

Una forma común de construir una pirámide de imágenes es mediante el proceso de convolución y submuestreo, utilizando un filtro de suavizado, como un filtro gaussiano, para reducir la resolución de la imagen. La operación de convolución con el filtro gaussiano tiene la ventaja de mantener la información estructural importante en la imagen mientras reduce el nivel de detalle. El proceso de convolución con un filtro gaussiano se muestra en la Ecuación 4.6:

$$I'(x,y) = (I * G)(x,y) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} I(x-i, y-j) \cdot G(i, j) \quad (4.6)$$

donde $I(x,y)$ es el valor del píxel en la imagen original en las coordenadas (x,y) , $I'(x,y)$ es el valor del píxel en la imagen suavizada en las mismas coordenadas, y $G(i, j)$ es el kernel del filtro gaussiano en las coordenadas (i, j) .

Una vez que se ha construido la pirámide de imágenes, la segmentación se realiza analizando cada nivel de la pirámide por separado. Esto permite segmentar la imagen en diferentes niveles de granularidad, lo que puede ser útil para detectar objetos a diferentes escalas. Además, se puede ahorrar tiempo y recursos al realizar la segmentación solo en las áreas de interés de las imágenes con mayor resolución, y omitir áreas menos relevantes en imágenes de resoluciones más bajas.

La segmentación multi-resolución ha demostrado ser efectiva en diversas aplicaciones, como el procesamiento de imágenes médicas, la detección de objetos en imágenes satelitales y la extracción de características en reconocimiento de patrones. Además, esta técnica se integra bien con otras estrategias de segmentación, como la segmentación basada en regiones y la



Figura 4.9: Imagen original.

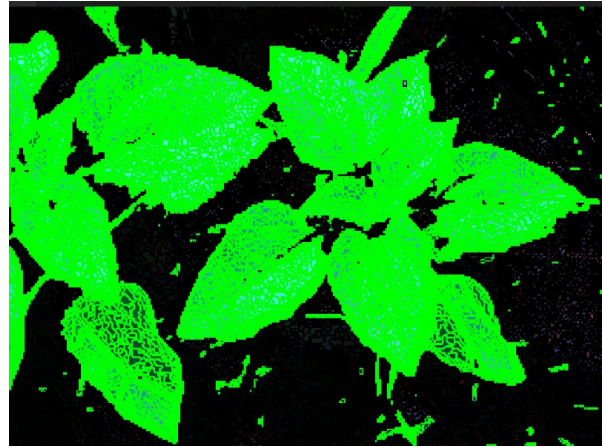


Figura 4.10: Imagen segmentada en regiones mediante multi-resolución.

segmentación jerárquica, para mejorar la precisión y eficiencia del proceso de segmentación. Las Figuras 4.9 y 4.10 ilustran un ejemplo de segmentación multi-resolución.

4.4 Técnicas de segmentación basadas en aprendizaje profundo (Deep Learning)

En esta sección se exponen las técnicas de segmentación basadas en aprendizaje profundo más destacadas en el ámbito de la visión por computador. Estas estrategias han ganado prominencia en la última década y han demostrado su eficacia en una variedad de aplicaciones. Cada una de estas metodologías se centra en abordar distintos desafíos dentro del campo de la segmentación, ofreciendo enfoques complementarios para extraer información de alto valor en las imágenes.

Las Redes Neuronales Convolucionales (CNN por sus siglas en inglés) han revolucionado la segmentación de imágenes al aprender patrones jerárquicos y características intrínsecas de las imágenes. Estas redes aprenden automáticamente a discernir entre diferentes regiones y objetos, lo que las convierte en una herramienta esencial para la segmentación semántica y la detección de objetos. Su capacidad para capturar detalles contextuales y su habilidad para generalizar a partir de ejemplos de entrenamiento las hacen altamente efectivas en la tarea de segmentación [41, 53].

La base conceptual de una CNN radica en la convolución y la operación de submuestreo, también conocida como pooling, que se aplica en las primeras capas para extraer características relevantes de las imágenes originales. Posteriormente, estas características se introducen en capas completamente conectadas para llevar a cabo la tarea final de clasificación.

En la arquitectura de las CNN, las capas de convolución desempeñan un papel central. En estas capas, se aplican filtros, también llamados núcleos o kernels convolucionales, a la imagen de entrada para detectar patrones específicos. Estos patrones pueden variar desde bordes y texturas hasta características más complejas. Durante el proceso de entrenamiento, estas capas de

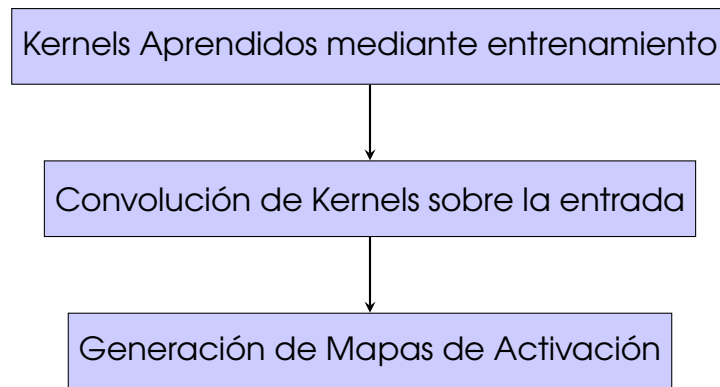


Figura 4.11: Esquema de la operación de una capa de convolución en una CNN, consistente en tres pasos. Paso 1: kernels definidos mediante aprendizaje esperando ser ejecutados. Paso 2: los kernels se aplican uno a uno sobre la imagen input utilizando una convolución. Paso 3: la salida de cada convolución produce un mapa de activación.

convolución aprenden automáticamente dichas características ajustando los pesos de los filtros. La convolución se realiza mediante una operación matemática que combina el filtro con una parte de la imagen, elemento a elemento, para producir una característica en la salida, tal y como se ilustra en el esquema de la Figura 4.11. En términos matemáticos, la convolución se puede expresar como una operación entre una matriz de filtro W y una parte de la entrada X , como se muestra en la Ecuación 4.7.

$$Y[i, j] = (W * X)[i, j] = \sum_m \sum_n W[m, n] \cdot X[i + m, j + n] \quad (4.7)$$

donde $Y[i, j]$ representa el valor resultante de la convolución en la posición (i, j) de la salida, $W[m, n]$ es el valor del filtro en la posición (m, n) y $X[i + m, j + n]$ es el valor correspondiente en la entrada.

En una CNN se definen varias capas de submuestreo. Las capas de submuestreo reducen la dimensionalidad espacial de las características extraídas por las capas de convolución. El submuestreo se realiza aplicando operaciones como la obtención del valor máximo en un área determinada de la imagen. Esta operación no solo disminuye la carga computacional, sino que también ayuda a prevenir el sobreajuste y a mantener la invariancia ante pequeñas transformaciones como la rotación o el escalado.

Las capas completamente conectadas, situadas al final de la arquitectura de una red neuronal convolucional, desempeñan un papel crucial en la tarea de interpretación y clasificación de características extraídas. En esta fase, las características previamente obtenidas a través de las capas de convolución y submuestreo son combinadas y estructuradas para realizar tareas específicas, como la clasificación de objetos en la imagen.

Para adecuar las características extraídas al formato requerido por las capas completamente conectadas, se realiza un proceso de "aplanamiento". Este proceso implica la transformación de datos multidimensionales en una estructura unidimensional, permitiendo que las capas conectadas procesen y analicen las características de manera eficiente.

Además, las funciones de activación no lineales, como la función ReLU (Rectified Linear Unit), son aplicadas después de las operaciones de convolución y en otros puntos estratégicos de la red. Estas funciones introducen la no linealidad necesaria para capturar relaciones complejas entre las características. Por ejemplo, la función ReLU permite que los valores positivos se mantengan sin cambios y los valores negativos se vuelvan cero, lo que contribuye a modelar mejor las relaciones en los datos.

Existen múltiples formas de implementar CNN. A continuación, se detallan las dos arquitecturas que se han utilizado en esta tesis: U-Net y FPN.

4.4.1 Segmentación basada en U-Net

U-Net es un modelo de red neuronal basado en codificador-decodificador, siendo este tipo de modelos los más populares para la segmentación basada en DL. U-Net fue propuesto en [90] para la segmentación de imágenes en el campo médico y biomédico [91].

La estrategia de U-Net consiste en complementar una red neuronal de contracción con capas sucesivas, donde las operaciones de agrupamiento son reemplazadas por operadores de aumento de tamaño para lograr salidas de mayor resolución en las imágenes de entrada. Se caracteriza por tener una estructura en forma de U, donde la primera parte de la red es una serie de capas convolucionales que van reduciendo la resolución de la imagen, mientras que la segunda parte de la red es una serie de capas convolucionales transpuestas que van aumentando la resolución de la imagen. Además, U-Net utiliza una técnica de concatenación de características, que ayuda a preservar la información de la imagen original durante el proceso de reducción y aumento de resolución.

En la Figura 4.12 se ilustra un ejemplo del conjunto de capas de convolución y de capas de “max pooling” que permiten crear un mapa de características de una imagen y reducir su tamaño para disminuir el número de parámetros de la red en la arquitectura U-Net.

La arquitectura U-Net ha demostrado ser muy efectiva en la segmentación de imágenes médicas como, por ejemplo, en la segmentación de células o tejidos en imágenes de microscopio, gracias a su capacidad para capturar detalles finos y preservar la información contextual de la imagen. También se ha utilizado en otros campos, como en la segmentación de carreteras, imágenes de satélite o de cultivos para agricultura de precisión [94].

4.4.2 Segmentación basada en redes FPN

FPN (Feature Pyramid Network) es un extractor de características que toma una imagen de una sola escala de tamaño arbitrario como entrada y genera mapas de características de tamaños proporcionales en múltiples niveles, de manera completamente convolucional [60]. FPN fue propuesta por Lin et al. en 2017 [61] y se basa en la idea de construir una pirámide de características que combina múltiples niveles de representaciones semánticas de la imagen. Es un tipo de arquitectura de red neuronal profunda utilizada en tareas de visión artificial, como la

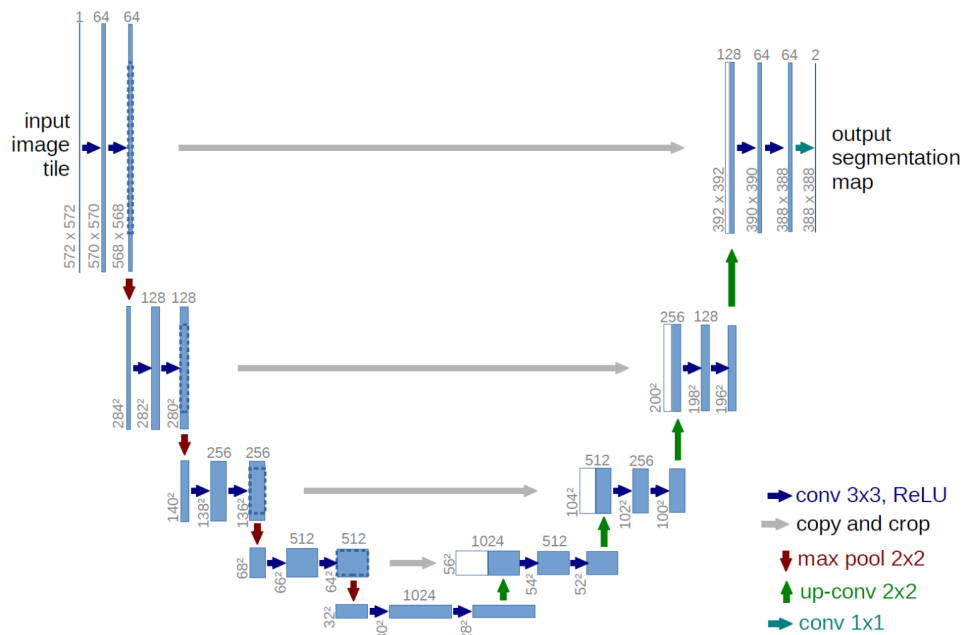


Figura 4.12: Esquema funcional de una red de segmentación convolucional U-Net, en forma de “U” [97].

detección de objetos y la segmentación de imágenes. Esta aproximación combina características de baja resolución y alta semántica con características de alta resolución y baja semántica a través de una vía descendente y conexiones laterales.

La arquitectura FPN utiliza una red convolucional para extraer características de una imagen en diferentes escalas y niveles de detalle, y luego utiliza una técnica de “upsampling” (aumento de tamaño) para combinar y fusionar estas características en una única representación de la imagen. Esto permite que la red tenga una comprensión más completa de la imagen, desde características de alto nivel hasta detalles finos. Los diferentes niveles de escalas conforman una “pirámide”, como se aprecia en la Figura 4.13.

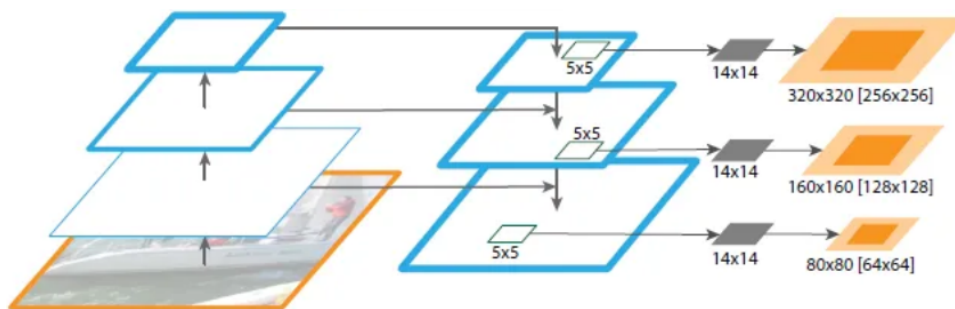


Figura 4.13: Esquema funcional de una red de segmentación convolucional FPN, en forma de pirámide.[18]

FPN es especialmente útil para tareas de detección de objetos y segmentación de imágenes en las que se requiere una comprensión detallada de las características de la imagen en diferentes escalas y niveles de detalle. La combinación de características de múltiples niveles de la pirámide

ayuda a la red a detectar objetos de diferentes tamaños y formas, y a distinguir entre objetos cercanos y superpuestos en la imagen.

4.5 Resumen de las técnicas de segmentación utilizadas en la tesis

Una vez conocidas las principales técnicas de segmentación, tanto basadas en visión artificial como en aprendizaje profundo, merece la pena particularizar cuáles de ellas se han utilizado los dos problemas abordados en la presente tesis.

En el primer caso de estudio, relacionado con la agricultura de precisión, se han utilizado diversas técnicas de segmentación. Por un lado, se ha considerado la segmentación basada en umbral y la segmentación basada en doble umbral (descritas en la Sección 4.1); y la segmentación basada en regiones, concretamente, la basada en crecimiento de regiones, la Quadtree y la multi-resolución (descritas en la Sección 4.2). Por otro lado, de forma adicional, se ha resuelto el problema de agricultura de precisión utilizando métodos de Deep learning. Específicamente, se han considerado las segmentaciones basada en U-Net y en FPN, descritas en las Secciones 4.4.1 y en 4.4.2, respectivamente.

En el segundo caso de estudio, basado en la implementación de un algoritmo sobre plataformas cuánticas, el algoritmo de segmentación clave ha sido el explicado en la Sección 4.1, concretamente la segmentación basada en doble umbral, que resultaba útil para resaltar con precisión regiones de interés específicas en una imagen determinada.

5. Caso de estudio de agricultura de precisión

En este capítulo se explica con detenimiento el primer caso de estudio abordado en la tesis. Este caso de estudio consiste en la implementación de una nueva metodología para ayudar a mejorar la eficiencia en la evaluación de herbicidas. El resultado de dicha investigación es una herramienta automática para cuantificar el porcentaje de malezas y de plantas de interés (girasoles) presentes en una determinada área. Para llevar a cabo este análisis se procesan imágenes del campo de cultivo tomadas con una cámara multi-espectral de cuatro bandas. En primer lugar, se mejora la calidad de las imágenes de cada banda mediante un preprocesado. Después, se compone la imagen multi-espectral utilizando las cuatro imágenes de las cuatro bandas. Y, posteriormente, las imágenes multi-espectrales resultantes se clasifican en tres clases (suelo, girasol y maleza) mediante un novedoso algoritmo implementado con el software eCognition, al que hemos denominado clasificación basada en técnicas de visión artificial con objetos (ClasOb de aquí en adelante). Adicionalmente, los resultados obtenidos al realizar la clasificación se han comparado con dos métodos de segmentación/clasificación basados en aprendizaje profundo (U-Net y FPN).

La estructura de este capítulo es la siguiente. La Sección 5.1 describe el caso de estudio que se ha abordado, así como el sistema de imagen que se ha empleado y los detalles del experimento de campo para capturar las imágenes de entrada. Además, se realiza una revisión del estado del arte del co-registro de imágenes en agricultura de precisión. La Sección 5.2 detalla el preprocesado que se ha llevado a cabo, tanto para calibrar las lentes como para corregir la distorsión y la perspectiva de las imágenes. En la Sección 5.3 se explica cómo se ha realizado la composición de las imágenes multi-espectrales. En la Sección 5.4 se describe en detalle el algoritmo ClasOb de clasificación automática de los píxeles en la imagen multi-espectral utilizando el software eCognition. Adicionalmente, se detallan otros dos enfoques que se han considerado a la hora de realizar la clasificación. Se trata de dos algoritmos de aprendizaje profundo, denominados U-Net y FPN. En la Sección 5.5 se discuten los resultados de los procesos de calibración y clasificación. En cuanto al proceso de calibración, la evaluación se realiza en términos del error promedio (en píxeles). Con respecto al proceso de clasificación, se presenta un análisis experimental en

términos de la métrica de Intersección sobre Unión (IoU). Finalmente, la Sección 5.6 muestra las conclusiones y los posibles trabajos futuros.

5.1 Definición del caso de estudio en agricultura de precisión

La importancia de la agricultura de precisión radica en su capacidad para aumentar la productividad, reducir costes y minimizar el impacto ambiental de la producción de cosechas. Concretamente, puede ayudar a reducir la contaminación del agua y del suelo, ya que se pueden evitar aplicaciones excesivas de fertilizantes y pesticidas. Sin embargo, la medición de la eficacia de los herbicidas recién desarrollados, se lleva a cabo, en la mayoría de los casos, utilizando métodos cualitativos, también conocidos como métodos de evaluación visual, a pesar de sus limitaciones [108]. Por lo tanto, en esta tesis se ha desarrollado una metodología basada en técnicas de visión artificial para automatizar la identificación de malas hierbas en pruebas de herbicidas.

Esta metodología se basa en el uso de imágenes multi-espectrales y técnicas de análisis de objetos (clúster de píxeles) para la clasificación de imágenes de cultivos. Esta metodología consta de dos pasos principales: realizar un co-registro preciso de las imágenes multi-espectrales mediante una calibración específica de la cámara; y clasificar los píxeles de las imágenes multi-espectrales obtenidas en tres clases (mala hierba, girasol y suelo), utilizando un novedoso algoritmo desarrollado con el software eCognition. Merece la pena destacar que la metodología propuesta en este trabajo es potencialmente aplicable a otros cultivos y malas hierbas.

Las imágenes del cultivo se han capturado utilizando una cámara multi-espectral (detallada en la Sección 5.1.2), que permite capturar imágenes en cuatro bandas. Posteriormente, estas bandas se superponen para componer imágenes multi-espectrales que serán las que nos proporcionarán información del cultivo. Una de las técnicas más utilizadas para la composición de imágenes multi-espectrales es el co-registro de las bandas de interés [6, 9, 113]. Sin embargo, las imágenes capturadas por las cámaras multi-espectrales muestran efectos significativos de desalineación de bandas debido a la distorsión de la lente y a los diferentes puntos de vista de cada lente o sensor [49, 50]. Por lo tanto, para obtener información espectral y geométrica precisa, es necesario aplicar un método de corrección de distorsión geométrica y un co-registro banda a banda [3].

Una vez que la imagen ha sido preprocesada para corregir las distorsiones y co-registrada, puede utilizarse para monitorizar indicadores relacionados con la agricultura, como, por ejemplo, para computar la cantidad de maleza en un cultivo de girasoles. Para esta etapa de monitorización es muy común utilizar procesos de segmentación y de clasificación que pueden tomar como punto de partida la imagen co-registrada.

La Figura 5.1 muestra un esquema detallado de las etapas que se han llevado a cabo en este caso de estudio. En primer lugar, se lleva a cabo un preprocesado de la imagen que consiste en: la calibración geométrica de las lentes de la cámara multi-espectral y la corrección de la distorsión y de la perspectiva de las imágenes del campo de cultivo. Posteriormente, se realiza la composición de las imágenes multi-espectrales con las que trabajará el sistema de clasificación.

Finalmente, gracias al proceso de clasificación, todos los píxeles de la imagen se clasifican en una de las siguientes clases: suelo, girasol o maleza.

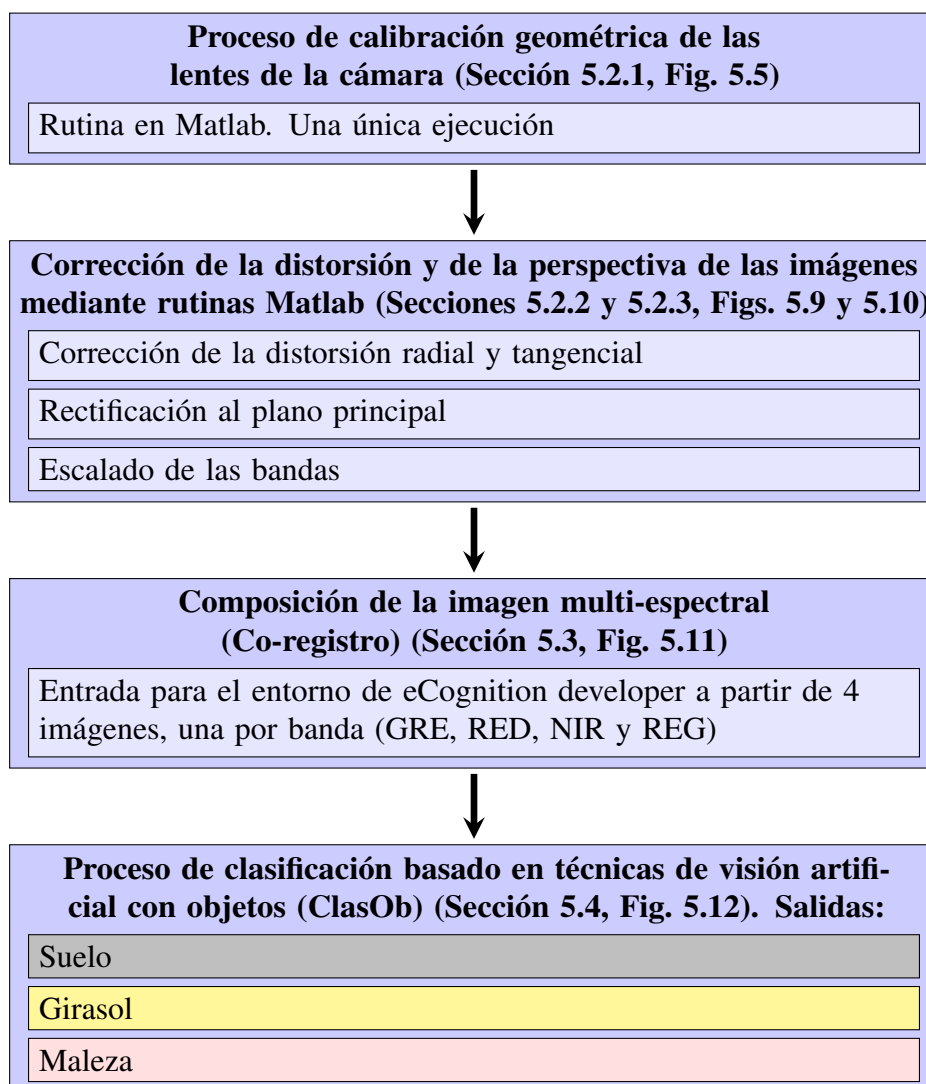


Figura 5.1: Esquema general que indica los principales pasos a seguir en el proceso de identificación de malas hierbas utilizando imágenes multi-espectrales.

Para el desarrollo de la herramienta automática de evaluación de herbicidas, se han llevado a cabo una serie de implementaciones utilizando técnicas de preprocesado de imágenes (calibración geométrica de las lentes de la cámara y corrección de la distorsión radial, tangencial y de la perspectiva de las imágenes) y técnicas de segmentación y clasificación, a través de las librerías de Matlab (Computer Vision Toolbox y Camera Calibration Toolbox) y Trimble eCognition. Matlab ha permitido realizar la calibración de las diferentes lentes, así como un preprocesado de la imagen. El software eCognition ha permitido un procesado multi-espectral basado en clusterización para poder realizar las labores de segmentación y clasificación de los píxeles de la imagen. De forma paralela, se ha utilizado la librería Tensorflow para entrenar varias redes neuronales con las imágenes y, de este modo, comparar la calidad de los resultados obtenidos con la herramienta desarrollada, con métodos del estado del arte basados en aprendizaje automático.

5.1.1 Estado del arte de la segmentación de imágenes multi-espectrales en agricultura de precisión

Como se ha comentado anteriormente, para monitorizar indicadores relacionados con la agricultura utilizando cámaras multi-espectrales es muy común implementar procesos de segmentación y clasificación. Estos procesos pueden tomar como punto de partida la imagen co-registrada. La automatización de ambos procesos ha sido estudiada en los últimos años. En [64] se revisan las técnicas de segmentación de imágenes más relevantes aplicadas a las plantas. En [119] se presenta un método automatizado para segmentar plantas de frambuesa del fondo utilizando un espectro seleccionado. Esta segmentación se realizó mediante umbrales y teoría de grafos. Centrándonos en la segmentación de imágenes de hojas de plantas, que es un punto importante para la clasificación de enfermedades, en [104] se lleva a cabo utilizando un algoritmo de optimización de nube de partículas en un cultivo de girasol.

Por otro lado, si se considera el Aprendizaje Profundo (Deep Learning o DL) para segmentar y clasificar en el campo de la agricultura, también se pueden encontrar algunos trabajos en la literatura. Por ejemplo, en [88] los autores exploran la idoneidad de diversas arquitecturas CNN para la segmentación semántica de cultivos como el algodón, el maíz, etc. Con respecto a la segmentación y clasificación de cultivos de girasol utilizando imágenes multi-espectrales, que es uno de los focos de la tesis, el trabajo más similar al nuestro es el propuesto en [26]. En ese trabajo, se utilizan datos RGB e información del infrarrojo cercano (NIR) para generar imágenes sintéticas multi-espectrales de cuatro canales para entrenar el modelo utilizando enfoques basados en DL. Para ampliar los datos de entrenamiento, se consideran Redes Generativas Antagónicas condicionales (GAN por sus siglas en inglés). Una de las principales limitaciones de los enfoques de DL es que requieren un gran número de imágenes de entrenamiento para obtener una precisión aceptable. Sin embargo, en ocasiones no es posible obtener una cantidad suficiente de imágenes, como es el caso de estudio que hemos abordado en esta tesis, por lo tanto, otros enfoques resultan más relevantes.

5.1.2 Sistema de imagen

Las fotografías que se han considerado para el presente estudio han sido proporcionadas por el departamento de Ingeniería de la Universidad de Almería. Para su captura, se ha utilizado una cámara multi-espectral Parrot Sequoia¹ de cuatro bandas.

La cámara Parrot Sequoia tiene una resolución de sensor de 1280×960 píxeles, 1.2 megapíxeles, un tamaño de 4.8×3.6 mm, tiene cuatro sensores y recopila datos en cuatro bandas espectrales discretas: Verde (GRE, 550 nm, con un ancho de banda de 40 nm), Rojo (RED, 660 nm, con un ancho de banda de 40 nm), Banda Roja (REG, 735 nm, con un ancho de banda de 10 nm) e Infrarrojo Cercano (NIR, 790, con un ancho de banda de 40 nm). Además, consta de una cámara RGB clásica independiente. Se puede montar un sensor de luz solar junto con la cámara para una corrección radiométrica precisa. El dispositivo es un sensor de luz solar completamente integrado que captura y registra las condiciones de iluminación actuales y calibra automáticamente las

¹<https://parrot.com/>

salidas de la cámara para que las mediciones sean absolutas. La Figura 5.2 muestra una cámara Sequoia con los sensores de imagen (NIR, RED, GRE y REG) y el sensor de luz solar.



Figura 5.2: Cámara Parrot Sequoia donde se observan las diferentes cámaras para las bandas de interés, así como el sensor de luz solar.

El hecho de utilizar una cámara de este tipo a corta distancia implica una serie de problemas, entre los que destacamos dos: la posición relativa de las lentes hace que no se perciba exactamente la misma escena desde cada cámara (error de perspectiva), y las características intrínsecas de cada lente generan niveles diferentes de distorsión óptica. Es importante aclarar que el nivel de distorsión de barril es tan fuerte que se puede percibir a simple vista en una cuadrícula, tal como se aprecia en la Figura 5.3.

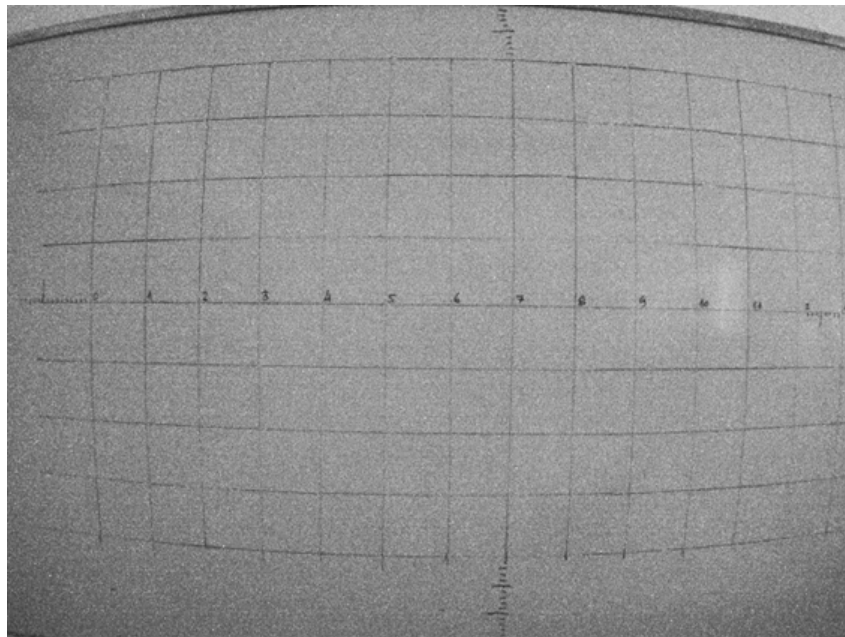


Figura 5.3: Fotografía del sensor GRE de una cuadrícula tomada con la cámara Sequoia. Se puede percibir a simple vista la fuerte distorsión de barril.

5.1.3 Experimento de campo

El campo de cultivo donde se han capturado las imágenes se encuentra en Córdoba, España (coordenadas UTM 349042, 4198307, zona 30). Se trataba de un cultivo de girasol (*Helianthus annuus*, L.) cuando las plantas tenían aproximadamente de cuatro a ocho hojas y una altura de aproximadamente 8 a 20 cm. Además, el suelo contenía algunas especies de malas hierbas en desarrollo temprano (*Chenopodium album*, L., *Convolvulus arvensis*, L.; y *Cyperus rotundus*, L.). Se utilizó un marco rectangular de 57×47 cm como referencia para escalar e identificar puntos equivalentes en todas las bandas. El marco se colocó en el suelo envolviendo algunas plantas de girasol y malas hierbas (Figura 5.4). Se tomó una fotografía con la cámara multi-espectral, que estaba montada en una plataforma que incluía la fuente de alimentación, reguladores de voltaje y un monitor. Esta plataforma fue transportada por un operador que enmarcó la escena, mientras que otro operador se encargó de tomar la fotografía de la cámara Sequoia a través del Wi-Fi del teléfono móvil.

La distancia desde la cámara hasta las plantas era de aproximadamente entre 1 y 1.5 metros, lo que equivale a las distancias utilizadas cuando se utilizan vehículos terrestres o UAV a baja altitud de vuelo como plataformas [65]. En el estudio experimental se consideraron un total de 26 imágenes multi-espectrales del cultivo de girasol (cada una consistente en cuatro bandas).



Figura 5.4: Fotografía en bruto del espectro NIR donde se muestra el marco utilizado en el experimento.

5.2 Preprocesado de las imágenes del caso de estudio

El co-registro automático de las bandas de las imágenes que proporciona la cámara Sequoia no es lo suficientemente preciso para la detección de malas hierbas. La superposición de las diferentes

bandas espectrales que componen una imagen multi-espectral requiere un complejo preprocesado, ya que las características intrínsecas de cada lente, sensor y su posición relativa hacen que exista una distorsión diferente para cada una de las imágenes individuales que compondrán la imagen multi-espectral

Por lo tanto, se ha considerado una etapa de preprocesado de dichas imágenes para cada banda específica. En esta sección se explica el preprocesado que se ha llevado a cabo. En primer lugar, se realiza un proceso de calibración geométrica de las lentes de la cámara, evaluando el resultado para comprobar su nivel de precisión. Esto se ha realizado tomando fotografías utilizando un patrón de referencia para calcular los parámetros intrínsecos de cada una de las lentes. Una vez caracterizadas las lentes, se ha realizado una corrección de todas las imágenes de campo recopiladas, así como un escalado y designación de la región de interés mediante la selección de cuatro puntos de referencia (las cuatro esquinas del marco) para cada fotografía en cada una de las bandas. El paso final ha sido realizar una transformación lineal de las imágenes mediante una homografía para traer al plano principal una imagen rectificadas de la región de interés. Teniendo las cuatro imágenes de las bandas sobre la misma región de interés, con la distorsión específica de cada lente corregida y rectificadas al plano principal, ha sido posible realizar el co-registro de las cuatro bandas y formar la imagen multi-espectral con la que, posteriormente, se realiza el proceso de clasificación.

5.2.1 Proceso de calibración geométrica de las lentes de la cámara

La calibración se ha realizado utilizando el algoritmo de calibración de cámara estenopeica (explicado en la Sección 3.5.1) presente en la librería Camera Calibration Toolbox de Matlab. Este algoritmo está basado en el modelo propuesto por Jean-Yves Bouguet [11] e incluye el modelo de cámara estenopeica [131] y la distorsión de las lentes [42]. El modelo de cámara estenopeica no tiene en cuenta la distorsión de las lentes porque una cámara estenopeica ideal no tiene lentes. Para representar con precisión una cámara real, el modelo de cámara completo utilizado por el algoritmo incluye distorsión de lentes radial y tangencial.

Para corregir la distorsión de la cámara se ha considerado un proceso de calibración de lentes que implica el cálculo de los parámetros intrínsecos de las lentes, tal y como se explicó en la Sección 3.5.3 [98]. Se ha utilizado un damero de celdas de 28×28 mm como objeto de referencia. El objeto de referencia ha sido fotografiado desde diferentes ángulos.

La Figura 5.5 muestra un esquema del proceso de calibración de la lente, donde las entradas son cuatro conjuntos de calibración de imágenes del damero (un conjunto por banda: GRE, NIR, RED y REG). Se utilizó un conjunto de 95 imágenes del damero para el proceso de calibración.

Evaluación de la calibración de lentes

Se ha desarrollado una rutina en Matlab para medir la calidad del proceso de calibración de lentes descrito en la Sección 5.2.1. El esquema del proceso de evaluación de la calibración de las lentes se muestra en la Figura 5.6. Las entradas son los parámetros intrínsecos de cada

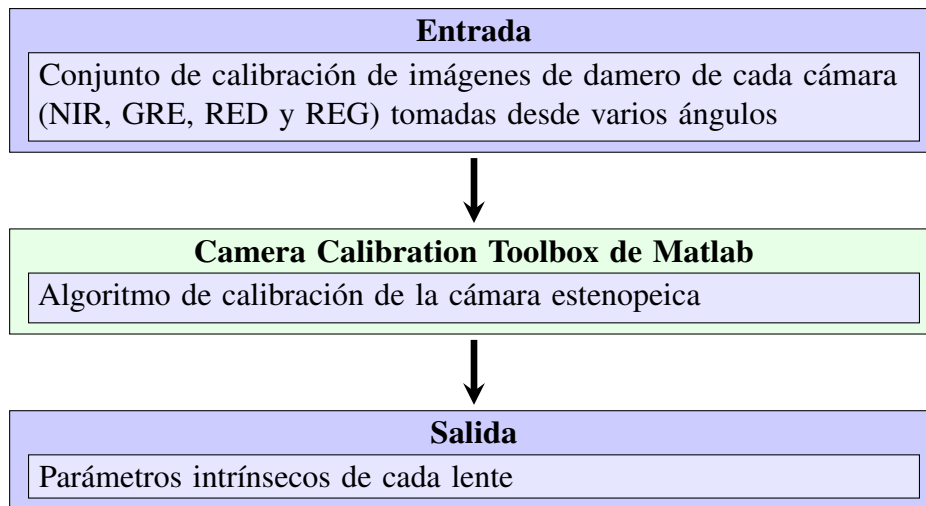


Figura 5.5: Diagrama del proceso de calibración de las lentes.

lente (calculados en la Sección 5.2.1) y cuatro conjuntos de prueba de imágenes del damero, un conjunto por sensor.

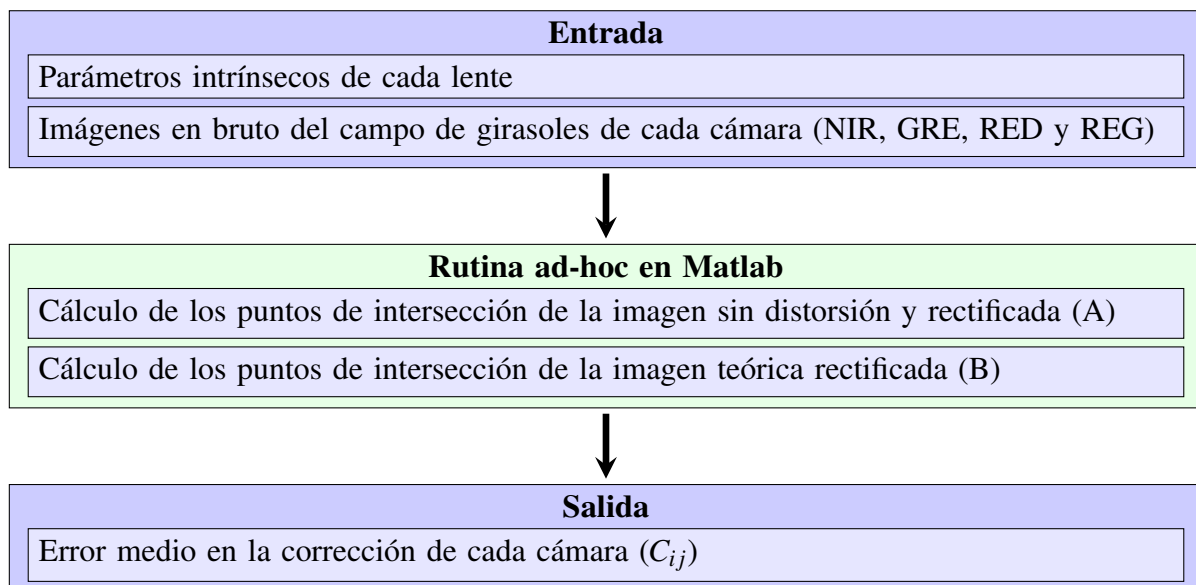


Figura 5.6: Diagrama del proceso de evaluación de la corrección de las lentes.

Las intersecciones de la imagen rectificadas se almacenan en la matriz A . Luego, se calculan y almacenan en la matriz B los puntos teóricos que deberían estar en los puntos de intersección. Finalmente, se calcula $C = (c_{ij}) = |(a_{ij} - b_{ij})|$, donde $A = (a_{ij})$ y $B = (b_{ij})$. La matriz C contiene el error de intersección por punto. El valor medio de (c_{ij}) es el error promedio de la imagen resultante del proceso de calibración. La Figura 5.7 muestra las 16 imágenes de prueba para la banda GRE con la distorsión de lentes eliminada. En color verde se marcan las cuatro intersecciones exteriores necesarias para llevar la imagen al plano principal y corregir la distorsión de perspectiva. El resultado de este paso se puede observar en la Figura 5.8. Las intersecciones de la imagen rectificadas se almacenan en la matriz A . Luego, se calculan y

almacenan en la matriz B los puntos teóricos que deberían estar en los puntos de intersección. Finalmente, $C = (c_{ij}) = |(a_{ij} - b_{ij})|$, donde $A = (a_{ij})$ y $B = (b_{ij})$.

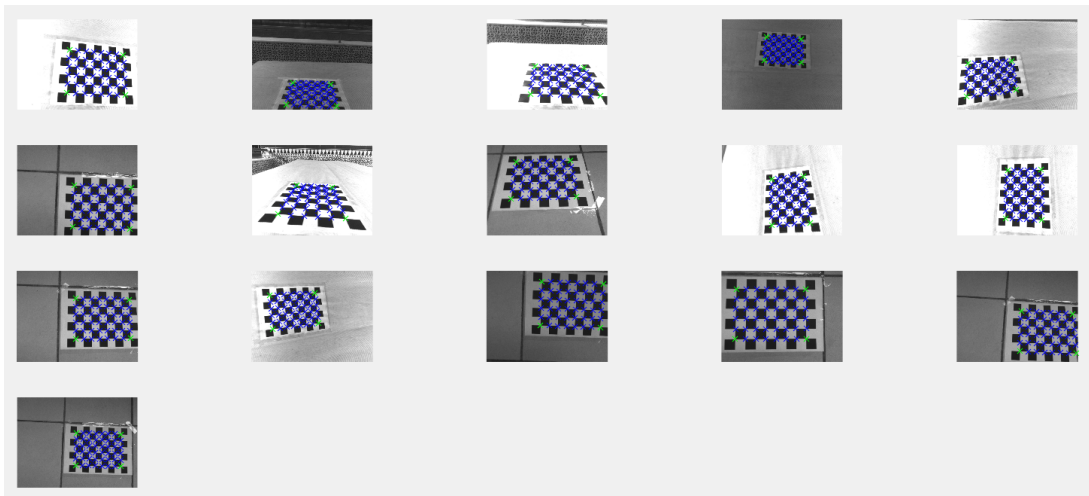


Figura 5.7: Imágenes de prueba del damero con distorsión de barril corregida en la banda GRE.

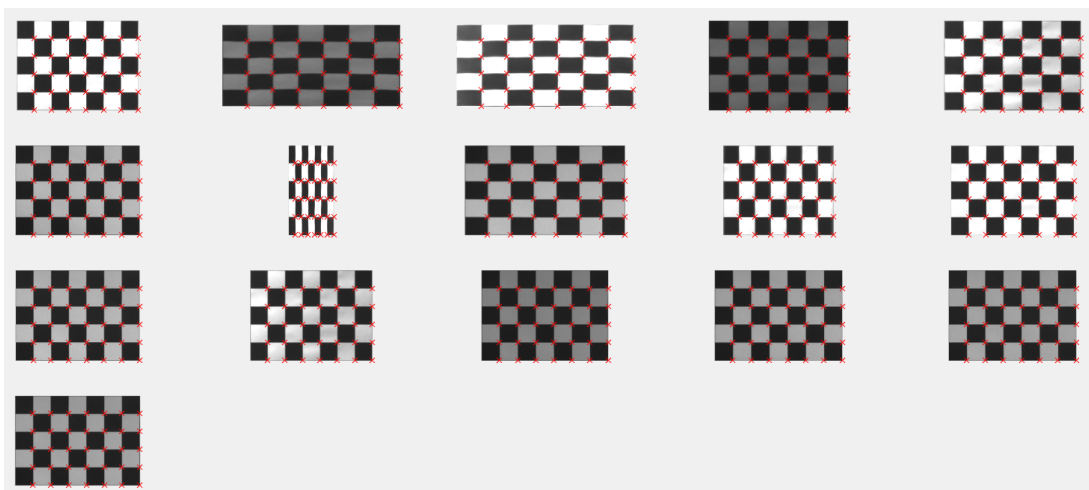


Figura 5.8: Imágenes de prueba del damero rectificadas en la banda GRE. Las cruces rojas marcan las intersecciones que se almacenarán en la matriz $A = (a_{ij})$. La matriz $B = (b_{ij})$ se calculará utilizando las posiciones teóricas de las intersecciones mostradas.

Los resultados de la calibración de lentes se discutirán en la Sección 5.5.1.

5.2.2 Corrección de la distorsión radial y tangencial de las imágenes del caso de estudio

Como parte del preprocesado de las imágenes del caso de estudio, se realiza una corrección de la distorsión radial y tangencial de todas ellas tal y como se explicó en la Sección 3.5.4. Esta corrección se ha implementado a través de una rutina utilizando la librería Computer Vision Toolbox de Matlab. En este proceso, que puede observarse en la Figura 5.9, las imágenes en bruto

del cultivo de girasoles se han corregido automáticamente para eliminar la distorsión empleando la función “undistortImage” de Computer Vision Toolbox y los parámetros intrínsecos de cada lente (calculados en la Sección 5.2.1). Al aplicar la rutina sobre las distintas imágenes se obtiene como resultado una serie de imágenes sin distorsión de barril que, posteriormente, se utilizarán como entrada en las siguientes fases del proceso de identificación de malas hierbas.

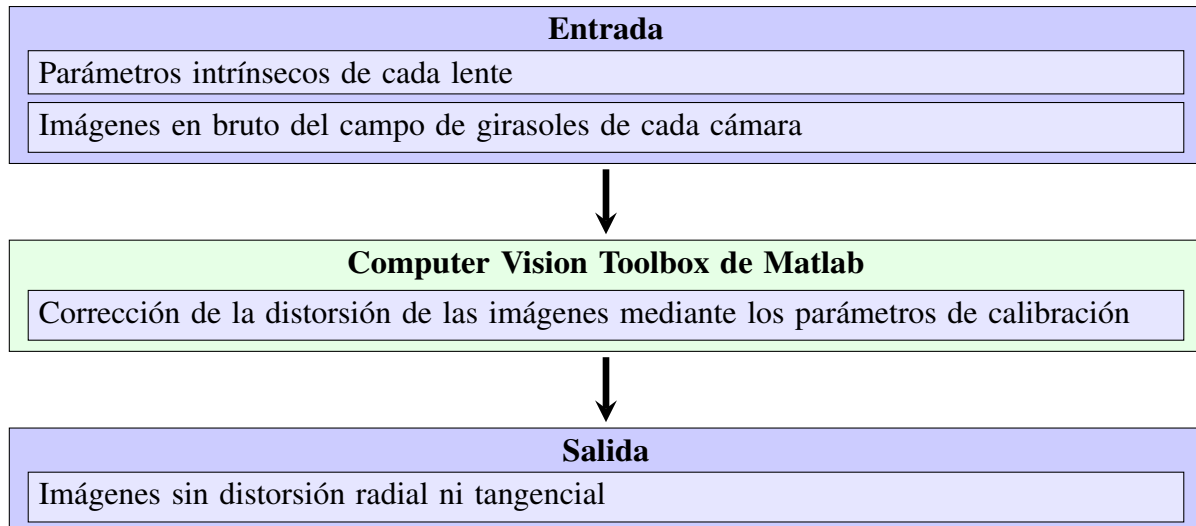


Figura 5.9: Esquema del proceso de corrección de distorsión geométrica.

5.2.3 Rectificación al plano principal y escalado

El proceso de rectificación se ha llevado a cabo detectando cuatro puntos de referencia (las esquinas interiores) en los marcos colocados en las fotografías y realizando una homografía para llevarlos al plano principal. Esta homografía consiste en una transformación proyectiva [35, 36], tal como se comentó en la Sección 3.5.5, utilizando la función “fitgeotrans” de Matlab.

Finalmente, las imágenes de los cuatro espectros se escalan al tamaño de la mayor de todas ellas, a la que llamaremos banda de referencia, para no perder resolución. Mediante una transformación afín, la Toolbox de Matlab realiza el escalado del resto de las bandas ajustando cada vértice del marco a su equivalente en la banda de referencia.

Este proceso completo se ilustra en la Figura 5.10. El proceso acumula pequeños errores, ya que la selección de los vértices no es perfecta, y la perspectiva de las 4 lentes es diferente, ya que las lentes ocupan posiciones diferentes en el espacio.

5.3 Composición de imágenes multi-espectrales (co-registro)

Después de corregir las distorsiones de lente y perspectiva, y de escalar las imágenes, se tienen cuatro imágenes que pueden co-registrarse y formar una imagen multi-espectral.

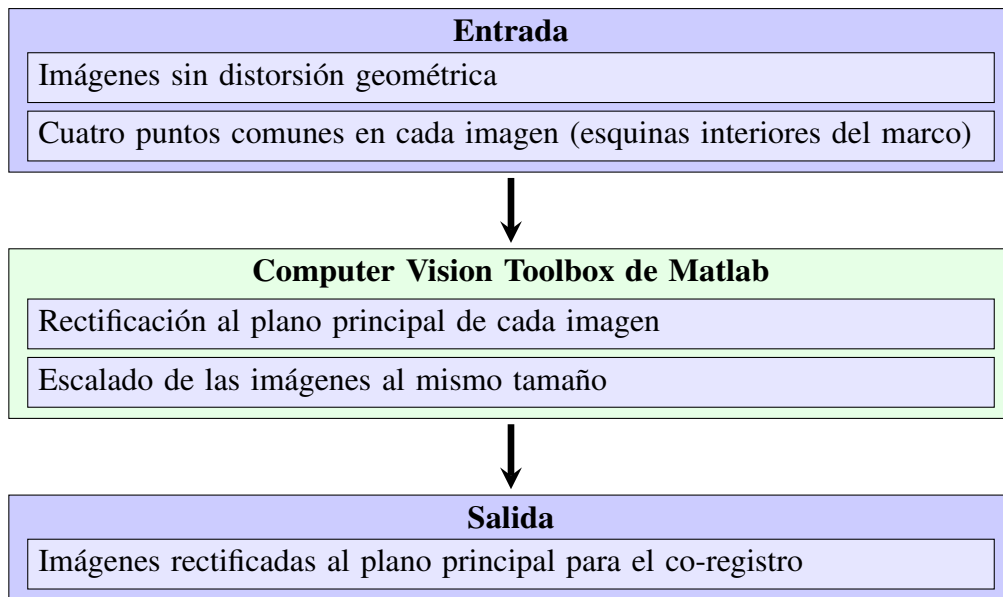


Figura 5.10: Esquema del proceso de rectificación al plano principal de las imágenes y su escalado.

Las imágenes multi-espectrales compuestas son las entradas del proceso de clasificación basado en técnicas de visión artificial con objetos (ClasOb) explicado en la Sección 5.4.1.

La Figura 5.11 muestra todo el proceso de co-registro de banda a banda para la composición de una imagen multi-espectral a partir de las imágenes en bruto del cultivo de girasoles. Concretamente, se toma como ejemplo la banda NIR. La Figura 5.11(d) es una representación RGB de las tres bandas más representativas (GRE, NIR y RED). Estas imágenes RGB serán la entrada para los métodos de DL explicados en la Sección 5.4.2.

5.4 Proceso de clasificación de las imágenes del caso de estudio

Las imágenes preprocesadas que han compuesto una imagen multi-espectral son la entrada de los diversos algoritmos de clasificación estudiados.

En esta sección definiremos el problema de la segmentación y clasificación de cultivos en las imágenes multi-espectrales. Para ello, se han utilizado dos estrategias diferentes y se ha realizado una comparación entre ambas.

Estos enfoques son: (1) el diseño de un algoritmo novedoso de visión artificial usando eCognition y utilizando un procesamiento de imágenes basado en objetos (ClasOb), y (2) la clasificación automática de cultivos utilizando redes neuronales convolucionales (en concreto dos modelos: FPN y U-Net).



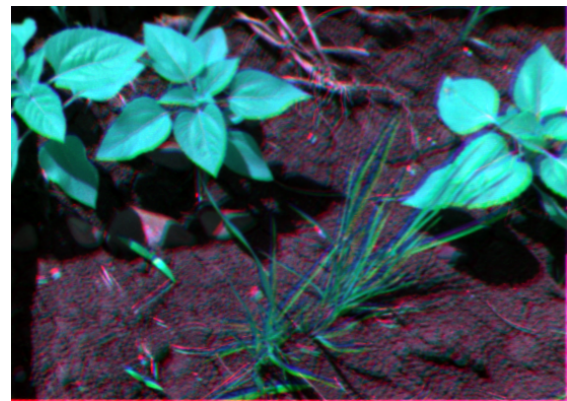
(a)



(b)



(c)



(d)

Figura 5.11: (a) Fotografía en crudo en la banda NIR para la muestra 38; (b) Banda NIR de la muestra 38 después de corregir la distorsión de la lente; (c) Banda NIR después de corregir la perspectiva tomando como referencia las cuatro esquinas internas del marco; y (d) Representación RGB de tres bandas superpuestas (GRE, NIR y RED).

5.4.1 Proceso de clasificación basado en técnicas de visión artificial con objetos (ClasOb)

En esta sección se describe un algoritmo novedoso para la clasificación de imágenes multi-espectrales, al que hemos denominado ClasOb.

El software desarrollado se fundamenta en una estrategia de procesamiento de imágenes basada en objetos (clústeres de píxeles con características similares). En ocasiones las técnicas de procesamiento de imágenes basadas en objetos pueden resultar más oportunas que las basadas en píxeles, ya que éstas últimas tienen dificultades para explotar de forma adecuada el conocimiento técnico experto o la información contextual. El procesamiento de clústeres de píxeles con características similares segmenta la imagen en objetos significativos de varios píxeles de diferentes tamaños, basados tanto en las características espectrales como espaciales de grupos de píxeles [30].

La Figura 5.12 muestra un esquema de los procesos llevados a cabo como parte del algoritmo ClasOb propuesto. En primer lugar, se representa la entrada como una imagen multi-espectral. Esta entrada se verá sometida a tres procesos dentro del software eCognition. El primer proceso es la clasificación entre materia vegetal y suelo, utilizando una operación de umbralización y un parámetro derivado de la imagen multi-espectral llamado NDVI, que comentaremos a lo largo de este capítulo. El segundo proceso es la detección de zonas concretas en la imagen que llamaremos “nodos semilla”, que permitirán el inicio de un algoritmo iterativo de crecimiento. Estos nodos semilla y su crecimiento harán un uso intensivo de la segmentación Quadtree. El tercer proceso se basa en estrategias de crecimiento sobre el resultado de varias segmentaciones Quadtree teniendo en cuenta factores de luminosidad y forma de los objetos definidos. Finalmente, como salida del algoritmo ClasOb obtendremos la imagen multi-espectral clasificada en tres elementos: girasol, maleza y suelo. A lo largo de este capítulo se explicará todo este proceso en detalle.

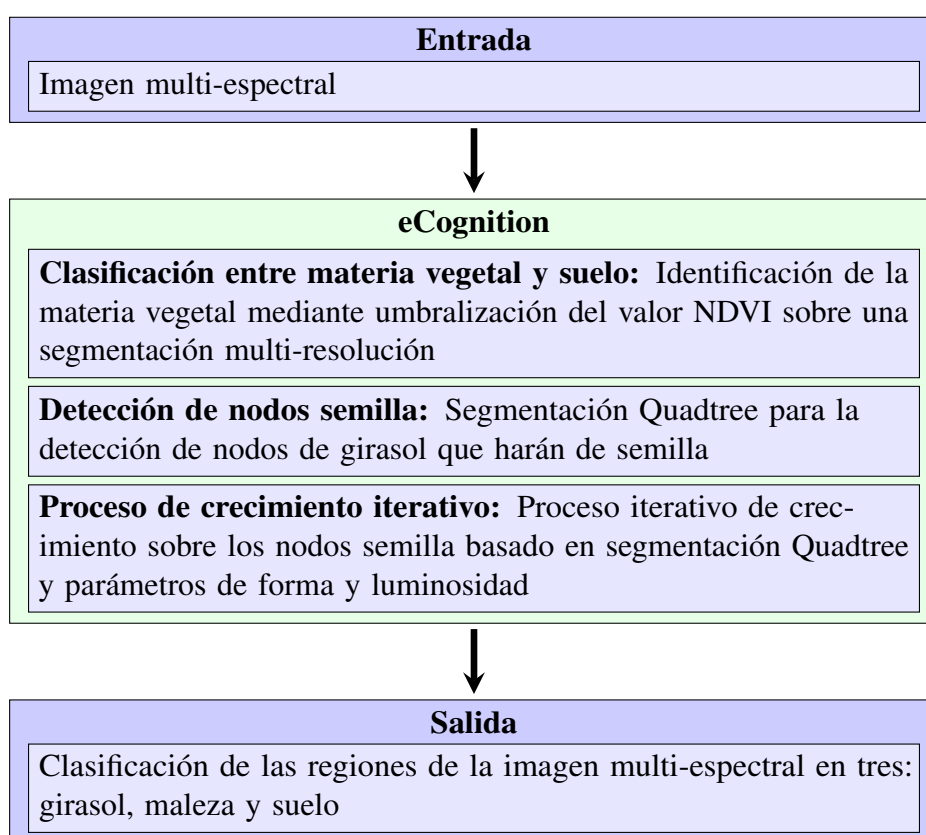


Figura 5.12: Diagrama del proceso de clasificación.

Las entradas del proceso son las imágenes multi-espectrales generadas previamente y el resultado es la clasificación de todos los píxeles de la imagen en tres clases: girasol, maleza y suelo. Cabe destacar que, en los experimentos realizados, no ha sido posible distinguir el girasol de la maleza utilizando solo la firma espectral. Esto es debido a que la firma espectral de las imágenes co-registradas era muy variable por diversos factores: las hojas pueden tener un nivel de iluminación diferente, hay sombras dentro de la planta y, además, las distintas partes de una planta tienen a su vez varias firmas espectrales (por ejemplo, en las ramas, en las hojas en diferentes etapas de crecimiento, etc.). Por tanto, en este caso de estudio, además de procesar las firmas espectrales se han considerado factores geométricos para diferenciar el girasol de la

maleza. Estos factores utilizan como premisa que la planta de girasol es más grande y está más desarrollada que la maleza circundante.

En este trabajo identificamos las plantas de girasol utilizando una estrategia de segmentación basada en crecimiento. Este tipo de algoritmos se basa en un conjunto de puntos iniciales llamados semillas, que crecen anexando regiones adyacentes que tienen propiedades similares (por ejemplo, textura o color) [5]. Estas semillas han sido determinadas utilizando las hojas más descriptivas de los girasoles, teniendo en cuenta factores de luminosidad de cada banda, forma y tamaño.

En el proceso se pueden distinguir tres bloques principales: clasificación entre materia vegetal y suelo, detección de nodos de girasol y un algoritmo iterativo de crecimiento de regiones. Los pasos llevados a cabo en cada bloque se describen a continuación.

En la clasificación entre materia vegetal y suelo, se realizan dos etapas. En primer lugar, se lleva a cabo una clusterización basada en una segmentación multi-resolución (descrita en la Sección 4.3.4), utilizando las cuatro bandas (GRE, NIR, RED y REG) y el Índice de Vegetación de Diferencia Normalizada (NDVI). El factor NDVI es un indicador de la presencia de materia vegetal [116], y se puede formular de la siguiente manera:

$$NDVI = \frac{NIR - RED}{NIR + RED}$$

Según este índice, las plantas tienen valores positivos entre 0.2 y 1. En el caso del estudio, el NDVI se considera el factor determinante para clasificar cada clúster como materia vegetal o suelo. El algoritmo de segmentación multi-resolución minimiza localmente la heterogeneidad promedio de los objetos de imagen para una resolución dada. Este bloque de procesamiento de eCognition segmenta las imágenes en base a una escala proporcionada por el usuario (umbral de homogeneidad) y segmenta la imagen en clústeres de píxeles similares llamados objetos. La Figura 5.13(a) ilustra la salida de este bloque para la muestra 38. Cualquier clúster con un valor de NDVI mayor que 0.2 se clasifica como materia vegetal y se muestra en color verde. Finalmente, agrupamos todos los clústeres de materia vegetal en nodos y descartamos falsos positivos (nodos iguales o inferiores al 0.03% del tamaño total de la imagen, correspondientes a ruido o errores en la perspectiva) (ver Figura 5.13(b)).

El siguiente bloque del proceso de clasificación (ver Figura 5.12) es la detección de nodos de girasol.

La materia vegetal se segmenta utilizando un algoritmo multi-espectral Quadtree (descrito en la Sección 4.3.3) para identificar grandes áreas cuadradas que pertenezcan a la misma hoja (ver Figura 5.13(c)). Las cuatro bandas y la capa de NDVI derivada se consideran como entradas para esta segmentación Quadtree. Se requiere un parámetro de escala grande (del orden de 25000 variaciones de color) para obtener cuadrados grandes que apunten a las hojas más relevantes de las plantas de girasol. El parámetro de escala es un elemento importante al realizar segmentación en teledetección, y su estimación correcta es objeto de estudio en varios artículos [23, 73, 126].

Las imágenes de entrada tienen una profundidad de color de 16 bits (65.536 valores). Para obtener un resultado de segmentación multi-espectral Quadtree ajustado al tamaño de las hojas de girasol, se ha estimado una variación promedio del 7.5% (5.000 variaciones de color en

promedio en cada una de las 5 capas utilizadas) del color en cada banda. Dado que hay 5.000 posibles variaciones en promedio en cada una de las 5 capas, el parámetro de escala considerado ha sido de 25.000.

El siguiente paso consiste en marcar como girasoles los clústeres obtenidos anteriormente debido a su tamaño, forma y nivel de luminosidad. Los mejores candidatos a ser hojas de girasol son clústeres grandes, cuadrados casi perfectos con un alto nivel de luminosidad en la banda NIR. Este paso se enfoca en la banda NIR porque se ha determinado que es la banda más brillante en las imágenes tomadas y esto permite identificar mejor las hojas más desarrolladas y descriptivas para clasificar una planta de girasol.

Los dos factores para clasificar estos clústeres como girasoles han sido el nivel de luminosidad por tamaño de clúster y la compactibilidad de los objetos (la compactibilidad depende de la forma del objeto). La Figura 5.13(d) muestra los nodos marcados como girasoles.

El último bloque del proceso de clasificación (ver Figura 5.12) se basa en un algoritmo de crecimiento de regiones (descrito en la Sección 5.4.1). El algoritmo de crecimiento utiliza clústeres en forma de cuadrados que pueden formar potencialmente parte de la misma hoja. Para identificar posibles candidatos a crecer, se calcula una capa derivada llamada Diff_RED_NIR, obtenida sumando la diferencia promedio a los clústeres vecinos en las bandas RED y NIR. El objetivo de esta nueva capa es identificar clústeres en forma de cuadrados con valores similares en las capas RED y NIR.

El algoritmo de crecimiento de regiones se lleva a cabo en tres etapas. La primera etapa tiene como objetivo marcar como girasol los clústeres de materia vegetal vecinos a los nodos de girasol. Para este caso de estudio, el umbral es $\text{Diff_RED_NIR} > -1000$. El umbral se ha calculado para tener diferencias de valores medios entre las dos bandas (RED y NIR) de menos del 1% relativo. Esto significa que estamos creciendo agregando clústeres muy similares (con un nivel de diferencia relativa muy pequeño en esas dos bandas).

Para evitar crecer hacia ramitas que pueden formar parte de otras plantas, se requiere un factor de forma concreto (compactibilidad < 1.5) para que los clústeres vecinos sean marcados como girasol. Esta etapa se repite diez veces por iteración. La Figura 5.13(e) muestra los resultados del primer paso del algoritmo de crecimiento de regiones.

En la segunda etapa del proceso iterativo, los objetos de materia vegetal encerrados por objetos clasificados como girasoles también se marcan como girasoles. Todas las regiones de girasol se fusionan y crecen hacia cualquier clúster vecino con un borde relativo mayor al 30%. Esto significa que el crecimiento ocurre con clústeres con los que se comparte más de un lado, lo que normalmente sería el 25%. Los objetos de girasol resultantes se fusionan y todos los demás objetos se clasifican como maleza (ver Figura 5.13(f)). La etapa final de este bloque es calcular una segmentación Quadtree multi-espectral en el área de maleza fusionada, que considera como entrada las capas calculadas, las bandas y un parámetro de escala de 25.000 (como se explicó en el proceso de detección de nodos de girasol). La imagen obtenida en este paso se muestra en la Figura 5.13(g). El proceso iterativo se lleva a cabo tres veces, ya que más iteraciones resultan en un crecimiento marginal. Este proceso iterativo da como resultado clústeres con un alto potencial para ser las hojas más representativas de los girasoles.

El último paso del algoritmo de crecimiento consiste en considerar como girasol cualquier clúster de maleza con más del 5% de borde común. Estas serán las partes más oscuras y los bordes de la planta de girasol. Es posible en este punto aceptar pequeñas partes de maleza como falsos positivos. Además, se elimina cualquier clúster pequeño de girasol, ya que se espera que las plantas de girasol tengan un tamaño mayor que la maleza. Es posible identificar erróneamente algunos nodos como girasoles si las malezas forman un gran clúster de hojas pequeñas que pueden ser confundidas como una hoja grande de girasol. Estos falsos positivos se eliminan clasificando cualquier clúster de girasol con un área inferior al 10% del clúster de girasol más grande (en nuestro caso, el umbral ha sido de 3500 píxeles) como maleza. La salida de este paso se muestra en la Figura 5.13(h). La Figura 5.13(i) muestra el resultado del proceso de clasificación, donde los colores amarillo, rojo y negro identifican las regiones de girasol, maleza y suelo, respectivamente.

A modo de resumen, el Algoritmo 2 muestra un esquema general del proceso de clasificación anteriormente descrito.

Algoritmo 2 Algoritmo de clasificación propuesto

Require:

Imágenes *GRE*, *RED*, *NIR* y *REG*

Bloque 1: Clasificación entre materia vegetal y suelo

- 1: **Calcular la capa derivada NDVI:** calcular la capa NDVI utilizando las capas RED y NIR.
- 2: **Segmentación multi-resolución para clusterizar:** realizar una segmentación multi-resolución para clusterizar la materia vegetal utilizando 4 bandas + NDVI.
- 3: **Clasificar la materia vegetal:** aplicar un umbral de 0.2 en la capa NDVI para clasificar los clústeres como materia vegetal. Cualquier otro clúster se declara como suelo.
- 4: **Filtrar nodos falsos positivos:** Los nodos de materia vegetal más pequeños son principalmente ruido que debe ser filtrado.

Bloque 2: Detección de nodos de girasol

- 5: **Agrupar todos los nodos de materia vegetal.**
- 6: **Segmentación Quadtree para clusterizar:** Crear nuevos clústeres para crecer hacia ellos.
- 7: **Determinar nodos semilla:** Se considera la luminosidad NIR, el tamaño del clúster y la compactabilidad (forma) para la detección de semillas para clasificar como girasol.

Bloque 3: Algoritmo de crecimiento de regiones iterativo

- 8: **Calcular capa derivada Diff_RED_NIR:** basada en similitudes NIR y RED con clústeres adyacentes.
 - 9: **for** $i=1$ **to** 3 **do**
 - 10: **for** $j=1$ **to** 10 **do**
 - 11: **Crecimiento de la clase girasol:** Crecer hacia clústeres cercanos basándose en la capa Diff_RED_NIR y la compactabilidad.
 - 12: **Fusionar encerrados:** Crecer hacia clústeres de maleza encerrados y clasificarlos como girasoles.
 - 13: **Fusionar clase de girasol:** Fusionar todos los nuevos clústeres de girasol.
 - 14: **Segmentación Quadtree en la maleza:** Nueva segmentación Quadtree sobre la clase de maleza utilizando las 4 bandas base, NDVI y la capa derivada Diff_RED_NIR.
 - 15: **Crecimiento de clase de girasol:** Crecer hacia clústeres vecinos para obtener los bordes del girasol.
 - 16: **Filtrado final de maleza:** Marcar los nodos de girasol demasiado pequeños como maleza.
 - 17: **return** Clasificación de *suelo*, *girasol* y *maleza*.
-

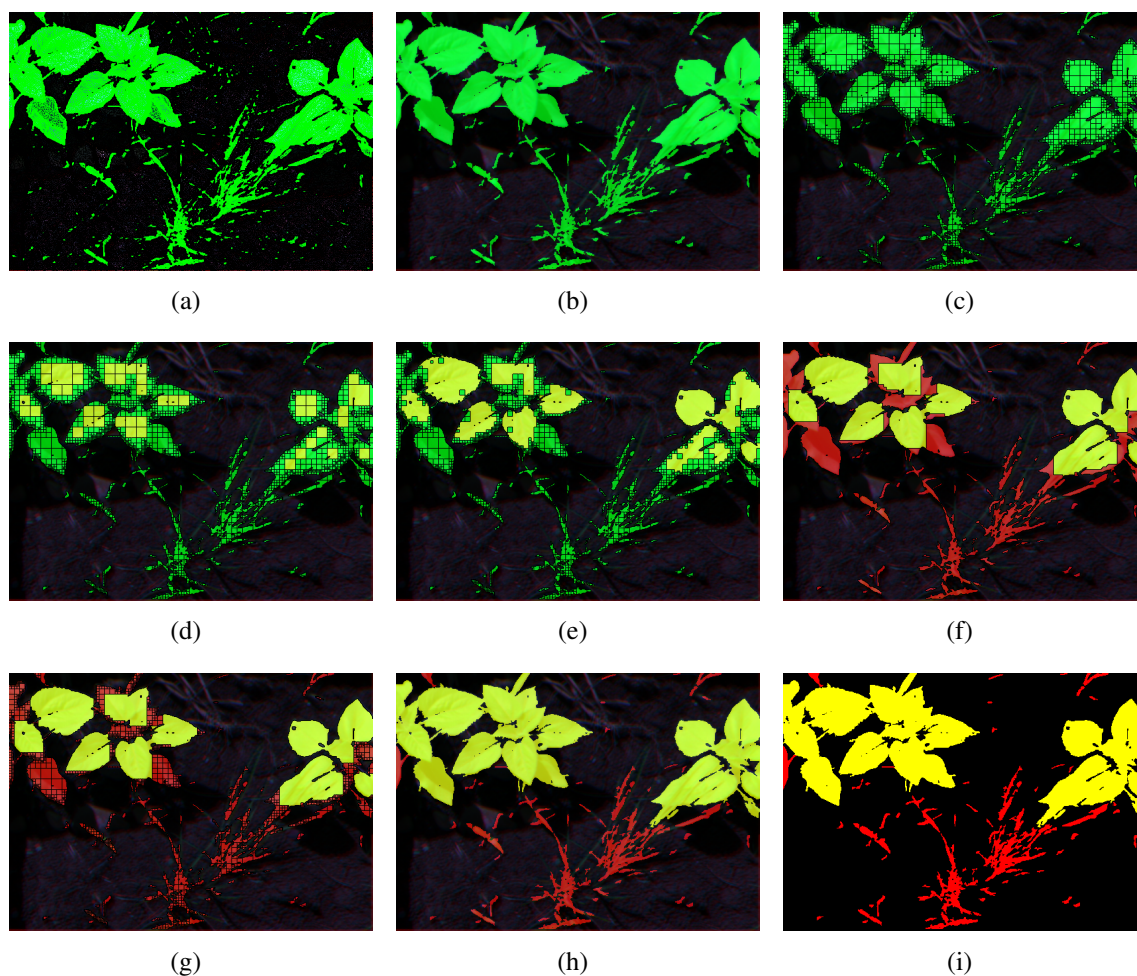


Figura 5.13: (a) Imagen de salida para la muestra 38 después de ejecutar el bloque de clasificación de materia vegetal-suelo. La materia vegetal y el suelo están marcados en color verde y negro, respectivamente; (b) Imagen después de agrupar la materia vegetal en nodos adyacentes y eliminar clústeres con menos del 0.03% del tamaño total de la imagen; (c) Imagen después de ejecutar la segmentación Quadtree multi-espectral; (d) Imagen con los clústeres en forma de cuadrados marcados (semillas); (e) Imagen después de la primera etapa del algoritmo de crecimiento; (f) Imagen de la segunda etapa del algoritmo de crecimiento. Las áreas que potencialmente corresponden a girasol están marcadas en color amarillo y las áreas potenciales de maleza en color rojo; (g) Imagen de la tercera etapa del algoritmo de crecimiento de regiones; (h) Imagen de salida del proceso de clasificación desde el visor de eCognition. Las áreas amarillas representan materia vegetal clasificada como girasol, las áreas rojas representan materia vegetal clasificada como maleza y las áreas negras identifican el suelo; e (i) Imagen de salida del proceso de clasificación.

5.4.2 Enfoques de Aprendizaje Profundo

La comparación del clasificador que hemos propuesto (ClasOb) con otros modelos de clasificación del estado del arte es de gran interés. Entre los métodos de segmentación basados en DL más utilizados se encuentran U-Net y FPN, descritos en las Secciones 4.4.1 y 4.4.2, respectivamente. Estas dos arquitecturas han sido implementadas para comparar los resultados de entrenar las redes neuronales con estos modelos frente a los resultados del clasificador ClasOb.

Los modelos de DL normalmente necesitan utilizar técnicas de pre y post-procesamiento para mejorar la calidad de los resultados obtenidos en problemas complejos. Algunos de los métodos más importantes son: transfer learning, fine-tuning y técnicas de aumento de datos o Data Augmentation (DA). Tanto en las redes neuronales creadas con U-Net como las de FPN se han considerado estas tres técnicas.

La técnica de transfer learning considera el uso de modelos pre-entrenados en una base de datos de millones de instancias (como ImageNet² [92]) como punto de partida para inicializar los pesos de la red, aprovechando así el conocimiento previamente aprendido. Con la técnica de fine-tuning, el nuevo modelo solo tiene que entrenar las últimas capas, lo que requiere menos tiempo para obtener resultados favorables. Por último, la técnica DA se utiliza para aumentar el número de imágenes de entrada aplicando variaciones a las imágenes originales. De esta manera, el modelo tendrá una base de datos más grande para el entrenamiento, lo que lo hará más generalizable. Teniendo en cuenta estas técnicas, se han configurado tres conjuntos de datos de entrenamiento como se describe a continuación:

- **I_Jes:** Considera ImageNet para transfer learning y el conjunto de datos Jesi_05_18 de cultivos de girasol para fine-tuning (146 imágenes RGB). El conjunto de datos Jesi_05_18 fue presentado en [26] y se puede descargar desde el siguiente enlace³. Cada imagen del conjunto Jesi_05_18 ha sido aumentada con 1000 imágenes mediante operaciones básicas en imágenes (rotación, desplazamiento, volteo, zoom y recorte). Además, en la misma página web, los autores proporcionan las máscaras necesarias para el entrenamiento. Estas imágenes consisten en máscaras de segmentación donde se identifican el cultivo, el suelo y la maleza. El conjunto de datos que hemos diseñado se ha denominado I_Jes y contiene un total de 146000 imágenes. El porcentaje de estas imágenes para el entrenamiento y para validación fue del 90% y 10%, respectivamente.
- **Multi:** El entrenamiento se lleva a cabo desde cero utilizando las representaciones RGB de 10 imágenes multi-espectrales (generadas en la Sección 5.3), junto con las máscaras correspondientes y el aumento de datos. En este caso, hemos creado manualmente las máscaras. De las 10 imágenes multi-espectrales, se consideraron 9 para el entrenamiento, aumentándose en 1000 ocasiones (obteniéndose un total de 9000 imágenes); y una imagen multi-espectral para validación y ajuste, también aumentada en 1000 ocasiones.
- **I_Jes_Multi:** Es una extensión del conjunto de datos I_Jes que agrega las 10000 imágenes del conjunto de datos Multi. El número total de imágenes es 156000. El porcentaje de estas imágenes para el entrenamiento y para validación fue del 90% y 10%, respectivamente.

5.4.3 Evaluación del proceso de clasificación

Para cuantificar la precisión del algoritmo de clasificación propuesto y otros enfoques de clasificación basados en DL, se ha llevado a cabo una comparación entre las predicciones para cada objeto segmentado y la información de referencia correspondiente. Para obtener la referencia, se han etiquetado manualmente todas las imágenes utilizadas para la evaluación. El rendimiento

²<https://www.image-net.org/>

³<http://www.diag.uniroma1.it/labrococo/fsd/sunflowerdatasets.html>

del modelo se ha medido en términos de Intersección sobre Unión (IoU, por sus siglas en inglés), que se describe como:

$$IoU = \frac{Predicción \cap Referencia}{Predicción \cup Referencia} = \frac{VP}{FP + VP + FN}$$

donde VP, FP y FN definen el número de muestras de verdaderos positivos, falsos positivos y falsos negativos, respectivamente. El IoU se mide para cada clase (girasol, maleza y suelo) del conjunto de datos, lo que da lugar al IoU promedio (mIoU) como el IoU promediado sobre cada clase. Los resultados se discuten en la Sección 5.5.2 de este capítulo.

5.5 Resultados del caso práctico de agricultura de precisión

En esta sección se detallan los resultados obtenidos, tanto del proceso de calibración de las diferentes lentes, como de la clasificación final resultante de todo el trabajo desarrollado.

5.5.1 Discusión de la calibración de la lente

La Tabla 5.1 muestra los resultados del proceso de calibración utilizando varias imágenes de damero (imágenes de muestra) por sensor. Esta tabla señala el número de imágenes para calibración y prueba, así como el error promedio (en píxeles) después de finalizar la prueba de calibración. Como se puede observar en esta tabla, se ha obtenido un error promedio de menos de 0.3 píxeles en todas las bandas de interés. La banda NIR es la que proporciona la mejor calidad en la cámara, ya que la mayoría de los dameros en las imágenes fueron reconocibles al haber un mejor contraste.

Tabla 5.1: Número de imágenes utilizadas para cada sensor en el proceso de calibración. La columna “Total de imágenes” identifica el número total de imágenes consideradas, mientras que “Imágenes para conjunto de calibración” e “Imágenes para conjunto de prueba” muestran cuántas de ellas se utilizaron para el conjunto de calibración y el conjunto de prueba, respectivamente. La columna “Media de c_{ij} (error en píxeles)” muestra el error promedio en píxeles después del proceso de calibración considerando las imágenes del conjunto de prueba.

Sensor	Total de imágenes	Imágenes para conjunto de calibración	Imágenes para conjunto de prueba	Media de c_{ij} (error en píxeles)
GRE	85	69	16	0.27
NIR	75	61	14	0.23
RED	95	78	17	0.29
REG	73	56	17	0.28

Con un error medio de 0.3 píxeles se puede considerar que se ha realizado una correcta calibración de todas las lentes, y se pueden utilizar los parámetros intrínsecos calculados para realizar la corrección de la distorsión geométrica con garantías.

5.5.2 Discusión del proceso de clasificación

Se han implementado tres enfoques diferentes para clasificar el suelo, el girasol y la maleza: una clasificación basada en técnicas de visión artificial con objetos (ClasOb) y dos clasificaciones basadas en dos modelos del estado del arte de DL (U-Net y FPN).

En lo que respecta a U-Net y FPN, se ha realizado un estudio preliminar para determinar los mejores modelos y conjuntos de datos para la clasificación del cultivo y la maleza. En este estudio, que se muestra en la Tabla 5.2, los modelos U-Net y FPN se han entrenado utilizando los tres conjuntos de datos detallados en la Sección 5.4.2, resultando en un total de seis redes neuronales (U-Net I_Jes, U-Net I_Jes_Multi, U-Net Multi, FPN I_Jes, FPN I_Jes_Multi y FPN Multi). Estas redes se han validado utilizando un conjunto de datos de prueba con 16 representaciones RGB de imágenes multi-espectrales (generadas como se explica en la Sección 5.3). El tamaño del lote (batch size) fue de 16. En la Tabla 5.2, la columna “Ciclos completos para el mejor IoU” identifica el número de ciclos completos (epochs) que la red neuronal ha necesitado para obtener el mejor valor de la función de pérdida en el conjunto de datos de validación. Es importante destacar que en este problema, las funciones de pérdida consideradas han sido Dice Loss y Focal Loss. Ambas funciones han demostrado mejores resultados que la función de pérdida de Jaccard para detectar las clases con menor representación (malas hierbas) en las imágenes del conjunto de datos. Además, la columna “Ciclos completos totales” identifica el número total de ciclos completos de cada experimento considerando el criterio de detención temprana con un valor de 30. Finalmente, en la misma tabla se muestran los valores de IoU promedio para el girasol y las malas hierbas, respectivamente.

Tabla 5.2: Estudio preliminar de las redes U-Net y FPN entrenadas en tres conjuntos de datos diferentes, y validadas con 16 representaciones RGB obtenidas de la Sección 5.3. Los mejores valores en términos de IoU promedio para cultivos (Girasol) y maleza están marcados en negrita.

Modelo	Conjunto de datos	Ciclos completos	Total ciclos	IoU promedio	IoU promedio
		para el mejor IoU	completos	Girasol	Maleza
U-Net	I_Jes	140	200	0.5	0.05
	I_Jes_Multi	314	374	0.68	0.90
	Multi	22	52	0.70	0.90
FPN	I_Jes	76	136	0.05	0.68
	I_Jes_Multi	376	436	0.70	0.90
	Multi	40	70	0.69	0.90

A partir de la Tabla 5.2, se pueden extraer las siguientes conclusiones: (1) utilizar un conjunto de datos de imágenes de un problema de clasificación de plantas y maleza (I_Jes) presentado en [26], no muestra un rendimiento suficiente para abordar el problema estudiado de clasificación de malas hierbas (valores de IoU promedio en maleza y girasol bajos); (2) utilizar imágenes ad-hoc de nuestro problema para entrenar la red neuronal (I_Jes_Multi y Multi) resulta en una gran mejora en la calidad, a pesar de utilizar un conjunto de datos de entrenamiento más pequeño; (3) el uso del conjunto de datos I_Jes_Multi aumenta considerablemente el número de ciclos completos necesarios para obtener el mejor valor de IoU. Esta última opción es ligeramente mejor para el modelo FPN y ligeramente peor para el modelo U-Net. Por otro lado, también merece la pena destacar de la tabla que aunque los resultados de los modelos U-Net y FPN utilizando I_Jes no obtuvieron un rendimiento suficiente para nuestro problema, se han validado

en el problema original de los autores del conjunto de datos, obteniendo resultados similares a nuestros mejores resultados de la Tabla 5.2. Finalmente, mencionar que en la Tabla 5.2 los modelos U-Net_Multi y FPN_I_Jes_Multi obtuvieron el mejor IoU promedio para clasificar cultivos y maleza, por lo que han sido los considerados para ser comparados con el enfoque que se ha desarrollado en la tesis, ClasOb.

La Tabla 5.3 muestra los valores de IoU para girasol, maleza y materia vegetal (girasol \cup maleza) utilizando la metodología propuesta en todo este capítulo para las tres aproximaciones estudiadas (propuesta ClasOb, U-Net_Multi y FPN_I_Jes_Multi) y 16 imágenes de muestra multi-espectrales. En esta tabla también se indica el porcentaje de materia vegetal, girasol y maleza de cada imagen de muestra. A partir de la tabla se puede observar que: (1) el IoU de girasol es ligeramente mejor utilizando DL que nuestra propuesta; (2) el IoU de maleza es ligeramente mejor en el enfoque basado en la clasificación ClasOb; y (3) en el caso de la materia vegetal, nuestra propuesta mejora en mayor medida la segmentación basada en DL.

La Tabla 5.4 muestra, en términos de IoU promedio, el rendimiento para clasificar materia vegetal, girasol y maleza utilizando la versión de clasificación propuesta (ClasOb) y los dos enfoques de DL. Se han considerado las mismas 16 imágenes de muestra de la Tabla 5.3 para el cálculo del promedio. Como se puede observar en la Tabla 5.4, el clasificador ClasOb es capaz de clasificar materia vegetal con una precisión del 91%, la planta de interés con una precisión del 89%, y maleza con una precisión del 72%. Si comparamos estos valores con respecto a los modelos basados en DL estudiados, nuestro modelo es el mejor para clasificar materia vegetal en general y maleza en particular. El enfoque U-Net_I_Multi es el mejor para clasificar girasol, aunque nuestro enfoque es bastante competitivo, ya que obtiene un IoU promedio muy cercano. Por lo tanto, podemos concluir que cuando no se dispone de un alto número de imágenes del campo de estudio y sus máscaras de segmentación correspondientes, el clasificador ClasOb se ha propuesto en esta tesis es muy adecuado.

Adicionalmente, se ha realizado otro estudio comparativo entre los resultados de clasificación (en términos de mIoU) obtenidos en este capítulo y los obtenidos en el artículo [26]. Como se comentó anteriormente, en ese artículo los autores consideraron varios enfoques de DL (incorporando una modificación de los métodos comunes de aumento de datos) en un cultivo de girasol utilizando imágenes multi-espectrales. El conjunto de datos que se ha considerado en [26] para entrenar y probar los modelos ha sido una extensión del conjunto de datos I_Jes (utilizando girasoles en diferentes etapas de crecimiento). El mejor valor de mIoU obtenido por los autores para la clasificación maleza fue de 0.69 utilizando la arquitectura Bonnet, frente al 0.72 que se ha obtenido en nuestra propuesta (ClasOb). Por lo tanto, podemos establecer que la metodología que hemos propuesto en la presente tesis es ligeramente mejor para la clasificación de maleza. Además, otra ventaja es que nuestro enfoque no requiere la generación de conjuntos de datos ni el proceso de entrenamiento del modelo, que son tareas de gran coste en tiempo y recursos.

Como ejemplo, la Figura 5.14 muestra la representación RGB de tres bandas (RED, NIR y GRE) para las muestras 34, 25 y 15, sus imágenes segmentadas correspondientes después de

Tabla 5.3: Resultados de IoU obtenidos para 16 imágenes de muestra multi-espectrales utilizando tres enfoques de clasificación: la que se ha propuesto en este capítulo (denominada ClasOb) y dos modelos de DL (U-Net_Multi y FPN_I_Jes_Multi). Además, se muestra el porcentaje de áreas de materia vegetal, girasol y maleza. Los mejores valores de IoU han sido resaltados en negrita.

Imagen de muestra	Materia Vegetal (%)	Girasol (%)	Maleza (%)	Método	IoU Materia Vegetal	IoU Girasol	IoU Maleza
1	27.16	25.97	1.19	Propuesta ClasOb	0.99	0.99	0.74
				U-Net_Multi	0.96	0.96	0.74
				FPN_I_Jes_Multi	0.96	0.97	0.73
2	19.13	16.63	2.50	Propuesta ClasOb	0.97	0.99	0.67
				U-Net_Multi	0.94	0.96	0.70
				FPN_I_Jes_Multi	0.93	0.96	0.68
4	14.59	8.69	5.89	Propuesta ClasOb	0.86	0.88	0.66
				U-Net_Multi	0.71	0.79	0.51
				FPN_I_Jes_Multi	0.69	0.45	0.35
10	43.09	33.76	9.33	Propuesta ClasOb	0.97	0.95	0.87
				U-Net_Multi	0.94	0.94	0.80
				FPN_I_Jes_Multi	0.94	0.95	0.81
12	42.18	22.72	19.46	Propuesta ClasOb	0.89	0.81	0.73
				U-Net_Multi	0.85	0.86	0.71
				FPN_I_Jes_Multi	0.87	0.91	0.74
13	20.61	12.63	7.97	Propuesta ClasOb	0.82	0.84	0.53
				U-Net_Multi	0.75	0.89	0.53
				FPN_I_Jes_Multi	0.75	0.90	0.54
14	21.97	7.58	14.39	Propuesta ClasOb	0.89	0.89	0.81
				U-Net_Multi	0.81	0.95	0.78
				FPN_I_Jes_Multi	0.80	0.93	0.76
15	35.70	23.11	12.59	Propuesta ClasOb	0.90	0.91	0.70
				U-Net_Multi	0.82	0.93	0.63
				FPN_I_Jes_Multi	0.83	0.93	0.60
16	23.26	20.38	2.88	Propuesta ClasOb	0.96	0.95	0.65
				U-Net_Multi	0.95	0.96	0.76
				FPN_I_Jes_Multi	0.95	0.96	0.77
17	24.10	21.35	2.75	Propuesta ClasOb	0.93	0.92	0.55
				U-Net_Multi	0.93	0.95	0.81
				FPN_I_Jes_Multi	0.93	0.94	0.78
20	34.14	20.94	13.20	Propuesta ClasOb	0.90	0.88	0.71
				U-Net_Multi	0.86	0.87	0.67
				FPN_I_Jes_Multi	0.88	0.87	0.66
22	38.26	25.61	12.66	Propuesta ClasOb	0.90	0.91	0.79
				U-Net_Multi	0.86	0.92	0.72
				FPN_I_Jes_Multi	0.87	0.93	0.75
24	28.54	16.22	12.32	Propuesta ClasOb	0.92	0.88	0.78
				U-Net_Multi	0.84	0.92	0.74
				FPN_I_Jes_Multi	0.87	0.93	0.75
25	25.18	9.43	15.76	Propuesta ClasOb	0.90	0.90	0.83
				U-Net_Multi	0.84	0.83	0.72
				FPN_I_Jes_Multi	0.84	0.89	0.75
26	36.28	19.42	16.87	Propuesto	0.79	0.58	0.65
				U-Net_Multi	0.82	0.71	0.70
				FPN_I_Jes_Multi	0.84	0.83	0.76
34	26.60	23.21	3.38	Propuesta ClasOb	0.96	0.96	0.77
				U-Net_Multi	0.93	0.95	0.71
				FPN_I_Jes_Multi	0.93	0.95	0.76

Tabla 5.4: IoU promedio para la estrategia propuesta ClasOb y dos enfoques de DL, para cada clase en la clasificación de 16 imágenes de muestra. Los mejores valores de IoU promedio se han resaltado en negrita.

Clase	mIoU Propuesto	mIoU U-Net_I_Multi	mIoU FPN_I_Jes_Multi
Materia Vegetal	0.91	0.86	0.87
Girasol	0.89	0.90	0.89
Maleza	0.72	0.70	0.70

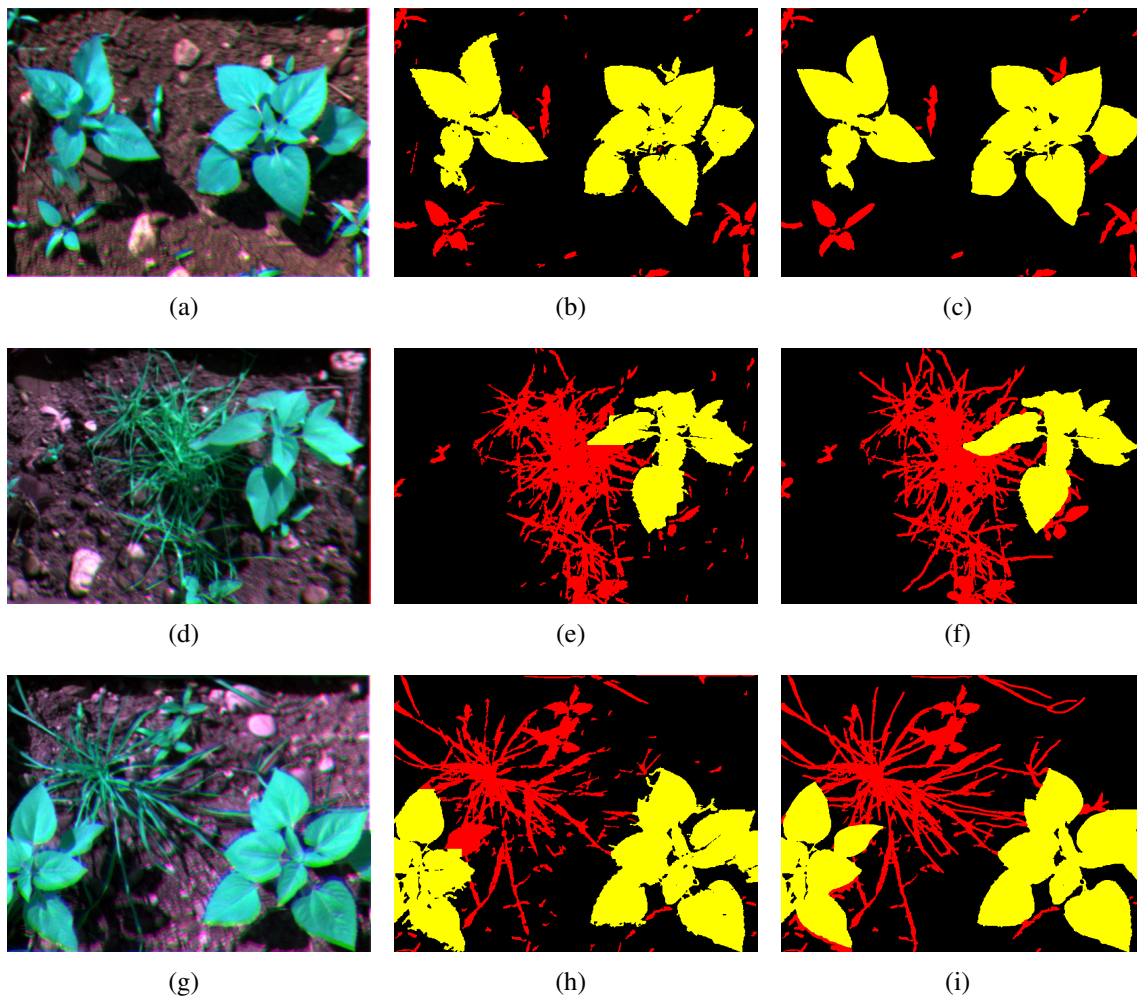


Figura 5.14: Tres imágenes de muestra (34, 25 y 15) segmentadas por el algoritmo clásico propuesto (una por fila). (a), (d) y (g) muestran una representación RGB de la imagen multi-espectral utilizando las bandas RED, NIR y GRE; (b), (e) y (h) muestran el resultado del algoritmo; y (c), (f) y (i) muestran la verdad terreno (ground truth) determinada por humanos.

aplicar el algoritmo clásico propuesto; y, finalmente, su verdad terreno (ground truth) como referencia.

5.6 Conclusiones y trabajo futuro del caso de estudio de agricultura de precisión

Este capítulo ha descrito una metodología completa para cuantificar automáticamente el impacto de un posible herbicida en la evolución de un cultivo de girasol y el crecimiento de malezas mediante imágenes multi-espectrales y el uso de segmentación basada en objetos.

Se ha realizado un preprocesado para minimizar el impacto de la distorsión (a través de la calibración de la lente) y el error provocado por la perspectiva resultante de la distancia entre las diferentes lentes (a través de la rectificación y co-registro) de la cámara multi-espectral Sequoia. Se han considerado las siguientes bandas: NIR, GRE, RED y REG, y el cultivo estudiado fue un campo de girasoles.

Las imágenes multi-espectrales obtenidas después de los procesos de calibración de la lente, rectificación y co-registro se han utilizado para la clasificación. Para este propósito, se ha llevado a cabo un nuevo algoritmo de clasificación basado en objetos mediante el software eCognition que utiliza información multi-espectral y geométrica (forma y tamaño de la planta de interés y la maleza). A este algoritmo lo hemos denominado ClasOb. El uso de información geométrica fue necesario porque la firma espectral de las plantas era altamente variable e insuficiente para diferenciar maleza y cultivo.

El algoritmo propuesto, ClasOb, ha clasificado todos los píxeles de las imágenes en varias clases: suelo, girasol y maleza. Dado que el suelo se obtiene por eliminación, no es un buen indicativo del rendimiento. Sin embargo, el IoU de la materia vegetal (girasol \cup maleza) es una métrica interesante para probar la precisión del algoritmo para la segmentación genérica de las plantas. Para cuantificar la precisión del algoritmo propuesto, se ha considerado la métrica de mIoU, obteniendo valores de 0.91, 0.89 y 0.72 para la clasificación de materia vegetal, girasol y maleza, respectivamente.

Nuestro enfoque ha sido comparado con dos métodos de segmentación/clasificación basados en DL que hemos entrenado con varios conjuntos de datos. Hemos realizado un estudio preliminar para elegir los enfoques de DL más competitivos (U-Net_Multi y FPN_I_Jes_Multi), susceptibles de ser comparados con los resultados de nuestra propuesta. El estudio preliminar ha demostrado que cuando los datos de entrenamiento son escasos, es mejor aumentar el conjunto de datos de nuestro problema específico con técnicas de aumento de datos que incluir un conjunto de datos de un problema relacionado. U-Net_Multi y FPN_I_Jes_Multi han sido las mejores redes neuronales obtenidas para este problema y, por tanto, han sido comparadas con nuestra propuesta. La estrategia propuesta (ClasOb) ha demostrado su superioridad frente a los enfoques basados en DL en la clasificación de materia vegetal y maleza. Otro aspecto positivo de estrategia es que no necesita la generación de conjuntos de datos más extensos, ni el proceso de entrenamiento del modelo, que son tareas que consumen tiempo y recursos. Por lo tanto, concluimos que en campos de estudio con pocas imágenes disponibles (como el estudiado en el primer problema de esta tesis), es más eficiente y se obtienen mejores resultados en la estrategia que hemos propuesto, denominada ClasOb, que en los modelos de segmentación/clasificación basados en DL estudiados.

Todos los algoritmos desarrollados, imágenes en bruto, mapas de verdad de terreno e imágenes obtenidas de la clasificación los hemos puesto disponibles públicamente en la siguiente página web <https://github.com/Heikelol/SegmentationForPlants>.

Como trabajo futuro, se pretende extender la metodología propuesta a otros cultivos diferentes del girasol. Además, se plantea la ampliación del estudio basado en DL para este problema particular. En concreto, se pretenden investigar las redes neuronales convolucionales basadas en objetos disponibles en la plataforma eCognition Developer.

6. Caso de estudio de procesamiento cuántico de imagen

Las técnicas cuánticas de procesamiento de imágenes son relevantes porque ofrecen un enfoque prometedor a la hora de resolver problemas complejos de procesamiento de imágenes. En algunos casos, el procesamiento cuántico permite procesar las imágenes de una forma más eficiente que los métodos clásicos. Por ello, se ha convertido en un tema de gran interés. Una de las técnicas más utilizadas en el procesamiento de imágenes es la segmentación, que en una de sus formas más básicas puede llevarse a cabo utilizando algoritmos de umbral. En esta tesis, tal y como se comentó en la Sección 1.2, uno de los casos de estudio está relacionado con un algoritmo de segmentación de doble umbral sobre plataformas cuánticas. Dicho algoritmo se ha diseñado utilizando puertas del grupo Clifford+T para que sea compatible con los códigos de detección y corrección de errores.

Este capítulo se ha estructurado de la siguiente forma. En la Sección 6.1 se realiza una descripción del caso de estudio que se ha abordado. La Sección 6.2 describe matemáticamente el modelo NEQR para representaciones de imágenes cuánticas basadas en niveles de gris. En la Sección 6.3 se indican las puertas cuánticas optimizadas en número de puertas T que se han utilizado para el diseño de los circuitos implementados. La Sección 6.4 describe el diseño de dos comparadores cuánticos eficientes que formarían parte del algoritmo de segmentación de doble umbral. Además, se ilustra la implementación en la plataforma IBM Quantum de la operación reset, utilizada en el estado del arte de circuitos cuánticos. La Sección 6.5 describe la implementación del algoritmo de segmentación de imágenes con doble umbral que se ha propuesto. En la Sección 6.6 se compara tanto el algoritmo de segmentación de doble umbral como los circuitos comparadores diseñados con sus homólogos del estado del arte. Por último, en la Sección 6.7, se encuentran las principales conclusiones e ideas de trabajos futuros.

6.1 Segmentación de doble umbral con computación cuántica

El segundo caso de estudio de la tesis consiste en un problema basado en procesamiento cuántico de imágenes. Por lo general, el primer paso para trabajar con algoritmos cuánticos sobre imágenes es codificar dichas imágenes en estados cuánticos [110]. Una vez que las imágenes están codificadas, se pueden aplicar varias transformaciones a las mismas. Una de las transformaciones más utilizadas es la segmentación de imágenes (descrita en la Sección 4), que se puede llevar a cabo mediante diversos métodos, como el cálculo de histogramas [13, 14], los algoritmos de búsqueda cuántica [59] o la segmentación de doble umbral [114, 127].

El cálculo de histogramas y los métodos de búsqueda cuántica utilizan la amplificación de la amplitud y pueden lograr una mejora significativa en la velocidad. Sin embargo, estos algoritmos carecen de una implementación real del circuito del oráculo o utilizan un número de cúbits tan alto que hacen que la simulación no sea viable [13, 14, 82, 127].

Por otro lado, los dos algoritmos del estado del arte que utilizan circuitos cuánticos para calcular la segmentación de imágenes con doble umbral son los propuestos por Yuan et al. [127] y Wang et al. [114]. El primero se basa en comparadores completos y el segundo en medios comparadores (solo determinan si $A > B$ o $A < B$, pero no si $A = B$, por lo que son más limitados). Merece la pena destacar que ninguno de los dos algoritmos es tolerante a fallos.

El algoritmo de segmentación de doble umbral que se ha tomado como punto de partida es el propuesto por Yuan et al. [127]. Este algoritmo establece todos los píxeles en una imagen como 0, excepto aquellos que se encuentren entre los dos umbrales dados. Supongamos que la imagen original es $f(x, y)$, $T1$ es el umbral inferior, $T2$ es el umbral superior y $g(x, y)$ es la imagen de salida. De este modo, esta segmentación se puede describir matemáticamente como se muestra a continuación:

$$\begin{cases} g(x, y) = f(x, y) & \text{si } T1 \leq f(x, y) \leq T2 \\ g(x, y) = 0 & \text{en cualquier otro caso} \end{cases} \quad (6.1)$$

En general, los circuitos cuánticos son muy sensibles al ruido externo e interno. El conjunto de puertas Clifford+T permite diseñar circuitos tolerantes a fallos gracias al uso de códigos de detección y corrección de errores [20, 81]. Debido a que las implementaciones tolerantes a fallos implican puertas T y el coste de estas puertas es considerablemente más elevado en términos de espacio y tiempo que el de otras puertas [72, 77, 129], la optimización del número de puertas T es un objetivo de diseño a la hora de implementar circuitos cuánticos robustos. Por lo tanto, como fruto de esta tesis se ha diseñado un circuito cuántico que implementa la Ecuación 6.1 solo con puertas Clifford+T. Además, como se puede deducir de la misma ecuación, el algoritmo de doble umbral implica dos operaciones de comparación.

En esta tesis se han propuesto dos alternativas de circuitos cuánticos comparadores tolerantes a fallos y que optimizan el número de puertas T para diferentes casos de uso. Cada circuito tiene valores de T-count, T-depth y número de cúbits auxiliares diferentes, y se podrán utilizar en problemas de un tamaño de imagen específico.

6.2 Codificación de imágenes utilizando la representación NEQR

El primer paso para trabajar con algoritmos cuánticos sobre imágenes es codificar dichas imágenes en estados cuánticos. Los modelos de representación cuántica de imágenes se pueden clasificar en varias categorías basadas en los requisitos definidos para almacenar el contenido de una imagen en un sistema cuántico, es decir, el modelo de color de la imagen, el modelo de coordenadas de la imagen y el modelo de codificación de información de color de la imagen [122]. Es importante destacar que aunque se hable de color, muy pocas codificaciones son capaces de trabajar con imágenes RGB, y el nivel de color se trata habitualmente de un nivel de intensidad dentro de la escala de grises.. Los dos métodos más comunes para codificar imágenes de niveles de gris en estados cuánticos son la Representación Flexible de Imágenes Cuánticas (Flexible Representation of Quantum Images, FRQI) [82] y la Nueva Representación Cuántica Mejorada (Novel Enhanced Quantum Representation, NEQR) [130].

En FRQI, la información de color (niveles de gris) se codifica en la amplitud de probabilidad de un estado de cúbit [82]. Por el contrario, en NEQR, la información de color se almacena en la base computacional de múltiples cúbits del estado de superposición, pero no en la amplitud de probabilidad [127]. Este último método es el considerado en este capítulo, ya que permite transformaciones avanzadas de imágenes. Por tanto, esta sección se dedica a definir matemáticamente esta codificación y a mostrar sus bondades frente a otras aproximaciones.

NEQR es una de las primeras formas de representación cuántica de imágenes y se utiliza ampliamente en QIMP [130]. El modelo de color para NEQR se basa en escala de grises y su modelo de coordenadas es cartesiano. En cuanto a la codificación de la información de color de la imagen, este modelo utiliza el estado base de una secuencia de cúbits para almacenar el valor en escala de grises de cada píxel.

NEQR ofrece cuatro ventajas sobre otros modelos de representación temprana como FRQI: (1) existe una aceleración cuadrática de la complejidad temporal para preparar la imagen cuántica NEQR (una complejidad temporal de $O(2^{4n})$ en FRQI frente a $O(qn2^{2n})$ para NEQR, donde n es el logaritmo en base 2 del tamaño de una imagen de tamaño $2^n \times 2^n$ y q representa el rango de colores como $[0, 2^{q-1}]$, aunque existen algunas variantes de FRQI, como IFRQI, que abordan esto reduciendo su complejidad temporal), (2) una relación de compresión de imagen optimizada de hasta 1,5 veces, (3) su recuperación de imagen es precisa (no probabilística como en FRQI) y (4) se pueden lograr operaciones avanzadas de color y segmentación [130].

En la codificación NEQR, una imagen de tamaño $2^n \times 2^n$ se representa mediante la siguiente ecuación:

$$|C_{YX}\rangle = \frac{1}{2^n} \sum_{Y=0}^{2^n-1} \sum_{X=0}^{2^n-1} |C^{q-1}YXC^{q-2}YX\dots C^1YXC^0YX\rangle \otimes |YX\rangle \quad (6.2)$$

donde $|C^{q-1}YXC^{q-2}YX\dots C^1YXC^0YX\rangle$ codifica el valor de color del píxel (Y, X) , n está relacionado con el tamaño de la imagen y q define el rango de colores como $[0, 2^{q-1}]$. YX codifica la ubicación espacial del píxel en un modelo de coordenadas cartesianas.

NEQR requiere $2n + q + 2n - 2$ cúbits para preparar una imagen cuántica de tamaño $2^n \times 2^n$ píxeles y un rango de grises de $[0, 2^{q-1}]$, donde se utilizan $2n$ cúbits para la información de posición, q cúbits para la información de escala de grises y $2n - 2$ son cúbits auxiliares [130]. Yuan et al. [127] propusieron una estrategia para reducir el número requerido de cúbits a $2n + q + 2$ mediante la reutilización de cúbits con la operación de reinicio, que permite restablecer el cúbit al estado $|0\rangle$ en medio de la computación [102]. Ambos enfoques son válidos para obtener una imagen codificada NEQR y realizar la operación de doble umbral cuántico, pero el último tiene la desventaja de utilizar la operación de reinicio. Esta operación puede no ser óptima en algunos casos, ya que no está disponible en todos los dispositivos cuánticos e implica puertas no-Clifford y, por tanto, no tolerantes a fallos.

6.3 Optimización de puertas cuánticas en términos de la métrica T

En esta sección se detalla la implementación considerada de algunas puertas para que pertenezcan al conjunto Clifford+T y así sean compatibles con códigos de detección y corrección de errores.

Los circuitos cuánticos escalables y robustos se basan en implementaciones tolerantes a fallos de puertas cuánticas como las puertas Clifford+T [72]. Esto se debe a que las puertas cuánticas son altamente sensibles al ruido, tanto interno como externo [80, 85, 115, 134]. Sin embargo, las implementaciones tolerantes a fallos de puertas cuánticas, así como los códigos de corrección de errores cuánticos pueden utilizarse para superar los límites impuestos por los errores de ruido [4, 134].

El algoritmo propuesto se ha construido solo con puertas del grupo Clifford+T. Los símbolos y las matrices asociadas a estas puertas cuánticas se pueden observar en la Figura 6.1.

Las puertas que se han seleccionado para el diseño de los circuitos se han elegido teniendo en cuenta las mejores estrategias de optimización de las métricas T-count y T-depth (descritas en la Sección 2.2.2). Por ejemplo, para implementar una operación AND, los circuitos propuestos utilizan la puerta lógica-AND temporal [33] para reducir el número de puertas T involucradas. Esta puerta realiza una operación AND de dos entradas (cúbits), guardando el resultado en un cúbit auxiliar. La puerta lógica-AND temporal y su puerta de descomputación (uncomputation) se mostraron en la Figura 2.14 de la Sección 2.2.2. Tanto la puerta lógica-AND temporal como la puerta Toffoli son similares, pero la puerta lógica-AND temporal tiene un mejor T-count (4 frente a 7 para la puerta Toffoli) y un mejor T-depth (2 frente a 3 para la puerta Toffoli). La descomputación de la puerta lógica-AND temporal agrega otro nivel de eficiencia en costes T, ya que no requiere puertas T a diferencia de la descomputación de la puerta Toffoli, que requiere otra puerta Toffoli.

Para reemplazar una puerta Toffoli por una operación lógica-AND temporal, la operación AND debe realizarse en un cúbit auxiliar y no en estados cuánticos ya inicializados. Por lo tanto, no todas las puertas Toffoli pueden ser reemplazadas directamente por una puerta lógica-AND temporal, y estos cambios requerirán algunos cúbits auxiliares adicionales para reducir el T-count.

Clifford+T Gate Set







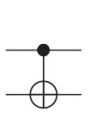
NOT Gate		$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Hadamard Gate		$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
T Gate		$\begin{bmatrix} 1 & 0 \\ 0 & e^{i \cdot \frac{\pi}{4}} \end{bmatrix}$
Hermitian of T Gate		$\begin{bmatrix} 1 & 0 \\ 0 & e^{-i \cdot \frac{\pi}{4}} \end{bmatrix}$
Phase Gate		$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$
Hermitian of Phase Gate		$\begin{bmatrix} 1 & 0 \\ 0 & -i \end{bmatrix}$
Feynman(CNOT)Gate		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

Figura 6.1: Símbolos y matrices del conjunto de puertas Clifford+T utilizadas en el algoritmo de segmentación de imágenes de doble umbral propuesto.

6.4 Implementaciones tolerantes a fallos de circuitos comparadores

En esta sección se describe el estado del arte de los circuitos comparadores involucrados en el desarrollo de los algoritmos de segmentación por doble umbral que estamos considerando. Además, se describen en detalle las dos implementaciones de circuitos cuánticos comparadores que hemos propuesto en esta tesis.

6.4.1 Estado del arte de los circuitos comparadores para la segmentación de doble umbral

Como se indicó en la Sección 6.1 la operación fundamental del algoritmo que hemos tomado como punto de partida [127] son los comparadores. Por lo tanto, como parte de su algoritmo los autores presentaron un comparador optimizado en términos de cúbits auxiliares [127]. Este comparador requiere $5N - 3$ puertas Toffoli (siendo N el tamaño de los números que se van a comparar, es decir, el número de bits), por lo que tiene un T-count de $35N - 21$. Este comparador utiliza la operación de reinicio (reset) presentada por Shende et al. en [102] para reducir la cantidad requerida de cúbits en el circuito. Gracias a esta operación de reinicio, los mismos cúbits se pueden reutilizar en diferentes partes del circuito. Sin embargo, las puertas cuánticas que realizan la operación de reinicio ni pertenecen al grupo Clifford+T ni son reversibles. Además, esta operación no es trivial de realizar, ya que requiere personalizar los ángulos para cada estado cuántico específico que se va a reiniciar, algo que los dispositivos cuánticos disponibles

actualmente no pueden hacer sin la intervención directa del programador. En la Sección 6.4.3 se realiza la descripción del diseño de la operación de reinicio en la plataforma IBM Quantum.

El comparador propuesto por Yuan et al. [127] se muestra en la Figura 6.2. Para evitar las $2N$ operaciones de reinicio, la descomputación de los cúbits auxiliares debe realizarse utilizando puertas Clifford+T. Dado que todo el circuito, excepto las operaciones de reinicio, está compuesto exclusivamente por puertas Clifford+T, se puede utilizar el esquema de eliminación de basura de Bennett para la descomputación [8]. Siguiendo este esquema, se obtiene un nuevo circuito (Figura 6.3), que está compuesto exclusivamente por puertas Clifford+T y, por lo tanto, es compatible con los códigos de detección y corrección de errores asociados a dichas puertas. Sin embargo, el precio a pagar es alto: la profundidad del circuito y el número de cúbits auxiliares se han duplicado. De manera aún más desfavorable, su T-count también se ha duplicado: de $35N - 21$ a $70N - 42$.

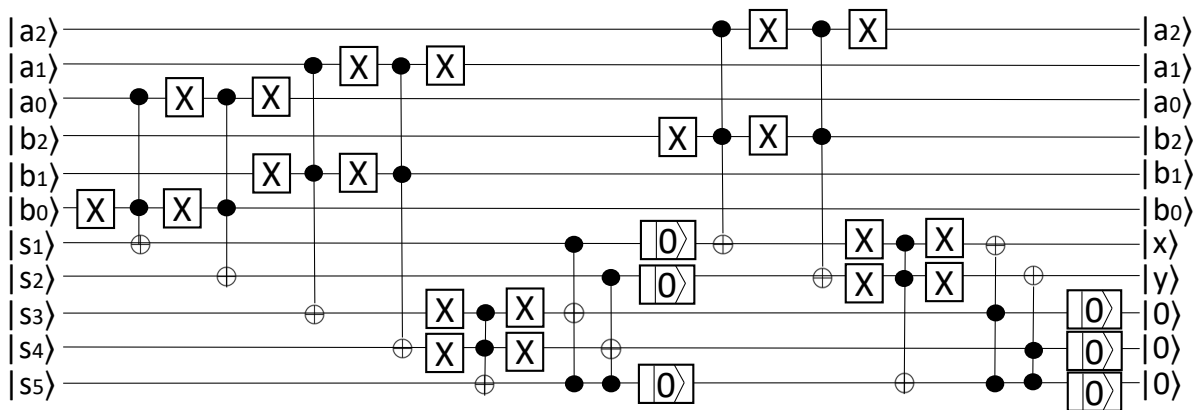


Figura 6.2: Circuito comparador propuesto por Yuan et al. [127], para el caso $N = 3$. Está optimizado en términos de cúbits auxiliares. Sin embargo, implica el uso de puertas no Clifford+T para realizar la operación de reinicio.

6.4.2 Circuitos comparadores propuestos para la segmentación de doble umbral

Como se mencionó anteriormente, el circuito mostrado en la Figura 6.3 tiene un número elevado de puertas T, pero se pueden proponer varias mejoras para implementar tolerancia a fallos y reducir las métricas de T-count y T-depth. El número de puertas T se puede reducir fácilmente reemplazando algunas puertas Toffoli por puertas AND lógicas temporales. La mayoría de las puertas Toffoli en el circuito de Yuan et al. (y en la versión de Clifford+T obtenida a partir de él) actúan sobre un cúbit auxiliar y no sobre estados cuánticos ya inicializados, por lo que se pueden reemplazar directamente por operaciones AND lógicas temporales. En particular, se pueden sustituir $3N - 1$ puertas Toffoli, mientras que $2N - 2$ puertas Toffoli actúan sobre estados existentes y no se pueden reemplazar (al menos no directamente). Considerando la parte de descomputación, se necesitarán $3N - 1$ puertas de descomputación de la operación AND lógica temporal y $2N - 2$ puertas Toffoli adicionales. El número total de puertas T de este circuito es $40N - 32$. Un ejemplo de esta nueva versión, para el caso $N = 3$, se muestra en la Figura 6.4.

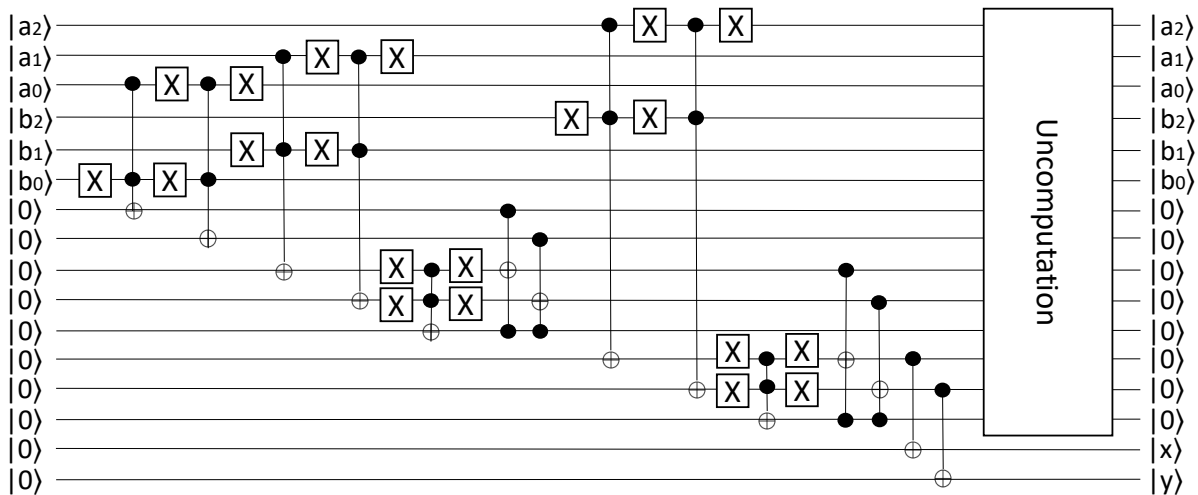


Figura 6.3: Circuito comparador obtenido al reemplazar las operaciones de reinicio en el circuito de Yuan et al. [127]. La parte de descomputación consiste en aplicar el mismo circuito, pero en orden inverso. Este circuito solo contiene puertas Clifford+T, por lo que es tolerante a fallos. Sin embargo, su T-depth se duplica y tiene el doble de cúbits auxiliares que el circuito original.

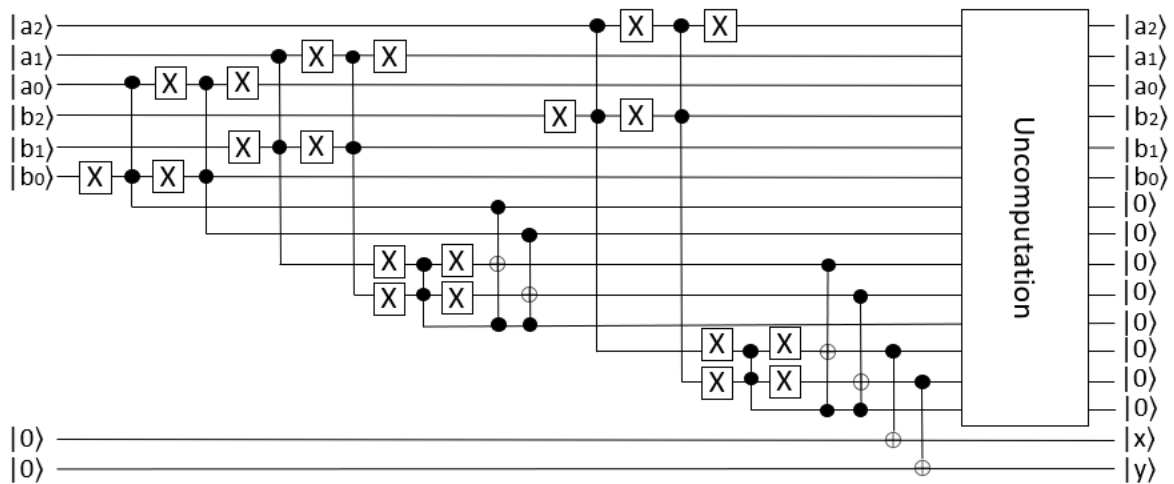


Figura 6.4: Primera propuesta de circuito comparador cuántico centrado en reducir el número de puertas T, para el caso $N = 3$. a_i y b_i son los cúbits de los dos números A y B que se van a comparar. Las salidas marcadas como x e y son las salidas de la comparación.

El número de puertas T del circuito obtenido es alto debido a las $2N - 2$ puertas Toffoli restantes. Como se mencionó, dichas puertas no se pueden reemplazar directamente ya que actúan sobre cúbits no auxiliares. Sin embargo, si el objetivo es reducir al máximo el número de puertas T, estas puertas se pueden reemplazar a costa de aumentar el número de cúbits auxiliares.

Por tanto, una puerta Toffoli que involucra tres cúbits $|A\rangle$, $|B\rangle$ (cúbits de control) y $|C\rangle$ (cúbit de objetivo) se puede reemplazar por una puerta AND lógica temporal y una puerta CNOT. La puerta temporal utilizará $|A\rangle$ y $|B\rangle$ como cúbits de control, y un cúbit auxiliar como cúbit de objetivo, mientras que la puerta CNOT utilizará el cúbit auxiliar anterior como cúbit de control, y $|C\rangle$ como cúbit de objetivo. Si se reorganizan las operaciones de manera que se mantenga el

resultado original, se pueden reemplazar las puertas Toffoli restantes. Por ejemplo, una puerta Toffoli aplicada a un $|C\rangle$ cúbit que luego actúa como control en una operación CNOT $|C\rangle$ sobre $|0\rangle$, es equivalente a una operación AND sobre un cúbit auxiliar (en lugar de actuar sobre $|C\rangle$) y aplicar una puerta CNOT entre el resultado obtenido y $|C\rangle$. El circuito obtenido reemplazando todas las puertas Toffoli (para el caso $N = 3$) se muestra en la Figura 6.5. Su número de puertas T (T-count) es solo $16N$.

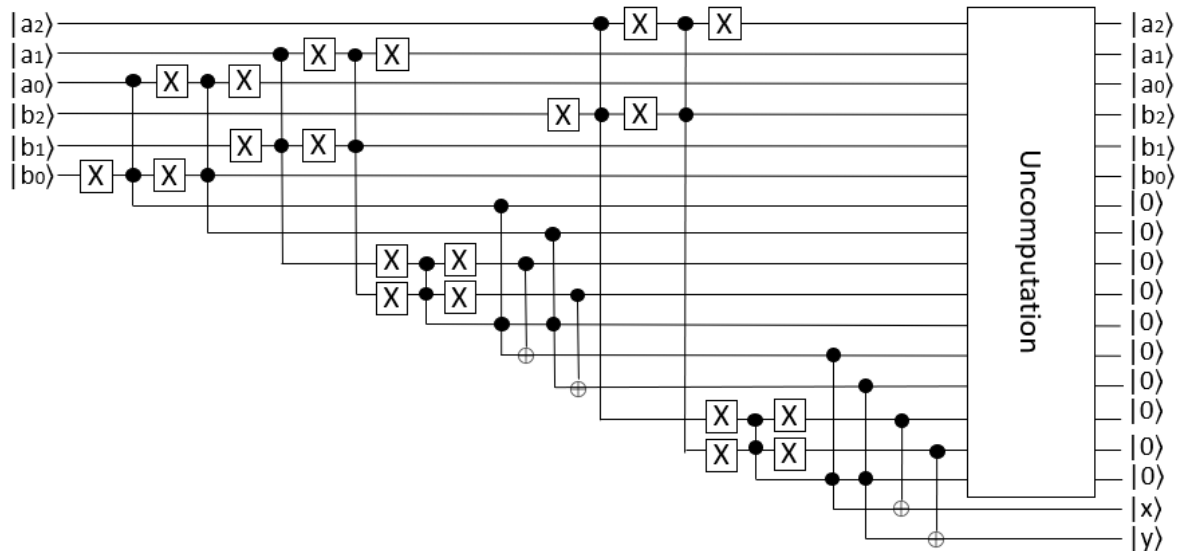


Figura 6.5: Segunda propuesta de circuito comparador cuántico centrado en reducir el número de puertas T, para el caso $N = 3$. a_i y b_i son los bits de los dos números A y B que se van a comparar. Las salidas marcadas como x e y son las salidas de la comparación.

6.4.3 Operador reset en IBM Quantum

El circuito comparador propuesto por Yuan et al. hace uso del “operador reset” u “operador de reinicio”. Este operador inicializa el valor de un cúbit en tiempo de ejecución, pero tiene una serie de implicaciones que resultan en un circuito no reversible. En algunas plataformas de computación cuántica esta operación solo está disponible como una operación que no utiliza puertas Clifford+T y que, además, no es reversible.

En esta sección se explica cómo se realiza la operación de reinicio sobre un cúbit utilizando la plataforma IBM Quantum. Imaginemos dos cúbits, q_0 y q_1 , que realizan operaciones cuánticas entre ellos. Tras inicializar ambos cúbits, se aplica una puerta Hadamard al cúbit q_0 . Posteriormente, se aplica una operación CNOT entre ambas puertas. Finalmente, se desea resetear el cúbit q_1 .

El circuito cuántico implementado en la plataforma IBM Quantum se ilustra en la Figura 6.6. En este circuito, se observa cómo al medir el cúbit q_1 en tiempo de ejecución, se eliminan las propiedades de entrelazamiento cuántico y el circuito se vuelve no reversible. Además, el cúbit q_0 también se vería afectado al estar entrelazado con el cúbit q_1 , y colapsaría. La constante c

auxiliar es necesaria para resetear el cúbit, ya que el resultado de la medición se compara con esta constante, y se se aplica una CNOT si el cúbit q_1 es distinto del valor 0.

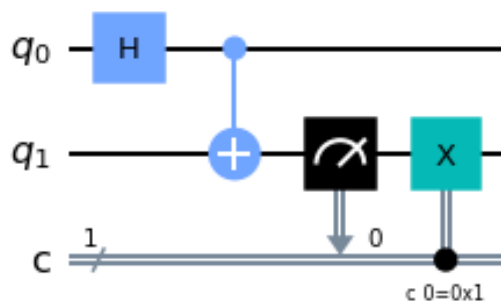


Figura 6.6: Circuito en IBM Quantum que representa la realidad de una operación reset.

El código equivalente en Qiskit se encuentra en el Listado 6.1.

```

1 from qiskit import QuantumCircuit
3 # Crear un circuito cuantico con dos cubits
4 qc = QuantumCircuit(2)
5
6 # Aplicar una puerta Hadamard al cubit q_0
7 qc.h(0)
8
9 # Aplicar una compuerta CNOT entre los cubits q_0 y q_1
10 qc.cx(0, 1)
11
12 # Resetear el cubit q_1
13 qc.reset(1)

```

Listado 6.1: Código Qiskit que ilustra la ejecución de una operación reset utilizando dos cúbits en la plataforma IBM Quantum.

Debido a los inconvenientes del operador reset, en este capítulo se ha su evitado en todas las implementaciones.

6.5 Algoritmo propuesto de segmentación de imágenes con doble umbral

En esta sección se discuten las implementaciones de los algoritmos de segmentación de doble umbral sobre plataformas cuánticas, tanto las realizadas por otros autores como la que hemos propuesto.

Una vez que diseñados los comparadores reversibles y tolerantes a fallos, se puede implementar el algoritmo completo de segmentación de imágenes con doble umbral, tomando como base el propuesto por Yuan et al. [127].

La Figura 6.7 muestra el algoritmo propuesto por Yuan et al. [127], para un ejemplo en el que los límites de umbral alto y bajo se establecen en 6 y 2, respectivamente. Por un lado, este circuito no necesita cúbits auxiliares adicionales, ya que reutiliza los del comparador. Por otro lado, está diseñado para ser utilizado con el comparador propuesto por los mismos autores, por lo que no es compatible con las puertas lógicas que hemos propuesto en la Sección 6.4.2. Además, también tiene operaciones de reinicio.

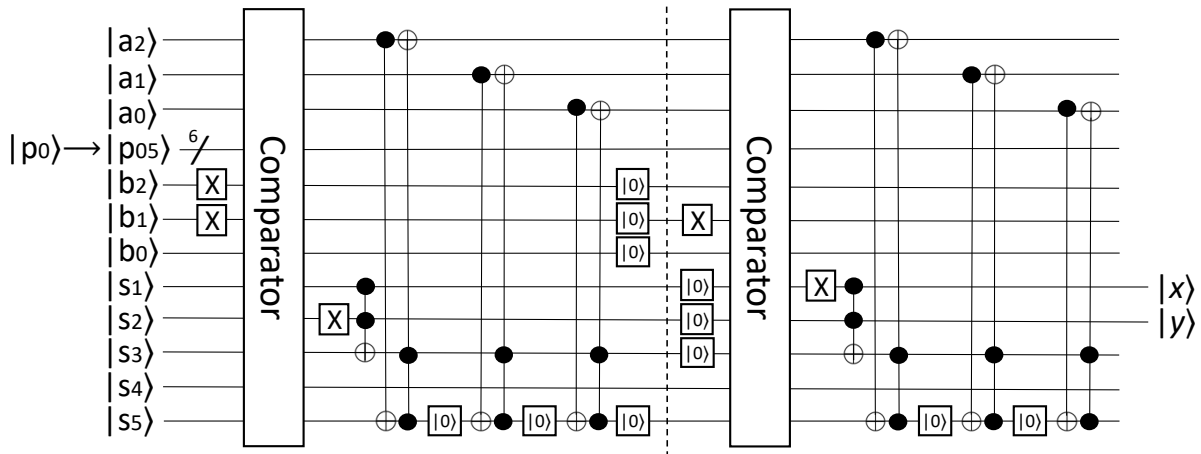


Figura 6.7: Circuito del algoritmo de segmentación de doble umbral propuesto por Yuan et al. [127]. En este ejemplo, el umbral bajo se establece en 2 y el umbral alto en 6. El uso de la operación de reinicio hace que el circuito no sea reversible.

A continuación, se proponen una serie de modificaciones sobre el algoritmo obteniéndose como resultado final nuestra versión de algoritmo de segmentación de doble umbral sobre plataformas cuánticas. En primer lugar, las operaciones de reinicio se pueden eliminar fácilmente aplicando las operaciones inversas. A continuación, las puertas Toffoli que no actúan sobre los cúbits a_i se pueden reemplazar por puertas AND lógicas temporales, reduciendo así el número de puertas T. Debido a que los comparadores que hemos implementado en la Sección anterior no tienen salidas basura, sus cúbits auxiliares se pueden utilizar en este circuito, sin necesidad de incluir cúbits auxiliares adicionales. Un ejemplo de esta nueva versión del algoritmo se muestra en la Figura 6.8.

6.6 Evaluación de los circuitos cuánticos comparadores y de los algoritmos cuánticos de segmentación de doble umbral

En esta sección se ha comparado tanto el algoritmo de segmentación de imágenes de doble umbral como los comparadores diseñados para su implementación, con circuitos y algoritmos similares del estado del arte. En particular, hemos centrado nuestra atención en optimizar dos métricas clave para reducir el coste de los circuitos/algoritmos tolerantes a fallos, T-count y T-depth. También hemos prestado una atención especial al número de cúbits auxiliares, minimizándolos en la medida de lo posible. Estas métricas fueron descritas en detalle en la Sección 2.2.2.

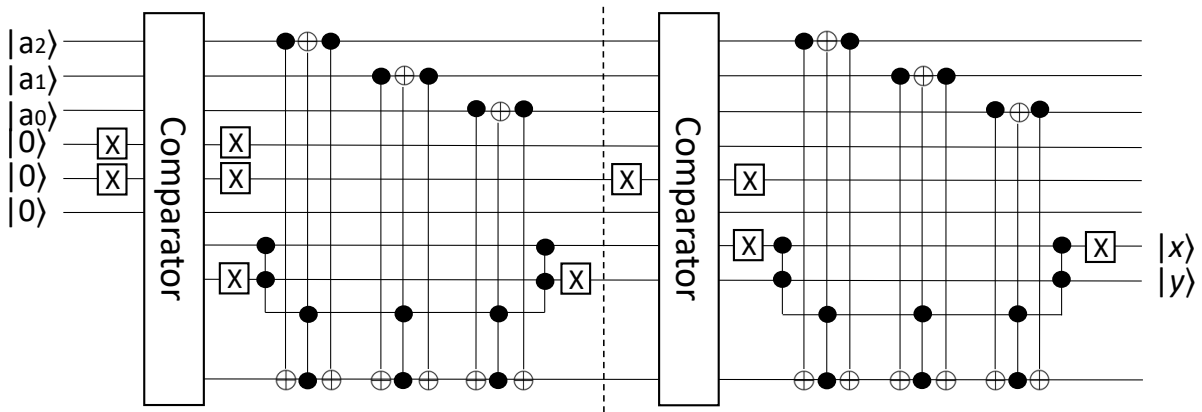


Figura 6.8: Implementación propuesta del algoritmo de segmentación de doble umbral. En este ejemplo, el umbral bajo se establece en 2 y el umbral alto en 6. Se implementa utilizando solo puertas Clifford+T, y reduce el T-count en 6 con respecto al circuito mostrado en la Figura 6.7. Los comparadores requerirán entradas auxiliares adicionales, como se ilustró en las Figuras 6.4 y 6.5.

En la Tabla 6.1 se muestra un estudio comparativo entre los circuitos completos de los comparadores que hemos diseñado (una generalización de las Figuras 6.4 y 6.5) y los comparadores del estado del arte. Si observamos la Tabla 6.1, es importante mencionar que los resultados para los circuitos de Yuan et al. [127] y Wang et al. [114] han requerido dos filas para indicar sus métricas. La primera para los valores originales y otra para mostrar los valores en el caso de reemplazar las operaciones de reinicio con puertas Clifford+T, haciéndolos tolerantes a fallos. Esta tabla muestra los valores de las métricas: T-count (obtenido a partir del número de puertas Toffoli y puertas AND lógicas temporales), T-depth, número de cúbits auxiliares. Además, indica si el circuito contiene solo puertas del tipo Clifford+T y si es un comparador completo o no. Es importante indicar si un comparador es completo o no, porque un comparador completo proporciona más información que un medio comparador y, por lo tanto, tiene más aplicaciones. Ambos algoritmos, el propuesto por Yuan et al. [127] y el que hemos propuesto en este caso de estudio, están preparados para funcionar con comparadores completos, siendo la razón más importante el poder utilizar dos umbrales (el superior y el inferior), demostrando que un medio comparador no es suficiente.

Si analizamos la Tabla 6.1, podemos extraer la siguiente información. Los mejores valores de T-count, T-depth, y número de cúbits auxiliares se han obtenido en los circuitos de Xia et al. [120] y Li et al. [58]. Sin embargo, estos circuitos tienen la limitación de no ser comparadores completos. Si centramos nuestra atención en los comparadores completos, las versiones comparables son las de Yuan et al. [127] y nuestras dos propuestas de circuitos. En este caso, se observa que nuestro segundo circuito propuesto mejora los valores de T-count y T-depth de las dos versiones de Yuan et al. (con y sin operaciones de reinicio). En el caso de T-count, el valor de nuestra segunda propuesta es $16N$ en comparación con los valores de Yuan et al., $35N - 21$ y $70N - 42$. En términos de T-depth, se puede llegar a la misma conclusión. Se obtiene un mejor valor de T-depth con nuestra segunda propuesta en comparación con la versión original y la versión sin reinicios de Yuan et al.: $15N - 9$ para la versión original y $30N - 18$ para la versión modificada, en comparación con $10N - 6$ en nuestro segundo circuito propuesto. En cuanto a los cúbits auxiliares y los comparadores completos, los mejores valores se han logrado con Yuan et

al. sin reinicios y nuestro primer circuito propuesto, ambos con un valor de $3N + 1$. Sin embargo, nuestra propuesta presenta la ventaja de ser tolerante a fallos.

Tabla 6.1: Comparación de T-count, T-depth y cúbits auxiliares entre los comparadores propuestos y los mejores diseños encontrados en la literatura. También se indica si los circuitos utilizan solo puertas Clifford+T y si son completos o medios comparadores.

Circuito	T-count	T-depth	Cúbits auxiliares	Clifford+T exclusivo	Comp. completo
Xia et al. [120]	$14N + 7$	$6N - 3$	3	Sí	No
Li et al. [58]	$14N - 7$	$6N - 3$	1	Sí	No
Yuan et al. [127]	$35N - 21$	$15N - 9$	5	No	Sí
Yuan et al. [127] sin reinicios	$70N - 42$	$30N - 18$	$3N + 1$	Sí	Sí
Wang et al. [114]	$21N - 14$	$6N + 3$	4	No	No
Wang et al. [114] sin reinicios	$42N - 35$	$12N + 6$	$3N - 2$	Sí	No
Circuito comparador propuesto 1	$40N - 32$	$18N - 4$	$3N + 1$	Sí	Sí
Circuito comparador propuesto 2	$16N$	$10N - 6$	$3N + 3$	Sí	Sí

Si ahora nos centramos en el algoritmo para la segmentación de imágenes de doble umbral sobre plataformas cuánticas, se ha realizado un análisis comparativo en la Tabla 6.2. Los enfoques considerados han sido: el propuesto por Yuan et al. [127] (Figura 6.7), el propuesto por Wang et al. [114] y el implementado a lo largo de este capítulo (Figura 6.8). La tabla muestra los valores de T-count, T-depth y número de cúbits auxiliares, y también indica si el circuito es tolerante al fallos mediante el uso en exclusiva de puertas Clifford+T o no. Debe tenerse en cuenta que el valor de T-count no incluye el coste de los comparadores de la Tabla 6.1, que se utilizan dos veces en cada segmentación. Como se puede observar en la Tabla 6.2, el algoritmo de segmentación de doble umbral propuesto mejora tanto el T-count como el T-depth en comparación con las implementaciones de Yuan et al. [127] y Wang et al. [114]. En cuanto al número de cúbits auxiliares, ninguno de los circuitos requiere nuevos cúbits auxiliares, ya que pueden reutilizar los cúbits auxiliares de los comparadores que incluyen. Finalmente, el algoritmo de doble umbral propuesto no requiere el uso de la operación de reinicio a diferencia de las otras dos alternativas. Esto puede considerarse una mejora, ya que como hemos comentado la operación de reinicio puede no estar disponible en la plataforma cuántica.

6.7 Conclusiones y trabajo futuro del caso de estudio de segmentación cuántica de doble umbral

En este capítulo se ha abordado un problema de segmentado de imágenes de doble umbral sobre plataformas cuánticas. Concretamente, hemos propuesto un algoritmo tolerante a fallos, es decir, utilizando exclusivamente puertas del grupo Clifford+T para su implementación. Este hecho es de relevancia debido a que los algoritmos homólogos presentes en la literatura no son compatibles con los códigos de detección y corrección de errores. Además, merece la pena destacar que el algoritmo de segmentación de imágenes de doble umbral ha optimizado los parámetros T-count

Tabla 6.2: Comparación, en términos de T-count, T-depth y número de cúbits auxiliares, entre el circuito original de segmentación de imágenes de doble umbral presentado por Yuan et al. [127], Wang et al. [114] y el propuesto en este trabajo, que incluye tolerancia a fallos mediante el uso en exclusiva del subconjunto de puertas Clifford+T. La métrica T-count no incluye el coste de los comparadores, por lo que a estos valores se debe agregar el del comparador elegido de los que se muestran en la Tabla 6.1 (teniendo en cuenta que el comparador se aplica dos veces). Los circuitos no requieren nuevos cúbits auxiliares, ya que pueden reutilizar los cúbits auxiliares de los comparadores.

Algoritmo	T-count	T-depth	Cúbits auxiliares	Clifford+T exclusivo
Yuan et al. [127]	$14N + 14$	$6N + 6$	0	No
Wang et al. [114]	$21N + 7$	$9N + 3$	0	No
Algoritmo de doble umbral propuesto	$14N + 8$	$6N + 4$	0	Sí

y T-depth. Por otro lado, en este capítulo también se han diseñado dos nuevos comparadores completos para su integración en el algoritmo. Estos comparadores mejoran los actuales en tres aspectos. Primero, están contruidos exclusivamente utilizando puertas Clifford+T, lo que los hace tolerantes a fallos. Segundo, tienen un menor T-count y T-depth, por lo que son más económicos en términos de puertas T y tienen menos sobrecarga. Tercero, son reversibles y fáciles de personalizar.

Merece la pena destacar que todos los circuitos implementados en este capítulo podrían probarse en plataformas cuánticas, como por ejemplo en IBM Quantum. Sin embargo, esto requeriría la implementación de un flujo completo de trabajo de QIMP (codificación de la imagen en NEQR, procesado y decodificación).

Algunas líneas de trabajos futuros son las que se detallan a continuación: (1) el uso de este tipo de comparadores en otros algoritmos (por ejemplo, binarización u ordenamiento); (2) la implementación de algoritmos de doble umbral y circuitos comparadores tolerantes a fallos para nuevas codificaciones, como IFRQI, INEQR, MCRQI, INCQI o QPIXL; y (3) explorar la ejecución del algoritmo propuesto sobre imágenes reales (imágenes de grandes dimensiones que tengan una aplicación práctica real). Los problemas de segmentación suelen utilizar imágenes de alta resolución, lo que implica la necesidad de un alto número de cúbits a codificar. Es necesario realizar un análisis del estado del arte de los computadores cuánticos para establecer un límite para la resolución de las imágenes (o el tamaño de los fragmentos de imagen) que este algoritmo podría segmentar.

7. Conclusiones y trabajo futuro

7.1 Español

Esta tesis se enmarca en el desarrollo de técnicas eficientes de procesamiento de imagen utilizando distintas tecnologías y plataformas computacionales. Para ello, en primer lugar se han explorado una serie de fundamentos teóricos relacionados con el preprocesado. Más tarde, se han explicado distintos tipos de segmentación y, a continuación, se utilizan algunos de estos métodos para resolver dos casos de estudio de interés para la comunidad científica, mejorando el estado del arte para ambos problemas.

La primera contribución de esta tesis ha sido la implementación de una metodología integral que permitiese evaluar de manera automática el efecto de un herbicida sobre el crecimiento de maleza en un campo de girasol. Este enfoque se basa en el análisis de imágenes capturadas mediante una cámara multi-espectral Sequoia, que consta de las bandas NIR, GRE, RED y REG. El hecho de capturar las imágenes a corta distancia utilizando una cámara multi-espectral conllevaba una serie de distorsiones: la causada por la calibración de la lente, así como el error derivado de la perspectiva que resulta de la variación de distancia entre las distintas lentes de la cámara multi-espectral considerada. Se ha efectuado un preprocesado con el fin de reducir estos efectos y así obtener imágenes multi-espectrales de mayor calidad. Las imágenes multi-espectrales obtenidas después de llevar a cabo los procesos de: calibración de la lente, rectificación al plano principal y co-registro, se han utilizado como entrada para un posterior proceso de segmentación/clasificación de todos los píxeles de la imagen en tres clases, maleza, girasol y suelo. El algoritmo de clasificación basado en objetos se ha implementado utilizando diferentes rutinas del software eCognition, que hace uso de información tanto multi-espectral como geométrica. La información de carácter geométrico abarca aspectos como la forma y el tamaño de la planta de interés y las malezas circundantes. La inclusión de datos geométricos se tornó imprescindible debido a que la firma espectral de las plantas presentaba una considerable

variabilidad, lo que limitaba su capacidad para discriminar de manera precisa entre la maleza y el cultivo de interés.

El algoritmo propuesto, denominado ClasOb, ha realizado una segmentación y posterior clasificación de todos los píxeles de las imágenes en varias categorías: suelo, girasol y maleza. Dado que la identificación del suelo se ha basado en el principio de eliminación, no ha proporcionado una medida precisa del rendimiento. Sin embargo, el coeficiente de similitud de Jaccard (IoU) para la materia vegetal (combinación de girasol y maleza) ha resultado ser una métrica de interés para evaluar la capacidad del algoritmo en la segmentación general de las plantas. Para cuantificar la precisión del algoritmo ClasOb, se ha utilizado la métrica mIoU (IoU promedio), que ha arrojado valores de 0.91, 0.89 y 0.72 para la clasificación de materia vegetal, girasol y maleza, respectivamente. Estos valores se consideran aceptables en términos de clasificación. Adicionalmente, se ha comparado la propuesta ClasOb con dos métodos de segmentación basados en aprendizaje profundo que hemos entrenado utilizando tres conjuntos de datos distintos, obteniendo un total de seis redes neuronales. Estas redes han sido entrenadas preliminarmente para seleccionar los enfoques de aprendizaje profundo más competitivos para el problema de clasificación, obteniéndose U-Net_Multi y FPN_I_Jes_Multi. Además, este estudio preliminar ha revelado que, cuando los datos de entrenamiento son limitados, es más efectivo aumentar el tamaño de nuestro conjunto de datos específico mediante técnicas de aumento de datos en lugar de utilizar conjuntos de datos de problemas relacionados. Si se compara nuestra propuesta ClasOb con U-Net_Multi y FPN_I_Jes_Multi, se ha demostrado que obtiene un mejor rendimiento en la clasificación de la materia vegetal y la maleza. Además, nuestra estrategia propuesta, a diferencia de las basadas en DL, no requiere la creación de conjuntos de datos adicionales (y de sus mapas de verdad) ni el proceso de entrenamiento del modelo, que son tareas que consumen tiempo y recursos. En consecuencia, podemos determinar que en campos de estudio con disponibilidad limitada de imágenes (como el abordado en esta tesis), es más eficiente y se obtienen resultados más satisfactorios al implementar un algoritmo basado en clasificación de objetos que uno de enfoque de aprendizaje profundo.

Podemos concluir que a través del trabajo desarrollado se ha conseguido un algoritmo novedoso (ClasOb) cuya ventaja principal radica en que no demanda grandes volúmenes de datos de entrenamiento, a diferencia de las técnicas basadas en redes neuronales convolucionales.

Merece la pena destacar que además de la contribución relacionada con la implementación de la herramienta automática para la identificación de malas hierbas en pruebas de herbicidas, en el Capítulo 5 hemos realizado la contribución de establecer una metodología para corregir de forma automática las aberraciones de las imágenes multi-espectrales de la cámara Sequoia. Esta contribución es muy relevante ya que el proceso de calibración y corrección que hemos desarrollado podrían extenderse a otras cámaras multi-espectrales o a otras áreas de estudio. Los resultados del Capítulo 5 han culminado en la publicación de un artículo de impacto publicado en el primer cuartil. Por último, todos los algoritmos desarrollados, las imágenes sin procesar, los mapas de verdad de terreno y las imágenes resultantes de la clasificación están disponibles públicamente en el siguiente enlace: <https://github.com/Heikelol/SegmentationForPlants>

En cuanto a los trabajos futuros, estamos considerando la posibilidad de extender nuestra investigación para trabajar con otros tipos de cultivos. Además, tenemos la intención de seguir explorando la estrategia basada en redes neuronales, abordando de manera más exhaustiva este

problema específico. Esto implica la inclusión de un mayor número de imágenes en nuestras bases de datos y la exploración de diferentes enfoques de aprendizaje profundo, como las redes neuronales convolucionales orientadas a objetos disponible en el software Trimble eCognition. Se plantea la posibilidad de realizar arquitecturas híbridas que hagan uso al mismo tiempo de enfoques clásicos y enfoques basados en redes neuronales, lo que podría llevar a una mejor tasa de detección y clasificación en los cultivos.

La segunda contribución de esta tesis ha sido el desarrollo de un algoritmo de segmentación de imágenes de doble umbral sobre plataformas cuánticas que se ha llevado a cabo en el Capítulo 6.

Este algoritmo ha sido diseñado en base a la metodología basada en circuitos. Concretamente lo que se ha realizado ha sido incorporar tolerancia a fallos mediante el uso exclusivo del subconjunto de puertas Clifford+T para su implementación, lo que lo hace apto para su utilización en simuladores, especialmente a pequeña escala. Debido a que la implementación tolerante a fallos de las puertas T tiene un coste mucho mayor que otras puertas cuánticas, el objetivo de diseño ha sido minimizar la cantidad de estas puertas. Los diseños se han optimizado en base a dos métricas de rendimiento clave, T-Count (número de puertas T que tiene un circuito) y T-Depth (número de puertas T que se deben computar secuencialmente). Este algoritmo ha sido comparado con el estado del arte demostrando que es mejor en coste de puertas T.

Debido a que la segmentación de imágenes de doble umbral involucra la operación de comparación, como parte de este trabajo hemos implementado dos circuitos cuánticos comparadores completos y tolerantes a fallos. Estos circuitos comparadores han presentado mejoras significativas en tres aspectos clave. En primer lugar, están contruidos exclusivamente utilizando puertas Clifford+T, lo que permite su tolerancia a fallos mediante el uso de códigos de error. En segundo lugar, los circuitos desarrollados requieren un menor número y profundidad de puertas T, lo que reduce la demanda de recursos y la sobrecarga. En tercer lugar, son reversibles y personalizables, lo que los convierte en viables para ejecutarse en plataformas de computación cuántica reales como IBM Quantum. Los resultados del Capítulo 6 han culminado en la publicación de un artículo de impacto publicado en el segundo cuartil.

En relación a futuras investigaciones vinculadas a este trabajo, se contemplan varias direcciones. En primer lugar, se podría evaluar la aplicabilidad de estos comparadores en otros algoritmos, como aquellos relacionados con la binarización y el ordenamiento. En segundo lugar, es posible diseñar algoritmos de doble umbral y circuitos comparadores tolerantes a fallos para nuevas codificaciones, tales como IFRQI, INEQR, MCRQI, INCQI o QPIXL. Estas codificaciones son menos conocidas que NEQR pero cuentan con algunas ventajas dependiendo de la aplicación.

Finalmente, sería interesante explorar la viabilidad de implementar el algoritmo propuesto sobre problemas reales de segmentación. Normalmente, los problemas reales involucran imágenes de alta resolución, por lo que será necesario realizar un análisis exhaustivo del estado del arte de las plataformas de computación cuántica para determinar el límite de la resolución de imágenes, o el tamaño de los fragmentos de imagen, que el presente algoritmo podría segmentar.

7.2 English

This thesis is defined within the development of efficient image processing techniques using various technologies and computational platforms. To achieve this, a series of theoretical foundations related to preprocessing have been explored initially. Subsequently, different segmentation types have been explained, and some of these methods are employed to address two case studies of interest to the scientific community, thereby improving the state of the art for both problems.

The primary contribution of this thesis involves the implementation of a comprehensive methodology to automatically evaluate the effect of an herbicide on weed growth in a sunflower field. This approach is based on the analysis of images captured by a Sequoia multispectral camera, which includes NIR, GRE, RED, and REG bands. Capturing images at close range using a multispectral camera introduced distortions caused by lens calibration and errors resulting from perspective variations between the camera's lenses. Preprocessing was conducted to mitigate these effects and obtain higher-quality multispectral images. The calibrated, rectified, and co-registered multispectral images served as input for a segmentation/classification process categorizing all pixels into three classes: weed, sunflower, and soil. The object-based classification algorithm was implemented using various eCognition software routines, incorporating both multispectral and geometric information. Geometric information included aspects such as the shape and size of the target plant and surrounding weeds. The inclusion of geometric data was crucial due to the considerable variability in the spectral signature of plants, limiting the ability to discriminate accurately between weeds and the target crop.

The proposed algorithm, named ClasOb, successfully segmented and classified all pixels into soil, sunflower, and weed categories. Since soil identification was elimination-based, it did not provide a precise measure of performance. However, the Jaccard similarity coefficient (IoU) for vegetation (combined sunflower and weed) proved to be a relevant metric for evaluating the algorithm's overall plant segmentation capability. To quantify ClasOb's accuracy, the mIoU metric (average IoU) yielded values of 0.91, 0.89, and 0.72 for vegetation, sunflower, and weed classification, respectively. These values are considered acceptable in classification terms. Additionally, ClasOb was compared with two deep learning-based segmentation methods trained on three different datasets, resulting in U-Net_Multi and FPN_I_Jes_Multi. This preliminary study revealed that, when training data is limited, augmenting the specific dataset size is more effective than using datasets from related problems. ClasOb outperformed U-Net_Multi and PN_I_Jes_Multi in vegetation and weed classification. Importantly, unlike DL-based approaches, ClasOb does not require creating additional datasets (and their ground truth maps) or training the model, tasks that are time and resource-consuming. Consequently, in fields with limited image availability (as addressed in this thesis), implementing an object-based classification algorithm is more efficient and yields more satisfactory results than a deep learning approach.

In conclusion, the developed work resulted in a novel algorithm (ClasOb) with a key advantage of not requiring extensive training data, unlike convolutional neural network-based techniques.

It is noteworthy that, in addition to the contribution related to implementing an automatic tool for weed identification in herbicide tests, in Chapter 5, we have contributed to establishing

a methodology for automatically correcting aberrations in Sequoia multispectral images. This contribution is significant as the calibration and correction process developed could be extended to other multispectral cameras or other study areas. The Chapter 5 results led to the publication of a high-impact article in the first quartile. Finally, all developed algorithms, raw images, ground truth maps, and resulting classification images are publicly available at the following link: <https://github.com/Heikelol/SegmentationForPlants>.

Regarding future work, we are considering extending our research to work with other types of crops. Additionally, we intend to further explore the neural network-based strategy, addressing this specific problem more comprehensively. This involves including more images in our databases and exploring different deep learning approaches, such as object-oriented convolutional neural networks available in Trimble eCognition software. The possibility of creating hybrid architectures that simultaneously use classical and neural network-based approaches is being considered, potentially leading to improved detection and classification rates in crops.

The second contribution of this thesis has been the development of a double-threshold image segmentation algorithm on quantum platforms, as discussed in Chapter 6.

This algorithm has been designed based on circuit-based methodology. Specifically, fault tolerance has been incorporated by exclusively using the subset of Clifford+T gates for its implementation, making it suitable for use in simulators, especially at a small scale. Since the fault-tolerant implementation of T gates has a much higher cost than other quantum gates, the design goal has been to minimize the number of these gates. Designs have been optimized based on two key performance metrics: T-Count (the number of T gates in a circuit) and T-Depth (the number of T gates that must be computed sequentially). This algorithm has been compared with the state of the art, demonstrating its superiority in terms of T gate cost.

Because double-threshold image segmentation involves the comparison operation, as part of this work, we have implemented two complete and fault-tolerant quantum comparator circuits. These comparator circuits have shown significant improvements in three key aspects. First, they are constructed exclusively using Clifford+T gates, allowing fault tolerance through error codes. Second, the developed circuits require fewer T gates and have lower depth, reducing resource demand and overhead. Third, they are reversible and customizable, making them viable for execution on real quantum computing platforms like IBM Quantum. The results of Chapter 6 led to the publication of a high-impact article in the second quartile.

Regarding future research related to this work, several directions are contemplated. Firstly, the applicability of these comparators in other algorithms, such as those related to binarization and sorting, could be evaluated. Secondly, it is possible to design fault-tolerant double-threshold algorithms and comparators for new encodings, such as IFRQI, INEQR, MCRQI, INCQI, or QPIXL. These encodings are less known than NEQR but have some advantages depending on the application.

Finally, it would be interesting to explore the feasibility of implementing the proposed algorithm on real segmentation problems. Real problems typically involve high-resolution images, necessitating a comprehensive analysis of the state of the art in quantum computing

platforms to determine the limit of image resolution or the size of image fragments that the present algorithm could segment.



Bibliografía

- [1] M. Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*. 2016. arXiv: 1603.04467 [cs.DC].
- [2] R. Adams and L. Bischof. “Seeded region growing”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16 (1994), pages 641–647.
- [3] F. Agüera-Vega, M. Agüera-Puntas, J. Agüera-Vega, P. Martínez-Carricondo, and F. Carvajal-Ramírez. “Multi-sensor imagery rectification and registration for herbicide testing”. In: *Measurement* 175 (2021), page 109049. ISSN: 0263-2241.
- [4] M. Amy, D. Maslov, and M. Mosca. “Polynomial-Time T-Depth Optimization of Clifford+T Circuits Via Matroid Partitioning”. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 33.10 (Oct. 2014), pages 1476–1489. DOI: 10.1109/tcad.2014.2341953. URL: <https://doi.org/10.1109/tcad.2014.2341953>.
- [5] M. Baatz and A. Schäpe. “Multiresolution Segmentation : an optimization approach for high quality multi-scale image segmentation”. In: *Proceedings of Angewandte Geographische Informationsverarbeitung XII*. 2000, pages 12–23.
- [6] J. Baluja et al. “Assessment of vineyard water status variability by thermal and multi-spectral imagery using an unmanned aerial vehicle (UAV)”. In: *Irrigation Science* 30 (Nov. 2012), pages 511–522. DOI: 10.1007/s00271-012-0382-9.
- [7] L. Barghout and L. W. Lee. “Perceptual information processing system”. U.S. Patent Application 10/618,543. July 2003.
- [8] C. H. Bennett. “Logical reversibility of computation”. In: *IBM journal of Research and Development* 17.6 (1973), pages 525–532. DOI: 10.1147/rd.176.0525.

- [9] J. Berni, P. Zarco-Tejada, L. Suárez, and E. Fereres. “Thermal and narrowband multi-spectral remote sensing for vegetation monitoring from an unmanned aerial vehicle”. In: *IEEE Transactions on Geoscience and Remote Sensing* 47.3 (2009), pages 722–738. DOI: 10.1109/TGRS.2008.2010457.
- [10] E. Borovitskaya. *Quantum Entanglement: Concepts and Applications*. Springer, 2019. ISBN: 978-3-030-20270-5.
- [11] J. Y. Bouguet. “Camera Calibration Toolbox for Matlab”. In: *Computational Vision at the California Institute of Technology*. 2015.
- [12] G. Bradski and A. Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*. Sebastopol, CA: O’Reilly, 2008.
- [13] S. Caraiman and V. I. Manta. “Histogram-based segmentation of quantum images”. In: *Theoretical Computer Science* 529 (2014), pages 46–60. ISSN: 0304-3975. DOI: <https://doi.org/10.1016/j.tcs.2013.08.005>. URL: <https://www.sciencedirect.com/science/article/pii/S0304397513005835>.
- [14] S. Caraiman and V. I. Manta. “Image Segmentation on a Quantum Computer”. In: *Quantum Information Processing* 14.5 (May 2015), pages 1693–1715. ISSN: 1570-0755. DOI: 10.1007/s11128-015-0932-1. URL: <https://doi.org/10.1007/s11128-015-0932-1>.
- [15] P. Cattin. *Image Restoration: Introduction to Signal and Image Processing*. Archived from the original on 2016-09-18. Retrieved 11 October 2013. MIAC, University of Basel, 2012. URL: https://web.archive.org/web/20160918000000/http://www.miac.unibas.ch:80/d/teaching/Vorlesungen/2012_REST/coursedescription.pdf.
- [16] C. Chen, X. Wang, A. A. Heidari, H. Yu, and H. Chen. “Multi-Threshold Image Segmentation of Maize Diseases Based on Elite Comprehensive Particle Swarm Optimization and Otsu”. In: *Frontiers in Plant Science* 12 (2021). ISSN: 1664-462X. DOI: 10.3389/fpls.2021.789911. URL: <https://www.frontiersin.org/articles/10.3389/fpls.2021.789911>.
- [17] T. Clarke and J. Fryer. “The Development of Camera Calibration Methods and Models”. In: *The Photogrammetric Record* 16 (1998), pages 51–66. DOI: 10.1111/0031-868X.00113. URL: <https://doi.org/10.1111/0031-868X.00113>.
- [18] T. Datascience. *Redes FPN*. Accedido el 14 de agosto de 2023. 2023. URL: <https://towardsdatascience.com>.
- [19] W. Demtröder. *Electrodynamics and Optics*. Undergraduate Lecture Notes in Physics. Published: 16 September 2019. Springer Cham, 2019. ISBN: 978-3-030-02289-1. DOI: 10.1007/978-3-030-02291-4.
- [20] S. J. Devitt, A. M. Stephens, W. J. Munro, and K. Nemoto. “Requirements for fault-tolerant factoring on an atom-optics quantum computer”. In: *Nature communications* 4.1 (2013), pages 1–8. DOI: <https://doi.org/10.1038/ncomms3524>.
- [21] *Diario Oficial de la Unión Europea, Reglamento (CE) n° 1107/2009 del Parlamento Europeo y del Consejo, de 21 de octubre de 2009, relativo a la comercialización de productos fitosanitarios y por el que se derogan las Directivas 79/117/CEE y 91/414/CEE del Consejo*. 2009. URL: <http://data.europa.eu/eli/reg/2009/1107/oj>.

- [22] D. P. DiVincenzo. “Quantum gates and circuits”. en. In: *Proc. Math. Phys. Eng. Sci.* 454.1969 (Jan. 1998), pages 261–276.
- [23] L. Drăguț, O. Csillik, C. Eisank, and D. Tiede. “Automated parameterisation for multi-scale image segmentation on multiple layers”. In: *ISPRS journal of photogrammetry and remote sensing* 88.100 (2014), pages 119–127.
- [24] S. G.-C. Elias F. Combarro. *A Practical Guide to Quantum Machine Learning and Quantum Optimization. Hands-on Approach to Modern Quantum Algorithms*. Packt, 2023.
- [25] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. “A density-based algorithm for discovering clusters in large spatial databases with noise”. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*. Edited by E. Simoudis, J. Han, and U. M. Fayyad. AAAI Press, 1996, pages 226–231. ISBN: 1-57735-004-9.
- [26] M. Fawakherji, C. Potena, A. Pretto, D. D. Bloisi, and D. Nardi. “Multi-Spectral Image Synthesis for Crop/Weed Segmentation in Precision Farming”. In: *Robotics and Autonomous Systems* 146 (C 2021).
- [27] R. Fergus, B. Singh, A. Hertzmann, S. Roweis, and W. Freeman. “Removing camera shake from a single photograph”. In: *SIGGRAPH*. 2006.
- [28] R. P. Feynman. *The Feynman Lectures on Physics: Volume III - Quantum Mechanics*. Basic Books, 2011. ISBN: 978-0465025015.
- [29] R. A. Finkel and J. L. Bentley. “Quad trees a data structure for retrieval on composite keys”. In: *Acta Informatica* 4.1 (1974), pages 1–9. DOI: 10.1007/BF00288933.
- [30] D. Flanders, M. Hall-Beyer, and J. Pereverzoff. “Preliminary evaluation of eCognition object-based software for cut block delineation and feature extraction”. In: *Can. J. Remote Sens./J. Can. Teledetect.* 29.4 (Jan. 2003), pages 441–452.
- [31] D. A. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, 2003.
- [32] K. Gangrade. “A Review of Various Digital Image Preprocessing Methods for Medical Image Analysis”. In: *International Journal of Computer Sciences and Engineering* 7 (Jan. 2019), pages 513–516. DOI: 10.26438/ijcse/v7i1.513516.
- [33] C. Gidney. “Halving the cost of quantum addition”. In: *Quantum* 2 (2018), page 74. DOI: <https://doi.org/10.22331/q-2018-06-18-74>.
- [34] R. Gonzalez and R. Woods. *Digital Image Processing*. 3rd edn. Pearson Education, London. 2008. ISBN: 9780131687288.
- [35] A. Goshtasby. “Piecewise linear mapping functions for image registration”. In: *Pattern Recognition* 19.6 (1986), pages 459–466. ISSN: 0031-3203.
- [36] A. Goshtasby. “Image registration by local approximation methods”. In: *Image and Vision Computing* 6.4 (1988), pages 255–261. ISSN: 0262-8856.
- [37] E. K. Grant and T. S. Humble. *Adiabatic Quantum Computing and Quantum Annealing*. July 2020. DOI: 10.1093/acrefore/9780190871994.013.32. URL: <https://oxfordre.com/physics/view/10.1093/acrefore/9780190871994.001.0001/acrefore-9780190871994-e-32>.
- [38] L. K. Grover. *A fast quantum mechanical algorithm for database search*. 1996. arXiv: quant-ph/9605043 [quant-ph].

- [39] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Second. Cambridge University Press, ISBN: 0521540518, 2004.
- [40] K. Hata and S. Savarese. *CS231A Course Notes 1: Camera Models*. University of Stanford. Feb. 2021. URL: https://web.stanford.edu/class/cs231a/course_notes/01-camera-models.pdf.
- [41] K. He, X. Zhang, S. Ren, and J. Sun. “Deep Residual Learning for Image Recognition”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).
- [42] J. Heikkila and O. Silven. “A four-step camera calibration procedure with implicit image correction”. In: *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 1997, pages 1106–1112. DOI: 10.1109/CVPR.1997.609468.
- [43] J. Heikkila and O. Silven. “A Four-step Camera Calibration Procedure with Implicit Image Correction”. In: *IEEE International Conference on Computer Vision and Pattern Recognition*. 1997.
- [44] J. Hemming and T. Rath. “PA—Precision Agriculture: Computer-Vision-based Weed Identification under Field Conditions using Controlled Lighting”. In: *Journal of Agricultural Engineering Research* 78 (Mar. 2001), pages 233–243. DOI: 10.1006/jaer.2000.0639.
- [45] R. Horodecki, P. Horodecki, M. Horodecki, and K. Horodecki. “Quantum entanglement”. In: *Reviews of modern physics* 81.2 (2009), page 865.
- [46] W. Hung, X. Song, G. Yang, J. Yang, and M. Perkowski. “Optimal synthesis of multiple output Boolean functions using a set of quantum gates by symbolic reachability analysis”. In: *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on* 25 (Oct. 2006), pages 1652–1663. DOI: 10.1109/TCAD.2005.858352.
- [47] IBM. “IBM Quantum Experience”. In: *Sitio web oficial de IBM Quantum* (2023). URL: <https://www.ibm.com/quantum/>.
- [48] B. Iqbal and H. Singh. *Identification of Desired Pixels in an Image Using Grover’s Quantum Search Algorithm*. 2021. arXiv: 2107.03053 [quant-ph].
- [49] J.-P. Jhan, J.-Y. Rau, and N. Haala. “Robust and adaptive band-to-band image transform of UAS miniature multi-lens multispectral camera”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 137 (2018), pages 47–60. ISSN: 0924-2716.
- [50] J.-P. Jhan, J.-Y. Rau, and C.-Y. Huang. “Band-to-band registration and ortho-rectification of multilens/multispectral imagery: A case study of MiniMCA-12 acquired by a fixed-wing UAS”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 114 (2016), pages 66–77. ISSN: 0924-2716. DOI: <https://doi.org/10.1016/j.isprsjprs.2016.01.008>.
- [51] G. Kochenberger et al. “The unconstrained binary quadratic programming problem: a survey”. In: *Journal of Combinatorial Optimization* 28 (2014), pages 58–81.
- [52] D. Korsch. *Reflective Optics*. Academic Press, 1991. ISBN: 0-12-421170-4.
- [53] A. Krizhevsky, I. Sutskever, and G. E. Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems*. 2012.

- [54] P. Kumar, P. Rao, and K. H. Kishore. “Optimal design of reversible parity preserving new Full adder / Full subtractor”. In: *2017 11th International Conference on Intelligent Systems and Control (ISCO)* (2017), pages 368–373.
- [55] E. Y. Lam. “Combining gray world and retinex theory for automatic white balance in digital photography”. In: *Proceedings of the Ninth International Symposium on Consumer Electronics, 2005.(ISCE 2005)*. IEEE. 2005, pages 134–139.
- [56] P. Le, A. Ilyasu, F. Dong, and K. Hirota. “Multi-dimensional color image storage and retrieval for a normal arbitrary quantum superposition state”. In: *IAENG International Journal of Applied Mathematics* 40.3 (2010), pages 113–123.
- [57] A. Levin, Y. Weiss, and F. Durand. “Understanding and evaluating blind deconvolution algorithms”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31.4 (2009), pages 648–663.
- [58] H.-S. Li, P. Fan, H. Xia, H. Peng, and G.-L. Long. “Efficient quantum arithmetic operation circuits for quantum image processing”. In: *Science China Physics, Mechanics & Astronomy* 63.8 (2020), pages 1–13. DOI: 10.1007/s11433-020-1582-8.
- [59] H.-S. Li et al. “Image storage, retrieval, compression and segmentation in a quantum system”. In: *Quantum Information Processing* 12.6 (2013), pages 2269–2290. DOI: 10.1007/s11128-012-0521-5.
- [60] T.-Y. Lin et al. “Feature Pyramid Networks for Object Detection”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pages 936–944. DOI: 10.1109/CVPR.2017.106.
- [61] T.-Y. Lin et al. “Feature Pyramid Networks for Object Detection”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.
- [62] X. Lin et al. “Recent Advances in Passive Digital Image Security Forensics: A Brief Review”. In: *Engineering* 4.1 (2018). Cybersecurity, pages 29–39. ISSN: 2095-8099. DOI: <https://doi.org/10.1016/j.eng.2018.02.008>. URL: <https://www.sciencedirect.com/science/article/pii/S2095809917307890>.
- [63] T. Lindeberg. “Feature Detection with Automatic Scale Selection”. In: *International Journal of Computer Vision* 30.2 (Nov. 1998), pages 79–116. ISSN: 1573-1405. DOI: 10.1023/A:1008045108935. URL: <https://doi.org/10.1023/A:1008045108935>.
- [64] S. Lomte and A. Janwale. “Plant Leaves Image Segmentation Techniques: A Review”. In: *International Journal of Computer Sciences and Engineering* 5 (5 2017), pages 147–150.
- [65] M. Louargant et al. “Unsupervised Classification Algorithm for Early Weed Detection in Row-Crops by Combining Spatial and Spectral Information”. In: *Remote Sensing* 10 (2018), pages 761–779. DOI: <https://doi.org/10.3390/rs10050761>.
- [66] A. Lucas. “Ising formulations of many NP problems”. In: *Frontiers in Physics* 2 (2014). DOI: 10.3389/fphy.2014.00005. URL: <https://doi.org/10.3389/fphy.2014.00005>.
- [67] L. MacDonald. *Digital Heritage*. Butterworth-Heinemann, 2006. ISBN: 0-7506-6183-6.
- [68] J. B. MacQueen. “Some Methods for Classification and Analysis of Multivariate Observations”. In: *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*. Volume 1. University of California Press, 1967, pages 281–297.

- [69] MathWorks. *Camera Calibration*. <https://es.mathworks.com/help/vision/ug/camera-calibration.html>. Último acceso el 1 de junio de 2023. Accedido en 2023.
- [70] M. Mohammadi and M. Eshghi. “On figures of merit in reversible and quantum logic designs”. In: *Quantum Information Processing* 8.4 (2009), pages 297–318.
- [71] U. Mujika Torrontegi. “Reconstrucción densa de modelos tridimensionales utilizando Visión Artificial”. Master’s thesis. Donostia: Universidad del País Vasco / Euskal Herriko Unibertsitatea, Sept. 2010.
- [72] E. Muñoz-Coreas and H. Thapliyal. “Quantum Circuit Design of a T-count Optimized Integer Multiplier”. In: *IEEE Trans. Computers* 68.5 (2019), pages 729–739. DOI: 10.1109/TC.2018.2882774.
- [73] A. M. El-naggar. “Determination of optimum segmentation parameter values for extracting building from remote sensing images”. In: *Alexandria Engineering Journal* 57.4 (2018), pages 3089–3097. ISSN: 1110-0168. DOI: <https://doi.org/10.1016/j.aej.2018.10.001>.
- [74] M. Nielsen and I. Chuang. *Quantum Computation and Quantum Information*. Cambridge, 2010.
- [75] NVIDIA. *DLSS: Deep Learning Super Sampling*. <https://www.nvidia.com/es-es/geforce/technologies/dlss/>. [Accedido el 24 de mayo de 2023].
- [76] S. Ojha and S. Sakhare. “Image processing techniques for object tracking in video surveillance- A survey”. In: Jan. 2015, pages 1–6. DOI: 10.1109/PERVASIVE.2015.7087180.
- [77] F. Orts, G. Ortega, A. C. Cucura, E. Filatovas, and E. M. Garzón. “Optimal fault-tolerant quantum comparators for image binarization”. In: *The Journal of Supercomputing* 77 (2021), pages 8433–8444. DOI: <https://doi.org/10.1007/s11227-020-03576-5>.
- [78] N. Otsu. “A threshold selection method from gray-level histograms”. In: *IEEE Transactions on Systems, Man, and Cybernetics* 9.1 (1979), pages 62–66. ISSN: 2168-2232. DOI: 10.1109/TSMC.1979.4310076. URL: <https://ieeexplore.ieee.org/document/4310076>.
- [79] T. Ottmann. “14th International Colloquium on Automata, Languages and Programming”. In: Karlsruhe, Germany: Springer-Verlag, 1987. ISBN: 0387180885.
- [80] A. Paler and S. J. Devitt. “An Introduction into Fault-Tolerant Quantum Computing”. In: *Proceedings of the 52nd Annual Design Automation Conference*. DAC ’15. San Francisco, California: Association for Computing Machinery, 2015. ISBN: 9781450335201. DOI: 10.1145/2744769.2747911. URL: <https://doi.org/10.1145/2744769.2747911>.
- [81] A. Paler, I. Polian, K. Nemoto, and S. J. Devitt. “Fault-tolerant, high-level quantum circuits: form, compilation and description”. In: *Quantum Science and Technology* 2.2 (2017), page 025003. DOI: <https://doi.org/10.1088/2058-9565/aa66eb>.
- [82] L. Q. Phuc, D. Fangyang, and H. Kaoru. “A flexible representation of quantum images for polynomial preparation, image compression, and processing operations”. In: *Quantum Information Processing* 10 (1 2011), pages 63–84. DOI: 10.1007/s11128-010-0177-y.
- [83] E. R. Pike and S. Sarkar. *Quantum Interference and Coherence: Theory and Experiments*. Oxford University Press, 2020. ISBN: 9780198786773.

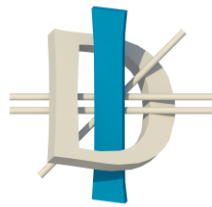
- [84] S. M. Pizer et al. “Adaptive histogram equalization and its variations”. In: *Computer Vision, Graphics, and Image Processing* 39.3 (1987), pages 355–368. ISSN: 0734-189X. DOI: [https://doi.org/10.1016/S0734-189X\(87\)80186-X](https://doi.org/10.1016/S0734-189X(87)80186-X). URL: <https://www.sciencedirect.com/science/article/pii/S0734189X8780186X>.
- [85] I. Polian and A. G. Fowler. “Design automation challenges for scalable quantum architectures”. In: *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*. 2015, pages 1–6. DOI: 10.1145/2744769.2747921.
- [86] W. K. Pratt. *Digital image processing*. Wiley, 2007.
- [87] S. Quegan and J. Yu. “Filtering of Multichannel SAR Images”. In: *IEEE Transactions on Geoscience and Remote Sensing* 39.11 (2001), pages 2373–2379.
- [88] B. S. Rao, M. Hota, and U. Kumar. “Crop Classification from UAV-Based Multi-spectral Images Using Deep Learning”. In: *Computer Vision and Image Processing. CVIP 2020. Communications in Computer and Information Science*. Edited by S. S.K., R. P., R. B., and N. P. Volume 11376. Singapore: Springer, 2021.
- [89] R. A. Robb. “Introduction to Visualization”. In: *Handbook of Medical Imaging*. Edited by Isaac N. Bankman. Biomedical Engineering. San Diego: Academic Press, 2000, pages 655–658. ISBN: 978-0-12-077790-7. URL: <http://www.sciencedirect.com/science/article/pii/B9780120777907500473> (visited on 09/10/2013).
- [90] O. Ronneberger, P. Fischer, and T. Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. 2015. DOI: 10.48550/ARXIV.1505.04597. URL: <https://arxiv.org/abs/1505.04597>.
- [91] O. Ronneberger, P. Fischer, and T. Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Edited by N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi. Cham: Springer International Publishing, 2015, pages 234–241. ISBN: 978-3-319-24574-4.
- [92] O. Russakovsky et al. *ImageNet Large Scale Visual Recognition Challenge*. 2014. DOI: 10.48550/ARXIV.1409.0575. URL: <https://arxiv.org/abs/1409.0575>.
- [93] S. E. Salcudean, H. Moradi, D. G. Black, and N. Navab. “Robot-Assisted Medical Imaging: A Review”. In: *Proceedings of the IEEE* 110.7 (2022), pages 951–967. DOI: 10.1109/JPROC.2022.3162840.
- [94] K. Sarra and A. Boulmerka. “Semantic segmentation of remote sensing images using U-net and its variants : Conference New Technologies of Information and Communication (NTIC 2022)”. In: *2022 2nd International Conference on New Technologies of Information and Communication (NTIC)*. 2022, pages 1–5. DOI: 10.1109/NTIC55069.2022.10100581.
- [95] D. Schmalstieg and T. Hollerer. *Augmented Reality: Principles and Practice*. Addison-Wesley Professional, 2016. ISBN: 978-0321883575.
- [96] E. Schrödinger. “Die gegenwärtige Situation in der Quantenmechanik”. In: *Naturwissenschaften* 23 (48 1935), pages 807–812.
- [97] D. Scientist. *Redes U-Net*. Accedido el 14 de agosto de 2023. 2023. URL: <https://datascientest.com/%7D>.

- [98] S. Shah and J. Aggarwal. “Intrinsic parameter calibration procedure for a (high-distortion) fish-eye lens camera with distortion model and accuracy estimation*”. In: *Pattern Recognition* 29.11 (1996), pages 1775–1788. ISSN: 0031-3203. DOI: [https://doi.org/10.1016/0031-3203\(96\)00038-6](https://doi.org/10.1016/0031-3203(96)00038-6).
- [99] Q. Shan, J. Jia, and A. Agarwala. “High-quality motion deblurring from a single image”. In: *SIGGRAPH*. 2008.
- [100] L. G. Shapiro and G. C. Stockman. *Computer Vision*. New Jersey: Prentice-Hall, 2001, pages 279–325. ISBN: 0-13-030796-3.
- [101] V. Shende, A. Prasad, I. Markov, and J. Hayes. “Synthesis of reversible logic circuits”. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 22.6 (2003), pages 710–722.
- [102] V. V. Shende, S. S. Bullock, and I. L. Markov. “Synthesis of quantum-logic circuits”. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 25.6 (2006), pages 1000–1010. DOI: <https://doi.org/10.1109/TCAD.2005.855930>.
- [103] P. W. Shor. “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer”. In: *SIAM Review* 41.2 (1999), pages 303–332. DOI: 10.1137/S0036144598347011. URL: <https://doi.org/10.1137/S0036144598347011>.
- [104] V. Singh. “Sunflower leaf diseases detection using image segmentation based on particle swarm optimization”. In: *Artificial Intelligence in Agriculture* 3 (2019), pages 62–68. ISSN: 2589-7217. DOI: <https://doi.org/10.1016/j.aiia.2019.09.002>.
- [105] R. P. Sishodia, R. L. Ray, and S. K. Singh. “Applications of Remote Sensing in Precision Agriculture: A Review”. In: *Remote Sensing* 12.19 (2020). ISSN: 2072-4292. DOI: 10.3390/rs12193136. URL: <https://www.mdpi.com/2072-4292/12/19/3136>.
- [106] M. Sonka, V. Hlavac, and R. Boyle. *Machine Vision: Algorithms, Architectures, and Systems*. Bellingham, WA, USA: SPIE Press, 1999. ISBN: 978-0819436134.
- [107] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. 4th edition. Cengage Learning, 2014. ISBN: 978-0495082527.
- [108] T. Souto. “Métodos para evaluar efectividad en el control de malezas”. In: *Revista Mexicana de la Ciencia de la Maleza* (2000), pages 25–35.
- [109] J.-L. Starck, F. Murtagh, and A. Bijaoui. *Astronomical Image and Data Analysis*. Berlin, Heidelberg: Springer, 1998. ISBN: 978-3540410635.
- [110] J. Su, X. Guo, C. Liu, S. Lu, and L. Li. “An improved novel quantum image representation and its experimental test on IBM quantum experience”. In: *Sci Rep* 11 (1 2021). DOI: [doi:10.1038/s41598-021-93471-7](https://doi.org/10.1038/s41598-021-93471-7).
- [111] Y. Sun et al. “Double-Threshold Segmentation of Panicle and Clustering Adaptive Density Estimation for Mature Rice Plants Based on 3D Point Cloud”. In: *Electronics* 10 (Apr. 2021), page 872. DOI: 10.3390/electronics10070872.
- [112] M. Tripsas and G. Gavetti. “Capabilities, cognition, and inertia: Evidence from digital imaging”. In: *The SMS Blackwell Handbook of Organizational Capabilities* (2017), pages 393–412.

- [113] D. Turner, A. Lucieer, Z. Malenovský, D. H. King, and S. A. Robinson. “Spatial Co-Registration of Ultra-High Resolution Visible, Multispectral and Thermal Images Acquired with a Micro-UAV over Antarctic Moss Beds”. In: *Remote Sensing* 6.5 (2014), pages 4003–4024. ISSN: 2072-4292. DOI: 10.3390/rs6054003.
- [114] L. Wang, Z. Deng, and W. Liu. “An improved two-threshold quantum segmentation algorithm for NEQR image”. In: *Quantum Information Processing* 21.8 (2022), pages 1–20. DOI: 10.1007/s11128-022-03624-4.
- [115] P. Webster, S. D. Bartlett, and D. Poulin. “Reducing the overhead for quantum computation when noise is biased”. In: *Phys. Rev. A* 92 (6 Dec. 2015), page 062309. DOI: 10.1103/PhysRevA.92.062309. URL: <https://link.aps.org/doi/10.1103/PhysRevA.92.062309>.
- [116] J. Weier and D. Herring. *Measuring Vegetation (NDVI & EVI)*. NASA Earth Observatory, Washington DC. 2000.
- [117] *What is eCognition?* 2022. URL: <https://geospatial.trimble.com/what-is-ecognition>.
- [118] C. P. Williams. *Explorations in Quantum Computing*. 2nd. Springer Publishing Company, Incorporated, 2008. ISBN: 184628886X.
- [119] D. Williams et al. “A method for automatic segmentation and splitting of hyperspectral images of raspberry plants collected in field conditions”. In: *Plant Methods* 13 (1 2017), pages 1–13. DOI: 10.1186/s13007-017-0226-y.
- [120] H. Xia, H. Li, H. Zhang, Y. Liang, and J. Xin. “Novel multi-bit quantum comparators and their application in image binarization”. In: *Quantum Information Processing* 18.7 (2019), pages 1–17. DOI: <https://doi.org/10.1007/s11128-019-2334-2>.
- [121] F. Yan, I. Abdullah, and P. Le. “Quantum image processing: A review of advances in its security technologies”. In: *International Journal of Quantum Information* 15.3 (2017). DOI: 10.1142/S0219749917300017.
- [122] F. Yan and S. Venegas-Andraca. *Quantum Image Processing*. Springer. 2020. DOI: 10.1007/978-981-32-9331-1.
- [123] F. Yan et al. “A parallel comparison of multiple pairs of images on quantum computers”. In: *International Journal of Innovative Computing and Applications* 5.4 (2013). PMID: 62955, pages 199–212. DOI: 10.1504/IJICA.2013.062955. eprint: <https://www.inderscienceonline.com/doi/pdf/10.1504/IJICA.2013.062955>. URL: <https://www.inderscienceonline.com/doi/abs/10.1504/IJICA.2013.062955>.
- [124] L. Yan, R. Zhu, L. Mi, C. Yihui, and D. Yao. “Segmentation of White Blood Cell from Acute Lymphoblastic Leukemia Images Using Dual-Threshold Method”. In: *Computational and Mathematical Methods in Medicine* 2016 (Jan. 2016), pages 1–12. DOI: 10.1155/2016/9514707.
- [125] G. Yang, X. Song, W. N. N. Hung, M. A. Perkowski, and C.-J. Seo. “Synthesis of reversible circuits with minimal costs”. In: *Calcolo* 45 (3 2008), pages 193, 206.
- [126] L. Yang, L. R. Mansaray, J. Huang, and L. Wang. “Optimal Segmentation Scale Parameter, Feature Subset and Classification Algorithm for Geographic Object-Based Crop Recognition Using Multisource Satellite Imagery”. In: *Remote Sensing* 11.5 (2019). ISSN: 2072-4292.

- [127] S. Yuan, C. Wen, B. Hang, and Y. Gong. “The dual-threshold quantum image segmentation algorithm and its simulation”. In: *Quantum Inf. Process.* 19.12 (2020), page 425. DOI: 10.1007/s11128-020-02932-x. URL: <https://doi.org/10.1007/s11128-020-02932-x>.
- [128] S. Yuan et al. “Quantum morphology operations based on quantum representation model”. In: *Quantum Information Processing* 14.5 (2015), pages 1625–1645. ISSN: 1573-1332. DOI: 10.1007/s11128-014-0862-3. URL: <https://doi.org/10.1007/s11128-014-0862-3>.
- [129] F. Zhang and J. Chen. “Optimizing T gates in Clifford+T circuit as $\pi/4$ rotations around Paulis”. In: *arXiv: Quantum Physics* (2019).
- [130] Y. Zhang, K. Lu, Y. Gao, and M. Wang. “NEQR: a novel enhanced quantum representation of digital images”. In: *Quantum Information Processing* 12.8 (2013), pages 2833–2860. DOI: 10.1007/s11128-013-0567-z.
- [131] Z. Zhang. “A Flexible New Technique for Camera Calibration”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.11 (2000), pages 1330–1334.
- [132] Y. Zhong, A. Ma, Y. soon Ong, Z. Zhu, and L. Zhang. “Computational intelligence in optical remote sensing image processing”. In: *Applied Soft Computing* 64 (2018), pages 75–93. ISSN: 1568-4946. DOI: <https://doi.org/10.1016/j.asoc.2017.11.045>. URL: <https://www.sciencedirect.com/science/article/pii/S1568494617307081>.
- [133] N.-R. Zhou, L.-X. Huang, L.-H. Gong, and Q.-W. Zeng. “Novel quantum image compression and encryption algorithm based on DQWT and 3D hyper-chaotic Henon map”. In: *Quantum Information Processing* 19.9 (2020), page 284. ISSN: 1573-1332. DOI: 10.1007/s11128-020-02794-3. URL: <https://doi.org/10.1007/s11128-020-02794-3>.
- [134] X. Zhou, D. W. Leung, and I. L. Chuang. “Methodology for quantum logic gate construction”. In: *Phys. Rev. A* 62 (5 Oct. 2000), page 052316. DOI: 10.1103/PhysRevA.62.052316. URL: <https://link.aps.org/doi/10.1103/PhysRevA.62.052316>.

Universidad de Almería



DEPARTAMENTO
DE INFORMÁTICA
UNIVERSIDAD DE ALMERÍA