



Universidad Católica “Nuestra Señora de la Asunción”

FACULTAD DE CIENCIAS Y TECNOLOGÍA (CYT)
DEPARTAMENTO DE ELECTRÓNICA E INFORMÁTICA (DEI)

Proyecto Final de Carrera
Grado en Ingeniería Informática

USO DE REDES NEURONALES CONVOLUCIONALES PARA INTERPRETACIÓN DE IMÁGENES SATELITALES

Alumno

Giovanni Rene Dueck Neufeld

Tutores

Dr. Alberto Ramírez
Dr. Félix Carvallo

Noviembre 2025

Dedicatoria

Dedico este trabajo a mi esposa, quien ha sido mi motivación y apoyo constante en esta última

dedicatoria: etapa de estudio.
complete

Agradecimientos

agradecimient

Índice

Índice de figuras	3
Índice de cuadros	4
1. Resumen	6
2. Abstract	7
3. Planteamiento del problema	8
3.1. Introducción a la problemática	8
3.2. Descripción del problema	8
3.3. Objetivos	9
3.3.1. Objetivo general	9
3.3.2. Objetivos específicos	9
3.4. Antecedentes	9
3.5. Estructura del documento	9
4. Estado del Arte	10
4.1. Estrategias de búsqueda	10
4.2. Extracción y síntesis de datos	11
4.3. Preguntas de investigación	11
4.4. Resultados	11
4.4.1. Discusión de los resultados	11
5. Marco Teórico	14
5.1. Teledetección	14
5.1.1. Aplicaciones	14
5.1.2. Características de los datos	15
5.1.3. Disponibilidad de recursos	15
5.2. Redes Neuronales Convolucionales	15
5.2.1. Aplicaciones	16
5.2.2. Técnicas y arquitecturas	16
5.3. Análisis de imágenes satelitales	17
5.3.1. Clasificación	17
5.3.2. Segmentación	17
5.3.3. Detección de cambios	18
5.3.4. Serie de tiempo	18
6. Marco Metodológico	19
6.1. Área de estudio	19
6.2. Estrategia de procesamiento de datos	19
6.2.1. Herramientas utilizadas	19
6.2.2. Recopilación de datos	19
6.2.3. Selección de archivos	19
6.2.4. Unificación y preparación de datos	20
6.2.5. Limpieza de datos	21
6.2.6. Resumen del procesamiento de los datos	21

7. Experimentos y Resultados	22
7.1. Síntesis de datos de entrenamiento	22
7.2. Herramientas utilizadas	22
7.3. Partición de datos de entrenamiento	22
7.4. Entrenamiento y validación	23
7.5. Resultados	23
7.5.1. FCN	24
7.5.2. FarSeg	25
7.5.3. U-Net	26
7.5.4. Predicción en conjunto	26
8. Discusión de los Resultados	27
8.1. Conclusión	27
8.2. Recomendaciones	27
8.3. Contribución	27
8.4. Trabajos futuros	27
9. Apéndice	28
Referencias	29

Índice de figuras

1.	Imágenes satelitales del Valle de la Muerte con diferentes resoluciones espetrales. El área superior de (a) coincide con el área inferior de (b).	14
2.	Espectro electromagnético visualizado. Diferentes sustancias y materiales reflejan una variedad de frecuencias más allá del espectro visible, que es relativamente reducido.	15
3.	Ejemplo de red neuronal convolucional con varias capas de convolución y pooling, que preprocesan y comprimen la imagen, y varias capas convencionales, densamente conectadas, que hacen el trabajo de clasificación sobre un vector unidimensional.	17

Índice de cuadros

1.	Algunos proyectos de clasificación y segmentación de imágenes de teledetección y CNN	12
2.	Ventajas y desventajas de CNN en comparación con otras técnicas	13
3.	Técnicas para abarcar el problema de pocos datos	13
4.	Metadatos incluidos con cada tesela del mosaico satelital	20
5.	Teselas utilizadas y fechas de sobrevuelo	20
6.	Parámetros de entrenamiento generales	24
7.	Hiperparámetros y resultados de la arquitectura FCN	24
8.	Hiperparámetros y resultados de la arquitectura FarSeg	25
9.	Hiperparámetros y resultados de la arquitectura U-Net	26

Todo list

dedicatoria: complete
agradecimientos
resumen	6
abstract	7
confirmar con el profe	9
planteamiento/antecedentes	9
agregar métricas al cuadro comparativo	11
map figures	19
ref	22
figure of swir image and mask	22
add software and cuda versions in a table	22
diagram of <i>random_grid_cell_assignment</i>	22
clasificacion	23
make diagram	24
try pretrained	25
figures of test plots and generated masks in appendix	26
try predictions with the same tile from a different day	26
figures of generated masks for training and test tiles	26
discusion de los resultados	27
apendice	28

1. Resumen

resumen

2. Abstract

abstract

3. Planteamiento del problema

3.1. Introducción a la problemática

Imágenes satelitales o teledetección se refiere a imágenes capturadas por un sensor montado en un satélite artificial, para extraer información. Estas imágenes contienen información multiespectro, es decir, además de la luz visible se toman imágenes de bandas invisibles como por ejemplo la luz infrarroja. [1]

Para la captura de estas imágenes se emplean varios métodos, que se dividen en dos categorías: sensores pasivos recolectan radiación electromagnética reflejada del sol, mientras que sensores activos emiten su propia radiación y captan la reflexión de la tierra. Sensores activos requieren de una cantidad importante de energía para operar, pero tienen la ventaja de operar a cualquier hora del día y la capacidad de crear imágenes en bandas que el sol no emite. [1]

Diferentes suelos, vegetación, o humedad reflejan diferentes bandas de radiación. Estas imágenes son usadas en varias aplicaciones, desde Sistemas de Información Geográfica y mapas a meteorología y monitoreo de la salud de vegetación forestal. Un índice bastante común es el Índice de Vegetación de Diferencia Normalizada, o NDVI por sus siglas en inglés, el cual es usado para estimar la cantidad, calidad y desarrollo de la vegetación con base a la mediación. Estos datos ya se usan en sistemas de advertencia temprana de sequías y la predicción del rendimiento de la agricultura en los Estados Unidos a partir de los datos de la NASA. [2]

La importancia de la producción agropecuaria y agroganadera en el Paraguay también invita a considerar estas tecnologías para el monitoreo de la salud de la vegetación y el uso adecuado de la tierra. Actualmente, ya se están empleando tecnologías de teledetección y el NDVI en el sector agrícola en aplicaciones como la detección de malezas y predicción del orden ideal de cosecha de campos cultivados. [3]

Con aproximadamente la mitad del territorio paraguayo hacia el norte del Río Paraguay en la región semi-árida del Chaco, tecnologías que alivien las sequías y precipitación baja son muy valiosas, tanto para la agricultura y ganadería en las estancias chaqueñas como para centros poblacionales aislados como por ejemplo las comunidades indígenas. Estos pueblos generalmente se caracterizan por la probreza, que se ve manifestada en una salud deteriorada producto de la deficitaria alimentación y falta de agua potable.

Un paleocauce es un cauce por el cual antiguamente fluía agua, como por ejemplo un antiguo lecho de un río. Los paleocaucos han sido propuestos como reservorios o conductos para el flujo subterráneo de agua dulce. Se consideran de interés principalmente los paleocaucos arenosos, y estos pueden ser aprovechados para acceder al agua en áreas en las que la distribución habitual del agua no existe o está dificultada de alguna forma. [4] Con la abundancia de paleocaucos en el Chaco central, que ocupan un 15 % de la región, esta propuesta es una bastante prometedora que ya ha sido considerada en investigaciones anteriores. [5]

La detección de estos paleocaucos se haría a partir de imágenes satelitales tomadas a lo largo de un periodo amplio por medio de redes neuronales. Las redes convolucionales son una categoría de redes neuronales especializadas para el procesamiento de imágenes. El principio básico de su funcionamiento consiste en la convolución de grupos píxeles cercanos, una operación que permite tener en cuenta no solo el valor de cada píxel individual, sino el contexto de los mismos. [6]

3.2. Descripción del problema

La motivación principal del proyecto es la necesidad de mitigar las sequías prolongadas del Chaco Paraguayo, que impacta de forma más severa a zonas remotas o rurales. El objetivo del trabajo es aplicar técnicas de clasificación e interpretación de imágenes satelitales por medio de redes neuronales al problema de la identificación de usos de suelo, particularmente para identificar paleocaucos.

Cabe resaltar que los experimentos realizados en el marco de esta investigación se limitan a la clasificación y segmentación de imágenes satelitales. No forma parte de este trabajo el sondeo de paleocaucos para determinar la cantidad o calidad del agua subterránea o la creación de pozos.

3.3. Objetivos

3.3.1. Objetivo general

Creación de modelos a partir de redes neuronales convolucionales para la clasificación y caracterización de imágenes satelitales.

3.3.2. Objetivos específicos

1. Análisis de imágenes satelitales correspondientes a la región occidental del Paraguay a lo largo de un periodo temporal amplio
2. Identificación y clasificación de componentes de uso de suelo
3. Determinación de áreas de ocurrencia de paleocauces

confirmar
con el
profe

3.4. Antecedentes

planteamiento

3.5. Estructura del documento

La estructura del resto del documento se desarrolla como sigue:

Sección 4: Análisis, criterios, procedimientos y preguntas de investigación, y desarrollos previos relacionados con el proyecto. Identificación de las tendencias y tecnologías de vanguardia en el campo de análisis de imágenes satelitales.

Sección 5: Marco teórico relevante, definiciones de conceptos clave para este proyecto.

Sección 6: Marco metodológico, se describen detalladamente el área de estudio y las estrategias y técnicas empleadas para el procesamiento de los datos.

Sección 7: Experimentos y resultados, una descripción detallada de las metodologías utilizadas y sus resultados, además de la justificación de la elección de los métodos y las herramientas utilizadas.

Sección 8: Discusión de los resultados, en donde se relata la conclusión del proyecto y su relevancia en el contexto académico. Se incluyen recomendaciones para posibles implementaciones de la metodología y posibles futuras investigaciones, de acuerdo al impacto esperado y los beneficios potenciales que pudiera generar.

4. Estado del Arte

En este capítulo se explora el estado del arte del uso de imágenes satelitales en diversas áreas y las técnicas de análisis relevantes para este proyecto. Esta investigación tiene el fin de entender la forma en que se aplican en sus diversos campos de aplicación y cuáles técnicas son las más eficaces en el campo a estudiarse.

4.1. Estrategias de búsqueda

Para la revisión de literatura se utilizaron términos referentes a [Redes Neuronales], [Teledetección], y [Clasificación y Detección]. Se tuvieron en cuenta principalmente obras en el idioma inglés, aunque se incluyen obras en español también. Para la búsqueda se usaron los siguientes términos:

Términos	Sinónimos
Neural Network	Convolutional Neural Network Deep Learning
Remote Sensing	Satellite Imagery
Classification	Detection
Lack of data	Small data

Las cadenas de búsqueda se construyen a partir de los términos y sus sinónimos. Las cadenas con mejores resultados fueron [*Neural Network AND Remote Sensing AND Classification*], [*Convolutional Neural Network AND Remote Sensing AND Classification*], [*Convolutional Neural Network AND Satellite Imagery AND Classification*], [*Convolutional Neural Network AND Satellite Imagery AND Detection*], y [*Deep Learning AND Remote Sensing AND Small Data*]

El motor de búsqueda utilizado fue Google Scholar, poniendo mayor enfoque en resultados provenientes de bases de datos reconocidas y establecidas como IEEE Xplore, ScienceDirect y ArXiv.

También se incluyeron trabajos relevantes para la universidad y la realidad local del país, proveídos por los tutores.

Criterios de selección Se incluyen artículos, papers, conferencias, y otros trabajos formales debidamente documentados. Se establecen los siguientes criterios para juzgar si un trabajo es incluido o excluido de esta investigación:

Inclusión 1 Trabajos que se enfoquen en la clasificación de imágenes satelitales por medio de redes neuronales en el rango de publicación de 2014 a 2024.

Inclusión 2 Trabajos que coincidan en su contenido con los términos de búsqueda.

Inclusión 3 Trabajos cuyo contenido sea relevante para la investigación.

Exclusión 1 Trabajos que no contengan las palabras claves o son irrelevantes para el campo de investigación.

Exclusión 2 Trabajos que se centran en un término de búsqueda pero no incluyen alguno de los demás.

Exclusión 3 Trabajos con una cantidad mayoritaria de información irrelevante para el tema estudiado.

Procedimientos de selección El proceso de selección de trabajos se basa en responder las preguntas de investigación presentadas en la siguiente sección, con el fin de responderlas con información válida y actual.

Se limita el número de artículos incluidos a 20, y en caso de que se supere la cantidad encontrada se filtran por medio de los siguientes criterios:

P.S.1. Los trabajos deben responder la mayor cantidad de preguntas de investigación.

P.S.2. Los trabajos deben contar con la mayor cantidad de incidencia de términos definidos anteriormente.

P.S.3. Artículos que incluyan las palabras clasificar, interpretar, imágenes satelitales, redes neuronales, redes neuronales convolucionales en su resumen, conclusión.

4.2. Extracción y síntesis de datos

Para la planilla de extracción de datos de cada estudio, se guardaron título, autores, año de publicación, resumen, palabras claves, fuente y conclusiones relacionadas a las preguntas de investigación. En el cuadro de la sección de resultados se listan las informaciones relevantes para responder las preguntas de investigación de este proyecto. Para determinar la inclusión de cada artículo se realizó un análisis de los objetivos y resultados de cada trabajo, teniendo en cuenta los criterios de selección. Para realizar la síntesis de los datos se realizó la estrategia descriptiva, que detalla y ordena las conclusiones principales de los autores de los artículos para una mejor compresión de las ideas principales.

4.3. Preguntas de investigación

El objetivo principal de este estudio es determinar cuál es el estado del arte en técnicas utilizadas para clasificar y caracterizar o interpretar imágenes satelitales por medio de redes neuronales convolucionales. Con este fin en mente, se plantean las siguientes preguntas de investigación:

P.1. ¿Qué proyectos se están llevando adelante para clasificar y caracterizar imágenes satelitales usando redes neuronales convolucionales?

P.2. ¿Cuáles son las ventajas y/o desventajas de la clasificación y caracterización de imágenes satelitales usando redes neuronales convolucionales en comparación con las alternativas?

P.3. ¿Qué soluciones existen para abordar la falta de datos de entrenamiento para las redes neuronales convolucionales?

4.4. Resultados

Las respuestas a las preguntas de investigación se encuentran resumidas en los cuadros 1, 2, y 3.

agregar métricas al cuadro comparativo

4.4.1. Discusión de los resultados

El campo de análisis de datos de teledetección ha visto muchos proyectos en los últimos años, y las redes convolucionales han sido una pieza crucial para crear modelos más poderosos y eficientes que las redes neuronales convencionales. La exploración de alternativas a redes neuronales, como RF y SVN, indica que la CNN es la tecnología más prometedora para la clasificación de una gran cantidad de datos.

La posibilidad de usar datos sin etiquetamiento en el entrenamiento indica que la falta de datos etiquetados no es razón para dejar de considerar el entrenamiento de modelos independientes. Sin embargo, gracias a la cantidad de cómputo necesaria para la tarea, el exploramiento que se realice en este trabajo necesariamente deberá ser limitado. Aun así, gracias a estos factores, el resultado debería reflejar con bastante precisión los métodos necesarios para crear modelos con objetivos poco comunes en terreno nuevo para la teledetección a nivel nacional.

También de importancia es la naturaleza altamente dinámica de los datos. Cada año la cantidad de datos de alta resolución se multiplica y se vuelve más accesible, y para mantenerse al tanto de estos cambios, son necesarios el mantenimiento y la actualización periódica de estos modelos. Por otro lado, una de las técnicas más utilizadas en los trabajos mencionados es la transferencia de modelos preentrenados por medio de “fine-tuning”, lo que hace el mantenimiento un obstáculo más fácil de superar.

Proyecto	Objetivos	Métodos y Observaciones
Clasificación y Segmentación de Ortofotografía Satelital Usando Redes Neuronales Convolucionales. [7]	Explorar el uso de CNN para la clasificación por píxel completa, rápida y exacta de una ciudad pequeña.	Un CNN es apto para el análisis de imágenes multiespectrales corregidas ortográficamente, junto con un modelo de superficie digital de una pequeña ciudad.
Identificación y Mapeo de Paleocauces Utilizando Imágenes Satelitales de Alta Resolución en la Llanura Costera de la Bahía Samborombón, Argentina [8]	Reconocimiento y análisis de paleocauces por medio de sensores remotos de los paleocauces de los ríos Samborombón y Salado, en la zona de su desembocadura, para la reconstrucción paleoambiental.	Se utilizan imágenes satelitales multiespectrales de alta resolución en combinación con algunos filtros de dirección convolucionales. Sin embargo, no se utilizan redes neuronales.
Redes Neuronales Completamente Convolucionales para Clasificación de Imágenes Satelitales [9]	Demostrar un modelo CNN que utiliza solamente capas convolucionales, sin capas tradicionales, para la clasificación de imágenes satelitales.	El modelo resultante tiene mejor rendimiento que modelos CNN con capas tradicionales y menor tiempo de entrenamiento.
Detección Automática de Objetos en Imágenes Aéreas basada en CNN [10]	Aprovechamiento de la cantidad de imágenes de teledetección por medio de CNN para la detección automática de objetos en diversas aplicaciones.	CNNs son útiles para la detección de objetos, mejor que métodos basados en “features” y varios otros modelos CNN.
SatCNN: clasificación de datos de imágenes satelitales usando CNN ágiles [11]	Diseño de una arquitectura CNN especializada para el análisis de imágenes satelitales en vez de adaptar un modelo existente creado para la clasificación de escenas naturales.	Pruebas en varias muestras de datos muestran una efectividad de más de 99.5 %.
Un CNN basado en pedazos (patch-based) para la clasificación de datos de teledetección [12]	Creación de un modelo para la clasificación de imágenes de resolución media, donde estructuras finas no existen para la clasificación por píxel.	Mejora de hasta 25 % por encima de CNN basados en clasificación por píxel.
Clasificación de Imágenes Satelitales con Deep Learning [13]	Creación de modelos CNN para la automatización del análisis de grandes cantidades de imágenes satelitales de alta resolución.	El modelo creado gana el segundo puesto en la competición de Functional Map of the World (fMoW) TopCoder, 15 de 63 clases clasificadas con exactitud de 95 %.
CNN para la Clasificación de Humedales Complejos Usando Imágenes de Teledetección Óptica [14]	Entrenamiento de un modelo preentrenado para la clasificación de imágenes de alta resolución.	Rendimiento mejor que RF, inclusive con menos características de entrada, con una exactitud de 94 %.
DeepSat V2: CNN de Características Aumentadas para la Clasificación de Imágenes Satelitales [15]	Creación de conjuntos de datos de imágenes satelitales etiquetadas y un modelo CNN con varias técnicas nuevas para su análisis.	El modelo resultante logra clasificar correctamente con más de 99 % de exactitud en los nuevos conjuntos de datos.
Un CNN para la Detección de Anomalías Térmicas Volcánicas en Imágenes Satelitales [16]	Entrenamiento de un modelo CNN para analizar imágenes de actividad volcánica en infrarrojo.	Un CNN preentrenado adaptado a nuevos datos seleccionados a mano logra buena clasificación.

Cuadro 1: Algunos proyectos de clasificación y segmentación de imágenes de teledetección y CNN

Tipo	Característica	Referencias
Ventaja	CNN patch-based (basado en pedazos) mejor que NN convencional o CNN basados en píxeles, SVN o RF	[12]
Ventaja	Clasificación de datos multifuente por medio de CNN mejor que SVN y ELM	[17]
Ventaja	Clasificación de uso de suelo por CNN mucho mejor que RF, especialmente para terrenos difíciles	[14]
Desventaja	CNN basado en píxeles comparable o peor que NN convencional, SVN o RF	[12]
Desventaja	Entrenamiento de modelos basados en redes neuronales es más computacionalmente costoso que SVN o RF	[12, 17, 14]

Cuadro 2: Ventajas y desventajas de CNN en comparación con otras técnicas

Técnica	Descripción, (+) Ventajas, (-) Desventajas	Referencias
Transferencia (Transfer, Fine-tuning)	Uso de modelo preentrenado con un conjunto de datos relevante y ajustado con un conjunto de datos nuevo. (+) Mejor rendimiento, menos datos de entrenamiento, mejor generalizabilidad. (-) Riesgo de reducción de rendimiento con transferencia a dominio diferente, tamaño de modelo grande.	[18, 19, 20] [21, 11, 16]
Auto supervisado (Self-supervised)	Creación de un modelo con etiquetas creadas por el modelo, seguido de entrenamiento supervisado con etiquetas proveídas. (+) Uso de datos no etiquetados, reconocimiento de patrones sin necesidad de etiquetación, mejor generalizabilidad. (-) Computacionalmente caro, posibilidad de que el modelo deje de entrenarse con algunas técnicas.	[18]
Semi supervisado (Semi-supervised)	Mezcla de entrenamiento supervisado y no supervisado con conjuntos de datos etiquetados y no etiquetados. (+) Uso de datos etiquetados y no etiquetados, mejor generalizabilidad. (-) Computacionalmente caro, riesgo de “overfitting”, sensible a calidad de datos.	[18]
Débilmente supervisado (Weakly-supervised)	Desarrollo de un modelo con datos etiquetados parcialmente, de manera imprecisa o con ruido. (+) Costo de etiquetamiento reducido, permite un modelo inexacto para permitir escalabilidad. (-) Computacionalmente caro, menos exacto que entrenamiento (completamente) supervisado.	[18]
Conjunto (Ensemble)	Combinación de muchos modelos individuales que aprendieron patrones de forma diferente para la predicción. (+) Mejor generalizabilidad, robustez contra perturbación de datos e incertidumbre. (-) Computacionalmente caro, peor interpretabilidad que un modelo simple.	[18, 7, 13]
Validación cruzada (Cross-validation)	Entrenar y validar un modelo varias veces usando diferentes particiones de datos para el entrenamiento y la validación. (+) Modelo menos sesgado, evita reportaje sobre-optimista de rendimiento, mejor generalizabilidad. (-) Computacionalmente caro	[18]

Cuadro 3: Técnicas para abarcar el problema de pocos datos

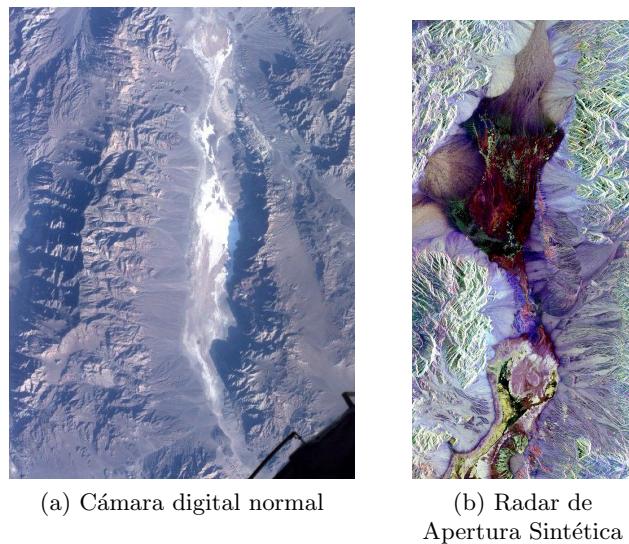


Figura 1: Imágenes satelitales del Valle de la Muerte con diferentes resoluciones espetrales. El área superior de (a) coincide con el área inferior de (b).

5. Marco Teórico

En esta sección se exploran brevemente los conceptos más fundamentales para el proyecto. En particular, los temas a desarrollarse son una revisión del estado de la teledetección y los recursos disponibles, las bases de las redes neuronales convolucionales y su uso en el análisis de imágenes satelitales, las técnicas modernas en uso en la investigación y en aplicaciones en el mundo real.

5.1. Teledetección

Teledetección se refiere a la captación o detección remota de alguna señal o imagen. En este contexto nos referimos específicamente a imágenes captadas por medio de un sensor montado en un satélite artificial o algún vehículo aéreo como un avión o un dron, para extraer información. Estas imágenes contienen información multiespectro, es decir, además de la luz visible se toman imágenes de bandas invisibles como por ejemplo la luz infrarroja. [1] A lo largo de este proyecto, el término “teledetección” se refiere a la captación de imágenes por medio de satélites.

Para la captura de estas imágenes se emplean varios métodos, que se dividen en dos categorías: sensores pasivos recolectan radiación electromagnética reflejada del sol, mientras que sensores activos emiten su propia radiación y captan la reflexión de la tierra. Sensores activos requieren de una cantidad importante de energía para operar, pero tienen la ventaja de operar a cualquier hora del día y la capacidad de crear imágenes en bandas que el sol no emite. [1]

Los primeros programas de observación de la tierra por medio de satélites surgieron en los años 70 y 80. El primero fue el programa Landsat de los Estados Unidos en 1972, y le siguieron programas similares en India, Francia y la Unión Europea. [22]

5.1.1. Aplicaciones

Imágenes satelitales proveen información muy útil para todo tipo de estadísticas en áreas relacionadas con el territorio, como por ejemplo la agricultura, silvicultura y el estudio de uso del suelo. El estudio de la agricultura a gran escala por medio de la teledetección se realizó por primera vez entre 1974 y 1977 por medio de datos de Landsat 1, a cargo de la NASA, la Oficina Nacional de Administración Oceánica y Atmosférica (NOAA) y el Departamento de Agricultura de los Estados Unidos (USDA). [23]

Dado que las imágenes producidas generalmente cubren toda o casi toda el área de estudio, y que suelen ser multiespectrales, lo que provee datos que fotografías ordinarias no contienen, cualquier

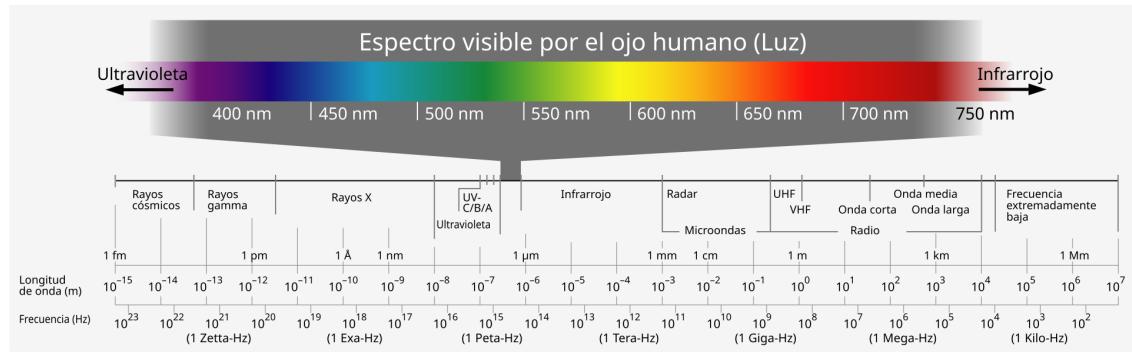


Figura 2: Espectro electromagnético visualizado. Diferentes sustancias y materiales reflejan una variedad de frecuencias más allá del espectro visible, que es relativamente reducido.

aplicación que involucre estudiar un área vasta puede beneficiarse de ellas. Dependiendo de la resolución, aplicaciones que involucren detalles más finos también las pueden aprovechar, como por ejemplo su uso en aplicaciones de mapas digitales.

5.1.2. Características de los datos

La calidad de imágenes recolectadas por teledetección se mide de cuatro formas, estas son su resolución espacial, espectral, radiométrica y temporal.

Resolución espacial: el tamaño de un píxel en una imagen rasterizada. Típicamente corresponde a un área cuadrada de entre 1 y 1000 m^2 .

Resolución espectral: la longitud de onda de las diferentes bandas de frecuencia capturadas, normalmente relacionada a la cantidad de bandas de frecuencia. El sensor Hyperion en “Earth Observing-1”, por ejemplo, observa 220 bandas entre 0,4 y 2,5 μm , con una resolución espectral de 0,10–1.11 μm por banda. [24] En imágenes de espectros no visibles, la visualización se hace con colores falsos, en donde cada banda es asignada un color visible. Un ejemplo se ve en la figura 1.

Resolución radiométrica: la cantidad de niveles de intensidad de radiación detectable por el sensor. Típicamente entre 8 y 14 bits de información, correspondiente a 256 a 16384 niveles en cada banda. La cantidad de ruido en el sensor también afecta la resolución radiométrica.

Resolución temporal: la cantidad de sobrevuelos del avión o satélite, importante solamente cuando se realizan series de tiempo, promedios o mosaicos, como por ejemplo en el monitoreo de la agricultura.

5.1.3. Disponibilidad de recursos

Existen varios repositorios de datos de teledetección disponibles para usos comerciales como académicos. Los programas de observación terrestre de la NASA y de la ESA, Landsat y Copernicus respectivamente, disponibilizan recursos por medio de portales en la internet. Para los datos de Landsat, uno de los recursos más accesibles es Google Earth Engine, que permite el procesamiento de imágenes en línea, de forma gratuita para usos no comerciales. [25] El programa Copernicus por otro lado provee un navegador de imágenes, una forma de descargar datos con algunos filtros, y todo esto de forma gratuita tanto para fines académicos como comerciales. [26] También ofrecen un espacio de trabajo en línea, similar en propósito a Google Earth Engine. [27]

5.2. Redes Neuronales Convolucionales

Una Red Neuronal Convolucional (o CNN por sus siglas en inglés) es un tipo de red neuronal artificial en la cual las neuronas procesan datos de entrada por medio de filtros de convolución. Esto implica el procesamiento de un grupo de datos cercanos, lo que permite interpretar el contexto de un dato, en contraste con redes neuronales típicas. Esta propiedad hace que las CNN sean el método preferido para el procesamiento de imágenes por medio de redes neuronales. [28] [29]

Este método de procesamiento permite procesar una gran cantidad de entradas con una cantidad reducida de neuronas, comparado con una red neuronal típica con la misma capacidad.

Por ejemplo, considerando una red neuronal con una capa de entrada y una capa siguiente con la misma cantidad de neuronas N en ambas, una red neuronal densa, es decir donde cada neurona de una capa está conectada a cada neurona de la siguiente capa, contiene $N \times N$ conexiones. En contraste, una red neuronal convolucional equivalente estaría compuesta de tan solo N neuronas, mientras que al mismo tiempo captura un grupo de píxeles en cada neurona en lugar de uno solo.

Para el procesamiento se utilizan los filtros de convolución, matrices de dimensiones reducidas comparadas con la imagen, cuyas celdas contienen coeficientes. Este filtro se superpone sobre una sección de la imagen, y los valores de los píxeles se multiplican con los de la celda superpuesta del filtro, y la suma de los productos es el resultado de la convolución del grupo de píxeles. Con una representación adecuada de los datos de cada píxel, estos filtros, también llamados *kernels*, pueden usarse en la detección de bordes en cualquier orientación, reducción de ruido, aumentación de intensidad de píxeles de cierto color o brillo, entre otros. [29]

5.2.1. Aplicaciones

CNN ya se han utilizado en todo tipo de aplicaciones relacionadas con imágenes y videos, entre ellas clasificación de imágenes, segmentación de imágenes, detección de objetos e inclusive en análisis de imágenes de inundación para predecir la gravedad de uno de estos tipos de desastre natural. [30]

También se ha usado extensivamente en aplicaciones relacionadas a la teledetección, con una gran colección de técnicas, conjuntos de datos, material de aprendizaje y software de libre acceso en artículos web, videos y repositorios de código. [31] [32] Queda claro que los modelos convolucionales son muy eficaces en el procesamiento de imágenes, y la cantidad de material estudiado relacionado a las imágenes satelitales facilitaría enormemente la aplicación en el tema de este proyecto.

5.2.2. Técnicas y arquitecturas

Las primeras CNN surgieron en los años 90, con LeNet siendo la primera implementación que ganó atención. Esta red se desarrolló para el reconocimiento de dígitos escritos a mano, y consistía de capas convolucionales, de *pooling*, el proceso de reducir la resolución y agrupar información de la capa anterior, y capas densamente, es decir completamente, conectadas.

Recién en 2012 con AlexNet se logró el siguiente salto, con una competencia de reconocimiento visual. AlexNet se diseñó con conjuntos de imágenes de gran escala en mente, compuesta de capas similares a LeNet, con algunas optimizaciones en las funciones de activación y en medidas contra overfitting.

Nuevos desarrollos en los años siguientes se enfocaron en la optimización y la solución de problemas específicos. VGGNet, originando en Oxford, se popularizó por su simplicidad, con kernels pequeños de 3x3 y capas convolucionales en secuencias. Google introdujo GoogLeNet demostrando la efectividad del paralelismo con sus módulos *inception*, que además mejoraron la capacidad de generalización usando kernels de diferentes tamaños al mismo tiempo. Otra arquitectura, Redes Residuales o ResNets, abordaron el desafío de entrenar redes muy profundas por medio de conexiones que saltan una o varias capas, facilitando el entrenamiento de redes de hasta cientos de capas. Otra red diseñada por Google es MobileNet, una arquitectura diseñada para ejecutarse en ambientes de recursos limitados como dispositivos móviles que busca equilibrar el rendimiento con la eficiencia.

Otra arquitectura importante es la U-Net, diseñada para la segmentación de imágenes biomédicas, que funciona mediante reducciones sucesivas de resolución de la imagen de entrada, seguidas por aumentos sucesivos que se combinan con las reducciones respectivas ya procesadas. Esta técnica permite entrenar modelos con mejor segmentación y menos datos de entrenamiento, y con tarjetas gráficas modernas (de 2015 en adelante), procesar una imagen de 512×512 toma menos de un segundo. [33] También se han utilizado U-Nets en aplicaciones de reducción de ruido en imágenes en modelos de difusión, lo cual se sigue utilizando en tecnologías de generación de imágenes como *DALL-E*, *Midjourney* y *Stable Diffusion*. [34] También se ha usado la arquitectura U-Net en

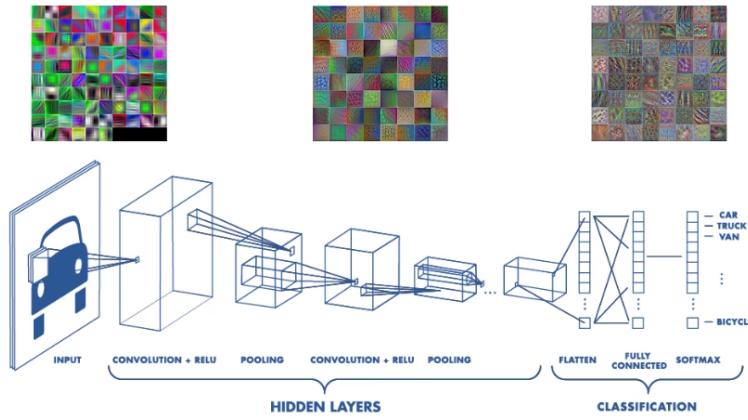


Figura 3: Ejemplo de red neuronal convolucional con varias capas de convolución y pooling, que preprocesan y comprimen la imagen, y varias capas convencionales, densamente conectadas, que hacen el trabajo de clasificación sobre un vector unidimensional.

segmentación de imágenes satelitales para identificar rasgos de imágenes, como recursos de agua, bosques o agricultura con una intersección de entre 81 y 96 % con marcaciones manuales. [35]

5.3. Análisis de imágenes satelitales

Las formas más comunes de análisis de imágenes son la clasificación, la segmentación, la detección de cambios y las series de tiempo. Existen muchas técnicas usadas en aplicaciones más específicas, como la predicción del rendimiento de una plantación o la salud de la vegetación, la reducción de ruido o redes generativas, que no se aplican tan directamente para el objetivo de este trabajo.

5.3.1. Clasificación

La clasificación es una tarea fundamental en el análisis de datos de teledetección, en el cual el objetivo es etiquetar cada imagen, como por ejemplo “área urbana”, “bosque”, “agricultura”, etc. El proceso de asignar etiquetas a imágenes se conoce como clasificación a nivel de imagen. [32]

Sin embargo, en algunos casos una imagen puede contener más de un tipo de uso de suelo, como por ejemplo un bosque con un río que lo divide, o una ciudad con áreas comerciales y residenciales. En estos casos, clasificación a nivel de imagen se vuelve más compleja e implica asignar múltiples etiquetas a cada imagen. Esto se puede lograr por medio de una combinación de extracción de características y algoritmos de *Machine Learning* para identificar los diferentes tipos de uso de suelo. [32]

Es importante no confundir la clasificación a nivel de imagen con la clasificación a nivel de píxel, también conocida como segmentación semántica. Mientras que clasificación a nivel de imagen asigna una etiqueta a una imagen entera, la segmentación semántica asigna una etiqueta a cada píxel de la imagen, lo que resulta en una representación detallada y precisa del uso de suelo en una imagen. [36]

5.3.2. Segmentación

La segmentación consiste en dividir una imagen en segmentos o regiones semánticamente significativas. El proceso de segmentación de imágenes asigna una etiqueta de clase a cada píxel de una imagen, transformándola de una grilla 2D de píxeles a una grilla 2D de etiquetas. Una aplicación común es la segmentación de calles o edificios, donde el objetivo es separar las calles y los edificios de otras características de la imagen. [32]

Para realizar esta tarea, modelos de una clase única son frecuentemente entrenados para detectar y diferenciar entre calles y el ambiente, o edificios y el ambiente. Estos modelos se diseñan para reconocer características específicas como el color, la textura y la forma que son típicas de una calle

o un edificio para que puedan etiquetar los píxeles que forman parte de estas estructuras en una imagen. [36]

Otras aplicaciones comunes se encuentran en la agricultura o clasificación de uso de suelo en una imagen. En este caso, se utilizan modelos multiclasificación que son capaces de diferenciar entre varias clases en una imagen, como por ejemplo bosques, áreas urbanas y tierra agrícola. Estos modelos son capaces de reconocer relaciones más complejas entre tipos de uso de suelo, y permiten un entendimiento más integral del contenido de la imagen. [36]

5.3.3. Detección de cambios

Detección de cambios es un componente vital del análisis de teledetección, permitiendo el monitoreo de cambios de un paisaje a lo largo del tiempo. Esta técnica se puede aplicar para identificar una amplia gama de cambios, entre otros el cambio de uso de suelo, desarrollo urbano, erosión costal y deforestación. [32]

Detección de cambios puede ser realizada entre dos imágenes tomadas en diferentes momentos, o analizando una serie de imágenes tomadas a lo largo de un periodo de tiempo. [32]

Una consideración importante es que la detección de cambios puede verse afectada por la presencia de nubes y sombras. Estos factores dinámicos pueden alterar la apariencia de un paisaje y causar falsos positivos en los resultados. Por ende, es importante considerar estos factores y emplear técnicas que puedan mitigar estos efectos. [32]

5.3.4. Serie de tiempo

La serie de tiempo consiste en una serie de datos ordenados por el tiempo. A menudo se trata de muestras tomadas en intervalos regulares, pero no necesariamente debe ser así. El análisis de series de tiempo se persigue con el fin de extraer estadísticas, patrones o características generales de los datos.

El análisis de series de tiempo en teledetección tiene numerosas aplicaciones, incluyendo mejorar la exactitud de modelos de clasificación y el pronóstico de patrones y eventos futuros, especialmente en la agricultura, por ejemplo en la predicción de la producción de una plantación. [32]

6. Marco Metodológico

En esta sección describimos la forma de la que se realizó el estudio. Se describen la recolección, selección, unificación, preparación y finalmente el análisis de los datos. Esta sección pretende describir un plan que asegure la calidad de los datos a ser analizados.

6.1. Área de estudio

El área de estudio incluye al Parque Nacional Médanos del Chaco en la zona noroeste del Chaco Paraguayo, en los departamentos de Boquerón y Alto Paraguay, y al abanico aluvial del río Pilcomayo, en las fronteras con Bolivia y Argentina.

Los alrededores del Parque Nacional son las áreas de mayor interés, ya que forman parte del área ocupada por una serie de paleocauces originando en el río Parapeti en Bolivia. Sin embargo, el abanico aluvial del Pilcomayo representa un extenso sistema de paleocauces que son más prominentes. Por ende, el área del Pilcomayo puede servir de zona de pruebas de técnicas de análisis de datos antes de aplicarlos a las imágenes del Chaco central.

map figures

6.2. Estrategia de procesamiento de datos

6.2.1. Herramientas utilizadas

Para el procesamiento de datos se utilizó el lenguaje de programación Python con algunas librerías ampliamente disponibles, en especial librerías de Machine Learning, como *PyTorch* [37], *Lightning* [38], y *TorchGeo* [39], y librerías de procesamiento de imágenes satelitales, como *Rasterio* [40]. La función de las librerías utilizadas se detalla en la sección *Tecnologías utilizadas* del capítulo siguiente, *Experimentos y Resultados*.

6.2.2. Recopilación de datos

Las imágenes utilizadas se obtuvieron del programa Copernicus de la ESA, a través de su portal *Copernicus Browser*. Las imágenes son multiespectrales, con 13 bandas capturadas por instrumentos a bordo de los satélites Sentinel-2, un par de satélites en la misma órbita polar desfasados en 180 grados. Estas imágenes son de acceso abierto a todo público, necesitando solamente la creación de un usuario. [41]

Los satélites capturan una franja de 290 kilómetros (km) de ancho con un tiempo de revisita de 5 días en el ecuador, capturando imágenes de resoluciones espaciales de 10, 20 y 60 metros (m). Estas imágenes se disponibilizan particionadas en teselas georeferenciadas de 110km x 110km que solapan con teselas adyacentes, en las cuales las bandas se acceden por medio de archivos individuales en formato JPEG2000 con valores entre 0 y 10'000 para cada píxel. [42]

Existen varios niveles que representan productos diferentes. El producto utilizado para este estudio es el nivel 2A (L2A).

Además de los datos de las imágenes, las teselas incluyen metadatos representados en el cuadro 4. [42]

6.2.3. Selección de archivos

Una vez establecida el área de estudio, se seleccionan las teselas relevantes y se agregan en un conjunto de datos de entrenamiento y de prueba. Como el objetivo es una detección robusta sin importar la temporada, se seleccionan datos de varias fechas distribuidas a lo largo de los años.

Ya que el tiempo de revisita de los satélites Sentinel-2 es corto, es posible ser exigente con la calidad de las imágenes. En este caso, solo se utilizaron imágenes con una cobertura de nubes no mayor que 10 %, lo cual el navegador de imágenes Copernicus Brower permite filtrar.

Las imágenes utilizadas se identifican por su número de tesela, asignado por Copernicus, y la fecha del sobrevuelo. Las imágenes utilizadas se listan en el cuadro 5.

	Metadato	Descripción
1	Extensión	Área geográfica que abarcan los datos. Cada tesela es una imagen en geometría cartográfica (proyección WGS84).
2	Dimensiones	Tamaño de la imagen en píxeles y la cantidad de bandas. Algunas imágenes se proveen en color, es decir contienen mas de una banda.
3	Información espectral	Longitudes de onda y resolución del instrumento para cada banda capturada.
4	Indicadores de calidad	Indicadores de interferencia atmosférica, por ejemplo porcentaje de cobertura de nubes y porcentaje de cobertura de vegetación o agua.

Cuadro 4: Metadatos incluidos con cada tesela del mosaico satelital

Nº de tesela	Fechas (año/mes/día)	Descripción	Función	Imagen
T20KNA	2024/10/03 2025/01/16 2025/06/20 2025/10/03	Abanico aluvial Pilcomayo, frontera Paraguay-Argentina.	Entrenamiento	
T20KNB	2024/10/03 2025/01/16 2025/06/20 2025/10/03	Abanico aluvial Pilcomayo, fronteras Paraguay-Argentina-Bolivia.	Entrenamiento	
T20KNC	2025/09/13	Médanos del Chaco, frontera Paraguay-Bolivia	Prueba	
T20KPC	2024/10/03 2025/01/16 2025/05/13 2025/09/13	Médanos del Chaco, noroeste del departamento Boquerón.	Entrenamiento	
T20KQV	2025/10/20	Ciudad de Neuland, aldeas de la colonia Neuland.	Prueba	
T20KRA	2025/10/20	Ciudades Filadelfia y Loma Plata, aldeas de las colonias Fernheim y Menno.	Prueba	

Cuadro 5: Teselas utilizadas y fechas de sobrevuelo

6.2.4. Unificación y preparación de datos

Seleccionados los datos, estos se mantienen en su formato y estructura de archivos original, de donde pueden ser leídas las imágenes para cada banda disponible para cada resolución (de 10m, 20m o 60m).

Como las estructuras que se buscan reconocer son extensas, basta con utilizar imágenes con una

resolución de 60m por píxel. Esto, además de reducir la potencial complejidad y el tamaño de los modelos que analizan las imágenes, permite utilizar ventanas de mayor área sin correr el riesgo de necesitar de una gran cantidad de recursos de memoria para realizar el análisis. Para la resolución de 60m existen once (11) capas, cada una representada en una imagen de un canal, es decir de un color.

Una clase proveída por la librería Torchgeo, *RasterDataset*, permite cargar imágenes satelitales de diversas bandas teniendo en cuenta su geolocalización. Esto permite un manejo abstracto del conjunto de datos, por lo cual no es necesario una preparación extensa de los datos obtenidos de Copernicus.

6.2.5. Limpieza de datos

Las teselas incluyen máscaras de píxeles erróneos. Como tenemos una gran cantidad de datos disponibles gracias al tiempo corto de revisita de los satélites, podemos simplemente descartar imágenes con significantes errores.

Una inevitabilidad de las imágenes producidas por Sentinel-2 es la falta de datos de una porción de ciertas teselas. Esta es una consecuencia de la división de imágenes en teselas, algunas de las cuales abarcan los bordes de la franja capturada en un sobrevuelo. Los valores de los píxeles en estas regiones son 0.

Como estas regiones faltantes (pero no erróneas) son sustanciales, dependiendo de la tesela escogida, estos píxeles no se modificaron en las imágenes afectadas. Nuestra expectativa fue que los modelos se adecuarían a este valor, ya que los valores posibles abarcan el rango entre 0 y 10'000, pero en imágenes libres de errores, nunca toman valores cercanos a 0. Un ejemplo de este fenómeno se encuentra en el cuadro 5, en el margen izquierdo de la tesela *T20KPC*.

6.2.6. Resumen del procesamiento de los datos

El área de estudio está conformada por los alrededores del parque nacional Médanos del Chaco, y el conjunto de datos de entrenamiento se suplementa con teselas del abanico aluvial del Pilcomayo, ya que existen estudios preexistentes que analizan la ocurrencia de paleocauces en esta zona. Además, algunas teselas de prueba se agregan para comprobar el funcionamiento de los modelos entrenados en áreas que no forman parte del conjunto de entrenamiento.

La fuente de datos principal proviene del programa Copernicus de la Agencia Espacial Europea, que adicionalmente provee una gran cantidad de metadatos y filtros para facilitar la selección de datos que requieren el menor preprocesamiento posible, lo que acelera el análisis de los datos.

7. Experimentos y Resultados

En este capítulo se presentan los experimentos realizados, los modelos y las metodologías utilizadas en su entrenamiento, y los resultados de dichos experimentos.

7.1. Síntesis de datos de entrenamiento

Para las tareas de clasificación y segmentación es necesaria una verdad fundamental, ya sea una o varias etiquetas por cada imagen en el caso de la clasificación, o una máscara en el caso de la segmentación. Este es el objetivo que se quiere obtener a partir del modelo final.

Las imágenes de referencia se sintetizaron a partir de estudios de la zona del abanico aluvial del [ref Pilcomayo](#), las cuales se imitaron a base del mejor esfuerzo. Para facilitar la identificación visual de los paleocaucas, se utilizaron imágenes de color falso, en especial utilizando imágenes de luz infrarroja (IR) de onda corta (SWIR por sus siglas en inglés, short wave infrared) utilizando las bandas B12 (IR de onda corta), B8A (IR cercano) y B4 (verde) en los canales rojo, verde y azul respectivamente. Diferentes tipos de suelo y vegetación reflejan estas bandas de formas variadas, y esta combinación puede resaltar estructuras geológicas como paleocaucas al ojo humano mejor que imágenes de color real. [43]

Para la creación de estas máscaras de entrenamiento se utilizaron herramientas de dibujo digital. Ejemplos de este tipo de programa incluyen *Krita* y *GIMP*.

7.2. Herramientas utilizadas

La herramienta fundamental utilizada en los experimentos es la librería PyTorch [37], que implementa una gran cantidad de componentes básicos de Machine Learning, como las capas que componen a los modelos, los conjuntos de datos de entrenamiento, funciones de activación, funciones de utilidad, entre otros.

La librería Lightning [38] expande sobre esta base, añadiendo una capa de abstracción adicional que provee bucles de entrenamiento y validación y módulos de datos que facilitan la interacción con conjuntos de datos, permitiendo al usuario enfocarse en la arquitectura del modelo y los datos de entrenamiento en lugar del proceso de entrenamiento a bajo nivel.

Finalmente, una capa de abstracción adicional sobre Lightning es proveída por TorchGeo [39], una librería especializada para el análisis de datos de observación terrestre. TorchGeo provee facilidades para la creación de conjuntos de datos que, además de los datos multiespectrales de la imagen propia, tienen en cuenta la geolocalización de cada imagen. También proveen algunos modelos, opcionalmente preentrenados.

Para la combinación de imágenes georreferenciadas, o para la inclusión de datos de georreferenciación a imágenes que no los tienen, se utilizó la librería Rasterio [40].

Los experimentos se realizaron exclusivamente en hardware de consumidor. Se utilizó una computadora portátil con un procesador de 12 núcleos lógicos, 16GiB de memoria RAM, y una tarjeta gráfica Nvidia GeForce GTX 1050 Ti, que cuenta con 4GiB de memoria de video. Cabe resaltar que los experimentos se vieron limitados por esta máquina en su envergadura, sin embargo son reproducibles en hardware más actual.

7.3. Partición de datos de entrenamiento

El conjunto de datos en PyTorch también se conoce como *Dataset*, y el componente que disponibiliza estos datos como *DataLoader*. Lightning provee una abstracción, el *DataModule*, que abstractea al Dataset y permite proveer DataLoaders distintos para las tareas de entrenamiento, validación, prueba y inferencia.

Para la división de datos en datasets de entrenamiento, validación y prueba, se utilizaron métodos proveídos por TorchGeo. El procedimiento consiste en dividir los datos de entrenamiento en celdas, las cuales se asignan pseudoaleatoriamente a subconjuntos de entrenamiento, validación y

ref

figure of swir image and mask

add software and cuda versions in a table

diagram of random grid pitting against inference

prueba en una proporción de 70/20/10 respectivamente. Para cada subconjunto, luego, definimos un DataLoader que provee acceso a los datos.

Las celdas en las cuales se dividen las imágenes no se solapan, y la estrategia de cada DataLoader puede ser elegida de acuerdo al objetivo de la etapa a la cual corresponde. Para el entrenamiento usamos un muestreo aleatorio, mientras que los DataLoaders de validación y de prueba hacen un muestreo por división en celdas, asegurando que toda el área de estos conjuntos se muestrean una sola vez y que el conjunto de imágenes creado es consistente para cada tanda de entrenamiento y para cada modelo entrenado sobre el conjunto.

Cada DataLoader produce imágenes de 128×128 píxeles recortadas de las celdas asignadas al DataLoader, correspondiente a un área de $7680\text{m} \times 7680\text{m}$. Las imágenes de cada tesela tienen dimensiones de 1830×1830 píxeles de $60\text{m} \times 60\text{m}$, abarcando un área de $109'800\text{m}$ por lado, o alrededor de 110km. La grilla en la que se dividen es de seis celdas de largo, abarcando cada celda un área cuadrada de 305 píxeles de largo, lo que permite producir 4 imágenes de validación o prueba por celda.

7.4. Entrenamiento y validación

clasificación

Para la tarea de segmentación, se utilizaron tres arquitecturas bien establecidas. Las arquitecturas proveídas por TorchGeo utilizadas fueron Foreground-Aware Relation Network, o FarSeg [44], y una red enteramente convolucional de cinco capas conocida simplemente como FCN por sus siglas en inglés, Fully-convolutional Network. Adicionalmente se utilizó una implementación de U-Net, una red enteramente convolucional muy conocida por sus aplicaciones en el campo de la medicina, pero tambien muy capaz en imágenes de teledetección. [33, 35]

Lightning provee un componente llamado Trainer, que implementa bucles de entrenamiento, validación y prueba, y toma de entrada un modelo, un conjunto de datos, y parámetros de entrenamiento. Algunos parámetros son la cantidad mínima y máxima de épocas de entrenamiento, donde una época es una iteración del entrenamiento sobre el dataset, y una condición de detención temprana del entrenamiento si no mejora a lo largo de una cierta cantidad de tiempo.

El Trainer también brinda la posibilidad de definir un *checkpoint*, es decir un punto de control, que se encarga de mantener una copia de los parámetros del modelo en el punto en el que muestra el mejor rendimiento a lo largo del entrenamiento. Este checkpoint es la forma en la que el Trainer provee el modelo entrenado para luego ser usado en tareas de inferencia.

Más allá de estos procedimientos simples, la mayoría del esfuerzo fue dedicada a la creación del conjunto de datos etiquetados y a la verificación de los resultados brindados por el entrenamiento sobre estos.

7.5. Resultados

Para todos los experimentos, las únicas variables son los modelos usados y sus parámetros configurables, también llamados hiperparámetros. Los parámetros del Trainer y del DataModule, así como la semilla del generador de números pseudoaleatorios (PRNG) son iguales para todos los experimentos, y se detallan en el cuadro 6.

Debido a limitaciones de hardware, el tamaño de lote es muy pequeño. Equivale a $128 \times 128 \times 11 \times 3 = 540'672$ píxeles de 4 bytes por lote, o 2,1MiB, cuando se usan las once bandas disponibles, y un número mayor causó errores de memoria al momento de realizar los experimentos.

Para cada arquitectura, se realizaron experimentos con diferentes hiperparámetros, los cuales se detallan a continuación, además de resultados concretos en la forma de valores de la función de pérdida para el conjunto de prueba.

También se demuestran resultados en teselas nuevas, no utilizadas en el DataModule de entrenamiento, para demostrar las predicciones en datos completamente desconocidos. Estas teselas son aquellas cuya función es “Prueba” en el cuadro 5 del capítulo anterior.

Para la visualización de las predicciones, se convierten las matrices de salida al formato de imagen georreferenciada GeoTiff. Las predicciones de los modelos son matrices de números decimales, para

Parámetro	Valor	Descripción
Nro. mínimo de épocas	1	Cantidad mínima de iteraciones por el conjunto de datos de entrenamiento.
Nro. máximo de épocas	50	Cantidad máxima de iteraciones por el conjunto de datos de entrenamiento.
Paciencia	7	Cantidad de épocas en las que el modelo no mejora la pérdida de validación antes de detener el entrenamiento.
Semilla PRNG	42	Valor inicial del estado del generador de números semialeatorios, empleado en, entre otras funciones, la creación de los valores iniciales de los diversos parámetros entrenables del modelo.
Tamaño de época	1200	Tamaño efectivo del conjunto de datos de entrenamiento. Cantidad de imágenes, recortadas al azar de las celdas del dataset de entrenamiento, usadas para el entrenamiento en cada época.
Tamaño de imagen	128	Dimensión de cada recorte de los datasets en píxeles. Los recortes son cuadrados.
Tamaño de lote (batch)	3	Cantidad de imágenes cargadas y procesadas en cada paso de entrenamiento. Subdivide a cada época.

Cuadro 6: Parámetros de entrenamiento generales

los cuales un valor cercano a 0 o mayor que 1,0 significa una predicción de ocurrencia de paleocauce en ese píxel, mientras que un valor cercano a 0 o menor que 0,0 significa una predicción de ausencia de paleocauce en ese píxel. Estos valores se fijan al intervalo [0, 1], con valores menores que 0 y mayores que 1 siendo convertidos a 0 y 1 respectivamente. La matriz resultante es la máscara de predicciones para una celda de 128×128 píxeles, y cada píxel se convierte a un valor en la escala de grises, con 0 y 1 correspondiendo a negro y blanco respectivamente.

Estas celdas de 128×128 píxeles se combinan para formar un mosaico de predicciones. Para mitigar los artefactos que pueden surgir en los bordes de estas celdas, estas se solapan ligeramente. Se toman pasos de 100 y 112 píxeles, se combinan mosaicos de cada tamaño de paso por separado, y finalmente se toma un promedio de ambos mosaicos para generar una máscara de predicciones. El diagrama

make diagram

muestra el procedimiento de forma visual.

7.5.1. FCN

La arquitectura más simple en este trabajo, consiste de cinco capas convolucionales con activación de tipo *LeakyReLU*, la cual mantiene valores positivos iguales y multiplica valores negativos por un número pequeño para minimizar su magnitud.

Los hiperparámetros seleccionados para los experimentos se detallan en el cuadro 7.

Versión	Canales de entrada	Nro. de filtros	Mejor época	Pérdida en prueba
1	Todos (11)	64	1	0.4675
2	Todos (11)	96	2	0.4678
3	Todos (11)	128	8	0.4721
4	Todos (11)	256	4	0.4638
5	Todos (11)	256	4	0.4630

Cuadro 7: Hiperparámetros y resultados de la arquitectura FCN

Una arquitectura así de simple necesita de patrones simples y obvios para ser útil. Tal vez prede-

ciblemente, esta arquitectura no dio resultados útiles, creando predicciones uniformes para cualquier tesela. Todos los experimentos resultan en un modelo que produce una máscara de probabilidades en la que cada píxel es negro, y la cantidad de épocas necesarias para alcanzar la menor pérdida sugiere que no se aprendieron patrones entre iteraciones.

7.5.2. FarSeg

La arquitectura Foreground-Aware Relation Network es una arquitectura diseñada para la segmentación geoespacial de objetos en imágenes de alta resolución espacial [44]. Mientras que solamente utilizamos imágenes de 60m de resolución espacial, dieron resultados mucho más prometedores que FCN. Estos se muestran en el cuadro 8, además de los hiperparámetros utilizados.

La FarSeg utiliza una arquitectura de red neuronal en lo que se conoce como *Backbone*, o columna vertebral, en este caso ResNet [45]. Esta columna cumple la función de extractor de características, las cuales son utilizadas por el resto del modelo para generar una predicción. Existe la opción de usar parámetros preentrenados para el Backbone, pero en estos experimentos se hizo el entrenamiento completo sin parámetros preentrenados.

Como la arquitectura FarSeg disponible a través de TorchGeo solo acepta tres canales de entrada, se utilizaron los conjuntos de bandas B4, B3, B2 (rojo, verde, azul) y B12, B8A, B4 (infrarrojo de onda corta o SWIR, infrarrojo de onda cercana o NIR, rojo).

try pre-trained

Versión	Canales de entrada	Backbone	Mejor época	Pérdida en prueba
1	B4, B3, B2	ResNet-18	1	0.6560
2	B12, B8A, B4	ResNet-18	3	0.5866
3	B4, B3, B2	ResNet-34	11	0.4682
4	B12, B8A, B4	ResNet-34	7	0.5324
5	B4, B3, B2	ResNet-50	19	0.4514
6	B12, B8A, B4	ResNet-50	2	1.444
7	B4, B3, B2	ResNet-50	27	0.4977
8	B12, B8A, B4	ResNet-50	4	1.056
9	B4, B3, B2	ResNet-101	12	0.5897
10	B12, B8A, B4	ResNet-101	10	0.9042
11	B4, B3, B2	ResNet-101	8	0.6233
12	B12, B8A, B4	ResNet-101	5	0.5351

Cuadro 8: Hiperparámetros y resultados de la arquitectura FarSeg

Mientras que el uso de backbone más simples no dio buenos resultados, los modelos más grandes dieron resultados significantes. Interesantemente, los modelos con backbone ResNet-50 y entrenados sobre las bandas de color visible tomaron, en promedio, muchas más épocas para alcanzar su mejor rendimiento que los demás.

Entre los resultados, los modelos con backbone ResNet-18 y ResNet-101 produjeron máscaras completamente negras, o con pocas áreas en las que predicen paleocauces. Sin embargo, aun más llamativo es el hecho que los modelos que utilizan las bandas B12, B8A, B4 producen máscaras enteramente negras, excepto con backbone ResNet-34.

Las máscaras producidas por los modelos que utilizan las bandas B4, B3, B2 presentan bordes de apariencia borrosa, lo que podría ser una consecuencia de la resolución espacial de las imágenes utilizadas.

Las máscaras producidas por los modelos entrenados con las bandas B12, B8A, B4 son similares a las producidas por FCN. Cada píxel de la máscara tiene el valor mínimo, y las pérdidas de las mejores versiones son mayores que la mayor versión de FCN.

7.5.3. U-Net

Una de las arquitecturas más conocidas para la tarea de segmentación es U-Net. La estructura consiste de varios niveles por los cuales desciende la imagen de entrada hasta llegar a un nivel de cuello de botella, de donde asciende nuevamente por los niveles hasta llegar a la resolución de la imagen de salida, que en este caso es igual a la resolución de la imagen de entrada.

En cada descenso a un nivel inferior, los datos crecen en la cantidad de canales y decrecen en la resolución de cada canal, es decir, la imagen disminuye en su tamaño. Inversamente, al ascender a un nivel superior, disminuye la cantidad de canales e incrementa la resolución de cada canal. El cuello de botella es el último nivel, en el cual la resolución es mínima y la cantidad de canales es máxima.

En adición a las conexiones entre niveles adyacentes, existen conexiones entre los caminos descendientes y ascendientes del mismo nivel, saltando niveles en estas conexiones.

Cada nivel tiene un tamaño que describe la cantidad de filtros, y debe ser el doble que el nivel anterior o la mitad del nivel siguiente. Cada nivel disminuye y aumenta la resolución de la imagen por un factor de 4, o sea un factor de 2 por cada dimensión.

Los hiperparámetros utilizados y los resultados de los experimentos se visualizan en el cuadro 9.

Versión	Canales de entrada	Niveles	Mejor época	Pérdida en prueba
1	Todos (11)	64, 128, 256, 512	10	0.4421
2	Todos (11)	64, 128, 256, 512	10	0.4576
3	Todos (11)	32, 64, 128, 256, 512	14	0.4535
4	Todos (11)	32, 64, 128, 256, 512	3	0.4520
5	Todos (11)	64, 128, 256, 512, 1024	10	0.5412
6	Todos (11)	64, 128, 256, 512, 1024	10	0.4426

Cuadro 9: Hiperparámetros y resultados de la arquitectura U-Net

La pérdida de estos modelos es en promedio menor que la pérdida promedio de los modelos anteriores, y las máscaras generadas son mucho más detalladas.

Entre las tres arquitecturas, la implementada por las versiones 3 y 4 producen las predicciones más parecidas a las máscaras de entrenamiento, con grandes áreas incluidas que las demás arquitecturas no capturan. La arquitectura menos compleja, de las versiones 1 y 2, produce predicciones cuyas áreas son menos extensas, pero cuyos bordes son más pronunciados. El peor resultado es el que producen las versiones 5 y 6, con una máscara mucho más ruidosa que la de las versiones 1 y 2, y cubriendo un menor área que la de las versiones 3 y 4.

7.5.4. Predicción en conjunto

Un problema con las máscaras generadas por los modelos es la cantidad de artefactos y diferencias en los resultados, aun con la misma arquitectura y la misma distribución de datos de entrenamiento. Por esta razón, también experimentamos con la creación de máscaras usando un conjunto de modelos, cuyos resultados fueron promediados para generar una sola máscara. Esto permite generar mapas de calor más consistentes, en los cuales los puntos ciegos de un modelo pueden ser cubiertos por los puntos fuertes de otro.

Aun con las diferencias entre los modelos, es común encontrar zonas en las cuales ningún modelo predice ocurrencias de paleocauces, mientras que en otras zonas la mayoría predice positivamente con alta certeza.

Sin embargo, se pueden observar varias áreas en las cuales parecen ocurrir paleocauces, pero en donde los modelos no predicen ocurrencias. Esto se puede atribuir a una cantidad limitada de datos de entrenamiento de alta calidad, lo que sería una forma de expandir sobre estos resultados.

figures of test plots and generated masks in appendix

try predictions with the same tile from a different day

figures of generated masks for training and test tiles

8. Discusión de los Resultados

- 8.1. Conclusión
- 8.2. Recomendaciones
- 8.3. Contribución
- 8.4. Trabajos futuros

discusion
de los re-
sultados

9. Apéndice

apendice

Referencias

- [1] Global Forest Link. *How does satellite imaging work?* Disponible en <https://globalforestlink.com/how-does-satellite-imaging-work/> (accedido 2024-06-24).
- [2] Earth Data. *Vegetation.* Disponible en <https://www.earthdata.nasa.gov/learn/find-data/near-real-time/hazards-and-disasters/vegetation> (accedido 2024-06-24).
- [3] Catherine Stepanova. *Ricardo Rodríguez: "Gracias a la aplicación de OneSoil ahorramos USD 100 por hectárea".* Disponible en <https://blog.onesoil.ai/es/interview-with-ricardo-rodriguez> (accedido 2024-06-24).
- [4] S.M. White et al. "Small Muddy Paleochannels and Implications for Submarine Groundwater Discharge near Charleston, South Carolina, USA. Geosciences". En: 13.232 (2023).
- [5] Antero J. N. Cabrera et al. "Sistemas de Captación y Almacenamiento de Agua en el Chaco Central". En: (2021), pág. 7. URL: https://www.conacyt.gov.py/sites/default/files/upload_editores/u454/MANUAL-SISTEMAS-CAPTACION-ALMACENAMIENTO-AGUA-CHACO.pdf.
- [6] Vladimir Iglovikov, Sergey Mushinskiy y Vladimir Osin. "Satellite Imagery Feature Detection using Deep Convolutional Neural Network: A Kaggle Competition". En: (jun. de 2017). URL: <https://arxiv.org/abs/1706.06169>.
- [7] Martin Längkvist et al. "Classification and Segmentation of Satellite Orthoimagery Using Convolutional Neural Networks". En: (2016). DOI: <https://doi.org/10.3390/rs8040329>. URL: <https://www.mdpi.com/2072-4292/8/4/329>.
- [8] Mariel Samanta Luengo et al. "Identificación y Mapeo de Paleocauces utilizando Imágenes Satelitales de Alta Resolución en la Llanura Costera de la Bahía Samborombón, Este de la Provincia de Buenos Aires, Argentina." En: (2016). URL: https://www.researchgate.net/publication/325827963_IDENTIFICACION_Y_MAPEO_DE_PALEOCAUACES_UTILIZANDO_IMAGENES_SATELITALES_DE_ALTA_RESOLUCION_EN_LA_LLANURA_COSTERA_DE_LA_BAHIA_SAMBOROMBON_ESTE_DE_LA_PROVINCIA_DE_BUENOS_AIRES_ARGENTINA.
- [9] Emmanuel Maggiori et al. "Fully convolutional neural networks for remote sensing image classification". En: (2016). DOI: <https://doi.org/10.1109/IGARSS.2016.7730322>. URL: <https://ieeexplore.ieee.org/abstract/document/7730322>.
- [10] Igor Ševo y Aleksej Avramović. "Convolutional Neural Network Based Automatic Object Detection on Aerial Images". En: (2016). DOI: <https://doi.org/10.1109/LGRS.2016.2542358>. URL: <https://ieeexplore.ieee.org/abstract/document/7447728>.
- [11] Yanfei Zhong et al. "SatCNN: satellite image dataset classification using agile convolutional neural networks". En: (2016). DOI: <https://doi.org/10.1080/2150704X.2016.1235299>. URL: <https://www.tandfonline.com/doi/abs/10.1080/2150704X.2016.1235299>.
- [12] Atharva Sharma et al. "A patch-based convolutional neural network for remote sensing image classification". En: (2017). DOI: <https://doi.org/10.1016/j.neunet.2017.07.017>. URL: <https://www.sciencedirect.com/science/article/abs/pii/S0893608017301806>.
- [13] Mark Pritt y Gary Chern. "Satellite Image Classification with Deep Learning". En: (2017). DOI: <https://doi.org/10.1109/AIPR.2017.8457969>. URL: <https://ieeexplore.ieee.org/abstract/document/8457969>.
- [14] Mohammad Rezaee et al. "Deep Convolutional Neural Network for Complex Wetland Classification Using Optical Remote Sensing Imagery". En: (2018). DOI: <https://doi.org/10.1109/JSTARS.2018.2846178>. URL: <https://ieeexplore.ieee.org/abstract/document/8401505>.
- [15] Qun Liu et al. "DeepSat V2: feature augmented convolutional neural nets for satellite image classification". En: (2019). DOI: <https://doi.org/10.1080/2150704X.2019.1693071>. URL: <https://www.tandfonline.com/doi/abs/10.1080/2150704X.2019.1693071>.

- [16] Eleonora Amato et al. “A Deep Convolutional Neural Network for Detecting Volcanic Thermal Anomalies from Satellite Images”. En: (2023). DOI: <https://doi.org/10.3390/rs15153718>. URL: <https://www.mdpi.com/2072-4292/15/15/3718>.
- [17] Xiaodong Xu et al. “Multisource Remote Sensing Data Classification Based on Convolutional Neural Network”. En: (2017). DOI: <https://doi.org/10.1109/TGRS.2017.2756851>. URL: <https://ieeexplore.ieee.org/abstract/document/8068943>.
- [18] Anastasiia Safonova et al. “Ten deep learning techniques to address small data problems with remote sensing”. En: (2023). DOI: <https://doi.org/10.1016/j.jag.2023.103569>. URL: <https://www.sciencedirect.com/science/article/pii/S156984322300393X>.
- [19] Emmanuel Maggiori et al. “Convolutional Neural Networks for Large-Scale Remote-Sensing Image Classification”. En: (2016). DOI: <https://doi.org/10.1109/TGRS.2016.2612821>. URL: <https://ieeexplore.ieee.org/abstract/document/7592858>.
- [20] Marco Castelluccio et al. “Land Use Classification in Remote Sensing Images by Convolutional Neural Networks”. En: (2015). DOI: <https://doi.org/10.48550/arXiv.1508.00092>. URL: <https://arxiv.org/abs/1508.00092>.
- [21] Keiller Nogueira, Otávio A.B. Penatti y Jefersson A. dos Santos. “Towards better exploiting convolutional neural networks for remote sensing scene classification”. En: (2017). DOI: <https://doi.org/10.1016/j.patcog.2016.07.001>. URL: <https://www.sciencedirect.com/science/article/abs/pii/S0031320316301509>.
- [22] European Space Agency. *50 years of Earth Observation*. Disponible en https://web.archive.org/web/20120130034957/http://www.esa.int/esaMI/Space_Year_2007/SEMP4FEVL2F_0.html (accedido 2024-07-30).
- [23] J. Donald Allen. “A Look at the Remote Sensing Applications Program of the National Agricultural Statistics Service”. En: *Journal of Official Statistics* 6.4 (1990), págs. 393-409.
- [24] Holli Riebeek. *Earth Observing-1: Ten Years of Innovation*. Disponible en <https://web.arc-hive.org/web/20220418062231/https://earthobservatory.nasa.gov/features/E01Ten-th> (accedido 2024-07-30).
- [25] Landsat Science. *Data Access*. Disponible en <https://landsat.gsfc.nasa.gov/data/data-access/> (accedido 2024-08-03).
- [26] Copernicus. *Copyright and licences*. Disponible en <https://www.copernicus.eu/en/access-data/copyright-and-licences> (accedido 2024-08-03).
- [27] Copernicus. *About the Copernicus Data Space Ecosystem*. Disponible en <https://dataspace.copernicus.eu/about> (accedido 2024-08-03).
- [28] Aurélien Géron. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*. Sebastopol, CA: O'Reilly Media, 2019.
- [29] Dan Ciresan et al. “Flexible, High Performance Convolutional Neural Networks for Image Classification”. En: *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence* 2 (2011), págs. 1237-1242.
- [30] R.J. Pally y S. Samadi. “Application of image processing and convolutional neural networks for flood image classification and semantic segmentation”. En: *Environmental Modelling & Software* 148 (2022). DOI: <https://doi.org/10.1016/j.envsoft.2021.105285>. URL: <https://www.sciencedirect.com/science/article/pii/S1364815221003273>.
- [31] Abdishakur. *Land use/Land cover classification with Deep Learning*. Disponible en <https://towardsdatascience.com/land-use-land-cover-classification-with-deep-learning-9a5041095ddb> (accedido 2024-08-10). Ago. de 2018.
- [32] Robin Cole y Mikolaj Czerkawski. Disponible en <https://github.com/satellite-image-deep-learning/techniques> (accedido 2024-08-10).

- [33] Olaf Ronneberger, Philipp Fischer y Thomas Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. 2015. arXiv: 1505.04597 [cs.CV]. URL: <https://arxiv.org/abs/1505.04597>.
- [34] Jonathan Ho, Ajay Jain y Pieter Abbeel. “Denoising Diffusion Probabilistic Models”. En: *arXiv preprint arxiv:2006.11239* (2020).
- [35] Vladimir Khryashchev et al. “Convolutional Neural Network for Satellite Imagery”. En: *FRUCT Conference 22* (2018). URL: <https://fruct.org/publications/volume-22/acm22/files/Khr.pdf>.
- [36] Robin Cole. *A brief introduction to satellite image segmentation with neural networks*. Disponible en <https://medium.com/@robmarkcole/a-brief-introduction-to-satellite-image-segmentation-with-neural-networks-33ea732d5bce> (accedido 2024-09-01). 2023.
- [37] Jason Ansel et al. “PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation”. En: *29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS '24)*. ACM, abr. de 2024. DOI: 10.1145/3620665.3640366. URL: <https://docs.pytorch.org/assets/pytorch2-2.pdf>.
- [38] William Falcon y The PyTorch Lightning team. *PyTorch Lightning*. Ver. 1.4. Mar. de 2019. DOI: 10.5281/zenodo.3828935. URL: <https://github.com/Lightning-AI/lightning>.
- [39] Adam J. Stewart et al. “TorchGeo: Deep Learning With Geospatial Data”. En: *ACM Transactions on Spatial Algorithms and Systems* (dic. de 2024). DOI: 10.1145/3707459. URL: <https://doi.org/10.1145/3707459>.
- [40] Sean Gillies et al. *Rasterio: geospatial raster I/O for Python programmers*. Mapbox, 2013-. URL: <https://github.com/rasterio/rasterio>.
- [41] *Sentinel-2*. Disponible en <https://dataspace.copernicus.eu/data-collections/copernicus-sentinel-data/sentinel-2> (accedido 2025-11-27).
- [42] *S2 Products*. Disponible en <https://sentiwiki.copernicus.eu/web/s2-products> (accedido 2025-11-27).
- [43] Holli Riebeek. *Why is that Forest Red and that Cloud Blue? How to Interpret a False-Color Satellite Image*. Disponible en <https://earthobservatory.nasa.gov/features/FalseColor> (accedido 2025-11-24).
- [44] Zhuo Zheng et al. *Foreground-Aware Relation Network for Geospatial Object Segmentation in High Spatial Resolution Remote Sensing Imagery*. 2020. arXiv: 2011.09766 [cs.CV]. URL: <https://arxiv.org/abs/2011.09766>.
- [45] Kaiming He et al. *Deep Residual Learning for Image Recognition*. 2015. arXiv: 1512.03385 [cs.CV]. URL: <https://arxiv.org/abs/1512.03385>.