



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA

# **Detección de monocultivos de palma de aceite en Colombia implementando un modelo de redes neuronales convolucionales a partir de imágenes satelitales**

**John David Carvajal Rivera**

Universidad Nacional de Colombia

Facultad de Ingeniería, Departamento de Ingeniería de Sistemas e Industrial

Bogotá, Colombia

2023

# **Detección de monocultivos de palma de aceite en Colombia implementando un modelo de redes neuronales convolucionales a partir de imágenes satelitales**

**John David Carvajal Rivera**

Trabajo de grado presentado como requisito parcial para optar al título de:  
**Magíster en Ingeniería de Sistemas y Computación**

Director:

Ph. D. Luis Fernando Niño Vásquez

Codirector:

Ph. D. Juan David Gutiérrez Torres

Línea de Investigación:

Sistemas Inteligentes

Grupo de Investigación:

Laboratorio de investigación en sistemas inteligentes - **LISI**

Universidad Nacional de Colombia

Facultad de Ingeniería, Departamento de Ingeniería de Sistemas e Industrial

Bogotá, Colombia

2023

*La tecnología es como una espada de doble filo. Esas herramientas pueden ayudarnos a salvar la tierra o destruirla.*

*David Suzuki*

## Declaración de obra original

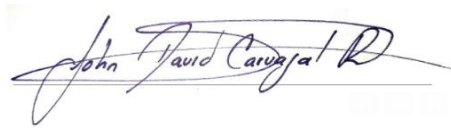
Yo declaro lo siguiente:

He leído el Acuerdo 035 de 2003 del Consejo Académico de la Universidad Nacional. «Reglamento sobre propiedad intelectual» y la Normatividad Nacional relacionada al respeto de los derechos de autor. Esta disertación representa mi trabajo original, excepto donde he reconocido las ideas, las palabras, o materiales de otros autores.

Cuando se han presentado ideas o palabras de otros autores en esta disertación, he realizado su respectivo reconocimiento aplicando correctamente los esquemas de citas y referencias bibliográficas en el estilo requerido.

He obtenido el permiso del autor o editor para incluir cualquier material con derechos de autor (por ejemplo, tablas, figuras, instrumentos de encuesta o grandes porciones de texto).

Por último, he sometido esta disertación a la herramienta de integridad académica, definida por la universidad.

A handwritten signature in black ink that reads "John David Carvajal R". The signature is written in a cursive style and is positioned above a horizontal line.

John David Carvajal Rivera

Fecha 1/08/2023

## • **Agradecimientos**

Quisiera expresar mi más sincero agradecimiento a todas las personas que han sido parte fundamental en mi camino hacia la culminación de esta tesis de maestría. En primer lugar, quiero agradecer a los profesores Luis Fernando Niño y Juan David Gutiérrez, cuya guía y conocimientos han sido imprescindibles para el desarrollo de este trabajo. Su dedicación, paciencia y valiosas sugerencias han contribuido de manera significativa a mi crecimiento académico y profesional.

No puedo dejar de mencionar a mis compañeros de maestría, en especial a los del grupo de investigación LISI, quienes han compartido conmigo este desafiante viaje. Su apoyo, colaboración y amistad han sido invaluable. Juntos hemos construido una red de aprendizaje y compañerismo que siempre recordaré con gratitud.

Quiero extender un agradecimiento especial a mis tres sobrinos, quienes se encuentran en Canadá. A pesar de la distancia, su amor y entusiasmo han sido un motor importante para seguir adelante. Su energía y alegría han iluminado mi camino y me han recordado la importancia de luchar por mis metas. A mi amada esposa, gracias por tu amor incondicional, comprensión y constante apoyo durante este exigente proceso. Tus palabras de aliento, tu paciencia y tu confianza en mí han sido un pilar fundamental.

Por último, pero no menos importante, quiero expresar mi profundo agradecimiento a mis padres. Su amor, sacrificio y dedicación han sido la base de mi educación y crecimiento. Gracias por creer en mí, por brindarme las herramientas necesarias para enfrentar cualquier desafío y por ser mi apoyo incondicional en cada paso que he dado.

# Resumen

## **Detección de monocultivos de palma de aceite en Colombia implementando un modelo de redes neuronales convolucionales a partir de imágenes satelitales**

Colombia es un país privilegiado para el desarrollo de la agricultura por su ubicación geográfica y diferentes pisos térmicos, los cuales permiten que se realicen variedad de cultivos de diferentes productos. Entre los más relevantes se encuentra la palma de aceite, donde Colombia ocupa el cuarto lugar como productor mundial, y el primero en América.

Sin embargo, los monocultivos, que son grandes extensiones de tierra dedicada al cultivo de una sola especie como lo es la palma de aceite traen consecuencias irreparables para el ecosistema como: la deforestación que pone en peligro la conservación del agua, la eliminación de la flora y la fauna, la propagación de inundaciones como el agravamiento de sequías en el territorio y zonas adyacentes, la erosión acelerada del suelo, la contaminación del agua por el uso de fertilizantes y la aparición de plagas por la ruptura del equilibrio ecológico [1], además de impactos socioambientales para la población rural [2], entre otros. Por lo tanto, es importante contar con herramientas que permitan la detección de áreas de monocultivo de palma de aceite.

En este contexto, el uso de imágenes satelitales se presenta como una herramienta eficaz para la detección de monocultivos de palma de aceite a gran escala. Además, el uso de modelos de redes neuronales convolucionales se ha demostrado como una técnica muy efectiva para el procesamiento de estas imágenes.

El propósito de este trabajo de investigación es implementar un modelo de redes neuronales convolucionales para la detección de monocultivos de palma de aceite en Colombia a partir de imágenes satelitales. Se obtuvieron resultados muy promisorios frente a lo esperado, incluso en imágenes que eran externas al entrenamiento del modelo que permitió una detección precisa y eficiente de áreas de monocultivo de palma de aceite en regiones de Colombia. Esto puede contribuir a la toma de decisiones en materia de uso y a la conservación del medio ambiente.

**Palabras clave:** Redes neuronales convolucionales, aprendizaje profundo, imágenes satelitales, transferencia de aprendizaje.

## **Abstract**

### **Detection of oil palm monocultures in Colombia by implementing a convolutional neural network model from satellite image**

Palm oil is one of the most important crops in Colombia, with Colombia being the fourth largest producer of palm oil in the world. However, the monoculture of palm oil can have serious environmental consequences, such as deforestation and loss of biodiversity. Therefore, it is important to have tools that allow the detection of areas of palm oil monoculture.

In this context, the use of satellite images is presented as an effective tool for the detection of palm oil monoculture on a large scale. In addition, the use of convolutional neural network models has been demonstrated as a very effective technique for image processing.

In this research work, we propose to implement a convolutional neural network model for the detection of palm oil monocultures in Colombia using satellite images. It is expected that the implementation of this model will allow for precise and efficient detection of areas of palm oil monoculture, which can contribute to decision-making on land use and environmental conservation.

**Keywords:** Convolutional neural networks (CNN), deep learning, satellite imagery, transfer Learning.

Este Trabajo Final de maestría fue calificado en octubre de 2023 por el siguiente evaluador:

Jean Pierre Charalambos Hernández Msc.  
Profesor Facultad de Ingeniería  
Universidad Nacional de Colombia



# Contenido

	Pág.
● Agradecimientos.....	5
● Lista de figuras.....	12
● Lista de tablas .....	14
● Siglas .....	15
<b>1. Introducción.....</b>	<b>1</b>
<b>2. Objetivos.....</b>	<b>3</b>
2.1.1 Objetivo General .....	3
2.1.2 Objetivos Específicos.....	3
<b>3. Marco teórico.....</b>	<b>4</b>
3.1 Justificación y Motivación.....	9
3.2 Revisión de Conceptos.....	11
3.2.1 Inteligencia Artificial (AI) .....	11
3.2.2 Red Neuronal Artificial .....	11
3.2.3 Redes Neuronales Convolucionales ( <i>Convolutional Neural Network</i> )...12	
3.2.4 Aprendizaje Profundo ( <i>Deep Learning, DL</i> ) vs Aprendizaje Automático ( <i>Machine Learning, ML</i> ).....	14
3.2.5 Aprendizaje por transferencia ( <i>Transfer Learning</i> ).....	16
3.2.6 ResNet50 .....	18
3.2.7 <i>Cross-validation</i> o Validación cruzada .....	19
3.2.8 Google Earth.....	20
3.2.9 Google Earth Engine.....	21
3.2.10 Sentinel 1 y 2.....	24
3.2.11 Proceso de Teledetección Satelital .....	25
<b>4. Metodología.....</b>	<b>27</b>
4.1 Comprensión del negocio .....	30

4.2	Preparación de los datos.....	33
4.2.1	Ubicación de las zonas para el <i>DataSet</i> .....	33
4.2.2	Imágenes satelitales base.....	38
4.2.3	Conjunto de imágenes para entrenamiento y prueba.....	41
4.2.3.1	División de las imágenes satelitales base.....	41
4.2.3.2	Limpieza del <i>Dataset</i> .....	42
4.2.3.3	Enriquecimiento del <i>DataSet</i> .....	44
4.2.4	<i>Dataset</i> Final.....	45
4.3	Preparación de Entorno de trabajo.....	47
4.3.1	Ambiente Local.....	47
4.3.2	Ambiente en la Nube.....	48
4.4	Entrenamiento del modelo.....	50
4.4.1	Preparación de los datos para el entrenamiento.....	50
4.4.2	Entrenamiento del modelo de CNN ResNet50.....	51
4.5	Evaluación del modelo.....	53
4.5.1	Gráficas de desempeño modelo inicial.....	54
4.5.2	Matriz de confusión de los datos de prueba para el modelo inicial.....	56
4.5.3	Entrenamiento implementando <i>Cross-Validation</i> .....	58
4.5.4	Selección de un modelo proporcionado por el ( <i>cross-validation</i> ).....	60
<b>5.</b>	<b>Resultados.....</b>	<b>62</b>
5.1	Conjunto de imágenes externas de pruebas.....	62
5.2	Evaluación del modelo en las imágenes externas.....	64
<b>6.</b>	<b>Conclusiones.....</b>	<b>68</b>
6.1	Conclusiones.....	68
6.2	Recomendaciones.....	69
6.3	Futuros trabajos.....	70
•	<b>Anexo: Tabla de productos generados.....</b>	<b>72</b>
	<b>Bibliografía.....</b>	<b>75</b>



## • Lista de figuras

	Pág.
<b>Figura 1:</b> Producción nacional de aceite de palma de aceite de 2009 -2019, tomado de [10].....	5
<b>Figura 2:</b> Modelo de red neuronal biológica, tomado de [30] .....	12
<b>Figura 3:</b> Arquitectura de la red neuronal convolucional LeNet, tomado de [34] .	13
<b>Figura 4:</b> CNN con tres diferentes secciones de extracción de características, tomado de [36].....	14
<b>Figura 5:</b> Posición del Aprendizaje Profundo ( <i>Deep Learning</i> ) comparado con Aprendizaje automático ( <i>Machine Learning</i> ) y la inteligencia artificial (IA), tomado de [39] .....	15
<b>Figura 6:</b> Estructura general del aprendizaje por transferencia, donde el conocimiento del modelo previamente entrenado se transfiere al nuevo modelo DL, tomado de [39] .....	17
<b>Figura 7:</b> Estructura de la red neuronal ResNet50, tomado de [43].....	18
<b>Figura 8:</b> Proceso de fusión de características con información de contexto, tomado de [43].....	19
<b>Figura 9</b> <i>Cross-Validation</i> o validación cruzada aleatoria con k iteraciones .....	20
<b>Figura 10:</b> Ejemplo de expansión de una mina de carbón a cielo abierto y el talado de árboles que tuvo lugar cerca de Elkford (British Columbia, Canadá) de 1984-2011, tomado de [49] .....	23
<b>Figura 11:</b> Infografía de los satélites Sentinel 1 y Sentinel 2, tomado de [54] .....	25
<b>Figura 12:</b> Proceso de teledetección e iteración con la radiación incidente y los objetos de interés. Grupo de investigación GOTA, Universidad de Laguna, tomado de [57].....	26
<b>Figura 13:</b> Fases de la metodología SEMMA, tomado de [58].....	27
<b>Figura 14</b> Dinámica de la metodología SEMMA, tomado de [58] .....	28
<b>Figura 15:</b> Estructura de carpetas del <i>DataSet</i> compartida por la plataforma Zenodo.....	31
<b>Figura 16</b> Estructura de carpetas de GitHub.....	33
<b>Figura 17:</b> Localización geográfica de cultivos y plantas de beneficio zona norte de Colombia año 2017, tomado de [60] .....	34
<b>Figura 18:</b> Zona norte de Colombia con presencia de cultivos de palma de aceite. ....	35
<b>Figura 19:</b> Ubicación de cultivos de palma de aceite en Colombia .....	36

<b>Figura 20:</b> Zonas adyacentes a cultivos de palma de aceite.....	38
<b>Figura 21:</b> Área de cultivos de palma de aceite, región Aguas Dulces Cesar.....	40
<b>Figura 22</b> Imágenes exportadas de los satélites Sentinel 1 y Sentinel 2 por medio del <i>script</i> [56] de Google Earth Engine, región Aguas Dulces Cesar.....	40
<b>Figura 23:</b> Representación gráfica de la división de una imagen satelital base, región Bella Cruz Cesar.....	42
<b>Figura 24:</b> Ejemplo de limpieza de imágenes que no tienen presencia de cultivos de palma de aceite, región Bella Cruz Cesar.....	43
<b>Figura 25:</b> Ejemplos de algunas imágenes extraídas del satélite Sentinel 1 con presencia de ruido. ....	43
<b>Figura 26:</b> Ejemplo de las técnicas aplicadas a una imagen para el enriquecimiento del <i>DataSet</i> . ....	45
<b>Figura 27:</b> Grupos de clasificación del conjunto definitivo de los datos de forma aleatoria tomado de [62]. ....	46
<b>Figura 28:</b> Flujo de entrenamiento de modelo de reconocimiento de cultivos de palma de aceite. ....	53
<b>Figura 29</b> Gráfica de desempeño: <i>Training and Validation accuracy</i> a través de las épocas.....	55
<b>Figura 30</b> Gráfica de desempeño: <i>Training and validation loss</i> a través de las épocas. ....	56
<b>Figura 31</b> Matriz de confusión del conjunto de datos de prueba. ....	57
<b>Figura 32</b> Gráfica de desempeño de <i>accuracy</i> con los datos de entrenamiento y validación de cada uno de los conjuntos de datos obtenidos de <i>cross-validation</i> .59	59
<b>Figura 33</b> Gráfica de desempeño de la pérdida ( <i>loss</i> ) con los datos de entrenamiento y validación de cada uno de los conjuntos de datos obtenidos de <i>cross-validation</i> .....	60
<b>Figura 34</b> Matriz de confusión del modelo seleccionado.....	61
<b>Figura 35</b> Cultivos de palma de aceite cerca del municipio de Chigorodó, Chocó. ....	63
<b>Figura 36</b> Cultivos de palma de aceite en Nariño.....	63
<b>Figura 37</b> Clasificación de datos externos para los departamentos de Nariño y Chocó, con los satélites Sentinel-1 y Sentinel-2.....	64
<b>Figura 38</b> Evaluación de clasificadores binarios del modelo seleccionados con las imágenes externas.....	65

## • Lista de tablas

	<b>Pág.</b>
<b>Tabla 1:</b> Comparación de rendimiento de CNN con otras técnicas en el área de Agricultura. Tabla adaptada de [38].....	8
<b>Tabla 2:</b> Resultados de la CNNs y otras técnicas de clasificación de imágenes satelitales. Imagen adaptada del artículo [22].....	9
<b>Tabla 3:</b> Principales elementos de GEE .....	24
<b>Tabla 4:</b> Fases de la metodología que se van a llevaron a cabo para el desarrollo de la investigación.....	29
<b>Tabla 5:</b> Ubicaciones de cultivos de palma de aceite en Colombia. Información extraída del KML Google Earth.....	37
<b>Tabla 6</b> Clasificación del <i>Dataset</i> .....	47
<b>Tabla 7:</b> Características del equipo de trabajo para el ambiente local.....	48
<b>Tabla 8:</b> Características de entorno en la Nube Google Colab .....	49
<b>Tabla 9:</b> Resultados de predicción del modelo seleccionado con los datos externos clasificados por departamento y satélite.....	65

## . Siglas

---

<b>Abreviatura</b>	<b>Término</b>
<i>IA</i>	Artificial Intelligence
<i>ANN</i>	Artificial Neural Networks
<i>CNN</i>	Convolutional Neural Network
<i>DL</i>	Deep Learning
<i>ML</i>	Machine Learning

---

# 1. Introducción

Hoy en día los países en desarrollo se enfrentan a un gran problema que tiene grandes consecuencias para los ecosistemas; los monocultivos, los cuales son grandes extensiones de tierra dedicada al cultivo de una sola especie. Colombia se ubica a nivel mundial en el cuarto puesto como productor de cultivos de palma de aceite, y en el primer lugar en América [1] Debido a su versatilidad y resiliencia este producto se ha convertido en el principal componente en una gran cantidad de productos diversos que usamos a diario, por lo cual en los últimos años la demanda de este producto ha aumentado considerablemente con la producción anual, cuadruplicándose entre 1995 y 2015, y se espera que se cuadruple nuevamente para el 2050 [3].

Ante esta preocupación de crecimiento de este y otros monocultivos, los países en desarrollo han creado leyes para su regulación y mitigar su expansión, donde uno de los factores claves es la detección de estos en zonas rurales para realizar un monitoreo y auditoría por parte de los entes de control. En opinión de este autor los métodos para la detección de monocultivos actualmente son muy manuales y poco eficientes al momento de detectar estos en zonas rurales apartadas de nuestro país y al identificar el tipo de especie como lo es la palma de aceite.

Actualmente los avances de Teledetección han acelerado la disponibilidad de datos de imágenes satelitales [2]. El uso de estas tecnologías puede ayudar a mitigar y regular la expansión de monocultivos de palma de aceite en Colombia, ya que por medio del análisis del aprendizaje automático de estas imágenes se puede desarrollar métodos para mapear zonas que cumplan con patrones. Estos se pueden implementar en zonas rurales de nuestro país que sean de difícil acceso, con el objetivo de detectar monocultivos de palma de aceite donde los métodos tradicionales no lo han logrado para que los profesionales del área agrícola-ambiental puedan realizar las respectivas inspecciones y auditorías forestales que demandan los entes de control.

Este trabajo propone como una alternativa de solución la implementación de métodos computacionales que permitan extraer patrones de imágenes satelitales de alta resolución [3], y desde un enfoque de aprendizaje profundo lograr un valor predictivo positivo [4] para identificar monocultivos de palma de aceite implementando redes neuronales convolucionales (CNN), las cuales se han



convertido en uno de los enfoques de aprendizaje automático preferidos para el análisis de imágenes debido a su capacidad de extraer patrones de representaciones de datos tanto de baja como de alta resolución.

A continuación, se encuentra una descripción de cada una de las secciones de este trabajo:

En el capítulo 2 se encuentran los objetivos tanto general como específicos que se esperan alcanzar con el desarrollo de este trabajo.

En el capítulo 3 se encuentra la justificación y motivación para el desarrollo de este trabajo, también se encuentra la revisión de conceptos relacionados con el mismo.

En el capítulo 4, se describe la metodología utilizada para el desarrollo del proyecto. Se explican los pasos que se siguieron desde la comprensión del negocio, la preparación de los datos de entrenamiento y la evaluación del modelo. Se detallan las fuentes, los criterios y las técnicas empleadas para obtener, limpiar, dividir y enriquecer el conjunto de datos de imágenes satelitales base. Se describen las características y los requisitos del ambiente local y del ambiente en la Nube que se utilizaron para el entrenamiento del modelo. Se muestran los resultados del entrenamiento.

En el capítulo 5, se presentan los resultados del modelo ya entrenado con un set de datos de prueba externos.

En el capítulo 6, se encuentran las conclusiones, recomendaciones y propuesta de trabajos futuros.

## **2. Objetivos**

### **2.1.1 Objetivo General**

Detectar monocultivos de palma de aceite sobre imágenes satelitales de dominio público extraídas de zonas rurales de nuestro país, implementando un modelo de CNN como un apoyo en el proceso de detección tradicional.

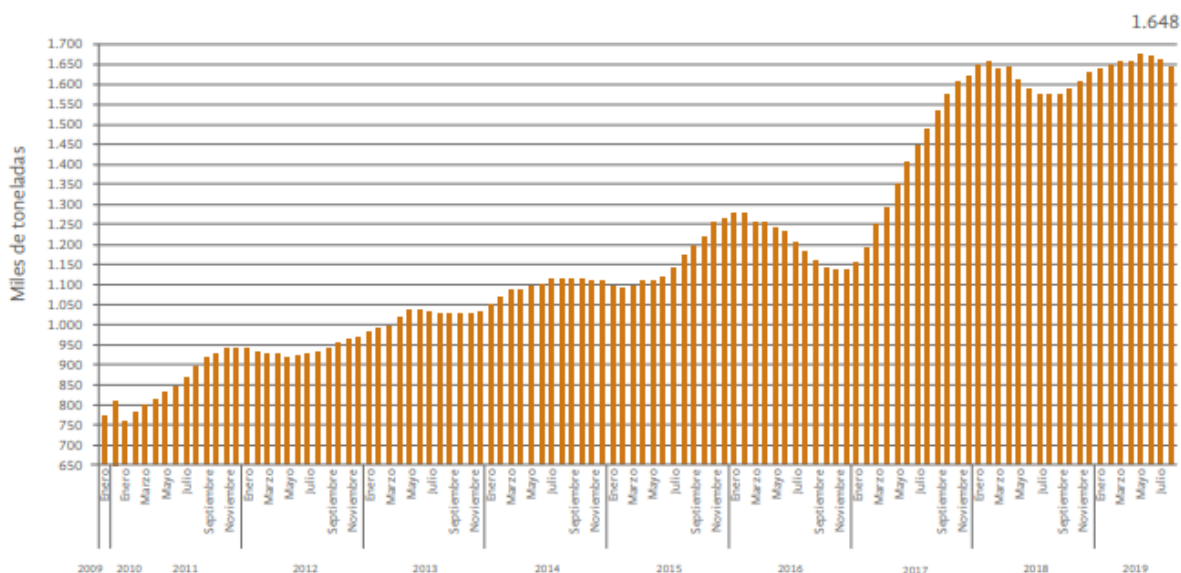
### **2.1.2 Objetivos Específicos**

- OE1: Recolectar un conjunto de imágenes satelitales de zonas rurales que contengan monocultivos de palma de aceite y que sean de dominio público para el desarrollo del proyecto.
- OE2: Implementar un modelo de CNN para el reconocimiento de monocultivos de palma de aceite sobre las imágenes satelitales.
- OE3: Evaluar el desempeño obtenido por el modelo de CNN implementado sobre el conjunto de imágenes satelitales recolectadas.

### **3.Marco teórico**

Aunque Colombia es un país privilegiado para el desarrollo de la agricultura por su ubicación geográfica, variedad cultural, climas diversos, flora, fauna, cuencas hidrográficas y recursos naturales, este sector económico está en riesgo por la expansión de monocultivos, los cuales traen consecuencias irreparables para el ecosistema como: la deforestación que pone en peligro la conservación del agua [5], la eliminación de la flora y fauna que tiene graves consecuencias para el medio ambiente, la propagación de inundaciones, el agravamiento de sequías, la erosión acelerada del suelo, la contaminación del agua por el uso de fertilizantes y la aparición de plagas debido a la ruptura del equilibrio ecológico [6]. Además, la ocupación de grandes extensiones de tierra tiene impactos socio-ambientales negativos en la población rural, ya que puede resultar en el desplazamiento de comunidades indígenas y campesinas que dependen de la tierra para su subsistencia [7]. Esto puede llevar a la pobreza y la informalidad en estas comunidades [8].

La palma de aceite es un producto bastante atractivo para la economía debido a su versatilidad ya que permite múltiples usos en las industrias de alimentos, oleoquímica energías renovables y producción derivados de la biomasa amigables con el medio ambiente como: cosméticos, jabones, detergentes, tintas de impresión, etc. [9]. En la Figura 1 se puede observar una curva ascendente correspondiente a la producción nacional solo de aceite de palma de aceite en los últimos años.



**Figura 1:** Producción nacional de aceite de palma de aceite de 2009 -2019, tomado de [10]

En la publicación de Fedepalma [9], [10] se dio a conocer que actualmente el cultivo de aceite de palma se encuentra en 161 municipios de 21 departamentos de Colombia, que es el cuarto productor de aceite de palma en el mundo y el primero en América, cifras que evidencian la gran preocupación de la expansión de este monocultivo en nuestro país.

Ante la situación de expansión de todo tipo de monocultivos incluyendo la palma de aceite, particularmente el Congreso Nacional de Colombia expidió en el año 2006 la Ley 1021 o Ley General Forestal, en donde uno de los puntos es la regulación a nivel nacional de los monocultivos en general [11], donde se hizo referencia a la detección de monocultivos por medio de encuestas realizadas por los entes departamentales e inspecciones forestales. Esto implica un desplazamiento de profesionales del área agrícola-ambiental a zonas rurales de Colombia para realizar una auditoría directa en los monocultivos.

Por otro lado, en el contexto de una reforma agraria que ha experimentado modificaciones a lo largo de muchos años bajo la presión de la CONTAG<sup>1</sup>, y en respuesta a la creciente preocupación por la deforestación en Brasil, un país con

<sup>1</sup> Confederação Nacional dos Trabalhadores na Agricultura.

una alta concentración de tierras agrícolas, se están llevando a cabo métodos de detección de monocultivos de diversos productos locales a través de censos agropecuarios. Al mismo tiempo, se están implementando medidas de control sobre los propietarios de grandes extensiones de tierra que se dedican a los monocultivos [12].

Desde la perspectiva del autor, validar los procesos que se utilizan para cumplir las leyes establecidas en los países en desarrollo revela que los métodos actuales para detectar monocultivos en zonas rurales remotas son manuales e ineficientes. Para mejorar esta situación, es importante aprovechar las herramientas disponibles, como las imágenes satelitales, para describir y estudiar los fenómenos terrestres, incluyendo los monocultivos de palma de aceite en zonas rurales de difícil acceso. Al hacerlo se puede mitigar y regular la expansión de estos monocultivos en áreas donde los métodos tradicionales no son efectivos, lo que permitiría a los profesionales del área agrícola-ambiental llevar a cabo las inspecciones y auditorías forestales requeridas por los entes de control. En resumen, para mejorar la detección de monocultivos y cumplir con las leyes establecidas, se recomienda utilizar las mejores prácticas disponibles, como el uso de imágenes satelitales y la tecnología de teledetección para la mitigación y regulación de la expansión de monocultivos en zonas rurales remotas.

Para esta implementación se propone el uso de redes neuronales convolucionales (CNN) que se han convertido en uno de los enfoques de aprendizaje automático más populares para el análisis de imágenes debido a su capacidad para extraer patrones de representaciones de datos de baja resolución en las llamadas capas convolucionales<sup>2</sup> [13], además las CNN son adecuadas para el procesamiento de datos espaciales y temporales [14], donde tienen la capacidad de reconocer representaciones características que se obtienen mediante el filtrado de los bordes.

Se han realizado varios estudios sobre la implementación de CNN en imágenes satelitales y agricultura durante los últimos años, obteniendo resultados óptimos y relevantes. A continuación, resalto algunas investigaciones:

En 2012, Kapach et al. introdujeron el término “agrovisión” para referirse a la visión por computadora en la agricultura [15]. En 2013, Mnih propuso una arquitectura específica de CNN para el etiquetado de imágenes aéreas [16]. En 2014, Payne y Walsh señalaron que la clasificación e identificación de imágenes satelitales de cultivos es desafiante debido a las variaciones en las condiciones de iluminación y otros factores [17].

En 2017, Suchet B. y James U. compararon una MLP y una CNN para detectar frutas en huertos de manzanas utilizando imágenes de alta resolución [18]. Ambas

---

<sup>2</sup> Tienen la capacidad de tomar grupos **de píxeles cercanos** de la imagen e ir operando matemáticamente (producto escalar) contra una matriz denominada kernel.

redes calcularon la probabilidad de que un píxel sea una fruta o no. La CNN tuvo una mayor confianza en la detección de frutas con una probabilidad de detección del 92%, en comparación con el 83% para MLP. Los autores sugieren que, en lugar de etiquetar manualmente grandes cantidades de datos, se deben tomar las imágenes más representativas y realizar una segmentación para luego etiquetar cada segmento.

En 2017, Maggiori E. et al. propusieron un marco de trabajo para la clasificación densa y por píxeles de imágenes satelitales con CNN [19]. Sugieren tener en cuenta tres aspectos para convertir una CNN en una herramienta de clasificación de imágenes satelitales: 1) Diseñar una arquitectura CNN que genere una clasificación por píxel en lugar de solo una categoría para toda la entrada. 2) Adquirir datos de entrenamiento a gran escala y manejar sus eventuales inexactitudes. 3) Producir una salida de alta resolución a pesar de la pérdida de resolución inherente al uso de CNN.

En 2017, Maggiori E. et al. propusieron un marco de trabajo para la clasificación densa y por píxeles de imágenes satelitales con CNN [20]. Sugieren tener en cuenta tres aspectos para convertir una CNN en una herramienta de clasificación de imágenes satelitales: 1) diseñar una arquitectura CNN que genere una clasificación por píxel en lugar de solo una categoría para toda la entrada; 2) adquirir datos de entrenamiento a gran escala y manejar sus eventuales inexactitudes; y 3) producir una salida de alta resolución a pesar de la pérdida de resolución inherente al uso de CNN.

En el 2018, Kamilaris A, realiza una relevante encuesta para nuestra investigación del uso de CNN en el dominio de la agricultura [20] , donde los hallazgos generales indican que las CNN constituyen una técnica prometedora con un alto rendimiento en términos de precisión y exactitud de clasificación comparado con otras técnicas como: *random forest* (RF), *Support vector regression* (SVR), *area base technique* (ABT), *best texture-based regression model*, entre otros. Cabe recalcar el énfasis que se hace en el éxito del modelo de una CNN empleado depende en gran medida de la calidad del conjunto de datos.


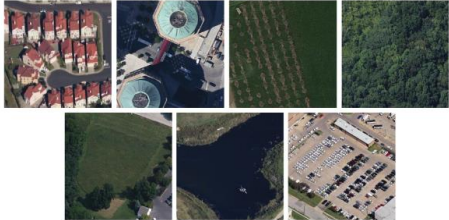

Aunque el estudio anterior es relativamente actual y no abarca una gran cantidad de artículos, en la mayoría de los desafíos agrícolas utilizados se ha observado una precisión satisfactoria, especialmente en comparación con otras técnicas empleadas para resolver el mismo problema. Esto indica una aplicación exitosa de CNN en varios dominios agrícolas. En particular, las áreas de detección de enfermedades de plantas y hojas [21], reconocimiento de plantas, clasificación de cobertura terrestre, lo cual nos da una orientación para nuestro campo frente a los monocultivos, conteo de frutos e identificación de malezas pertenecen a las categorías donde se ha observado la mayor precisión. En la tabla 3 se realizó una adaptación de la información presentada por Kamilaris, haciendo énfasis sólo en

los problemas que tienen que ver con segmentación de suelo, clasificación de cobertura y detección de algún tipo sobre áreas agrícolas.

Área de Agricultura	Descripción del problema	Datos implementados	Precisión	DL Modelo usado	Framework usado	Comparación con otras técnicas	Referencia
Detección de enfermedades en las plantas de un cultivo	Identificar 14 especies de cultivos y 26 enfermedades	PlantVillage conjunto de datos públicos de 54306 imágenes de hojas de plantas enfermas y	0,9935 (F1)	AlexNet, GoogleNet	Caffe	Margen sustancial en puntos de referencia estándar con enfoques que utilizan características	Mohanty et al. (2016)
	Clasificación de enfermedades en las hojas de un cultivo de banano	Conjunto de datos de 3700 imágenes de enfermedades del banano obtenidas del conjunto de datos PlantVillage	96 % + (CA), 0,968 (F1)	LeNet	Deeplearning4j	Los métodos que utilizan características hechas a mano no se generalizan bien	Amara et al. (2017)
Clasificación de cobertura terrestre	Identificar 13 clases diferentes de cobertura del suelo en KSC y nueve clases diferentes en Pavía, Italia	Un sitio de vegetación mixta sobre el Centro Espacial Kennedy (KSC), FL, EE. UU., y un sitio urbano sobre la ciudad de Pavía, Italia	98,00 % (CA)	Hybrid of PCAM Autoencoder and logistic regression	Desarrollado por los autores	1% más precisa que RBF -SVM	Chen et al(2014)
	Identificar 21 clases de uso de la tierra que contienen una variedad de patrones espaciales	Conjunto de datos de uso de la tierra de UC Merced	93,48 % (CA)	Author-defined	Theano	Aprendizaje de funciones no supervisado (UFL) (82-90% y SIFT (85%))	luus et al (2015)
	Extraer información sobre tierras cultivadas	Imágenes de UAV en las áreas Pengzhuo Country y Guanghan Country, Sichuan Province, China	88-91%(CA)	Author-defined	N/A	N/A	Lu et al (2017)
Segmentación de raíz y suelo	Identificar raíces de suelos	Imágenes de suelo procedentes de tomografía de rayos X	Quality measure QM= 0,23 (simulation) QM = 0,57 (real roots)	Author-defined CNN with SVM for classification	MatConvNet	N/A	Douarre et al. (2016)
Identificación de malezas en cultivos	Clasifica 91 tipos de malezas	Conjunto de datos de 3980 imágenes que contienen 91 tipos de malezas	90,96 % (CA)	PCANet+LMC cassifiers	Desarrollado por los autores	Mejores resultados que las técnicas de extracción de características (sin detalles)	Xinshao and Cheng(2015)
	Clasificación de especies de malezas en cultivos (22 en total)	Conjunto de datos de 10.413 imágenes, tomadas principalmente de BBCH 12-16 que contiene 22 especies de malezas y cultivos en la etapa de crecimiento inicial	86,20 % (CA)	Variation of VGG16	Theano-based Lasagne library for Phyton	Características locales de forma y color (42,5% y 12,2%, respectivamente)	Dyrmann et al (2016)
	Identificar imágenes de trigo en invierno , y cebada en primavera	Un total de 4500 imágenes de 10, 20, 30 y 50 m de altitud capturadas por una cámara Canon PowerShot G15	97,00% (CA)	DenseNet	Caffe	Herramienta de cardo basada en características de color (95%)	Sorensen et al. (2017)

**Tabla 1:** Comparación de rendimiento de CNN con otras técnicas en el área de Agricultura. Tabla adaptada de [38]

A pesar de que en la Tabla 1 se presenta una lista de CNNs y otras técnicas en diferentes investigaciones, no se hace un énfasis en un conjunto de datos (*dataset*) con imágenes satelitales como lo requiere esta investigación. Por lo tanto, para la selección de la CNN que se va a implementar en el conjunto de datos de esta investigación, se tomó en cuenta la investigación que realizaron Shabbir et al. [22], quienes tomaron 6 diferentes tipos de *datasets*, 3 de ellos basados en imágenes satelitales como nuestra investigación, luego realizaron la implementación y comparación de resultados de diferentes CNNs y otras técnicas teniendo como objetivo una clasificación de manera efectiva y eficiente del conjunto de datos. En la Tabla 2 se presenta una adaptación de la comparación obtenida en los conjuntos de datos que tienen imágenes satelitales.

Dataset	Name of Algorithm/model	Classification Accuracy (%)	Sample Images
SIRI-WHU	POVH	80.14%	
	VGGNet	86.6%	
	Inception-V3	89%	
	GoogLeNet	89.3%	
	AlexNet	90.2%	
	<b>ResNet50</b>	<b>94.03%</b>	
RSSCN	RGSIR	81.44%	
	POVH	84.07%	
	AlexNet	85.6%	
	GoogLeNet	85.84%	
	Inception-V3	87%	
	VGG-VD-16	87.18%	
	CaffeNet	88.25%	
	<b>ResNet50</b>	<b>92%</b>	
UCM	Inception-V3	91.1%	
	Feature RCG SVM	93.81%	
	AlexNet	94.2%	
	GoogLeNet	94.31%	
	CaffeNet	95.02%	
	VGG-VD-16	95.21%	
	<b>ResNet50</b>	<b>97.78%</b>	

**Tabla 2:** Resultados de la CNNs y otras técnicas de clasificación de imágenes satelitales. Imagen adaptada del artículo [22]

Como se puede observar en la Tabla 2, teniendo la implementación de diferentes arquitecturas de CNN como lo son: AlexNet, VGG, ResNet y GoogLeNet, los mejores resultados, con mejor *accuracy*, en los diferentes *datasets* de imágenes satelitales lo tiene la CNN ResNet50 la cual es una versión de ResNet con 50 capas que maneja el problema de la degradación de la precisión de clasificación al aumentar el número de capas. Esta red se obtuvo ajustando su arquitectura y los hiperparámetros. Así pues, se obtuvo un modelo que puede clasificar imágenes satelitales de manera más efectiva y eficiente en comparación a otros modelos. Por tal motivo en esta investigación, se usa este modelo para clasificar el conjunto de datos de imágenes satelitales de palma de aceite en Colombia.

### 3.1 Justificación y Motivación

Colombia ha tenido antecedentes que evidencian las grandes consecuencias que pueden traer la implementación de monocultivos, como el deterioro de ecosistemas en departamentos [23], el desplazamiento en zonas rurales, sequías y muerte de fauna en regiones. Específicamente, los monocultivos de palma de aceite han sido



fuertemente cuestionados por su sostenibilidad, por deforestar y comprometer la vida silvestre de ciertas especies en los territorios donde se ejecutan, desplazamiento de comunidades, trabajo muy mal remunerado, expansión de plagas, enfermedades [24], sequías y muerte de fauna en regiones como lo fue la emergencia que se dio en el 2016 en Paz de Ariporo, Casanare [25], entre otros.

Ante estos antecedentes, la detección de monocultivos de palma de aceite en Colombia es un tema de gran importancia y relevancia, ya que como se vio anteriormente estos siguen en un incremento considerable siendo de las principales causas del deterioro ambiental y social en las regiones del país donde se implementan. Por ello resulta fundamental implementar herramientas y estrategias que permitan monitorear la expansión de estos cultivos, de manera que se puedan tomar medidas preventivas y correctivas para mitigar su impacto tanto en el medio ambiente como en la sociedad.

En este contexto, la utilización de imágenes satelitales y modelos de CNNs se presenta como una alternativa innovadora y efectiva para la detección de monocultivos de palma de aceite en Colombia, sobre todo en zonas rurales remotas. Este enfoque permite analizar grandes cantidades de datos de manera rápida y eficiente, identificando patrones y características específicas de estos monocultivos, lo que facilita la toma de decisiones informadas por parte de los entes de control del manejo de los recursos naturales.

Además, la implementación de este tipo de tecnología contribuye a la promoción de prácticas sostenibles en la agricultura y el desarrollo de políticas encaminadas a proteger el medio ambiente y garantizar la sostenibilidad económica y social de las comunidades rurales.

Otro de los factores que motivan la realización de este trabajo es plantear un proceso para la obtención de imágenes satelitales para estos monocultivos u otro tipo de cultivos, incluso para otros tipos de estudios, ya que cuando se realizó la revisión del estado del arte no se encontró información clara de cómo obtener y trabajar este tipo de datos. Uno de los objetivos es entonces poder dejar claro este proceso y divulgar de manera pública el conjunto de datos que se utilizó esta investigación como un aporte ante nuevas investigaciones que trabajen con este tipo de imágenes.

Por lo anteriormente mencionado, la presente investigación tiene como objetivo principal desarrollar un modelo de redes neuronales convolucionales que permita la detección de monocultivos de palma de aceite en Colombia a partir de imágenes satelitales. Con esta investigación se espera, por un lado, contribuir al fortalecimiento en el proceso de detección tradicional por parte de los entes reguladores, y, por otro, dejar un proceso claro para la obtención de imágenes satelitales que pueda orientar en nuevas investigaciones.

## 3.2 Revisión de Conceptos

A continuación, se describen brevemente los diferentes conceptos, métodos, librerías y tecnologías fundamentales exploradas en el presente trabajo.

### 3.2.1 Inteligencia Artificial (AI)

La inteligencia artificial (IA) es un campo interdisciplinario que se dedica al desarrollo de algoritmos y sistemas capaces de realizar tareas que normalmente requerirían inteligencia humana, como el reconocimiento de patrones, el aprendizaje y la toma de decisiones. Uno de los pioneros en el campo de la IA fue Alan Turing, quien en 1950 propuso una prueba para determinar si una máquina podía exhibir un comportamiento inteligente indistinguible del de un ser humano [26], esta prueba se conoce como la Prueba de Turing [27].

La inteligencia artificial abarca los subcampos de *machine learning* y *deep learning*. La inteligencia artificial general (IAG) es una forma teórica de IA en la que una máquina tendría una inteligencia igual a la de los humanos; sería autoconsciente y tendría la capacidad de resolver problemas, aprender y planificar para el futuro [28].

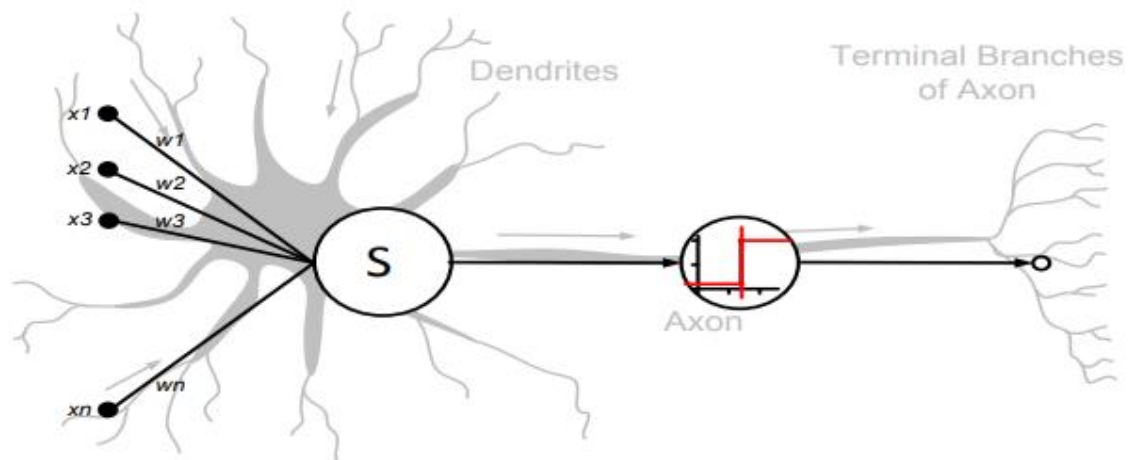
### 3.2.2 Red Neuronal Artificial

Una red neuronal artificial es un modelo matemático inspirado en el comportamiento biológico de las neuronas y en la estructura del cerebro y al igual que este, intentan “aprender” a partir de datos que se les suministran. Tienen una interconexión masiva y en paralelo de elementos simples organizados jerárquicamente; basándose en la emulación por medio de hardware o software de la actividad de las neuronas del sistema nervioso biológico.

En una serie de neuronas individuales (elementos procesadores), cada neurona actúa como un elemento procesador independiente, las entradas y los pesos de interconexión son procesados por una función de suma, el resultado de esta sumatoria es mapeado por una función de transferencia no lineal. La salida de la función no lineal es la salida de la neurona [29].

En la Figura 2 se presenta un modelo de red neuronal biológica en el cual se basan las redes neuronales artificiales [30]. Este modelo cumple con las siguientes condiciones:

- Si la suma de las entradas eléctricas es lo suficientemente potente como para activar la neurona, dispara o transmite estas señales a otras neuronas.
- Es importante tener en cuenta que una neurona se dispara sólo si la señal total recibida en el cuerpo celular excede un cierto nivel.
- Cada neurona realiza una suma ponderada de sus entradas y dispara una señal (binaria, en su forma más simple), si la entrada total excede un cierto nivel.



**Figura 2:** Modelo de red neuronal biológica, tomado de [30]

Las redes neuronales artificiales están compuestas por múltiples capas de neuronas que pueden estar interconectadas con otras neuronas en la misma o diferentes capas. Los dispositivos procesadores, ya sean algoritmos o hardware, tratan de modelar la estructura de la corteza cerebral animal, pero en una menor escala. Una red neuronal artificial grande puede tener cientos o miles de unidades procesadoras, mientras que el cerebro animal tiene billones de neuronas con su respectivo incremento de magnitud debido a la iteración del comportamiento con el entorno. Bernard Widrow y Marcial Hoff manifiestan que una red neuronal artificial es un “... *sistema computacional compuesto por un alto número de simples elementos procesadores, pero altamente interconectados, los cuales procesan la información por la respuesta en estado dinámico de entradas externas*” [31].

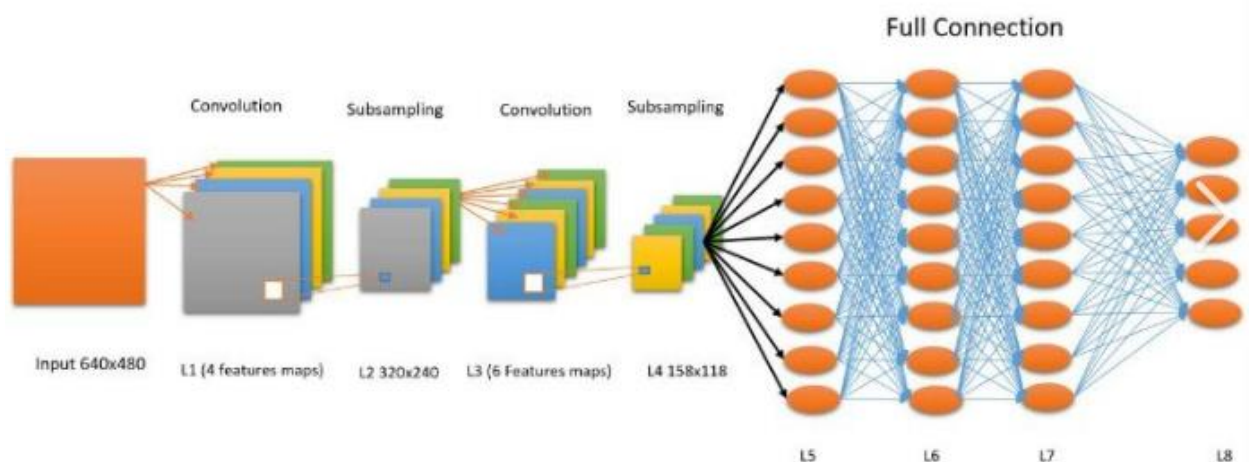
### 3.2.3 Redes Neuronales Convolucionales (***Convolutional Neural Network***)

Una red neuronal convolucional (CNN) es un tipo de red neuronal artificial que se utiliza principalmente para analizar imágenes visuales. En lugar de utilizar la multiplicación matricial general en al menos una de sus capas, las CNN utilizan una operación matemática llamada convolución. Están diseñadas especialmente para

procesar datos de píxeles y se utilizan en el reconocimiento, procesamiento, clasificación y segmentación de imágenes [32].

Las CNN han sido utilizadas en una amplia variedad de aplicaciones, desde la clasificación de imágenes hasta la detección de objetos y la segmentación semántica<sup>3</sup>. Al igual que otros tipos de redes neuronales artificiales, una CNN tiene una capa de entrada, una capa de salida, y varias capas ocultas [33]. Algunas de estas capas son convolucionales, utilizando un modelo matemático para transmitir los resultados a capas sucesivas. Esto simula algunas de las acciones en la corteza visual humana.

En la Figura 3 se puede ver la arquitectura de la CNN LeNet o LeNet 5, propuesta por Yann LeCun en 1989, esta es una red neuronal convolucional sencilla la cual define los elementos fundamentales en una CNN. Sin embargo, cabe recalcar que cuando fue presentada, no tuvo gran acogida debido a la falta de equipos de hardware, especialmente GPU (unidad de procesamiento gráfico<sup>4</sup>) y diferentes algoritmos como SVM podían tener efectos similares o incluso superar a LeNet.



**Figura 3:** Arquitectura de la red neuronal convolucional LeNet, tomado de [34]

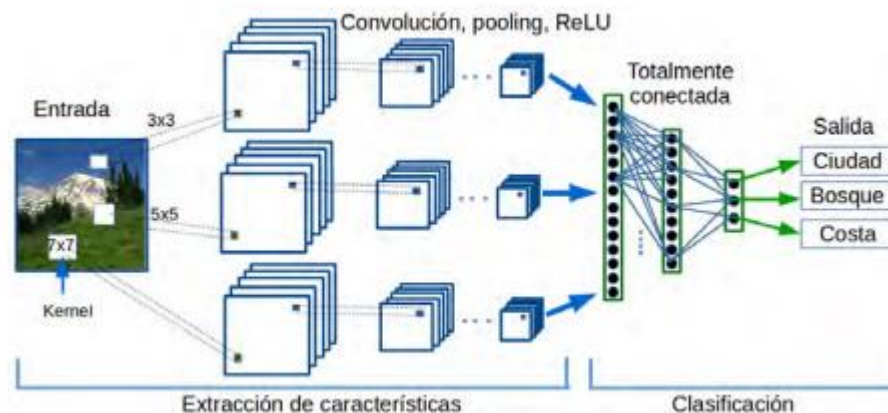
Las CNN son un ejemplo fundamental de aprendizaje profundo, donde un modelo más sofisticado impulsa la evolución de la inteligencia artificial, al ofrecer sistemas que simulan diferentes tipos de actividad biológica del cerebro humano. Desde el 2012 estas redes se han convertido en el algoritmo más usado para todas las tareas de visión computacional, generalmente trabajando con todas las tareas

<sup>3</sup> La segmentación semántica es un algoritmo de *deep learning* que asocia una etiqueta o categoría a cada píxel presente en una imagen.

<sup>4</sup> GPU es un tipo de procesador especializado en realizar operaciones matemáticas complejas sobre datos gráficos, como imágenes o vídeos. Se utiliza para acelerar el rendimiento de aplicaciones que requieren un alto nivel de procesamiento.

perceptuales [35]. Se especializan en procesar datos que tienen una topología de retícula (*grid*) conocida, los “datos de series de tiempo” se consideran de retícula 1D que muestra intervalos regulares de tiempo, y los “datos de imagen” se consideran una retícula de píxeles 2D (cuadrícula). Estos tipos de redes se basan en una operación llamada convolución, la cual es una operación matemática que se utiliza para combinar dos funciones y producir una tercera función que representa cómo una de las funciones se superpone a la otra [35]. En el contexto de las CNN, la convolución se utiliza para extraer características de las imágenes y obtener una representación compacta de las mismas. La convolución se realiza mediante la aplicación de un filtro o kernel a la imagen de entrada, que se desplaza por toda la imagen para producir una matriz de características.

Sus atributos los hacen perfectos para analizar imágenes, uno de los problemas que cualquier arquitectura de aprendizaje profundo sufre principalmente es el gran número de pesos, ya que comparan el patrón aprendido con respecto a su entrada, suponiendo que la entrada es la imagen (altura x ancho x canales). Cada neurona aprendería tantos pesos como píxeles tenga la imagen, lo cual es técnicamente difícil. Las imágenes replican sus patrones más básicos bastante, lo que hace que la red neuronal convolucional sea más fácil ya que (convolución 2D para imágenes) básicamente buscará el mismo patrón, pero en diferentes posiciones de la imagen [33] como se ve en la Figura 4.



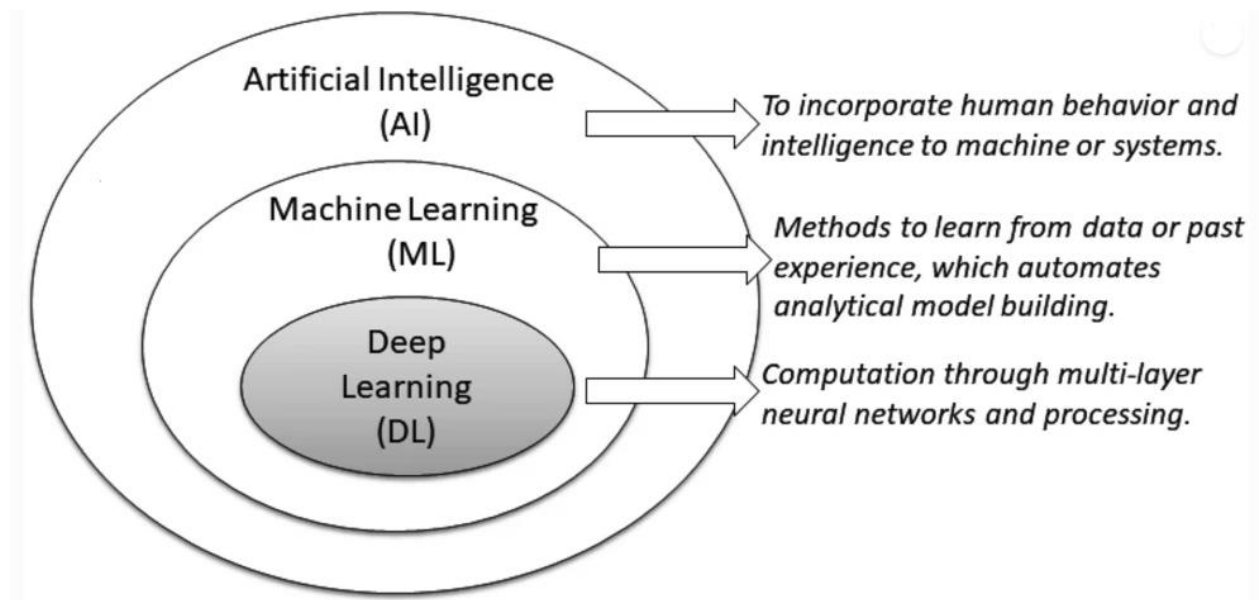
**Figura 4:** CNN con tres diferentes secciones de extracción de características, tomado de [36]

### 3.2.4 Aprendizaje Profundo (*Deep Learning, DL*) vs Aprendizaje Automático (*Machine Learning, ML*)

El aprendizaje profundo es un campo perteneciente a la IA cuyo objetivo es el estudio y construcción de sistemas de cómputo capaces de aprender a partir de la experiencia, inspirándose ligeramente en algunos principios del funcionamiento del

cerebro animal. En general, estos sistemas deben ser entrenados a partir de ejemplos conocidos, de manera similar en la que se le enseña a un niño pequeño a reconocer objetos y sonidos a su alrededor.

Hoy en día la inteligencia artificial (IA), el aprendizaje automático (*Machine Learning, ML*) y el aprendizaje profundo (*Deep Learning, DL*) son tres términos populares que a veces se usan indistintamente para describir sistemas o software que se comportan de manera inteligente. En la Figura 5, DL es una parte de ML, y también una parte del área amplia de la IA. En general, la IA incorpora el comportamiento y la inteligencia humanos a las máquinas o sistemas [37], mientras que el ML es el método para aprender de los datos o la experiencia [38], que automatiza la construcción de un modelo analítico. DL también representa métodos de aprendizaje a partir de datos donde se realiza el cálculo a través de redes neuronales y procesamiento de múltiples capas. El término “profundo” en la metodología de aprendizaje profundo se refiere al concepto de múltiples niveles o etapas a través de los cuales se procesan los datos para construir un modelo impulsado por datos.



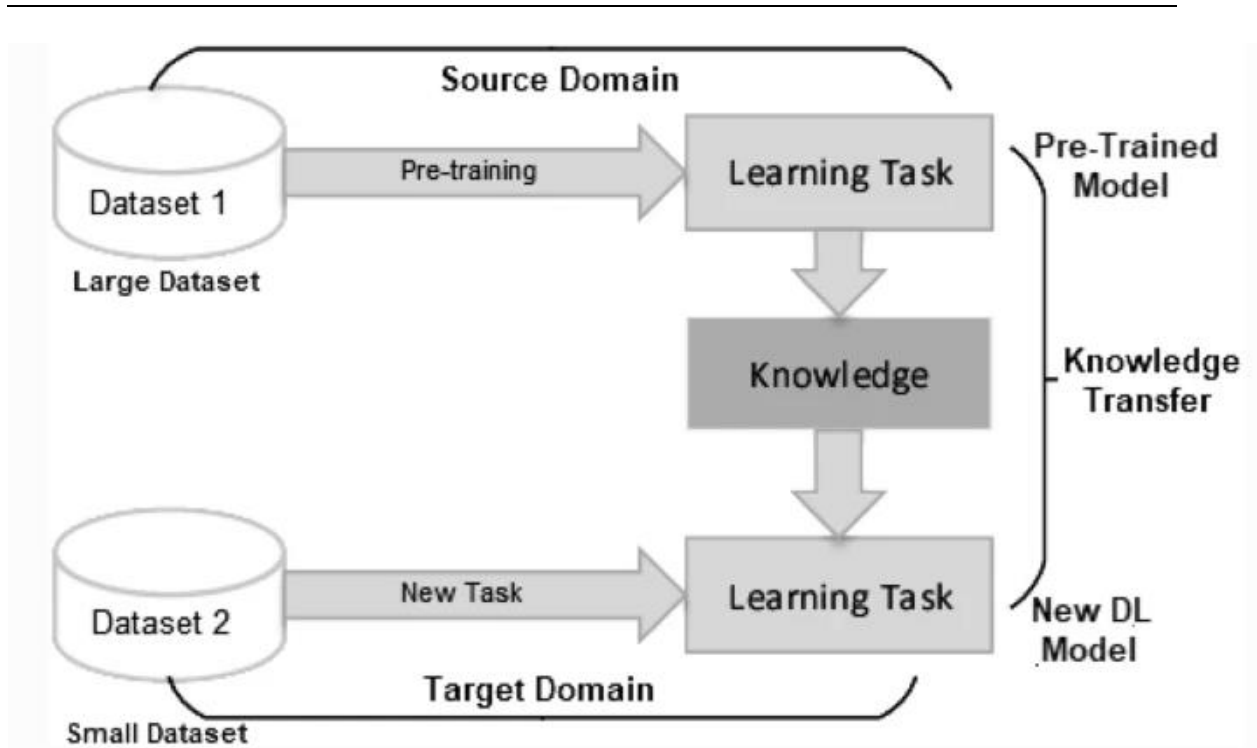
**Figura 5:** Posición del Aprendizaje Profundo (*Deep Learning*) comparado con Aprendizaje automático (*Machine Learning*) y la inteligencia artificial (IA), tomado de [39]

El aprendizaje profundo (DL) es una rama del aprendizaje automático (ML) y de la inteligencia artificial (IA) que se considera una tecnología central de la Cuarta Revolución Industrial (4IR o Industria 4.0). DL es capaz de aprender de los datos y se utiliza ampliamente en diversas áreas de aplicación como la atención médica, el reconocimiento visual, el análisis de texto, la ciberseguridad entre otros. DL es capaz de cambiar el mundo actual en términos de un motor computacional potente

y contribuir a la automatización impulsada por la tecnología, sistemas inteligentes según lo previsto por la Industria 4.0. Además, DL empuja a la IA a un nuevo nivel llamado “IA más inteligente”. En resumen, DL es capaz de cambiar el mundo actual en términos de un motor computacional potente y contribuir a la automatización impulsada por la tecnología de sistemas inteligentes [40].

### **3.2.5 Aprendizaje por transferencia (*Transfer Learning*)**

El aprendizaje por transferencia (*transfer learning*) es una técnica para utilizar efectivamente el conocimiento del modelo previamente aprendido para resolver una nueva tarea con un mínimo de entrenamiento o ajuste fino. En comparación con las técnicas típicas de aprendizaje automático [38], el aprendizaje profundo requiere una gran cantidad de datos de entrenamiento. Además, el modelo estándar de aprendizaje profundo requiere muchos recursos computacionales, como un servidor habilitado para GPU. Por lo tanto, el aprendizaje profundo por transferencia (*deep transfer learning*), un método de aprendizaje por transferencia basado en DL, podría ser útil para abordar este problema. En la Figura 6 se muestra la estructura general de este proceso de aprendizaje por transferencia, donde se transfiere el conocimiento del modelo pre-entrenado a un nuevo modelo de DL. Es especialmente popular en este momento ya que permite entrenar redes neuronales profundas con muy pocos datos [41].



**Figura 6:** Estructura general del aprendizaje por transferencia, donde el conocimiento del modelo previamente entrenado se transfiere al nuevo modelo DL, tomado de [39]

El aprendizaje por transferencia es un enfoque de dos etapas para entrenar un modelo de aprendizaje profundo que consta de una etapa de pre entrenamiento y una etapa de ajuste fino en la que se entrena el modelo en la tarea objetivo. Según las técnicas utilizadas en la literatura, el aprendizaje por transferencia profunda se puede clasificar en cuatro categorías [39]:

- I. Aprendizaje por transferencia profunda basado en instancias que utiliza instancias en el dominio fuente mediante un peso adecuado.
- II. Aprendizaje por transferencia profunda basado en mapeo que mapea instancias de dos dominios en un nuevo espacio de datos con mejor similitud.
- III. Aprendizaje por transferencia profunda basado en red que reutiliza la parte de la red pre-entrenada en el dominio fuente.
- IV. Aprendizaje por transferencia profunda basado en adversarios que utiliza tecnología adversaria para encontrar características transferibles que sean adecuadas para dos dominios.



Debido a su alta efectividad y practicidad, el aprendizaje por transferencia profunda basado en adversarios ha explotado en popularidad en los últimos años. El aprendizaje por transferencia también se puede clasificar en aprendizaje por transferencia inductivo, transductivo y no supervisado según las circunstancias entre los dominios fuente y objetivo y las actividades [42]. Las técnicas de aprendizaje por transferencia son útiles en una variedad de campos, incluido el procesamiento del lenguaje natural, la clasificación de sentimientos, el reconocimiento visual, el reconocimiento de voz, la filtración de correo *spam* y otros aspectos relevantes.

### 3.2.6 ResNet50

ResNet50 es una red neuronal convolucional profunda que fue desarrollada por Kaiming He, Xiangyu Zhang, Shaoqing Ren y Jian Sun de Microsoft Research en el 2015. Esta red es una versión mejorada de la arquitectura ResNet que se utiliza para la clasificación de imágenes. La arquitectura de ResNet se caracteriza por tener conexiones directas entre las capas, lo que permite que la información fluya más fácilmente a través de la red.

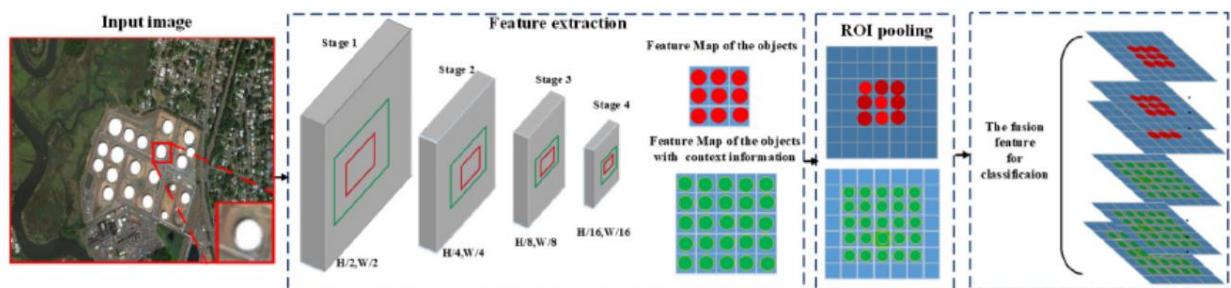
En [43], los autores implementan esta red en un problema de detección de objetos geoespaciales a múltiples escalas en imágenes de teledetección de alta resolución (HRRSI). Uno de los desafíos que enfrentan es que las regiones propuestas para contener los objetos (anclas) no se adaptan bien a las dimensiones reales de los mismos, lo que reduce la precisión y genera falsos positivos. Por tal razón los autores fueron combinando y probando diferentes arquitecturas de CNN como AlexNet, GoogleLeNet y ResNet para HRRSI. Sin embargo, encontraron que para este tipo de datos ResNet funciona mejor que otros métodos de clasificación al aumentar la profundidad de la red para generar la característica con más información semántica. En la Figura 7 se muestra la estructura de la CNN ResNet50.



**Figura 7:** Estructura de la red neuronal ResNet50, tomado de [43]

La Figura 8 muestra que el tamaño del mapa de características en Resnet50 se reduce debido a la capa de agrupación en cada etapa. Después de cuatro etapas, la relación entre el tamaño de la imagen original y el del mapa de características finales es de 16 a 1. Las cajas delimitadoras predichas tienen diferentes tamaños,

por lo que los mapas de características para diferentes cajas delimitadoras tienen diferentes tamaños. Para asegurar que la característica de cada caja tenga la misma dimensión en la clasificación, se adopta el ROI-pooling para normalizar diferentes cajas delimitadoras con el mismo tamaño. El tamaño de los mapas de características después del ROI-pooling generalmente se establece en  $7 \times 7$  píxeles empíricamente, lo que significa que el tamaño más adecuado de las imágenes originales es de  $112 \times 112$  píxeles. Como se muestra en la Tabla 1, podemos encontrar que la forma de objetos pequeños como vehículos y tanques de almacenamiento es solo de aproximadamente  $40 \times 40$  píxeles, lo que es mucho más pequeño que  $112 \times 112$  píxeles. Aunque el ROI-pooling puede cambiar el tamaño del mapa de características de  $7 \times 7$  mediante un muestreo ascendente o descendente, el mapa de características pequeño puede disminuir la precisión en la detección de objetos debido a una capacidad menos discriminativa para expresar la información de etiqueta. Por lo tanto, es fundamental aumentar la capacidad discriminativa de las características para obtener una mayor precisión en la clasificación [43].



**Figura 8:** Proceso de fusión de características con información de contexto, tomado de [43]

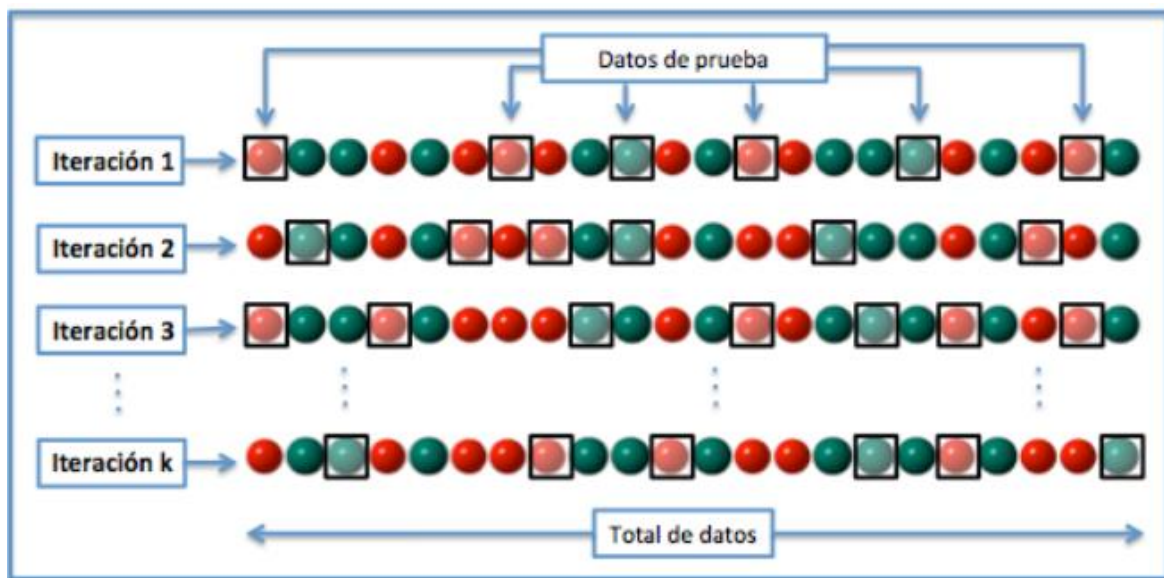
Durante los últimos años, la arquitectura del ResNet50 se ha utilizado con éxito en la detección de objetos en imágenes satelitales de alta resolución debido a su capacidad para aprender características complejas y su capacidad para manejar grandes conjuntos de datos [44]. Además, la arquitectura de ResNet50 es capaz de detectar objetos a diferentes escalas y orientaciones, lo que la hace adecuada para la detección y clasificación de objetos en imágenes satelitales de alta resolución [45]. En conclusión, la arquitectura de ResNet50 es una opción popular para la detección de objetos en imágenes satelitales de alta resolución debido a su capacidad para aprender características complejas, manejar grandes conjuntos de datos y detectar objetos a diferentes escalas y orientaciones.

### 3.2.7 Cross-validation o Validación cruzada

La validación cruzada es una técnica para evaluar el desempeño de un modelo que se quiere aplicar en la práctica. Consiste en dividir los datos de muestra en

subconjuntos de entrenamiento y prueba, y repetir el proceso varias veces con diferentes particiones [46].

Una de las formas de hacer validación cruzada es la aleatoria, que consiste en dividir los datos al azar en cada iteración. La ventaja de este método es que la división de datos entrenamiento-validación-prueba no depende del número de iteraciones, pero la desventaja es que algunos datos pueden quedar sin evaluar o evaluarse más de una vez [47].



**Figura 9** *Cross-Validation* o validación cruzada aleatoria con  $k$  iteraciones

En la Figura 9 se muestra una representación de validación cruzada aleatoria con  $k$  iteraciones, donde cada color representa una partición diferente de los datos.

### 3.2.8 Google Earth

Google Earth es una plataforma de software que muestra un globo virtual que permite visualizar la Tierra desde varias perspectivas gracias a la superposición de imágenes obtenidas por satélite y fotografías aéreas. Además, permite explorar el mundo en 3D, con edificios y terrenos en relieve. Fue creado y presentado por Michael Jones en el 2005, sin embargo, en el año 2004 Hanke y Keyhole crearon el programa EarthViewer 3D, el cual despertó el interés de Google, que compró la empresa y sus productos en el 2004. Con esto, EarthViewer 3D se transformó en lo que se conoce como Google Earth y también influyó en el desarrollo de Google Maps.

Google Earth se puede trabajar de forma directa en su página (<https://earth.google.com/>) o descargar localmente, no hay diferencias entre las limitaciones de las versiones. A continuación, se encuentran las actividades que puede realizar un usuario con Google Earth:

- Explorar el mundo en 3D directamente desde el navegador.
- Descubrir sitios nuevos.
- Disfrutar de una visita guiada utilizando Voyager.
- Crear mapas seleccionando un rango en un territorio específico.
- Especificar coordenadas de un lugar.
- Crear rutas entre diferentes lugares.

Google Earth utiliza imágenes satelitales de diferentes operadoras de satélites para crear su mapa en línea. La ESA, la Comisión Europea y la NASA son algunas de las organizaciones que han colaborado con Google Earth para proporcionar imágenes satelitales.

Las imágenes por satélite son esenciales para el funcionamiento de Google Earth, el cual ha integrado 24 millones de fotografías satelitales tomadas durante los últimos 37 años en una nueva capa de Google Earth. Con la nueva función de Google Earth denominada “*Timelapse*”, es posible contemplar casi cuatro decenios de evolución por todo el mundo. La actualización ofrecerá pruebas visuales de las drásticas modificaciones que tienen lugar en nuestro planeta, incluidos los efectos del cambio climático y la acción humana. La nueva función permite a los usuarios explorar la Tierra desde una dimensión totalmente nueva: el tiempo. Los usuarios pueden viajar por todo el mundo y descubrir la forma cambiante de los litorales, seguir el crecimiento de las megalópolis, ver el progreso de la deforestación y mucho más. La actualización ha sido posible gracias a los datos abiertos proporcionados por el programa Copernicus de la Unión Europea y sus satélites Sentinel, así como al programa Landsat de la NASA y el Servicio Geológico de los Estados Unidos [48].

### **3.2.9 Google Earth Engine**

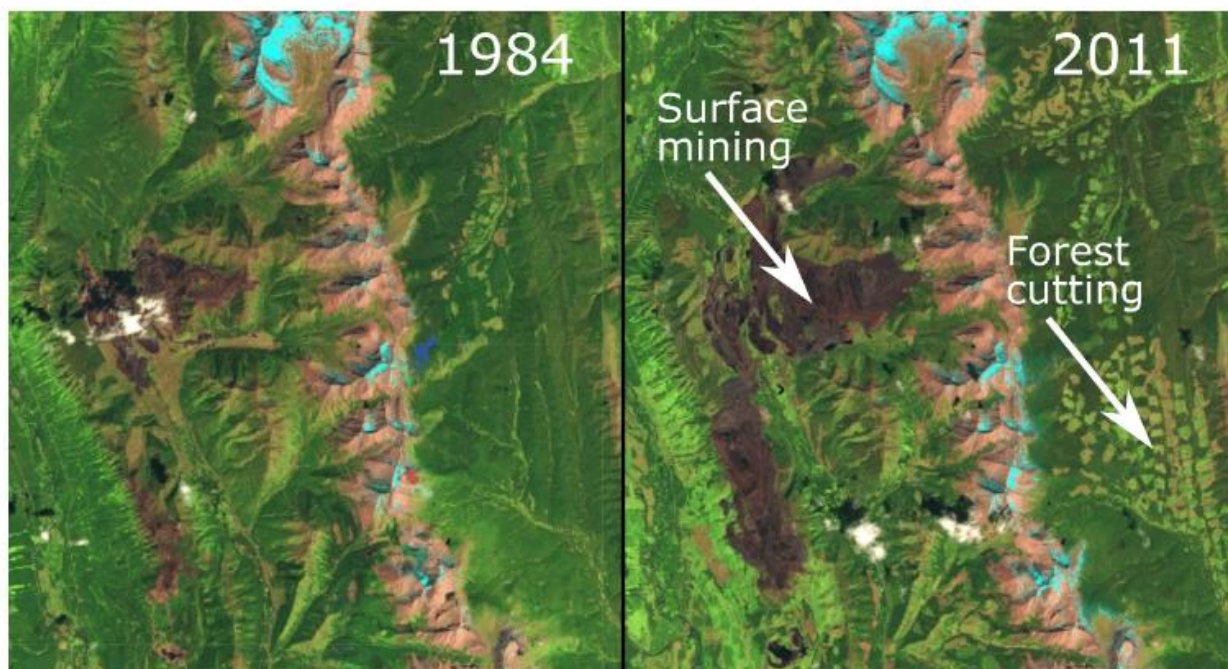
Google Earth Engine es una Plataforma de análisis geoespacial basada en la Nube que combina un catálogo de imágenes satelitales y conjunto de datos geoespaciales de varios petabytes con capacidades de análisis a escala planetaria. Los científicos, investigadores y desarrolladores utilizan Earth Engine para detectar cambios, mapear tendencias y cuantificar diferencias en la superficie terrestre.

La plataforma ofrece imágenes espaciales disponibles en la Nube de Google, lo que brinda a cada uno de los usuarios la oportunidad de observar y analizar contenido. A pesar de que en Google Earth el usuario puede crear mapas, esta plataforma no permite descargar imágenes con alta resolución, pero Google Earth Engine es una herramienta enfocada en facilitar el procesamiento de grandes volúmenes de información geoespacial que permite la descarga de este tipo de imágenes.

Este componente de Google Earth ha permitido realizar análisis previamente imposibles de ejecutar en una computadora personal y acotar de forma significativa el tiempo de procesamiento requerido para realizar diferentes tipos de análisis. Google Earth Engine ofrece otras funciones avanzadas como la clasificación de la superficie terrestre, la descarga de conjuntos de datos y la posibilidad de crear algoritmos propios.

Cuenta con algoritmos para modificar la visualización de bandas de datos, de tal forma que pueden verse en una escala de grises de una sola banda, en pseudocolor de una sola banda o en RGB de tres bandas. La visualización de tres bandas es útil para ver datos de imagen en los que se asigna cada una de las tres bandas seleccionadas a gradientes de color rojo (R), verde (G) y azul (B), respectivamente. Al mezclar las bandas en el espacio RGB, se crea un color de visualización final. Esto permite crear una imagen que muestre características que no son visibles en la visualización de una sola banda, como la textura, la forma y la composición. La visualización en color falso es una técnica que se utiliza para resaltar estas características mediante la asignación de colores artificiales a las bandas individuales. Por ejemplo, se puede utilizar la visualización en color falso para resaltar áreas con vegetación, agua, nieve o suelo desnudo en una imagen de satélite [49].

Google Earth Engine además permite ver cambios a lo largo del tiempo sobre una zona específica. En la Figura 10 se muestra un ejemplo práctico de cómo aplicar una comparación de fechas. Se han fusionado dos clips de pantalla de EE Explorer y se han añadido anotaciones para proporcionar información sobre el cambio de la superficie terrestre que ha registrado el satélite Landsat. El ejemplo muestra la expansión de una mina de carbón a cielo abierto y el talado de árboles que tuvo lugar cerca de Elkford.



**Figura 10:** Ejemplo de expansión de una mina de carbón a cielo abierto y el talado de árboles que tuvo lugar cerca de Elkford (British Columbia, Canadá) de 1984-2011, tomado de [49]

GEE ha sido utilizado para generar datos geográficos, a menudo a escala global, sobre diferentes temas como el monitoreo de vegetación y bosques, el mapeo de aguas superficiales, la detección de islas de calor, de minas, de incendios, etc. [50]. En la Tabla 3 se encuentran los elementos que componen GEE:

**Tabla 3:** Principales elementos de GEE

Elementos Principales		Descripción
1	Infraestructura de Google	Pone a disposición del usuario sus servidores, permitiendo así hacer análisis en paralelo con cerca de 10 000 CPUs. Esto agiliza la velocidad de procesamiento, en comparación con una computadora individual [51]
2	Datasets	Google tiene almacenado todas las imágenes de varios sensores (Landsat, Sentinel, MODIS, entre otros). Estas bases de datos se actualizan a medida que se toman nuevas imágenes (cerca de 6000 nuevas escenas diarias), creando así un enorme catálogo de datos geoespaciales. Estas bases se pueden consultar a través de diferentes criterios (calidad, localización, fechas) sin necesidad de descargar ni solicitar acceso a las imágenes [51]. Igualmente, GEE permite usar información vectorial, pero a diferencia de las imágenes, los vectores no se encuentran organizados en un catálogo y su búsqueda puede resultar tediosa (Google Developers, 2018).
3	API ( <i>Application Program Interface</i> )	Consiste en una serie de comandos y funciones preestablecidas, escritos en lenguaje JAVA, que permite una programación sencilla al desarrollar algoritmos para las investigaciones, donde solo se permite usar las funciones del API. Existe también un API para el lenguaje Python, pero debe ser descargado. Esto limita el procesamiento ya que consume memoria de la computadora del usuario.
4	Code Editor	Es un entorno de desarrollo integrado en línea, donde se juntan todos los elementos. Aquí es donde el usuario puede, a través de código de trabajo (" <i>scripts</i> "), llamar a los datos, procesar y visualizarlos de manera virtual con los servidores de Google, teniendo así sus resultados e información en la Nube

### 3.2.10 Sentinel 1 y 2

Tanto Sentinel 1 como Sentinel 2 son dos satélites desarrollados por la ESA<sup>5</sup> dentro del programa Copérnico para desarrollar observaciones del planeta tierra para dar servicios como el seguimiento de la evolución de los bosques, los cambios en la corteza terrestre y la gestión de los desastres naturales [52].

Sentinel 1 es un satélite que órbita alrededor de la Tierra en una órbita polar y proporciona imágenes de radar de alta resolución. Por otro lado, el objetivo de Sentinel 2 es el monitoreo terrestre y la misión estará compuesta por dos satélites

<sup>5</sup> Agencia Espacial Europea, organización intergubernamental de 22 estados miembros dedicada a la exploración del espacio. Fue fundada en 1975 y tiene su sede en París.

polar – orbitales que proporcionarán imágenes ópticas de alta resolución temporal (de 10 m a 60 m), es de 10 días por satélite o 5 días por conjunto [53].

El radiométrico de Sentinel 2 es de 12 bits, lo que da un potencial rango de niveles de brillo desde 0 hasta 4.095.

En la Figura 11 se encuentra la infografía de cada uno de los satélites [54] :



**Figura 11:** Infografía de los satélites Sentinel 1 y Sentinel 2, tomado de [54]

Algunas de las aplicaciones de Sentinel 1 incluyen el monitoreo de la extensión del hielo marino y los icebergs, la vigilancia marítima ( monitoreo de derrames de petróleo, detección de barcos, pesca ilegal), dentro del monitoreo terrestre se encuentra riesgos de movimientos como hundimientos y deslizamientos de la tierra, la comprensión de los procesos terrestres (terremotos, volcanes), el monitoreo de la infraestructura y el mapeo para la gestión forestal, del agua y del suelo y el mapeo para apoyar la ayuda humanitaria y situaciones de crisis.

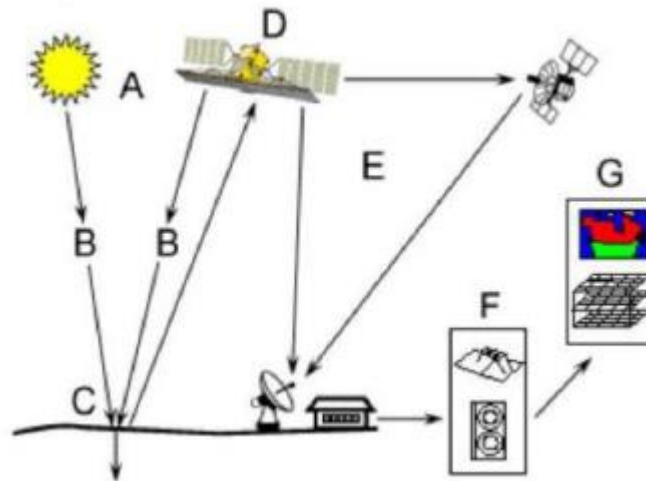
Por otro lado, el Sentinel 2 se utilizó para monitorear la superficie terrestre y forestal y los cambios en la superficie. Sus aplicaciones principales incluyen la agricultura, el monitoreo de los ecosistemas terrestres, la gestión de los bosques y el monitoreo de la calidad de agua costera e interior, el mapeo de desastres y la seguridad civil.

### 3.2.11 Proceso de Teledetección Satelital

Para obtener las imágenes satelitales se realizó el proceso de teledetección, el cual es la recopilación de información de la superficie terrestre sin entrar en contacto con ella. Como se mencionó anteriormente este proceso se realizó mediante sensores a bordo de satélites artificiales [55], los cuales miden la cantidad de radiación electromagnética que refleja la superficie de la tierra y los objetos que



hay en ella [56]. La Figura 12, tomada del grupo de investigación GOTA<sup>6</sup> de La Universidad de la Laguna [57], ilustra el proceso de teledetección.



**Figura 12:** Proceso de teledetección e iteración con la radiación incidente y los objetos de interés. Grupo de investigación GOTA, Universidad de Laguna, tomado de [57]

En la Figura 12:

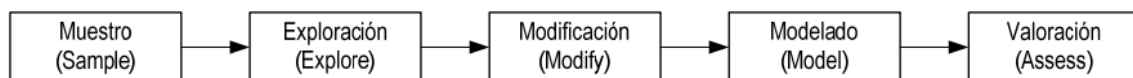
- A. Se encuentra la fuente de energía o de iluminación, la cual provee energía electromagnética al objeto de interés.
- B. Ya que la energía se transporta desde la fuente al objeto, en este punto se encuentra la radiación y la atmósfera.
- C. Interacción con el objeto dependiendo las propiedades de este y radiación incidente.
- D. En este punto se presenta la detección de la energía por el sensor remoto, el cual graba la radiación electromagnética emitida por el objeto.
- E. El sensor transmite la información grabada de forma electrónica a una estación de recepción.
- F. Se presenta la interpretación y análisis, se procesa e interpreta la imagen visual o digitalmente para extraer información del objeto que emitió la radiación.
- G. Paso final del proceso de teledetección en el que se toma la información extraída de las imágenes del objeto para un mejor conocimiento del mismo, revelando nueva información para resolver un problema particular.

<sup>6</sup> Grupo de observación de la tierra y la atmósfera, Universidad de la Laguna

## 4. Metodología

Para el desarrollo y orientación de esta investigación se seleccionó la metodología SEMMA creada por el SAS Institute. Esta se define como el proceso de selección, exploración y modelado de grandes cantidades de datos para revelar patrones de negocio desconocidos [58].

El nombre de esta terminología corresponde al acrónimo de las fases básicas del proceso: *Sample* (Muestreo), *Explore* (Exploración), *Modify* (Modificación), *Model* (Modelado) y *Assess* (Valoración).



**Figura 13:** Fases de la metodología SEMMA, tomado de [58]

A continuación, se realiza una descripción de cada una de las fases de la metodología [59]:

**Fase I Muestreo (*Sample*)** -> Extracción de una muestra representativa:  
En esta fase se debe seleccionar una muestra representativa de la población a analizar, siendo la aleatoriedad el método más común de selección denominado muestreo aleatorio simple. Se enfatizó la necesidad de establecer un nivel de confianza en la muestra seleccionada.

**Fase II Exploración (*Explore*)** -> Exploración de los datos de la muestra seleccionada:  
En esta fase se realiza un análisis de los datos extraídos en la muestra, donde se propone el uso de herramientas de visualización o de diferentes técnicas estadísticas para la exploración de la información seleccionada. Esta fase permite simplificar el problema y optimizar la eficiencia del modelo.

**Fase III Modificación (*Modify*)** -> Modificación de los datos:

Esta fase involucra la modificación de los datos que van a ser ingresados al modelo para que tengan el formato adecuado, mejorando la definición de los mismos.

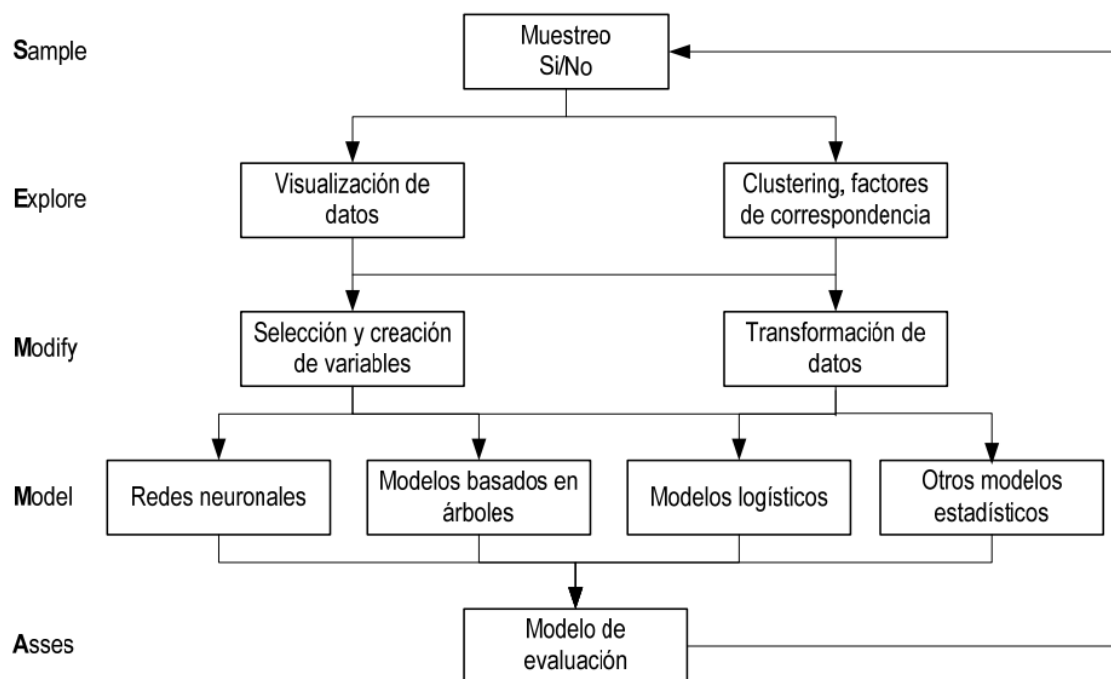
**Fase IV Modelado (Model) -> Modelación de los datos:**

En esta fase, se modela el conjunto de datos para predecir los resultados esperados de manera confiable. El objetivo es establecer una relación entre las variables objeto del estudio y las variables explicativas, de manera tal que posibiliten inferir el valor de las mismas con un nivel de confianza determinado. Las técnicas utilizadas para el modelado de los datos incluyen técnicas adaptativas, lógica difusa, reglas de asociación, árboles de decisión, redes neuronales y computación evolutiva; también involucran métodos estadísticos tradicionales.

**Fase V Valoración (Asses) -> Evaluación de los datos**

La última fase consiste en la valoración de los datos obtenidos para determinar el grado de confiabilidad de estos y así poder evaluar el modelo, mediante la comparación de otros o con nuevas poblaciones muestrales.

En la Figura 14 se puede ver un esquema de la dinámica de la metodología:



**Figura 14** Dinámica de la metodología SEMMA, tomado de [58]

Teniendo en cuenta la descripción de las fases de esta metodología, en la Tabla 4 se presentan las fases que se llevaron a cabo para el desarrollo de esta investigación.

Fase	Descripción	Objetivo
<b>I. Recolección de Datos</b>	Se realizó la extracción de una muestra de datos que permita representar características comunes de nuestro objeto de estudio.	Extraer una muestra significativa de imágenes satelitales de un repositorio de dominio público.
<b>II. Transformación de Datos</b>	En esta fase se realizó una exploración y transformación en las imágenes obtenidas a través técnicas (niveles de intensidad, lógica, interpolación, etc.) para estandarizar la información que se va a procesar. Además, si es necesario, se realiza una eliminación de datos que presenten anomalías o ruido entre los datos.	Realizar una transformación de las imágenes extraídas de acuerdo con el estándar que se defina (nivel de intensidad, tamaño).
<b>III. Selección y validación del modelo de CNN</b>	Selección del modelo de CNN, validar la posibilidad de combinar diferentes técnicas de acuerdo con la revisión de la literatura, el cual se aplicará a las imágenes satelitales extraídas para la detección de monocultivos de palma de aceite.	Realizar una revisión bibliográfica para seleccionar un modelo de CNN que se va a aplicar sobre las imágenes satelitales extraídas.
<b>IV. Implementación del modelo CNN</b>	Se realizó la implementación del modelo seleccionado de CNN sobre las imágenes satelitales extraídas para obtener una clasificación de la información y un resultado con un alto nivel de confianza.	Implementar el modelo de CNN seleccionado sobre las imágenes satelitales para realizar una clasificación como resultado, integrando una traza del desempeño del modelo (medidas de eficiencia, tiempos de ejecución, etc.).
<b>V. Análisis del desempeño y Resultados</b>	Valoración del modelo implementado en la fase anterior. Se realizó una evaluación tanto de los resultados como del desempeño del modelo para verificar el éxito del proyecto.	Evaluar el desempeño y resultados del modelo de CNN que se implementó, validación de matriz de confusión.

**Tabla 4:** Fases de la metodología que se van a llevaron a cabo para el desarrollo de la investigación

En las siguientes secciones se presenta de manera detallada la aplicación de las etapas de la metodología descrita previamente.

## 4.1 Comprensión del negocio

Para llevar a cabo el desarrollo de la investigación de Detección de monocultivos de palma de aceite en Colombia implementando un modelo de redes neuronales convolucionales a partir de imágenes satelitales, se plantea un modelo de CNN con arquitectura ResNet50 aplicada sobre imágenes satelitales de monocultivos de Colombia, extraídas de los satélites Sentinel 1 y Sentinel 2.

Para esta implementación y ejecución del modelo son necesarios los siguientes insumos:

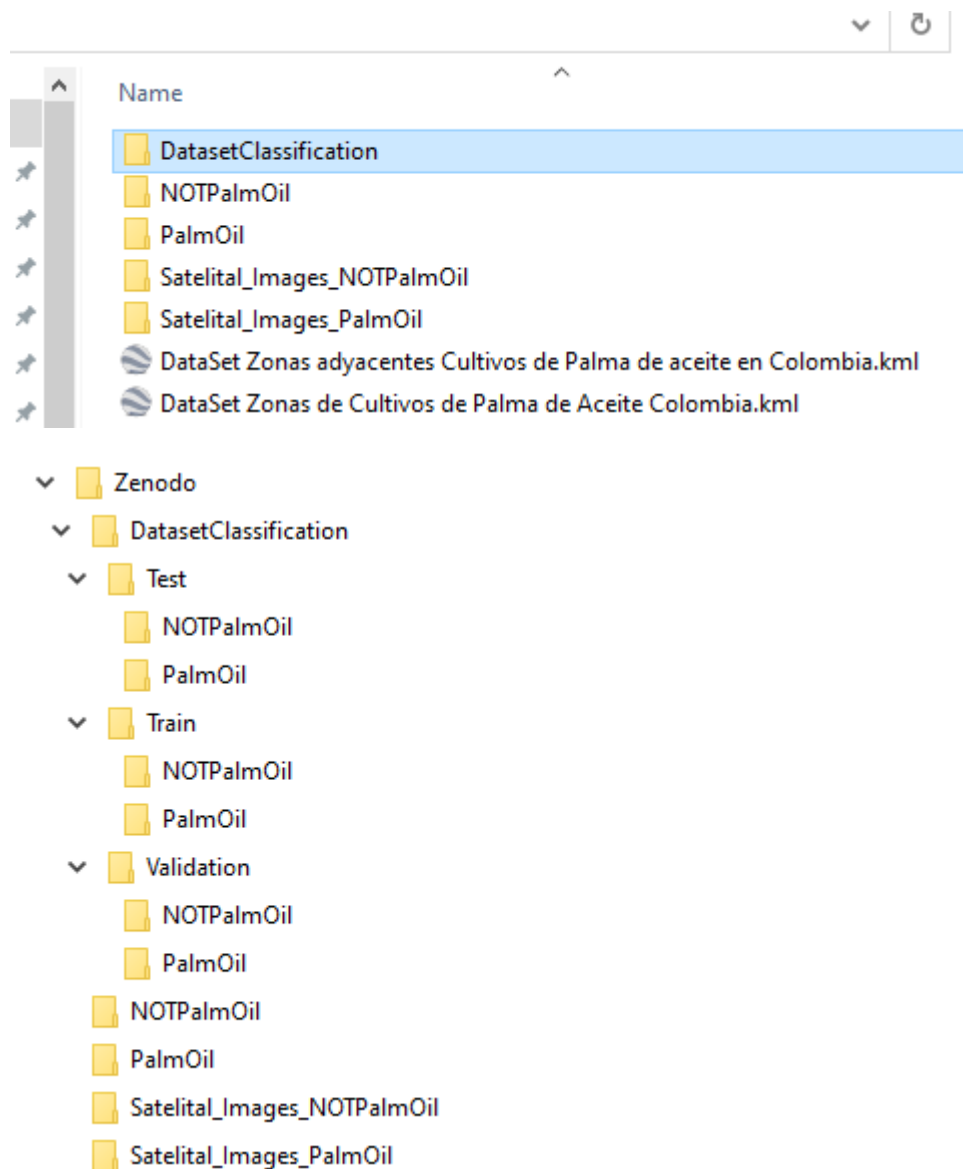
- a) Archivo KML<sup>7</sup> de Google Earth con la ubicación de los monocultivos de palma de aceite de nuestro país.
- b) Archivo KML de Google Earth con la ubicación de zonas donde no se encuentren presencia de monocultivos de palma de aceite.
- c) *Scripts* de Google Earth Engine para exportar imágenes de los satélites Sentinel 1 y Sentinel 2, de cada una de las ubicaciones, tanto de los monocultivos como de las zonas donde no se presentan estos.
- d) Notebook de Colab para generar los siguientes *datasets*:
  - Imágenes de monocultivos de palma de aceite para entrenamiento.
  - Imágenes de monocultivos de palma de aceite para pruebas.
  - Imágenes de zonas donde NO se encuentran monocultivos de palma de aceite para entrenamiento.
  - Imágenes de zonas donde NO se encuentran monocultivos de palma de aceite para pruebas.

---

<sup>7</sup> Keyhole Markup Language, es un lenguaje de marcado basado en XML para representar datos geográficos en tres dimensiones.

Dando cumplimiento a uno de los objetivos de esta investigación de compartir el conjunto de datos que se utilizó, esta información se subió a la plataforma de Zenodo (<https://zenodo.org/>), la cual permite compartir datos para diferentes

investigaciones. El *DataSet* se encuentra en el siguiente enlace web publico: <https://zenodo.org/record/7869523#.ZEne33bMIkk> y se encuentra relacionado en el apartado de anexos, tabla de productos generados, ítem 1. Esta información tiene la siguiente estructura de carpetas:



**Figura 15:** Estructura de carpetas del *DataSet* compartida por la plataforma Zenodo

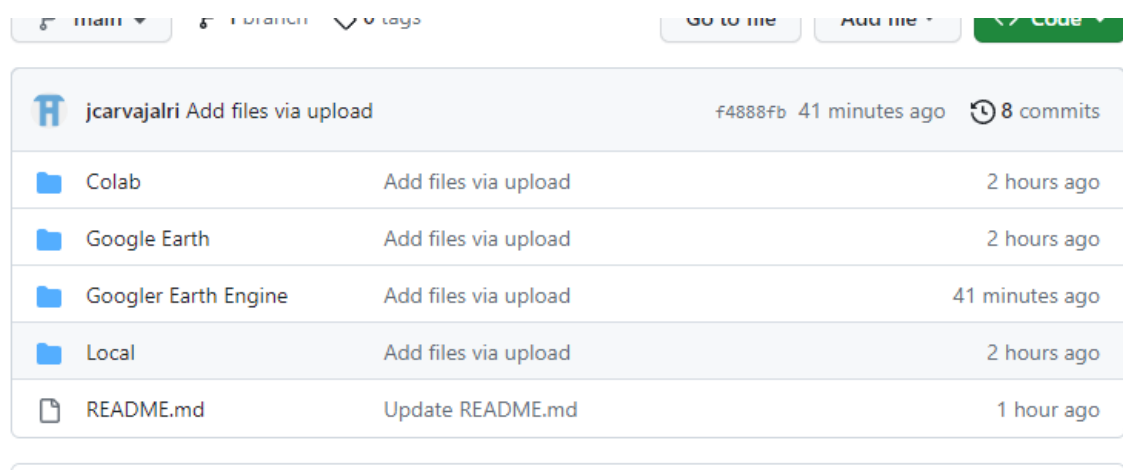
---

La estructura de las carpetas del *Dataset* se encuentra de la siguiente forma:

- *Satelital\_Images\_PalmOil*: imágenes originales de los satélites Sentinel 1 y 2, extraídas de Google Earth Engine de monocultivos de palma de aceite en Colombia.
- *Satelital\_Images\_NOTPalmOil*: imágenes originales de los satélites Sentinel 1 y 2, extraídas de Google Earth Engine de zonas que NO contienen monocultivos de palma de aceite en Colombia.
- *DataSet Zonas adyacentes Cultivos de Palma de aceite en Colombia*: Archivo para abrir con Google Earth con las ubicaciones de los monocultivos de palma de aceite en Colombia.
- *DataSet Zonas de Cultivos de Palma de Aceite Colombia*: Archivo para abrir con Google Earth con las ubicaciones de zonas que NO contienen monocultivos de palma de aceite en Colombia.
- Una carpeta con el *datasetClassification*, el cual dentro tiene la siguiente información:
  - *Train: Dataset* de datos para entrenamiento de zonas que NO contienen monocultivos de palma de aceite en Colombia (NOTPalmOil). *Dataset* de datos para entrenamiento de monocultivos de palma de aceite en Colombia (PalmOil).
  - *Validation: Dataset* de datos para validación en la etapa de entrenamiento de zonas que NO contienen monocultivos de palma de aceite en Colombia (NOTPalmOil). *Dataset* de datos para validación en la etapa de entrenamiento de cultivos de palmas de aceite en Colombia (PalmOil).
  - *Test: Dataset* de datos para pruebas de zonas que NO contienen monocultivos de palma de aceite en Colombia (NOTPalmOil). *Dataset* de datos para pruebas de monocultivos de palma de aceite en Colombia (PalmOil).

**Nota:** Las imágenes de *datasetClassification* tiene un tamaño de 256x256.

De igual forma, en la Figura 16 se encuentra la estructura de archivos de Github:



The screenshot shows a GitHub repository interface. At the top, there are navigation buttons: 'Main', 'Branches', 'Tags', 'Go to file', 'Add file', and 'Code'. Below this is a table listing the repository's structure:

File/Folder	Commit Message	Commit Time
Root	Add files via upload	41 minutes ago
Colab	Add files via upload	2 hours ago
Google Earth	Add files via upload	2 hours ago
Googler Earth Engine	Add files via upload	41 minutes ago
Local	Add files via upload	2 hours ago
README.md	Update README.md	1 hour ago

**Figura 16** Estructura de carpetas de GitHub.

En el apartado de Anexos tabla de productos generados se encuentra la relación y el contenido de cada una de estas carpetas.

## 4.2 Preparación de los datos

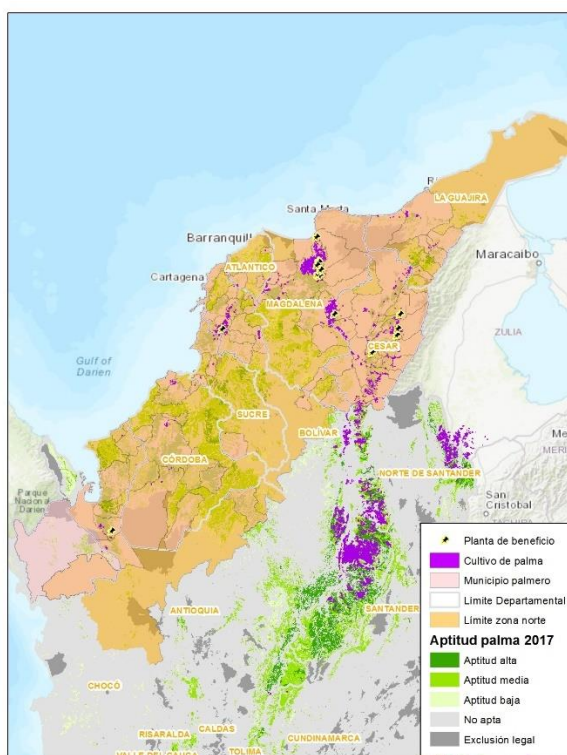
En la presente sección se encuentra la descripción del proceso para llevar a cabo la recolección de los *DataSets* utilizados en las fases de Entrenamiento y Prueba.

### 4.2.1 Ubicación de las zonas para el *DataSet*

Inicialmente, para obtener la ubicación de zonas de monocultivos de palma de aceite en Colombia, se buscó un conjunto de datos público. Al no encontrar al respecto información, se entró a validar con entidades como la Federación Nacional de Cultivadores de Palma de Aceite, conocida como Fedepalma. Esta entidad tiene en su página web una sección para la localización geográfica de cultivos y plantas de beneficio. En esta sección se encuentra información de la parte oriental, norte, centro, suroccidental de nuestro país donde hay presencia de cultivos de palma de aceite.

Sin embargo, como se observa en la Figura 17, la información proporcionada por Fedepalma carece de la ubicación exacta de donde se encuentran estos cultivos de palma de aceite, siendo éste el eje central para poder extraer imágenes satelitales necesarias para la investigación.

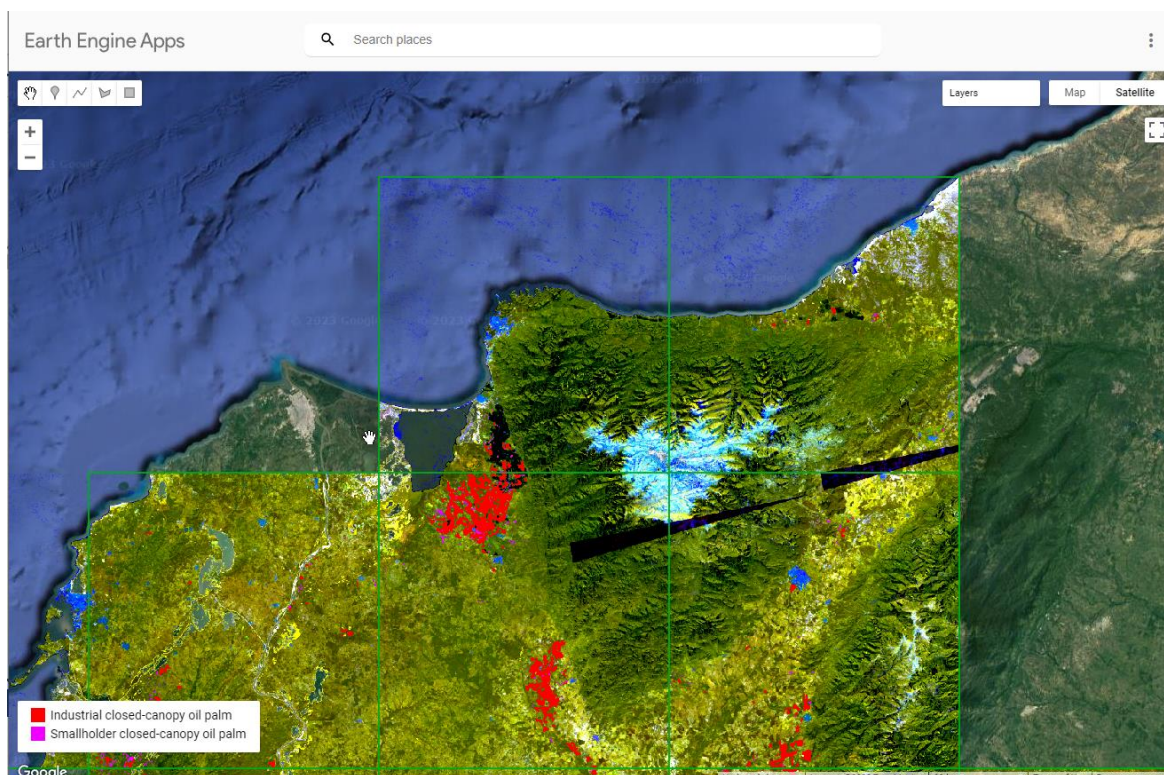




**Figura 17:** Localización geográfica de cultivos y plantas de beneficio zona norte de Colombia año 2017, tomado de [60]

En la investigación de Descals Adria et al. [61] comparten de manera pública un conjunto de datos que contiene 634 mosaicos de 100x100km, que cubren áreas donde se detectaron plantaciones de palma de aceite a nivel mundial para el año 2019. La diferencia de este conjunto de datos es que lo comparten por medio de un mapa de Google Earth Engine que se puede encontrar en el siguiente [link](https://adriadescals.users.earthengine.app/view/global-oil-palm-map-2019) <https://adriadescals.users.earthengine.app/view/global-oil-palm-map-2019>.

Haciendo una navegación de este mapa, en la Figura 18 se puede ver en la zona norte de nuestro país la presencia de cultivos de palma de aceite. A diferencia de Fedepalma, sin embargo, zonas del mapa concuerdan con la información suministrada, esté si permite extraer las ubicaciones exactas de donde se encuentran los cultivos de palma de aceite.



**Figura 18:** Zona norte de Colombia con presencia de cultivos de palma de aceite.

Con este mapa se procedió a definir las zonas donde se encuentran los cultivos de palma de aceite en Colombia. Utilizando Google Earth, en la Figura 19 se encuentran ubicaciones de los monocultivos en los departamentos de Magdalena, Cesar, Santander, Meta y Casanare.



**Figura 19:** Ubicación de cultivos de palma de aceite en Colombia

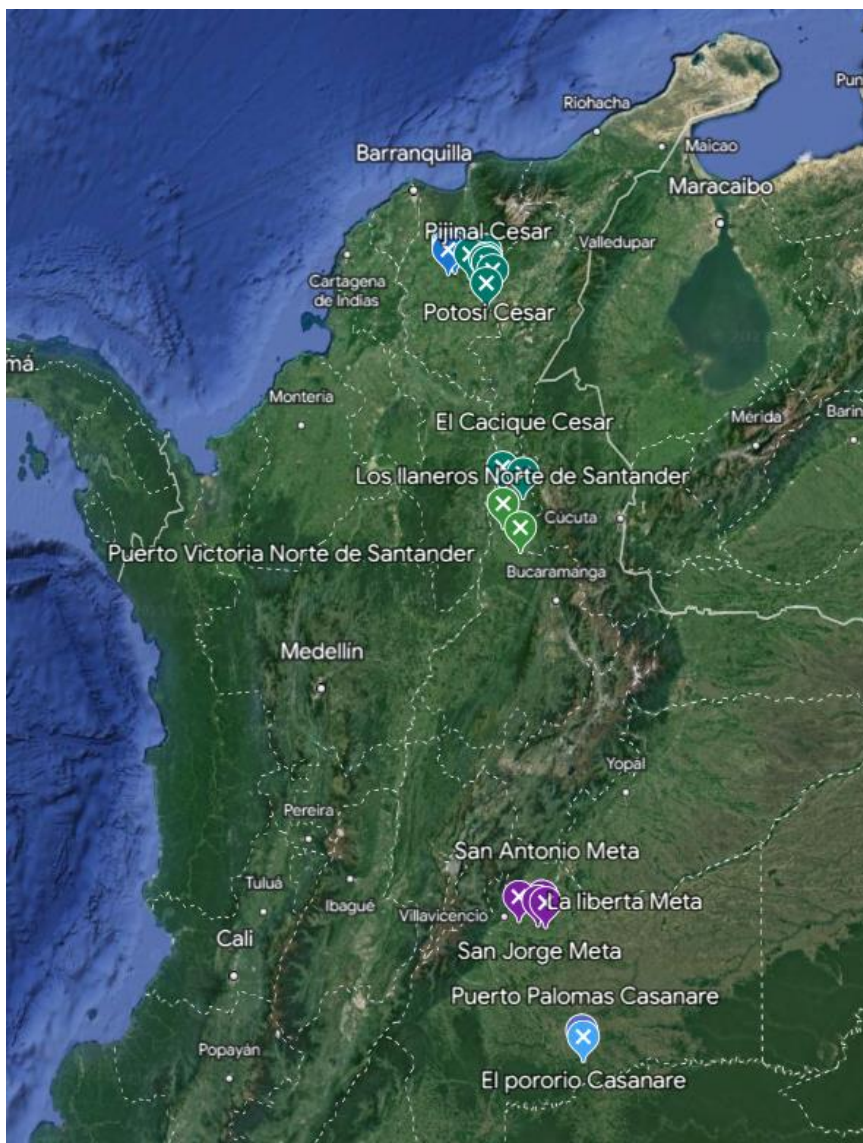
Al trabajar con Google Earth, cada ubicación queda con su meta data de la posición y las anotaciones que se desee agregar. Esta es exportada en un archivo de tipo KML. En la Tabla 5 se encuentran cada una de las regiones y departamentos de los cultivos de palma de aceite en Colombia con su respectiva posición geográfica, la cual será necesaria cuando se extraigan las imágenes satelitales.

Región	Departamento	Longitud	Latitud
Vereda Carretal	Magdalena	-74.34659449183918	10.63152451118795
Vereda los Guayabos	Magdalena	-74.24888937775287	10.70101041099089
Santa Cruz	Magdalena	-74.291345112191	10.70101041099089

La Granja	Magdalena	-74.23332331730769	10.63330253165864
La Calma	Magdalena	-74.19906821520803	10.77523994104465
San Luis	Cesar	-74.02880675212832	10.19210997365469
Agrias Dulces	Cesar	-74.04552065482265	10.15047277342473
El Delirio	Cesar	-74.03705251192156	10.09185506658664
Bella Cruz	Cesar	-73.70478079482341	8.635152952465271
Las Minas	Cesar	-73.46171509659673	7.889686851584138
La Llana	Cesar	-73.44453164995022	7.697452784288545
Maribel	Santander	-73.38148313946525	7.438038220025881
San Cayetano	Meta	-73.61067505961252	3.979302347988663
Pompeya	Meta	-73.37477866964039	3.925387741620258
Vera Cruz	Meta	-73.25470103320576	4.224333997902439
Floralta	Meta	-73.15031108309576	4.109407937564502
Pompeya II	Meta	-73.2383373631375	3.942833908019982
El Estereo	Meta	-73.37958358717324	3.826994366140771
Mapiripan	Casanare	-72.0897246457908	2.929798551312425

**Tabla 5:** Ubicaciones de cultivos de palma de aceite en Colombia. Información extraída del KML Google Earth

Para el *dataset* de las imágenes satelitales que NO contienen cultivos de palma de aceite, la estrategia que se implementó dado que estos cultivos, por lo general, se encuentran en tierras llanas, fue seleccionar territorios adyacentes a la ubicación de cultivos de palma de aceite. En la Figura 20 se encuentran las ubicaciones de estas zonas.



**Figura 20:** Zonas adyacentes a cultivos de palma de aceite

Los archivos KML, tanto de las ubicaciones de cultivo de palma de aceite como de las zonas adyacentes que no contienen cultivos de palma de aceite, se encuentran en el anexo de tabla de productos generados ítem 2.

#### 4.2.2 Imágenes satelitales base

Para la generación de imágenes base se extrajeron imágenes de los satélites Sentinel 1 y Sentinel 2 con las ubicaciones de los cultivos de palma de aceite, por

medio de la implementación de un *script* de Google Earth Engine, como aporte de la investigación de Adrià Descals et al. [61] del instituto CREAM<sup>8</sup>.

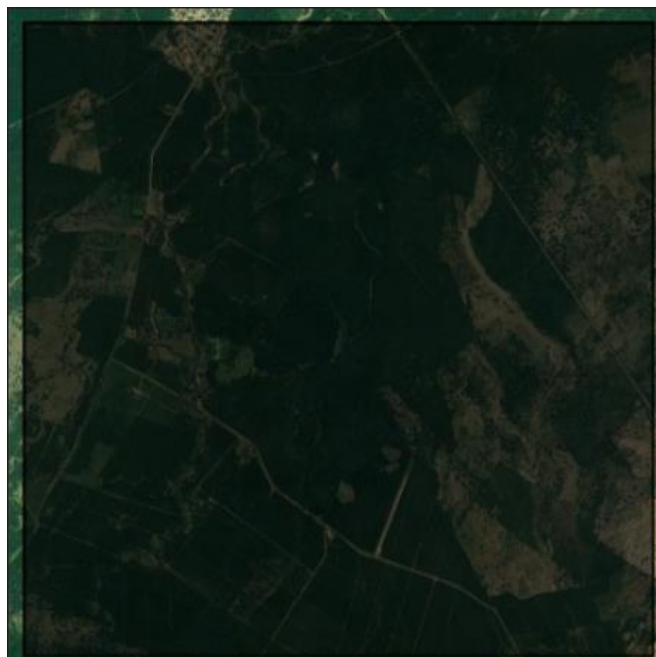
Este *script* (el cual se encuentra en el anexo de tabla de productos generados ítem 3), dada una ubicación permite seleccionar un área cuadrada con un rango en kilómetros que define el usuario y extraer 2 imágenes de los satélites Sentinel 1 y Sentinel 2. A continuación se describen las secciones principales del *script*.

- I. **Configuración (Setup):** Define las fechas finales y el tamaño de la ventana temporal (en meses) para los compuestos (*composites*) Sentinel-1 y Sentinel-2. Además, define el área de exportación como un rectángulo alrededor de un punto específico. Para la investigación se seleccionó un rango de 4 kilómetros.
- II. **Creación de Compuesto Sentinel I (S1 Composite):** En esta sección se creó un compuesto Sentinel-1 a partir de imágenes en modo ascendente (ASC) y descendente (DSC). Se aplicaron varias funciones para calcular el ángulo de incidencia local (LIA) y corregir las imágenes según el ángulo de incidencia. Luego, se promediaron las imágenes en modo ascendente y descendente. El resultado es un compuesto de imágenes Sentinel-1 en dos bandas (VV y VH) que es escalado y convertido a uint8.
- III. **Creación de Compuesto Sentinel II (S2 Composite):** En esta sección, se creó un compuesto de imágenes Sentinel-2. Primero, se aplicó una máscara de probabilidad de nubes (*Cloud Probability Map*) y otra máscara basada en la clasificación de escenas (*Scene Classification Map*) para eliminar las áreas nubladas. Además, se aplicó una máscara adicional para eliminar las sombras de las nubes. Luego, se calcula el índice de vegetación de diferencia normalizada (NDVI) para cada imagen y se agrega como una banda adicional. El resultado es un compuesto de imágenes Sentinel-2 con la banda NDVI incluida.
- IV. **Exportar Compuestos a Drive (Export Drive):** En el *script*, esta sección está vacía, permitiendo al usuario realizar una configuración con la cuenta de Google drive personal para exportar las imágenes de la sección.

A continuación, en la Figura 21 se muestra un ejemplo con cultivos de palma de aceite del área original de la región de Aguas Dulce Cesar con un área rectangular de 4 km x 4 km (centrada en el punto de intereses), antes de ser exportada por el *script*.

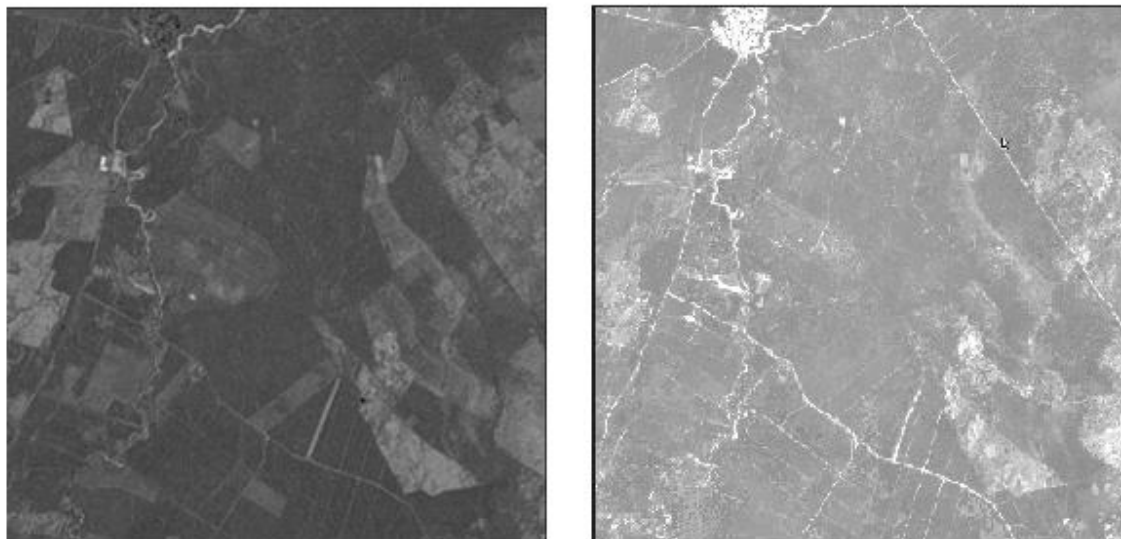
---

<sup>8</sup> Centro de investigación ecológica y aplicaciones forestales, dedicado a la ecología terrestre, el análisis territorial y el cambio global.



**Figura 21:** Área de cultivos de palma de aceite, región Aguas Dulces Cesar.

En la Figura 22 se encuentran dos imágenes generadas por el *script* de la misma región de Aguas Dulces Cesar. Se observa como en la imagen exportada por el satélite de Sentinel 1, se hizo relevancia en las texturas del área, mientras que en la imagen del satélite Sentinel 2, sobresalen más los bordes y los límites de los cultivos de palma de aceite.



**Figura 22** Imágenes exportadas de los satélites Sentinel 1 y Sentinel 2 por medio del *script* [56] de Google Earth Engine, región Aguas Dulces Cesar.

Estas imágenes exportadas en formato TIFF<sup>9</sup>, se denominarán “imágenes satelitales base” para la investigación. En total se tienen 36 imágenes con presencia de cultivos de palma de aceite, y 46 imágenes que NO tienen cultivos de palma de aceite. Cada imagen está relacionada con las ubicaciones de las zonas que se establecieron en el punto anterior.

### 4.2.3 Conjunto de imágenes para entrenamiento y prueba

A continuación, se encuentra el proceso que se llevó a cabo para la generación del *Dataset* de entrenamiento y de prueba, teniendo en cuenta las imágenes satelitales base. Cada una de las siguientes actividades se llevaron a cabo por medio de un notebook de Colab que está relacionado en el anexo de tabla de productos generados ítem 4.

#### 4.2.3.1 División de las imágenes satelitales base.

Obtener las imágenes satelitales base por medio del *script* de GEE es un proceso tedioso, y exportarlas es un proceso muy manual, ya que GEE no permite que se automatice esta actividad. Además, al ser las imágenes de alta calidad, toma un tiempo de generación y de exportación. Llegar a obtener una cantidad significativa de imágenes satelitales directamente desde GEE, puede llegar a ser una actividad que demande un esfuerzo adicional y de tiempo.

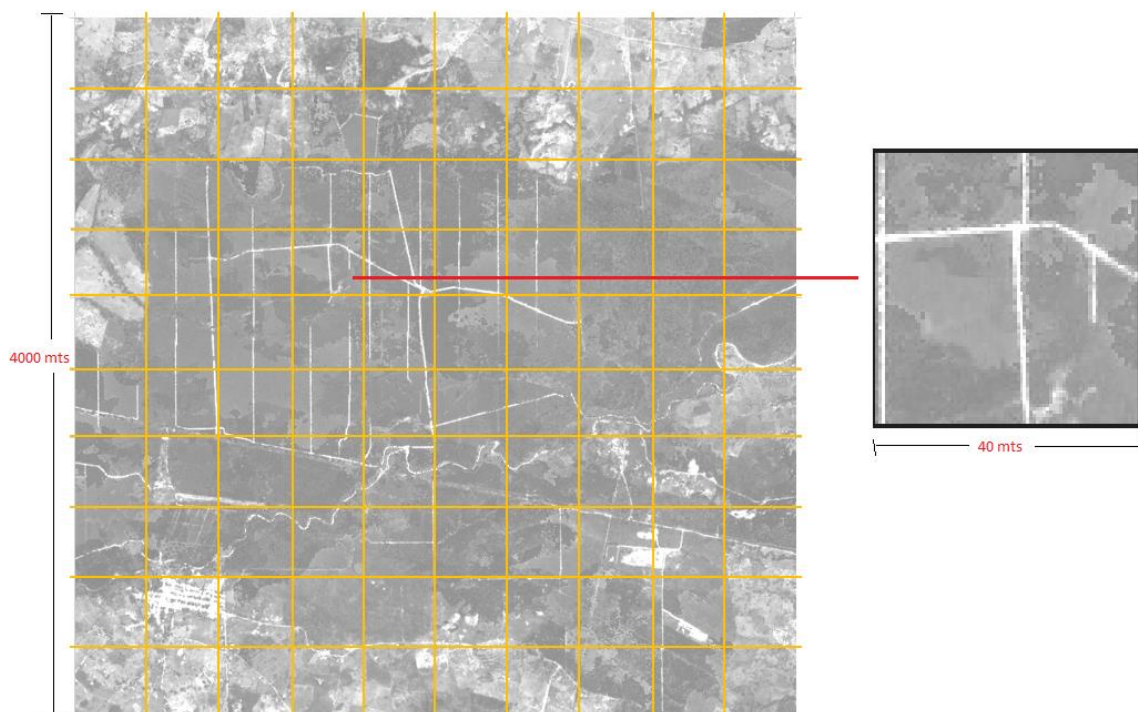
Ante lo manifestado anteriormente, se creó una estrategia de dividir cada una de las imágenes satelitales base en 100 imágenes. Como cada imagen tiene un área rectangular de 4 km x 4 km (centrada en el punto de intereses), cada una de las imágenes después de la división tendría un área de 1600 metros cuadrados (40 m x 40 m).

En la Figura 23 se encuentra una representación de la división de una imagen satelital base con su respectiva escala de la región de Bella Cruz Cesar. De esta forma, por cada una de las imágenes satelitales base, se obtuvieron 100 imágenes.

---

<sup>9</sup> Tagged Image File Format, es un formato de archivo informático para almacenar imágenes de mapas de bits. Es relevante en la industria gráfica y fotográfica por su alta calidad.





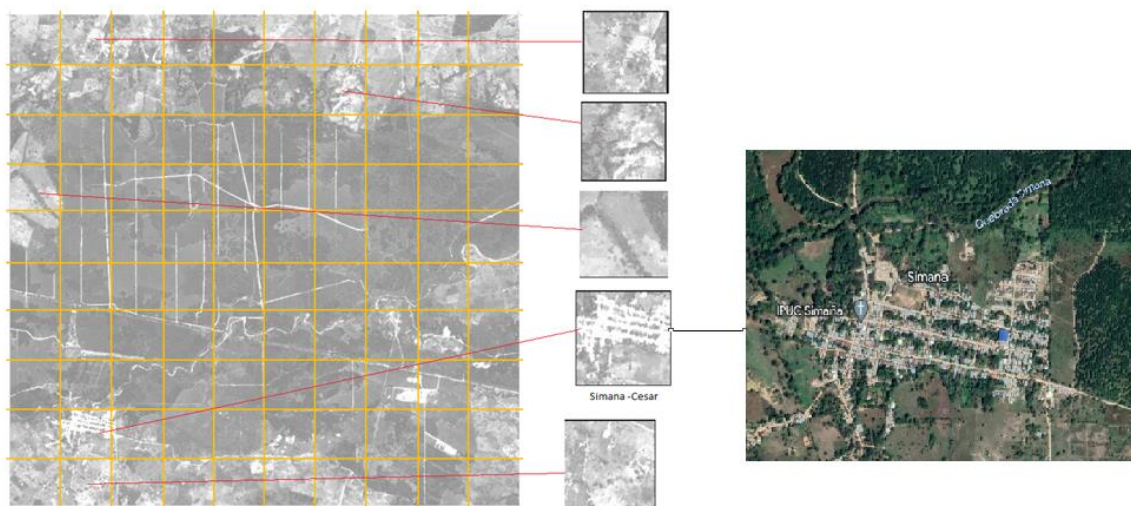
**Figura 23:** Representación gráfica de la división de una imagen satelital base, región Bella Cruz Cesar.

Cada una de estas imágenes resultantes de la división se guardó en formato JPG.

#### 4.2.3.2 Limpieza del *Dataset*

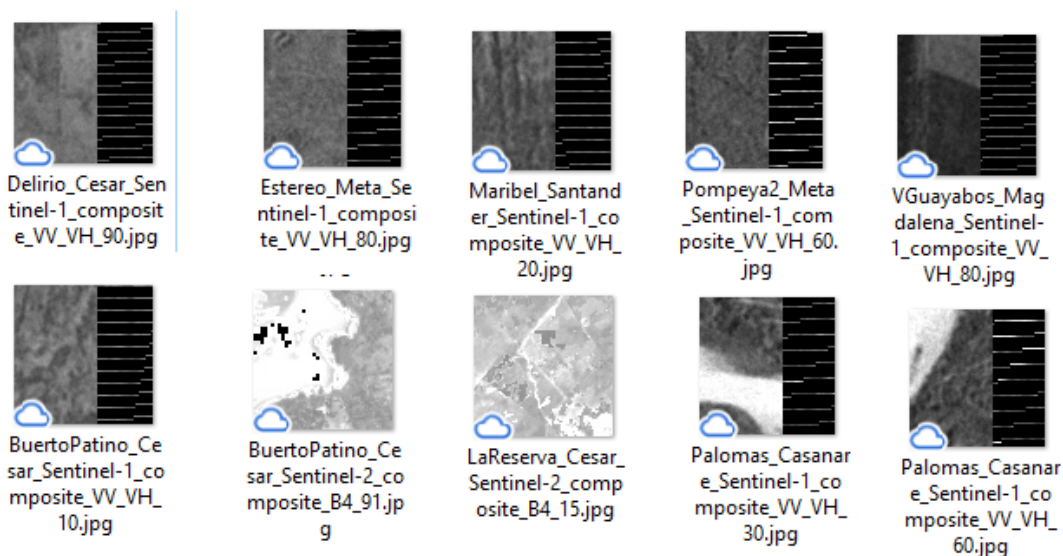
Teniendo en cuenta que las imágenes satelitales base representan un cuadrado de una región, para el *DataSet* de las imágenes que contienen cultivos de palma de aceite, no toda el área de la imagen satelital base tiene el cultivo de palma de aceite, por lo tanto, se debió realizar una actividad de limpieza.

En la Figura 24 se muestra un ejemplo de la limpieza en el área de Bella Cruz Cesar, donde a pesar de que en este sector se encuentran presentes cultivos de palma de aceite, después de realizar la división de la imagen satelital base, algunas de las imágenes resultado no contienen cultivos de palma de aceite. En algunos casos adyacentes a estos cultivos de palma de aceite se encuentran municipios aledaños, para el ejemplo se tiene el corregimiento de Simana Cesar.



**Figura 24:** Ejemplo de limpieza de imágenes que no tienen presencia de cultivos de palma de aceite, región Bella Cruz Cesar

Por otro lado, al realizar la división de las imágenes base específicamente del satélite Sentinel 1, algunas imágenes quedaban con ruido (franja negra), o en otros casos aparecían con algunas manchas de color negro. En la Figura 25 se muestran algunos ejemplos de estas imágenes; cabe recalcar que esta limpieza se realizó tanto para las imágenes que tienen cultivos de palma de aceite como para las zonas adyacentes a estos cultivos, que no cuentan con presencia de cultivos de palma de aceite.



**Figura 25:** Ejemplos de algunas imágenes extraídas del satélite Sentinel 1 con presencia de ruido.

### 4.2.3.3 Enriquecimiento del *DataSet*

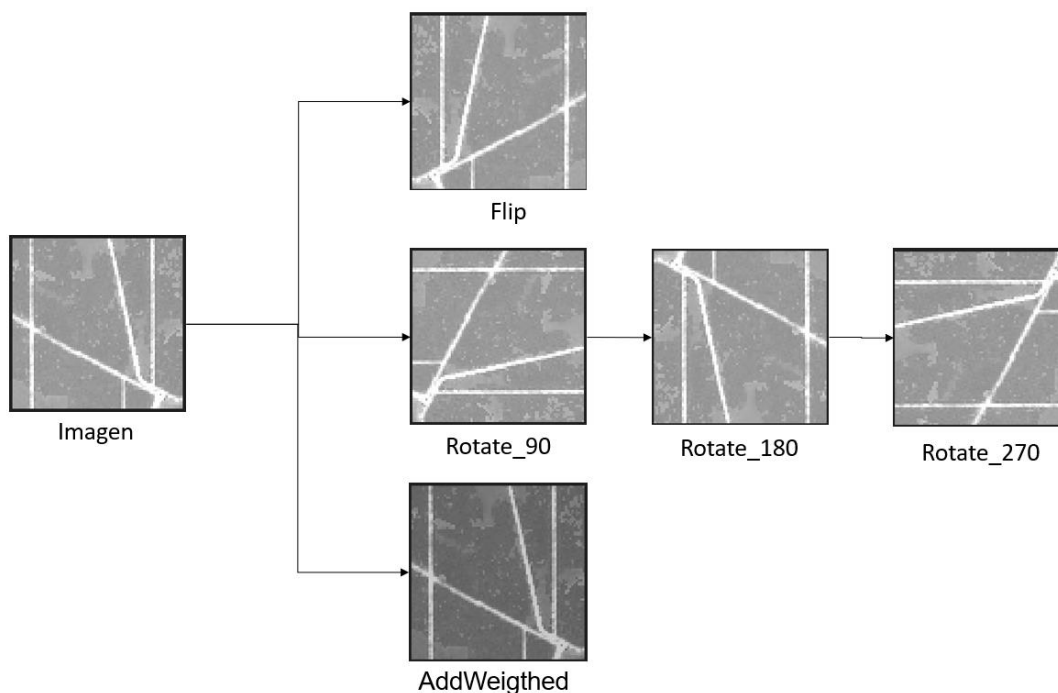
Después de realizar la limpieza del *DataSet*, se procedió a realizar un enriquecimiento de los datos por medio de las siguientes técnicas que proporciona la librería de OpenCV<sup>10</sup> de Python.

- I. ***Flip***: Se utilizó para voltear una imagen o matriz 2D alrededor de los ejes vertical, horizontal o ambos. También es conocido como efecto espejo.
- II. ***Rotate***: Este método se utilizó para rotar una imagen o una matriz 2D en múltiplos en ángulos múltiplos de 90 grados; 90,180,270.
- III. ***AddWeighted***: Esta función se utilizó con el objetivo de ajustar el brillo y contraste de la imagen original.

En la Figura 26 se muestra un ejemplo de la implementación de estas técnicas sobre una imagen del *DataSet*. Estas técnicas fueron aplicadas a todo el conjunto de datos que se tenía de las imágenes que contienen cultivos de palma de aceite. Posteriormente, se aplicaron también al otro conjunto de datos de zonas adyacentes a los cultivos, pero que no contienen cultivos de palma de aceite, con el objetivo de igualar la cantidad obtenida por el *Dataset* de cultivos de palma de aceite.

---

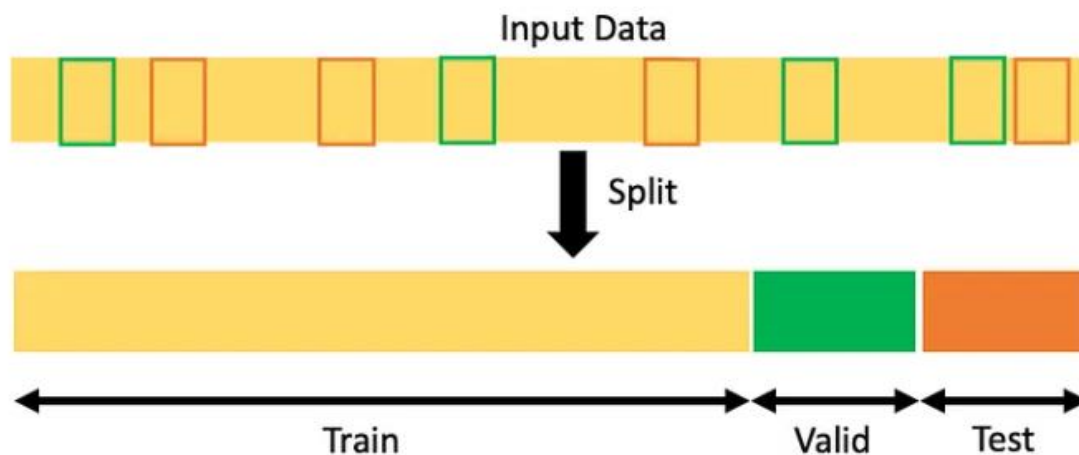
<sup>10</sup> OpenCV es una librería de Python de funciones de programación orientada principalmente a la visión artificial en tiempo real.



**Figura 26:** Ejemplo de las técnicas aplicadas a una imagen para el enriquecimiento del *DataSet*.

#### 4.2.4 *Dataset* Final

Teniendo el total de imágenes para las dos clases (Palm Oil y NOT PalmOil), para el desarrollo de la clasificación del *DataSet* definitivo de los datos por medio de una función random (siendo la división aleatoria de datos el método más utilizado para una evaluación imparcial), se definieron 3 conjuntos de datos para cada clase como se observa en la Figura 27: Un conjunto de datos de entrenamiento (*Train*), un conjunto de datos de validación (*Validation*) y un conjunto de datos de prueba (*Test*).



**Figura 27:** Grupos de clasificación del conjunto definitivo de los datos de forma aleatoria tomado de [62].

De acuerdo con los autores en [62][63], se realiza una descripción de cada conjunto de datos:

- *Train:* El conjunto de datos de entrenamiento se compone de muestras utilizadas para ajustar el modelo en construcción, es decir, llevar a cabo el desarrollo real de la IA. Este conjunto de datos es utilizado para el aprendizaje, es decir, para ajustar los parámetros al modelo de aprendizaje automático.
- *Validation:* En contraste, los conjuntos de datos de validación contienen muestras diferentes para evaluar y ajustar el modelo de ML en la etapa de entrenamiento. Este conjunto de datos se utilizó para proporcionar una evaluación imparcial de un modelo ajustado en el conjunto de datos de entrenamiento, mientras se ajustaron los hiperparámetros del modelo. Juega también un papel en otras formas de preparación del modelo, como la selección de características, la selección de corte del umbral.
- *Test:* El conjunto de datos de prueba es una muestra separada para proporcionar la evaluación final imparcial del ajuste del modelo. Las entradas son similares a las etapas anteriores, pero no son los mismos datos, se simplifica en decir que estos datos no los conoce el modelo que se entrenó.

Los conjuntos de datos utilizados para entrenar y validar pueden desempeñar roles adicionales en la preparación del modelo, como la selección de características opcionales. El ajuste del modelo final es un resultado combinado del conjunto total de estas entradas.

En la Tabla 6 se encuentra el total de imágenes que se utilizó en el modelo de CNN seleccionado.

Clasificación del Dataset				
	Palm Oil		NOT_Palm Oil	
	Cantidad	%	Cantidad	%
Imágenes después de la limpieza	2,236	17%	4,326	32%
Subtotal Imágenes generadas	11,255	83%	9,165	68%
<b>Total de imágenes Dataset</b>	<b>13,491</b>	<b>100%</b>	<b>13,491</b>	<b>100%</b>
<b>Total imágenes de entrenamiento (Train)</b>	<b>9,442</b>	<b>70%</b>	<b>9,442</b>	<b>70%</b>
<b>Total imágenes de Validación (Validation)</b>	<b>2,699</b>	<b>20%</b>	<b>2,699</b>	<b>20%</b>
<b>Total imágenes de pruebas (Test)</b>	<b>1,350</b>	<b>10%</b>	<b>1,350</b>	<b>10%</b>

**Tabla 6** Clasificación del *Dataset*

### 4.3 Preparación de Entorno de trabajo

La presente sección se centra en la preparación del entorno de trabajo necesario para llevar a cabo el desarrollo del presente trabajo. La preparación del entorno de trabajo es fundamental para garantizar la eficiencia y efectividad del proceso de detección. Para ello se abordaron dos aspectos fundamentales: la preparación del ambiente local y el ambiente en la Nube.

El ambiente local se refiere a la configuración, optimización del equipo de cómputo utilizado, incluyendo la instalación de las herramientas y librerías necesarias, así como la preparación de los conjuntos de datos y su procesamiento. Por otro lado, el ambiente en la Nube se enfocó en aprovechar los recursos computacionales y el almacenamiento remoto disponible.

En esta sección se describen detalladamente los pasos y consideraciones necesarias para establecer tanto el ambiente local como el ambiente en la Nube, sentando las bases para el desarrollo y la implementación del modelo seleccionado.

#### 4.3.1 Ambiente Local

Antes de describir la lista de paquetes a instalar y la preparación del ambiente local, en la Tabla 7 se encuentran las características del equipo de trabajo para el ambiente local, el cual contiene una tarjeta de video NVIDIA Geforce GTX 1650 Super, para tener una ejecución de los modelos de aprendizaje profundo de manera eficiente.

Sistema Operativo	CPU	GPU	RAM	Disco
Windows 10 Home 64-BIT	AMD Ryzen 9 3900X 12-Core Processor 3.79 GHz (24 CPUs)	NVIDIA Geforce GTX 1650 Super	Memoria 32 GB 64-bit LPDDR4 25.6 GB/s.	1 Tera

**Tabla 7:** Características del equipo de trabajo para el ambiente local.

Cabe recalcar que, inicialmente, se intentó realizar una ejecución de todo el proceso en el ambiente de la Nube, pero debido a la cantidad de imágenes, el proceso tomaba un tiempo muy alto y se presentaron errores de exceder el tiempo (*time out*), debido a que en Google Colab la demanda y asignación a aceleradores de hardware funciona en tiempo real, lo que produce que en ocasiones haya fluctuaciones en el acceso a las GPU<sup>11</sup> y TPU<sup>12</sup>.

Para el desarrollo de trabajo en el ambiente local, se trabajó con Python versión 3.7 y Jupyter Notebook, como entorno de trabajo. En los anexos se encuentra la lista de todas las librerías de Python instaladas para el desarrollo de este trabajo con su respectiva versión. Se proporciona esta información, ya que comúnmente se presentan errores de compilación por las versiones de las librerías. Sin embargo, en cada *notebook* se encuentra la instalación de cada librería, pero dado el caso que se presente un error, se puede verificar que tenga las mismas versiones.

En el entorno de trabajo local, se trabajaron dos *scripts* que encuentra en el anexo de tabla de productos generados ítem 5:

- *Image Resizing and Classification.ipynb*: Notebook para el redimensionamiento de las imágenes, y la clasificación de los conjuntos de datos de *Train*, *Validation* y *Test*.
- *CNN\_RestNet50\_PalmOil.ipynb*: Notebook que contiene la CNN del proyecto.

#### 4.3.2 Ambiente en la Nube

Para el ambiente en la Nube, se trabajó con 3 aplicaciones web: Por un lado, se tienen Google Earth y a Google Earth Engine, las cuales se nombraron en la sección anterior para definir tanto las zonas donde se encuentran los monocultivos,

<sup>11</sup> Unidad de procesamiento gráfico.

<sup>12</sup> Unidad de procesamiento tensorial.

como las que no, y extraer las imágenes satelitales. Por otro lado, se trabajó con la aplicación web de Google Colab, una herramienta que ha sido desarrollada por Google para otorgar acceso gratuito de GPU y TPU a cualquier persona.

Esta herramienta está diseñada específicamente para el desarrollo de aplicaciones de IA y análisis de datos. En esencia es un IDE que proporciona múltiples características entre las que se destacan:

- Ejecución de comandos de terminal.
- Importación de conjuntos de datos de fuentes externas como Kaggle.
- Guardar los documentos editados en Google Colab en Google Drive.
- Importar *notebook* desde Google Drive.
- Integración con Pythorch, Tensor Flow, Open CV para el desarrollo y análisis de datos de IA.
- Importación o publicación directa o desde GitHub.

En la Tabla 8 se encuentran las características del entorno de ejecución de Google Colab.

Sistema Operativo	CPU		GPU	RAM	Disco
Ubuntu Linux	Intel CPU GHz	Xeon 2.20	Nvidia Tesla T4	13 GB	Capacidad de almacenamiento de acuerdo con la configuración de Google Drive

**Tabla 8:** Características de entorno en la Nube Google Colab

En el entorno de trabajo de la Nube, se trabajó el *script* DataSetPalmOil.ipynb que se encuentra en el anexo de tabla de productos generados ítem 4. Este tiene como objetivo realizar la preparación de los datos teniendo las imágenes satelitales extraídas desde Google Earth Engine.



## **4.4 Entrenamiento del modelo**

En esta sección, se aborda el proceso de entrenamiento del modelo de detección de monocultivos de palma de aceite en Colombia utilizando redes neuronales convolucionales (CNN). Como se dio a conocer en el marco teórico, se ha seleccionado la arquitectura de la red convolucional ResNet50 debido a su destacado desempeño en investigaciones previas relacionadas con el análisis de imágenes satelitales.

ResNet50 es un modelo profundo que ha demostrado una notable capacidad para extraer características relevantes de imágenes complejas, lo cual resulta fundamental para la detección precisa de monocultivos de palma de aceite. Esta elección se basó en la confianza generada por los resultados prometedores obtenidos con esta arquitectura en investigaciones similares.

El entrenamiento del modelo se llevó a cabo utilizando un conjunto de datos etiquetados, donde cada muestra representa una región de interés en las imágenes satelitales. A través del proceso de entrenamiento, se espera que la CNN ResNet50 aprenda a reconocer patrones característicos de los monocultivos de palma de aceite, permitiendo así la detección automática de estas zonas a partir de nuevas imágenes satelitales. A lo largo de esta sección, se describen detalladamente los pasos y técnicas empleadas en el entrenamiento del modelo, incluyendo la configuración de los hiperparámetros, el proceso de optimización y la evaluación del desempeño. Con esto, se buscó establecer una base sólida para el desarrollo de un modelo preciso y confiable en la detección de monocultivos de palma de aceite en Colombia.

### **4.4.1 Preparación de los datos para el entrenamiento.**

La preparación adecuada de los datos es un paso crucial en el proceso de entrenamiento del modelo de detección de monocultivos de palma de aceite. Para ello, se contó con un conjunto de datos compuesto por dos clases principales: 'Palm Oil' (palma de aceite) y 'NOT Palm Oil' (no palma de aceite). Estos datos representan regiones de interés extraídas de imágenes satelitales que han sido previamente etiquetadas.

Con el objetivo de entrenar y evaluar el modelo de manera efectiva, se procedió a realizar la clasificación de los datos en conjuntos de entrenamiento y validación. El

conjunto de entrenamiento será utilizado para ajustar los parámetros del modelo, mientras que el conjunto de validación se utilizó para estimar su rendimiento y realizar ajustes adicionales, si es necesario.

La clasificación de los datos se llevó a cabo siguiendo una proporción adecuada para garantizar un entrenamiento robusto y una evaluación confiable del modelo. Se asignó una porción significativa de los datos, aproximadamente el 70%, al conjunto de entrenamiento, lo que permitirá al modelo aprender de manera amplia y capturar las características clave de los monocultivos de palma de aceite. El restante 20% de los datos se destinó al conjunto de validación, que servirá para medir la precisión y generalización del modelo en datos no vistos durante el entrenamiento.

Además de la clasificación de los datos, se realizaron otros procesos de preparación para asegurar la calidad y coherencia del conjunto de datos. Estos incluyeron la normalización de las imágenes, la eliminación de posibles ruidos o artefactos no deseados, y la segmentación precisa de las regiones de interés correspondientes a los monocultivos de palma de aceite.

En resumen, la preparación de los datos para el entrenamiento del modelo involucró una división de los datos adecuada en conjuntos de entrenamiento y validación, asegurando una proporción representativa de las clases 'Palm Oil' y 'NOT Palm Oil'. Estas etapas de preparación se realizaron con el objetivo de proporcionar al modelo una base sólida y confiable para su entrenamiento y evaluación.

#### 4.4.2 Entrenamiento del modelo de CNN ResNet50

En el *notebook* CNN\_ResNet50\_PalmOil que se encuentra relacionado en el anexo de productos generados ítem 5, se encuentra el proceso de construcción y entrenamiento del modelo que utiliza la arquitectura ResNet50 pre-entrenada. A continuación, se hace una descripción de este proceso:

En primer lugar, se cargó el modelo pre-entrenado ResNet50 utilizando la función `tf.keras.applications.ResNet50()`. Se establecieron los parámetros `include_top=False` para excluir las capas totalmente conectadas de la parte superior del modelo y `weights='imagenet'` para utilizar los pesos pre-entrenados en el conjunto de datos ImageNet. Además, se especificó `input_shape=(256, 256, 3)` para indicar el tamaño de las imágenes de entrada.

Después de cargar el modelo, se procede a congelar todas las capas base utilizando un bucle `for`. Esto impide que las capas base se modifiquen durante el entrenamiento, lo cual es útil cuando se utiliza un modelo pre-entrenado para tareas de transferencia de aprendizaje (*Transfer Learning*).

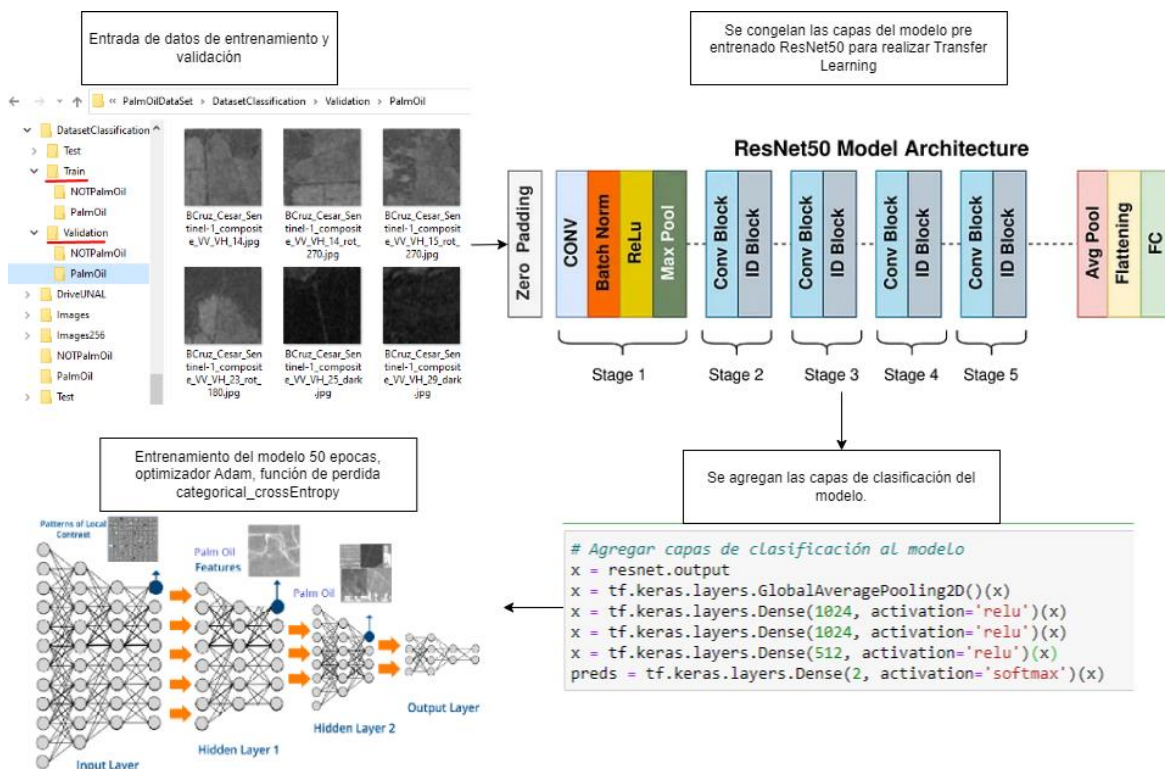
Posteriormente, se agregan capas de clasificación al modelo. Se tomó la salida de la última capa convolucional de ResNet50, se aplicó una capa de promediado global (`GlobalAveragePooling2D`) para reducir la dimensionalidad de los mapas de características y luego se añadieron varias capas densas (`Dense`) con activación ReLU para capturar patrones más complejos. Finalmente, se agregó una capa densa de salida con activación *softmax* para la clasificación binaria de las clases 'Palm Oil' y 'NOT Palm Oil'.

Una vez que se han agregado todas las capas al modelo, se creó el modelo final utilizando `tf.keras.models.Model` y se especificaron las entradas y salidas correspondientes.

A continuación, se compiló el modelo utilizando el optimizador '`adam`', la función de pérdida '`categorical_crossentropy`' y se eligió la métrica de precisión ('`accuracy`') para evaluar el rendimiento del modelo.

Finalmente, se realizó el entrenamiento del modelo utilizando el método `fit` de Keras. Se estableció el número de épocas (`epochs=50`) y se proporcionaron los generadores de datos de entrenamiento y validación previamente preparados (`train_generator` y `val_generator`, respectivamente). Durante el entrenamiento, se mostró información detallada del progreso (`verbose = 1`).

En la Figura 28 se encuentra el flujo de entrenamiento del modelo. En resumen, en este flujo se realizó una preparación de los datos de entrenamiento y validación, se cargó un modelo pre-entrenado ResNet50, donde se añadieron las capas de clasificación de las dos clases (Palm Oil y NOT Palm Oil). Se compiló el modelo con los parámetros necesarios y se entrenó utilizando los generadores de datos. Este proceso permite utilizar la transferencia de aprendizaje para adaptar ResNet50 a la detección de monocultivos de palma de aceite en Colombia.



**Figura 28:** Flujo de entrenamiento de modelo de reconocimiento de cultivos de palma de aceite.

## 4.5 Evaluación del modelo

En la presente sección, se realizó una evaluación exhaustiva del modelo de detección de monocultivos de palma de aceite en Colombia. Para ello se presentan dos elementos clave que brindan una visión completa del desempeño del modelo: las gráficas de desempeño y la matriz de confusión.

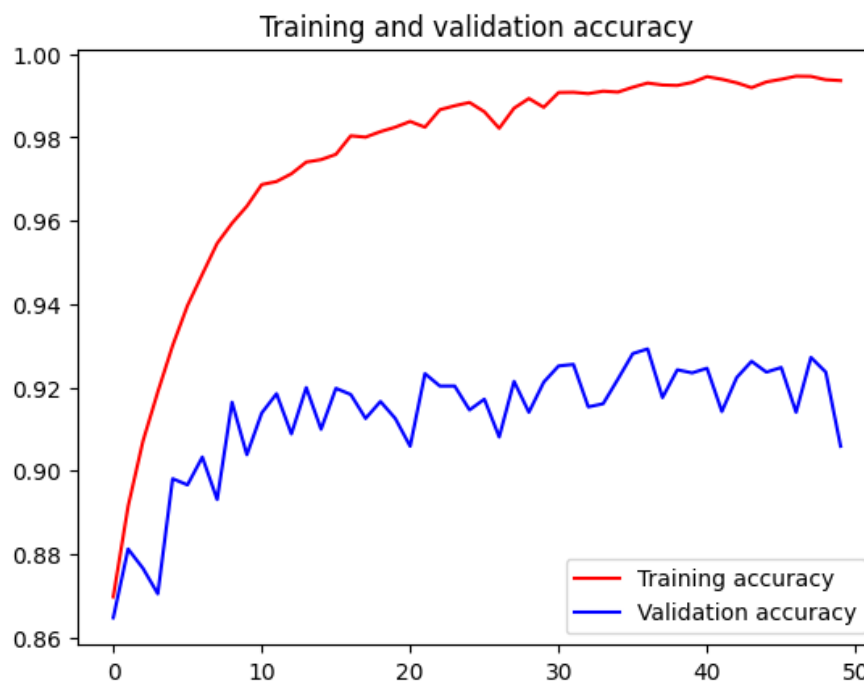
Inicialmente, como exploración, se tomó un modelo inicial, con el objetivo de observar su comportamiento inicial en la detección de monocultivos de palma de aceite. Posteriormente, se aplicó una técnica de validación cruzada (*cross-validation*) con un valor de  $k=4$ , dividiendo el conjunto de datos en cuatro particiones aleatorias para evaluar la capacidad de generalización del modelo. Para el entrenamiento del modelo en cada una de las iteraciones se ajustó el parámetro de número épocas, y se determinó el mejor modelo con base en su desempeño y métricas de evaluación.

### 4.5.1 Gráficas de desempeño modelo inicial

Las gráficas de desempeño muestran la evolución de la presión y pérdida del modelo durante el entrenamiento y la validación a lo largo de las diferentes épocas que se definieron (`epochs=50`). Este proceso de entrenamiento se trabajó en el entorno local y tomó un tiempo de alrededor de 22 horas en el hardware disponible.

En la Figura 29 se muestra la gráfica de desempeño de *Training* y *Validation Accuracy*, que proporciona una visión detallada del rendimiento del modelo a lo largo del proceso de entrenamiento. En el caso de los datos de entrenamiento, se observa una tendencia ascendente que se acerca gradualmente al valor máximo de 1. Esto indica que el modelo ha aprendido eficientemente los patrones y características necesarios para clasificar correctamente los monocultivos de palma de aceite en el conjunto de entrenamiento.

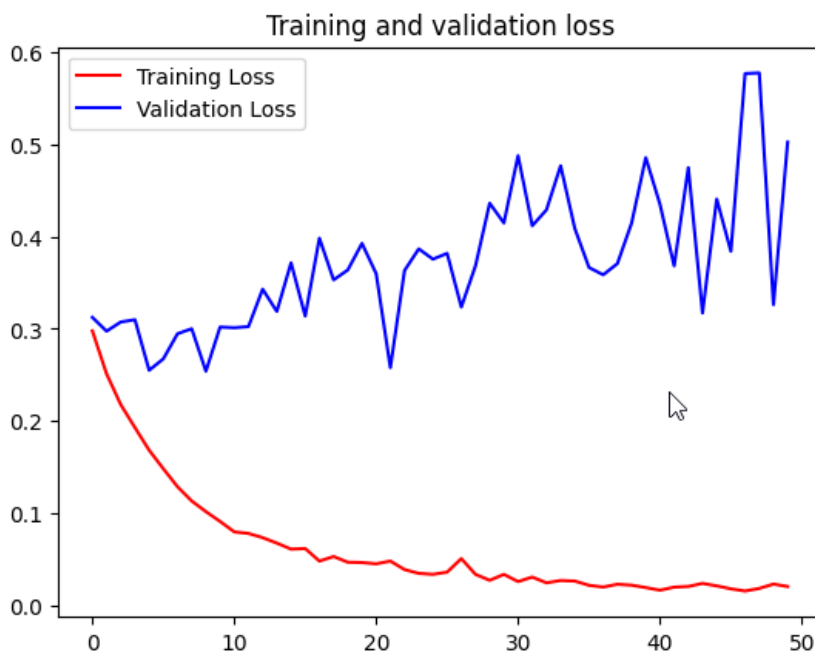
Es importante destacar que el punto de partida de ambas gráficas de desempeño se encuentra en un valor inicial de 0.86, debido a la transferencia de aprendizaje realizada a partir del modelo de CNN ResNet50 pre-entrenado. Esta transferencia de conocimiento inicial permite al modelo comenzar con un rendimiento prometedor desde el inicio del entrenamiento, acelerando el proceso de convergencia hacia una precisión alta y consistente durante el entrenamiento y la validación.



**Figura 29** Gráfica de desempeño: *Training and Validation accuracy* a través de las épocas

Por otro lado, los datos de validación también exhiben una tendencia creciente, aunque se estabilizan alrededor del valor de 0.94 a lo largo de las 50 épocas. Esta estabilización indica que el modelo ha alcanzado una capacidad de generalización sólida, siendo capaz de clasificar de manera precisa los monocultivos de palma de aceite en nuevas imágenes.

Por otro lado, en la Figura 30 se presenta la gráfica de desempeño de *Training y Validation Loss* que muestra la evolución de la pérdida del modelo durante el proceso de entrenamiento. Tanto para los datos de entrenamiento como para los de validación, se parte de un valor inicial de 0.3. En el caso de los datos de entrenamiento, se observó una tendencia descendente a lo largo de las 50 épocas, acercándose cada vez más a 0. Esta disminución de la pérdida indica que el modelo está aprendiendo de manera efectiva a clasificar correctamente los monocultivos de palma de aceite en el conjunto de entrenamiento.



**Figura 30** Gráfica de desempeño: *Training and validation loss* a través de las épocas.

Sin embargo, para los datos de validación, la gráfica muestra una oscilación y un aumento máximo hasta 0.6. A lo largo de las épocas, el valor de la pérdida (*loss*) en los datos de validación fluctúa entre el valor inicial de 0.3 y 0.5, con un máximo de 0.6.

Estas oscilaciones indican que el modelo puede enfrentar ciertas dificultades para generalizar en nuevas imágenes y clasificarlas con la misma precisión que en los datos de entrenamiento. A pesar de estas fluctuaciones, es importante destacar que el valor de la función de pérdida de validación se mantuvo en un rango relativamente bajo, lo que indica que el modelo está logrando una buena capacidad de generalización y evitando un sobreajuste. Aunque se alcanzó un máximo de 0.6 en un punto, el valor de la pérdida de validación se mantuvo en un nivel aceptable a lo largo del proceso de entrenamiento, lo que indica una estabilidad en el desempeño del modelo.

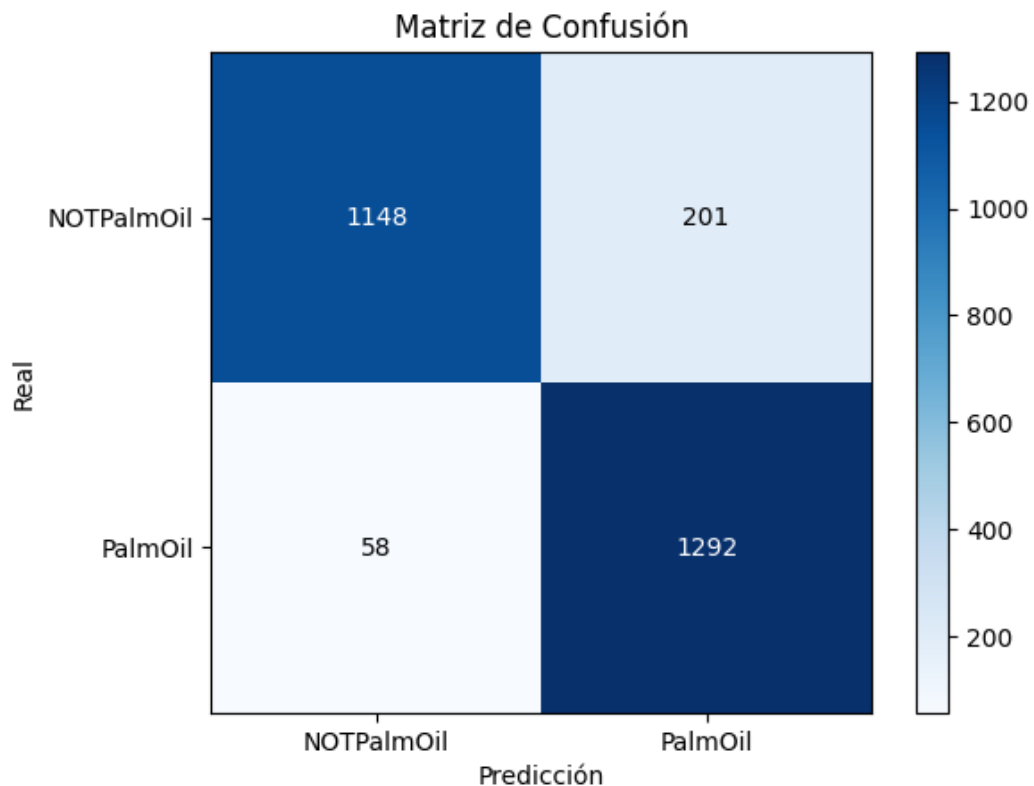
#### 4.5.2 Matriz de confusión de los datos de prueba para el modelo inicial.

La matriz de confusión muestra la distribución de las predicciones del modelo en función de las clases reales. Para la generación de esta matriz se tomó el modelo

ya entrenado y el conjunto de datos de prueba (*Test*), el cual contiene un total de 1.350 imágenes por cada clase.

En este caso, la matriz de confusión presenta los resultados de la clasificación de las clases **PalmOil** (cultivos de palma de aceite) y **NOTPalmOil** (No cultivos de palma de aceite). La matriz muestra el número de muestras clasificadas erróneamente. Esto permite evaluar la capacidad del modelo para distinguir entre las clases y proporciona información sobre los posibles errores de clasificación cometidos.

En la Figura 31 se muestra la matriz de confusión del conjunto de datos de prueba, la cual evalúa el rendimiento del modelo entrenado y analiza los diferentes tipos de aciertos y errores en la clasificación.



**Figura 31** Matriz de confusión del conjunto de datos de prueba.

En la diagonal principal de la matriz se encuentran los verdaderos positivos (TP) y los verdaderos negativos (TN), que representan las clasificaciones correctas realizadas por el modelo [64]. En este caso, se observó que se clasificaron correctamente 1148 muestras en la clase '**NotPalmOil**' y 1292 muestras en la clase '**PalmOil**'. Estos valores indican la capacidad del modelo para identificar correctamente los monocultivos de palma de aceite y las muestras que no corresponden a palma de aceite.



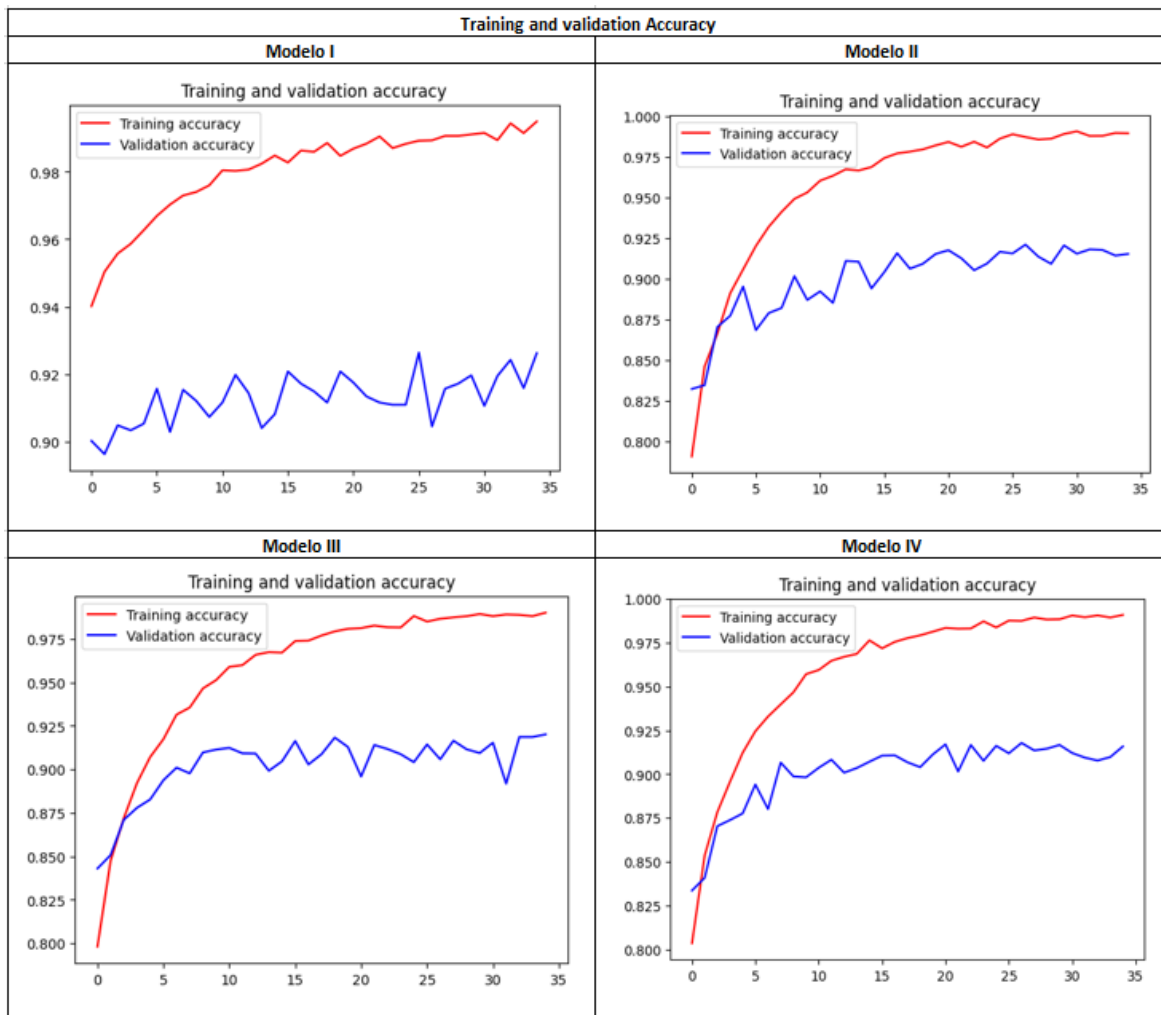
Por otro lado, fuera de la diagonal principal, se encuentran los falsos positivos (FP) y los falsos negativos (FN), que representan los errores de clasificación cometidos por el modelo. En este caso, se observan 201 falsos positivos, lo que significa que se clasificaron erróneamente muestras como 'NotPalmOil' cuando en realidad eran 'PalmOil'. Además, se registraron 58 falsos negativos, lo que indica que se clasificaron incorrectamente muestras de 'PalmOil' como 'NotPalmOil'.

La matriz de confusión revela que el modelo logró clasificar correctamente la mayoría de las muestras en ambas clases, obteniendo un alto número de verdaderos positivos y verdaderos negativos. Sin embargo, se cometieron algunos errores de clasificación, representados por los falsos positivos y falsos negativos. Estos resultados brindan información valiosa sobre el rendimiento del modelo en la detección de monocultivos de palma de aceite, y pueden servir como base para realizar ajustes y mejoras en el modelo con el objetivo de reducir los errores de clasificación y mejorar su desempeño en futuras aplicaciones.

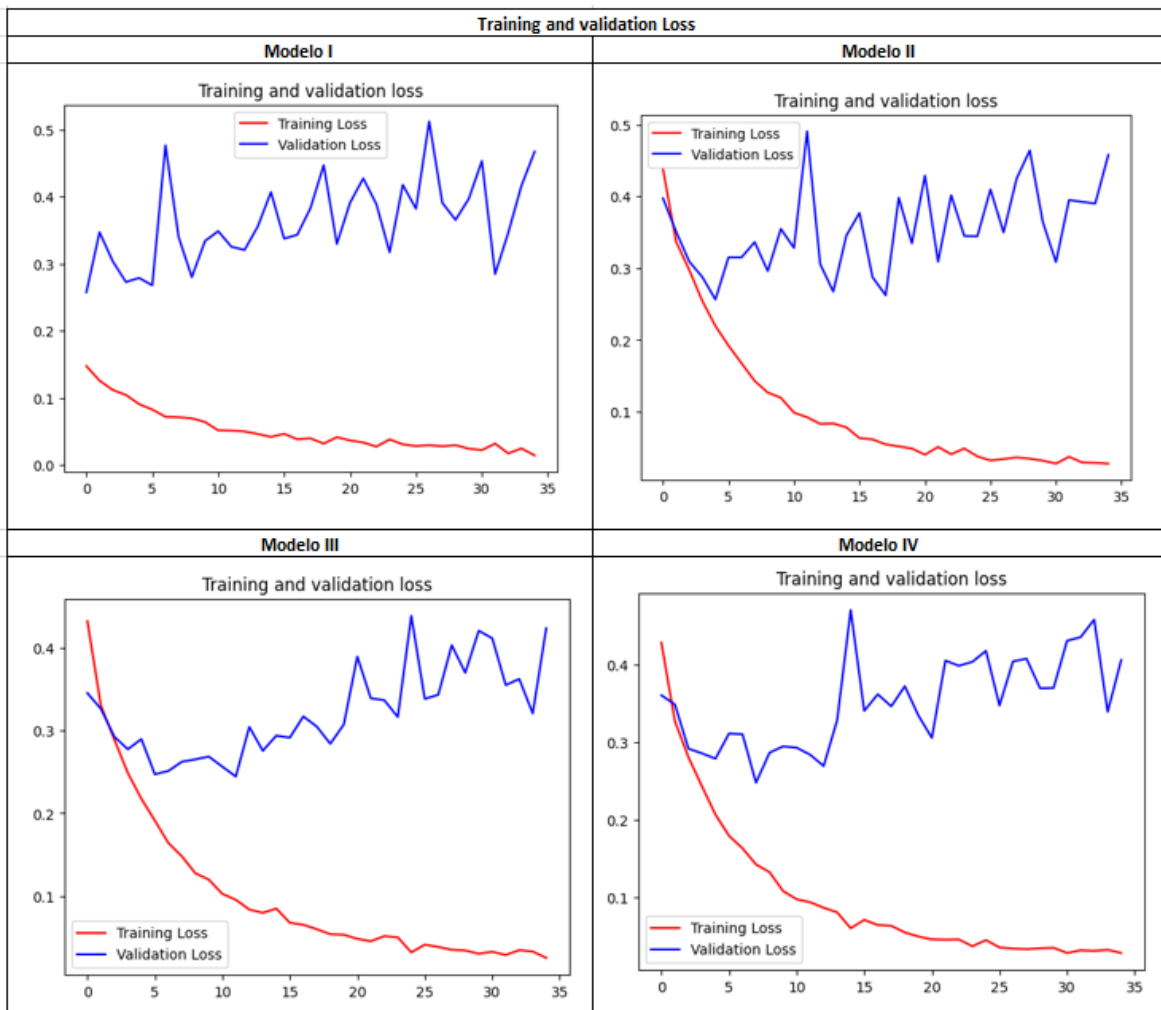
#### **4.5.3 Entrenamiento implementando *Cross-Validation***

Después de tener el modelo inicial, y al realizar un análisis tanto de la matriz de confusión con los datos de pruebas como la gráfica de desempeño, se procedió a realizar nuevamente un entrenamiento implementando la técnica *cross-Validation* (validación cruzada) aleatoria con un valor de  $k=4$ , es decir, para obtener 4 grupos (pliegues), y se decidió ajustar el parámetro a 35 épocas para optimizar el desempeño del modelo.

En las Figuras 32 y 33 se presentan las gráficas de desempeño de *accuracy* y *loss* con los datos de entrenamiento y validación de cada grupo de datos obtenidos por la validación cruzada, para detectar monocultivos de palma de aceite en imágenes satelitales.



**Figura 32** Gráfica de desempeño de *accuracy* con los datos de entrenamiento y validación de cada uno de los conjuntos de datos obtenidos de *cross-validation*.

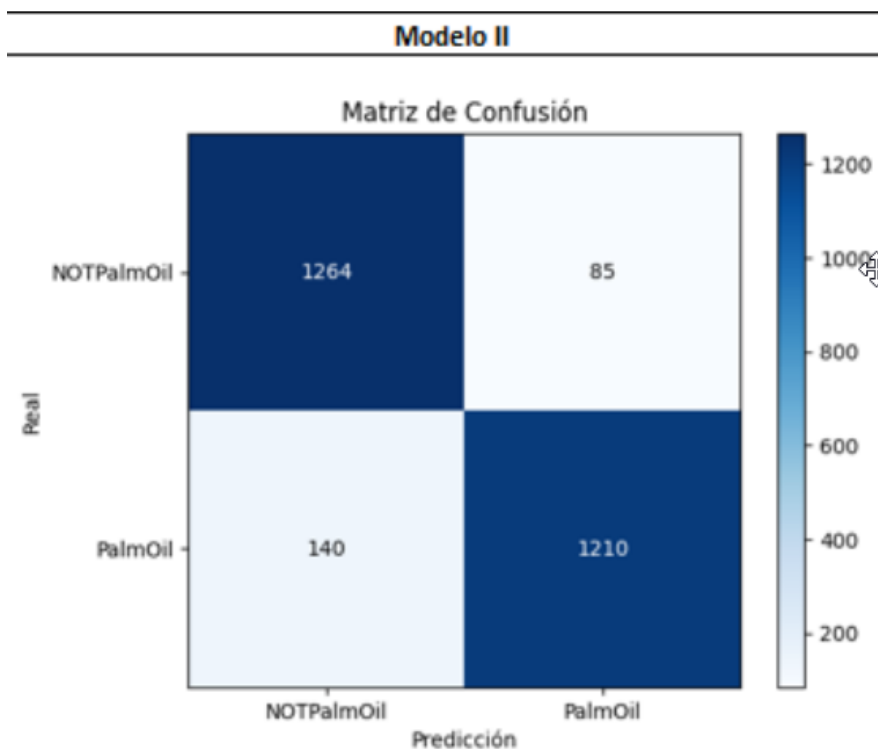


**Figura 33** Gráfica de desempeño de la pérdida (*loss*) con los datos de entrenamiento y validación de cada uno de los conjuntos de datos obtenidos de *cross-validation*.

#### 4.5.4 Selección de un modelo proporcionado por el (*cross-validation*)

Con los modelos obtenidos en el entrenamiento de los conjuntos de datos del punto anterior, se procedió a generar la matriz de confusión para los datos de prueba. Para la elección de uno de los modelos proporcionados por *cross-validation*, se procedió a calcular la métrica de exactitud (*accuracy*), que representa el porcentaje de clasificaciones correctas sobre el total de clasificaciones.

Tras calcular el indicador de exactitud (*accuracy*) para cada uno de los modelos, se determinó que uno de los modelos presentó un rendimiento cercano al 92% en los datos de prueba, mientras que los otros modelos se situaban alrededor del 91%. En la Figura 34 se encuentra la matriz de confusión del modelo seleccionado.



**Figura 34** Matriz de confusión del modelo seleccionado.

Al realizar una comparación con el modelo inicial, se evidencia como la aplicación de la técnica de validación cruzada, demostró ser fundamental para mejorar los resultados obtenidos en el modelo inicial. Esta técnica permitió una evaluación más robusta y precisa del modelo al dividir el conjunto de datos en 4 particiones aleatorias, asegurando que todas las muestras sean utilizadas tanto para entrenamiento como para la evaluación. Al utilizar  $k=4$  en la validación cruzada, se garantizó una evaluación del modelo en diferentes combinaciones de conjuntos de entrenamiento, validación y prueba.

El modelo seleccionado se utilizará para la implementación de datos de pruebas con imágenes externas en el siguiente capítulo, con el objetivo de detectar monocultivos de palma de aceite en imágenes satelitales.

## 5.Resultados

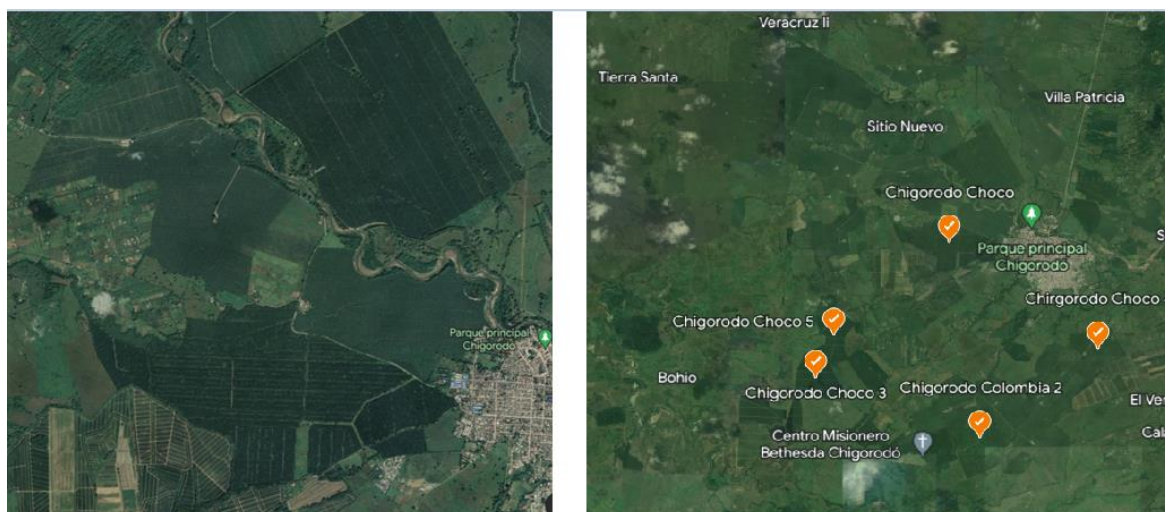
Para la presentación de resultados del modelo elegido, se seleccionaron imágenes de monocultivos de palma de aceite de zonas (departamentos) de Colombia externos a las zonas con que se entrenó el modelo. Para la obtención de las imágenes se aplicó la misma estrategia descrita en la sección de preparación de datos.

El conjunto de datos abarca una diversidad geográfica y temporal, proporcionando una representación exhaustiva de los monocultivos de palma de aceite en Colombia, externo a los datos con los que se entrenó el modelo.

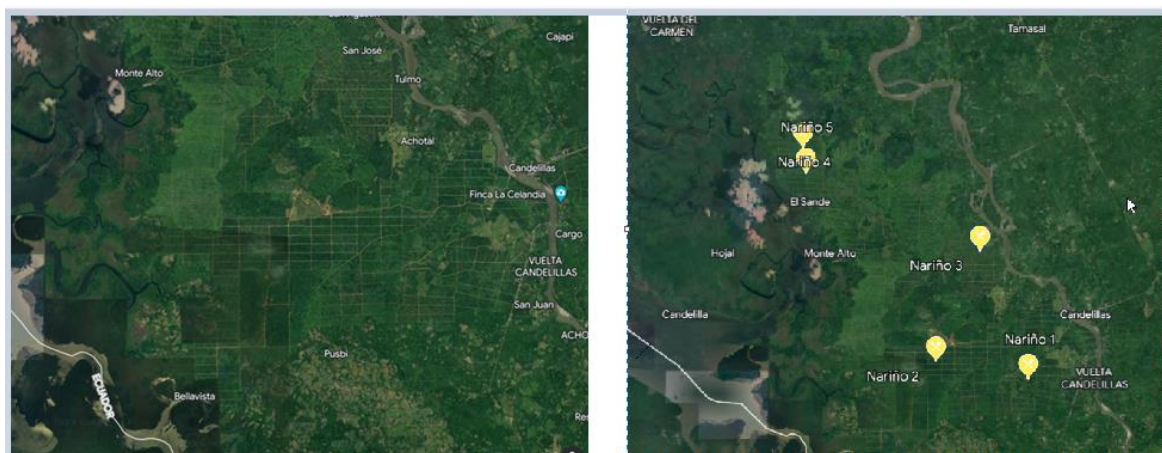
Posteriormente, se procedió a realizar la evaluación del modelo seleccionado utilizado en conjunto de datos recolectado utilizando métricas de evaluación de clasificadores binarios.

### 5.1 Conjunto de imágenes externas de pruebas

Para la selección de imágenes externas de prueba del modelo se extrajeron imágenes de cultivos de palma de aceite del departamento de Chocó, adyacentes al municipio de Chigorodó (Figura 35) y del departamento de Nariño (Figura 36). Las ubicaciones de estos monocultivos se encuentran en el anexo de tabla de productos generados ítem 2, archivo de Google Earth: *Cultivos Palma de Aceite externas al Modelo*.



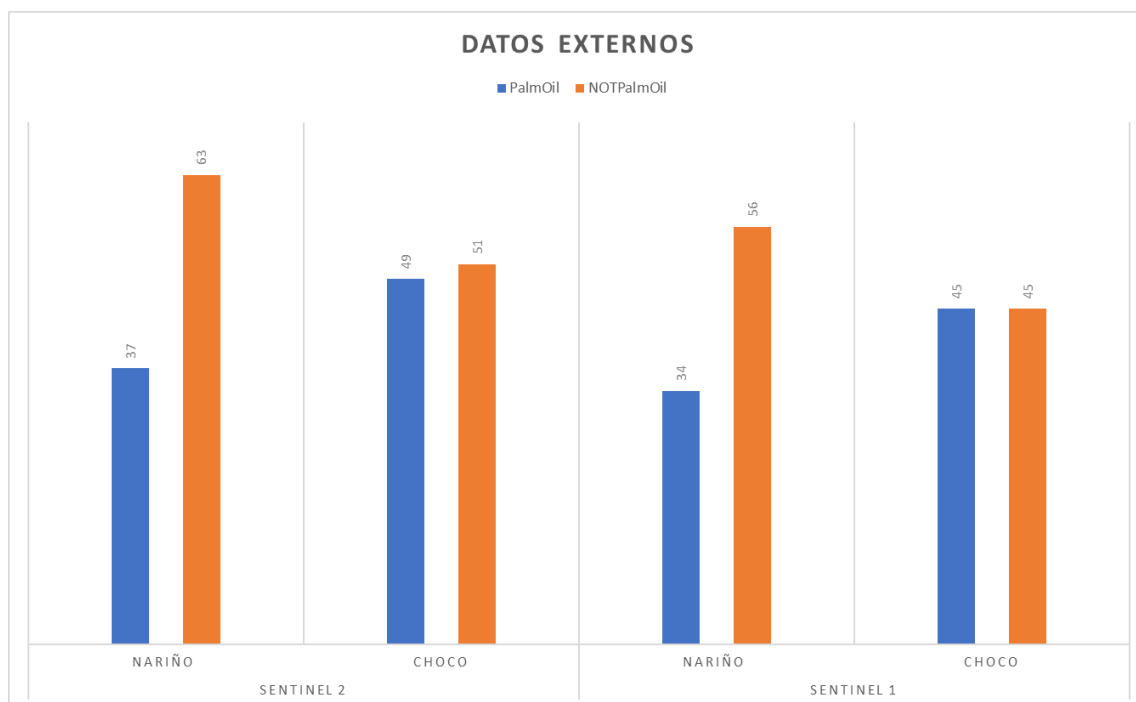
**Figura 35** Cultivos de palma de aceite cerca del municipio de Chigorodó, Chocó.



**Figura 36** Cultivos de palma de aceite en Nariño

Para la extracción de estas imágenes se utilizó la misma estrategia del capítulo de preparación de datos, tomando un área rectangular de 4 km x 4 km en cada una centrada en el punto de intereses (ubicación), y extrayendo de los satélites Sentinel-1 y Sentinel-2. Posteriormente, se procedió en cada departamento a realizar una división de 100 imágenes, de tal forma que cada imagen representa un área de 1600 metros cuadrados (40 m x 40 m).

Luego de realizar una limpieza de los datos, en la Figura 37 se muestra la cantidad de datos etiquetados (PalmOil o NOTPalmOil) para los departamentos de Nariño y Chocó, de las imágenes extraídas de los satélites Sentinel-1 y Sentinel-2.



**Figura 37** Clasificación de datos externos para los departamentos de Nariño y Chocó, con los satélites Sentinel-1 y Sentinel-2.

Estas imágenes permitirán evaluar la capacidad de generalización y robustez del modelo seleccionado entrenado, y evaluar el modelo con imágenes externas.

## 5.2 Evaluación del modelo en las imágenes externas

Una vez que se clasificaron las imágenes externas por departamento y satélite, se llevó a cabo el procesamiento utilizando el modelo seleccionado para realizar la predicción en cada conjunto de datos. En la Tabla 9 se encuentran los resultados obtenidos de una manera organizada por departamento y satélite.

Sentinel 1							
Fold1							
Choco				Nariño			
PalmOil		NOTPalmOil		PalmOil		NOTPalmOil	
35	78%	10	22%	27	79%	7	21%
6	13%	39	87%	1	2%	55	98%

Sentinel 2							
Fold1							
Choco				Nariño			
PalmOil		NOTPalmOil		PalmOil		NOTPalmOil	
42	86%	7	14%	33	89%	4	11%
5	10%	46	90%	4	6%	59	94%

**Tabla 9:** Resultados de predicción del modelo seleccionado con los datos externos clasificados por departamento y satélite.

Se evidencia que la predicción, en general, fue buena, pero se observó un notable aumento en el desempeño al trabajar con las imágenes del satélite Sentinel-2. La mejora en la predicción de las imágenes del satélite Sentinel-2 para ambos departamentos se atribuye principalmente a la capacidad de este satélite para capturar y resaltar con mayor precisión las texturas presentes en los monocultivos de palma de aceite, tales como: texturas, estructuras de las hojas y las características del suelo.

Para realizar una evaluación del modelo con las imágenes externas más profunda, se procedió a unir las imágenes, tanto de los dos departamentos, como de los dos satélites. Posteriormente, se procedió a realizar una evaluación de clasificadores binarios, obteniendo los resultados mostrados en la Figura 38.

	precision	recall	f1-score	support
0	0.91	0.92	0.92	215
1	0.92	0.91	0.91	165
accuracy			0.92	380
macro avg	0.92	0.92	0.92	380
weighted avg	0.92	0.92	0.92	380

**Figura 38** Evaluación de clasificadores binarios del modelo seleccionados con las imágenes externas.

A continuación, se describen los resultados obtenidos:



- *Precision* (Precisión): Es la proporción de predicciones correctas entre las etiquetas predichas como positivas. Para nuestro caso, la predicción para la clase 0 (NOTPalmOil) es del 91%, lo que indica que el 91% de las predicciones positivas para esta clase fueron correctas. Para la clase 1 (PalmOil), la precisión es del 92%, lo que significa que el 92% de las predicciones positivas para esta clase fueron correctas.
- *Recall* (Exhaustividad): Es la proporción de instancias positivas que se han identificado correctamente. Para nuestro caso, el *recall* para la clase 0 (NOTPalmOil) es del 92%, lo que indica que se identificó correctamente el 92% de las instancias reales de no monocultivo de palma de aceite. Para la clase 1 (PalmOil), el *recall* es del 91%, lo que significa que se identificó correctamente el 91% de las instancias reales de monocultivo de palma de aceite.
- *F1-score*: Es una medida que combina la *precision* y el *recall* en una sola métrica. Es útil cuando hay un desequilibrio entre las clases. Para nuestro caso, el *F1-score* para la clase 0 (NOTPalmOil) es de 0.92, mientras que para la clase 1 (PalmOil) es de 0.91. Un *F1-score* más cercano a 1 indica un mejor rendimiento en términos de *precision* y *recall* equilibrados.
- *Accuracy* (Exactitud): Es la proporción de predicciones correctas en general. En nuestro caso, la exactitud es del 92%, lo que significa que el 92% de las predicciones realizadas fueron correctas.
- *Macro avg*: Es la media aritmética de las métricas (*precision*, *recall* y *F1-score*) calculadas para cada clase por separado. Para nuestro caso, el *macro avg* tiene un valor de 0.92 para la *precision*, *recall* y *F1-score*. Esto indica que, en promedio, el modelo tiene un desempeño equilibrado y consistente en todas las clases.
- *Weighted avg*: Es una media ponderada de las métricas calculadas para cada clase, donde la ponderación se realiza en función del soporte (número de muestras) de cada clase. Para nuestro caso, el *weighted avg* también tiene un valor de 0.92 para la *precision*, *recall* y *F1-score*. Esto indica que el modelo tiene un desempeño equilibrado, teniendo en cuenta el número de muestras de cada clase en la evaluación.

En resumen, los resultados indican que el modelo tiene un buen desempeño en la clasificación de los monocultivos de palma de aceite y los no monocultivos. La *precision*, el *recall* y el *F1-score* son altos para ambas clases, lo que sugiere que el modelo es capaz de realizar predicciones precisas y abarcar una gran proporción de las instancias reales de cada clase. La exactitud general del modelo también es alta, lo que indica un buen rendimiento en la clasificación global de las imágenes de prueba.

## 6. Conclusiones

En este capítulo, se presentan las conclusiones obtenidas a partir del desarrollo de la investigación sobre la detección de monocultivos de palma de aceite en Colombia mediante el uso de redes neuronales convolucionales, específicamente, la Resnet50.

A lo largo de este trabajo, se ha abordado el desafío de identificar y clasificar los monocultivos de palma de aceite a partir de imágenes satelitales, con el objetivo de brindar una herramienta eficiente para monitorear y gestionar estos cultivos a gran escala.

### 6.1 Conclusiones

A continuación, se encuentran los hallazgos y logros más relevantes derivados de la investigación. Se analizan los resultados obtenidos al implementar el modelo de CNN Resnet50 en la detección de los monocultivos de palma de aceite, evaluando su precisión y rendimiento. Además, se discuten las limitaciones y desafíos encontrados durante el desarrollo del proyecto, así como las soluciones propuestas:

- Para el desarrollo de esta investigación, dado que no se tomó un conjunto de datos de otros trabajos preexistente en otras investigaciones, por medio de la metodología SEMMA se recopiló imágenes satelitales de cultivos de palma de aceite con presencia en departamentos de Colombia, lo que permitió tener una variedad de datos que representan diversas características de estos.
- Durante la etapa de exploración y limpieza de las imágenes, se llevó a cabo un exhaustivo análisis visual y estadístico para comprender las particularidades de los monocultivos de palma de aceite y los desafíos para el modelo asociados a su detección. Este proceso fue fundamental para identificar las características clave y las posibilidades que deberían abordarse durante el entrenamiento del modelo.

- Cuando se trabaja con imágenes satelitales, dada la alta resolución de estas imágenes, establecer estrategias como fraccionar la imagen original, y estandarizar su tamaño para el entrenamiento, fue una buena decisión para la etapa de entrenamiento del modelo.
- Los resultados obtenidos mediante la implementación del modelo de CNN ResNet-50, siendo esta uno de los modelos que se han usado en imágenes satelitales, al ajustar adecuadamente las épocas de entrenamiento e implementar técnicas como *cross-validation*, permitió obtener modelos que demostraron ser altamente prometedores en la detección de monocultivos de palma de aceite en Colombia.
- Los resultados del modelo fueron prometedores y abren la puerta a diversas aplicaciones prácticas en la gestión y monitoreo de los monocultivos de palma de aceite en Colombia. La capacidad de identificar y clasificar de manera precisa estas áreas permite un mejor control y una toma de decisiones más informada para garantizar la sostenibilidad y mitigar los impactos ambientales asociados.
- Los resultados obtenidos del informe de clasificación (del modelo seleccionado en las imágenes externas) demuestran un buen desempeño del modelo en la detección y clasificación de los monocultivos de palma de aceite y los no monocultivos. Con una *precisión*, *recall* y *F1-score* de aproximadamente 0.92 para ambas clases, el modelo ha logrado realizar predicciones precisas y capturar la mayoría de las instancias reales de cada clase.

## 6.2 Recomendaciones

A continuación, se ofrecen recomendaciones basadas en los conocimientos adquiridos y los resultados obtenidos. Estas recomendaciones están dirigidas a los profesionales, investigadores y actores involucrados que deseen realizar alguna investigación similar o continuar con futuros trabajos de esta. Se presentan sugerencias prácticas para mejorar la precisión del modelo entrenado, optimizar el uso de recursos y fortalecer estrategias a la hora de trabajar con estos modelos y datos:

- Antes de entrenar un modelo de CNN, después de realizar una estandarización, la división adecuada de los datos en conjuntos de entrenamiento (*train*), validación (*validation*) y prueba (*test*) fue un factor clave para evaluar de manera objetiva la capacidad del modelo de generalizar y realizar predicciones precisas en nuevas imágenes.

- Una recomendación importante a la hora de trabajar con imágenes satelitales es aplicar algoritmos de limpieza de ruido y resaltar las texturas en todas las imágenes que se van a pasar al modelo. Este proceso puede ayudar a eliminar artefactos no deseados (para esta investigación el ruido de las nubes) y resaltar características relevantes en las imágenes, lo que mejorará la precisión de los modelos de CNN.
- Cuando se realizó el enriquecimiento de datos, inicialmente, se estandarizó un tamaño de imágenes de 80x80 píxeles, con el objetivo de que la etapa de entrenamiento fuera más eficiente en tiempo. Sin embargo, al aumentar este tamaño a 256x256 píxeles, a pesar de que el entrenamiento tomó más tiempo, los resultados fueron mejores.
- Contar con un equipo de cómputo con características adecuadas para entrenar un modelo de CNN es importante. Algunos aspectos por considerar son: Potencia de cálculo, GPU, Memoria RAM, Almacenamiento.
- Se recomienda utilizar *transfer learning* en lugar de comenzar desde cero al entrenar un modelo de CNN. Esta estrategia aprovecha el conocimiento previo de modelos pre-entrenados, reduce la necesidad de datos masivos, mejora la capacidad de generalización del modelo y ahorra tiempo y recursos computacionales.

### 6.3 Futuros trabajos.

En los posibles trabajos futuros se plantean líneas de investigación y desarrollos que surgen a partir de este trabajo. Se identifican áreas de mejora y oportunidades para expandir y profundizar el enfoque de detección de monocultivos de palma de aceite. Se proponen nuevas direcciones, como la integración de tecnologías complementarias, la exploración de otros conjuntos de datos y la ampliación del modelo en diferentes regiones geográficas. A continuación, se presentan algunas propuestas:

- Una posible propuesta sería seguir los pasos de este trabajo para explorar terrenos adicionales para obtener imágenes con más características, sin limitarse en usar solo un satélite, si no a la utilización de una variedad de satélites (Landsat, Modis, entre otros). Cada uno de estos satélites tiene características únicas y proporciona información complementaria sobre las imágenes.

- Otra investigación prometedora es el uso de técnicas de fusión de datos, que permiten combinar y aprovechar la información proveniente de diferentes fuentes, como imágenes satelitales, datos climáticos o información de sensores terrestres, entre otros. La fusión de datos puede enriquecer aún más la calidad de los datos utilizados en el entrenamiento del modelo y proporcionar una visión más completa de los monocultivos de palma de aceite.
- Una propuesta de investigación futura es aplicar el enfoque desarrollado en este trabajo a otros tipos de monocultivos y problemáticas relacionadas con el uso del suelo como:
  - Detección de otros monocultivos: Utilizar el mismo enfoque para monitorear monocultivos de maíz, caña de azúcar, arroz, entre otros. Incluso en Colombia se tiene la problemática de cultivos ilícitos. Esto permitiría desarrollar herramientas y modelos específicos para diferentes cultivos agrícolas.
  - Análisis de la deforestación: Aplicar un enfoque de detección basado en CNN para analizar y monitorear la deforestación en diferentes regiones, permitiría identificar áreas de deforestación, evaluar su magnitud y velocidad, y proporcionar información valiosa para la toma de decisiones en términos de conservación y protección de los bosques.
  - Monitoreo de actividades mineras: Utilizar el mismo enfoque para analizar y monitorear la actividad minera y sus impactos en el medio ambiente. La detección de áreas afectadas por la minería ilegal o la evaluación de la expansión de las actividades mineras puede proporcionar información crucial para la gestión y mitigación de los impactos ambientales asociados a la industria.
  - Estudio de cambio de suelo: Utilizar el enfoque desarrollado para analizar y monitorear cambios en el uso del suelo en general. Esto incluye la identificación y seguimiento de cambios en la cobertura del suelo, la evaluación de la expansión urbana, la detección de áreas de degradación del suelo y otros cambios relacionados con el uso del suelo.

Para llevar a cabo cualquier de estas investigaciones el eje central se encuentra en la obtención y calidad de los datos para la investigación, si no se tiene una cantidad significativa de los datos de buena calidad, para el modelo será muy difícil que tenga unos resultados esperados. Este trabajo entrega una ruta de trabajo para obtención de los datos basados en imágenes satelitales.

## • Anexo: Tabla de productos generados

A continuación, se relacionan los enlaces de los diferentes productos generados en este trabajo.

Ítem	Entorno	Producto	Enlace de Acceso
1	Zenodo	Repositorio de imágenes	<a href="https://s/zenodo.org/record/7979649">https://s/zenodo.org/record/7979649</a>
		Modelo entrenado ResNet50	
2	Google Earth	DataSet Zonas de Cultivos de Palma de Aceite Colombia.kml	<a href="https://github.com/jcarvajalri/PalmOil_CNN_UNAL/tree/main/Google%20Earth">https://github.com/jcarvajalri/PalmOil_CNN_UNAL/tree/main/Google%20Earth</a>
		DataSet Zonas adyacentes Cultivos de Palma de aceite en Colombia.kml	
		Cultivos Palma de Aceite externas al Modelo.kml	
3	Google Earth Engine	Donwload_S1_S2_Transormation.js	<a href="https://github.com/jcarvajalri/PalmOil_CNN_UNAL/tree/main/Googler%20Earth%20Engine">https://github.com/jcarvajalri/PalmOil_CNN_UNAL/tree/main/Googler%20Earth%20Engine</a>
		ExternallImages.js	
4	Colab	DataSetPalmOil.ipynb	<a href="https://github.com/jcarvajalri/PalmOil_CNN_UNAL/tree/main/Colab">https://github.com/jcarvajalri/PalmOil_CNN_UNAL/tree/main/Colab</a>
5	Python Local	Image Resizing and Classification.ipynb	<a href="https://github.com/jcarvajalri/PalmOil_CNN_UNAL/tree/main/Local">https://github.com/jcarvajalri/PalmOil_CNN_UNAL/tree/main/Local</a>
		CNN_RestNet50_PalmOil.ipynb Image Resizing and Classification-cross-validation.ipynb	

En el siguiente Anexo se encuentran la lista de librerías de Python que se utilizaron en el desarrollo de este trabajo con sus respectivas versiones:

absl-py==1.4.0	jupyter-server==1.24.0	QtPy==2.3.1
anyio==3.6.2	jupyter_client==7.4.9	requests==2.29.0
argon2-cffi==21.3.0	jupyter_core==4.12.0	requests-oauthlib==1.3.1
argon2-cffi- bindings==21.2.0	jupyterlab-pygments==0.2.2	rsa==4.9
astunparse==1.6.3	jupyterlab-widgets==3.0.7	scikit-learn==1.0.2
attrs==23.1.0	keras==2.7.0	sipo==1.7.3
backcall==0.2.0	Keras-Preprocessing==1.1.2	seaborn==0.12.2
beautifulsoup4==4.12.2	kiwisolver==1.4.4	Send2Trash==1.8.2
bleach==6.0.0	libclang==16.0.0	six==1.16.0
cachetools==5.3.0	Markdown==3.4.3	sniffio==1.3.0
certifi==2022.12.7	MarkupSafe==2.1.2	soupsieve==2.4.1
cffi==1.15.1	matplotlib==3.5.3	tensorboard==2.11.2
charset-normalizer==3.1.0	matplotlib-inline==0.1.6	tensorboard-data-server==0.6.1
colorama==0.4.6	mistune==2.0.5	tensorboard-plugin-wit==1.8.1
cycler==0.11.0	mlxtend==0.22.0	tensorflow==2.7.0
		tensorflow-io-gcs- filesystem==0.31.0
debugpy==1.6.7	nbclassic==0.5.6	termcolor==2.3.0
decorator==5.1.1	nbclient==0.7.4	terminado==0.17.1
defusedxml==0.7.1	nbconvert==7.3.1	threadpoolctl==3.1.0
entrypoints==0.4	nbformat==5.8.0	tinycss2==1.2.1
fastjsonschema==2.16.3	nest-asyncio==1.5.6	tornado==6.2
flatbuffers==2.0.7	notebook==6.5.4	traitlets==5.9.0
fonttools==4.38.0	notebook_shim==0.2.3	typing_extensions==4.5.0
gast==0.4.0	numpy==1.21.6	urllib3==1.26.15
google-auth==2.17.3	oauthlib==3.2.2	wcwidth==0.2.6
google-auth-oauthlib==0.4.6	opencv-python==4.7.0.72	webencodings==0.5.1
google-pasta==0.2.0	opt-einsum==3.3.0	websocket-client==1.5.1
grpcio==1.54.0	packaging==23.1	Werkzeug==2.2.3
h5py==3.8.0	pandas==1.3.5	widetsnbextension==4.0.7
idna==3.4	pandocfilters==1.5.0	zipp==3.15.0
importlib-metadata==6.6.0	parso==0.8.3	pyparsing==3.0.9
importlib-resources==5.12.0	pickleshare==0.7.5	pyrsistent==0.19.3
ipykernel==6.16.2	Pillow==9.5.0	python-dateutil==2.8.2
ipython==7.34.0	pkgutil_resolve_name==1.3.10	
		pytz==2023.3
ipython-genutils==0.2.0	prometheus-client==0.16.0	pywin32==306
ipywidgets==8.0.6	prompt-toolkit==3.0.38	pywinpty==2.0.10
jedi==0.18.2	protobuf==3.20.3	pyzmq==25.0.2
Jinja2==3.1.2	psutil==5.9.5	



joblib==1.2.0

jsonschema==4.17.3

jupyter==1.0.0

pyasn1==0.5.0

pyasn1-modules==0.3.0

pycparser==2.21

qtconsole==5.4.2

Pygments==2.15.1

jupyter-console==6.6.3

## Bibliografía

- [1] Niall McCarthy, “Which Countries Produce The Most Palm Oil?,” <https://www.forbes.com/sites/niallmccarthy/2020/10/02/which-countries-produce-the-most-palm-oil-infographic/?sh=5011ecd51e42>, Apr. 08, 2023.
- [2] S. Radhika, Y. Tamura, and M. Matsui, “Application of remote sensing images for natural disaster mitigation using wavelet based pattern recognition analysis,” in *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2016, pp. 84–87. doi: 10.1109/IGARSS.2016.7729012.
- [3] T. Bai, D. Li, K. Sun, Y. Chen, and L. Wenzhuo, “Cloud Detection for High-Resolution Satellite Imagery Using Machine Learning and Multi-Feature Fusion,” *Remote Sens (Basel)*, vol. 8, p. 715, Apr. 2016, doi: 10.3390/rs8090715.
- [4] E. Bruzelius *et al.*, “Satellite images and machine learning can identify remote communities to facilitate access to health services,” *Journal of the American Medical Informatics Association*, vol. 26, no. 8–9, pp. 806–812, Aug. 2019, doi: 10.1093/jamia/ocz111.
- [5] A. D. Baimey *et al.*, “WRM in English | World Rainforest Movement,” *Nine Reasons to Say NO to Contract Farming with Palm Oil Companies*, Feb. 23, 2023. <https://www.wrm.org.uy/publications/nine-reasons-to-say-no-to-contract-farming-with-palm-oil-companies> (accessed Feb. 22, 2023).
- [6] T. Oberhuber, “Ecologistas en Acción,” Biodiversidad: Camino de la sexta gran extinción de especies,” *Ecologistas en Acción*,” *Biodiversidad: Camino de la sexta gran extinción de especies*, Nov. 04, 2022. <https://www.ecologistasenaccion.org/> (accessed Nov. 03, 2023).
- [7] Houghton Juan, “Conflictos territoriales de los pueblos indígenas en Colombia,” 2008.
- [8] Colorado Barrientos Jorge Eliecer, “La implementación de semillas transgénicas en Colombia,” 2014.
- [9] Fedepalma, “Federación Nacional de Cultivadores de Palma de Aceite,” <https://web.fedepalma.org/la-palma-de-aceite-en-colombia-departamentos>, Apr. 08, 2023.
- [10] A. P. Cooman, A. P. Bochoo, and E. Hernandez, “De la investigación al cambio productivo. Sostenibilidad y eficiencia para la palmicultura colombiana,” pp. 44–49, 2009.
- [11]

- Congreso de la República de Colombia, “Ley 1021 o Ley Forestal (2006),” [http://www.secretariassenado.gov.co/senado/basedoc/ley\\_1021\\_2006.html](http://www.secretariassenado.gov.co/senado/basedoc/ley_1021_2006.html), Apr. 08, 2023.
- [12] W. Robles, “Revisiting Agrarian Reform in Brazil, 1985–2016,” *J Dev Soc*, vol. 34, no. 1, pp. 1–34, Jan. 2018, doi: 10.1177/0169796X17749658.
- [13] W. Zhang, K. Itoh, J. Tanida, and Y. Ichioka, “Parallel distributed processing model with local space-invariant interconnections and its optical architecture.,” *Appl Opt*, vol. 29 32, pp. 4790–7, 1990.
- [14] J. Iqbal, M. Vogt, and J. Bajorath, “Activity landscape image analysis using convolutional neural networks,” *J Cheminform*, vol. 12, no. 1, May 2020, doi: 10.1186/s13321-020-00436-5.
- [15] K. Kapach, E. Barnea, R. Mairon, Y. Edan, and O. Ben-Shahar, “Computer vision for fruit harvesting robots - state of the art and challenges ahead,” *Int. J. Comput. Vis. Robotics*, vol. 3, pp. 4–34, 2012.
- [16] V. Mnih, “Machine Learning for Aerial Image Labeling,” 2013.
- [17] A. Payne and K. Walsh, “Machine Vision in Estimation of Fruit Crop Yield,” 2013. doi: 10.1201/b17441-17.
- [18] S. Bargouti and J. Underwood, “Image Segmentation for Fruit Detection and Yield Estimation in Apple Orchards,” *J Field Robot*, vol. 34, Oct. 2016, doi: 10.1002/rob.21699.
- [19] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, “Convolutional Neural Networks for Large-Scale Remote-Sensing Image Classification,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 2, pp. 645–657, 2017, doi: 10.1109/TGRS.2016.2612821.
- [20] A. Kamilaris and F. X. Prenafeta-Boldú, “A review of the use of convolutional neural networks in agriculture,” *J Agric Sci*, vol. 156, no. 3, pp. 312–322, 2018, doi: DOI: 10.1017/S0021859618000436.
- [21] J. Amara, B. Bouaziz, and A. Algergawy, “A Deep Learning-based Approach for Banana Leaf Diseases Classification.”
- [22] A. Shabbir *et al.*, “Satellite and Scene Image Classification Based on Transfer Learning and Fine Tuning of ResNet50,” *Math Probl Eng*, vol. 2021, 2021, doi: 10.1155/2021/5843816.
- [23] C. E. Vargas Manrique, “Cultivos ilícitos y erradicación forzosa en Colombia,” *Cuadernos de Economía*, vol. 23, pp. 109–141, 2004.
- [24] M. R. Chandran, “Producción Sostenible de aceite de palma: retos y oportunidades,” 2007.
- [25] G. Finzi, “El caso de Poligrow en Mapiripán, Meta: entre acaparamiento (ilegal) de tierras y capitalismo verde,” *Ciencia Política*, vol. 12, no. 24, pp. 21–50, Jul. 2017, doi: 10.15446/cp.v12n24.63533.
- [26] “Maquinaria computacional e Inteligencia Alan Turing, 1950.”
- [27] A. M. Turing, “COMPUTING MACHINERY AND INTELLIGENCE,” 1950.
- [28] IBM, “¿Qué es la inteligencia artificial?,” <https://www.ibm.com/mx-es/topics/artificial-intelligence>, Apr. 20, 2023.
- [29] R. Lackses, D. Mack, J. Ziola, and K. Ahern, *Neuronal Networks: Basics and Applications*, CBT., vol. 1. Verlag Berlin Heidelberg: 1998, 1998.
- [30]

- Y. Ding, Q. Feng, T. Wang, and X. Fu, "A modular neural network architecture with concept," *Neurocomputing*, vol. 125, pp. 3–6, Feb. 2014, doi: 10.1016/j.neucom.2012.06.051.
- [31] Bernard Widrow and Marcian E Hoff, "Adaptative Switching Circuits".
- [32] M. V Valueva, N. N. Nagornov, P. A. Lyakhov, G. V Valuev, and N. I. Chervyakov, "Application of the residue number system to reduce hardware costs of the convolutional neural network implementation," *Math Comput Simul*, vol. 177, pp. 232–243, 2020, doi: <https://doi.org/10.1016/j.matcom.2020.04.031>.
- [33] J. JALOMO, E. PRECIADO, and J. GUDIÑO, "Design of a convolutional neural network for classification of biomedical signals," *Journal of Research and Development*, pp. 15–20, Jun. 2020, doi: 10.35429/jrd.2020.17.6.15.20.
- [34] S. Contreras and F. De la Rosa, *Aplicación de Deep Learning en Robótica Móvil para Exploración y Reconocimiento de Objetos basados en Imágenes*. 2016. doi: 10.1109/ColumbianCC.2016.7750800.
- [35] Amrita Pathak, "Redes neuronales convolucionales (CNN): una introducción," <https://geekflare.com/es/convolutional-neural-networks/>, Aug. 2022.
- [36] Lopez Saca Fidel, "Clasificación de imágenes usando redes neuronales convolucionales."
- [37] Sarker IH, Furhad MH, and Nowrozy R, "Ai-driven cybersecurity: an overview, security intelligence modeling and research directions.," *SN Computer*, pp. 1–18, 2021.
- [38] Sarker IH, "Machine learning: Algorithms, real-world applications and research directions," *SN Comput Sci*, vol. 2, no. 3, pp. 1–21, 2021.
- [39] I. H. Sarker, "Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions," *SN Comput Sci*, vol. 2, no. 6, p. 420, 2021, doi: 10.1007/s42979-021-00815-1.
- [40] Abadi M *et al.*, "Tensorflow: a system for large-scale machine learning. In: 12th {USENIX} Symposium on operating systems design and implementation," *{OSDI} 16*, pp. 265–283, 2016.
- [41] K. Weiss, T. M. Khoshgoftaar, and D. D. Wang, "A survey of transfer learning," *J Big Data*, vol. 3, no. 1, Dec. 2016, doi: 10.1186/s40537-016-0043-6.
- [42] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *IEEE Trans Knowl Data Eng*, vol. 22, no. 10, pp. 1345–1359, 2010, doi: 10.1109/TKDE.2009.191.
- [43] N. Mo, L. Yan, R. Zhu, and H. Xie, "Class-Specific Anchor Based and Context-Guided Multi-Class Object Detection in High Resolution Remote Sensing Imagery with a Convolutional Neural Network," *Remote Sens (Basel)*, vol. 11, p. 272, Apr. 2019, doi: 10.3390/rs11030272.
- [44] P. Zavala and C. Zavala, "USO DE IMÁGENES SATELITALES DE ALTA RESOLUCIÓN PARA GENERAR CARTOGRAFÍA," *Revista Facultad de Ingeniería - Universidad de Tarapacá*, vol. 10, pp. 35–43, Apr. 2002, doi: 10.4067/S0718-13372002001000005.
- [45] Felipe Espinosa Wang, "Satélite chino podría tomar imágenes de alta resolución de ciudades de EE. UU. en segundos," <https://www.dw.com/es/sat%C3%A9lite-chino-podr%C3%ADa-tomar-im%C3%A1genes-de-alta-resoluci%C3%B3n-de-ciudades-de-ee-uu-en-segundos/a-60289654>, China, Apr. 22, 2023.
- [46]

- P. A. Devijver, J. Kittler, Jean-Philippe Lang, and Jeff Schneider, “Cross-Validation,” [https://es.wikipedia.org/wiki/Validaci%C3%B3n\\_cruzada#cite\\_note-vca-6](https://es.wikipedia.org/wiki/Validaci%C3%B3n_cruzada#cite_note-vca-6), Apr. 16, 2023.
- [47] Andrew W. Moore, “Cross-validation for detecting and preventing overfitting,” *Wayback Machine - Camegie Mellon University*, Aug. 2011.
- [48] The European Space Agency, “Las imágenes por satélite, esenciales para el funcionamiento de Google Earth,” [https://www.esa.int/Space\\_in\\_Member\\_States/Spain/Las\\_imagenes\\_por\\_satelite\\_esenciales\\_para\\_el\\_funcionamiento\\_de\\_Google\\_Earth](https://www.esa.int/Space_in_Member_States/Spain/Las_imagenes_por_satelite_esenciales_para_el_funcionamiento_de_Google_Earth), Apr. 22, 2023.
- [49] EE Explorer, “Introducción a Google Earth Engine,” [https://www.google.com/intl/es\\_es/earth/outreach/learn/introduction-to-google-earth-engine/](https://www.google.com/intl/es_es/earth/outreach/learn/introduction-to-google-earth-engine/), Apr. 22, 2023.
- [50] L. Kumar and O. Mutanga, “Google Earth Engine applications since inception: Usage, trends, and potential,” *Remote Sens (Basel)*, vol. 10, no. 10, Oct. 2018, doi: 10.3390/rs10101509.
- [51] N. Gorelick, M. Hancher, M. Dixon, S. Ilyushchenko, D. Thau, and R. Moore, “Google Earth Engine: Planetary-scale geospatial analysis for everyone,” *Remote Sens Environ*, vol. 202, pp. 18–27, Dec. 2017, doi: 10.1016/j.rse.2017.06.031.
- [52] CHRONICLES DENGEN, “What is the difference between Sentinel 1 and 2?,” <https://chroniclesdengen.com/what-is-the-difference-between-sentinel-1-and-2/>, Apr. 24, 2023.
- [53] Wikipedia, “Sentinel-2,” <https://en.wikipedia.org/wiki/Sentinel-2>, Apr. 24, 2023.
- [54] ESA, “Sentinel Online,” <https://sentinels.copernicus.eu/web/sentinel/missions/sentinel-2>, Apr. 24, 2023.
- [55] C. Yang *et al.*, “Big Earth data analytics: a survey,” *Big Earth Data*, vol. 3, no. 2, pp. 83–107, 2019, doi: 10.1080/20964471.2019.1611175.
- [56] X. Kang, Y. Huang, S. Li, H. Lin, and J. A. Benediktsson, “Extended Random Walker for Shadow Detection in Very High Resolution Remote Sensing Images,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 2, pp. 867–876, 2018, doi: 10.1109/TGRS.2017.2755773.
- [57] GOTA Universidad de Laguna, “Proceso de teledetección e iteración con la radiación incidente y los objetos de interés,” <https://www.ull.es/investigacion/grupos-investigacion/grupo-de-observacion-de-la-tierra-y-la-atmosfera-gota/>, Nov. 28, 2022.
- [58] Britos Paola, “PROCESOS DE EXPLOTACION DE INFORMACION BASADOS EN SISTEMAS INTELIGENTES,” Tesis presentada para obtener el grado de Doctor en Ciencias Informaticas, Universidad Nacional de la Plata, La Plata, Buenos Aires , 2008.
- [59] F. Peralta, “Proceso de Conceptualización del Entendimiento del Negocio para Proyectos de Explotación de Información,” *Revista Latinoamericana de Ingenieria de Software*, vol. 2, p. 273, Apr. 2014, doi: 10.18294/relais.2014.273-306.
- [60] Fedepalma, “Localización geográfica de cultivos y plantas de beneficio,” <http://sispaweb.fedepalma.org/sispaweb/default.aspx?Control=Pages/areas>, Apr. 26, 2023.
- [61]

- D. Adrià, W. Serge, M. Erik, G. David, P. Stephen, and S. Zoltan, “High resolution global industrial and smallholder oil palm map for 2019,” Jan. 2021, doi: 10.5281/ZENODO.4473715.
- [62] Samarth Agrawal, “How to split data into three sets (train, validation, and test) And why?,” <https://towardsdatascience.com/how-to-split-data-into-three-sets-train-validation-and-test-and-why-e50d22d3e54c>, May 17, 2021.
- [63] d’Archimbaud Edouard, “Training, Validation and Test Sets: How To Split Machine Learning Data,” <https://kili-technology.com/training-data/training-validation-and-test-sets-how-to-split-machine-learning-data>, May 21, 2023.
- [64] Samarth Agrawal, “Understanding the Confusion Matrix from Scikit learn,” <https://towardsdatascience.com/understanding-the-confusion-matrix-from-scikit-learn-c51d88929c79>, Jan. 01, 2021.