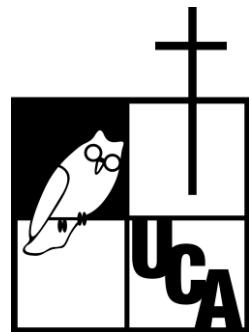


UNIVERSIDAD CENTROAMERICANA  
JOSÉ SIMEÓN CAÑAS



SEGMENTACIÓN AUTOMÁTICA DE PARCELAS AGRÍCOLAS CON  
INTELIGENCIA ARTIFICIAL EN DATOS SATELITALES

TRABAJO DE GRADUACIÓN PREPARADO PARA LA  
FACULTAD DE INGENIERÍA Y ARQUITECTURA

PARA OPTAR AL GRADO DE  
INGENIERO INFORMÁTICO

POR:

EDITH YAMILÉT LÓPEZ TOBAR  
MIGUEL ÁNGEL MESQUITA PORTILLO  
ALEJANDRO ENRIQUE PINTO SANTOS  
JAIME ORLANDO SERRANO RIVAS

JULIO 2023,  
ANTIGUO CUSCATLÁN, EL SALVADOR, C.A.

RECTOR

ANDREU OLIVA DE LA ESPERANZA, S.J.

SECRETARIA GENERAL

LIDIA GABRIELA BOLAÑOS TEODORO

DECANO DE LA FACULTAD DE INGENIERÍA Y ARQUITECTURA

CARLOS ERNESTO RIVAS CERNA

DIRECTOR DE LA CARRERA DE INGENIERÍA INFORMÁTICA

JOSÉ ENMANUEL AMAYA ARAUJO

DIRECTOR DEL TRABAJO

METZI RUTILIA AGUILAR MUNGUA

LECTOR

NAPOLEÓN EDUARDO CORNEJO BOLAÑOS

## **DEDICATORIA**

Quiero expresar mi más profundo agradecimiento a todas aquellas personas que han sido parte fundamental en el proceso de realización de esta tesis. En primer lugar, quiero dar gracias a Dios, mi guía durante todo este tiempo y que me ha acompañado en cada paso de este camino.

A mis padres, quienes han sido mi mayor apoyo y ejemplo de perseverancia y dedicación. Gracias por su constante aliento, paciencia y sacrificio, sin su amor incondicional y su confianza en mí este logro no hubiera sido posible. A ustedes les debo todo lo que soy, y siempre estaré agradecida por su presencia y apoyo en mi vida. Desde el inicio de mi educación, ustedes han estado a mi lado, alentándome y creyendo en mí, han sido mi mayor motivación para esforzarme y superar cualquier obstáculo que se me ha presentado. Su confianza en mis habilidades es lo que siempre me ha empujado a perseguir mis sueños. Agradezco profundamente por cada sacrificio que han hecho por mí, han trabajado incansablemente para brindarme las mejores oportunidades y para asegurarse de que no me falte nada. Les agradezco por garantizar mi bienestar y mi crecimiento personal y no hay palabras suficientes para expresar cuánto los amo y cuánto significan para mí. Este logro, esta tesis, es tan suyo como mío.

A la Ingeniera Metzi y al Ingeniero Napoleón, quienes con su experiencia y conocimiento nos han guiado y orientado en cada etapa de este proyecto. Sus consejos y enseñanzas han sido fundamentales en el desarrollo de este. Agradezco su tiempo, dedicación y el esfuerzo que han puesto en ayudarme a alcanzar mis metas.

A mis amigos y seres queridos, quienes han estado a mi lado brindándome su apoyo y palabras de aliento en los momentos de duda y dificultad. Gracias por comprender mis ausencias y por celebrar mis logros. Su presencia en mi vida ha sido invaluable y me siento bendecida por tenerlos a mi lado. Y a mis abuelitos que sé que están orgullosos desde el cielo viendo lo que he logrado.

Y por último, mi más profundo agradecimiento a mis compañeros de tesis: Jaime, Pinto y Migue. No solo hemos compartido el desafío intelectual que ha sido nuestra tesis, sino que también durante estos casi 6 años hemos forjado una amistad extraordinaria y duradera. A través de estos años se convirtieron en mis amigos más cercanos, y por eso me siento feliz de haber logrado esto junto a ustedes. Gracias por confiar en mis habilidades y por haber tenido más fe en mí que la que yo misma me tenía en momentos difíciles de la carrera. Y, aunque esta tesis llegue a su fin, sé que nuestra amistad perdurará en el tiempo, de verdad espero que sigamos apoyándonos mutuamente en nuestros futuros proyectos y celebrando juntos nuestros éxitos.

## **DEDICATORIA**

Primero que nada, dedicarle este trabajo de tesis a Dios y la Virgen de Guadalupe, definitivamente desde el día en que hice el examen de admisión para entrar a la Universidad puse en sus manos esta carrera. Recuerdo día a día decirles: “Que todo sea su voluntad”. Durante el camino enfrenté duras adversidades, obstáculos, dos materias reprobadas; pero con su ayuda me supe levantar, seguir luchando, nunca dejar de creer en mí y ahora que estoy más cerca que nunca del objetivo principal solo puedo estar eternamente agradecido porque sé que sin su ayuda la historia tal vez sería distinta. Segundo, a mis padres, en especial a mi madre, la cual desde niño me enseñó una gran lección: “Todo en esta vida cuesta, hay que luchar y poner de nuestra parte para lograr nuestros sueños”. A pesar de que ella dejó de trabajar justo cuando comencé la carrera, hasta la fecha nunca dejó de apoyarme, empujarme a lograr mis metas; definitivamente solo ella sabe cuánto ha luchado por ver a sus hijos triunfar, se merece más que nadie una alegría como esta, y espero dársela pronto. Luego, a mi “yo” de 2018, que veía muy lejos este momento y creía que no lo iba a lograr: el momento ha llegado y estamos a unos pasos de lograrlo. A mi “yo” del presente: todo ha valido la pena, el esfuerzo de todos estos años está rindiendo frutos y debo estar orgulloso de todo el recorrido, ha sido simplemente increíble. A mi novia, un pilar fundamental en mi vida, que cuando a mitad del camino me desanimé fue quien me levantó y me llenó de fuerzas para culminar la carrera y ahora, a terminar todo el proceso hasta graduarme. Y la cual ahora me ha hecho parte de su familia, a la cual también hago parte de mis alegrías y logros, mi segunda familia. Ahora mis logros son de ella también y de nuestro amado hijo, el cual apareció en nuestras vidas al final del trabajo de tesis. A la Mtra. Metzi Aguilar y el Ing. Napoleón Cornejo, directora del trabajo de tesis y encargado del proyecto respectivamente, por su apoyo durante todo el proceso y su amplio conocimiento. A los catedráticos que lograron sembrar su conocimiento en mí y que ahora pongo en práctica no solo en este trabajo de tesis sino también en mi vida personal y laboral. A todos los amigos, porque no son compañeros, que hice en estos cinco años de carrera, gracias infinitas por tantas alegrías y buenos momentos, las risas jamás faltaron, me los llevo en mi corazón y sé que muchas amistades continuarán fuera de la universidad. A mis amigos de infancia, los cuales también se encuentran a las puertas de culminar sus respectivas carreras, espero ser un claro ejemplo de que ellos también pueden y sé que lo lograrán. A Luis, mi gran amigo, a quién le prometí mi título de ingeniero, sé que desde el cielo estará celebrando conmigo y estará muy orgulloso de mi. Finalmente, a Edith, a “Pinto”, y a Jaime. Su calidad como personas es indiscutible, la comunicación, el trabajo en equipo y sobre todo el apoyo mutuo tanto académico como humano ha sido lo más valioso de toda esta experiencia. Que alegría, ilusión y placer poder terminar la carrera juntos, hacer la tesis juntos y si todo sale bien, graduarnos juntos.

- *Miguel Mesquita*

## **DEDICATORIA**

Quiero comenzar mi dedicatoria agradeciendo a Dios. Todos mis logros, éxitos, y metas alcanzadas se las debo a él, porque nunca ha dejado de darme bendiciones en mi vida, me ha mantenido siempre en el camino de su palabra, con valores y principios fuertes. Este gran hito en mi vida es una obra más de su bondad y amor conmigo. Quiero agradecer y dedicar este trabajo a mis amados padres, Dimas Pinto y Aracely Santos, quienes me lo han dado todo, me han criado con amor, estando presentes en todos los momentos importantes en mi vida. Esta es la culminación de años de esfuerzo y trabajo, para darme la mejor educación. Gracias por inspirarme y animarme a ser un buen profesional, y una persona de bien, así como lo son ustedes. Gracias, mamá y papá por estar siempre a mi lado, por transmitirme todo lo que han aprendido en sus vidas, por guiarme, cuidarme, y darme siempre la mejor educación, para que así pueda, con todo lo que he aprendido, hacer un bien al mundo, y ser feliz. A mi hermano mayor Rodrigo Pinto, quien al tener siete años de diferencia conmigo, y recorrer el mismo camino antes que yo, siempre se ha asegurado de guiarme, aconsejarme, motivarme, y transmitirme todo el conocimiento que él sabía que me sería útil para mi futuro. Gracias por entenderme cuando tuve dudas de mi futuro, de ayudarme a superar el miedo, y seguir siempre adelante. A mi novia Diana Klee, quién ha estado a mi lado apoyándome y dándome ánimos en momentos difíciles, y celebrando juntos los momentos alegres, quien hoy en día es un pilar importante en mi vida, y me motiva siempre a dar la mejor versión de mí mismo. A mis compañeros de tesis, y muy queridos amigos Jaime, Edith y Migue, gracias por todo el trabajo hecho, y por confiar en mí en todo momento. Gracias por los lazos de amistad forjados durante todos los años de carrera, los cuales deseo que se mantengan durante toda la vida. Juntos hemos aprendido mucho durante todo este tiempo, y no puedo hacer más que agradecer y sentirme orgulloso de ustedes, y del trabajo que hemos realizado. A Metzi y Napoleón, quienes nos asesoraron durante este proceso, y nos han transmitido mucho de su conocimiento, y se tomaron el tiempo de asegurarse que entendiéramos estos temas, que eran completamente nuevos para nosotros al momento de iniciar esta investigación. Quiero agradecer de forma especial también, a mis entrenadores de ajedrez, Gustavo Zelaya y Lemnys Arias, quienes durante los más de 8 años que estuve entrenado semana tras semana en esta disciplina me transmitieron los valores del deporte, el esfuerzo, la superación, la dedicación, y el hambre de perseguir mis objetivos. Todo lo que aprendí con ellos, lo he aplicado durante toda mi vida personal, y vida académica. El impacto que tuvo el ajedrez, mis entrenadores, y todos mis compañeros ajedrecistas fue tan grande, que simplemente soy incapaz de imaginar cómo sería mi vida hoy en día, si todo lo que viví durante todos esos años. Por último, dedico esto a mi yo de hace 5 años. Todo el esfuerzo valió la pena, Alejandro, todos los momentos complicados fueron superados, y hemos llegado hasta aquí. Seguimos siempre viendo hacia adelante.

- Alejandro Pinto

## DEDICATORIA

Primero que todo dedicar el presente trabajo a Dios y a la Virgen María, a quienes desde el inicio deje en sus manos todo lo que cursar mis estudios universitarios significaba, y que han intercedido por mi desde el primer día hasta el último. Siempre estaré agradecido por la bendición que se me fue dada al poder cursar y aprobar cada una de mis materias, aprobar cada evaluación y completar cada trabajo, al igual que estoy y estaré agradecido por todas las fuerzas y motivación que me brindaron incluso en los momentos en los que estaba muy cansado y ya no quería continuar. Mi corazón siempre estará con Dios y la Virgencita, espero poder seguir agradeciéndoles hoy y siempre. También dedicarle a mi papá y a mi mamá, Orlando y Evelin, a quienes amo con toda mi alma y corazón, y me apoyaron y motivaron para siempre salir adelante y dar lo mejor de mí en todo lo que me proponga a lograr, su apoyo incondicional siempre ha estado presente y para mi ellos son uno de los grandes pilares por las que me esfuerzo por mejorar día con día tanto como profesional como persona, y que sin ellos no sería todo lo que soy hoy en día. A mis hermanos, Taty y Diego, mi fuente de inspiración y de quienes día con día aprendo algo nuevo, me ayudan a superar mis miedos y me apoyan y alientan a nunca rendirme y a seguir adelante siendo como soy en busca de mejores oportunidades tanto para mí como para las personas que me rodean. Agradecer también a mis amigos y compañeros de tesis, *Pinto*, Edith y *Migue*, con quienes he disfrutado desde el primer momento su compañía y apoyo antes, durante y después de la elaboración de esta investigación, los considero a todos personas excepcionales, cada uno con muchas virtudes y cualidades que les hacen ser los mejores compañeros de investigación que se pueden llegar a pedir, y que sé que están destinados a lograr grandes cosas y cumplir todos sus sueños y metas que se propongan, para mí ha sido un honor trabajar junto a personas tan increíbles, y espero que este no sea un final para trabajar a su lado, sino el inicio de nuevas oportunidades donde podremos encontrarnos y seguir trabajando juntos. A la Ingeniera Metzi y al Ingeniero Napoleón, quienes nos compartieron sus conocimientos y nos brindaron apoyo durante todo el desarrollo de la investigación, que aun siendo nosotros novatos en muchos temas, nos guiaron de la mejor manera posible lograr el cometido, por lo que estamos muy agradecidos y honrados de haber tenido la oportunidad de trabajar junto a dos grandes profesionales en la materia. A mi amada mascota, mi querido perrito Luka, quien enfermó durante el desarrollo de la investigación pero que aún sigue fuerte y hasta el día de hoy luchando contra su enfermedad, jamás deje de pensar en él mientras desarrollaba la investigación, y estoy feliz y agradecido de poder tenerlo conmigo hasta la fecha. Por último, al Jaime de hace 5 años, quien en busca de nuevas oportunidades y herramientas que puedan ser utilizadas para volver aunque sea un poco menos gris el mundo de locura en el que vivimos, jamás se rindió y siempre siguió adelante, esforzándose al máximo y ayudando a quien le fuera posible en el camino.

## **RESUMEN**

El desarrollo del presente trabajo de investigación consiste en la creación de un marco de referencia en El Salvador aplicado al tema de “Segmentación Automática de Parcelas Agrícolas con Inteligencia Artificial en Datos Satelitales”, el cual se enfoca específicamente en el municipio de Santa María, departamento de Usulután. A partir de un conjunto de datos (imágenes satelitales) de parcelas agrícolas de la zona del municipio ya mencionado y aplicando algoritmos de Inteligencia Artificial orientados a la segmentación de imágenes y detección de bordes; se busca obtener el conjunto de imágenes satelitales con todas las parcelas agrícolas de la zona debidamente segmentadas, de modo que cada terreno se diferencie de los demás a partir de sus respectivos límites.

El desarrollo de toda la investigación, incluyendo la implementación, desarrollo de los modelos y algoritmos a aplicar en el proyecto consta de siete capítulos; los cuales se desglosan en una parte teórica-metodológica, una parte práctica-experimental en la que se detalla toda la implementación de los algoritmos, y finalmente el capítulo de conclusiones y recomendaciones finales de toda la investigación.

En la parte teórica-metodológica se detallan los objetivos generales y específicos de la investigación, los alcances-limitaciones de esta y antecedentes de proyectos o trabajos de investigación relacionados al tema en cuestión. De igual forma, se pone en contexto al lector con el marco teórico, donde se explica a profundidad los temas-conceptos aplicados en la investigación y que respaldan el trabajo. Se explica todo lo relacionado a temas de Inteligencia Artificial (IA), Machine y Deep Learning (ML y DL), algoritmos de segmentación de imágenes, algoritmos de detección de bordes, entre otros temas de interés.

En la parte modular del trabajo de investigación, se expone la metodología del trabajo. En esta se explica las herramientas tanto de apoyo (herramientas o repositorios en la nube, aplicaciones de control de flujos de trabajo) como de desarrollo (editores de código, lenguajes de programación, librerías de desarrollo, entre otros) aplicadas en la investigación y se abre paso al primer acercamiento de los modelos y algoritmos a implementar en el trabajo, se explica de forma detallada el funcionamiento de estos algoritmos y se deja preparado al lector para los capítulos posteriores. Los siguientes capítulos son la parte fundamental de la investigación, en estos se explica a detalle todo el proceso de limpieza de la información, preprocesamiento de datos, entrenamiento de los datos aplicando ya los algoritmos seleccionados y la puesta en práctica (entrenamiento y prueba) de los algoritmos encargados de la segmentación de imágenes, detección de bordes y posterior generación de predicciones con los resultados obtenidos.

En la parte final del trabajo, los capítulos finales explican todo el proceso de prueba de los algoritmos y los resultados obtenidos de cada uno. Con ello se procede a la parte de análisis de los resultados obtenidos y una comparativa de la eficiencia en los resultados que cada algoritmo mostró al final de todo el proceso de entrenamiento; esto permite que se puedan generar conclusiones respecto a los algoritmos implementados, la eficiencia de los algoritmos en cuanto a los resultados obtenidos y posibles recomendaciones a aplicar en futuras investigaciones que pretendan retomar la presente investigación. Para más información y comprensión lectora, se provee en la parte final del trabajo una sección de bibliografía y referencias bibliográficas, las cuales sirvieron de apoyo para el desarrollo de este trabajo de investigación.

# ÍNDICE

RESUMEN .....	I
ÍNDICE DE FIGURAS .....	V
ÍNDICE DE TABLAS .....	VIII
SIGLAS IX	
CAPÍTULO 1. INTRODUCCIÓN .....	1
1.1. Definición del Problema .....	1
1.2. Objetivos .....	2
1.2.1. Objetivo General.....	2
1.2.2. Objetivos Específicos.....	2
1.3. Límites y Alcances .....	2
1.3.1. Límites .....	2
1.3.2. Alcances.....	3
1.4. Antecedentes .....	3
1.5. Estado del Arte .....	4
CAPÍTULO 2. MARCO TEÓRICO .....	7
2.1 Agricultura: Segmentación de Parcelas Agrícolas.....	7
2.1.1 Aplicaciones de la segmentación de parcelas agrícolas.....	7
2.2. Imágenes Satelitales .....	8
2.2.1 Teledetección.....	8
2.2.2. Obtención de Imágenes Satelitales .....	9
2.2.3. Imágenes Satelitales: Bandas Espectrales e Índices de Vegetación .....	10
2.2.4. Aplicación de Imágenes Satelitales .....	12
2.3. Inteligencia Artificial: Machine Learning.....	13
2.3.1. Machine Learning .....	13
2.3.2. Aprendizaje Supervisado.....	14
2.3.3. Aprendizaje No Supervisado.....	15
2.3.4. Aprendizaje Reforzado.....	15
2.3.5. Aprendizaje Profundo .....	16
2.4. Metodologías de Segmentación de Parcelas Agrícolas .....	17
2.4.1 Algoritmos utilizados para la segmentación de parcelas agrícolas .....	17
2.4.2 Implicaciones, ventajas y desventajas de los algoritmos de segmentación.....	19
CAPÍTULO 3. METODOLOGÍA .....	21
3.1 Fuentes de Datos y Procesamiento .....	21
3.1.1 Selección, Obtención y Clasificación de Datos Satelitales .....	21
3.1.2. Preprocesamiento de Datos .....	29
3.2. Metodologías Aplicadas para la Detección de Bordes .....	34
3.2.1. Algoritmo de Canny .....	34
3.2.2. Red Neuronal .....	35

3.3. Análisis de Resultados: Métricas de Análisis de Datos.....	39
3.3.1. Precisión y Exhaustividad (Precision and Recall).....	39
3.3.2. Puntuación F1 (F1 Score).....	40
3.4. Herramientas a Utilizar.....	41
3.4.1. Herramientas de desarrollo .....	41
3.4.2. Herramientas de apoyo.....	44
CAPÍTULO 4. DESARROLLO Y CONSTRUCCIÓN DE METODOLOGÍAS PARA SEGMENTACIÓN DE PARCELAS .....	45
4.1. Construcción y Desarrollo de Metodologías .....	45
4.1.1 Algoritmo de Canny .....	45
4.1.2. Red Neuronal.....	48
CAPÍTULO 5. PRUEBAS Y ANÁLISIS DE RESULTADOS .....	56
5.1. Algoritmos de Canny .....	56
5.2. Red Neuronal .....	68
CAPÍTULO 6. CONCLUSIONES Y RECOMENDACIONES .....	76
6.1. Conclusiones.....	76
6.2. Recomendaciones.....	77
GLOSARIO.....	80
REFERENCIAS.....	83
ANEXO A.....	87
ANEXO B.....	109
ANEXO C.....	110
ANEXO D .....	113
ANEXO E.....	117
ANEXO F .....	118
ANEXO G.....	125

## ÍNDICE DE FIGURAS

Fig 1. Geografía de El Salvador (Obtenido de Wikipedia).....	21
Fig 2. Área de Estudio: Municipio de Santa María, Usulután (Elaboración propia). .....	22
Fig 3. Geografía de los Países Bajos (Obtenido de Wikipedia).....	22
Fig 4. Área de Estudio: Región de Zelanda, Países Bajos (Elaboración propia). .....	23
Fig 5. Esquema de Modelo de Datos mejorado para el formato NetCDF (Fuente: Unidata. Data Services and Tools for Geoscience).....	24
Fig 6. Estructura de Cubos de Datos Bajo el Formato NetCDF (Elaboración Propia).....	25
Fig 7. <i>Ground Truth</i> (Terreno Real) para Santa María, Usulután, El Salvador (Elaboración Propia)...	28
Fig 8. Porción de <i>Ground Truth</i> (Terreno Real) para Zelanda, Países Bajos (Elaboración Propia). 28	
Fig 9. Ejemplo de Mejora de Imagen tras aplicar la Calibración Radiométrica (Fuente: IDEAM, Instituto de Hidrología, Metereología y Estudios Ambientales).....	30
Fig 10. Máscara de Bordes Reales para Santa María, Usulután, El Salvador (Elaboración Propia). ..33	
Fig 11. Máscara de Bordes Reales para Zelanda, Países Bajos (Elaboración Propia). .....	33
Fig 12. Metodología para Aplicación del Algoritmo de Canny.....	35
Fig 13. Áreas de Estudio: Municipio de Santa María, Usulutan, El Salvador, y región de Zelanda, Países Bajos (Elaboración Propia). .....	38
Fig 14. Metodología para Aplicación de la Red Neuronal Convolutional .....	38
Fig 15. Importación de paquetes utilizados para la construcción del algoritmo. ....	45
Fig 16. Lectura del archivo con la información del municipio de Santa María.....	45
Fig 17. Toma de banda B08 del cubo de datos.....	45
Fig 18. Normalización del valor de los píxeles de la banda escogida. ....	46
Fig 19. Aplicación del suavizado Gaussiano.....	46
Fig 20. Aplicación del algoritmo de Canny .....	46
Fig 21. Resultado de la detección de bordes con el algoritmo de Canny .....	46
Fig 22. Búsqueda de contornos.....	47
Fig 23. Aplicación de contornos sobre la imagen original.....	47
Fig 24. Resultados y comparación del mapa con bordes. ....	48

Fig 25. Importación de paquetes para el desarrollo de la Red Neuronal Convolutacional.....	49
Fig 26. Obtención de cubos de datos y máscaras de parcelas reales.....	49
Fig 27. Obtención de valores de fechas de análisis para ambas áreas de estudio .....	50
Fig 28. Definición de variables de apoyo: Dimensiones de cubos de datos y de imágenes de análisis . .....	50
Fig 29. Cálculo de índices de vegetación .....	51
Fig 30. Sustitución de pixeles cubiertos de nubes.....	51
Fig 31. Obtención de máscaras de bordes de entrenamiento .....	52
Fig 32. Construcción de listas cubos y máscaras de entrenamiento .....	52
Fig 33. Combinación de bandas para construcción de imagen de entrenamiento .....	53
Fig 34. Definición de bloque de convolución.....	53
Fig 35. Definición de bloque de reducción .....	54
Fig 36. Definición de bloque de muestra ascendente .....	54
Fig 37. Construcción de modelo de red neuronal.....	55
Fig 38. Creación de sets de entrenamiento y pruebas.....	55
Fig 39. Banda B04 (Izquierda) y Banda B08 (Derecha) utilizadas como entrada del algoritmo de Canny para Santa María, Usulután .....	56
Fig 40. Resultados del Algoritmo de Canny bajo diferentes umbrales en la banda B04 en Santa María: a) Máscara Real, b) 30-90, c) 50-150, d) 90-30, e) 120-40, f) 100-300 .....	57
Fig 41. Resultados del Algoritmo de Canny bajo diferentes umbrales en la banda B08 en Santa María: a) Máscara Real, b) 30-90, c) 50-150, d) 90-30, e) 120-40, f) 100-300 .....	58
Fig 42. Resultados del Algoritmo de Canny sobre índices de vegetación de Santa María: a) NDVI, b) NDWI, c) SAVI.....	59
Fig 43. Gráfica comparativa de las métricas aplicadas en la banda B04 sobre el algoritmo de Canny en Santa María, Usulután.....	62
Fig 44. Grafica comparativa de las métricas aplicadas en la banda B08 sobre el algoritmo de Canny en Santa María, Usulután.....	62
Fig 45. Gráfica comparativa de las métricas aplicadas sobre los índices de vegetación sobre el algoritmo de Canny en Santa María, Usulután.....	62
Fig 46. Banda B04 (Izquierda) y Banda B08 (Derecha) utilizadas como entrada del algoritmo de Canny para Zelanda, Países Bajos.....	63

Fig 47. Resultados del Algoritmo de Canny bajo diferentes umbrales en la banda B04 en Zelanda: a) Máscara Real, b) 30-90, c) 50-150, d) 90-30, e) 120-40, f) 100-300 .....	64
Fig 48. Resultados del Algoritmo de Canny bajo diferentes umbrales en la banda B08 en Zelanda: a) Máscara Real, b) 30-90, c) 50-150, d) 90-30, e) 120-40, f) 100-30 .....	64
Fig 49. Resultados del Algoritmo de Canny sobre índices de vegetación en Zelanda: a) NDVI, b) NDWI, c) SAVI.....	65
Fig 50. Gráfica comparativa de las métricas aplicadas en la banda B04 sobre el algoritmo de Canny en Zelanda, Países Bajos .....	67
Fig 51. Gráfica comparativa de las métricas aplicadas en la banda B08 sobre el algoritmo de Canny en Zelanda, Países Bajos .....	67
.....	68
Fig 52. Gráfica comparativa de las métricas aplicadas en los índices de vegetación sobre el algoritmo de Canny en Zelanda, Países Bajos .....	68
Fig 53. Resultados del entrenamiento de la CNN en base a las imágenes satelitales de la región de Santa María, Usulután .....	69
Fig 54. Resultados de predicciones al modelo de CNN para Santa María, Usulután.....	70
Fig 55. Gráfica comparativa de las métricas aplicadas sobre la CNN en Santa María, Usulután .....	72
Fig 56. Resultados del entrenamiento de la CNN en base a las imágenes satelitales de la región de Zelanda, Países Bajos.....	73
Fig 57. Resultados de predicciones al modelo de CNN para Santa María, Usulután.....	73
Fig 58. Gráfica comparativa de las métricas aplicadas sobre la CNN en Zelanda, Países Bajos.....	75

## ÍNDICE DE TABLAS

Tabla 1. ....	Lista de Bandas de Teledetección del Satélite Sentinel-2 (Fuente: Sentinel Hub)	26
Tabla 2. ....	Fechas de Análisis de Imágenes Satelitales para Santa María, Usulután, El Salvador	27
Tabla 3. ....	Fechas de Análisis de Imágenes Satelitales para Zelanda, Países Bajos	27
Tabla 4. ....	Métricas de Resultados de Algoritmo de Canny sobre Santa María, Usulutan, El Salvador	59
Tabla 5. ....	Métricas de Resultados de Algoritmo de Canny sobre Zelanda, Países Bajos	65
Tabla 6. ....	Métricas de análisis de resultados sobre predicciones del modelo de CNN aplicado a Santa María, Usulután	71
Tabla 7. ....	Métricas de análisis de resultados sobre predicciones del modelo de CNN aplicado a Zelanda	74

## SIGLAS

API	Application Programming Interface
ARD	Analysis Ready Data
BOC	Reflectancia de la parte inferior atmósfera
CE	Error de comisión
CNN	Convolutional Neural Network
CNR	Centro Nacional de Registros
CO	Error de omisión
CPU	Central Processing Unit
EBS	Edges Based Segmentation
ED	Distancia Euclidiana
EDC	Euro Data Cube
ESA	Agencia Espacial Europea
FORCE	Framework for Operational Radiometric Correction for Environmental Monitoring)
GEOGLAM	Group on Earth Observations Global Agricultural Monitoring Initiative
GIS	Geographic Information Systems
GPU	Graphical Processing Unit
GSAA	Aplicación de ayuda geoespacial
IA	Inteligencia Artificial
LSTM	Memoria a largo y corto plazo
MAE	Error absoluto medio
ML	Machine Learning
MMC	Compuestos medios mensuales
MRS	Multi-Resolution Segmentation
MSAVI2	Índice de Vegetación Ajustado al Suelo Modificado
MTS	Multi-Threshold Segmentation
NASA	National Aeronautics and Space Administration
NDVI	Normalized Difference Vegetation Index
NDWI	Normalized Differential Water Index
NetCDF4	Network Common Data Form
NIR	Near Infrared Reflectance
OAD	Dataset basado en objetos agregados
OEA	Organización de Estados Americanos
PAD	Dataset de parches ensamblados

PR	Precisión-recuperación
QGIS	Quantum Geographic Information Systems
RAE	Real Academia de la lengua Española
RAor	Índice de sobre segmentación
RAos	Índice de infra segmentación
RBS	Region Based Segmentation
RCL	Capas convolucionales recurrentes
SAVI	Soil-adjusted vegetation index
SSH	Secure SHell
SWIR	Short Wave Infrared
TOA	Reflectancia de la parte superior atmósfera
TPU	Tensor Processing Unit
TT	Placas de prueba
WS	Watershed Segmentation

## CAPÍTULO 1. INTRODUCCIÓN

### 1.1. Definición del Problema

El catastro, según la Organización de los Estados Americanos (OEA, 2010), se define como un tipo de registro público que contiene la delimitación de las parcelas individuales, detalles relacionados a la tierra, información de propiedad y los derechos asociados a cada una de estas. Es decir, un censo y padrón estadístico de las fincas rústicas y urbanas de un territorio con su respectiva valoración.

En El Salvador, actualmente se cuenta con un catastro legal registrado en el Centro Nacional de Registros (CNR), sin embargo, este no se encuentra completo y tampoco actualizado, sobre todo en la zona rural, la cual representa la mayor zona de producción y progreso agrícola del país. Asimismo, cabe resaltar que tampoco existe un registro de uso de tierras en El Salvador. Todo esto trae consigo una problemática, y es que, sin un catastro completo es imposible establecer y separar el comportamiento espacio temporal de cada terreno y sus respectivos límites, lo cual es algo imprescindible. Esto ralentiza y vuelve poco práctico el proceso de análisis, control de límites y manejo de información de estas áreas agrícolas.

Dado que cada parcela cuenta con diferentes usos y diferentes propietarios, surge la necesidad de detallar, mediante geolocalización, los límites de cada parcela en el área rural de El Salvador. Ante esta problemática, surge la posibilidad de que, mediante el uso de tecnologías de teledetección y segmentación esta situación se pueda solucionar. La metodología de segmentación automática es posible gracias a la implementación de un sensado remoto satelital que utiliza algoritmos de Machine Learning, rama de la Inteligencia Artificial. Estos algoritmos ya existentes son entrenados mediante sets de datos previamente recolectados que permiten que estos puedan aprender y arrojar resultados lo más certeros posibles basados en patrones de aprendizaje y la calidad de la data a entrenar.

Se espera que los algoritmos devuelvan un conjunto de polígonos que determinan las fronteras de cada terreno en un área predeterminada. Con este producto final será posible ofrecer métricas individuales de cada terreno a sus respectivos propietarios, así como también insumos con los que podrán tomar acciones en pro de sus rendimientos agrícolas, favoreciendo la productividad agrícola en las zonas rurales de El Salvador, lo cual se traducirá en mejoras en el sector agrícola y mejoras en la seguridad alimentaria a nivel nacional.

## **1.2. Objetivos**

### **1.2.1. Objetivo General**

Implementar algoritmos que permitan la detección y segmentación de parcelas agrícolas, en ciertas zonas del territorio salvadoreño, y en zonas fuera de El Salvador con características similares al territorio nacional, de forma automática utilizando metodologías de teledetección, e implementación de Machine Learning.

### **1.2.2. Objetivos Específicos**

- Hacer una investigación y repaso de la literatura concerniente al uso de inteligencia artificial en imágenes satelitales para segmentación de terrenos.
- Proponer en base al estudio anterior, dos metodologías de teledetección a implementar.
- Recolectar y preparar un dataset para el entrenamiento de un modelo, esto mediante los algoritmos seleccionados: los límites geoespaciales de parcelas agrícolas en una región (en cualquier parte del mundo, idealmente similar a El Salvador) y sus datos satelitales en ARD (analysis ready data) para las metodologías propuestas.
- Entrenar el modelo, esto mediante los algoritmos seleccionados y con los datos recolectados y preparados en el punto anterior, e implementarlo en una región rural y agrícola en El Salvador para la evaluación de su desempeño.

## **1.3. Límites y Alcances**

### **1.3.1. Límites**

Dentro de las limitaciones que posee el desarrollo de la presente investigación, se tiene como primer punto que el objetivo del proyecto no se trata de comparar las diferentes metodologías y/o algoritmos de Machine Learning que existen actualmente.

De igual manera, a pesar de que la información obtenida para el entrenamiento de los modelos es de uso libre, posee como limitante el hecho de que no se contará con imágenes de alta definición. A su vez, las condiciones meteorológicas reducirán la calidad de la data, puesto que pueden interferir con el entrenamiento de los modelos.

### **1.3.2. Alcances**

El alcance se ha limitado solamente a las regiones escogidas para entrenar al modelo, no se pretende que el algoritmo resultante de este proyecto sea aplicable a todas las regiones del mundo. La información ocupada será tomada de los satélites Sentinel-2, los cuales son de uso libre.

De los muchos modelos que existen para la teledetección y separación de campos en imágenes satelitales, se ha delimitado la utilización de solo dos modelos. La investigación se centrará en la construcción y prueba de los modelos seleccionados para trabajar con la información seleccionada.

### **1.4. Antecedentes**

La segmentación de parcelas agrícolas es un tema muy conocido no solamente dentro del ámbito de las aplicaciones de la Inteligencia Artificial, sino que ha sido tema de investigación y trabajo desde muchísimo tiempo atrás, puesto que la agricultura ha sido de los pilares más importantes tanto para la producción de alimento como para la economía principal de muchos países, en especial dentro de la región Latinoamericana.

El uso de la Inteligencia Artificial ha contribuido enormemente al desarrollo de actividades que tradicionalmente tomarían una mayor cantidad de tiempo en ser terminadas. Gracias a la gran capacidad de manejo y análisis de datos que presenta el uso de la Inteligencia Artificial, algoritmos de todo tipo son utilizados para la automatización de procesos dentro del sector agrícola, como detección de malas hierbas, control de plagas y segmentación de parcelas destinadas para el cultivo. (Liakos, et al, 2018)

Los modelos de teledetección y/o clasificación desarrollados mediante la utilización de algoritmos de aprendizaje automático no son nada nuevo dentro del área de la agricultura, es posible encontrar múltiples trabajos e investigaciones relacionadas a aplicaciones en la resolución de problemas dentro de este sector. En el año 2019, Barry Watkins y Adriaan Niekerk del Departamento de Geografía y Estudios Medioambientales de la Universidad de Stellenbosch en Sudáfrica, realizaron una investigación centrada en la evaluación de diferentes metodologías de observación de la tierra para la delimitación automática de parcelas en base a imágenes multitemporales provenientes del satélite Sentinel-2, durante la temporada de cultivo. La investigación se basó en la combinación de algoritmos de detección de bordes (Canny y Scharr)

y de segmentación de imagen (watershed, multi-threshold y multiresolución) (Watkins, et al, 2019).

De igual manera, en 2018, un grupo de científicas argentinas que participaron del programa “AI for Earth” de Microsoft, se adentraron en la tarea del desarrollo y creación de un mapa de uso y cobertura del suelo para el Valle Inferior del Río Chubut, zona que basa su sostenibilidad principalmente en sectores como la ganadería y la agricultura, por lo que la asignación y control organizado de tierras se volvió una actividad de alta prioridad. La investigación se centró en el desarrollo de un nuevo modelo bautizado “SatNet”, una nueva arquitectura de Deep Learning que se encargaría de clasificar el uso y cobertura de la tierra, en base a imágenes conseguidas del satélite Sentinel-2, realizando clasificaciones a nivel de píxel basadas en una densa e interconectada red neuronal (Garcia, 2017).

Por otro lado, integrantes del grupo “Group on Earth Observations Global Agricultural Monitoring Initiative” (GEOGLAM), llevaron a cabo una investigación en el año 2021 relacionada a la evaluación de la delimitación de campos agrícolas mediante el uso de imágenes satelitales provenientes del Sentinel-1 y Sentinel-2. Para ello, se hizo uso de algoritmos como la segmentación multi-resolución para poder obtener un set de información óptimo para el análisis posterior, centrándose en determinar la precisión en el segmentado de los campos delimitados (Tetteh, et al, 2021).

A pesar de que el área de estudio y/o aplicación de algoritmos de Machine Learning y segmentación de parcelas agrícolas aparece en múltiples investigaciones de diferentes partes del mundo realizadas con el pasar de los años, dentro del sector centroamericano es muy poca la información disponible referente a este tipo de investigaciones. Por ello, el desarrollo de este trabajo contribuye enormemente no solamente a la pila de conocimiento y avance del estudio de la segmentación de parcelas agrícolas mediante la Inteligencia Artificial, sino también a la motivación para futuras investigaciones al respecto tanto a nivel nacional como centroamericano.

## **1.5. Estado del Arte**

El siguiente Estado del Arte tiene como objetivo presentar una revisión crítica y actualizada del estado actual de la investigación en la segmentación de parcelas agrícolas con inteligencia artificial y datos satelitales, con el fin de identificar las lagunas de conocimiento existentes y justificar la necesidad de realizar una investigación adicional en este campo.

La segmentación de parcelas agrícolas es un tema de gran relevancia en la actualidad, ya que la eficiencia y la productividad en la agricultura dependen en gran medida de la precisión en la identificación de las diferentes áreas de cultivo. En este contexto, la aplicación de técnicas de inteligencia artificial y el uso de datos satelitales se han convertido en herramientas clave para mejorar la segmentación de parcelas agrícolas.

La segmentación de parcelas agrícolas implica la identificación y delimitación de las diferentes áreas de cultivo dentro de una extensión de tierra determinada. Esta tarea puede ser compleja y costosa cuando se realiza de forma manual, especialmente en grandes extensiones de tierra. Por esta razón, el uso de tecnologías de inteligencia artificial y datos satelitales ofrece la posibilidad de automatizar el proceso de segmentación, lo que permite una mayor eficiencia y precisión en los resultados.

En este Estado del Arte se presentará una revisión crítica y actualizada de los estudios e investigaciones previas relacionadas con la segmentación de parcelas agrícolas con el uso de inteligencia artificial y datos satelitales. Se describirán las principales técnicas y algoritmos utilizados para llevar a cabo la segmentación, así como las fuentes de datos satelitales utilizadas y las herramientas de procesamiento de imágenes que se han empleado.

Además, se identificarán en la sección 1.3 Limitaciones y Alcances, las limitaciones y desafíos asociados con la segmentación de parcelas agrícolas con inteligencia artificial y datos satelitales. Se discutirán también en el capítulo 2 las diferentes aplicaciones potenciales de la segmentación de parcelas agrícolas en diferentes áreas de la agricultura, como la optimización de la fertilización y el riego, y la detección temprana de enfermedades y plagas.

La segmentación de parcelas agrícolas ha sido tema de investigación el cual ha despertado un gran interés en los últimos años, y eso puede reflejarse en los diferentes reportes que distintos grupos de investigadores y organizaciones se han encargado de realizar. Haciendo referencia al anexo A, el cual funciona como cuadro resumen de investigaciones realizadas por diferentes equipos con objetivos similares al presente documento, es posible detallar múltiples metodologías y escenarios utilizadas para el estudio de la segmentación de parcelas agrícolas.

En el año 2019, Barry Watkins y Adrian Van Niekerk presentaron el documento “A comparison of object-based image analysis approaches for field boundary delineation using multi-temporal sentinel-2 imagery.”, donde se evaluaron diferentes metodologías para la delineación de los bordes de campos de cultivos, utilizando datos multitemporales provenientes del satélite Sentinel-2 tomados durante una sesión de crecimiento en campos, huertos y viñedos. Es aquí donde se

vería la aplicación de algoritmos de segmentación como el MRS (Multi-Resolution Segmentation), WS (Watershed Segmentation) y MTS (Multi-Threshold Segmentation), algoritmos que de por sí solos podrían ser usados para obtener resultados de segmentación de parcelas pero que en combinación junto a otras metodologías estos resultados mejoran de manera significativa.

De esta manera, se introduce el uso de diferentes metodologías para el trabajo de la detección de bordes y la segmentación de parcelas agrícolas, siendo uno de los métodos más utilizados el desarrollo y uso de redes neuronales convolucionales construidas en base a diferentes arquitecturas. En 2020, un grupo de investigadores de la Universidad de Stanford, publicaron el documento “Farm Parcel Delineation Using Spatio-temporal Convolutional Networks”, donde utilizando imágenes satelitales provenientes del Sentinel-2, se trabajó en el entrenamiento de redes neuronales en base al modelo de arquitectura U-net, tanto redes entrenadas desde cero como redes neuronales pre entrenadas. Así se obtenían resultados en forma de máscaras binarias que representan las delimitaciones de los campos analizados.

La combinación de metodologías es una alternativa que puede ser revisada en el documento publicado en el año 2021, en la Revista Internacional de Observación Aplicada de la Tierra y Geoinformación (*International Journal of Applied Earth Observation and Geoinformation*), donde un grupo de investigadores crearon una metodología para la delineación de campos de agricultura en base a la combinación de algoritmos de detección de bordes, como Canny, algoritmos de segmentación como WS, y la construcción de una red neuronal en base a la arquitectura de capas U-Net, todos procesos los cuales juntos presentaron resultados bastante prometedores, indicando que la combinación de múltiples metodologías podía mejorar la delimitación de las parcelas.

Y es así como muchas otras investigaciones y documentos han abarcado el problema de la segmentación de parcelas agrícolas definiendo múltiples escenarios diferenciados desde el tipo de datos que se encuentran utilizando hasta las metodologías desarrolladas y los algoritmos utilizados. Analizar y comprender este tipo de investigaciones fue un proceso enriquecedor para el desarrollo de esta investigación, ya que aunque dentro de la región de análisis principal de la investigación no se hayan llevado a cabo análisis de esta índole, el poder conocer los parámetros y métodos que se utilizan en otras regiones del mundo presenta una pauta para la construcción de una metodología sencilla pero eficaz para comenzar a trazar la ruta de análisis geográfico en la región salvadoreña.

## CAPÍTULO 2. MARCO TEÓRICO

### 2.1 Agricultura: Segmentación de Parcelas Agrícolas

La agricultura es la actividad humana que se dedica a cultivar plantas y la cría de animales para obtener alimentos, materias primas y otros productos útiles para la sociedad, aparte de ser una actividad esencial para la supervivencia de la humanidad, genera empleo y contribuye al desarrollo social y económico, desempeñando un papel importante en la economía de muchas regiones y países.

Una parcela agrícola es una superficie de tierra continua, declarada por un agricultor, en la que no se cultiva más de un único grupo de cultivos (Papadakis et al, 2016). Estas parcelas pueden variar en tamaño, forma y características, y para una gestión agrícola efectiva, es esencial dividirlas en unidades más pequeñas y distintas, y establecer sus límites por medio del proceso de segmentación de parcelas agrícolas. Este proceso se basa en el análisis de imágenes satelitales, imágenes aéreas o datos geoespaciales para identificar y delimitar visualmente las áreas de cultivo dentro de un campo, delineando de forma manual tradicionalmente, y ahora cada vez con más ayuda de diferentes algoritmos. Estas técnicas incluyen algoritmos y modelos matemáticos que utilizan características espectrales, textuales y espaciales de los datos para detectar los límites de las parcelas y separarlas de otros elementos presentes en la imagen, como caminos, cuerpos de agua o áreas no cultivadas.

La segmentación de parcelas agrícolas es necesaria para una gestión precisa de los cultivos, la optimización de recursos, la planificación y el seguimiento de cultivos, así como para el análisis espacial y la toma de decisiones informadas. Al proporcionar información detallada sobre las áreas de cultivo, la segmentación ayuda a mejorar la productividad, eficiencia y sostenibilidad en la agricultura.

#### 2.1.1 Aplicaciones de la segmentación de parcelas agrícolas

La segmentación de parcelas agrícolas tiene diferentes aplicaciones en el ámbito agrícola y de la agricultura de precisión. Algunas de estas aplicaciones son las siguientes:

- Gestión de cultivos: la segmentación de parcelas agrícolas permite una gestión de cultivos más precisa al proporcionar información detallada sobre cada parcela agrícola. Esto ayuda a los agricultores a tomar decisiones más informadas sobre las diferentes prácticas agrícolas como el riego, la fertilización, la aplicación de pesticidas y otros temas. Al

comprender las características únicas de cada parcela, se pueden diseñar estrategias de manejo para maximizar la productividad y minimizar el desperdicio de recursos.

- Monitoreo del crecimiento de los cultivos: La segmentación de parcelas agrícolas permite un seguimiento detallado tanto del crecimiento, como del desarrollo de los cultivos. Al analizar las imágenes o datos segmentados a lo largo del tiempo, se pueden detectar patrones de crecimiento, e identificar áreas con problemas de salud de las plantas, y tomar medidas correctivas ante estos. Por lo tanto, facilita el control de enfermedades, plagas y malezas, y ayuda a maximizar el rendimiento de los cultivos.
- Análisis de rendimiento agrícola: el rendimiento de cada parcela agrícola se puede evaluar de manera precisa y específica al dividirlas. Esto permite comparar el rendimiento entre diferentes parcelas, encontrar las áreas más productivas y analizar las causas de las variaciones en el rendimiento. Esto permite a los agricultores poder mejorar la productividad global del campo utilizando esta información para hacer cambios en sus prácticas agrícolas y tomar decisiones estratégicas.
- Planificación de siembras y rotaciones de cultivos: la segmentación de parcelas agrícolas proporciona información espacial detallada que es útil para planificar siembras y rotaciones de cultivos. Los agricultores pueden determinar las áreas más adecuadas para ciertos cultivos, considerar la estructura del suelo y las condiciones climáticas específicas de cada parcela, y optimizar la secuencia de siembra y rotación para mejorar la salud del suelo y maximizar los rendimientos.

## 2.2. Imágenes Satelitales

Las imágenes satelitales son capturas que se obtienen gracias a sensores multiespectrales, regularmente, instalados en satélites que orbitan el planeta tierra. Los sensores captan la radiación electromagnética que emite el cuerpo celeste. El resultado de esto es enviado a una estación en tierra para su posterior procesamiento y análisis (Aguilar, 2014).

### 2.2.1 Teledetección

La teledetección es un campo que posee múltiples definiciones. Tomando en cuenta el propósito de esta investigación, la teledetección se define como la práctica que busca derivar información sobre la superficie terrestre, o la superficie de cuerpos de agua, utilizando imágenes obtenidas

desde una perspectiva aérea, utilizando la radiación electromagnética en una o varias regiones del espectro electromagnético (Campbell, 2002).

La práctica de la teledetección consiste en recoger información de objetos físicos, tales como edificios, agua, vegetación, etc. Esto por medio del análisis de las imágenes obtenidas por los sensores (pueden ser obtenidas también por cámaras, u otro tipo de sensores), las cuales son presentadas como arreglos numéricos que pueden ser mostrados y analizados más fácilmente por una computadora. Este formato de imagen puede ser poco familiar debido al uso de diferentes resoluciones, y la utilización de regiones espectrales fuera del espectro de luz visible. Para convertir esa data en información aplicable es necesario el efectivo análisis e interpretación de ella. (Campbell, 2002)

### **2.2.2. Obtención de Imágenes Satelitales**

La obtención de imágenes para teledetección se remonta a los orígenes de la fotografía (Campbell, 2002). El primer registro de obtención de imágenes de la tierra desde una vista aérea fue en 1858, obtenidas por medio de fotografías tomadas desde un globo cautivo. Durante los años posteriores, se fueron mejorando la calidad de estas imágenes, siendo tomadas desde globos aerostáticos y cometas.

Con la mejora de la aviación, y el poder de los nuevos aviones, empezaron a ser utilizados también como plataformas para la obtención de imágenes aéreas. Durante la primera guerra mundial (1914 - 1918) se creó equipamiento específico para la toma de fotografías aéreas, y personal fue entrenado para la obtención, procesamiento e interpretación de imágenes. En los años posteriores, durante la gran depresión (1929 - 1939) y la segunda guerra mundial (1939 - 1945), se hicieron importantes avances y mejoras al campo de la fotografía, y se le dio más usos a este. Se creó un gran interés al uso de regiones no visibles del espectro electromagnético, especialmente al infrarrojo y microondas, al encontrarse muchos usos que se les podía dar (Campbell, 2002). La investigación y entrenamiento práctico durante la guerra dejó bastante conocimiento y experiencia para los usos cotidianos de los campos no visibles.

En los años sesenta fue lanzado con éxito el primer satélite meteorológico, TIROS-1, cuyo objetivo era el monitoreo del clima. Este avance sentó las bases para los posteriores satélites de observación de la tierra. También durante estos años, muchos instrumentos que eran utilizados para fines militares fueron liberados a la población civil lo que permitió el mejoramiento del estudio aéreo fuera del espectro visible. En 1972 fue lanzado el primer satélite dedicado específicamente para el monitoreo de áreas terrestres, el Landsat 1. Cada imagen tomada del

satélite describe una amplia superficie de la tierra en varias regiones del espectro electromagnético, y con suficiente precisión y detalle para ser usada en diferentes campos de aplicación (Campbell, 2002).

En la década siguiente, el Jet Propulsion Laboratory, ubicado en Pasadena, California, con la ayuda de la NASA (por sus siglas en inglés, National Aeronautics and Space Administration) empezaron el desarrollo de instrumentos que permitieran obtener imágenes de la tierra con un detalle espectral sin precedentes.

En el año 2014, la Agencia Espacial Europea inició el programa Sentinel, el cual consta de varias misiones satelitales que tienen como objetivo el monitoreo continuo de la tierra, y sus recursos. Sentinel-2, es la misión de la cual se extrajeron imágenes para esta investigación. De acuerdo con la Agencia Espacial Europea (ESA), esta consiste en dos satélites con órbitas polares, los cuales tienen el propósito de obtener imágenes multiespectrales de alta resolución de vegetación, cuerpos de agua, caminos de agua sobre superficie terrestre, áreas costeras, etc. Los satélites que conforman esta misión son Sentinel-2A, lanzado en junio de 2015, y Sentinel-2B, lanzado posteriormente en marzo de 2017.

### **2.2.3. Imágenes Satelitales: Bandas Espectrales e Índices de Vegetación**

Una de las características más destacables que poseen los sensores remotos es la capacidad de detectar y filtrar la luz en longitudes de onda que serían invisibles a simple vista. Esta categorización se hace mediante la definición de múltiples bandas donde cada una de ellas corresponde a un rango específico de longitudes de onda las cuales transmiten información sobre una determinada característica. La combinación tanto de imágenes como de bandas brindan una mayor visualización de datos que de otro modo sería muy complicado observar (EOS, 2022).

Existen múltiples tipos de bandas en los diferentes sensores presentes en las herramientas de teledetección. Den entre las más comunes pueden considerarse las siguientes:

- Las bandas roja, verde y azul que se encargan de cubrir el rango de 400 a 700 nanómetros correspondiente al espectro visible, las cuales en conjunto conforman lo que se conoce como banda pancromática, la cual es utilizada para resaltar detalles y mejorar la calidad de la imagen cuando se combina con otras bandas

- Las bandas de las longitudes de onda del infrarrojo cercano (NIR) poseen la característica de reflejarse en las hojas y el dosel de las plantas, siendo una herramienta muy útil para la observación y vigilancia de plantas y contrastes con masas de agua.
- La banda costera diseñada para el realce de aguas poco profundas y la medición de los cambios de color presentes en el océano , al igual que para la detección de las partículas de aerosol en la atmósfera.

El ojo humano solamente es capaz de ver dentro de un pequeño rango visible del espectro, la combinación de estas y muchas otras bandas contribuyen a la obtención de imágenes mejor definidas y con características resaltadas, logrando así que el estudio y análisis de estas características pueda realizarse de una manera más sencilla y eficaz, así como poder llegar a explorar nuevas conclusiones.

En investigaciones centradas en el análisis del terreno, los índices de vegetación son un parámetro muy importante los cuales brindan información adicional y precisa sobre las características presentes en el terreno analizado. Mediante operaciones que pueden incluir dos o más bandas espectrales, la cubierta vegetal puede verse resaltada en función de la respuesta de estas operaciones, logrando a su vez atenuar aspectos con menor relevancia como lo es el suelo, la iluminación, entre otros. El resultado de los índices de vegetación permite obtener nuevas imágenes que destacan características relacionadas a las coberturas vegetales, como lo son la densidad, el área foliar y la actividad clorofílica. (Sergieieva, 2022).

Existe una gran variedad de índices de vegetación que pueden ser utilizados para resaltar características del terreno, entre ellos se encuentran siguientes índices: Índice de Vegetación de Diferencia Normalizada (NDVI), Índice de Vegetación Ajustado al Suelo (SAVI) y el Índice Diferencial de Agua Normalizado (NDWI).

- Índice de Vegetación de Diferencia Normalizada (NDVI): Se trata de los índices de vegetación más utilizados y adecuados para el seguimiento de la dinámica de desarrollo de la vegetación sobre el terreno. Uno de los principales factores al cual está ligado el NDVI es la fenología de las masas vegetales y su estado de salud.
- Índice de Vegetación Ajustado al Suelo (SAVI): El objetivo principal del uso de este índice como parte de las investigaciones es que busca mitigar el impacto del brillo del suelo. Utilizando un factor de ajuste del suelo (L), el SAVI se encarga de corregir los efectos del ruido del suelo (color, humedad, etc.).

- Índice Diferencial de Agua Normalizado (NDWI): Utilizado para observar el estado hídrico presente en cultivos y terreno en general. Por este medio, se logra identificar si existe un déficit o saturación de humedad en el cultivo.
- Índice de Vegetación Ajustado al Suelo Optimizado (OSAVI): Considerado una modificación del SAVI, se trata de un índice que también se encarga de utilizar la reflectancia del espectro del infrarrojo cercano y del rojo, siendo la principal diferencia de un factor de ajuste del fondo del dosel (0,16). El valor de su uso puede ser encontrado para el monitoreo de diferentes zonas en donde la densidad de la vegetación sea muy baja y el uso del dosel tenga un mayor peso sobre el suelo analizado.
- Índice de Vegetación Resistente a la Atmósfera (ARVI): De igual manera, se trata de una corrección realizada sobre la fórmula del NDVI, siendo considerada como uno de los primeros índices considerados insensibles a los factores atmosféricos, logrando mitigar la dispersión atmosférica, mediante la duplicación de las mediciones presentes del espectro rojo y utilizando cada vez más ondas azules.

#### **2.2.4. Aplicación de Imágenes Satelitales**

Hay muchos tipos de imágenes satelitales, las cuales tienen diferentes áreas de aplicación tales como monitoreo climático, espionaje militar, evaluaciones de vegetación, etc. Su aplicación se determina según su resolución espacial, su tiempo de revisita, y su información espectral, y está ligada a menudo a otro tipo de datos obtenidos por distintas disciplinas. Su uso creció mucho más con la aparición de los GIS (Geographic Information Systems), los cuales son programas dedicados al análisis de información geoespacial. (Aguilar, 2014).

A nivel de un país, las imágenes satelitales son importantes para el estudio de la expansión urbana, riesgos urbanos, análisis para el uso de la tierra y campos de cultivo, administración de espacios y recursos naturales, y el monitoreo de desastres, o de problemas medioambientales. En agricultura brindan información sobre el estado de cultivos, posibles plagas, crecimiento, cosecha, etc. A nivel mundial, las imágenes pueden ser usadas para investigaciones de desarrollo, de gestión de cambios ambientales en zonas que incluyen varios países. Para que todo esto sea posible, el desarrollo de herramientas, diseño de equipamiento, capacitación personal, investigación y materiales es apoyado por empresas e industrias continuamente. (Campbell, 2002)

### **2.3. Inteligencia Artificial: Machine Learning**

El término “Inteligencia Artificial” (IA) ha estado muy presente en los últimos años, tanto como parte de una conversación casual entre compañeros de trabajo, como de tema de investigación para el desarrollo y mejoras de herramientas que puedan ser utilizadas para la facilitación de tareas tanto complejas como repetitivas. Buscar definir la IA de primera mano puede llegar a ser complicado sin conocer antes la definición de “Inteligencia” como tal. La Real Academia Española (RAE), define la Inteligencia como la capacidad de entender o comprender problemas y buscar soluciones a estos, en base a la habilidad, destreza y experiencia. Definido esto, es posible aplicarlo dentro del contexto de la IA, detallando que el desarrollo de máquinas inteligentes hace referencia a la construcción y desarrollo de máquinas capaz de entender y resolver problemas en base a la experiencia que esta posea frente a escenarios similares (Galipienso et al, 2003).

Dentro del mundo de las ciencias computacionales, la IA se entiende como la encargada del estudio y desarrollo de soluciones, conocidas como modelos, que sean capaces de estudiar, aprender y replicar actividades propias de los seres humanos (Takeyas, 2007). Para ello, el desarrollo de herramientas enfocadas en el uso de la IA debe de hacerse siempre teniendo en cuenta características importantes como lo son el razonamiento de problemas o escenarios, el aprendizaje y detección de patrones, y la habilidad para tomar decisiones en base a todo lo estudiado y aprendido.

Para que el desarrollo y aplicación de una solución que implemente la IA se deben tener en cuenta distintos fundamentos que son importantes para una correcta y sobre todo eficiente herramienta autosuficiente. Estos fundamentos buscan que la solución implementada conste con las siguientes características: Autoaprendizaje, buscar siempre aprender más para traer soluciones más rápidas o eficientes; interacción con usuario, tener la capacidad de interactuar con un usuario o administrador con el fin de entender lo que se esté solicitando; reacción en tiempo real, poder trabajar en tiempo real indicando estimaciones de tiempo de realización de tareas y respuestas a las problemáticas solicitadas; y por último, autonomía, que aunque no se trata de una autonomía total, la IA debe de tener la capacidad de trabajar por su cuenta cuando ya se le haya asignado una tarea en específico (Serna, Acevedo, & Serna, 2017).

#### **2.3.1. Machine Learning**

Mediante la IA, se ha hecho notar que el proceso de aprendizaje y toma de decisiones no es una habilidad que pueda ser adjudicada únicamente a los seres humanos, sino que se pueden desarrollar máquinas capaces de analizar, entender y utilizar información histórica para la

resolución de problemas y/o análisis de resultados. Sin embargo, decir que una máquina o programa pueda aprender de su “experiencia” no es tan literal como suena, y es ahí donde entra en acción técnicas como el Machine Learning.

El Machine Learning (ML), también conocido como Aprendizaje Automático, puede ser visto como el conjunto de métodos computacionales que se encargan de mejorar el rendimiento de los sistemas informáticos. Así como el ser humano desarrolla la experiencia en base a distintas experiencias y actividades a lo largo de su vida, las maquinas encuentran dicha experiencia y/o conocimiento en forma de datos o información, y la principal tarea de la cual se encarga el ML es el desarrollo de algoritmos de aprendizaje que resultan en modelos creados a partir de la información analizada. (Zhou, 2021)

Es posible identificar una clasificación para los algoritmos de aprendizaje que son utilizadas para la construcción de sistemas basados en el Aprendizaje Automático. Dicha clasificación puede resumirse en:

- A. Aprendizaje Supervisado
- B. Aprendizaje No Supervisado
- C. Aprendizaje Reforzado

### **2.3.2. Aprendizaje Supervisado**

El Aprendizaje Supervisado busca la utilización de un algoritmo que tenga como propósito la generación de una función encargada en asignar su respectivo resultado a un valor de entrada dado en específico (Ayodele, 2010). Dentro de esta clasificación, se definen una serie de conjunto de datos etiquetados que serán utilizadas para el entrenamiento del algoritmo seleccionado, el cual una vez terminado dicho entrenamiento, será el encargado de clasificar nueva información o predecir futuros resultados de la manera más precisa posible.

Visto de otra forma, el Aprendizaje Supervisado se encarga de enseñar al algoritmo como debe de realizar su trabajo utilizando un conjunto de datos clasificados bajo una serie de criterios para la generación de etiquetas que contribuyan a la identificación de patrones que pueden ser utilizados para analizar y estudiar la información con el fin de producir un resultado en forma de predicciones. (Manrique, 2020)

### **2.3.3. Aprendizaje No Supervisado**

También conocido como “Machine Learning No Supervisado”, el Aprendizaje No Supervisado se basa en el uso de algoritmos de aprendizaje automático al igual que el Aprendizaje Supervisado, con la diferencia de que este busca analizar y agrupar en clústeres conjuntos de datos sin ninguna etiqueta en especial, sino que se basa su análisis en la identificación de agrupaciones de datos o patrones que pueden llegar a pasar desapercibidos sin necesidad de que estos datos hayan sido manipulados por el usuario.

Los Algoritmos de Aprendizaje No Supervisado presentan características muy útiles para campos como el análisis de datos exploratorios, segmentación de productos y/o clientes y reconocimiento de imágenes, todo esto, gracias a la gran capacidad que presentan para descubrir tanto similitudes como diferencias dentro de un conjunto de información.

Es importante recalcar que el Aprendizaje No Supervisado guarda una mayor similitud con el funcionamiento de un cerebro humano que el que tendría el Aprendizaje Supervisado. Puede verse con el ejemplo de los millones de fotorreceptores que existen dentro del ojo humano los cuales se encargan de percibir cambios constantemente para poder llegar a interpretar todos los visuales presentes en el día a día, cada escenario siendo tan diferente como el anterior. La cantidad de luz, el orden de los objetos, su color y forma, son características que contribuyen a la generación e identificación de patrones en el mundo real, que al igual que en los algoritmos de Aprendizaje No Supervisado, todas estas características no se conocen de un primer momento, en este caso, desde el momento del nacimiento. Es la capacidad de identificar dichas propiedades y agruparlas de la mejor manera posible la que construye el mundo que rodea al ser humano, y en consecuencia, es esta habilidad que convierte a los métodos de Aprendizaje No Supervisado en herramientas poderosas para el uso dentro y para la construcción de sistemas informáticos (Dayan, et al, 1999).

### **2.3.4. Aprendizaje Reforzado**

Hoy en día, las aplicaciones de inteligencia artificial (IA) están presentes en numerosos ámbitos de la vida cotidiana. Estás IA "aprenden" gracias a la variedad de técnicas existentes, incluido el aprendizaje por refuerzo que tiene el potencial de crear máquinas o agentes inteligentes, capaces de realizar tareas de forma muy parecida a como lo hacemos nosotros los humanos, esta se enfoca principalmente en un método de constante prueba y error, de forma similar a como aprenden los humanos y los animales en general.

Al igual que los niños pequeños que aprenden a caminar, ajustan las acciones según los resultados que experimentan, como dar un paso más pequeño si el paso más grande que hicieron antes les hizo caer, las máquinas y los agentes de software utilizan algoritmos de aprendizaje de refuerzo para determinar el comportamiento ideal en función de los comentarios del entorno. Y, dependiendo de la complejidad del problema, los algoritmos de aprendizaje por refuerzo pueden seguir adaptándose al entorno a lo largo del tiempo si es necesario, para maximizar la recompensa a largo plazo (Dell Technologies , 2018).

Este tipo de aprendizaje se emplea especialmente en áreas como la robótica, la optimización de recursos o los sistemas de control, permitiendo definir políticas o estrategias óptimas de actuación en entornos concretos.

### **2.3.5. Aprendizaje Profundo**

El aprendizaje profundo, o *Deep Learning*, se trata de un subcampo del ML el cual se basa en la construcción de una estructura de múltiples redes neuronales interconectadas de manera jerárquica, de tal forma que se logre obtener una estructura neuronal similar a la del cerebro humano, lo que permite poder desarrollar un análisis no lineal de la información que se utilice como entrada para la red. Esta tecnología es uno de los pilares utilizados en grandes avances tecnológicos como lo son los vehículos automatizados y el control de voz presente en múltiples dispositivos como teléfonos inteligentes, tabletas y altavoces (Centeno, 2022).

Gracias al renovado interés que surgió en 1980 por las redes neuronales, los investigadores se dieron a la tarea de buscar nuevas técnicas para su entrenamiento, siendo uno de los avances más importantes el descubrimiento de la retropropagación, una técnica de entrenamiento que utiliza múltiples capas de unidades ocultas trabajando con la propagación del error desde la capa de salida de nuevo a la capa de entrada, con el fin de ajustar los pesos de la red a medida avance (Hinton, 2006).

El desarrollo de estas nuevas técnicas permitió el desarrollo y entrenamiento de redes neuronales capaces de aprender patrones mucho más complejos a partir de grandes volúmenes de datos, convirtiéndose hasta hoy día en uno de los avances más importantes y poderosos en relación con la IA.

## 2.4. Metodologías de Segmentación de Parcelas Agrícolas

Tomando como base lo explicado en el apartado de imágenes satelitales, la segmentación subdivide una imagen en sus partes constituyentes u objetos con el fin de separar las partes de interés del resto de la imagen. En el proceso de detectar las partes en una imagen se identifican bordes de la imagen, o se segmentan en regiones, líneas o curvas, entre otras características. Otra definición considera a la segmentación como la clasificación de los puntos de la imagen (píxeles), indicando las clases a la que pertenecen los diferentes píxeles (La Serna Palomino & Román Concha, 2007).

La información de parcelas agrícolas se obtiene a partir de imágenes satelitales, y tanto su preparación como su procesamiento de segmentación de límites se realiza mediante algoritmos especializados, los cuales pueden mostrar los resultados en base a los bordes, regiones, líneas, entre otras características de estos terrenos agrícolas.

En la actualidad, existen algoritmos de Inteligencia Artificial (IA) y Machine Learning (ML) que pueden ser utilizados para la segmentación de límites de parcelas agrícolas, estos son creados y ejecutados mediante el procesamiento de imágenes satelitales o aéreas. Estos algoritmos se basan en técnicas de procesamiento de imágenes y aprendizaje automático para identificar y segmentar los límites de parcelas agrícolas a partir de las características de las imágenes obtenidas.

### 2.4.1 Algoritmos utilizados para la segmentación de parcelas agrícolas

- **Redes neuronales convolucionales (CNN):** Una red neuronal convolucional es un tipo de algoritmo de aprendizaje profundo que se utiliza mayormente para analizar y aprender atributos visuales de grandes cantidades de datos (*Redes neuronales convolucionales (CNN) y aprendizaje profundo*, s. f.-a). Estas funcionan procesando grandes cantidades de datos y luego extrayendo características detalladas para su posterior clasificación y detección.

Se caracterizan por tener tres tipos de capas: la capa convolucional, la capa de agrupación y la capa totalmente conectada. Estas redes neuronales son capaces de aprender patrones complejos y detectar objetos en imágenes con alta precisión, por lo que son una alternativa óptima para realizar tareas de segmentación y detección de límites de parcelas agrícolas.

- **Algoritmos de segmentación basados en regiones (RBS):** técnicas o métodos que consisten en dividir imágenes en regiones y luego agruparlas en base a características similares. Estos algoritmos son eficientes en la segmentación de objetos o estructuras irregulares, otra característica especial es que el algoritmo divide una imagen en diferentes regiones o segmentos que comparten características similares, como el color, la textura o la intensidad.

Uno de los algoritmos más representativos de este apartado es el llamado “Algoritmo de crecimiento de super pixeles”, el cual consiste en una agrupación de píxeles adyacentes en super píxeles que son conjuntos compactos de píxeles con características similares (Oscar, 2017 ).

- **Algoritmos de segmentación basados en bordes (EBS):** técnica en la que se detectan bordes de objetos en una imagen. Este algoritmo es adecuado para la segmentación de objetos con bordes definidos o regulares.

Entre sus más representativos algoritmos aplicables en este apartado se mencionan:

- **Algoritmo de Canny:** Conocido como el detector o algoritmo de Canny fue desarrollado por John F. Canny, teniendo como objetivo principal la detección de bordes mediante un algoritmo de multietapa. Los puntos más destacables de este criterio son: Detección, para evitar la eliminación de bordes importantes o proporcionar bordes falsos; Localización, que define que la distancia entre la posición real y la localizada del borde debe ser la mínima posible; y la capacidad de proporcionar una respuesta generalizada la cual sea construida en base a múltiples respuestas correspondientes a un único borde (Valverde Rebaza, 2010).
- **Algoritmos de segmentación basados en grafos:** son técnicas que modelan imágenes en forma de grafos y, aplicando técnicas de la teoría de grafos, segmentan las imágenes en regiones.

Entre sus algoritmos más populares se encuentran:

**Algoritmo de corte mínimo (Minimum Cut):** es utilizado para partitionar un grafo en dos conjuntos disjuntos de nodos minimizando el costo de cortar las aristas entre los conjuntos (Datta & Datta, 2023). En el contexto de segmentación de imágenes, cada píxel se representa como un nodo en el grafo, y las conexiones entre los píxeles se definen mediante las aristas ponderadas. El algoritmo encuentra el corte mínimo que separa los

nodos del grafo en dos conjuntos, lo que resulta en una segmentación de la imagen (Datta & Datta, 2023).

- **K-means:** Es un algoritmo de agrupamiento que se puede utilizar para dividir una imagen en diferentes regiones. Se utiliza para segmentar imágenes satelitales y detección de campos agrícolas en base a su tonalidad.

El algoritmo consta de tres pasos:

- A. **Inicialización:** una vez escogido el número de grupos,  $k$ , se establecen  $k$  centroides en el espacio de los datos, por ejemplo, escogiendo aleatoriamente.
  - B. **Asignación objetos a los centroides:** cada objeto de los datos es asignado a su centroide más cercano.
  - C. **Actualización centroides:** se actualiza la posición del centroide de cada grupo tomando como nuevo centroide la posición del promedio de los objetos pertenecientes a dicho grupo (Universidad de Oviedo, 2018).
- **Clasificación supervisada:** es un tipo de aprendizaje que utiliza un conjunto de imágenes etiquetadas para entrenar un modelo que identifique características de estas. Se puede utilizar para segmentar parcelas agrícolas en imágenes satelitales o aéreas en base a la textura, forma y color. Los algoritmos de clasificación supervisada varían en la forma en que identifican y describen las regiones en el espacio espectral. Algunos pueden manejar clases definidas que se superponen entre sí, y otros generan límites firmes entre clases, entre los algoritmos más utilizados se encuentran: Random Forest, máxima verosimilitud, mínima distancia, paralelepípedos (RPubs, 2018).

#### 2.4.2 Implicaciones, ventajas y desventajas de los algoritmos de segmentación.

Cabe destacar que cada uno de estos algoritmos presentan tanto ventajas como desventajas, y la elección del algoritmo a utilizar dependerá de varios factores. Previamente se han mostrado las bondades y beneficios de cada uno de los algoritmos, sin embargo, es importante resaltar algunas desventajas o limitaciones que algunos de estos algoritmos podrían presentar al usuario:

- Dificultades en la delimitación precisa de las parcelas agrícolas debido a la variabilidad de los cultivos y las condiciones ambientales. Esto dependerá sobre todo de la porción de tierra, ubicación de los terrenos agrícolas o la temporada del año en la que se extraigan las imágenes satelitales que se desean poner a prueba.
- Influencia de factores como la calidad de las imágenes, la resolución espacial y temporal, y la disponibilidad de datos en la segmentación de parcelas. Esto debido a que puede

darse el caso en el que el conjunto de datos a probar pueda contener imágenes satelitales con información nula e incluso casos en los que las imágenes extraídas pierden resolución al momento de ponerla a prueba con estos algoritmos.

- Consideraciones técnicas y computacionales en la implementación de algoritmos de segmentación de parcelas agrícolas a gran escala. Esto hace referencia a que, ante la necesidad de aplicar estos algoritmos a grandes cantidades de datos, esto tiende a afectar directamente al funcionamiento y velocidad de procesamiento de los equipos computacionales, por lo que se recomienda contar previamente con un equipo con las especificaciones suficientes o necesarias para poder sacar todo el potencial de los algoritmos.

## CAPÍTULO 3. METODOLOGÍA

### 3.1 Fuentes de Datos y Procesamiento

#### 3.1.1 Selección, Obtención y Clasificación de Datos Satelitales

##### A. Área de Estudio

La principal área de estudio utilizada para el desarrollo de la investigación constituye una pequeña porción del territorio salvadoreño. El Salvador es un país localizado en América Central, entre el norte y sur del Continente Americano, limitando al poniente con Guatemala, al norte con Honduras, al oriente con Honduras y Nicaragua en el Golfo de Fonseca y al sur con el Océano Pacífico. Con una extensión territorial de aproximadamente 21,040.79 kilómetros cuadrados, El Salvador se encuentra ubicado en latitud del hemisferio norte, siendo sus coordenadas geográficas entre el 13° y 14° latitud norte y entre 87° y 90° longitud occidental. La información geográfica de El Salvador se puede apreciar en la figura 1.

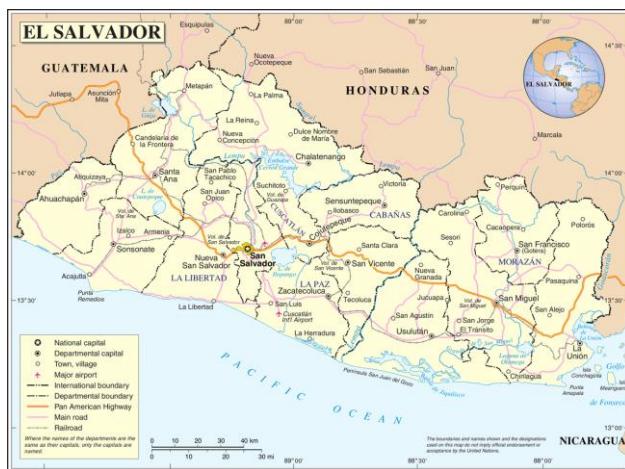


Fig 1. Geografía de El Salvador (Obtenido de Wikipedia).

La porción de terreno seleccionada para el análisis y desarrollo de segmentación de las parcelas agrícolas se trata del municipio de Santa María. Ubicado en el departamento de Usulután, El Salvador, el municipio de Santa María posee una superficie total aproximada de 11.9 kilómetros cuadrados, con una altura media de 93 metros sobre el nivel del mar. A pesar de no ser de los municipios con mayor extensión dentro del departamento de Usulután, según el censo oficial del realizado en el año 2007, Santa María posee población cercana a 10,731 habitantes, con una densidad aproximada a 901,76 habitantes por kilómetro cuadrado. En la figura 2 se puede apreciar un poco sobre la localización y geografía del área de Santa María, Usulután.

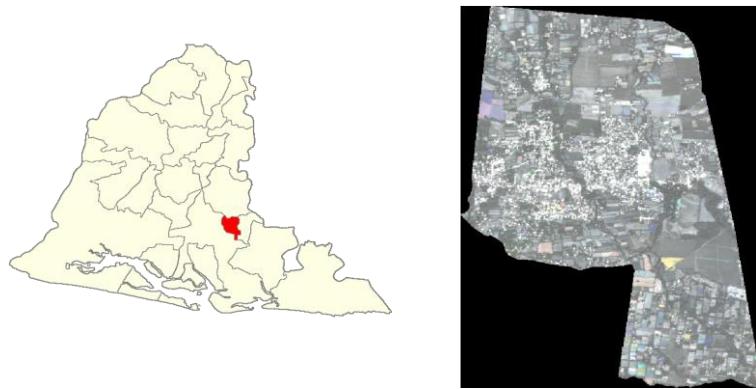


Fig 2. Área de Estudio: Municipio de Santa María, Usulután (Elaboración propia).

Ahora bien, a pesar de que la principal zona de análisis de la investigación se centre en el territorio salvadoreño, se hizo uso de un área de estudio adicional ubicada en el continente europeo, de esta manera se pudieron probar las implementaciones construidas sobre una nueva zona de estudio, la cual poseía tanto propiedades similares como diferentes en relación del área de estudio principal.

Para la segunda área de estudio, se seleccionó la provincia de Zeeland, perteneciente a los Países Bajos. Los Países Bajos se trata de un estado de Europa Occidental a orillas del Mar del Norte, ubicado entre los países de Bélgica y Alemania. Se trata de un país muy llano, teniendo la mitad de su territorio por debajo del nivel del mar y dos quintas de este ubicadas entre 0 y 5 metros sobre el nivel del mar, encontrando su punto de altura máxima en la provincia de Limburgo alcanzado los 321 metros sobre el nivel mar. Posee una superficie aproximada de una combinación de 33, 893 kilómetros cuadrados de tierra y 7,650 kilómetros cuadrados de agua. En la figura 3 puedes observarse su geografía.



Fig 3. Geografía de los Países Bajos (Obtenido de Wikipedia).

Para la investigación se trabajó con información referente a la provincia de Zeeland (“Zelanda” en español). Ubicada al suroeste del país, consta de 1783 kilómetros cuadrados de superficie con

una población aproximada de 382, 304 habitantes según el censo oficial del año 2018. La provincia de Zeeland es considerada un gran delta fluvial al encontrarse situada en la desembocadura de varios ríos importantes, como lo son el río Escalda y Rin, tal y como se observa en la figura 4.

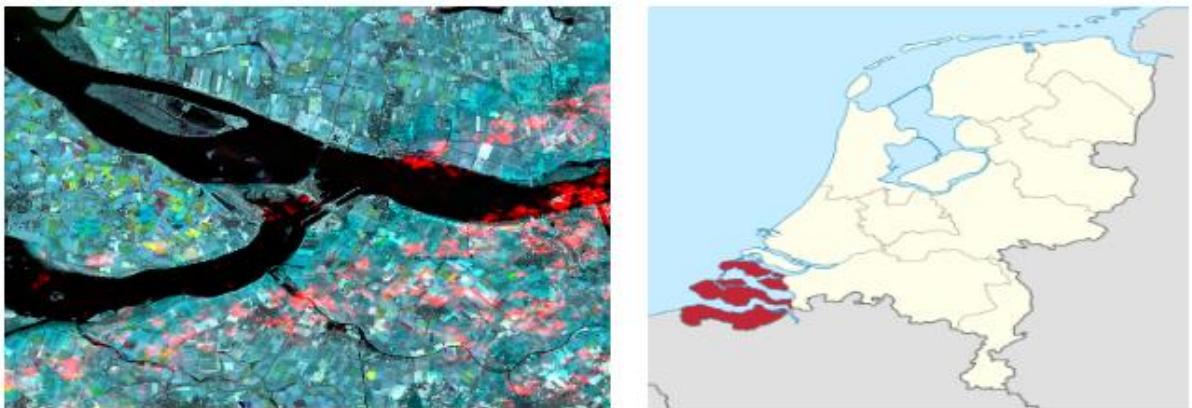


Fig 4. Área de Estudio: Región de Zelanda, Países Bajos (Elaboración propia).

De esta manera se han definido las áreas de estudio de las cuales se obtuvieron las imágenes satelitales que se utilizaron para el análisis de la investigación y el entrenamiento de los modelos desarrollados.

## B. Información Geográfica: Imágenes Satelitales

Para poder analizar las áreas de estudio definidas en la sección anterior se necesitaba poder tener las características necesarias dentro de un formato que tuviera la capacidad de dar un acceso sencillo a las propiedades principales de las imágenes satelitales obtenidas. Por ello, y gracias al apoyo de las herramientas que la empresa Geokapti posee para la captura y procesamiento de datos satelitales, se logró almacenar la información geográfica proveniente del satélite Sentinel-2 dentro del cubo de datos utilizando el formato NetCDF.

El formato NetCDF es un conjunto de bibliotecas de software y formatos de datos que poseen la característica de ser autodescriptivos e independientes de su fuente de creación y permiten acceder y compartir datos científicos mediante el uso de matrices. NetCDF se basa en un modelo de datos de dimensiones, variables y atributos, lo cual resume lo que se conoce como modelo clásico, el cual fue extendido y mejorado para ser compatible con nuevas funciones como lo son los grupos, dimensiones múltiples ilimitadas y nuevos tipos incluyendo tipos definidos por el usuario. El esquema de este formato se puede apreciar en la figura 5.

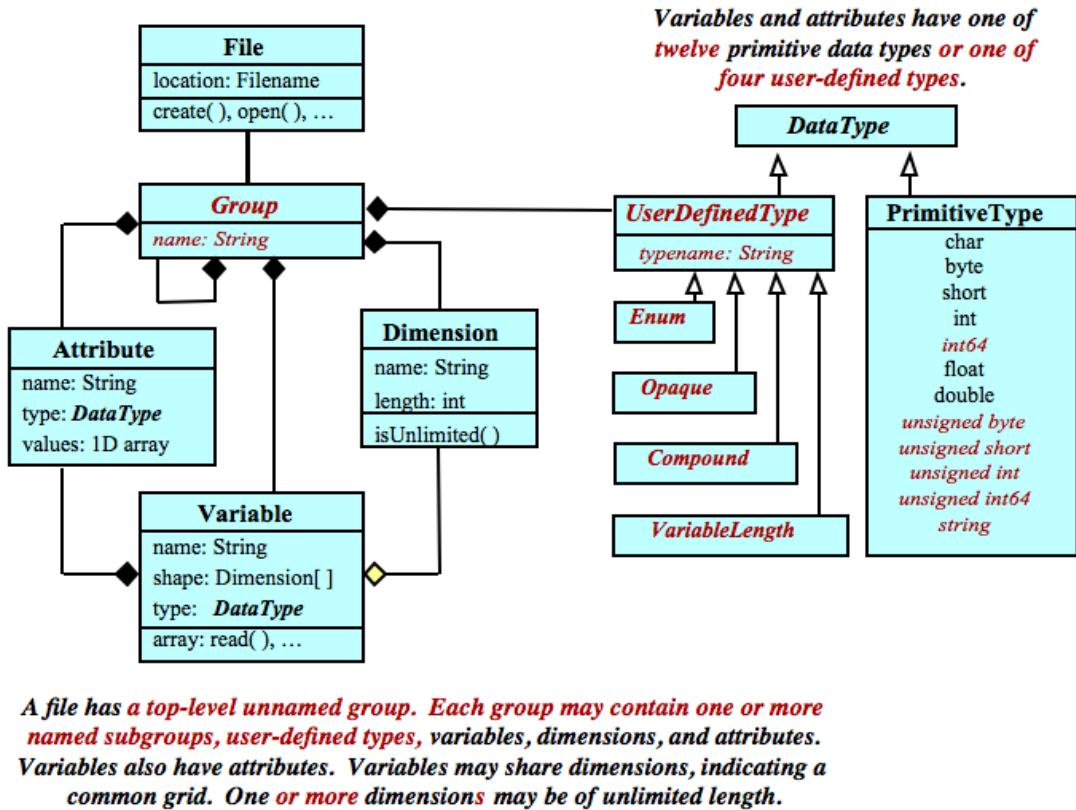


Fig 5. Esquema de Modelo de Datos mejorado para el formato NetCDF (Fuente: Unidata. Data Services and Tools for Geoscience)

Muchos tipos de datos pueden ser almacenados utilizando el formato NetCDF, pero su principal uso se encuentra dentro de la comunidad del estudio de las ciencias de la Tierra. NetCDF considera el mundo de los datos científicos muy relacionado a cómo se consideran los datos de información atmosférica: como una serie de matrices relacionadas. Gracias a esto, es posible almacenar magnitudes físicas situadas en puntos de latitud y longitud, o también pueden guardar información de apoyo como unidades o fuente de los datos.

Gracias al formato NetCDF fue posible almacenar la información proveniente directamente del satélite sentinel-2 y así poder acceder a cada una de sus propiedades mediante el uso de múltiples librerías que permiten abrir dichos archivos .nc y poder manipularlos como cualquier otra variable. A manera de resumen, la estructura de los cubos de datos puede ser apreciada en la figura 6.

```

Dimensions:          (lat: 631, lon: 413, time: 70)
Coordinates:
  * time            (time) datetime64[ns] 2023-01-01 2023-01-06 ... 2022-12-27
  * lon             (lon) float64 -88.43 -88.43 -88.43 ... -88.39 -88.39 -88.39
  * lat             (lat) float64 13.37 13.37 13.37 13.37 ... 13.31 13.31 13.31
Data variables: (12/19)
  granule        (time) uint64 dask.array<chunksize=(70,), meta=np.ndarray>
  fname          (time) object dask.array<chunksize=(70,), meta=np.ndarray>
  B01            (time, lat, lon) float32 dask.array<chunksize=(70, 150, 150), meta=np.ndarray>
  B09            (time, lat, lon) float32 dask.array<chunksize=(70, 150, 150), meta=np.ndarray>
  B08            (time, lat, lon) float32 dask.array<chunksize=(70, 150, 150), meta=np.ndarray>
  B04            (time, lat, lon) float32 dask.array<chunksize=(70, 150, 150), meta=np.ndarray>
  ...
  ...
  B05            (time, lat, lon) float32 dask.array<chunksize=(70, 150, 150), meta=np.ndarray>
  cloud_mask      (time, lat, lon) float32 dask.array<chunksize=(70, 150, 150), meta=np.ndarray>
  solar_zenith    (time, lat, lon) float32 dask.array<chunksize=(70, 150, 150), meta=np.ndarray>
  solar_azimuth   (time, lat, lon) float32 dask.array<chunksize=(70, 150, 150), meta=np.ndarray>
  view_zenith     (time, lat, lon) float32 dask.array<chunksize=(70, 150, 150), meta=np.ndarray>
  view_azimuth    (time, lat, lon) float32 dask.array<chunksize=(70, 150, 150), meta=np.ndarray>
Attributes:
  polygon_wkt:  POLYGON ((-88.39261368004998 13.335147549599093, -88.389039...
  index_name:   S2_FULL
  version:     20211108.1

```

Fig 6. Estructura de Cubos de Datos Bajo el Formato NetCDF (Elaboración Propia).

De esta forma se puede observar las distintas variables y coordenadas las cuales se encuentran presentes en los cubos de datos, pudiendo acceder a cada una de ellas mediante la manipulación del cubo como si de matrices se tratara. Al analizar el contenido del cubo, se pudo observar información relacionada a las coordenadas de los puntos de superficie a analizar, así como diferentes fechas que detallan de cierta manera durante qué etapa del año fueron tomadas las capturas satelitales.

Ahora bien, en capítulos anteriores se mencionó un poco sobre lo que eran las imágenes satelitales y su uso en diferentes ámbitos científicos. Para este punto de la investigación, las imágenes satelitales sirvieron tanto para el entrenamiento y prueba de la red neuronal construida como para el análisis mediante los algoritmos de detección de bordes. Sabiendo esto, fue importante comprender y definir la información que contenían los cubos de datos, puesto que se trataba de datos obtenidos del satélite sentinel-2, el punto de partida consistió en conocer con qué bandas de información se contaba para el desarrollo de la investigación, ubicadas en la tabla 1.

Tabla 1. Lista de Bandas de Teledetección del Satélite Sentinel-2 (Fuente: Sentinel Hub)

Banda	Nombre	Uso	Resolución
B01	Aerosol	Detección de Aerosoles	60 m/px
B02	Blue	Útil para la discriminación del suelo y la vegetación, la cartografía de tipos de bosque y la identificación de elementos artificiales	10 m/px
B03	Green	Proporciona un excelente contraste entre el agua clara y el agua turbia (fangosa), penetrando bastante bien en el agua clara	10 m/px
B04	Red	Se refleja fuertemente en el follaje muerto, siendo útil para la identificación del tipo de vegetación, suelos y zonas urbanas	10 m/px
B05	Red Edge	Clasificación de vegetación	20 m/px
B06	Red Edge	Clasificación de vegetación	20 m/px
B07	Red Edge	Clasificación de vegetación	20 m/px
B08	NIR	Buena para cartografiar las costas y el contenido de biomasa	10 m/px
B8A	Red Edge	Clasificación de Vegetación	20 m/px
B09	Water Vapour	Detección de vapor de agua	60 m/px
B11	SWIR 1	Medición del contenido de la humedad del suelo y la vegetación, proporciona un contraste entre los distintos tipos de vegetación	20 m/px
B12	SWIR 2	Medición del contenido de la humedad del suelo y la vegetación, proporciona un contraste entre los distintos tipos de vegetación	20 m/px

Un detalle para destacar de la información incluida en la Tabla 1 es que todas las bandas fueron reproyectadas y alineadas a una resolución de 10 m, de esta manera no sería necesario realizar un preprocesamiento adicional que significase una posible pérdida de información al cambiar las resoluciones de cada una de las bandas.

Otro punto importante para resaltar sobre la información satelital utilizada para el desarrollo de la investigación se trata de las fechas contenidas dentro de los cubos de datos. La coordenada de

“tiempo” es muy importante porque ayuda a conocer y relacionar los datos obtenidos y analizados con las posibles propiedades meteorológicas que puedan llegar a existir en los rangos de fechas que se seleccionaron para el análisis. El resumen de las fechas almacenadas tanto para el área de Santa María, Usulután, como para Zelanda, Países Bajos, puede ser revisado en las tablas 2 y 3 respectivamente.

Tabla 2. Fechas de Análisis de Imágenes Satelitales para Santa María, Usulután, El Salvador

<b>Fechas de Análisis para Santa María, Usulután</b>						
2022-01-01	2022-02-20	2022-04-11	2022-06-10	2022-08-19	2022-10-08	2022-11-27
2022-01-06	2022-02-25	2022-04-16	2022-06-15	2022-08-24	2022-10-13	2022-12-02
2022-01-11	2022-03-02	2022-04-21	2022-06-20	2022-08-29	2022-10-18	2022-12-07
2022-01-16	2022-03-07	2022-04-26	2022-06-25	2022-09-03	2022-10-23	2022-12-12
2022-01-21	2022-03-12	2022-05-06	2022-06-30	2022-09-08	2022-10-28	2022-12-17
2022-01-26	2022-03-17	2022-05-11	2022-07-05	2022-09-13	2022-11-02	2022-12-22
2022-01-31	2022-03-22	2022-05-16	2022-07-10	2022-09-18	2022-11-07	2022-12-27
2022-02-05	2022-03-27	2022-05-21	2022-07-30	2022-09-23	2022-11-12	2023-01-01
2022-02-10	2022-04-01	2022-05-26	2022-08-04	2022-09-28	2022-11-17	2023-01-06
2022-02-15	2022-04-06	2022-06-05	2022-08-14	2022-10-03	2022-11-22	2023-01-11

Tabla 3. Fechas de Análisis de Imágenes Satelitales para Zelanda, Países Bajos

<b>Fechas de Análisis para Zeeland, Países Bajos</b>			
2022-02-14	2022-05-15	2022-08-08	2022-12-01
2022-03-06	2022-06-09	2022-08-13	2022-12-06
2022-03-11	2022-07-04	2022-08-18	2022-12-16
2022-03-21	2022-07-19	2022-09-02	2023-02-14
2022-03-26	2022-07-24	2022-11-01	2023-03-01
2022-04-20	2022-07-29	2022-11-16	2023-04-05

Así, se ha logrado detallar el conjunto de información que sirvió para el desarrollo de la investigación. Sin embargo, todavía hay un conjunto que hace falta definir, y es el que fue utilizado para comprobar la precisión y la eficacia de los modelos desarrollados. Para ello, se contó con una serie de archivos en dos tipos de formatos que contenían lo que se conoce como “Verdad Fundamental”, “Ground Truth” en inglés, el cual es considerado como un término conceptual que se refiere al conocimiento de una verdad sobre una cuestión en concreto, es decir, el resultado ideal esperado (Pickles, 1995). Este término es muy utilizado en modelos estadísticos para probar o refutar hipótesis de investigación.

Para la investigación, se tuvieron archivos que fueron las fuentes principales del *Ground Truth* para el análisis de los resultados de los modelos desarrollados. Por un lado, un *shapefile*, el cual es un formato de archivo simple que sirve para almacenar la ubicación geométrica y brindar atributos de caracteres geográficos, el cual contenía el *Ground Truth* para el área de Santa María, Usulután. Al cargar la información dentro del archivo, se puede apreciar las diferentes parcelas “reales” que se han detectado en la zona respectiva, como se puede apreciar en la figura 7.



Fig 7. *Ground Truth* (Terreno Real) para Santa María, Usulután, El Salvador (Elaboración Propia).

De igual manera, se tiene un archivo de formato Geopackage, el cual es un formato universal construido sobre la base de SQLite ampliamente utilizado para compartir información de datos espaciales vectoriales y raster. Dentro de este archivo, al igual que con el *shapefile*, es posible observar las parcelas “reales” las cuales sirvieron para medir la eficiencia y precisión, su visualización se encuentra en la figura 8.



Fig 8. Porción de *Ground Truth* (Terreno Real) para Zelanda, Países Bajos (Elaboración Propia).

De esta forma fueron definidos los datos a utilizar para el desarrollo y análisis de los modelos seleccionados para la investigación. En base a esto, fue posible comenzar a trabajar y procesar los datos según sea necesario para el avance de cada uno de los desarrollos.

### **3.1.2. Preprocesamiento de Datos**

Antes de comenzar con el desarrollo de los algoritmos y modelos de aprendizaje automático es importante asegurarse de que los datos con los que se estén trabajando no presenten errores o inconsistencias que puedan significar fallos abruptos al momento de correr las implementaciones, es por eso por lo que el preprocesamiento de datos toma un papel muy importante en las fases tempranas de desarrollo.

El preprocesamiento de datos se trata de un proceso que consiste principalmente en tomar los datos brutos de su fuente y convertirlos en datos aptos para que un modelo de aprendizaje automático o algoritmo pueda utilizarlo como entrada inicial. Al comenzar a trabajar con aprendizaje automático no siempre se tiene la suerte de encontrarse con datos limpios y formateados, es por ello por lo que se tienen múltiples tipos de operaciones que son de mucha ayuda para ayudar a reducir y/o eliminar aquellas debilidades que puedan presentarse en el set de datos a utilizar.

Los sets de datos que contiene información del mundo real suelen estar llenos de ruidos, valores faltantes o incluso valores con un formato que difiere del formato principal de todo el set de datos lo que ocasiona que no puedan ser utilizadas para entrenamiento o pruebas de modelos, y es aquí en donde la limpieza de datos y preparación de nuevos valores se convierte en una tarea de mucho peso para este tipo de investigaciones.

#### **A. Limpieza y Construcción de Datos**

El primer paso siempre suele ser la limpieza de datos, la cual consiste en resolver posibles anomalías o datos inexactos que puedan encontrarse en el conjunto de datos a utilizar. Pero esta parte del procesamiento no solamente se limita a corregir formatos o eliminar o reemplazar datos vacíos o erróneos, sino que también se engloba todos aquellos procesos que se encargan de mejorar la calidad de los datos y la información que estos pueden proveer a los modelos o cualquier otro proceso que se encargue de utilizarlos.

Existen múltiples procesos que contribuyen al mejoramiento y limpieza de la información cuando se trabaja con datos satelitales, los cuales se encargan de construir más información en base a las

imágenes satelitales obtenidas o simplemente filtrar información en referencia a fechas o coordenadas específicas.

Gracias a que los cubos de datos fueron proporcionados y construidos utilizando las herramientas que posee el equipo de investigación de Geokapti, varios trabajos que pueden ser incluidos como parte del preprocesamiento de datos no se ven incluidos dentro de esta investigación, sin embargo, es importante mencionarlos de manera que quede registro de qué tipo de transformaciones sufrieron los datos satelitales previo a su uso dentro de la investigación.

Dentro del trabajo de preprocesamiento previo que se realizó, puede recalcarse el proceso de Re-Muestreo y Calibración Radiométrica, los cuales son procesos encargados de mejorar la calidad de los datos satelitales, eliminando ruido y construyendo una serie de datos con características similares. Por un lado, el re-muestreo se encargó de colocar todas las imágenes dentro de una misma resolución, esto para evitar realizar futuros cortes o selecciones de datos y evitando que se pierda información en dichas selecciones o procesos dentro de la elaboración de los modelos. De igual manera, la calibración radiométrica permite convertir cada píxel de la información obtenida directamente de los satélites a un conjunto de datos limpios de cualquier efecto de dispersión o absorción causados por las partículas presentes en la atmósfera. Un ejemplo de esto, se presenta en la figura 9.

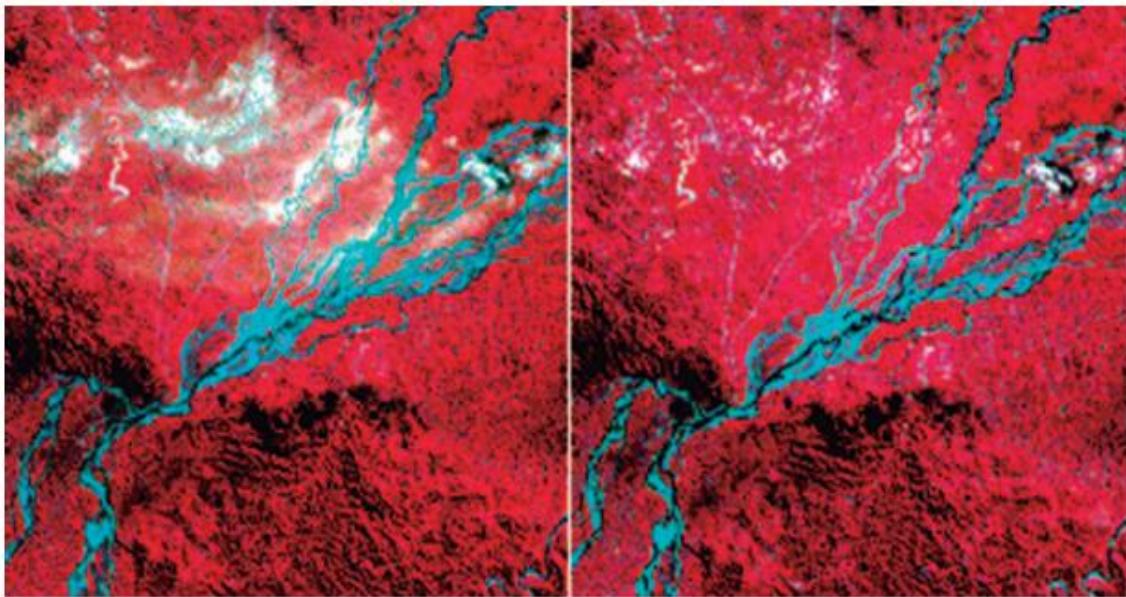


Fig 9. Ejemplo de Mejora de Imagen tras aplicar la Calibración Radiométrica (Fuente: IDEAM, Instituto de Hidrología, Metereología y Estudios Ambientales)

Por otro lado, también se generaron las respectivas máscaras de nubes para cada uno de los cubos de datos, las cuales son de mucha utilidad en relación con la limpieza y mejoramiento de la calidad de los datos, debido a que una porción de tierra que pueda estar cubierta por nubes es una porción

que solamente podría llegar a confundir el modelo en lugar de entrenarlo de la mejor manera. Existen múltiples formas de poder aplicar la máscara de nubes sobre las diferentes bandas, por motivos de simplicidad para la investigación actual, los valores afectados por los píxeles considerados como nubes, fueron sustituidos por el valor medio de cada banda donde se aplica la máscara de nubes, siendo este valor medio calculado en base a las “capturas” obtenidas a partir de cada fecha dentro del cubo de datos, para así no colocar un peso significativo que pueda llegar a afectar al entrenamiento del modelo.

Por último, uno de los procesos más importantes y comunes dentro de la limpieza de datos, se trata de la eliminación o reemplazamiento de valores nulos dentro de las estructuras de datos a utilizar. En el caso de los cubos de datos, fuera de los píxeles reemplazados por la máscara de nubes, los datos se encontraban bastante bien construidos sin dar lugar a algún valor que pueda romper el procesamiento y entrenamiento de los modelos. Sin embargo, los archivos que contienen el *Ground Truth* presentaban ciertas filas de información con valores geométricos nulos, las cuales fueron eliminadas para evitar errores durante el proceso de construcción de la información de entrada de los modelos y los modelos como tal.

## B. Creación de Índices

Así como se detalló en el marco teórico, el uso de índices de vegetación puede llegar a ser de mucha ayuda a la hora de buscar que las características del suelo y de la vegetación como tal sean resaltadas para una mejor capacidad de análisis de estas. Gracias a que los cubos de datos poseen la información necesaria de las bandas utilizadas por el satélite Sentinel-2, fue posible calcular los índices de vegetación seleccionadas para la implementación dentro de la investigación, y en su defecto, del desarrollo de los diferentes modelos de segmentación de parcelas.

Por ello, se hizo una selección de los índices que aportarían mayor y mejor calidad de información para el objetivo de segmentación que se quería alcanzar, los cuales fueron los siguientes:

- Índice de Vegetación de Diferencia Normalizada (NDVI): Su aplicación dentro de la investigación se decidió debido a la naturaleza de la vegetación presente en la principal área de estudio: Santa María, Usulután, la cual es una zona con abundante vegetación y el conocer su dinámica de desarrollo y características resaltables brindaría información importante para que los modelos puedan aprender de sus patrones.

$$\text{NDVI} = (\text{NIR} - \text{RED}) / (\text{NIR} + \text{RED})$$

Ec. 1

- Índice de Vegetación Ajustado al Suelo (SAVI): Se decidió aplicar el SAVI dentro de la investigación para poder resaltar las características del suelo y poder tener una mejor definición de las regiones definidas por las diferentes parcelas agrícolas.

$$SAVI = ((NIR - RED) / (NIR + RED + L)) (1 + L) \quad Ec. 2$$

- El uso del NDWI junto a otros índices como el NDVI, es posible evaluar el contexto de los diferentes sectores y su cambio aparente dentro del área de análisis, identificando cuerpos de agua separándolos de las porciones de tierra.

$$NDWI = (NIR - SWIR) / (NIR + SWIR) \quad Ec. 3$$

### C. Obtención de Máscaras de Bordes

Como último punto de preparación de la información utilizada para la construcción, entrenamiento y análisis de los modelos y sus resultados, se trabajó en la construcción de máscaras de bordes en base al *Ground Truth* definido anteriormente, de esta manera, se tendrá una máscara binaria que servirá para verificar la eficiencia y precisión de las parcelas segmentadas y los bordes que definen cada una de sus áreas.

Para ello, se hizo uso de todas las herramientas de rasterización y detección de bordes que presenta la plataforma QGIS. El siguiente proceso, se realizó tanto como para la región de Santa María, Usulutan, como para Zelanda, Países Bajos. Empezando por Santa María, como primer punto se cargó el archivo *shapefile* como una capa de vector, de esta manera era posible visualizar todos los polígonos que constituyen las parcelas “reales” encontradas dentro de la región de análisis.

Luego de la carga de los polígonos, se utilizó la herramienta de “Bordes” para encontrar cada uno de los bordes de los polígonos pertenecientes al *shapefile* y así tener una idea de cómo se verían separadas cada una de las parcelas sin hacer énfasis en la superficie de estas. Como último paso, se hace la conversión de los vectores, en este caso las líneas, que conforman los bordes de cada una de las parcelas almacenadas. Haciendo uso de la herramienta de Rasterización (Vector a Raster), se especificó que todo lo que se trate de un píxel de borde sea catalogado con un valor de “1.0” y todo lo que no se encuentre sobre ese vector, pasará a ser de valor “0.0”, obteniendo así, una máscara binaria donde los valores diferentes a “0.0” serán considerados como parte de un borde de la parcela agrícola. Los resultados de los bordes y su respectiva máscara están representados en la figura 10.



Fig 10. Máscara de Bordes Reales para Santa María, Usulután, El Salvador (Elaboración Propia).

Para el caso del área de estudio de Zelanda, Países Bajos, el proceso fue casi el mismo, con la única diferencia que debido al tamaño del Geopackage donde se encontraba el *Ground Truth* era mucho más extenso que el área contenida dentro de su respectivo cubo de datos. Por ello, primero se limitaron los polígonos a utilizar para procesar únicamente los que se encontraban dentro de las coordenadas almacenadas dentro del cubo de datos de la región de Zelanda. De igual manera, la máscara obtenida puede visualizarse en la figura 11.



Fig 11. Máscara de Bordes Reales para Zelanda, Países Bajos (Elaboración Propia).

Cada una de estas máscaras fue almacenada como un archivo GeoTiff para su mejor manipulación dentro de cada una de las implementaciones de los algoritmos y los modelos desarrollados. De esta forma, quedan definidos los elementos a utilizar para la construcción de las herramientas para el cálculo y análisis de los bordes resultantes de los procesos de segmentación.

### **3.2. Metodologías Aplicadas para la Detección de Bordes**

Existen muchas técnicas para lograr la segmentación de parcelas agrícolas, una de las tareas más comunes y efectivas que se puede realizar es la detección de bordes de cada una de las parcelas. Al hablar sobre los bordes de la parcela, se hace referencia a la separación y/o transición que existe de un trozo de tierra denominado como una parcela hacia otro punto del terreno que ya sea considerada como una segunda parcela.

Para comenzar, hay que tener claro el concepto de borde, el cual puede definirse como el conjunto de píxeles que forman una recta o curva que se encarga de la separación de regiones distintas dentro de una misma imagen pero con valores diferentes. Una característica importante sobre los bordes es que el cambio de valores de una región a otra debe de darse de una manera drástica en los píxeles o puntos que sean considerados como bordes (Leiva, 2017).

Existen múltiples métodos para lograr la detección de bordes en base a imágenes digitales, en este caso imágenes satelitales, los que se basan en diferentes operaciones matemáticas que se encargan de ir detectando los patrones y/o características necesarias para poder definir si un punto dentro de la imagen se trata de un borde o no.

A continuación, se detallaron los diferentes métodos de detección de bordes que se investigaron y desarrollaron para su aplicación sobre la información satelital obtenida:

#### **3.2.1. Algoritmo de Canny**

El Algoritmo de Canny, también conocido como detector de Canny, se trata de un operador que se basa en un algoritmo multietapa encargado de detectar los bordes de una imagen (Leiva, 2017). Fue desarrollado por John F. Canny, el cual también fue fundador de la teoría computacional de la detección de bordes, la cual describe los siguientes puntos para el desarrollo exitoso de un detector de bordes:

1. El algoritmo debe de ser capaz de detectar la mayor cantidad de bordes que le sea posible con la mayor precisión.
2. Los puntos de bordes detectados deben de ser localizados en el centro del borde.
3. No se debe caer en la creación de bordes falsos, cada borde debe de ser detectado una y solo una vez.

Actualmente existen múltiples implementaciones del algoritmo de Canny presentes en diferentes paquetes disponibles para el lenguaje de programación Python. Para el desarrollo de la investigación, se hizo uso de una de esas implementaciones presente en la biblioteca de Python OpenCV, sin embargo, todavía había un proceso que seguir con el fin de preparar la información utilizada como datos de entrada para la implementación.

Así, los pasos a seguir constan de la selección de una o múltiples bandas que servirán como punto de partida para construir la entrada para el algoritmo de Canny. Haciendo uso de las características de los cubos de datos, es posible obtener bandas únicas o calcular índices en base a los valores almacenados.

Como segundo punto, se comienza la preparación de la información la cual siguiendo con los pasos de

desarrollos generales para la implementación del algoritmo de Canny, consiste en la normalización de

los datos, seguido de un suavizado en base a un filtro Gaussiano que puede tener diferentes propiedades, siendo el paso final, el uso de la implementación del algoritmo de Canny a utilizar.

El resumen de la metodología que se siguió para la implementación del algoritmo de Canny puede encontrarse en la figura 12.

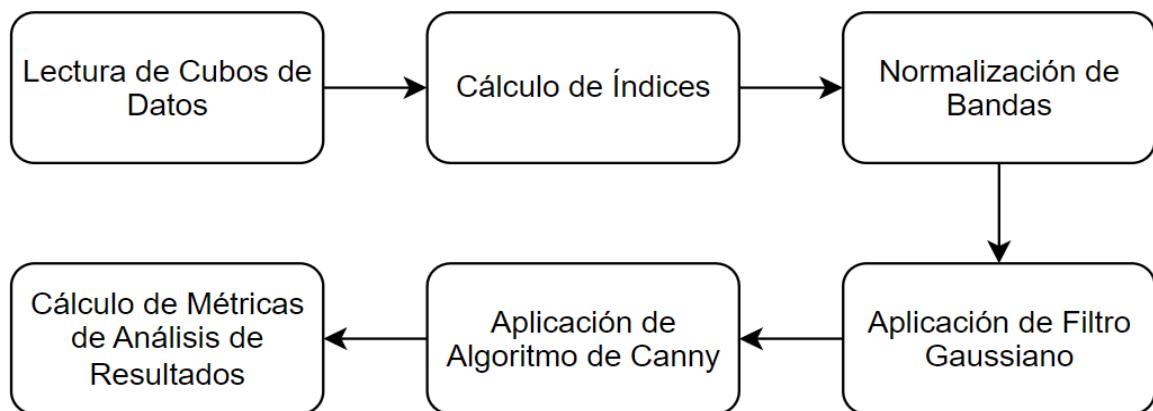


Fig 12. Metodología para Aplicación del Algoritmo de Canny.

### 3.2.2. Red Neuronal

Tal y como se definió en apartados anteriores de este documento, una red neuronal consiste en una gran estructura de capas las cuales se alimentan de grandes cantidades de información, que puede ser cualquier tipo de información y cualquier tipo de dato, y cuyo objetivo principal es el de simular lo más que se pueda el nivel de aprendizaje y funcionamiento a nivel del cerebro

humano. En el campo de la Inteligencia Artificial (AI) y Machine Learning (ML) se abre un amplio abanico de posibilidades y campos de aplicación en los que se pueden implementar redes neuronales y así encontrar soluciones más eficientes e innovadoras. Para las tareas de segmentación de imágenes y teledetección se recomienda hacer uso de redes neuronales ya que se sabe que la información a entrenar tiende a ser en grandes cantidades, lo que propicia un mejor aprendizaje de la red neuronal y con ello, la obtención de resultados más precisos; además de contar con una gran variedad de arquitecturas que se pueden implementar las cuales funcionan mejor o peor dependiendo del escenario de estudio y el tipo de información que se procesa.

Habiendo definido estos conceptos, durante la investigación en cuestión, uno de los objetivos principales es el de utilizar un tipo de red neuronal capaz de segmentar y detectar límites de parcelas agrícolas en determinadas zonas del territorio salvadoreño. Sin embargo, el reto más grande consiste en elegir la arquitectura y modelo de red neuronal que mejor se adapte a las necesidades y que mejores resultados arroje al final de todo el proceso.

Por ello, se presenta a continuación de forma más detallada tanto el tipo de red neuronal elegida para la investigación como el tipo de arquitectura a utilizar y todo lo que esto implica.

En la presente investigación se decidió trabajar el tema de segmentación automática de parcelas agrícolas haciendo uso de una red neuronal convolucional, la cual consiste en un tipo de red neuronal con características bien definidas, por mencionar algunas: la característica especial en los datos de entrada (inputs) que permiten simplificar las arquitecturas de la red, reduciendo en gran medida el número de variables de entrada. Además de contar con capas más definidas en las cuales la pérdida de información es poca o nula ya que el tránsito de la información es directo y manteniendo los datos originales de principio a fin.

Como se mencionó en el marco teórico, las capas principales de una red neuronal convolucional son tres: la capa convolucional o capa de entrada, la capa de agrupación y la capa totalmente conectada o capa de salida. En la primera capa la red neuronal se encarga de detectar y extraer diferencias simples (formas simples, colores, bordes, etc.) del conjunto de datos, en este caso, de imágenes satelitales que contienen toda la información de las parcelas agrícolas en determinada área. En cambio, la capa final se encarga de detectar características más complejas (bordes irregulares, colores y formas extraídas a partir de la aplicación de máscaras, etc.), las cuales al final del proceso le permiten separar las imágenes, definirlas y finalmente clasificarlas. Cabe mencionar que, para obtener mejores resultados, el conjunto o conjuntos de datos que se entrena en una red neuronal deben de contar con un preprocesamiento previo en el cual se descarte todo tipo de información nula o ambigua que pueda entorpecer el aprendizaje de la red neuronal.

En casos como el de nuestra investigación, se vuelve necesario descartar aquellas imágenes satelitales que, por las condiciones climáticas existentes al momento de ser capturadas las imágenes (nubosidad, gases, entre otros), no se puedan apreciar los terrenos agrícolas, ya que la red neuronal no se entrenará de forma correcta con este tipo de imágenes. Para el caso del entrenamiento de la región de Santa María, se descartaron 280 imágenes, quedando para entrenamiento y pruebas 840 imágenes. Por otro lado, para el caso de Zelanda se descartaron 400 imágenes, dejando 1458 imágenes válidas. Además, es posible y a la vez muy importante aplicar diferentes técnicas de filtros o máscaras a las imágenes ya que eso permite que los algoritmos aprendan y detecten los límites de parcelas agrícolas en distintos escenarios, lo que a la larga se traduce en resultados más precisos y eficientes. Con esto se espera obtener como resultado todos y cada uno de los terrenos dentro de un área determinada ya segmentados y definidos con sus respectivos bordes. Las áreas de estudio pueden apreciarse en la figura 13.

En cuanto a la arquitectura elegida para la implementación de la red neuronal se eligió el modelo U-Net, una de las arquitecturas de red neuronal más utilizadas en el campo de la segmentación de imágenes, ya que es un modelo totalmente convolucional. El modelo adopta su nombre dado que es una arquitectura de dos vías. La primera es una vía de contracción llamada “codificador” la cual consiste en varias capas de convolución y, en el contexto de análisis semántico de imágenes, se encarga de tomar el conjunto de imágenes, crear mapas con sus respectivas características y reducir el tamaño de estas para reducir los parámetros de la red neuronal. Por otra parte, pero muy relacionada a la primera, se tiene la segunda vía llamada “decodificador”, cuyo objetivo es proyectar las características aprendidas por el codificador en el espacio de píxeles y así obtener una clasificación densa. Cabe mencionar que, en lenguaje más técnico, los desarrolladores explican el modelo U-net como dos fases: una descendente (downsampling) la cual hace referencia a la primera vía en la que el codificador hace su trabajo y por otro lado una fase ascendente (upsampling) en la que el proceso comienza a mostrar los resultados y de ahí la razón del porqué se llama “modelo U-net”. En el anexo B se puede apreciar de forma visual la forma de operar de la arquitectura de una red neuronal con el modelo U-net.

La ejecución e implementación de esta red neuronal basada en la arquitectura y modelo U-net pretende mostrar resultados del proceso de segmentación y delimitación de límites de parcelas agrícolas en el territorio de Santa María, pueblo perteneciente al departamento de Usulután, El Salvador; de forma precisa y eficiente. Para poder llevar a cabo dicha ejecución, fue necesario entrenar con anticipación a la red neuronal con una fuente de datos que contuviera imágenes satelitales de parcelas agrícolas similares a las existentes en el territorio salvadoreño, así la red neuronal sería capaz de “alimentarse”, aprender con anticipación y realizar predicciones con

resultados más precisos. Por ello, para la fase de entrenamiento se contó con un cubo de datos que contenía imágenes satelitales provenientes de la provincia de Zeeland, Países Bajos y que fueron extraídas del satélite Sentinel-2. Asimismo, la información satelital de las parcelas de Santa María, Usulután fueron extraídas del Sentinel-2.



Fig 13. Áreas de Estudio: Municipio de Santa María, Usulutan, El Salvador, y región de Zelanda, Países Bajos  
(Elaboración Propria).

La planeación y preparación de los conjuntos de datos (limpieza de datos, preprocesado, creación/uso de máscaras, entre otras aplicaciones) así como la ejecución e implementación de la red neuronal con sus respectivas fases (prueba, entrenamiento y predicciones) se explica de forma más detallada en los siguientes capítulos de este documento. A manera de resumen, se tiene en la figura 14 un gráfico de la metodología a seguir para la implementación de la red neuronal.

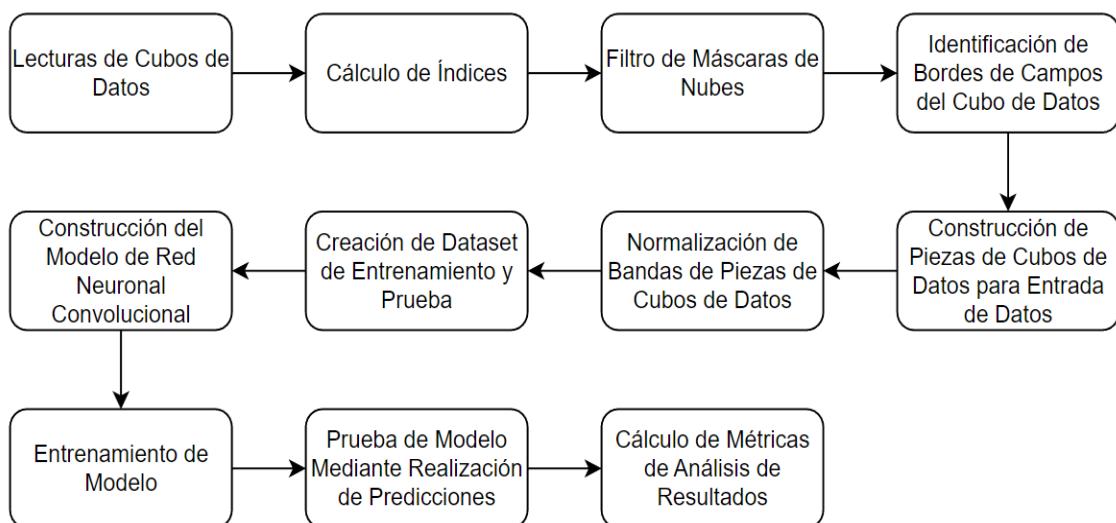


Fig 14. Metodología para Aplicación de la Red Neuronal Convolutacional

### **3.3. Análisis de Resultados: Métricas de Análisis de Datos**

Habiendo definido ya el camino para el desarrollo y construcción de los modelos de segmentación de parcelas agrícolas también es importante hacer hincapié en el desarrollo y aplicación de técnicas para poder analizar los datos obtenidos, y así poder definir si los modelos construidos presentan un buen rendimiento y precisión a la hora de realizar la segmentación de parcelas agrícolas.

Para ello, existen múltiples conceptos y fórmulas que ayudan a realizar un análisis de los resultados obtenidos y poder clasificar los modelos o algoritmos desarrollados en base a valores cuantitativos. Para el desarrollo de la presente investigación, seleccionaron cuatro diferentes métricas las cuales sirvieron para llevar un control y puntaje de los valores obtenidos por los modelos.

#### **3.3.1. Precisión y Exhaustividad (Precision and Recall)**

Naturalmente, en los problemas de clasificación como lo es la segmentación de parcelas agrícolas, se definen valores que pueden ser o no considerados por las múltiples clases que posea el modelo. En el caso de la detección de bordes, estas clases se reducen en que si un elemento o píxel analizado pueda ser clasificado como borde o no. Sin embargo, al estar trabajando en la construcción, entrenamiento y pruebas de modelos, pueden encontrarse valores que puedan aparecer un valor, cuando en realidad se refieren a otros. La existencia de los valores de Falsos Positivos y Falsos Negativos en los resultados de los modelos de desarrollo puede llegar a afectar el rendimiento de estos, debido a que existencia puede llegar a confundir el modelo creando escenarios en donde la clasificación pueda verse afectada de manera negativa a como en un principio podría verse (Koehrsen, 2023).

La Exhaustividad (Recall), la habilidad del modelo de identificar los casos más relevantes dentro del conjunto de datos, junto con la Precisión (Precision), la habilidad de encontrar los puntos específicos que son más relevantes dentro de los resultados del modelo, son dos métricas muy importantes que contribuyen a la identificación de estos valores que ensucian los resultados obtenidos de los modelos.

$$Precision = \frac{Verdaderos\ Positivos}{Verdaderos\ Positivos + Falsos\ Positivos} \quad Ec.\ 4$$

$$Recall = \frac{Verdaderos\ Positivos}{Verdaderos\ Positivos + Falsos\ Negativos} \quad Ec.\ 5$$

Mediante las fórmulas definidas en la Ecuación 1 y Ecuación 2, fue posible realizar un análisis y búsqueda dentro de los resultados obtenidos de todos aquellos valores que fueron tomados positivos cuando deberían de serlo, y con qué frecuencia se detectó un valor positivo sin importar de que si este fuera verdadero o no. Los valores tanto para la Precisión como la Exhaustividad pueden tener valores entre 0 y 1, y así la combinación de estas dos métricas pueden tener múltiples interpretaciones las cuales dan a lugar a conclusiones sobre el comportamiento de los resultados obtenidos.

### 3.3.2. Puntuación F1 (F1 Score)

Habiendo definido y demostrado el valor de análisis que aportan métricas como la Precisión y la Exhaustividad, existe otra métrica la cual combina ambos análisis dentro de uno solo. La Puntuación F1 (F1 Score) se trata de la media armónica calculada en base a los resultados obtenidos del análisis de la Precisión y la Exhaustividad. Al decir que se trata de una media armónica, se refiere a que dentro de todos los resultados obtenidos, esta tendrá una tendencia de acercarse a los últimos valores obtenidos, logrando así mitigar el ruido que puedan causar los valores atípicos presentes en un conjunto de resultados, pagando con el hecho de que los valores pequeños tienen a tener un gran impacto sobre la métrica (Chauhan, 2020).

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad \text{Ec. 6}$$

Combinando mediante operaciones los valores obtenidos de la Precisión y la Exhaustividad como se puede apreciar en la ecuación 3, es posible obtener una nueva métrica de análisis muy útil para cuando se tiene grandes cantidades de información como es el caso de la investigación, ya que ayuda a determinar el equilibrio que existe entre los valores que se encarga de definir la cantidad de casos verdaderos, o exitosos, que se lograron predecir con los modelos desarrollados.

A pesar de que no existe un rango “estándar” de los valores óptimos que la Puntuación F1 debe de tomar para ser considerado un buen modelo, si es posible recalcar que ya que estos valores se mueven dentro del rango de 1 y 0, un valor obtenido cercano o igual a 1 sirve para considerar un modelo con un buen nivel de precisión y exhaustividad, es decir, menor número de falsos positivos y falsos negativos se encontraron, caso contrario cuando los valores son cercanos a 0, donde la precisión y la exhaustividad presentan un gran espacio de mejora.

### **3.4. Herramientas a Utilizar**

Para el desarrollo del trabajo de investigación se hizo uso de las siguientes herramientas de trabajo:

#### **3.4.1. Herramientas de desarrollo**

- **QGIS**

QGIS es un Sistema de Información Geográfica profesional de fácil uso, gratis y de código abierto, que posibilita la creación, visualización, análisis, edición y publicación de información geoespacial.

Al igual que los demás Sistemas de Información Geográfica existentes, QGIS permite la creación de mapas con numerosas capas que pueden ser ensambladas bajo diferentes formatos, dependiendo de la aplicación.

- **Python**

Python es un lenguaje de programación ampliamente utilizado en las aplicaciones web, el desarrollo de software, la ciencia de datos y el machine learning (ML). Python es eficiente y además se puede ejecutar en muchas plataformas diferentes.

- **Paquetes de Python**

**A. Tensorflow:** Se trata de una librería de código libre para Machine Learning (ML). Fue desarrollado por Google para satisfacer las necesidades a partir de redes neuronales artificiales. TensorFlow permite construir y entrenar redes neuronales para detectar patrones y razonamientos usados por los humanos. Además de trabajar con redes neuronales, TensorFlow es multiplataforma. Trabaja con GPUs y CPUs e incluso con las unidades de procesamiento de tensores (TPUs).

**B. Keras:** Es una de las API de redes neuronales más utilizadas para el desarrollo y las pruebas de redes neuronales. Facilita enormemente la creación de capas para las redes neuronales o la configuración de arquitecturas complejas.

Un modelo Keras consiste en una secuencia o un gráfico independiente. Hay varios módulos totalmente configurables que pueden combinarse para crear nuevos modelos. Una de las ventajas de este es la modularidad que presenta, lo que resulta en una alternativa muy fácil para añadir nuevas funcionalidades como módulos independientes. Keras es, por tanto, muy flexible y adecuada para la investigación y la innovación.

- C. **XArray:** es una librería popular para el uso del análisis de datos de las geociencias por la naturaleza multidimensional de los datos en los que se trabajan (dimensiones espaciales, datos como precipitaciones, temperaturas, dimensiones temporales, etc.).
- D. **Salem:** Es una pequeña biblioteca para procesar y trazar datos geocientíficos. Extiende xarray para agregar subconjuntos geolocalizados, enmascaramiento y operaciones de trazado a las estructuras DataArray y DataSet de xarray.
- E. **Matplotlib:** Es una librería open source que pertenece a Python en la que se pueden crear visualizaciones animadas, estáticas e interactivas en Python. Con esta librería se pueden crear trazados, histogramas, diagramas de barra y otros tantos tipos de gráficas con poco código.
- F. **SKLearn:** Es una librería de código abierto para Python. Cuenta con algoritmos de clasificación, regresión, clustering y reducción de dimensionalidad. Además, presenta la compatibilidad con otras librerías de Python como NumPy, SciPy y matplotlib.

La gran variedad de algoritmos y utilidades de Scikit-learn la convierten en la herramienta básica para empezar a programar y estructurar los sistemas de análisis de datos y modelado estadístico.

- G. **Numpy:** Es una librería de Python especializada en el cálculo numérico y el análisis de datos, especialmente para un gran volumen de datos. Incorpora una nueva clase de objetos llamados arrays que permite representar colecciones de datos de un mismo tipo en varias dimensiones, y funciones muy eficientes para su manipulación.
- H. **Rasterio:** Es una librería de Python que permite la lectura, inspección, visualización y la escritura de raster geoespaciales. La librería usa rasters en formato GeoTIFF y otros formatos y es capaz de trabajar con imágenes

satelitales, modelos de elevación digital, productos de imágenes, e imágenes procesadas de dron.

- I. **Skimage:** o también conocida como *scikit-image*, es una librería de código abierto, la cual nos proporciona herramientas y algoritmos para el procesamiento de imágenes digitales. Nos proporciona funcionalidades como la manipulación, mejora y análisis de las imágenes. Esta librería está basada en Numpy.
- J. **NetCDF4:** Es una librería de código abierto que permite leer, escribir, y manipular archivos “Network Common Data Form” o netCDF, esta librería está basada en NetCDF, incluyendo más funcionalidades y otras mejoras.
- K. **Geopandas:** Es un proyecto de código abierto que agrega soporte para datos geográficos a objetos pandas. Los objetos de GeoPandas pueden actuar sobre objetos shapely y realizar operaciones geométricas. Geopandas depende además de Fiona para el acceso a archivos y matplotlib para trazar.
- L. **OpenCV:** OpenCV es una biblioteca de código abierto de Python que permite realizar tareas de procesamiento de imágenes y visión por ordenador. Esta biblioteca fue clave para el desarrollo del modelo en base al algoritmo de Canny, puesto a que presenta una implementación lista para usar del mismo.  
(Implementación del Algoritmo de Canny:  
[https://docs.opencv.org/4.x/da/d22/tutorial\\_py\\_canny.html](https://docs.opencv.org/4.x/da/d22/tutorial_py_canny.html))

- **Google Collaboratory**

Google Collaboratory es una herramienta que permite ejecutar scripts de Python a través de los servidores de Google. Esto permite ejecutar celdas de código como si se tratara de un cuaderno de Jupyter Notebook. Es ideal para implementar algoritmos de aprendizaje máquina, ya que no te limita a los recursos de la computadora, si no que brinda el GPU y las TPU's de Google para potencializar el cómputo del proyecto.

- **Máquina virtual**

Los procesos y metodologías desarrollados durante la investigación tienen la característica de ser procesos pesados y de mucha duración, por lo que el uso de un equipo de buena calidad y bastante potencia era una herramienta esencial para el buen desarrollo de la investigación. Para ello, la Universidad Centroamericana “José Simeón Cañas” proporcionó una máquina virtual capaz de soportar procesos complejos y de mucha duración, lo cual fue un gran apoyo para los

entrenamientos y pruebas de los modelos desarrollados. Las especificaciones de la máquina virtual son las siguientes: 64 GB de memoria RAM, 8 núcleos de procesamiento, sistema operativo Ubuntu Linux 20.04, con acceso a los siguientes puertos HTTP: 22, 443, 80, 8080, 8081).

De igual manera, se utilizaron herramientas de apoyo para el manejo de la máquina virtual, estas fueron las siguientes:

**A. Putty**

PuTTY es un cliente SSH y Telnet con el que es posible conectarse a servidores remotos iniciando una sesión en ellos y permitiendo ejecutar comandos o programas.

**B. Filezilla**

FileZilla es un software gratuito y de código abierto que funciona a nivel cliente/servidor. Significa, que los usuarios, pueden conectarse a un servidor para consultar, adquirir y manipular contenido del mismo. Resulta realmente útil si se busca mantener una conexión directa a una fuente de datos sin perder excesivo tiempo en el proceso.

### **3.4.2. Herramientas de apoyo**

- Google Drive**

Google Drive es una plataforma gratuita que permite almacenar archivos y acceder a ellos desde cualquier lugar utilizando la nube, allí también se pueden crear documentos, hojas de cálculo, presentaciones y más.

- Trello**

Trello es una herramienta visual que permite gestionar cualquier tipo de proyecto y flujo de trabajo, así como supervisar tareas. Añadiendo archivos, checklists o incluso automatizaciones.

## CAPÍTULO 4. DESARROLLO Y CONSTRUCCIÓN DE METODOLOGÍAS PARA SEGMENTACIÓN DE PARCELAS

### 4.1. Construcción y Desarrollo de Metodologías

#### 4.1.1 Algoritmo de Canny

Para la construcción y desarrollo del algoritmo de Canny se hizo uso del lenguaje de programación Python, utilizando el entorno de desarrollo de Visual Studio Code. En primer lugar se importaron las librerías de Python: Matplotlib, netCDF4, skimage, y cv2. Las importaciones se encuentran en la figura 15.

```
import cv2
from skimage import io
import netCDF4
import matplotlib.pyplot as plt
```

Fig 15. Importación de paquetes utilizados para la construcción del algoritmo.

Como primer paso, como se ve en la figura 16, se carga el cubo de datos del municipio de Santa María, utilizando la librería netCDF4, para poder manipularla. Dicho archivo se encuentra almacenado en la misma carpeta del código.

```
# Carga el archivo NetCDF
dataset = netCDF4.Dataset('./STAMARIA_S2_2023_cube.nc');
```

Fig 16. Lectura del archivo con la información del municipio de Santa María.

Luego en la figura 17, del dataset, se toma una sola banda, para la aplicación de este algoritmo, se tomó la decisión de hacer uso de la banda B8 (Infrarrojo cercano), ya que esta banda brinda información útil de sobre la vegetación de la zona, y poder distinguir entre zonas de cultivo, y zonas con mucha vegetación sana.

```
# Tomando una banda
B0_8 = dataset.variables['B08'][0, 0:, 0:];
```

Fig 17. Toma de banda B08 del cubo de datos

Una vez que se toma la banda deseada, se procede con la normalización de los datos, haciendo uso de la función **normalize()** de la librería cv2. Se le pasa como argumento la matriz de entrada **B0\_8**, la cual es la que se desea normalizar. Y el resultado se almacena en la variable **datos\_norm**. El objetivo de normalizar la matriz es tener los valores de los píxeles de la imagen en un rango más fácil de manipular, en este caso, su valor se normaliza entre 0 - 255. Dicho proceso se encuentran en la figura 18.

```
datos_norm = cv2.normalize(B0_8, None, 0, 255, cv2.NORM_MINMAX, cv2.CV_8U);
```

Fig 18. Normalización del valor de los píxeles de la banda escogida.

Una vez obtenidos los datos normalizados, se procede a aplicar un suavizado Gaussiano para la eliminación de ruido de la imagen. Esto se observa en la figura 19.

```
# Aplicar suavizado Gaussiano  
gauss = cv2.GaussianBlur(datos_norm, (5,5), 0);
```

Fig 19. Aplicación del suavizado Gaussiano.

La detección de los bordes de la imagen se realiza por medio de la función *Canny()*, la cual recibe como argumento de entrada la matriz recién suavizada por el filtro gaussiano, y se definen los umbrales inferior y superior para la detección de bordes. En un principio se fueron probando a partir valores estándar para el umbral inferior, y para el valor del umbral superior se multiplicaba por tres el valor del umbral inferior. Este proceso se fue repitiendo, aumentando el valor de ambos umbrales hasta llegar a un resultado visualmente satisfactorio. La aplicación del algoritmo de Canny se encuentra en la figura 20 y su resultado puede apreciarse en la figura 21.

```
# Detectamos los bordes con Canny  
canny = cv2.Canny(gauss, 50, 150) # Imagen resultante del mapa de contornos
```

Fig 20. Aplicación del algoritmo de Canny

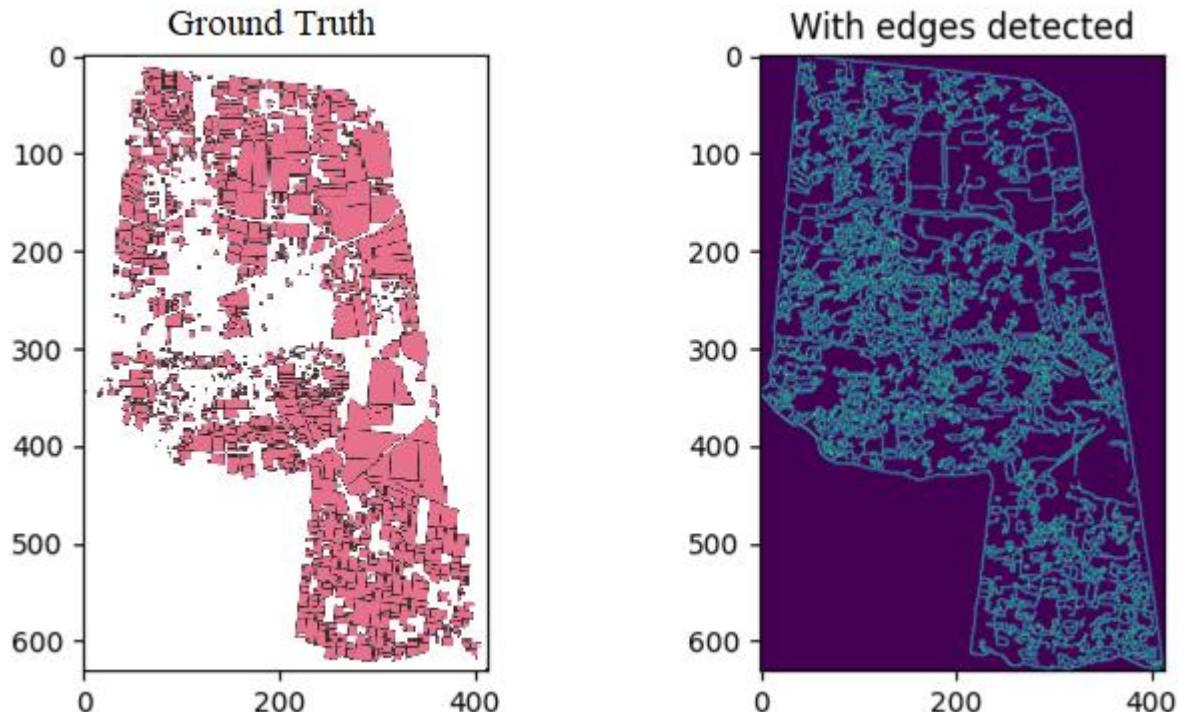


Fig 21. Resultado de la detección de bordes con el algoritmo de Canny

El siguiente paso del algoritmo es encontrar los contornos de la imagen. Para ello, se hace uso de la función *findContours()* de la librería cv2. Esta recibe la matriz de bordes obtenida anteriormente, la cual se procesa para encontrar sus contornos. También se especifica la forma en que estos contornos serán extraídos y aproximados. Con los modos **RETR\_EXTERNAL**, y **CHAIN\_APPROX\_SIMPLE**, los contornos que se extraerán serán solamente los externos de los objetos de la imagen, y de estos se guardan todos los puntos que fueron detectados de la imagen, esto para poder obtener una imagen más detallada como resultado final. Esto se ve en la figura 22.

```
# Buscamos los contornos  
(contornos,_) = cv2.findContours(canny,cv2.RETR_EXTERNAL,  
cv2.CHAIN_APPROX_NONE)
```

Fig 22. Búsqueda de contornos

Finalmente, se dibujan los contornos sobre la imagen original guardada en la variable **B0\_8**. Se especifica en la función *drawContours()* que se deben de dibujar todos los contornos, el color, y el grosor de estos sobre la imagen original, como se ve en la figura 23 y su resultado en la figura 24. La implementación completa se encuentra en el anexo E.

```
cv2.drawContours(B0_8,contornos,-1,(0,0,255), 1)
```

Fig 23. Aplicación de contornos sobre la imagen original

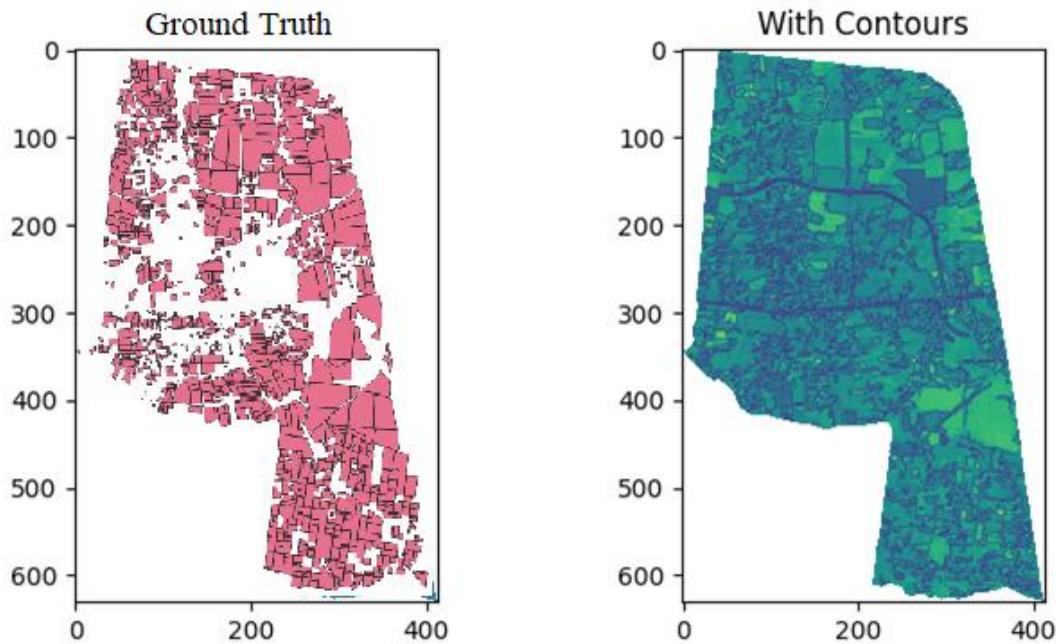


Fig 24. Resultados y comparación del mapa con bordes.

#### 4.1.2. Red Neuronal

El trabajo de construcción e implementación de la red neuronal en base a la metodología definida en el capítulo 3, puede dividirse en tres sectores: La obtención y preparación de la información de entrenamiento y prueba, la construcción del modelo en base a la arquitectura de la red neuronal definida en el Anexo B, y por último, el entrenamiento y obtención de predicciones hacia el modelo.

El primer sector se basa en la obtención de los datos que servirán como entrada tanto para el entrenamiento del modelo como para las predicciones sobre el mismo. Como primer punto, se definió el entorno de desarrollo, instalando e importando cada uno de los paquetes que serán utilizados para el manejo de la información, creación de estructuras y métodos de utilidad. Los paquetes utilizados se observan en la figura 25.

```

import numpy as np
import geopandas as gpd
import xarray as xr
import rasterio
import tensorflow as tf
from sklearn.model_selection import train_test_split
from tensorflow.keras import layers

```

Fig 25. Importación de paquetes para el desarrollo de la Red Neuronal Convolutacional

Tras haber definido e importado todos los paquetes necesarios, lo siguiente fue obtener toda la información referente a las imágenes satelitales y las máscaras que poseía el *Ground Truth*. Para ello, tal y como se ve en la figura 26, haciendo uso del método *open\_dataset* del paquete xarray y el método *open* del paquete Rasterio, fue posible crear instancias capaces de almacenar la estructura de los cubos de datos de Santa María y Zelanda, al igual que se obtuvieron instancias de las máscaras binarias que serán utilizadas para el análisis y verificación de resultados. Por último, se obtuvo la instancia del *shapefile* que posee los polígonos “reales” para los campos de Santa María, utilizando a su vez el método *dropna* del paquete geopandas para limpiar la información obtenida eliminando aquellas filas de datos que no poseen un valor para la geometría del polígono.

```

datacube = xr.open_dataset("data/sta_maria/STAMARIA_S2_2023_cube.nc", chunks=150)
datacube_zeeland = xr.open_dataset("data/europe/zeeland_s2_mcclouds_marea_cube.nc", chunks=150)
mask_zeeland = rasterio.open('data/europe/europe_mask.tif')
mask_stmaria = rasterio.open('data/sta_maria/stamaria_mask.tif')
y_true_stmaria = mask_stmaria.read(1)
y_true_eu = mask_zeeland.read(1)
shapefile = gpd.read_file("data/sta_maria/fields.shp")
shapefile = shapefile.dropna(subset=['geometry'])

```

Fig 26. Obtención de cubos de datos y máscaras de parcelas reales.

Con las instancias de ambos cubos listas para ser utilizadas, fue posible acceder a todas las variables y propiedades que los cubos poseían. Primero, se obtuvieron listas para las fechas presentes dentro de los cubos, esto para poder almacenarlas para su posterior uso para obtener más información de un mismo cubo. Este proceso se encuentra en la figura 27.

```

dates = datacube['time']
dates = dates[0:70]

dates_eu = datacube_zeeland['time']
dates_eu = dates_eu[0:24]

```

Fig 27. Obtención de valores de fechas de análisis para ambas áreas de estudio

Ahora, se definieron una serie de variables de apoyo. Las variables pertenecientes a la familia *dc\_shape*, *lat\_len* y *lon\_len*, se utilizaron para almacenar las dimensiones de cada uno de los cubos de datos, y a la vez, la variable *image\_size* se encargaba de especificar el tamaño de cada una de las imágenes que servirán como entrada para el modelo construido. Por último, se definieron múltiples listas que tiene como objetivo almacenar los trozos de cubos de datos, la lista de máscaras de los respectivos trozos de los cubos de datos y una lista que almacena las imágenes completas de las máscaras respectivamente, como se ve en la figura 28.

```

dc_shape = datacube["B01"].shape
lat_len = dc_shape[1]
lon_len = dc_shape[2]

dc_shape_eu = datacube_zeeland["B01"].shape
lat_len_eu = dc_shape_eu[1]
lon_len_eu = dc_shape_eu[2]

image_size = 128

datacube_subset_list = []
border_subset_list = []
all_boundaries_list = []

```

Fig 28. Definición de variables de apoyo: Dimensiones de cubos de datos y de imágenes de análisis

Luego se prosiguió a calcular cada uno de los índices que se agregaron a los cubos de datos para ser utilizados como información adicional con la cual alimentar al modelo desarrollado. Tal y como se definió en el capítulo 3, se calcularon y utilizaron los índices: NDVI, NDWI y SAVI, siguiendo el proceso que dicta la figura 29.

```

nir = datacube["B08"]
red = datacube["B04"]
blue = datacube["B02"]
green = datacube["B03"]

ndvi = (nir - red) / (nir + red)
ndwi = (green - nir) / (green + nir)

L = 0.5
savi = ((nir - red) / (nir + red + L)) * (1 + L)

datacube = datacube.assign(NDVI=ndvi)
datacube = datacube.assign(NDWI=ndwi)
datacube = datacube.assign(SAVI=savi)

```

Fig 29. Cálculo de índices de vegetación

A su vez, se trabajó en la sustitución de los pixeles cubiertos por nubes en cada una de las bandas a utilizar utilizando el valor promedio presente en cada una de las bandas como sustituto, de esta manera, se asemejarían más a píxeles de terreno y no a nubes que podrían confundir a la red neuronal. Nuevamente, eso puede verse en la figura 30.

```

cloud_mask = datacube['cloud_mask']

band_2 = datacube['B02']
mean_2 = np.nanmean(band_2)
masked_2 = np.where(np.isnan(cloud_mask), band_2, mean_2)
band_2.values = masked_2

band_3 = datacube['B03']
mean_3 = np.nanmean(band_3)
masked_3 = np.where(np.isnan(cloud_mask), band_3, mean_3)
band_3.values = masked_3

band_4 = datacube['B04']
mean_4 = np.nanmean(band_4)
masked_4 = np.where(np.isnan(cloud_mask), band_4, mean_4)
band_4.values = masked_4

band_8 = datacube['B08']
mean_8 = np.nanmean(band_8)
masked_8 = np.where(np.isnan(cloud_mask), band_8, mean_8)
band_8.values = masked_8

datacube['B02'] = band_2
datacube['B03'] = band_3
datacube['B04'] = band_4
datacube['B08'] = band_8

```

Fig 30. Sustitución de pixeles cubiertos de nubes

El siguiente paso fue calcular los bordes para la información contenida dentro de los cubos de datos, de esta manera se podrían obtener las respectivas máscaras binarias que representan cada uno de los bordes respectivos a las fechas presentadas en los cubos. En la figura 31, solo se aprecia el proceso realizado sobre el cubo de datos de Zelanda, pero el algoritmo construido también fue utilizado para obtener las máscaras respectivas para la región de Santa Marfa. Utilizando las fechas obtenidas anteriormente, por cada uno de los polígonos dentro del *shapefile* se selecciona su geometría presente dentro del cubo de datos, la cual es utilizada para poder aplicar una convolución con un kernel 3x3 para aumentar los pixeles de la geometría original, siendo este

aumento el borde detectado. Luego, solo hace falta eliminar la geometría original para obtener solamente la máscara de bordes de la geometría extraída.

```
datacube_zeeland = datacube_zeeland.assign(ONES=lambda x: 1 * np.isnan((x.B02 * np.nan)))
for date in dates:
    all_boundaries = np.full((datacube.dims['lat'], datacube.dims['lon']), False)
    for field_geo in shapefile.geometry:
        datacube_roi = datacube.sel(time=date).salem.roi(geometry=field_geo, other=0.0)

        o_fields_zero_one = datacube_roi['ONES']

        conv_1_kernel = np.ones((3, 3))
        big_field = convolve2d(o_fields_zero_one, conv_1_kernel, 'same')

        big_field_zero_one = (big_field > 0) * 1.0

        field_boundary = (big_field_zero_one - o_fields_zero_one) == 1

        all_boundaries = np.logical_or(all_boundaries, field_boundary)

    all_boundaries_list.append(all_boundaries)
```

Fig 31. Obtención de máscaras de bordes de entrenamiento

De esta forma se posee tanto los cubos de datos como las respectivas máscaras para cada una de las fechas analizadas. A partir de esto, se definen las bandas que no serán almacenadas en los trozos de cubos para mejorar el rendimiento ignorando información que no se utilizará. El proceso para seguir es construir “trozos” de los cubos de datos originales que sean de tamaño 128x128 ya que este es el tamaño de imágenes que espera el modelo recibir. La principal razón por la cual se seleccionó este tamaño para las imágenes de entrada fue debido a que no se poseía un gran volumen de datos de prueba, por lo al utilizar imágenes más pequeñas fue posible abarcar más capturas de menor tamaño del territorio total a analizar. Esto puede apreciarse en la figura 32.

```
var_to_drop = ["cloud_mask", "solar zenith", "solar_azimuth", "view zenith",
               "view azimuth", "B01", "B05", "B06", "B07", "B09", "B8A", "B11", "B12"]
datacube = datacube.drop_vars(var_to_drop)
for date in dates:
    for lat in range(0, lat_len, image_size):
        lat_sup = lat + image_size
        if lat == image_size * (lat_len // image_size):
            lat_sup = lat_len
            continue
        for lon in range(0, lon_len, image_size):
            lon_sup = lon + image_size
            if lon == image_size * (lon_len // image_size):
                lon_sup = lon_len
                continue
            subset = datacube.sel(time=date).isel(lat=slice(lat, lat_sup), lon=slice(lon, lon_sup))
            datacube_subset_list.append(subset)
            border_subset = loaded_all_boundaries[0, lat:lat_sup, lon:lon_sup]
            border_subset_list.append(border_subset)
```

Fig 32. Construcción de listas cubos y máscaras de entrenamiento

Así se obtuvo una lista de cubos de datos derivados y máscaras de bordes, todos los elementos del tamaño de la entrada del modelo a construir. Pero todavía hace falta convertir los cubos de datos a una estructura que el modelo pueda comprender, para ello, para cada trozo de cubo construido se obtuvieron y normalizaron los valores de cada banda e índice a utilizar, elaborando un *stack* de todos los valores juntos sobre una misma estructura de datos, la cual ya es capaz de ser comprendida por el modelo. La combinación de bandas se observa en la figura 33.

```
input_field_list = []

for datacube_piece in datacube_subset_list:
    datacube_piece["B02"] = datacube_piece["B02"].fillna(0)
    band_b02 = datacube_piece["B02"].values
    band_b02 = normalize_band(band_b02)

    datacube_piece["B03"] = datacube_piece["B03"].fillna(0)
    band_b03 = datacube_piece["B03"].values
    band_b03 = normalize_band(band_b03)

    datacube_piece["B04"] = datacube_piece["B04"].fillna(0)
    band_b04 = datacube_piece["B04"].values
    band_b04 = normalize_band(band_b04)

    datacube_piece["B08"] = datacube_piece["B08"].fillna(0)
    band_b08 = datacube_piece["B08"].values
    band_b08 = normalize_band(band_b08)

    datacube_piece["NDVI"] = datacube_piece["NDVI"].fillna(0)
    band_ndvi = datacube_piece["NDVI"].values
    band_ndvi = normalize_band(band_ndvi)

    datacube_piece["NDWI"] = datacube_piece["NDWI"].fillna(0)
    band_ndwi = datacube_piece["NDWI"].values
    band_ndwi = normalize_band(band_ndwi)

    datacube_piece["SAVI"] = datacube_piece["SAVI"].fillna(0)
    band_savi = datacube_piece["SAVI"].values
    band_savi = normalize_band(band_savi)

    input_field = np.dstack((band_b02, band_b03, band_b04, band_b08, band_ndvi, band_ndwi, band_savi))

    input_field_list.append(input_field)

input_field_list = np.array(input_field_list)
```

Fig 33. Combinación de bandas para construcción de imagen de entrenamiento

Así es como se definió la lista de imágenes y máscaras que servirán para entrenar el modelo. Es en este momento en que el segundo sector comienza: la construcción del modelo de red neuronal. Ya se ha hablado sobre la metodología y arquitectura utilizada para la construcción del modelo. Para comenzar con la implementación, primero se definieron métodos encargados de definir las diferentes capas presentadas en cada nivel de la red neuronal. Cada una de las partes de la red neuronal se encuentran en la figura 34, 35 y 36.

El método *double\_conv\_block* se trata de un bloque de convolución, el cual se encarga de definir dos capas convolucionales 2D, las cuales mediante la aplicación de un kernel, o filtro, el modelo puede aprender patrones locales presentados en la información de entrada.

```
def double_conv_block(x, n_filters):
    # Conv2D then ReLU activation
    x = layers.Conv2D(n_filters, 3, padding="same", activation="relu", kernel_initializer="he_normal")(x)
    # Conv2D then ReLU activation
    x = layers.Conv2D(n_filters, 3, padding="same", activation="relu", kernel_initializer="he_normal")(x)

    return x
```

Fig 34. Definición de bloque de convolución

El siguiente método, *downsample\_block*, se utiliza el anterior bloque de convolución doble previo a reducir las dimensiones de la imagen de entrada, reduciendo la cantidad de parámetros y protegiendo al modelo del sobreajuste, lo que conceptualmente se conoce como generalizarla entrada.

```
def downsample_block(x, n_filters):
    f = double_conv_block(x, n_filters)
    p = layers.MaxPool2D(2)(f)
    p = layers.Dropout(0.3)(p)

    return f, p
```

Fig 35. Definición de bloque de reducción

Por último, el bloque *upsample\_block* se encarga de realizar una convolución transpuesta, siendo la operación opuesta a la convolución, pudiendo aumentar también las dimensiones de las imágenes. Luego se encarga de concatenar los resultados obtenidos de las convoluciones pasadas con la convolución transpuesta, para terminar aplicando un Dropout y aplicando nuevamente una convolución doble.

```
def upsample_block(x, conv_features, n_filters):
    # upsample
    x = layers.Conv2DTranspose(n_filters, 3, 2, padding="same")(x)
    # concatenate
    x = layers.concatenate([x, conv_features])
    # dropout
    x = layers.Dropout(0.3)(x)
    # Conv2D twice with ReLU activation
    x = double_conv_block(x, n_filters)

    return x
```

Fig 36. Definición de bloque de muestra ascendente

De esta forma, es posible juntar cada uno de estos bloques para construir la arquitectura entera de la red neuronal, definiendo una imagen de entrada de 128x128 en base a 7 canales, que hacen referencia a la cantidad de bandas e índices utilizados para construir las imágenes de entrenamiento, como se aprecia en la figura 37.

```

def build_unet_model():
    # inputs
    inputs = layers.Input(shape=(128, 128, 7))

    # encoder: contracting path - downsample
    # 1 - downsample
    f1, p1 = downsample_block(inputs, 64)
    # 2 - downsample
    f2, p2 = downsample_block(p1, 128)

    # 3 - bottleneck
    bottleneck = double_conv_block(p2, 256)

    # decoder: expanding path - upsample

    u3 = upsample_block(bottleneck, f2, 128)
    u4 = upsample_block(u3, f1, 64)

    # outputs
    outputs = layers.Conv2D(1, 1, padding="same", activation="sigmoid")(u4)

    # unet model with Keras Functional API
    unet_model = tf.keras.Model(inputs, outputs, name="U-Net")

    return unet_model

```

Fig 37. Construcción de modelo de red neuronal

Y así queda finalizado el segundo sector destinado a la definición del modelo a utilizar. Por último, comienza el sector de la construcción y entrenamiento del modelo. Para ello, simplemente se hace la preparación de la información de entrenamiento y pruebas, se construye y compila el modelo para terminar con el entrenamiento, pudiendo seleccionar las épocas necesarias para un buen entrenamiento del modelo, tal y como se observa en la figura 38. La implementación completa se encuentra en el anexo F.

```

train_images, test_images, train_masks, test_masks = train_test_split(input_field_list, input_border_list, test_size=0.2, random_state=42)

model = build_unet_model()
model.compile(optimizer=tf.keras.optimizers.Adam(), loss="binary_crossentropy", metrics="accuracy")

history = model.fit(x=train_images, y=train_masks, validation_data=(test_images, test_masks), epochs=10, batch_size=32)

```

Fig 38. Creación de sets de entrenamiento y pruebas

Esta última fase del entrenamiento del modelo puede llegar a tomar más o menos tiempo, dependiendo de la cantidad de información de entrenamiento utilizada y la cantidad de épocas que se quieran analizar. Finalizado el entrenamiento, el modelo se encuentra listo para poder recibir predicciones y comenzar el análisis de los resultados obtenidos, lo cual será desarrollado en el siguiente capítulo.

## CAPÍTULO 5. PRUEBAS Y ANÁLISIS DE RESULTADOS

### 5.1. Algoritmos de Canny

Tal y como se definió en el desarrollo del algoritmo de Canny, para poder probar y obtener resultados en base a la información de las imágenes satelitales fue necesario seleccionar una banda de la cual poder obtener dicha información y llevarla por todo el proceso de transformación hasta el punto de que sirva como entrada para el algoritmo. Es por ello por lo que se definieron múltiples escenarios en donde se utilizaron diferentes bandas como principal punto de entrada para el procesamiento de información y la ejecución del algoritmo de Canny. De igual manera, el uso de diferentes umbrales de detección de bordes como parámetros del algoritmo para poder comparar los resultados de diferentes detecciones en base a los rangos de umbrales de detección.

Como primer segmento de pruebas, se utilizaron la banda B04 y la banda B08 como entrada de datos. El motivo de utilizar estas dos bandas como primeras pruebas fue debido a que ambas poseen características que hacen resaltar la vegetación y las características del suelo, logrando así una visión más clara del suelo de las parcelas agrícolas. La representación de estas bandas se encuentra en la figura 39.



Fig 39. Banda B04 (Izquierda) y Banda B08 (Derecha) utilizadas como entrada del algoritmo de Canny para Santa María, Usulután

De igual manera, tal y como se mencionó en la metodología, se calcularon índices de vegetación utilizando la información de las bandas respectivas, con el fin de poder comparar los resultados obtenidos tanto mediante los valores brutos de las bandas, como por la combinación de estas mediante operaciones que daban como resultado los índices de vegetación.

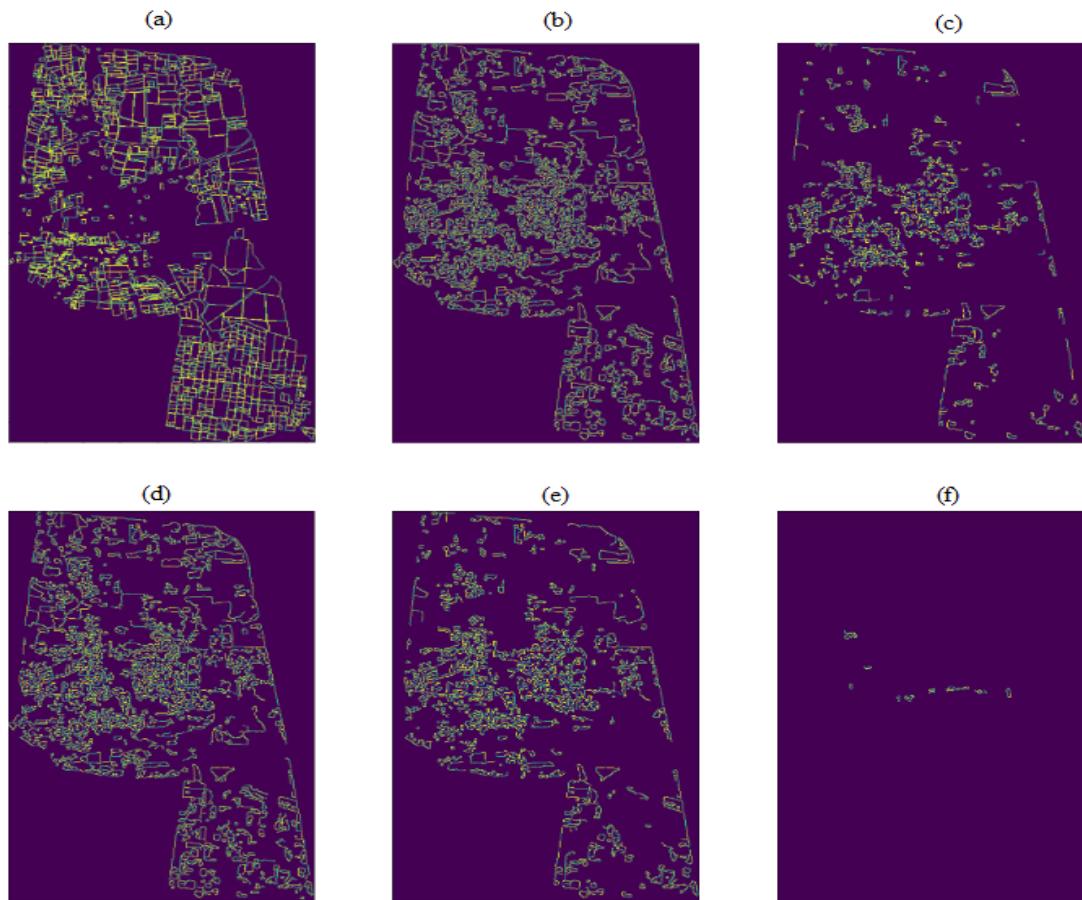


Fig 40. Resultados del Algoritmo de Canny bajo diferentes umbrales en la banda B04 en Santa María: a) Máscara Real, b) 30-90, c) 50-150, d) 90-30, e) 120-40, f) 100-300

En la figura 40 se pueden observar resultados obtenidos utilizando la banda B04 como entrada y aplicando diferentes rangos de umbrales para la detección de bordes. De esta forma, es posible apreciar que las máscaras generadas presentan varias zonas en donde no se detectan ningún borde, sin embargo al compararlo con la máscara del *Ground Truth*, se aprecia que debería de existir polígonos dentro de esas zonas vacías.

Otro detalle importante, es que al observar las zonas donde las parcelas son de tamaño más reducido, el algoritmo presenta cierta incompatibilidad con las parcelas detectadas con lo que indica la máscara verdadera. Es este detalle lo cual determina que muchos de los bordes detectados en estas zonas donde existen múltiples parcelas de menor tamaño acompañadas con una variedad de vegetación y suelos, pueden llegar a ser considerados como falsos positivos.

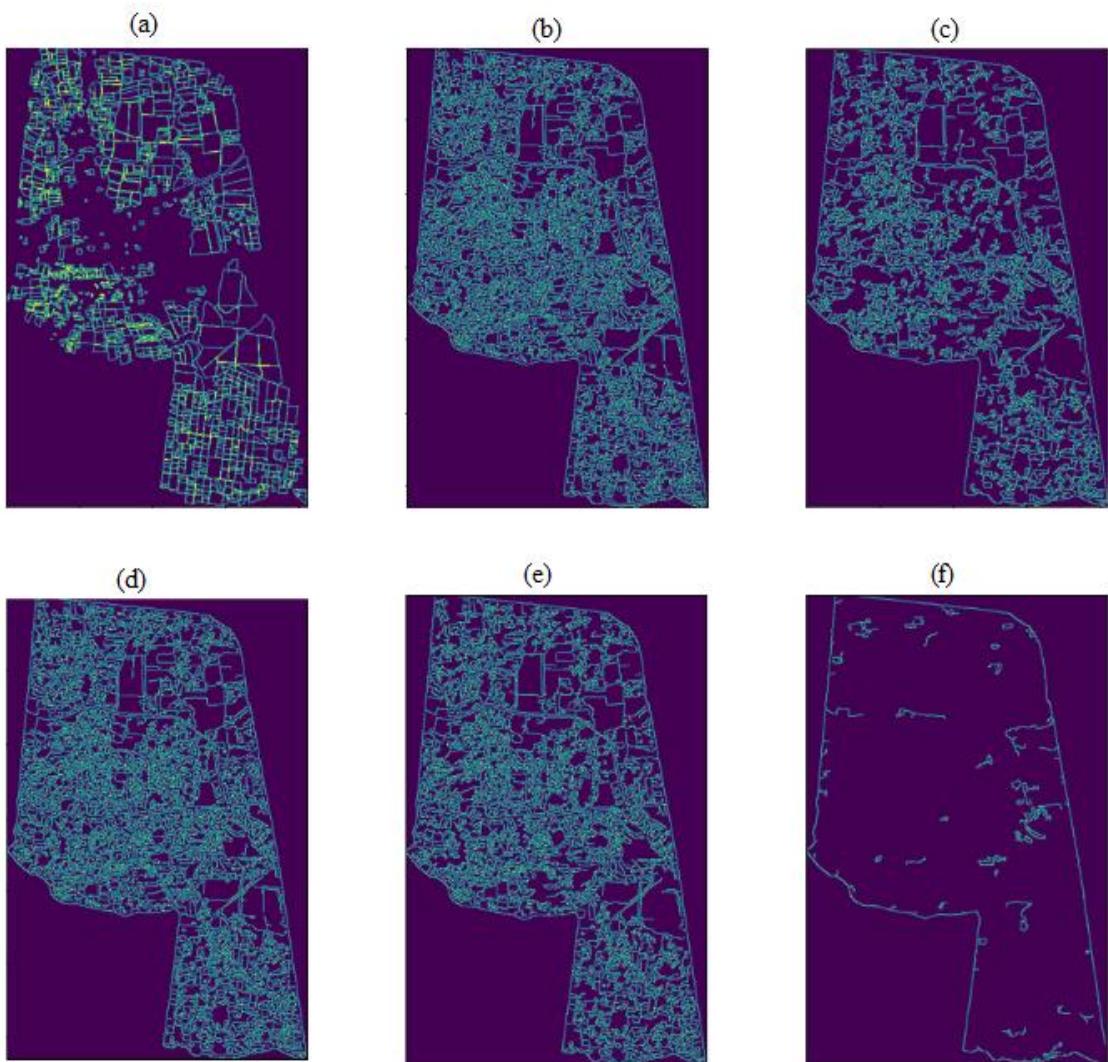


Fig 41. Resultados del Algoritmo de Canny bajo diferentes umbrales en la banda B08 en Santa María: a) Máscara Real, b) 30-90, c) 50-150, d) 90-30, e) 120-40, f) 100-300

Por otro lado, en la figura 41 se detallan los resultados obtenidos bajo los mismos escenarios definidos para la banda B04, pero esta vez utilizando la banda B08. Aquí a simple vista se puede apreciar que los espacios vacíos fueron reducidos considerablemente, y la cantidad de bordes “desconectados” unos de otros disminuyen en los diferentes rangos de los umbrales. Esta mejora en los resultados es muy importante debido a que ayuda a verificar que el uso de una banda exclusiva para el análisis de la vegetación y los suelos puede presentar mejoras considerables en los resultados de segmentación. Sin embargo, el problema de los falsos positivos y los bordes curvilíneos sigue presente, porque el cambio de banda no es suficiente para obtener mejores resultados.

Al observar la banda B04 y B08 en la figura 36, es posible apreciar que existen múltiples cambios de terreno en porciones muy pequeñas en múltiples lugares de la zona de Santa María, Usulután,

esto es importante debido a que el algoritmo de Canny al tratarse directamente de un algoritmo de detección de bordes, este busca los cambios más “bruscos” o notorios de una región a otra, por lo que al encontrarse con múltiples cambios, detecta múltiples bordes que en realidad no deberían de aparecer, dando como resultado los campos considerados como falsos positivos.

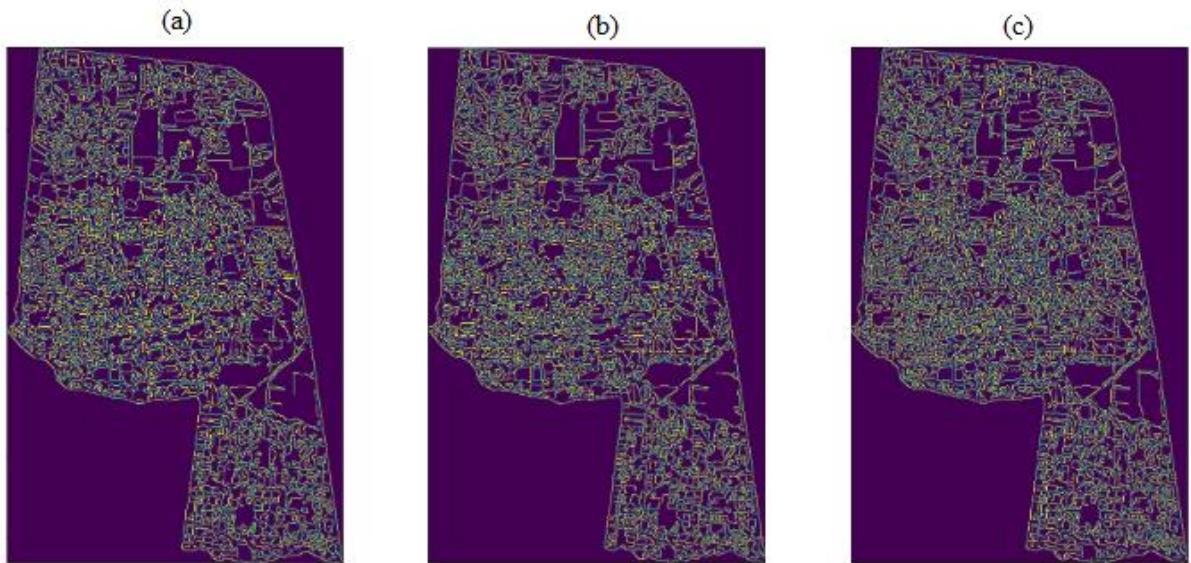


Fig 42. Resultados del Algoritmo de Canny sobre índices de vegetación de Santa María: a) NDVI, b) NDWI, c) SAVI

En la figura 42 se encuentran los resultados obtenidos mediante la aplicación del algoritmo de Canny a las bandas de los índices calculados, es decir, al índice NDVI, NDWI y SAVI. En estos resultados se pueden observar polígonos mejor definidos y menos espacios en blancos, esto es debido a que tal y como se explicó anteriormente los índices seleccionados se encargan de resaltar las características tanto de la vegetación como del suelo. Sin embargo, a pesar de que se detecten más bordes, todavía existe el problema de la segmentación indebida de algunas zonas, creando bordes en donde no debería.

Otro detalle observado sobre los resultados obtenidos en el área de Santa María es el hecho de que muchos de los bordes detectados no se llegan a cerrar en su totalidad, por lo tanto quedan polígonos incompletos, lo cual es un factor determinante para indicar si es una solución factible o no.

Es así cómo fue posible analizar de manera cualitativa los resultados obtenidos del algoritmo de Canny sobre la región de Santa María, Usulután, pero por otro lado, también se calcularon una serie de métricas presentes en la tabla 4 las cuales sirven para definir de una manera contable como fue el rendimiento y la precisión de la implementación desarrollada de algoritmo seleccionado.

Tabla 4. Métricas de Resultados de Algoritmo de Canny sobre Santa María, Usulután, El Salvador

Banda B04			
Umbral	Precisión	Recall	F1
30-90	0.17444737052110765	0.14440480908970801	0.15801077017600923
50-150	0.1357686184326906	0.05070022460034350 6	0.07383002260593526
90-30	0.17444737052110765	0.14440480908970801	0.15801077017600923
120-40	0.15012120853781116	0.083861804729819	0.10760982432346522
100-300	0.05604719764011799	0.00062755978332672 7	0.00124122162338722 8
Banda B08			
Umbral	Precisión	Recall	F1
30-90	0.17687853006363052	0.24514466904478796	0.20549026122347278
50-150	0.1701639238255359	0.1738340599815035	0.17197941344661385
90-30	0.17687853006363052	0.24514466904478796	0.20549026122347278
120-40	0.17512533848092443	0.21574844761527281	0.19332593414724378
100-300	0.15494274363168964	0.02189853349187475 4	0.03837360729272175
NDVI			
Umbral	Precisión	Recall	F1
30-90	0.17893982441321765	0.23090897080195535	0.20162953349196047
NDWI			

Umbral	Precisión	Recall	F1
30-90	0.17628156850931365	0.23193288413264632	0.2003137926116103
SAVI			
Umbral	Precisión	Recall	F1
30-90	0.17770051552571633	0.24478134495970405	0.20591544991039304

Analizando más a detalle las métricas obtenidas en la tabla 4, a primera vista se observa que las métricas no presentan valores muy altos, indicando que la implementación del algoritmo de Canny presenta un margen de mejora bastante amplio. Los valores de la precisión y la exhaustividad indican que el modelo a pesar de que si es capaz de detectar muchos bordes, la gran mayoría de ellos no están relacionados con los que se espera en base al *Ground Truth* utilizado para comparar, lo cual por un lado pueden indicar que en las porciones de tierra analizadas pueden existir más bordes de lo que en un principio se cree, o por otro lado es necesario hacer una preparación de la información más exhaustiva, teniendo en cuenta a su vez que la resolución de la imagen junto a la cantidad de vegetación que presenta contribuye al incremento de dificultad de la detección.

Por otro lado, analizando cada resultado sobre cada banda o índice utilizado, al igual que en el análisis cualitativo es posible apreciar que los valores de los índices tienden a ser más altos que las métricas obtenidas utilizando la información directamente de las bandas. Esta información es muy útil, debido a que indica que las características realizadas por medio de los índices tanto del suelo, como de la vegetación y los cuerpos de agua son de mucha utilidad para los modelos a la hora de estas trazando los bordes, por lo que su uso individual puede llevar a mejores resultados, o inclusive abre la posibilidad de que al combinarlos pueda mejorar aún más el rendimiento del algoritmo.

De igual forma, se construyeron una serie de gráficos para poder visualizar de una manera más amigable la información obtenida mediante el cálculo de las métricas para el análisis de resultados. Gracias a esto, es posible verificar la diferencia de los valores en relación con el uso de distintos tipos de bandas e índices, así como el uso de múltiples niveles de umbrales. Para observar las métricas de los resultados obtenidos sobre el área de Santa María, Usulután, se pueden consultar las figuras 43, 44 y 45.

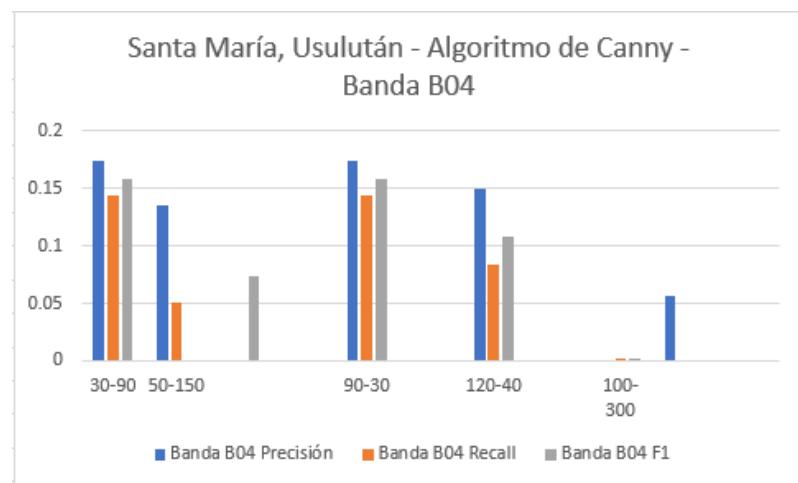


Fig 43. Gráfica comparativa de las métricas aplicadas en la banda B04 sobre el algoritmo de Canny en Santa María, Usulután

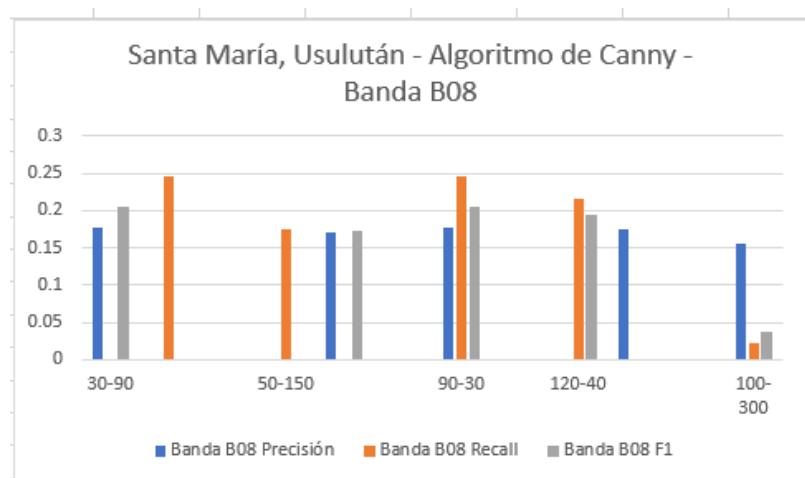


Fig 44. Grafica comparativa de las métricas aplicadas en la banda B08 sobre el algoritmo de Canny en Santa María, Usulután

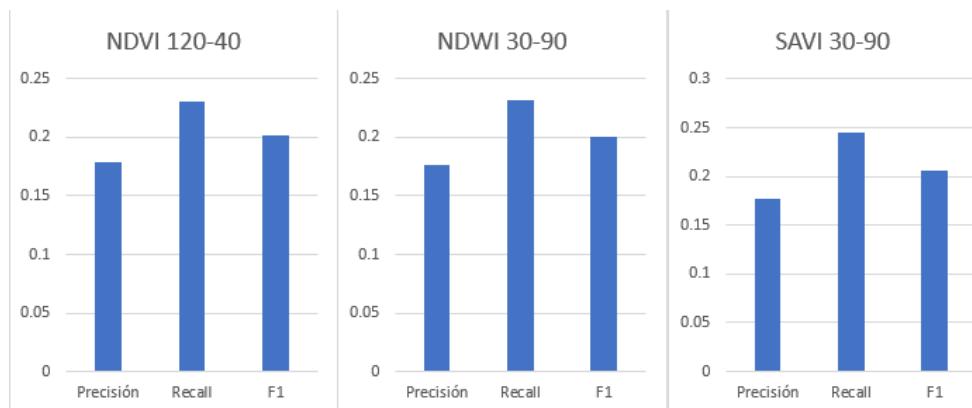


Fig 45. Gráfica comparativa de las métricas aplicadas sobre los índices de vegetación sobre el algoritmo de Canny en Santa María, Usulután

Es posible apreciar que la métrica de F1 no posee un comportamiento lineal o de algún tipo en específico, y no es de extrañar, debido a que como se ha mencionado, el valor del puntaje F1 es una combinación de los valores de precisión y exhaustividad, por lo que al probar con múltiples combinaciones de umbrales es posible encontrarse escenarios donde la precisión puede haber aumentado junto con la exhaustividad, o una hubiera aumentada y la otra no, que es precisamente lo que el puntaje F1 se encarga de verificar.

Como segundo segmento de pruebas para el algoritmo de Canny, se tomaron en cuenta los mismos escenarios para la aplicación del algoritmo sobre la región de Zelanda. Seleccionando las mismas bandas y calculando los mismos índices, se trabajó en recorrer todo el proceso necesario para llevar los datos de entrada de tal manera que puedan ser utilizados y trabajados por el algoritmo de identificación de bordes. En la figura 46 se observa la representación de estas bandas.

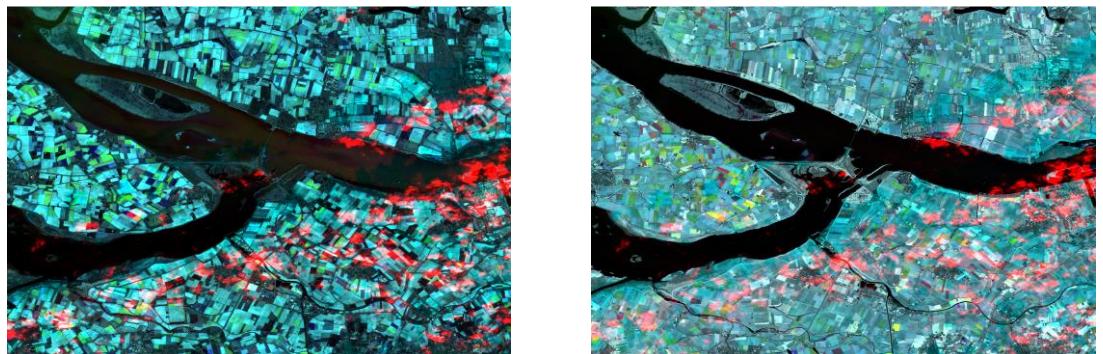


Fig 46. Banda B04 (Izquierda) y Banda B08 (Derecha) utilizadas como entrada del algoritmo de Canny para Zelanda, Países Bajos

Aplicando el algoritmo de Canny utilizando diferentes umbrales sobre la banda B04 del nuevo set de datos de la región europea, se consiguieron resultados que a simple vista difieren bastante de los conseguidos en la región de Santa María utilizando la misma banda. En la figura 47 se puede apreciar que en los diferentes umbrales no se alcanza a formar prácticamente ningún polígono completo sino que más bien el algoritmo arroja diferentes bordes en diferentes zonas del terreno, sin llegar a concretarse ningún polígono en específico. Esta información no es de mucha utilidad a la hora de realizar un análisis, sin embargo, si contribuye a poder identificar qué tipo de información de entrada puede o no ser efectiva para el problema de la detección de bordes

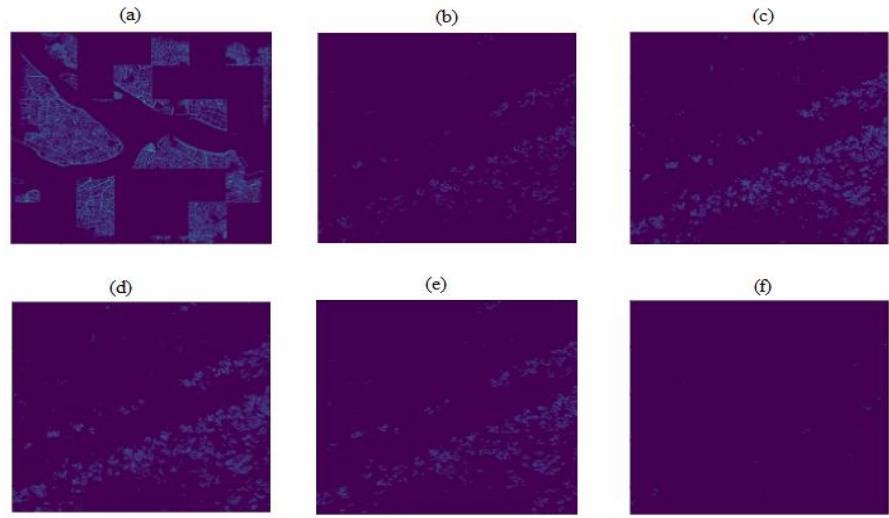


Fig 47. Resultados del Algoritmo de Canny bajo diferentes umbrales en la banda B04 en Zelanda: a) Máscara Real, b) 30-90, c) 50-150, d) 90-30, e) 120-40, f) 100-300

Debido a que los resultados obtenidos mediante la información de la banda B04 no proveyeron mucho valor, se prosiguió a utilizar el segundo conjunto de datos provenientes de la banda B08. En esta nueva prueba, los resultados mejoraron notablemente. En la figura 48 aparecen los resultados de la aplicación del algoritmo de Canny sobre la banda B08, en donde se puede apreciar ahora sí una mayor cantidad de bordes encontrados y de polígonos delimitados.

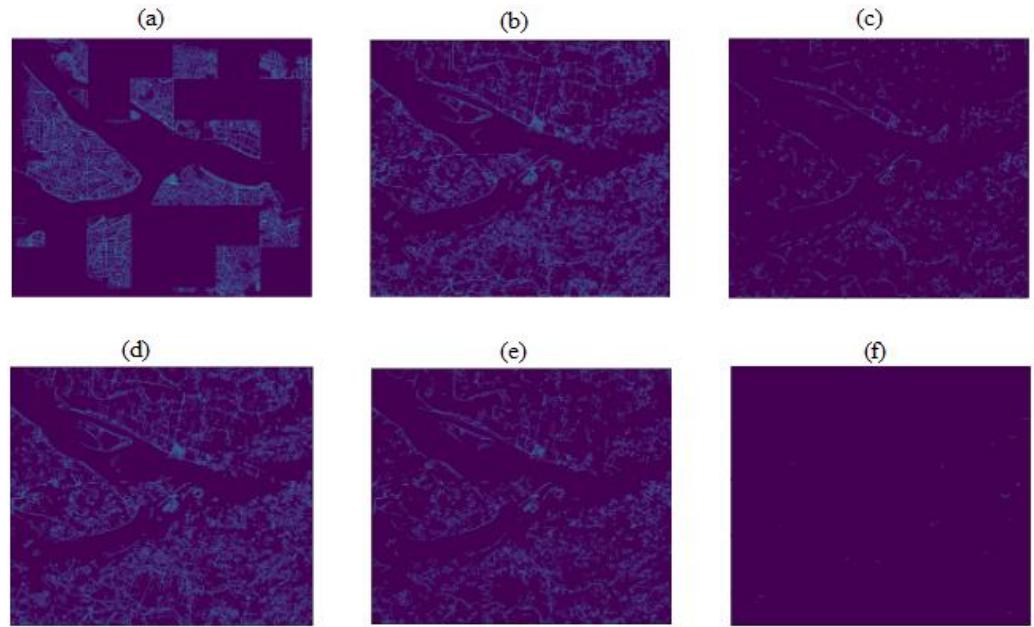


Fig 48. Resultados del Algoritmo de Canny bajo diferentes umbrales en la banda B08 en Zelanda: a) Máscara Real, b) 30-90, c) 50-150, d) 90-30, e) 120-40, f) 100-30

Siguiendo la misma metodología que para la región de Santa María, se continuó con el análisis de los resultados obtenidos al aplicar el algoritmo de Canny sobre los índices de vegetación

calculados. Observando la figura 49, al igual que para la región de Santa María, los resultados obtenidos mediante los índices de vegetación fueron bastante más prometedores que mediante el uso directo de las bandas. Se puede apreciar una gran cantidad de polígonos formados y bordes detectados, incluyendo la detección de los cuerpos de agua presentes en el territorio analizado. Nuevamente, es posible encontrar el problema recurrente de los falsos positivos, los cuales afectan de manera negativa en el rendimiento y la precisión de los bordes generados. Sin embargo, un factor que mejoró bastante con respecto a la zona de Santa María es la detección de múltiples polígonos de diferentes tamaños y orientaciones, dibujando bordes más rectos y polígonos más regulares.

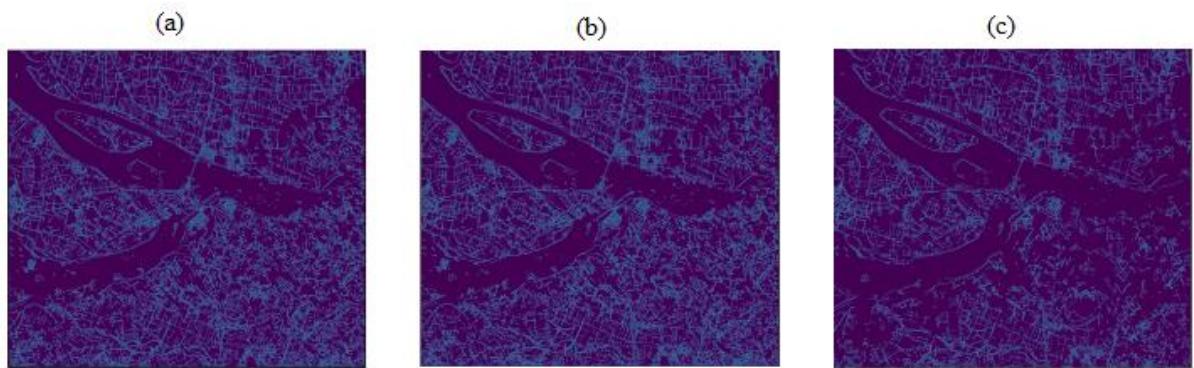


Fig 49. Resultados del Algoritmo de Canny sobre índices de vegetación en Zelanda: a) NDVI, b) NDWI, c) SAVI

Por último, tras haber realizado el análisis cualitativo de los resultados obtenidos, también es importante revisar los valores que arrojan las métricas en relación con los valores que se esperaban encontrar en la máscara de bordes real de la región. En la tabla 5, se pueden encontrar valores considerablemente menores a lo que se obtuvieron para la región de Santa María. Esto indica que a pesar de que se pudieron encontrar una mayor cantidad de polígonos, muchos de los bordes detectados no coinciden con los bordes del *Ground Truth*, sin embargo, también es un factor determinante el hecho de que la máscara de bordes reales no se encuentra completa, por lo que puede afectar en el rendimiento de las métricas calculadas.

Tabla 5. Métricas de Resultados de Algoritmo de Canny sobre Zelanda, Países Bajos

Banda B04			
Umbral	Precisión	Recall	F1
30-90	0.03711140679811298	0.01150770431051297	0.017567869351855562
50-150	0.04003839846406144	0.003754632338599571	0.006865451552613981
90-30	0.03711140679811298	0.01150770431051297	0.017567869351855562

120-40	0.03779651633235382	0.006657814586427811	0.011321380506815151
100-300	0.05990338164251208	0.0002325546503428305 6	0.0004633106535295678
Banda B08			
Umbral	Precisión	Recall	F1
30-90	0.047225762596652225	0.052118497846994044	0.04955164586710031
50-150	0.04881225959470586	0.014898501147769725	0.022829095026395418
90-30	0.047225762596652225	0.052118497846994044	0.04955164586710031
120-40	0.048192843061114765	0.030258360714767972	0.037175629156163445
100-300	0.05155482815057283	0.000236305531799973	0.0004704546981995774
NDVI			
Umbral	Precisión	Recall	F1
30-90	0.04630468541455991	0.08913969782898981	0.060948790449261686
NDWI			
Umbral	Precisión	Recall	F1
30-90	0.047120357756495715	0.06058798817722165	0.05301219053932758
SAVI			
Umbral	Precisión	Recall	F1
30-90	0.04631372228131904	0.0891659539991898	0.0609627560951216

Nuevamente, analizando más detalladamente los valores de las métricas obtenidos, se observa que aunque los valores esta vez sean muy bajos, quienes mantienen valores por encima del resto se trata de las métricas encontradas en base a los índices de vegetación, y no es de extrañar ya que en el análisis cualitativo también se apreciaba una mejor construcción de bordes que utilizando

directamente las bandas, con mayor cantidad de polígonos detectados y parcelas con formas más regulares y cerradas.

Al igual que para la región de Santa María, Usulután, se construyeron una serie de gráficos resumen para poder visualizar la información de las métricas obtenidas para el análisis de datos de una manera más sencilla y amigable. Estos gráficos pueden ser encontrados en las figuras 50, 51 y 52.

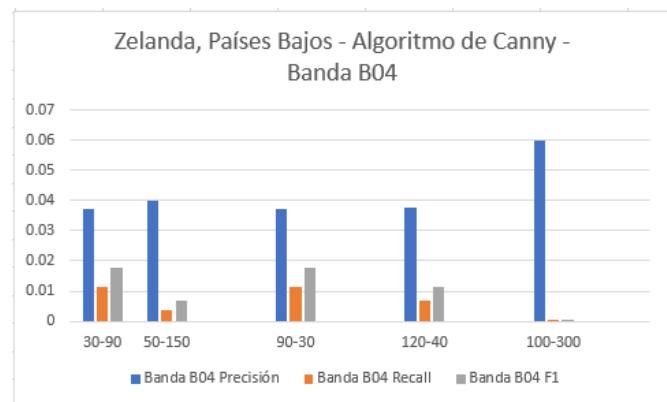


Fig 50. Gráfica comparativa de las métricas aplicadas en la banda B04 sobre el algoritmo de Canny en Zelanda, Países Bajos

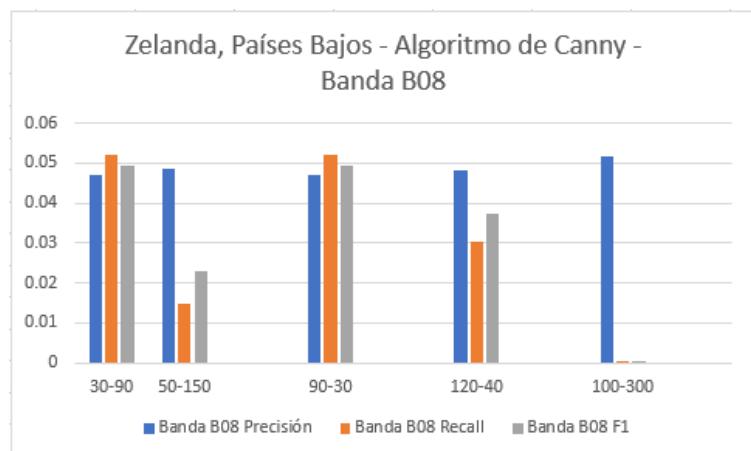


Fig 51. Gráfica comparativa de las métricas aplicadas en la banda B08 sobre el algoritmo de Canny en Zelanda, Países Bajos

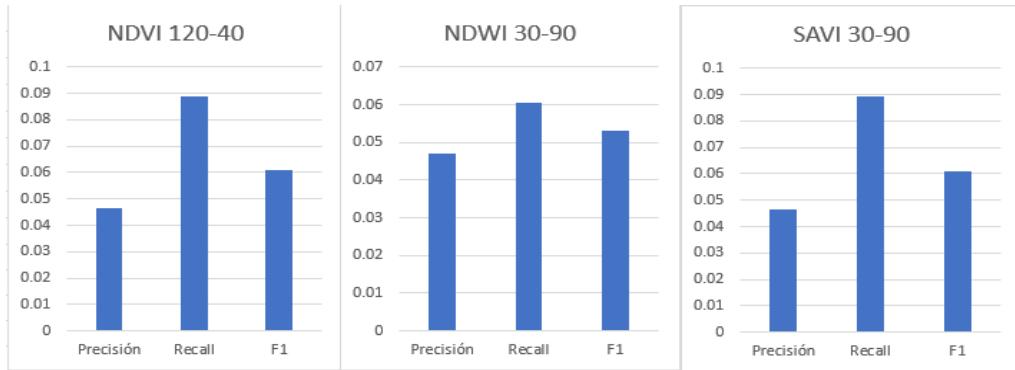


Fig 52. Gráfica comparativa de las métricas aplicadas en los índices de vegetación sobre el algoritmo de Canny en Zelanda, Países Bajos

Estos fueron los resultados obtenidos tras probar la implementación del algoritmo de Canny desarrollada para esta investigación. De manera general, es correcto decir que el algoritmo se encarga de generar bordes en base a las imágenes satelitales que se le colocan como entrada, construyendo los bordes respectivos en base a las características del suelo y la vegetación que se encuentren en las regiones analizadas. Dichos bordes se encargan de crear polígonos en su mayoría irregulares, pero que representan las parcelas que pueden ser detectadas en las regiones de Santa María y Zelanda. Sin embargo, aunque los bordes se hayan detectado, estos no cumplen con un buen nivel de precisión, por lo que la implementación tiene una gran gama de mejoras que puede aplicarse en busca de mejorar los resultados de las métricas de análisis de datos utilizadas.

## 5.2. Red Neuronal

Tras haber desarrollado tanto la estructura de la red neuronal utilizada, como obtenido y preparado la información que servirá para el entrenamiento la prueba de las misma, se realizaron múltiples escenarios para poder observar el modelo de aprendizaje que desarrolla la red neuronal, y así poder observar como ésta en base a la información presente en las múltiples bandas e índices de vegetación utilizados, pueda ser capaz de extraer los detalles más importantes y útiles para comprender la división de lo que se trate de un borde o no, es decir, que aprenda a distinguir los cambios de valor presentes de una porción de terreno a otro.

Para comenzar, se trabajó con la región de Santa María, Usulután, por lo que se utilizaron las imágenes satelitales provenientes del cubo de datos de la región respectiva. Tal y como se detalló en la sección del desarrollo del modelo, se construyó tanto un set de entrenamiento como un set de pruebas destinado para realizar predicciones una vez que el modelo quede listo para su uso. Debido a que la información de una sola fecha dentro del cubo de datos no sería suficiente para construir un set de entrenamiento lo suficientemente robusto, se utilizaron todas las fechas

presentes en el cubo para obtener más imágenes satelitales para entrenar. Así, el entrenamiento de la CNN se basó en una entrada de 672 imágenes de la región de Santa María, dejando 168 imágenes para realizar predicciones, en una separación 80-20. De igual manera, tanto para Santa María como para Zelanda, se utilizó únicamente un pequeño subconjunto de todas las bandas disponibles dentro de los cubos de datos, el cual estaría compuesto por las siguientes bandas: B02, B03, B04, B08 y las bandas con los índices calculados NDVI, NDWI y SAVI.

```

Epoch 1/10
21/21 [=====] - 187s 9s/step - loss: 0.4996 - accuracy: 0.8074 - val_loss: 0.4264 - val_accuracy: 0.8205
Epoch 2/10
21/21 [=====] - 178s 9s/step - loss: 0.4225 - accuracy: 0.8197 - val_loss: 0.4197 - val_accuracy: 0.8205
Epoch 3/10
21/21 [=====] - 179s 9s/step - loss: 0.4187 - accuracy: 0.8197 - val_loss: 0.4176 - val_accuracy: 0.8205
Epoch 4/10
21/21 [=====] - 182s 9s/step - loss: 0.4173 - accuracy: 0.8197 - val_loss: 0.4163 - val_accuracy: 0.8205
Epoch 5/10
21/21 [=====] - 178s 9s/step - loss: 0.4157 - accuracy: 0.8197 - val_loss: 0.4156 - val_accuracy: 0.8205
Epoch 6/10
21/21 [=====] - 179s 9s/step - loss: 0.4149 - accuracy: 0.8197 - val_loss: 0.4138 - val_accuracy: 0.8205
Epoch 7/10
21/21 [=====] - 182s 9s/step - loss: 0.4140 - accuracy: 0.8197 - val_loss: 0.4127 - val_accuracy: 0.8205
Epoch 8/10
21/21 [=====] - 187s 9s/step - loss: 0.4134 - accuracy: 0.8198 - val_loss: 0.4138 - val_accuracy: 0.8205
Epoch 9/10
21/21 [=====] - 184s 9s/step - loss: 0.4123 - accuracy: 0.8198 - val_loss: 0.4129 - val_accuracy: 0.8206
Epoch 10/10
21/21 [=====] - 178s 8s/step - loss: 0.4121 - accuracy: 0.8199 - val_loss: 0.4110 - val_accuracy: 0.8206

```

Fig 53. Resultados del entrenamiento de la CNN en base a las imágenes satelitales de la región de Santa María, Usulután

En la figura 53 se presentan los resultados del entrenamiento realizado en 10 épocas, donde se puede apreciar un valor de pérdida final del 0.4121, indicando que en promedio las predicciones de los bordes detectados por el modelo se desvían ese valor de la máscara real. Por otro lado, se observa un valor del 0.8199 de precisión, lo que indica que aproximadamente el 82% de los escenarios (imágenes) durante el entrenamiento las predicciones fueron correctas. Aunque el valor de la precisión indique resultados muy favorables para el modelo, no es una métrica determinante, es por eso por lo que la necesidad del uso de métodos de análisis de datos se hace presente como se verá más adelante.

Habiendo entrenado el modelo, ahora es posible realizar predicciones para analizar los resultados. Utilizando el set de entrenamiento definido anteriormente, el modelo ya entrenado se encarga de predecir cómo quedará construida la máscara de bordes detectados, sin embargo, al igual que en el algoritmo de Canny es necesario definir un umbral que determine a partir de qué cifra, los valores probabilísticos de la predicción serán considerados un borde. En la figura 54 es posible apreciar múltiples predicciones realizadas dentro de un rango de valores de umbral para observar cómo se comportan las predicciones y si efectivamente el modelo fue capaz de reproducir las máscaras de bordes con la cual se entrenó. En la figura 45 se encuentra un pequeño extracto de los resultados obtenidos mediante las predicciones del modelo entrenado con las imágenes de la región de Santa María, Usulután. Los resultados completos pueden apreciarse en el anexo C.

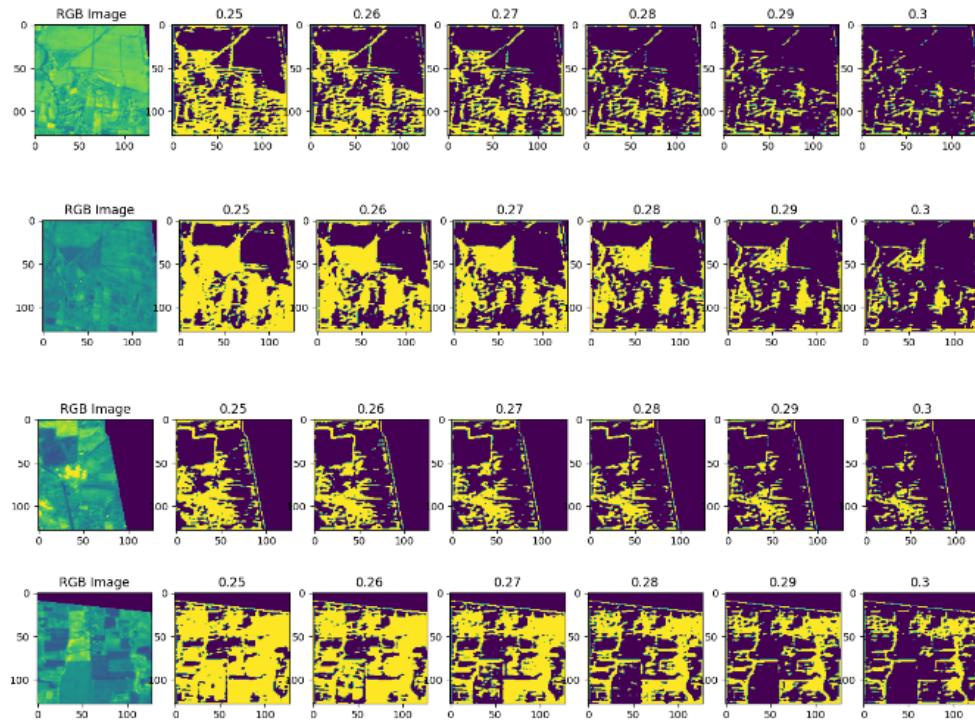


Fig 54. Resultados de predicciones al modelo de CNN para Santa María, Usulután

Tal y como se puede apreciar tanto en la figura 54 como en el anexo C, se utilizaron múltiples valores de umbrales para observar cómo se comportan las probabilidades obtenidas dentro de las predicciones. Las imágenes y/o máscaras resultantes son muy interesantes, puesto que en las parcelas de gran tamaño, el modelo pudo delimitar casi completamente todos los bordes de esta, aunque al igual que sucedía con el algoritmo de Canny, en muchos casos y escenarios estos bordes no llegan a cerrarse.

Por otro lado, los bordes de parcelas más pequeñas fueron un poco más difíciles de detectar utilizando los umbrales seleccionados, sin embargo, en muchos de los escenarios se logran apreciar la construcción y cierre de los mismos aunque parecen ser casos más aislados que una construcción general, por lo que aunque se trate de caso de éxitos, su número se ve muy reducido en comparación a los casos en donde no se presenta la generación de los bordes.

De manera general, el uso de umbrales dentro del rango de valores de 0.2-0.35 aproximadamente, a pesar de que presentan buenos resultados todavía no pueden ser considerados excelentes. Primeramente, debido a que se trata de la construcción de una máscara binaria en donde los valores de 1 hacen referencia a un borde y los valores de 0 al terreno, el umbral ideal para el análisis debería de ser un punto medio de valores alrededor de 0.5, puesto que es la mitad de la unidad que es el valor máximo que pueden tomar los valores de la matriz de probabilidad obtenidas mediante el modelo de la CNN. Utilizando el rango de umbrales mencionados en un inicio, se está muy cerca del valor intermedio, pero todavía hay mucho camino por recorrer. De igual manera, debido a que utilizando umbrales de valor bajo todavía no quedan definidos muchos de los bordes que deberían aparecer según las imágenes de entrenamiento y las máscaras de bordes reales, los resultados tienen un amplio nivel de mejora.

Estas son algunas de las observaciones que pueden ser descritas tras realizar un análisis cualitativo de los resultados obtenidos. Pero, al igual que para los resultados de las máscaras obtenidas mediante el algoritmo de Canny, se hizo uso de métricas destinadas al análisis de los resultados provenientes de modelos de aprendizaje automático, tal y como es el caso de la CNN. Tomando múltiples predicciones del set de pruebas, se hicieron los cálculos necesarios para obtener cada una de las métricas para los escenarios respectivos. De esta forma se construyó la tabla 6 en la cuales se pueden visualizar un conjunto de métricas obtenidas para su respectiva comparación y análisis, así como también en la figura 55 se puede encontrar una gráfica comparativa de los mismos valores.

Tabla 6. Métricas de análisis de resultados sobre predicciones del modelo de CNN aplicado a Santa María, Usulután.

Umbral	Precisión	Recall	F1 Score
0.25	0.19741935483870968	0.5668766209707299	0.2928509905254091 4
	0.2191039729501268	0.48017784364579474	0.3009055026700719 3
	0.1646742057081314	0.5665061133753242	0.255173564753004
0.26	0.2	0.4749907373101149	0.2814798550883741 4
	0.17606649139173947	0.7691737680622452	0.2865424430641822
	0.21931007901329735	0.4216376435716932	0.2885395537525355
0.27	0.1840016283329941	0.6698777324935161	0.2887025948103792 4
	0.21932754397684256	0.3649499814746202	0.2739916550764951 4
	0.15848032564450476	0.4327528714338644	0.2319992054821730 4
0.28	0.2078508895792149	0.5453871804371989	0.3009917186381760 3
	0.22088458518712378	0.3127084105224157	0.2588957055214723 4
	0.15379815941350802	0.36532048907002596	0.2164654226125137 2
0.29	0.20940868997059783	0.23749536865505744	0.2225694444444444
	0.20968992248062016	0.20044460911448686	0.2049630611858306 6
	0.21439700841383608	0.2549092256391256	0.2329045362220717 7

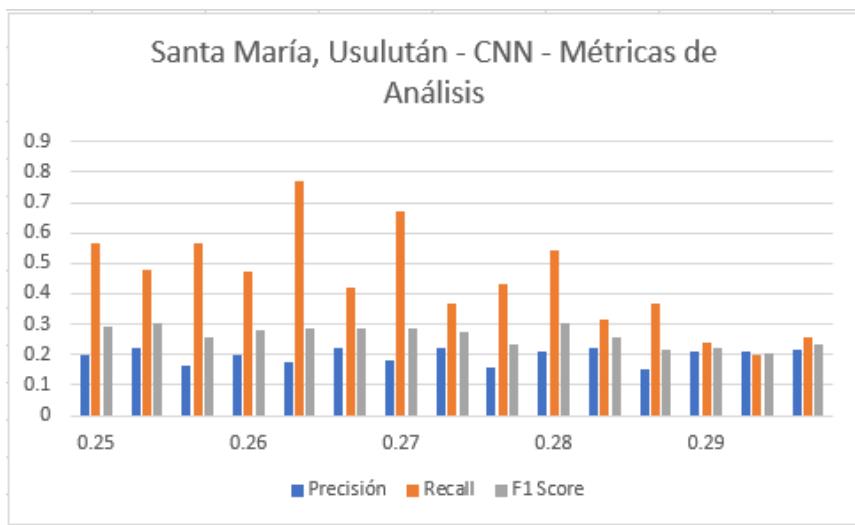


Fig 55. Gráfica comparativa de las métricas aplicadas sobre la CNN en Santa María, Usulután

Analizando la tabla 6, se observa que al igual que con el algoritmo de Canny, las métricas no arrojan valores muy altos. Por el lado de la precisión, ya que en muchos de los umbrales los valores oscilan entre 0.2, puede decirse que las predicciones del modelo son 20% precisas en comparación con lo el *Ground Truth*. Por otro lado, la métrica *recall*, indica que el modelo es muy bueno identificando casos positivos, sin embargo, esto no significa que esos positivos sean valores reales, y eso se puede ver en el valor de la precisión y el hecho de usar umbrales tan bajos, ya que el volumen de falsos positivos tiende a ser muy alto, lo cual afecta el rendimiento y eficacia del modelo.

Como segundo y último punto, se realizó el mismo proceso de construcción, entrenamiento y prueba sobre la región de estudio de Zelanda, Países Bajos. En cuestión de preparamiento de la información de entrada y construcción del modelo, no difiere mucho de el caso de Santa María, puesto que la principal diferencia es que para este escenario, se utilizó el cubo de datos que contenía las imágenes satelitales provenientes del Sentinel-2 para la región europea. Entrenando el modelo con un total de 1458 imágenes, de las cuales 1166 fueron destinadas al set de entrenamiento y 292 al set de pruebas, se consiguió el rendimiento de la red neuronal que aparece en la figura 56, obteniendo un valor de pérdida de 0.2032 y una precisión del 94.85%.

```

37/37 [=====] - 440s 12s/step - loss: 0.2359 - accuracy: 0.9398 - val_loss: 0.1932 - val_accuracy: 0.9520
Epoch 2/10
37/37 [=====] - 450s 12s/step - loss: 0.2052 - accuracy: 0.9485 - val_loss: 0.1925 - val_accuracy: 0.9520
Epoch 3/10
37/37 [=====] - 444s 12s/step - loss: 0.2042 - accuracy: 0.9485 - val_loss: 0.1926 - val_accuracy: 0.9520
Epoch 4/10
37/37 [=====] - 415s 11s/step - loss: 0.2042 - accuracy: 0.9485 - val_loss: 0.1938 - val_accuracy: 0.9520
Epoch 5/10
37/37 [=====] - 413s 11s/step - loss: 0.2036 - accuracy: 0.9485 - val_loss: 0.1921 - val_accuracy: 0.9520
Epoch 6/10
37/37 [=====] - 413s 11s/step - loss: 0.2034 - accuracy: 0.9485 - val_loss: 0.1920 - val_accuracy: 0.9520
Epoch 7/10
37/37 [=====] - 454s 12s/step - loss: 0.2036 - accuracy: 0.9485 - val_loss: 0.1943 - val_accuracy: 0.9520
Epoch 8/10
37/37 [=====] - 470s 13s/step - loss: 0.2040 - accuracy: 0.9485 - val_loss: 0.1942 - val_accuracy: 0.9520
Epoch 9/10
37/37 [=====] - 470s 13s/step - loss: 0.2026 - accuracy: 0.9485 - val_loss: 0.1919 - val_accuracy: 0.9520
Epoch 10/10
37/37 [=====] - 463s 12s/step - loss: 0.2032 - accuracy: 0.9485 - val_loss: 0.2039 - val_accuracy: 0.9520

```

Fig 56. Resultados del entrenamiento de la CNN en base a las imágenes satelitales de la región de Zelanda, Países Bajos

Los valores obtenidos del entrenamiento pueden parecer prometedores, pero nuevamente, no son valores definitivos para indicar que el modelo produce buenos resultados de predicciones, aunque puede ser considerado como un punto de partida. Al convertir las matrices de probabilidades en una máscara binaria para representar los bordes, es posible encontrarse con el primer factor relacionado al rendimiento del modelo, y es el hecho de que los valores de umbral utilizados son bajos. Tal y como se puede observar en la figura 57 y en el anexo D, el umbral utilizado para la construcción de las máscaras binarias gira en torno a los valores cerca de un umbral de 0.1, lo cual está mucho más alejado que los valores utilizados para las predicciones del modelo entrenado para Santa María, Usulután.

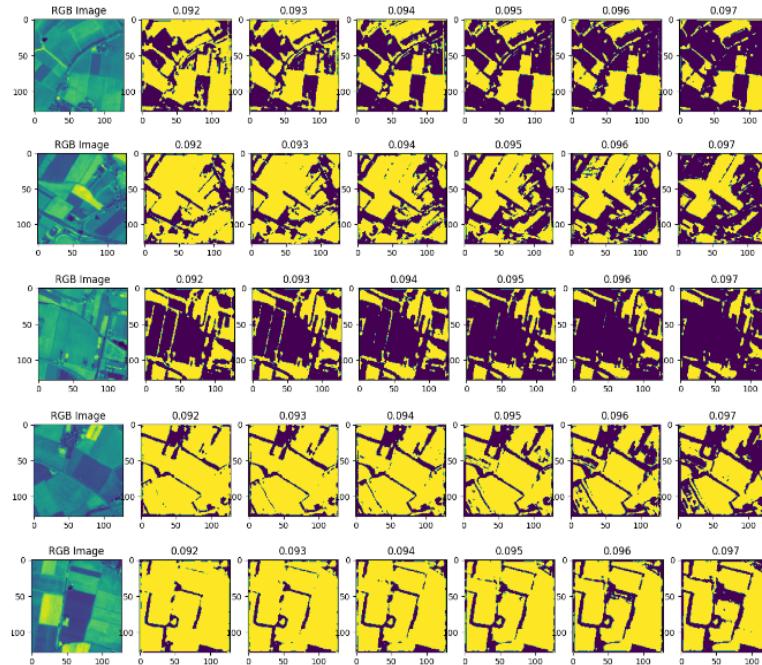


Fig 57. Resultados de predicciones al modelo de CNN para Santa María, Usulután

El hecho de tener la necesidad de utilizar un umbral tan bajo es considerado un punto de interés para indicar que el modelo no logró percibir una buena cantidad de información para poder desarrollar su aprendizaje de categorización de bordes. Sin embargo, ocurre un efecto interesante que fue posible apreciar también en el análisis de resultados del algoritmo de Canny y es que al observar los resultados de las predicciones, ya sea en la figura 57 o en el anexo D, las parcelas delimitadas por las predicciones del modelo para la región de Zelanda se encuentran mejor definidas que en los resultados de la región de Santa María.

Así como se ha venido observando desde el inicio del análisis de resultados, las imágenes satelitales de la región de Zelanda muestran terrenos mejor definidos en lo que respecta sus separaciones, conteniendo parcelas con figuras menos irregulares y con cortes más rectos. Esto pudo verse en las parcelas definidas por el algoritmo de Canny, y aquí puede verse nuevamente. A pesar de que los umbrales dejan un par de píxeles sin identificar, la construcción y el dibujo de los diferentes contornos de las parcelas se logra distinguir mucho mejor.

Por otro lado, el análisis cualitativo no es el único que importa, sino que al igual que para Santa María, los valores obtenidos mediante las métricas de análisis de resultados brindan más información sobre el rendimiento y la precisión obtenida con la arquitectura y entrenamiento de los modelos. Para el caso de Zelanda, estos valores pueden encontrarse en la tabla 7, así como en la figura 58.

Tabla 7. Métricas de análisis de resultados sobre predicciones del modelo de CNN aplicado a Zelanda.

Umbral	Precisión	Recall	F1 Score
0.92	0.08147905759162304	0.784251968503937	0.14762116496220543
	0.060533413245430025	0.47716535433070867	0.10743728392872973
	0.0765464450727536	0.8740157480314961	0.14076469469279057
0.93	0.08304468306200208	0.7551181102362204	0.14963332813231395
	0.05943857256506421	0.41181102362204725	0.10388320587943192
	0.07759477495385489	0.8606299212598425	0.14235477989059653
0.94	0.08493201713540696	0.7181102362204724	0.15189873417721517
	0.05581275720164609	0.3417322834645669	0.09595401282334734
	0.07824992596979567	0.8322834645669291	0.1430504804439031

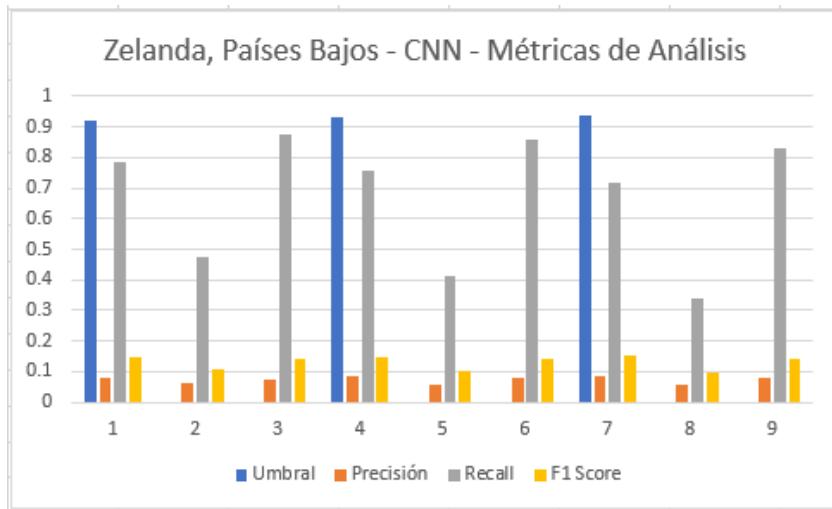


Fig 58. Gráfica comparativa de las métricas aplicadas sobre la CNN en Zelanda, Países Bajos

Al igual que con el algoritmo de Canny, a pesar de que de manera cualitativa las imágenes de bordes generados contenían mejor calidad de los polígonos encontrados, al analizar las métricas calculadas se puede observar que la precisión es muy baja en comparación a las métricas de análisis obtenidas en el modelo de Santa María, Usulután. De igual forma, con la métrica de *recall* se puede apreciar que al tener un valor muy alto, es muy probable que existan muchos valores que sean considerados falsos positivos, por lo que es uno de los principales problemas a la hora de medir la precisión del modelo.

De esta forma quedaron expresados los resultados de los modelos de CNN para ambas regiones, y se pudo observar que ambas implementaciones no alcanzaron valores de precisión muy altos, pero que si lograban cumplir con la tarea de detección de bordes. Para la región de Santa María, los bordes estaban menos definidos pero se acercaban más a lo que se esperaba según el *Ground Truth*. Por otro lado, para Zelanda se obtuvieron bordes mejor definidos pero con una mayor existencia de falsos positivos y valores clasificados como bordes de manera errónea.

Estos resultados dan lugar a múltiples observaciones y cambios que pueden llegar a hacerse para futuras implementaciones, pero son un punto de partida muy bueno para comenzar a definir mejores sets de entrenamiento, máscaras verdaderas con menos espacios nulos, y más procesos de pre y post procesamiento de la información tanto de entrada como de salida.

## CAPÍTULO 6. CONCLUSIONES Y RECOMENDACIONES

### 6.1. Conclusiones

- I. La segmentación de parcelas agrícolas es una ciencia del estudio del suelo que lleva practicando durante muchos años, evolucionando de su implementación manual hasta la creación de algoritmos y procesos automatizados dedicados a realizar todo el trabajo. El uso de algoritmos de segmentación y/o detección de bordes, como el algoritmo de Canny, así como el desarrollo de modelos de aprendizaje automático, como las redes neuronales convolucionales son dos soluciones las cuales presentan buenos resultados de segmentación sobre imágenes satelitales. Sin embargo, tras analizar los resultados obtenidos en la investigación, su implementación no debe quedarse en el desarrollo únicamente del algoritmo o el entrenamiento de la CNN, sino que deben combinarse tanto procesos de preprocesamiento como postprocesamiento de datos, de esta manera, la información de entrada para los modelos puede ser más detallada y llena de características a analizar, y los resultados pueden pasar por metodologías de suavizado y operaciones morfológicas para mejorar la precisión de los mismos.
- II. La detección de bordes mediante el algoritmo de Canny dio como resultado máscaras y/o imágenes en donde se podían apreciar múltiples polígonos formados y bordes dibujados a lo largo de toda la superficie de las zonas analizadas, Santa María en Usulután y Zelanda en los Países Bajos. Sin embargo, el uso de índices de vegetación como el NDVI o el SAVI, arrojaron resultados mucho más elaborados y bordes más logrados y terminados, lo contrario al uso de información directamente obtenida de las bandas satelitales. Esto se traduce a que las operaciones entre bandas presentan mejores resultados que su uso directo, debido a la combinación y exaltación de características de parámetros de análisis importantes como lo son el suelo, la vegetación y los cuerpos de agua presentes en las zonas de análisis.
- III. Según lo observado, el algoritmo de Canny creó el mapa de contornos en la zona de estudio con resultados satisfactorios de forma cualitativa, ya que se puede apreciar muchas coincidencias entre el mapa obtenido y el *Ground Truth* disponible. Sin embargo, con una inspección más detallada se observó la presencia de muchos bordes irregulares, imprecisos o incompletos con respecto a la referencia. Estas observaciones coincidieron con la lectura de los valores obtenidos de las métricas establecidas, las cuales no obtuvieron valores elevados. Por lo tanto, se concluye que existe un amplio margen de mejora para el refinamiento de los bordes detectados, y obtener mejores resultados cuantitativos, entre los procesos de refinamiento a tener en cuenta, se encuentra la utilización de filtros para eliminar los píxeles cubiertos con nubes que pueden llegar a confundir al algoritmo y estropear los resultados.
- IV. En términos cualitativos (vistosidad de los resultados, resolución y calidad de imagen de los resultados obtenidos), y habiendo analizado los resultados obtenidos luego de ejecutar-entrenar el algoritmo de Canny y la red neuronal con ambos conjuntos de datos (Santa María, Usulután y Zelanda, Países Bajos) es posible concluir que ambos métodos mostraron mejores resultados con el conjunto de datos de la región de Zelanda, Países Bajos. Esto es debido a factores puntuales como: una mayor área en kilómetros cuadrados de la región europea, por lo que los datos eran más consistentes y vistosos; y la forma de

los terrenos en la región europea (bordes más regulares). Esto permitió que el aprendizaje fuera más sencillo y las predicciones visualmente fueran más precisas para dicha región.

- V. En términos cuantitativos, la eficiencia de las predicciones expuestas tanto por el algoritmo de Canny como las expuestas por la red neuronal mostraron mayor porcentaje de eficiencia con el conjunto de datos del municipio de Santa María, Usulután. Esto a pesar de que la cantidad de información en el conjunto de datos de Santa María fue menor en comparación al de Zelanda en Países Bajos. Tal y como se explicó a lo largo de la investigación, estos algoritmos computacionales prometen mejores resultados si se entrena con grandes cantidades de información. Sin embargo, la calidad de la información es muy importante al momento de obtener las predicciones, y es por esto por lo que la región de Santa María mostró mejores resultados; la explicación parte del hecho de que dicha región es más pequeña en territorio en comparación a la región europea, lo que reduce el margen de error al momento del entrenamiento y posteriores predicciones. También, sus imágenes contaban con menores índices de nubosidad y las máscaras que se les aplicaron presentaban menos espacios vacíos en comparación a la región de Zelanda; lo que permitió a la región de Santa María, a pesar de contar con menos información pero con más calidad, mostrar mejores porcentajes de eficiencia en sus predicciones finales.
- VI. La red neuronal convolucional resultó ser una herramienta muy poderosa y acertada para la detección de bordes, sin embargo, necesita de cantidad de información considerable para poder exprimir su máximo potencial, a su vez de que el preprocesamiento de la información juega un papel muy importante para poder construir la entrada del modelo generado por la red neuronal. Tanto para la región de Santa María, Usulután, como para Zelanda, Países Bajos, se obtuvieron múltiples resultados en donde los bordes definidos por la red neuronal eran de buena calidad, sin embargo, en la mayor parte de los escenarios existían valores que ensuciaba la máscara generada por las predicciones.
- VII. En general, ambas metodologías presentan resultados muy similares para cada una de las respectivas áreas de estudio, teniendo puntos fuertes y débiles al compararlos entre sí. Por un lado, mediante el algoritmo de Canny se pudo obtener imágenes con mejor definición de todos los bordes encontrados, haciendo más sencillo el trabajo la observación de la separación de las parcelas y el análisis cualitativo. Por otro lado, la red neuronal presenta un modelo que puede ser mejorado constantemente, el cual irá aprendiendo a medida que la información de entrada aumenta o simplemente cada vez se hagan más predicciones, lo que a largo plazo permitirá tener una herramienta muy útil para el análisis de terrenos y la segmentación de parcelas. Ambas soluciones tienen mucho potencial para crecer y mejorar sus resultados, ya sea utilizando preprocesamiento de su información de entrada, o utilizando sus resultados como la entrada de datos a otros procesos que se encarguen de mejorar la calidad y definición de los bordes

## 6.2. Recomendaciones

- I. Para lograr un buen entrenamiento de las Redes Neuronales es importante alimentar el algoritmo con grandes volúmenes de datos. Con el fin de mejorar la precisión del

resultado del algoritmo y reducir el ruido generado para el área de Santa María en Usulután, considerando que no se contó con muchas imágenes del área , siendo este un factor que influye mucho en los resultados del algoritmo, se recomienda obtener más imágenes satelitales de la región o de lugares similares a la zona a segmentar, para poder así obtener más información. De esta forma sería más accesible entrenar el algoritmo y por consecuencia, se obtendría un mejor resultado en la identificación y segmentación de las parcelas.

- II. Para mejorar la implementación del algoritmo de Canny en el área de Santa María en Usulután, se sugiere explorar y probar otros tipos de índices, como índices relacionados a la textura del terreno, además de los utilizados, estos índices adicionales podrían proporcionar información complementaria sobre diferentes aspectos del área. Además, se recomienda considerar la utilización de diferentes bandas espectrales e incluso combinaciones con las mismas y la realización de operaciones entre ellas. Al combinar bandas o realizar operaciones, es posible obtener resultados más precisos.
- III. Se recomienda el uso de un algoritmo o post procesado al mapa de contornos resultante obtenido del algoritmo de Canny, esto con el objetivo de poder definir los detalles, y corregir errores como conexiones incompletas entre bordes, imprecisiones, irregularidades, o bordes extras que se hayan detectado, pero que no estén reflejados en el *Ground Truth* (falsos positivos). A su vez, filtrar los píxeles de nubes ya sea por sustitución de valores o eliminación de estos, contribuiría a mejorar la tierra y/o imágenes analizadas. Con esto podría mejorarse bastante los valores de las métricas para evaluar la calidad de la detección de bordes.
- IV. Los resultados de una red neuronal convolucional como la desarrollada en la investigación, pueden llegar a ser muy buenos y precisos dependiendo del tiempo de entrenamiento y la cantidad de información que se reciba. Sin embargo, el uso de procesamiento que se encarguen de mejorar la calidad de las predicciones realizadas luego de ser devueltas por el modelo puede contribuir en una mejora considerable de la precisión de este. Por lo que se recomienda el uso de métodos de post procesamiento de las predicciones, ya sea de suavizado de bordes u operaciones morfológicas, que pueden llegar a influir positivamente y reducir la existencia de errores o falsos positivos dentro de los resultados.
- V. La arquitectura U-Net utilizada para el desarrollo de la red neuronal convolucional resulta ser muy útil para la segmentación de imágenes. Al no existir un límite para las capas o

filtros que puedan agregarse a la red neuronal, se invita a poder construir más filtros dentro de la arquitectura base para poder extraer la máxima cantidad de detalles posibles de las imágenes satelitales y así poder darle más información para el entrenamiento del modelo. De igual forma, las funciones de activación y de pérdida juegan un papel muy importante en el desarrollo de los modelos de CNN, por lo que el uso y comparación de otros diferentes al utilizado en la investigación, puede llegar a ser favorable en relación con los resultados que se pueden llegar a obtener.

- VI. El trabajar con implementaciones en base a inteligencia artificial, trata de muchos procesos los cuales piden muchos recursos y tiempo, es por ello que la capacidad de tener un equipo capaz de soportar grandes volúmenes de información y procesos complejos y pesados forma un papel muy importante en el desarrollo de investigaciones de esta índole. Por lo tanto, se recomienda el uso de un equipo robusto y moderno capaz de soportar procesos complejos durante largas jornadas, así como tener la capacidad de memoria y procesamiento de datos en grandes cantidades, llámese grandes cantidades, a la capacidad de trabajar con archivos mayores a los 50 GB de información, durante procesos que pueden llegar a tardar horas en finalizar.

## GLOSARIO

**Algoritmo:** Conjunto ordenado de operaciones sistemáticas que permite hacer un cálculo y hallar la solución de un tipo de problemas.

**Autoaprendizaje de máquinas:** conocido como aprendizaje automático o aprendizaje automatizado o aprendizaje de máquinas o aprendizaje computacional, es el subcampo de las ciencias de la computación y una rama de la inteligencia artificial, cuyo objetivo es desarrollar técnicas que permitan que las computadoras aprendan.

**Biomasa:** Se refiere a la estimación o medición de la cantidad de masa biológica presente en una región geográfica específica.

**Cartografía:** proceso de crear, almacenar, administrar, analizar y visualizar datos geográficos en forma de mapas digitales.

**Catastro:** Registro público que contiene información detallada sobre los bienes inmuebles de un determinado territorio, como parcelas de tierra, edificios, viviendas, etc. El catastro proporciona datos sobre la ubicación, dimensiones, características físicas y valoración económica de los bienes inmuebles.

**Centroide:** se refiere al punto central o al centro geométrico de una entidad espacial, como un polígono o una región. Es el punto equidista de todos los puntos en el contorno o límites de la entidad.

**Ciencia de datos:** es el estudio de datos con el fin de extraer información significativa para empresas. Es un enfoque multidisciplinario que combina principios y prácticas del campo de las matemáticas, la estadística, la inteligencia artificial y la ingeniería de computación para analizar grandes cantidades de datos. Este análisis permite que los científicos de datos planteen y respondan a preguntas como “qué pasó”, “porqué pasó”, “qué pasará” y “qué se puede hacer con los resultados”.

**Clustering:** es un conjunto de técnicas utilizadas para particionar datos en grupos o clústers.

**Convolución:** es una operación matemática que combina dos señales y genera una tercera señal.

**Deep Learning:** Es una técnica dentro del campo del Machine Learning que se basa en el uso de redes neuronales artificiales con múltiples capas ocultas.

**Delta fluvial:** es una estructura geológica formada por la acumulación de sedimentos transportados por el río y depositados en el punto donde el río se encuentra con el cuerpo de agua.

**Espectro:** un espectro se refiere a la distribución de energía radiante en diferentes longitudes de onda que es emitida, reflejada o absorbida por un objeto o una superficie. El espectro electromagnético abarca todo el rango de longitudes de onda posibles, desde ondas de radio hasta rayos gamma.

**Filtro Gaussiano:** Es un tipo de filtro utilizado en procesamiento de imágenes y procesamiento de señales para suavizar una imagen o reducir el ruido. Se basa en la función matemática conocida como la distribución normal o gaussiana.

**Follaje:** Se refiere a la vegetación, específicamente a las hojas y ramas de los árboles y otras plantas. Esta información se utiliza para representar y analizar la distribución y densidad de la vegetación en un área determinada.

**Fotorreceptor:** Es un componente esencial de los sensores remotos utilizados para capturar la radiación electromagnética proveniente de la Tierra. También se conoce como detector o receptor de luz.

**Geociencia:** Es un conjunto de disciplinas que estudian la estructura interna, la morfología, la dinámica superficial y la evolución del Planeta Tierra.

**Geoespacial:** Perteneciente o relativo a información geográfica obtenida mediante la utilización de Tecnologías de la Información y la Comunicación.

**Geopackage:** Es un formato de archivos universal para compartir y transferir datos espaciales vectoriales y raster.

**Grafo:** es un tipo abstracto de datos, que consiste en un conjunto de nodos (también llamados vértices) y un conjunto de arcos (aristas) que establecen relaciones entre los nodos.

**Inteligencia Artificial:** También conocida solo como IA es una disciplina de la informática que busca desarrollar sistemas o máquinas capaces de realizar tareas que requieren de inteligencia humana.

**Lenguaje de programación:** es el conjunto de instrucciones a través del cual los humanos interactúan con las computadoras. Un lenguaje de programación nos permite comunicarnos con las computadoras a través de algoritmos e instrucciones escritas en una sintaxis que la computadora entiende e interpreta en lenguaje de máquina.

**Machine Learning:** Es una rama de la Inteligencia Artificial que se centra en el desarrollo de algoritmos y modelos que permiten a las máquinas aprender automáticamente a partir de los datos, sin ser programadas explícitamente.

**Multiplataforma:** Aplicación, programa o sistema operativo que puede utilizarse en diversos entornos o sistemas operativos.

**Raster:** Es una representación de datos espaciales que se organiza en una estructura de cuadrícula regular o malla bidimensional de celdas o píxeles. Cada celda o píxel contiene un valor que representa una característica o atributo específico en un área geográfica determinada.

**Red Neuronal:** Modelo computacional inspirado en la estructura y funcionamiento del cerebro humano. Está compuesta por un conjunto interconectado de nodos llamados neuronas artificiales, que procesan y transmiten información a través de conexiones ponderadas.

**Regresión:** El término regresión se usa cuando intenta encontrar la relación entre variables. En Machine Learning y en el modelado estadístico, esa relación se usa para predecir el resultado de eventos futuros.

**Segmentación:** Se conoce como segmentación al acto y consecuencia de segmentar (es decir, de dividir o formar segmentos o porciones).

**Shapefile:** es un formato de archivo geoespacial ampliamente utilizado para almacenar datos de mapas vectoriales. Fue desarrollado por la empresa ESRI como un formato propietario aunque se le considera un “estándar” para su uso en el software de sistemas de información geográfica (SIG).

**SSH:** SSH son las siglas de Secure SHell y es un protocolo de red destinado principalmente a la conexión con máquinas a las que accedemos por línea de comandos. Con SSH es posible conectarse con servidores, usando la red Internet como vía para las comunicaciones.

**Teledetección:** Detección a distancia de informaciones que se producen en la superficie de la Tierra y sobre otros astros solares y que se realiza mediante satélites y sondas artificiales.

**Tonalidad:** La tonalidad se refiere a la apariencia visual o el matiz de un objeto o superficie en una imagen o en un conjunto de datos espectrales. También se conoce como color o tono en el contexto de la representación de imágenes.

## REFERENCIAS

Aguilar, C. G. (2014). Aplicación de índices de vegetación derivados de imágenes satelitales Landsat 7 ETM+ y ASTER para la caracterización de la cobertura vegetal en la zona centro de la provincia de Loja, Ecuador. Retrieved from

[http://sedici.unlp.edu.ar/bitstream/handle/10915/34487/Documento\\_completo.pdf?sequence=1&isAllowed=y](http://sedici.unlp.edu.ar/bitstream/handle/10915/34487/Documento_completo.pdf?sequence=1&isAllowed=y)

Ayodele, T. O. (2010). Types of machine learning algorithms. En New advances in machine learning (Vol. 3, pp. 19-48). <https://books.google.es/books?id=XAqhDwAAQBAJ>

Koehrsen W. (2023, February 10). Precision and Recall in Classification Models. Retrieved from <https://builtin.com/data-science/precision-and-recall>

Centeno Franco, A. (2022). Estudio de la influencia de la música en el rendimiento académico de los estudiantes universitarios. Undergraduate thesis, Universidad de Sevilla, Sevilla, Spain. Retrieved from <https://idus.us.es/bitstream/handle/11441/90004/Centeno%20Franco%20Alba%20TFG.pdf?sequence=1&isAllowed=y>

Chauhan, S. (2022, February 18). Precision and Recall as Simple as Possible. Medium. Retrieved from <https://shubhamchauhan125.medium.com/precision-and-recall-as-simple-as-possible-516f0223d7bd>

Campbell, J. B. (2002). Introduction to remote sensing (3rd ed.). The Guilford Press.

European Space Agency. (s.f.). The Sentinel missions. Recuperado de [https://www.esa.int/Applications/Observing\\_the\\_Earth/Copernicus/The\\_Sentinel\\_missions](https://www.esa.int/Applications/Observing_the_Earth/Copernicus/The_Sentinel_missions)

Datta, S., & Datta, S. (2023). Minimum Cut on a Graph Using a Maximum Flow Algorithm | Baeldung on Computer Science. Baeldung on Computer Science. <https://www.baeldung.com/cs/minimum-cut-graphs>

Dayan, P., Sahani, M., & Deback, G. (1999). Unsupervised learning. The MIT encyclopedia of the cognitive sciences, 857-859. <https://web.math.princeton.edu/~sswang/developmental-diaschisis-references/dun99b.pdf>

*El algoritmo k-means aplicado a clasificación y procesamiento de imágenes.* (2019, marzo). Recuperado 26 de marzo de 2023, de [https://www.unioviedo.es/compnum/laboratorios\\_py/kmeans/kmeans.html](https://www.unioviedo.es/compnum/laboratorios_py/kmeans/kmeans.html)

EOS. (2022, June 24). Cómo hacer un análisis espectral. EOS. Retrieved from <https://eos.com/es/make-an-analysis/>

Galipienso, A., Isabel, M., Cazorla Quevedo, M. A., Colomina Pardo, O., Escolano Ruiz, F., & Lozano Ortega, M. A. (2003). Inteligencia artificial: modelos, técnicas y áreas de aplicación. Ediciones Paraninfo, SA. <https://books.google.com.sv/books?id=spC6S7UfZgC>

García Santillán, I. D. (2017). Métodos de visión por computador para detección automática de líneas de cultivo curvas/rectas y malas hierbas en campos de maíz. Recuperado de <https://eprints.ucm.es/id/eprint/49697/1/T40479.pdf>

Hinton, G. E., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7), 1527-1554.

I. Papadakis, I. Papatheodorou, H. Faria Viegas, N. Milionis, M. Szymura, J. Huth, K. Sniter, R. Bortnowski, R. Resegotti. (2016). Sistema de identificación de parcelas agrícolas. En ECA Europa. Recuperado 4 de junio de 2023, de [https://www.eca.europa.eu/Lists/ECADocuments/SR16\\_25/SR\\_LPIS\\_ES.pdf](https://www.eca.europa.eu/Lists/ECADocuments/SR16_25/SR_LPIS_ES.pdf)

La Serna Palomino, N., & Román Concha, U. (2007). Técnicas de segmentación en procesamiento digital de imágenes [Proyecto de Investigación]. Universidad Nacional Mayor de San Marcos.

Leiva, M. (2017). Informe de la simulación de un sistema de control de inventarios. Retrieved from [https://www-2.dc.uba.ar/trabajosFinalesOrga2/2017\\_LEIVA/informe.pdf](https://www-2.dc.uba.ar/trabajosFinalesOrga2/2017_LEIVA/informe.pdf)

Liakos, K., Busato, P., Moshou, D., Pearson, S., & Bochtis, D. (2018). Machine Learning in Agriculture: A Review. *Sensors*, 18(8), 2674. <http://dx.doi.org/10.3390/s18082674>

Manrique, E. (2020). Machine Learning: análisis de lenguajes de programación y herramientas para desarrollo.

Oscar. (2017). *Implementación de Algoritmo de Superpixeles para la Segmentación de Imágenes a Color*. Tomado de:

<https://www.boletin.upiita.ipn.mx/index.php/ciencia/717-cyt-numero-61/1397-implementacion-de-algoritmo-de-superpixeles-para-la-segmentacion-de-imagenes-a-color>

Pickles, J. (1995). Ground Truth: The Social Implications of Geographical Information Systems (p. 179). APA Publishing.

Redes neuronales convolucionales (CNN) y aprendizaje profundo. (s. f.-a). Intel.  
<https://www.intel.es/content/www/es/es/internet-of-things/computer-vision/convolutional-neural-networks.html#:~:text=Una%20red%20neuronal%20convolucional%20es,de%20grandes%20cantidades%20de%20datos>

Rojas, E. M. (2020). Machine Learning: análisis de lenguajes de programación y herramientas para desarrollo. *Revista Ibérica de Sistemas e Tecnologias de Informação*, (E28), 586-599.

RPubs (2018). Clasificación supervisada.  
[https://rpubs.com/marialorena/clasificacion\\_supervisada](https://rpubs.com/marialorena/clasificacion_supervisada)

K. Sergieieva. (2023, March 8). Índices de Vegetación. EOS Data Analytics. Retrieved from <https://eos.com/es/blog/indices-de-vegetacion/>

Serna, A., Acevedo, E., & Serna, E. (2017). Principios de la inteligencia artificial en las ciencias computacionales. Desarrollo e Innovación en Ingeniería, 161.  
[https://www.researchgate.net/profile/Edgar-Serna-M/publication/331385353\\_Desarrollo\\_e\\_innovacion\\_en\\_ingenieria\\_ed\\_2/links/5c76e4ce92851c69504663b5/Desarrollo-e-innovacion-en-ingenieria-ed-2.pdf#page=162](https://www.researchgate.net/profile/Edgar-Serna-M/publication/331385353_Desarrollo_e_innovacion_en_ingenieria_ed_2/links/5c76e4ce92851c69504663b5/Desarrollo-e-innovacion-en-ingenieria-ed-2.pdf#page=162)

Takeyas, B. L. (2007). Introducción a la inteligencia artificial. Instituto Tecnológico de Nuevo Laredo.

<http://itnuevolaredo.edu.mx/takeyas/Articulos/Inteligencia%20Artificial/ARTICULO%20Introducción%20a%20la%20Inteligencia%20Artificial.pdf>

Tetteh, G. O., Gocht, A., Erasmi, S., Schwieder, M., & Conrad, C. (2021). Evaluation of Sentinel-1 and Sentinel-2 feature sets for delineating agricultural fields in heterogeneous landscapes. IEEE Access, 9, 116702-116719. <https://ieeexplore.ieee.org/abstract/document/9515974>

Technologies, D., & Technologies, D. (2018). Inteligencia Artificial: ¿Qué es el aprendizaje de refuerzo? Una Explicación Simple y Ejemplos Prácticos. Dell Technologies. Inteligencia Artificial: ¿Qué es el aprendizaje de refuerzo? Una Explicación Simple y Ejemplos Prácticos

Valverde Rebaza, J. (2010). Detección de bordes mediante el algoritmo de Canny [Trabajo de investigación]. Escuela Académico Profesional de Informática, Universidad Nacional de Trujillo.

Watkins, B., & Van Niekerk, A. (2019). A comparison of object-based image analysis approaches for field boundary delineation using multi-temporal Sentinel-2 imagery. Computers and Electronics in Agriculture, 158, 294-302.  
<https://www.sciencedirect.com/science/article/abs/pii/S0168169918317733?via%3Dhub>

Zhou, Z. H. (2021). Machine learning. Springer Nature. Recuperado de  
<https://books.google.es/books?id=ctM-EAAAQBAJ>

## ANEXO A

Nombre del Documento	Segmentation of agricultural parcels in satellite images based on historical vegetation index data
Autor	Maria Pavlova, Valerii Timofeev, Dmitry Bocharov, Irina Kunina, Anna Smagina, Mikhail Zagarev
Referencia Bibliográfica (APA 7)	Pavlova, M.; Timofeev, V.; Bocharov, D.; Kunina, I.; Smagina, A.; Zagarev, M. Segmentation of Agricultural Parcels in Satellite Images Based on Historical Vegetation Index Data. Preprints 2021, 2021120243. <a href="https://doi.org/10.20944/preprints202112.0243.v1">https://doi.org/10.20944/preprints202112.0243.v1</a> .
Palabras Claves del Artículo	Teledetección, administración de tierras, procesamiento de imagen multiespectral, agricultura digital, mapa de la cubierta terrestre, límite del campo agrícola.
Descripción	En este artículo se propone un nuevo algoritmo basado en la data histórica del índice de vegetación obtenido a través de los satélites Sentinel-2. Este algoritmo incluye pasos básicos, detección de parcelas, cálculo de mapas de contorno, la segmentación de parcelas utilizando los contornos, y la computación de componentes conectados, y la extracción de su contorno. Se muestra cómo al uso del índice de datos de vegetación, y mapas de contornos pueden dar como resultado una segmentación de campos agrícolas mucho más precisa.
Observaciones (Datos utilizados, método/algoritmos utilizados, Métodos de Preprocesado, Parámetros de Evaluación,	<ul style="list-style-type: none"> <li>• Se definieron 4 regiones en Rusia y Ucrania (dos en cada país). Cada área era rectangular de 57300x57300, con</li> </ul>

<p>entre otros.)</p>	<p>una resolución de 10 m /píxel.</p> <ul style="list-style-type: none"> <li>● Los datos fueron tomados de los sistemas de teledetección Sentinel-2A y Sentinel-2B. El set contaba con 13 imágenes en formato Jpeg2000, tomados en diferentes rangos espectrales y distintas resoluciones desde 60 m / píxel a 10 m / píxel, con un tamaño máximo de 10800 x 10800 pixeles.</li> <li>● El proceso de reconocimiento de límites se dividió en 5 etapas: (1) Cálculo MSAVI2 y agregación, (2) Detección de regiones agrícolas, (3) Obtención y agregación de mapa de límites, (4) La segmentación de regiones, tomando en cuenta los límites detectados. (5) Computación de los 8 componentes conectados.</li> </ul> <p>La localización de regiones fue realizada agrupando pixeles en dos grupos: pixeles de campos agrícolas, y pixeles de campos no agrícolas, para esto fue implementado el criterio Otsu. Para la obtención del mapa de contornos, fue implementado el algoritmo de Canny.</p> <ul style="list-style-type: none"> <li>● Fue usado un enfoque basado en objetos para la evaluación de los resultados de la estimación de la segmentación de los límites de los campos de cultivo. También se realizaron métricas de estimación basadas en pixeles.</li> </ul>
----------------------	---

Nombre del Documento	A comparison of object-based image analysis
----------------------	---

	approaches for field boundary delineation using multi-temporal sentinel-2 imagery.
Autor	Barry Watkins, and Adriaan Van Niekerk
Referencia Bibliográfica	Watkins, B., & Van Niekerk, A. (2019). A comparison of object-based image analysis approaches for field boundary delineation using multi-temporal Sentinel-2 imagery. Computers and Electronics in Agriculture, 158, 294-302. <a href="https://doi.org/10.1016/j.compag.2019.02.009">https://doi.org/10.1016/j.compag.2019.02.009</a>
Palabras Claves del Artículo	Segmentación, Sentinel-2, Observación de tierra, delineación de bordes de campos de cultivo, detección de límites, clasificación basada en el conocimiento.
Descripción	<p>En este artículo, se evalúa las metodologías para la delineación de contornos de campos de cultivos, haciendo uso de data multi temporal del Sentinel-2. Se utiliza la detección de contornos aplicada a muchas imágenes obtenidas durante una sesión de crecimiento, y segmentación de imagen para delimitar los campos, huertos y viñedos.</p> <p>Se utilizaron dos operadores de detección, y tres algoritmos de segmentación de imagen combinados y evaluados, obteniendo seis escenarios. La evaluación de la precisión consistió tanto en métricas de área como de borde. Los resultados finales mostraron que la segmentación “watershed” combinada con la detección Canny obtuvo los bordes más precisos. Por otro lado, el algoritmo Scharr produjo bordes gruesos, lo que causó errores de posición a lo largo de los contornos. Y los</p>

	<p>algoritmos de segmentación de multi-resolución y multi-umbral producían imprecisiones de entre 1 a 3 píxeles. Por lo que se concluyó que la combinación de la detección de bordes Canny, y el algoritmo de segmentación “watershed” es lo recomendado para la delineación de bordes de campos.</p>
Observaciones (Datos utilizados, método/algoritmos utilizados, Métodos de Preprocesado, Parámetros de Evaluación, entre otros.)	<p><b>Definición del área de estudio</b></p> <p>El área de estudio fue una zona de Suráfrica, cerca de los pueblos de Warrenton y Hartswater. El área escogida tenía un área aproximada de 369.5 kilómetros cuadrados. La zona tiene clima semiárido, inviernos secos, y largos veranos húmedos. La lluvia anual es de 450 mm. En la zona los cultivos principales eran de maíz, cebada, cacahuetes, nueces, etc.</p> <p>Para el estudio, se puso énfasis en la estación de verano, cuando hay más diversidad de cultivos.</p> <p><b>Definición de data:</b></p> <p>Se tomaron siete imágenes de Sentinel-2 libres de nubes, tomadas entre noviembre de 2016 y abril de 2017. Las imágenes tomadas de los satélites tenían el formato Leve-1C, el cual es corregido a la parte superior de reflectancia de la atmósfera (TOA).</p> <p><b>Detección de contornos:</b></p> <p>Se ocupa de un algoritmo de detección de contorno, el cual genera una imagen en escala de grises que representa los bordes de los objetos, donde los niveles altos de grises sugieren una discontinuidad con los píxeles de alrededor. Para este estudio se ocuparon los operadores Canny y</p>

	<p>Scharr, debido a su superior rendimiento en comparación de otros algoritmos.</p> <p><b>Agregación de capa de contornos:</b></p> <p>Se agregan las capas multitemporales (en este caso 28 capas, resultado de 4 bandas por 7 imágenes). Se suman los pesos de las capas de contorno para combinar todo en una sola composición.</p> <p><b>Segmentación de imágenes:</b></p> <p>Para esto se ocuparon 3 algoritmos de segmentación: MRS (Multi-resolution segmentation), WS (Watershed segmentation) y MTS (Multi-threshold segmentation). Los primeros dos, fueron implementados por su frecuente uso en la agricultura. El MTS fue usado por ser una técnica efectiva para la rápida segmentación de imágenes basada en la distribución del histograma de las imágenes de entrada.</p> <p><b>Exclusión de tierra no cultivada</b></p> <p>Se tomaron características del NDVI como entradas de una clasificación y un algoritmo de árbol de regresión, para generar un árbol de decisión para separar los campos cultivados y los no cultivados.</p> <p><b>Eliminación de ruido</b></p> <p>Se agregaron reglas para descartar pequeños objetos (menores de 40 píxeles) que estuviesen completamente rodeados por cultivos.</p> <p><b>Evaluación</b></p> <p>Para evaluar la precisión de los contornos, se</p>
--	---

	utilizó una métrica llamada “el error absoluto medio” (MAE). Este considera la distancia euclíadiana (ED) entre el centro de dos celdas representando la referencia y el borde generado. Se calcularon dos instancias de MAE, para evaluar qué tanto coinciden los bordes extraídos, con los de bordes referencia en la base de datos.
--	--

Nombre del Documento	Evaluation of Sentinel-1 and Sentinel-2 feature sets for Delineating Agricultural Fields in Heterogeneous Landscapes
Autor	Gideon Okpoti Tetteh, Alexander Gocht, Stefan Erasmi, Marcel Schwieder, Christopher Conrad
Referencia Bibliográfica	Tetteh, G. O., Gocht, A., Erasmi, S., Schwieder, M., & Conrad, C. (2021). Evaluation of Sentinel-1 and Sentinel-2 Feature Sets for Delineating Agricultural Fields in Heterogeneous Landscapes. <i>IEEE Access</i> , 9, 116702–116719. <a href="https://doi.org/10.1109/access.2021.3105903">https://doi.org/10.1109/access.2021.3105903</a>
Palabras Claves del Artículo	Delineación de campos agrícolas, índices de bandas, variables agrícolas esenciales, combinación de características, segmentación de imágenes, intersección sobre unión, detección remota, optimización de segmentación, segmentación de imagen, resolución espacial, sensores remotos, satélites artificiales.
Descripción	En este artículo se evalúa la precisión con la que se pueden delinear los campos agrícolas a partir de imágenes Sentinel-1 (S1) y Sentinel-2 (S2) en diferentes paisajes agrícolas durante la

	<p>temporada de crecimiento.</p> <p>De igual manera, busca identificar el conjunto de características óptimo de las imágenes S1 y S2 para segmentar campos agrícolas, analizar la evolución de la precisión de los campos agrícolas segmentados de los conjuntos de características S1 y S2 a lo largo de la temporada de crecimiento, y evaluar la precisión que se puede lograr para diferentes tamaños de campo. Para lograrlo se emplea el algoritmo de segmentación de resolución múltiple (MRS) en eCognition para segmentar campos agrícolas de diferentes conjuntos de características generados a partir de imágenes S1 y S2 adquiridas entre marzo y octubre de 2018 en Baja Sajonia, Alemania.</p>
Observaciones (Datos utilizados, método/algoritmos utilizados, Métodos de Preprocesado, Parámetros de Evaluación, entre otros.)	<p><b>Segmentación de campos agrícolas:</b></p> <ul style="list-style-type: none"> <li>Uso de la segmentación supervisada basada en el algoritmo de segmentación multirresolución (MRS) para identificar el conjunto de características óptimas de las imágenes S1 y S2 para la delineación del campo.</li> </ul> <p><b>Evaluación de la precisión de la segmentación</b></p> <ul style="list-style-type: none"> <li>Categorización de los campos según su tamaño.</li> </ul> <p><b>Datos del satélite</b></p> <ul style="list-style-type: none"> <li>Para el mapeo a gran escala de los tipos de uso de la tierra agrícola se hace uso de compuestos medios mensuales (MMC) de imágenes S1 y S2 de marzo a octubre de 2018.</li> </ul>

	<p><b>Procesamiento de datos</b></p> <ul style="list-style-type: none"> <li>• El procesamiento de S1 fue realizado con CODE-DE con la plataforma de aplicaciones Sentinel (SNAP) utilizando el procedimiento estándar de aplicar un archivo de órbita, eliminar el ruido del borde GRD, eliminar el ruido térmico, la calibración y la corrección del terreno. Para S2 se hace el uso de FORCE (Framework for Operational Radiometric Correction for Environmental Monitoring).</li> </ul> <p><b>Datos de referencia para evaluación y optimización de la segmentación</b></p> <ul style="list-style-type: none"> <li>• Para la evaluación y optimización de la segmentación, se hace uso de los datos de la Aplicación de ayuda geoespacial (GSAA) de 2018 que cubren los TT.</li> </ul> <p><b>Selección de las placas de prueba(TTs)</b></p> <ul style="list-style-type: none"> <li>• La selección de los TT(SELECCIÓN DE LAS PLACAS DE PRUEBA) basada en cuatro criterios: un alto porcentaje de cobertura de LU agrícola, un alto número de parcelas de referencia para la evaluación de la segmentación, la presencia de campos agrícolas tanto grandes como pequeños, y una distribución variable del factor de forma por mosaico.</li> </ul> <p><b>Extracción de áreas no agrícolas</b></p> <ul style="list-style-type: none"> <li>• Uso de capa de vector agrícola extraída de ATKIS para eliminar todas las áreas no agrícolas.</li> </ul>
--	---

	<p><b>Corrección de efectos atmosféricos</b></p> <ul style="list-style-type: none"> <li>• Proceso automatizado basado en FORCE utilizado para identificar y reemplazar nubes. Las nubes y las sombras de las nubes se detectan y enmascaran mediante el algoritmo Fmask</li> </ul>
--	--

Nombre del Documento	Farm Parcel Delineation Using Spatio-temporal Convolutional Networks
Autor	Han Lin Aung, Burak Uzkent, Marshall Burke, David Lobel, Stefano Ermón.
Referencia Bibliográfica	Aung, H. L., Uzkent, B., Burke, M., Lobell, D. B., & Ermon, S. (2020). Farm Parcel Delineation Using Spatio-temporal Convolutional Networks. <i>Computer Vision and Pattern Recognition</i> . <a href="https://doi.org/10.1109/cvprw50498.2020.00046">https://doi.org/10.1109/cvprw50498.2020.00046</a>
Palabras Claves del Artículo	Satélites, cambio climático, segmentación de imagen, Meteorología, Aprendizaje automático, Visión por computador, agricultura, mitigación climática, Segmentación de imagen , irrigación, delimitación de parcelas agrícolas, límites de parcela, áreas de parcela, redes convolucionales espacio-temporales, políticas gubernamentales aguas abajo , asignación de tierras, fertilización, gases de invernadero, sectores seguros agrarios, imágenes satelitales, U-Nets.
Descripción	Este artículo propone el uso de conjuntos de

	<p>datos de código abierto para segmentar áreas y límites de parcelas agrícolas en imágenes de satélite. En particular, entrenando variantes del modelo U-Net en las imágenes de Sentinel-2 dadas las máscaras de área/límites correspondientes.</p> <p>El uso de imágenes satelitales puede ser una forma escalable y rentable de realizar la tarea de delimitación de parcelas agrícolas para recopilar datos y utilizando estas imágenes satelitales en dos enfoques: 1) Segmentación de los límites de las parcelas y 2) Segmentación de las áreas de las parcelas.</p>
Observaciones (Datos utilizados, método/algoritmos utilizados, Métodos de Preprocesado, Parámetros de Evaluación, entre otros.)	<p><b>Datos Utilizados</b></p> <ul style="list-style-type: none"> <li>• El conjunto de datos representados consiste en imágenes satelitales Sentinel-2 de Francia (2017), junto con las correspondientes máscaras</li> <li>• { binarias de límites y áreas de parcelas agrícolas</li> </ul> <p><b>Generación de datos</b></p> <ul style="list-style-type: none"> <li>• Descarga del archivo de forma que consiste en los polígonos (en el sistema de coordenadas Lambert-93) de los límites de las parcelas agrícolas de Francia (2017), se presenta un polígono de muestra del shapefile. Posteriormente, se proyecta el sistema de coordenadas existente de todos los polígonos al sistema de coordenadas geográficas de latitud/longitud. Muestreo aleatorio de 2000Ci de 2000 regiones que tienen al menos una</li> </ul>

	<p>parcela agrícola en sus proximidades (dentro de un radio de 2,24 km) en Francia.</p> <ul style="list-style-type: none"> <li>Para dibujar los límites, se hace uso de la funcionalidad de polilíneas de OpenCV.</li> </ul> <p><b>Métodos de segmentación</b></p> <ul style="list-style-type: none"> <li>Dado el conjunto de datos representado con <math>(X_t, Y)</math> se establecen dos métodos para segmentar los límites y áreas de las tierras agrícolas. El primer método usa solo una imagen que es representativa de rango de tiempo (abril-junio). El segundo método utiliza 3 imágenes de 3 rangos de tiempo.</li> <li>Uso de UNet espacial para la segmentación de imágenes y generar la máscara binaria. Utilizando una U-Net espacial entrenada desde cero y también con una U-Net espacial previamente entrenada cuyo codificador se inicializa con lo aprendido en la tarea de clasificación de ImageNet.</li> <li>Uso de Spatio-temporal U-Net que toma en cuenta 3 imágenes de diferentes marcas de tiempo para generar la máscara binaria.</li> </ul> <p><b>Evaluación de los modelos</b></p> <ul style="list-style-type: none"> <li>Evaluación de los modelos en la puntuación de DICE y por medio de la precisión por píxel.</li> </ul> <p><b>Detalles de implementación</b></p> <ul style="list-style-type: none"> <li>Todos los modelos U-Net son implementados en Keras.</li> </ul>
--	---

	<ul style="list-style-type: none"> <li>Los modelos fueron entrenados usando el optimizador Adam.</li> </ul>
--	---

Nombre del Documento	A Sentinel-2 multi-year, multi-country benchmark dataset for crop classification and segmentation with deep learning
Autor	Dimitrios Sykas, Maria Sdraka, Dimitrios Zografakis, Ioannis Papoutsis. Institute of Astronomy, Astrophysics, Space Applications & Remote Sensing, National Observatory of Athens
Referencia Bibliográfica	Sykas, D., Sdraka, M., Zografakis, D., & Papoutsis, I. (2022). A Sentinel-2 multi-year, multi-country benchmark dataset for crop classification and segmentation with deep learning. <i>arXiv (Cornell University)</i> . <a href="https://doi.org/10.48550/arxiv.2204.00951">https://doi.org/10.48550/arxiv.2204.00951</a>
Palabras Claves del Artículo	Multipaís, marco de referencia, dataset, declaraciones de granjeros, imágenes satelitales, taxonomía de armonización de cultivos, Deep Learning.
Descripción	Este paper explica a detalle la implementación, preparación y ejecución de un modelo de Deep Learning para la clasificación y segmentación de cultivos en Francia y la ciudad de Cataluña en España en el período 2016-2020. Este marco de referencia llamado “Sen4AgriNet” es un conjunto de datos adaptado para Machine Learning que puede ser utilizado en diversidad de aplicaciones como la clasificación del tipo de cultivo, extracción de parcelas, conteo de parcelas y segmentación semántica. Los

	<p>creadores diseñaron dos prototipos de este modelo: uno adaptado para aplicaciones basadas en píxeles y otro adaptado para aplicaciones basadas en objetos.</p>
Observaciones (Datos utilizados, método/algoritmos utilizados, Métodos de Preprocesado, Parámetros de Evaluación, entre otros.)	<p><b>Datos Utilizados:</b></p> <ul style="list-style-type: none"> <li>Los datos utilizados en el conjunto de datos son una extracción de imágenes satelitales provenientes del Sentinel-2, estas referentes a campos de cultivo en Francia y la ciudad de Cataluña en España.</li> </ul> <p><b>Generación de datos:</b></p> <ul style="list-style-type: none"> <li>Se estima que el tamaño total de Sen4AgriNet para el periodo de 2016-2020 oscila entre los 10 TB de almacenamiento, lo que se traduce en alrededor de 225,000 parches.</li> </ul> <p><b>Definición de los Datasets:</b></p> <ul style="list-style-type: none"> <li>El dataset de parches ensamblados (PAD) por sus siglas en inglés, generaliza el problema de clasificación para extracción de parcelas y segmentación y etiquetado semántico.</li> <li>El segundo se denomina dataset basado en objetos agregados (OAD), este se construye sobre el PAD, agregando valores de ráster a la parcela, de modo que se produce una muestra de parcela que contiene todos los valores espectrales promediados con estadísticas para todas las bandas.</li> <li>Ambos datasets se examinan bajo tres escenarios donde buscan mostrar y</li> </ul>

modelar los efectos de la variación espacio-temporal con el paso de los años y los efectos en diferentes países.

#### **Entrenamiento de los datasets:**

- Ambos datasets se sometieron al análisis dentro de 3 escenarios, los cuales iban a ser comparados en base a sus resultados en la fase de evaluación.
- En el dataset de parches ensamblados (PAD) se utilizaron tres modelos populares para la segmentación de imágenes: U-Net, ConvLSTM y ConvSTAR.
- Para el dataset basado en objetos (OAD), se hizo uso de un algoritmo de Deep Learning llamado TempCNN, la cual es una red neuronal convolucional (CNN) y en la que, cada convolución representa las etapas para capturar el aspecto temporal del dataset. Además, se llevó a comparación en simultáneo a otros dos tipos de algoritmos, un algoritmo de tipo LSTM y una red de transformadores, la cual representa un digno oponente a evaluar frente a una LSTM.

#### **Parámetros y proceso de evaluación:**

- El principal parámetro de evaluación en este proyecto fue el de analizar y determinar cuál de los tres algoritmos implementados en cada uno de los datasets presentaba un mejor porcentaje de precisión.
- Asimismo se buscó determinar con cuál

	<p>de los dos datasets se obtenían mejores resultados.</p> <ul style="list-style-type: none"> <li>● Finalizado el proceso de entrenamiento y prueba, los resultados determinaron lo siguiente:           <ol style="list-style-type: none"> <li>1. Luego del análisis en los tres escenarios, para el dataset de parches ensamblados (PAD), el algoritmo que mejor porcentaje de precisión mostró fue el U-Net.</li> <li>2. En el caso del dataset basado en objetos (OAD), luego de la experimentación en los tres escenarios el algoritmo que mejores resultados mostró fue el algoritmo LSTM.</li> </ol> </li> <li>● Basado en los resultados obtenidos se determinó que la experimentación con el dataset basado en objetos (OAD) arrojó mejores resultados en comparación con el otro dataset para todos los escenarios expuestos. Sin embargo, se hace énfasis de que para este proyecto el dataset basado en objetos ha sido más útil y preciso, y que a nivel general ambos tipos de datasets son muy útiles y capaces de resolver problemas asociados al Machine Learning.</li> </ul>
--	--

Nombre del Documento	An open AI-ready dataset to map field boundaries with Sentinel-2 an aerial photography
----------------------	--

Autor	<p>Raphaël d'Andrimont, Martin Claverie, Pieter Kempeneers, Davide Muraro, Momchil Yordanov, Devis Peressutti, Matej Batič, and François Waldner.</p> <p>European Commission Joint Research Centre (JRC), Ispra, Italy.</p> <p>Sinergise, Ljubljana, Slovenia.</p>
Referencia Bibliográfica	<p>d'Andrimont, R., Claverie, M., Kempeneers, P., Muraro, D., Yordanov, M., Peressutti, D., Batič, M., &amp; Waldner, F. (2023). AI4Boundaries: an open AI-ready dataset to map field boundaries with Sentinel-2 and aerial photography. <i>Earth System Science Data</i>, 15(1), 317-329. <a href="https://doi.org/10.5194/essd-15-317-2023">https://doi.org/10.5194/essd-15-317-2023</a></p>
Palabras Claves del Artículo	<p>Límites de campo, ortofotografía, mapeo de tipo de cultivos, estimación de rendimiento, imágenes satelitales/aéreas, Sentinel-2, Machine Learning, Deep Learning,</p>
Descripción	<p>Este paper explica el proyecto titulado “AI4Boundaries”, un modelo de Machine Learning,utilizable para entrenar y comparar modelos en la tarea de detección de límites de parcelas agrícolas. En este proyecto se diseñaron dos tipos de modelos basados en datasets extraídos de imágenes satelitales provenientes del satélite Sentinel-2: un compuesto imágenes a gran escala (10m) y un dataset con un set de ortofotografías a escala (1m) provenientes de muchos países en Europa.</p>

<p>Observaciones (Datos utilizados, método/algoritmos utilizados, Métodos de Preprocesado, Parámetros de Evaluación, entre otros.)</p>	<p><b>Datos utilizados:</b></p> <ul style="list-style-type: none"> <li>Este proyecto implementa dos datasets: El primero es un dataset mensual del Sentinel-2 de escala 10m para análisis a gran escala en retrospectiva. El segundo es un dataset ortofotográfico de escala 1m para análisis a escala regional.</li> <li>El dataset resultante de “AI4Boundaries” consta de 7831 muestras de 256x256 píxeles (referentes al primer dataset) y otras con medidas de 512x512 píxeles para las ortofotografías aéreas de 1m.</li> </ul> <p><b>Generación de datos:</b></p> <ul style="list-style-type: none"> <li>Para la obtención de las muestras se diseñó un método de muestreo aleatorio estratificado para extraer chips de imagen de una variedad de paisajes y utilizando una cuadrícula de 4x4 km que se superpuso en cada país/región se pudo analizar en cada celda de la cuadrícula el porcentaje calculado de fracción de campo y la relación perímetro-área de cada porción de terreno.</li> </ul> <p><b>Entrenamiento del dataset:</b></p> <ul style="list-style-type: none"> <li>Se utilizó el procesador Sen2Cor para obtener datos de reflectancia de la parte superior de la atmósfera (TOA) de Sentinel-2.</li> <li>Del dataset resultante se descartaron alrededor de 233 muestras (casi exclusivamente provenientes de</li> </ul>
--	---

	<p>Suecia) que no contaban con ortofotografía disponible.</p> <ul style="list-style-type: none"> <li>● Los datos para entrenamiento y pruebas se distribuyeron de la siguiente manera: 70% de los datos para entrenamiento, 15% para validación y 15% para prueba.</li> <li>● Las imágenes extraídas y analizadas del Sentinel-2 muestran etiquetas/composiciones de diferentes tipos como: NDVI, RGB, NIR. Además la etiqueta en la misma resolución y extensión consta de cuatro capas: Máscara de extensión, máscara de borde, máscara de distancia y una enumeración de campos.</li> </ul> <p><b>Observaciones extra:</b></p> <ul style="list-style-type: none"> <li>● El conjunto de datos Sentinel-2 de “AI4boundaries” se ha utilizado para entrenar un modelo que está disponible en Euro Data Cube (EDC) como un algoritmo para la delimitación automática a pedido de los límites de los campos agrícolas en un área de interés definida por el usuario.</li> </ul>
--	---

Nombre del Documento	Automated delineation of agricultural field boundaries from Sentinel-2 images using recurrent residual U-net
Autor	Huanxue Zhang, Mingxu Liu, Yuji Wang, Jiali Shang, Xiangliang Liu, Bin Li, Aiqi Song, Qiangzi Li

Referencia Bibliográfica	Zhang, H., Liu, M., Wang, Y., Shang, J., Liu, X., Li, B., Song, A., & Li, Q. (2021). Automated delineation of agricultural field boundaries from Sentinel-2 images using recurrent residual U-Net. International Journal of Applied Earth Observation and Geoinformation, 105, 102557. <a href="https://doi.org/10.1016/j.jag.2021.102557">https://doi.org/10.1016/j.jag.2021.102557</a>
Palabras Claves del Artículo	Agricultura, Delimitación Automatizada, Límite del Campo, Red-U Residual Recurrente, Sentinel-2, teledetección
Descripción	
Observaciones (Datos utilizados, método/algoritmos utilizados, Métodos de Preprocesado, Parámetros de Evaluación, entre otros.)	<ul style="list-style-type: none"> <li>● Provincia de Heilongjiang, situada al noreste de China, con una superficie total de 473.000 kilómetros cuadrados. Se eligieron cuatro áreas de estudio: Condado de Mudanjiang (YM), condado de Huachuan en la ciudad de Jiamusi (HC), el condado de Kedong en la ciudad de Qiqihar (KD) y el condado de Wangkui en la ciudad de Suihua (WK). YM está situado en una zona montañosa con tierras de cultivos fragmentadas y los otros tres condados se distribuyen principalmente en zonas llanas.</li> <li>● Entre mediados de julio y mediados de septiembre se recogieron siete imágenes del Sentinel-2 libres de nubes que cubrían las cuatro zonas de estudio. El Sentinel-2 proporciona resoluciones espaciales de 10 m, 20 m y 60 m en 13 bandas con un tiempo de revisión de 5</li> </ul>

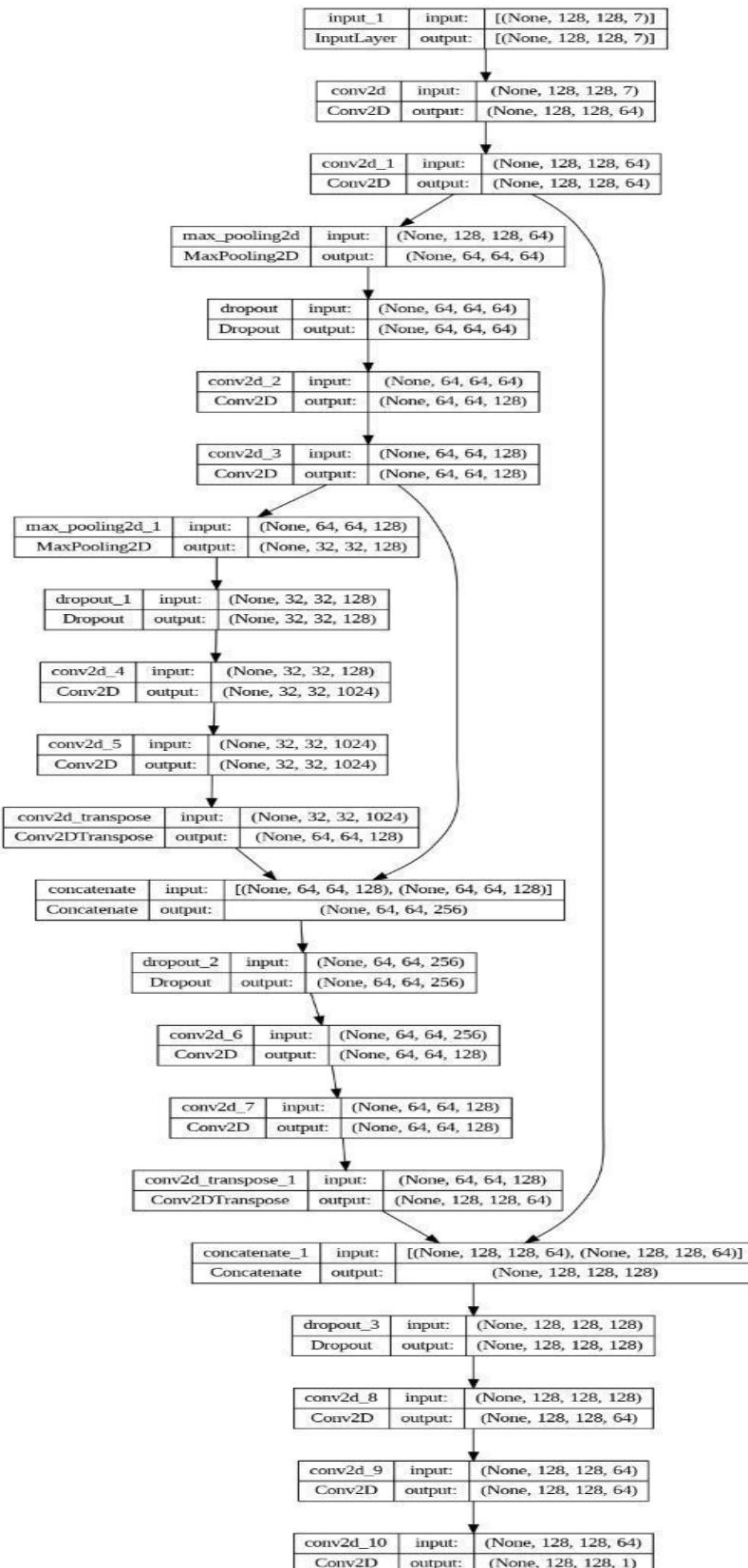
	<p>días. Para el estudio, se utilizaron nueve bandas con una resolución de 10 m y 20 m para la extracción de los límites de campo.</p> <ul style="list-style-type: none"> <li>● Los datos de Sentinel-2 Nivel-1C se obtuvieron y procesaron utilizando el procesador Sen2Cor para convertir la reflectancia de la parte superior de la atmósfera (TOA) en reflectancia de la parte inferior de la atmósfera (BOC). Posteriormente, los resultados se re procesaron a una resolución espacial de 10 m mediante el método de interpolación bilineal utilizando la plataforma de aplicaciones Sentinel (SNAP).</li> <li>● Detección de Bordes basada en Canny: Se realizó un proceso de varias etapas, las cuales fueron: Suavizado mediante un filtro gaussiano, cálculo del gradiente, la supresión no máxima y la segmentación por umbral de histéresis. Las nueve bandas de Sentinel-2 fueron utilizadas para el proceso de detección de bordes, para así obtener nueve imágenes de bordes para cada imagen del set de entrenamiento.</li> <li>● Segmentación de Cuencas: Es un algoritmo basado en regiones adecuado para una imagen de gradiente único. Tiene como objetivo segmentar una región en subregiones utilizando los máximos locales como límites de</li> </ul>
--	--

	<p>separación. El algoritmo utiliza el valor mínimo local como punto semilla.</p> <ul style="list-style-type: none"> <li>● Para este estudio, se utilizó el modelo U-Net con capas y unidad residual, es decir, R2U-Net. Manteniendo las unidades de codificación y descodificación convolucionales del modelo original, también se utilizaron capas convolucionales recurrentes (RCL) con unidades residuales, contribuyendo al desarrollo de un modelo más profundo y eficiente.</li> <li>● Junto con la red se adopta un proceso de normalización por lotes, la cual es una técnica para acelerar el entrenamiento de redes profundas aliviando el problema del desplazamiento interno de covariables.</li> <li>● Las métricas basadas en el área se calculan mediante una matriz de confusión. Los indicadores de precisión incluyen la precisión global (OA), el coeficiente Kappa (K), el error de comisión (CE), el error de omisión (OE) y la precisión-recuperación (PR).</li> <li>● Las métricas basadas en el área se calculan mediante una matriz de confusión. Los indicadores de precisión incluyen la precisión global (OA), el coeficiente Kappa (K), el error de comisión (CE), el error de omisión (OE) y la precisión-recuperación (PR).</li> </ul>
--	--

- La precisión basada en los bordes se evalúa mediante cuatro indicadores (tasa de sobresegmentación, tasa de infrasegmentación, tasa de infrasegmentación).
- La precisión basada en los bordes se evalúa mediante cuatro indicadores (tasa de sobresegmentación, tasa de infrasegmentación, desplazamiento de posición y excentricidad), que reflejan la forma, el tamaño y los desplazamientos de los campos extraídos en relación con los campos objetivo o de referencia, respectivamente. Primero se extrajeron las regiones superpuestas entre los campos de referencia y los campos delineados y, a continuación, se calcularon las cuatro métricas: (1) el área relativa de una región superpuesta a un campo de referencia (el índice de sobre segmentación, RAor); (2) el área relativa de una región superpuesta a un campo extraído (el índice de infra segmentación, RAos); (3) la discrepancia de posición del campo extraído ( $s$ ) respecto a un campo de referencia (el desplazamiento de ubicación, Dsr) calculado como la distancia media entre los centroides de los campos extraídos y los centroides de los campos de referencia; y (4) las diferencias absolutas de forma (factor de excentricidad,  $\varepsilon$ ).

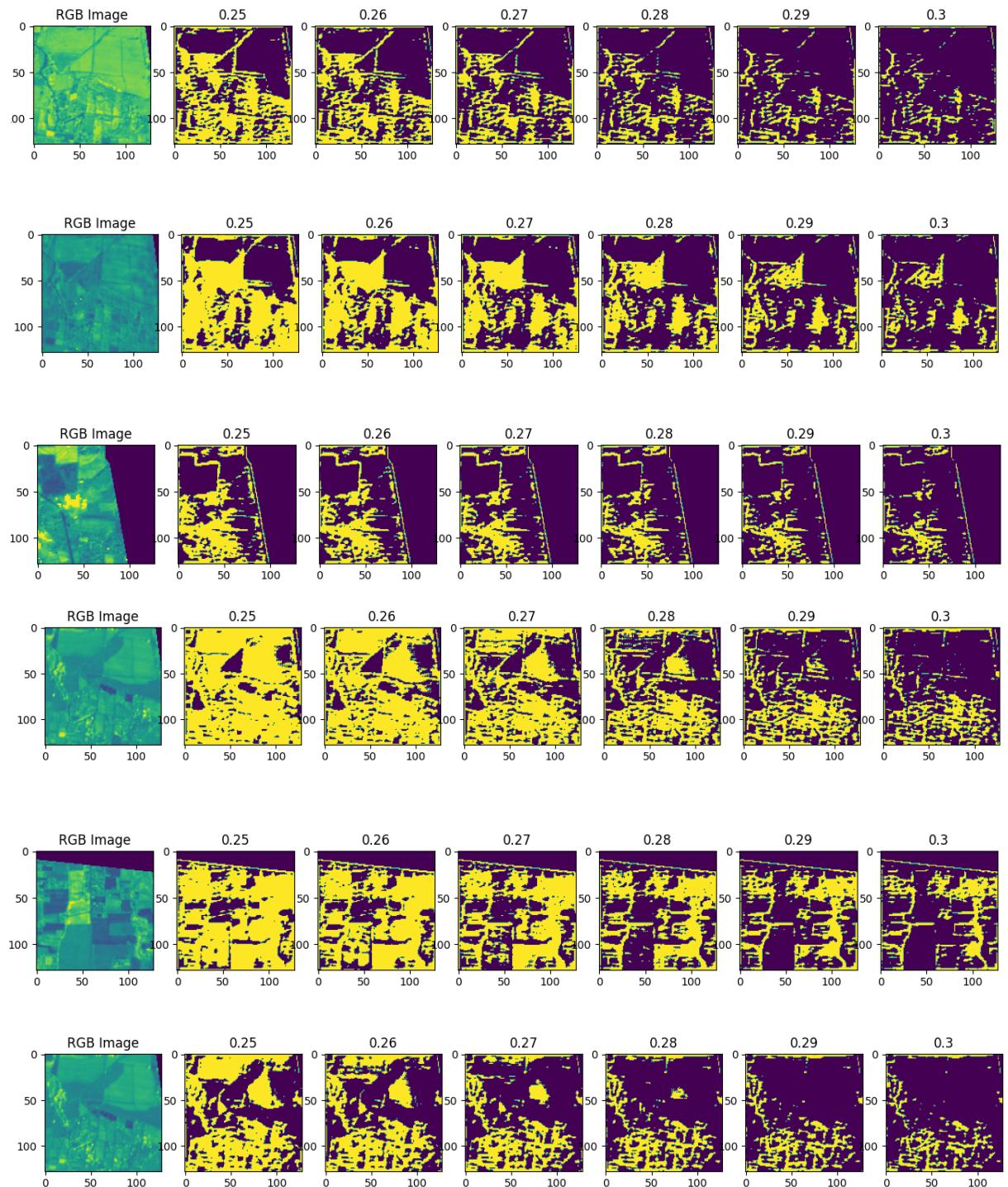
## ANEXO B

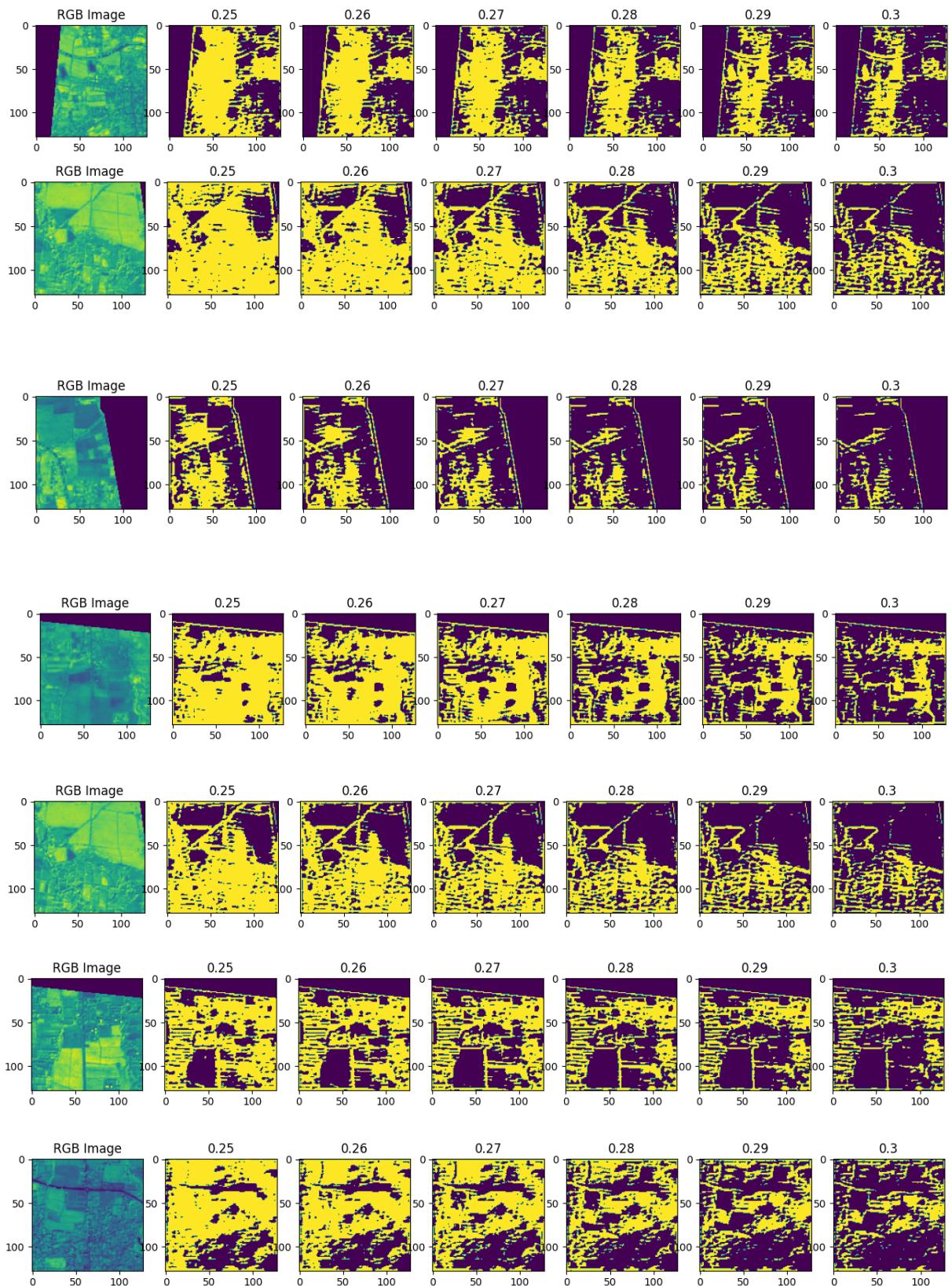
Arquitectura utilizada para la construcción de modelo de red neuronal convolucional.

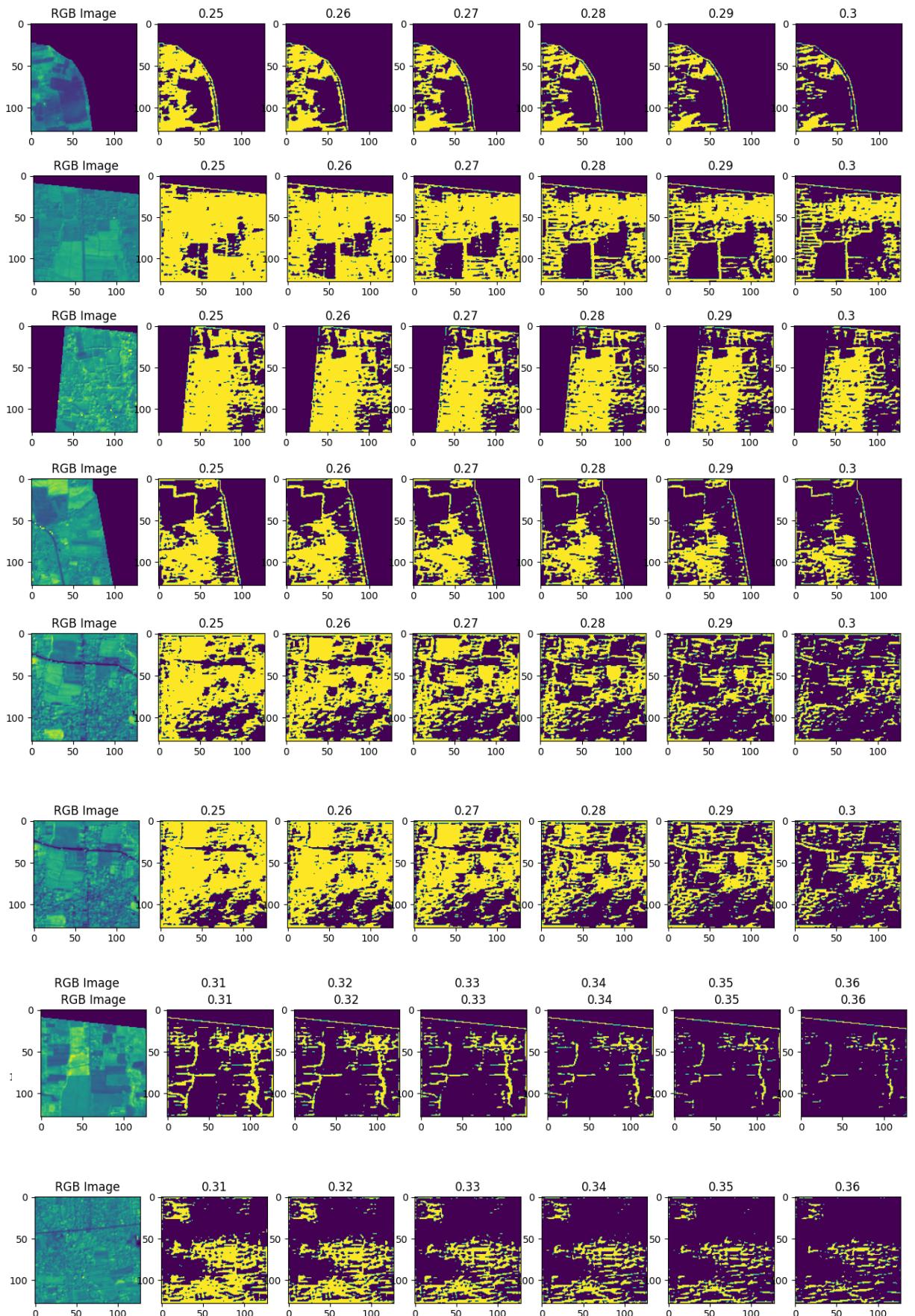


## ANEXO C

Resultados de predicciones a modelo de red neuronal convolucional para la región de Santa María, Usulután.

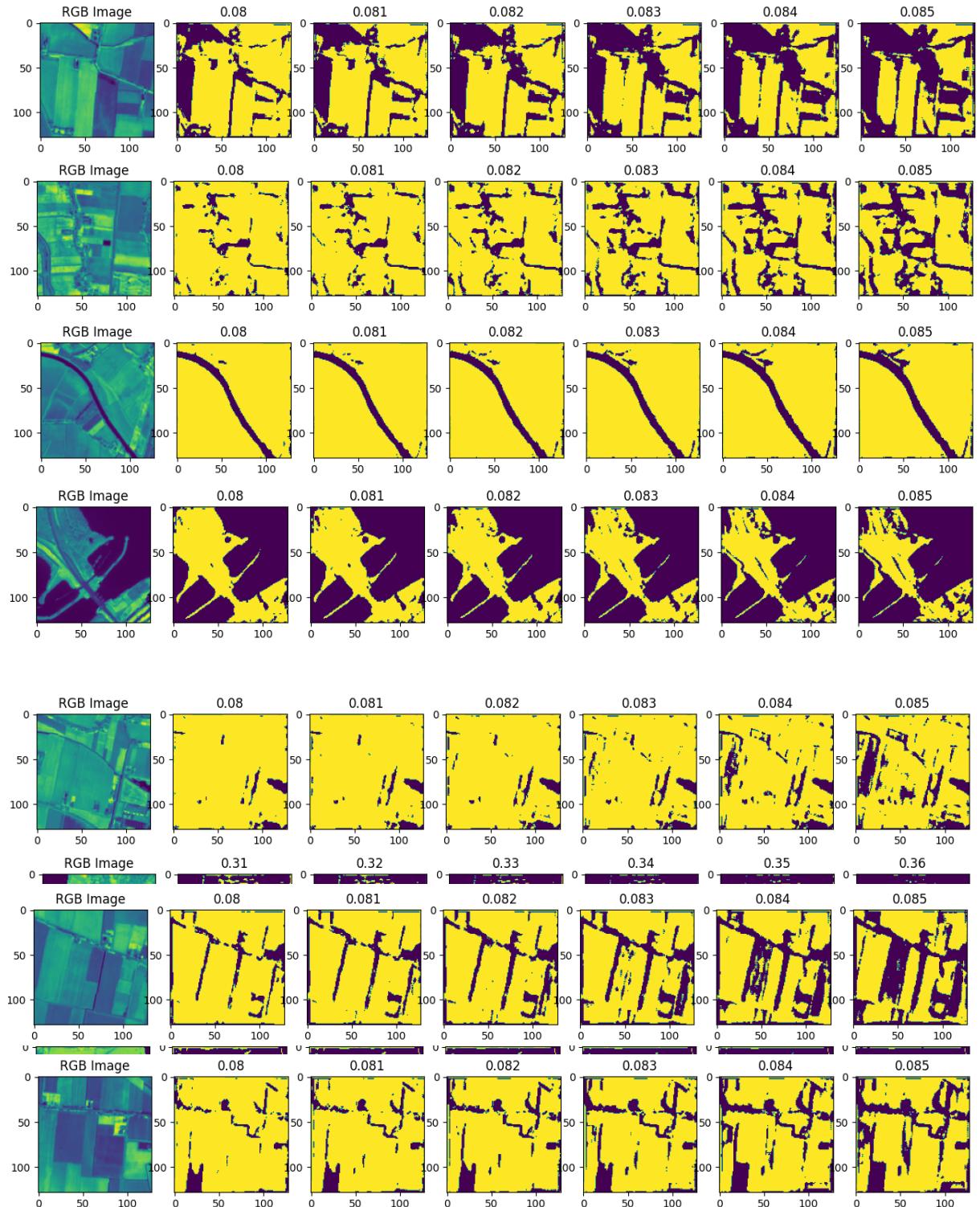


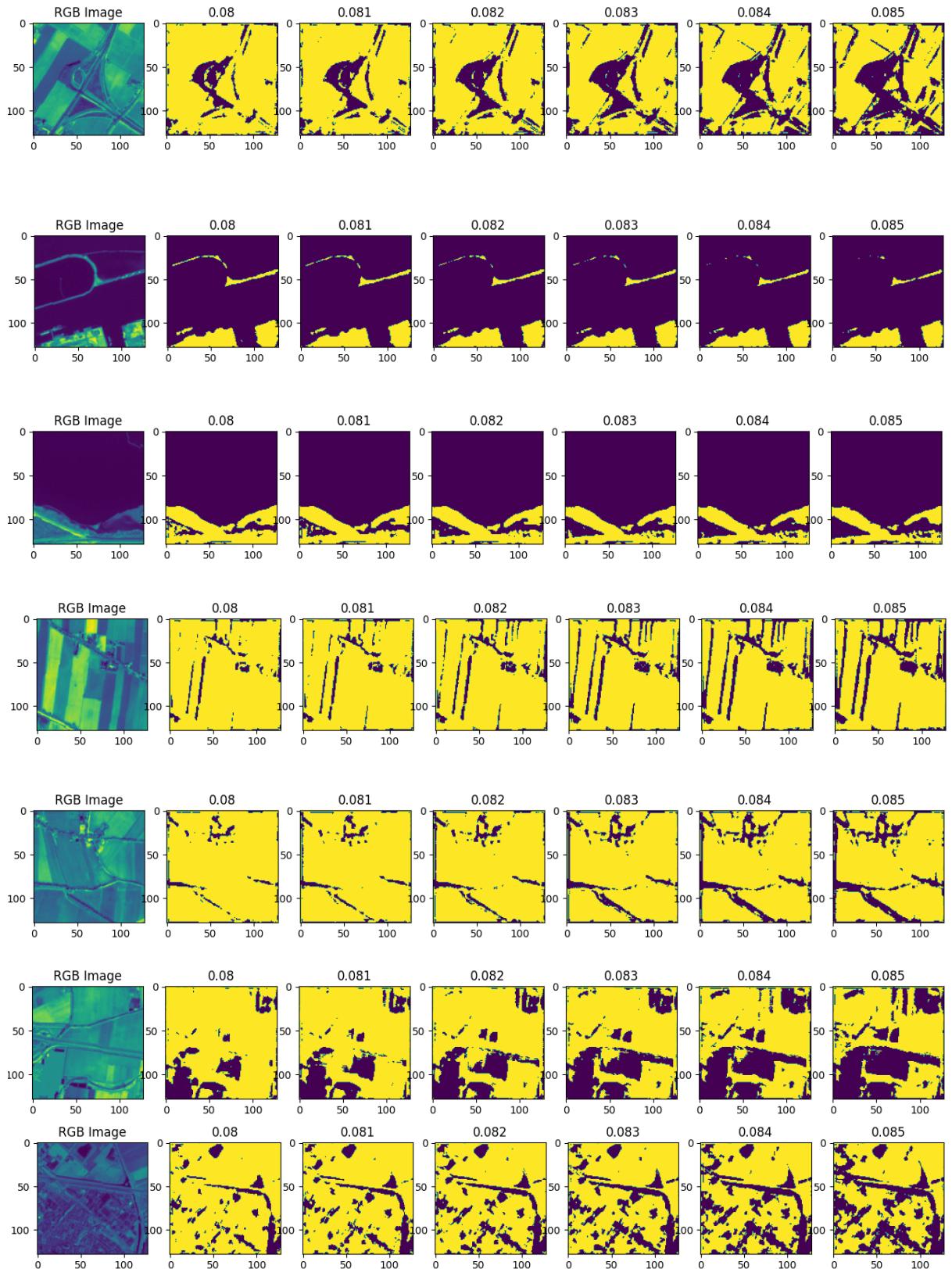


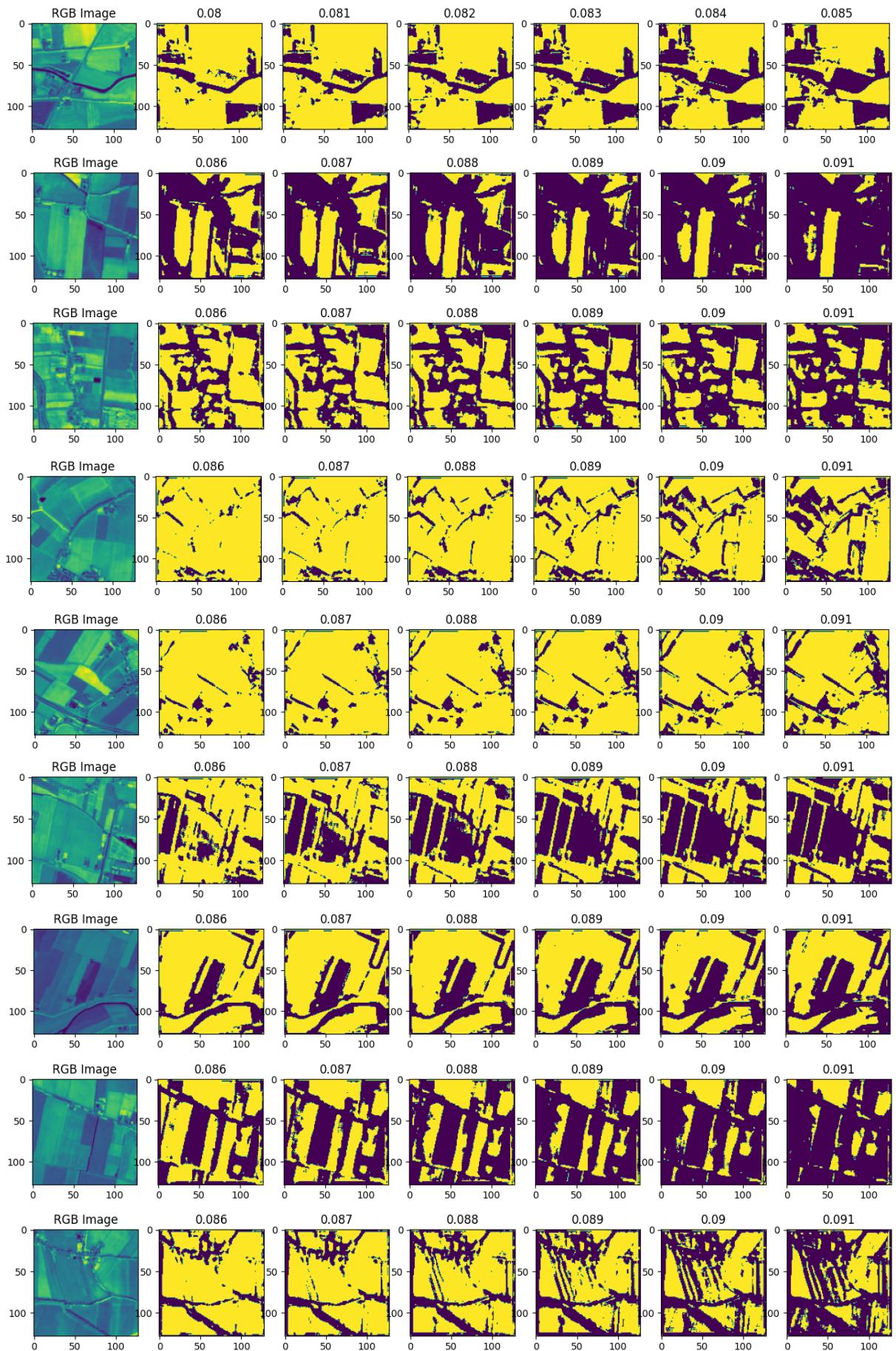


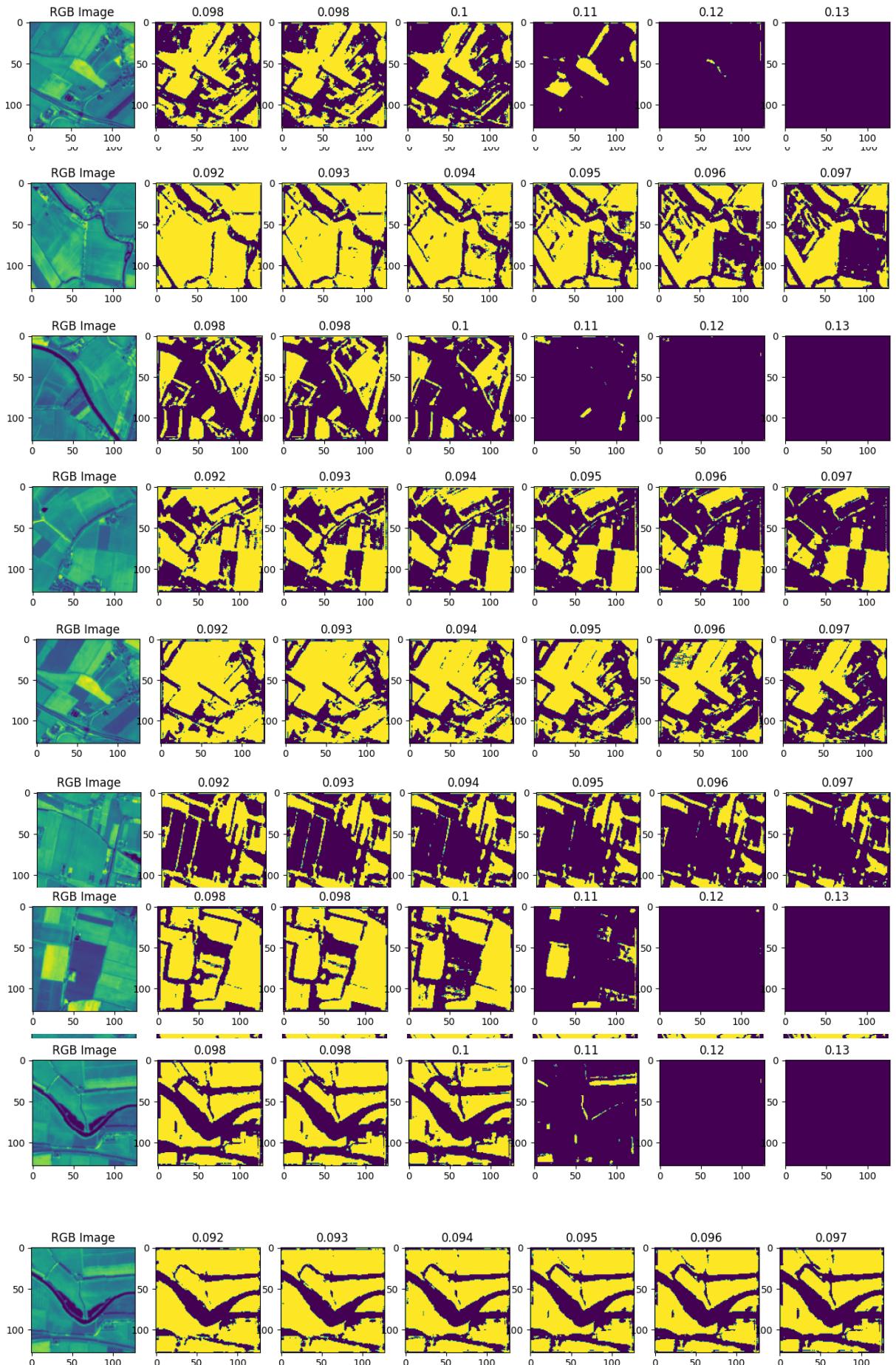
## ANEXO D

Resultados de predicciones a modelo de red neuronal convolucional para la región de Zelanda, Países Bajos









## ANEXO E

Código fuente en el lenguaje de programación Python de implementación del algoritmo de Canny

```
import cv2
from skimage import io
import netCDF4
import matplotlib.pyplot as plt

# Carga el archivo NetCDF
dataset = netCDF4.Dataset('./STAMARIA_S2_2023_cube.nc');

# Tomando una banda
B0_4 = dataset.variables['B08'][0, 0:, 0:];

# Normaliza los valores del arreglo a un rango entre 0 y 255
datos_norm = cv2.normalize(B0_4, None, 0, 255, cv2.NORM_MINMAX, cv2.CV_8U);

# Aplicar suavizado Gaussiano
gauss = cv2.GaussianBlur(datos_norm, (5,5), 0);

# Detectamos los bordes con Canny
canny = cv2.Canny(gauss, 50, 150)

# Buscamos los contornos
(contornos,_) = cv2.findContours(canny, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_NONE)

# Se dibujan los contornos sobre la imagen
cv2.drawContours(B0_4,contornos,-1,(255,0,0), 2)

# Mostrar los datos
fig, (ax1, ax2) = plt.subplots(ncols=2, figsize=(8, 4));

ax1.imshow(canny)
ax1.set_title('Canny result');
ax2.imshow(B0_4)
ax2.set_title('With Contours');
```

## ANEXO F

Código fuente en el lenguaje de programación Python de implementación de la red neuronal convolucional

```
# Se importan los paquetes necesarios
import numpy as np
import geopandas as gpd
import xarray as xr
import rasterio
import tensorflow as tf
from sklearn.model_selection import train_test_split
from tensorflow.keras import layers
from scipy.signal import convolve2d

# Metodo para la normalizacion de las bandas
def normalize_band(band):
    band = np.nan_to_num(band)
    band_min = band.min()
    band_max = band.max()
    band_range = band_max - band_min

    if band_range == 0:
        band_range = 1

    normalized_band = (band - band_min) / band_range
    return normalized_band

# Bloque de Convolución
def double_conv_block(x, n_filters):
    # Conv2D then ReLU activation
    x = layers.Conv2D(n_filters, 3, padding="same", activation="relu",
                     kernel_initializer="he_normal")(x)
    # Conv2D then ReLU activation
    x = layers.Conv2D(n_filters, 3, padding="same", activation="relu",
                     kernel_initializer="he_normal")(x)

    return x

# Bloque de Reducción
def downsample_block(x, n_filters):
    f = double_conv_block(x, n_filters)
    p = layers.MaxPool2D(2)(f)
    p = layers.Dropout(0.3)(p)

    return f, p
```

```

# Bloque de Muestra Ascendente
def upsample_block(x, conv_features, n_filters):
    # upsample
    x = layers.Conv2DTranspose(n_filters, 3, 2, padding="same")(x)
    # concatenate
    x = layers.concatenate([x, conv_features])
    # dropout
    x = layers.Dropout(0.3)(x)
    # Conv2D twice with ReLU activation
    x = double_conv_block(x, n_filters)

    return x

# Construccion de la Red Neuronal en base al modelo U-Net
def build_unet_model():
    # inputs
    inputs = layers.Input(shape=(128, 128, 7))

    # encoder: contracting path - downsample
    # 1 - downsample
    f1, p1 = downsample_block(inputs, 64)
    # 2 - downsample
    f2, p2 = downsample_block(p1, 128)

    f3, p3 = downsample_block(p2, 256)

    # 5 - bottleneck
    bottleneck = double_conv_block(p3, 512)

    # decoder: expanding path - upsample

    u1 = upsample_block(bottleneck, f3, 256)
    u2 = upsample_block(u1, f2, 128)
    u3 = upsample_block(u2, f1, 64)

    # outputs
    outputs = layers.Conv2D(1, 1, padding="same", activation="sigmoid")(u3)

    # unet model with Keras Functional API
    unet_model = tf.keras.Model(inputs, outputs, name="U-Net")

    return unet_model

# 1) Obtencion de las fuentes de datos

datacube = xr.open_dataset("data/sta_maria/STAMARIA_S2_2023_cube.nc",
                           chunks=150)
datacube_zeeland =

```

```

xr.open_dataset("data/europe/zeeland_s2_mcclouds_marea_cube.nc", chunks=150)
mask_zeeland = rasterio.open('data/europe/europe_mask.tif')
mask_stmaria = rasterio.open('data/sta_maria/stamaria_mask.tif')
y_true_stamaria = mask_stmaria.read(1)
y_true_eu = mask_zeeland.read(1)
shapefile = gpd.read_file("data/sta_maria/fields.shp")
shapefile = shapefile.dropna(subset=['geometry'])

dates = datacube['time']
dates = dates[0:70]

dates_eu = datacube_zeeland['time']
dates_eu = dates_eu[0:5]

# Variables de apoyo

dc_shape = datacube["B01"].shape
lat_len = dc_shape[1]
lon_len = dc_shape[2]

dc_shape_eu = datacube_zeeland["B01"].shape
lat_len_eu = dc_shape_eu[1]
lon_len_eu = dc_shape_eu[2]

image_size = 128

datacube_subset_list = []
border_subset_list = []
all_boundaries_list = []

i = 0

# Creacion de nuevas bandas - Indices (Dependiendo de la zona a trabajar)
nir = datacube["B08"]
red = datacube["B04"]
blue = datacube["B02"]
green = datacube["B03"]

ndvi = (nir - red) / (nir + red)

ndwi = (green - nir) / (green + nir)

L = 0.5
savi = ((nir - red) / (nir + red + L)) * (1 + L)

datacube = datacube.assign(NDVI=ndvi)
datacube = datacube.assign(NDWI=ndwi)
datacube = datacube.assign(SAVI=savi)

```

```

cloud_mask = datacube['cloud_mask']

band_2 = datacube['B02']
mean_2 = np.nanmean(band_2)
masked_2 = np.where(np.isnan(cloud_mask), band_2, mean_2)
band_2.values = masked_2

band_3 = datacube['B03']
mean_3 = np.nanmean(band_3)
masked_3 = np.where(np.isnan(cloud_mask), band_3, mean_3)
band_3.values = masked_3

band_4 = datacube['B04']
mean_4 = np.nanmean(band_4)
masked_4 = np.where(np.isnan(cloud_mask), band_4, mean_4)
band_4.values = masked_4

band_8 = datacube['B08']
mean_8 = np.nanmean(band_8)
masked_8 = np.where(np.isnan(cloud_mask), band_8, mean_8)
band_8.values = masked_8

datacube['B02'] = band_2
datacube['B03'] = band_3
datacube['B04'] = band_4
datacube['B08'] = band_8

datacube = datacube.assign(ONES=lambda x: 1 * np.isnan((x.B02 * np.nan)))

# Obtención de los bordes de los campos
for date in dates:
    all_boundaries = np.full((datacube.dims['lat'], datacube.dims['lon']), False)
    for field_geo in shapefile.geometry:
        datacube_roi = datacube.sel(time=date).salem.roi(geometry=field_geo, other=0.0)

        o_fields_zero_one = datacube_roi['ONES']

        conv_1_kernel = np.ones((3, 3))
        big_field = convolve2d(o_fields_zero_one, conv_1_kernel, 'same')

        big_field_zero_one = (big_field > 0) * 1.0

        field_boundary = (big_field_zero_one - o_fields_zero_one) == 1

        all_boundaries = np.logical_or(all_boundaries, field_boundary)

    all_boundaries_list.append(all_boundaries)

# Nota: Los bordes generados anteriormente se almacenan en un archivo .npy para
# optimizar procesos

```

```

loaded_all_boundaries = np.load('all_boundaries.npy', allow_pickle=True)

# Creación de las imágenes de entrada con un tamaño de 128x128
var_to_drop = ["cloud_mask", "solar_zenith", "solar_azimuth", "view_zenith",
               "view_azimuth", "B01", "B05", "B06", "B07", "B09", "B8A", "B11", "B12"]
datacube = datacube.drop_vars(var_to_drop)
for date in dates:
    for lat in range(0, lat_len, image_size):
        lat_sup = lat + image_size
        if lat == image_size * (lat_len // image_size):
            lat_sup = lat_len
            continue
        for lon in range(0, lon_len, image_size):
            lon_sup = lon + image_size
            if lon == image_size * (lon_len // image_size):
                lon_sup = lon_len
                continue
            subset = datacube.sel(time=date).isel(lat=slice(lat, lat_sup), lon=slice(lon,
            lon_sup))
            datacube_subset_list.append(subset)
            border_subset = loaded_all_boundaries[0, lat:lat_sup, lon:lon_sup]
            border_subset_list.append(border_subset)

# Apilando bandas para crear la entrada de datos para la CNN
input_field_list = []

for datacube_piece in datacube_subset_list:
    datacube_piece["B02"] = datacube_piece["B02"].fillna(0)
    band_b02 = datacube_piece["B02"].values
    band_b02 = normalize_band(band_b02)

    datacube_piece["B03"] = datacube_piece["B03"].fillna(0)
    band_b03 = datacube_piece["B03"].values
    band_b03 = normalize_band(band_b03)

    datacube_piece["B04"] = datacube_piece["B04"].fillna(0)
    band_b04 = datacube_piece["B04"].values
    band_b04 = normalize_band(band_b04)

    datacube_piece["B08"] = datacube_piece["B08"].fillna(0)
    band_b08 = datacube_piece["B08"].values
    band_b08 = normalize_band(band_b08)

    datacube_piece["NDVI"] = datacube_piece["NDVI"].fillna(0)
    band_ndvi = datacube_piece["NDVI"].values
    band_ndvi = normalize_band(band_ndvi)

    datacube_piece["NDWI"] = datacube_piece["NDWI"].fillna(0)
    band_ndwi = datacube_piece["NDWI"].values

```

```

band_ndwi = normalize_band(band_ndwi)

datacube_piece["SAVI"] = datacube_piece["SAVI"].fillna(0)
band_savi = datacube_piece["SAVI"].values
band_savi = normalize_band(band_savi)

input_field = np.dstack((band_b02, band_b03, band_b04, band_b08, band_ndvi,
band_ndwi, band_savi))

input_field_list.append(input_field)

input_field_list = np.array(input_field_list)

# Creando respectiva lista de máscaras para entrenamiento
input_border_list = []

for border in border_subset_list:
    input_border_list.append(border)

input_border_list = np.array(input_border_list)

# Creando los sets de entrenamiento y prueba
train_images, test_images, train_masks, test_masks = train_test_split(input_field_list,
input_border_list, test_size=0.2, random_state=42)

# Construcción y compilación del modelo
model = build_unet_model()
model.compile(optimizer=tf.keras.optimizers.Adam(), loss="binary_crossentropy",
metrics="accuracy")

# Entrenamiento del modelo
history = model.fit(x=train_images, y=train_masks, validation_data=(test_images,
test_masks), epochs=10, batch_size=32)

# Generación de predicciones
predicted_masks = []
tests_images_lst = []
test_masks_lst = []

for i in range(100):
    image_test = test_images[i]
    mask_test = test_masks[i]
    image_test = np.expand_dims(image_test, axis=0)
    tests_images_lst.append(image_test)
    test_masks_lst.append(mask_test)

    predicted_mask = model.predict(image_test)
    predicted_mask = np.squeeze(predicted_mask)

    predicted_masks.append(predicted_mask)

```

```
tests_images_lst = np.array(tests_images_lst)
test_masks_lst = np.array(test_masks_lst)
predicted_masks = np.array(predicted_masks)

np.save('images_zelanda.npy', tests_images_lst)
np.save('predicciones_zelanda.npy', predicted_masks)
np.save('mascaras_zelanda.npy', test_masks_lst)
```

## **ANEXO G**

Paquetes utilizados para el desarrollo de la implementación de la red neuronal convolucional

Salem 0.3.7

Numpy 1.23.5

Geopandas 0.9.0

Xarray 0.17.0

Matplotlib 3.7.1

Rasterio 1.3.6

Scipy 1.9.1

Tensorflow 2.12.0

scikit-image 0.20.0

scikit-learn 1.2.2

netCDF4 1.5.8

Pillow 9.5.0

Paquetes utilizados para el desarrollo de la implementación del algoritmo de Canny

scikit-image 0.20.0

netCDF4 1.6.3

matplotlib 3.7.1

opencv-python 4.7.0.72