

# **Sistema de monitoreo de *Rhynchophorus Ferrugineus* en palmeras de la ciudad de Montevideo**

Ing. Bruno Masoller

**Carrera de Especialización en Inteligencia Artificial**

**Director:** Ing. Juan Ignacio Cavalieri (FIUBA)

**Jurados:**

Jurado 1 (pertenencia)

Jurado 2 (pertenencia)

Jurado 3 (pertenencia)

*Ciudad de Montevideo, junio de 2025*



## *Resumen*

Esta memoria describe la implementación de una plataforma para la intendencia de Montevideo que utiliza visión por computadora para detectar la plaga del picudo rojo en las palmeras esta ciudad. El sistema se desarrolló con el objetivo de reducir costos operativos y mejorar la toma de decisiones para el servicio de arbolado de la institución.

Para el desarrollo del trabajo fueron necesarios conocimientos de visión por computadora, análisis de datos y aprendizaje profundo, así como de despliegue y operaciones de modelos de aprendizaje de máquinas e infraestructura de soporte.



## *Agradecimientos*

Esta sección es para agradecimientos personales y es totalmente **OPCIONAL**.



# Índice general

<b>Resumen</b>	<b>I</b>
<b>1. Introducción general</b>	<b>1</b>
1.1. Descripción del problema . . . . .	1
1.2. Estado del arte . . . . .	3
1.2.1. Redes convolucionales . . . . .	3
1.2.2. Detección del picudo rojo en imágenes aéreas . . . . .	4
1.3. Objetivos y alcance . . . . .	6
<b>2. Introducción específica</b>	<b>7</b>
2.1. Imágenes generadas por vehículos aéreos . . . . .	7
2.1.1. Imágenes de drones . . . . .	8
2.2. Herramientas de soporte en utilizadas en IA . . . . .	8
2.2.1. Gestión de código . . . . .	8
2.2.2. Gestión de infraestructura . . . . .	8
2.2.3. Gestión de usuarios y permisos . . . . .	9
2.2.4. Gestión de etiquetado de datos . . . . .	9
2.2.5. Gestión de repositorio de datos . . . . .	9
2.3. Modelos de visión por computadora . . . . .	10
2.3.1. Clasificación de objetos . . . . .	10
2.3.2. Detección de objetos . . . . .	10
Modelos de dos etapas . . . . .	10
Modelos de una etapa . . . . .	11
2.4. Métricas en visión por computadora . . . . .	11
2.4.1. Precisión, Recuperación y <i>F1-Score</i> . . . . .	11
2.4.2. Intersección sobre Unión (IOU) . . . . .	12
2.4.3. <i>Average Precision</i> (AP) y <i>Mean Average Precision</i> (mAP) . . . . .	12
2.4.4. Otras métricas relevantes . . . . .	12
<b>3. Diseño e implementación</b>	<b>15</b>
3.1. Análisis del software . . . . .	15
<b>4. Ensayos y resultados</b>	<b>17</b>
4.1. Pruebas funcionales del hardware . . . . .	17
<b>5. Conclusiones</b>	<b>19</b>
5.1. Conclusiones generales . . . . .	19
5.2. Próximos pasos . . . . .	19
<b>Bibliografía</b>	<b>21</b>





# Índice de figuras

1.1. Picudo rojo <sup>1</sup> . . . . .	1
1.2. Infección y muerte de palmeras por el picudo rojo. . . . .	2
1.3. Ejemplo de ortomosaico <sup>2</sup> . . . . .	3
1.4. Red convolucional VGG <sup>3</sup> . . . . .	4
1.5. Proceso de detección realizado por Kagan et al <sup>4</sup> . . . . .	5
2.1. Matriz de confusion <sup>5</sup> . . . . .	12



# Índice de tablas

1.1. <a href="#">comparativa-papers</a> . . . . .	6
---	---



***Dedicado a... [OPCIONAL]***



# Capítulo 1

## Introducción general

En este capítulo se presenta una introducción general al trabajo realizado. Se describe el problema que el *Rhynchophorus Ferrugineus* (picudo rojo) presenta en las palmeras de la ciudad de Montevideo, el estado del arte en cuanto a trabajos similares y los objetivos planteados por la Intendencia de Montevideo (IM) y por el equipo de trabajo.

### 1.1. Descripción del problema

El picudo rojo, que puede observarse en la figura 1.1, es un insecto que afecta a las palmeras, especialmente a la *Phoenix Canariensis*, que es la especie más común en Montevideo. Este insecto ha causado un daño significativo en la flora de la ciudad, lo que ha llevado a la IM a enfocarse en su control y erradicación.



FIGURA 1.1. Picudo rojo <sup>1</sup>.

Este escarabajo supone una amenaza grave para las palmeras, ya que una vez infectada, como puede verse en la figura 1.2a, sus larvas se alimentan de su tejido interno, causando el colapso estructural de estas plantas en un período de meses, dado su corto ciclo de vida [1]. Esta amenaza aumenta según la época del año, ya que el insecto tiene diferentes tasas de dispersión y reproducción dependiendo de la temperatura y la humedad. Existen varios tipos de picudos [2], algunos autóctonos y otros introducidos.

La plaga del picudo rojo llegó a Uruguay en 2022 [3], esparciéndose rápidamente por la ciudad de Montevideo. De las 25.000 palmeras que forman una parte

---

<sup>1</sup>Imagen tomada de [https://es.wikipedia.org/wiki/Rhynchophorus\\_ferrugineus](https://es.wikipedia.org/wiki/Rhynchophorus_ferrugineus)

esencial de la ciudad, muchas ya han sucumbido a la plaga, donde se estima que para el año 2030 el ecosistema se verá ampliamente afectado [4] si no se toman medidas de control adecuadas. Últimamente también se ha esparcido por el interior del país, como puede observarse en la figura 1.2b, perteneciente a la ruta 5 de Uruguay, donde se han encontrado palmeras muertas por la plaga.



(A) Palmera infectada por el picudo rojo.

(B) Palmeras muertas en la ruta 5 de Uruguay.

FIGURA 1.2. Infección y muerte de palmeras por el picudo rojo.

Entre los métodos de control fitosanitarios que se utilizan para combatir la plaga, se encuentran endoterapia [5], baño, cirugía [6] y remoción. Sin embargo, estos métodos son costosos y requieren de un monitoreo constante para detectar la presencia del picudo rojo. Para la IM no es solamente una cuestión ecológica sino también económica. En este sentido, el servicio de arbolado realiza un seguimiento de las palmeras afectadas por la plaga. Para ello, se registra su ubicación y estado de salud mediante campañas de detecciones a pie. Este proceso manual requiere de mucho tiempo y recursos humanos, por lo que resulta imprescindible un sistema automatizado que permita detectar la presencia de la plaga en lugares específicos de Montevideo.

Uno de los recursos aún no explotados por la IM para el monitoreo de la plaga es el uso de imágenes aéreas obtenidas mediante drones, disponibles por el servicio de geomática de esta institución. Estos ortomosaicos, que se pueden observar en la figura 1.3, permiten obtener información detallada sobre el estado de las palmeras y su entorno.



FIGURA 1.3. Ejemplo de ortomosaico <sup>2</sup>.

El análisis de estas imágenes es un proceso complejo que requiere de técnicas avanzadas de procesamiento de imágenes y aprendizaje automático, así como también herramientas que brinden soporte a estas actividades.

## 1.2. Estado del arte

En los últimos años, el campo de la visión por computadora (VPC) ha experimentado un crecimiento exponencial, impulsado principalmente por los avances en el aprendizaje profundo y las redes neuronales convolucionales (CNN). Estas tecnologías han permitido el desarrollo de soluciones innovadoras para tareas de detección, clasificación y segmentación en imágenes, lo que ha abierto nuevas posibilidades para aplicaciones en diversos ámbitos, tales como la monitorización ambiental y la agricultura de precisión. La detección de plagas mediante imágenes aéreas, en particular la del picudo rojo, ha cobrado relevancia debido al creciente impacto que este insecto tiene en la salud de las palmeras urbanas, principalmente en países del Mediterráneo y en el Medio Oriente [2]. En esta sección se revisan las principales técnicas y enfoques que han marcado el estado del arte en la detección de plagas a partir de imágenes aéreas. Se aborda tanto arquitecturas tradicionales como las más recientes, destacando sus fortalezas, limitaciones y la evolución de sus aplicaciones en escenarios reales.

### 1.2.1. Redes convolucionales

Visión por computadora se define como el proceso en donde se extrae, analiza y comprende información significativa a partir de imágenes o secuencias de imágenes [7]. Este proceso abarca una amplia gama de tareas, tales como la detección de objetos, la segmentación de imágenes, el reconocimiento de patrones y la clasificación de imágenes. Dentro del ámbito de la inteligencia artificial (IA), la VPC se reconoce como un subcampo destinado a dotar a las máquinas de la capacidad de interpretar y comprender la información visual del entorno de manera análoga a la percepción humana [8]. Asimismo, esta tecnología ha adquirido un

---

<sup>2</sup>Imagen tomada del sistema de información geográfica de la IM (SIG): <https://sig.montevideo.gub.uy/>

papel preponderante en diversas industrias, al contribuir de forma significativa a la optimización de la eficiencia operativa y a la mejora de los procesos de toma de decisiones.

El notable desarrollo de la VPC ha sido posible gracias a los avances en el campo del aprendizaje automático, en particular mediante la adopción de metodologías de aprendizaje profundo. Estas técnicas han permitido realizar importantes descubrimientos en el reconocimiento y procesamiento de imágenes, generando aplicaciones concretas en el mundo real [9].

Desde la introducción de las técnicas de aprendizaje profundo a comienzos de la década de 2010, y especialmente con el surgimiento de las CNN, tanto la investigación como la aplicación práctica de la VPC han experimentado transformaciones significativas. El hito representado por *AlexNet* en la competición *ImageNet* de 2012 [10], así como el desarrollo de arquitecturas modernas como *You Only Look Once* (YOLO) [11], ejemplifican este progreso.

Las CNN constituyen una clase de modelos de aprendizaje profundo diseñados para el procesamiento y análisis de datos visuales. Estas redes se caracterizan por una arquitectura compuesta por capas convolucionales, capas de *pooling* y capas densas (*fully connected*), lo que les confiere la capacidad de detectar y aprender características jerárquicas presentes en las imágenes de entrada. Una representación típica de la arquitectura de estas redes se ilustra en la figura 1.4.

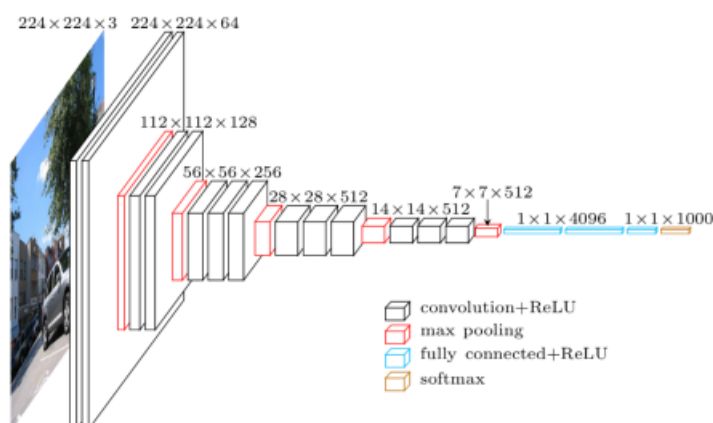


FIGURA 1.4. Red convolutiva VGG<sup>3</sup>.

Las tendencias recientes en el ámbito de las CNN evidencian avances significativos en el diseño de sus arquitecturas y en el procesamiento de datos, orientados a la solución de problemáticas emergentes. En particular, se ha puesto especial énfasis en el tratamiento de imágenes de alta resolución capturadas mediante vehículos aéreos no tripulados, tales como drones, lo que representa un área de aplicación de creciente interés académico y práctico [12] [13].

### 1.2.2. Detección del picudo rojo en imágenes aéreas

La relevancia de la detección de la plaga del picudo rojo a nivel global ha propiciado el surgimiento de diversas líneas de investigación. En este sentido, se han

<sup>3</sup>Imagen obtenida de: <https://paperswithcode.com/method/vgg>

desarrollado enfoques basados en VPC a partir de imágenes aéreas, haciendo especial uso de arquitecturas de CNN descritas en la sección 1.2.1.

Entre los estudios analizados, la implementación de la detección mediante imágenes aéreas de forma exclusiva representa una metodología relativamente novedosa. En este contexto, en el estudio *Automatic large scale detection of red palm weevil infestation using street view images* [14], se emplea un enfoque combinado. Inicialmente, se localizan las palmeras a través de una CNN (Faster R-CNN [15]) aplicada a imágenes aéreas. Posteriormente, utilizando la misma arquitectura, se extrae la corona de la palmera a partir de imágenes capturadas en *street view*, lo que permite clasificar a la planta en función de la presencia o ausencia de infección. Un resumen de este trabajo se presenta en la figura 1.5.

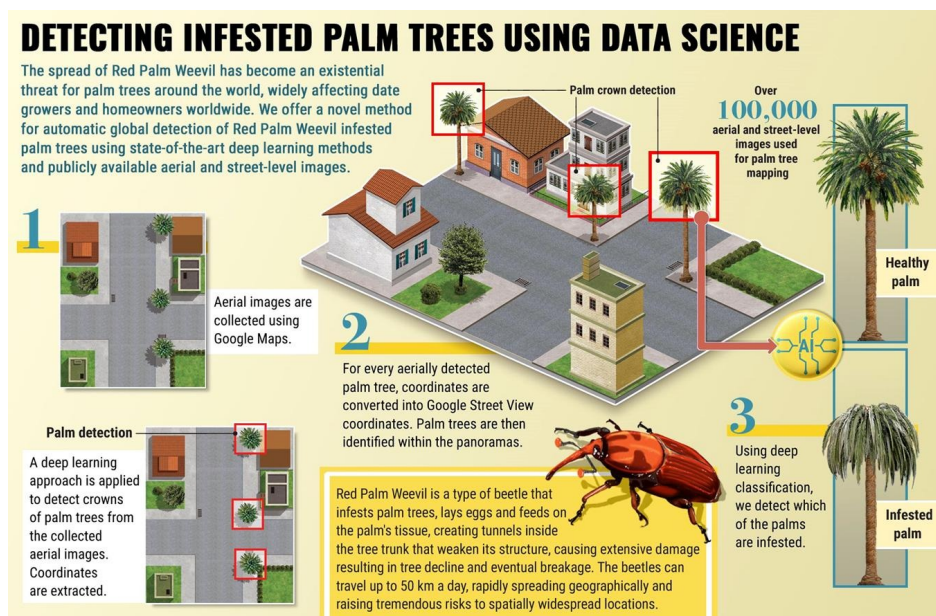


FIGURA 1.5. Proceso de detección realizado por Kagan et al <sup>4</sup>.

En cambio, la detección de palmeras si es ampliamente estudiada en la literatura, siendo un tema recurrente en la investigación de VPC. En este sentido, se han desarrollado diversos estudios. En *Implementation of slicing aided hyper inference (SAHI) in YOLOv8 to counting oil palm trees using high-resolution aerial imagery data* [16] se presenta un enfoque que utiliza la arquitectura YOLOv8 para detectar palmeras en imágenes aéreas de alta resolución. Además, utiliza *Slicing Aided Hyper Inference* para mejorar la precisión de la detección. Este enfoque se basa en la idea de dividir las imágenes en regiones más pequeñas y realizar inferencias en cada una de ellas, lo que permite una detección más precisa de los objetos de interés, algo esencial en el caso de objetos pequeños como las palmeras. Para el estudio, se utilizaron imágenes RGB capturadas con un dron Trinity F90+ a 200 metros de altura y una implementación de segmentación (SAHI) de 3000 × 3000 px. Si bien el estudio se centra en la detección de palmeras de aceite, su metodología puede ser adaptada para la detección del picudo rojo. La tabla 1.1 que resume los métodos utilizados en la detección de palmeras en diferentes estudios anteriores.

<sup>4</sup>Imagen obtenida de: <http://arxiv.org/abs/1506.01497>

TABLA 1.1. Comparación entre diferentes estudios de detección de palmeras<sup>5</sup>.

Autor y año	Modelo	Métricas
Mukhles Sir Monea et al, 2022 [17]	YOLOv3	5.76 % (MAPE)
Hery Wibowo et al, 2022 [18]	YOLOv3, YOLOv4, YOLOv5m	97.28 % (v3), 97.74 % (v4), 94.94 % (v5m). (F1-Score)
Adel Ammar et al, 2022 [19]	Faster R-CNN	94.99 % (Precision), 84 % (Recall), 83 % (AP IoU)
Wardana et al, 2023 [20]	YOLOv8	98.50 % (Overall Accuracy)
Deta Sandya Prasitha et al, 2022 [21]	YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x	0.82 (v5s), 0.84 (v5m), 0.85 (v5l), 0.86 (v5x). (Average F1-Score)
Nuwar et al, 2022 [22]	YOLOv5	0.895 (F1-Score)
Junos et al, 2022 [23]	YOLOv3n	97.20 % (mAP), 0.91 (F1-Score)

También, se han realizado estudios controlados como en *Red Palm Weevil Detection in Date Palm Using Temporal UAV Imagery* [24], en donde se concluye que la detección del picudo rojo puede ser posible mediante imágenes aéreas en el rango infrarrojo.

### 1.3. Objetivos y alcance

La finalidad principal del presente trabajo consistió en validar una prueba de concepto para la detección del picudo rojo en palmeras de Montevideo, utilizando imágenes aéreas capturadas mediante drones. Con este objetivo, se desarrolló un sistema fundamentado en técnicas de VPC y aprendizaje profundo, que permite identificar la presencia de la plaga. Para ello, se utilizó la información centralizada en el SIG [25], complementado con herramientas adicionales que facilitaron la generación del conjunto de datos de entrenamiento.

Adicionalmente, se planteó ante la IM la necesidad de optimizar los procesos de detección de la plaga, con miras a una mejora en la asignación de recursos humanos y económicos. El alcance inicial se centró exclusivamente en la detección del picudo rojo. En los capítulos siguientes se detalla tanto la descripción de las herramientas utilizadas como el proceso metodológico implementado.

<sup>5</sup>Tabla obtenida de *Implementation of slicing aided hyper inference (SAHI) in YOLOv8 to counting oil palm trees using high-resolution aerial imagery data* [16]

## Capítulo 2

# Introducción específica

En este capítulo se presentan en detalle las tecnologías y conceptos utilizados para el trabajo. Se parte de una introducción a las imágenes generadas por vehículos aéreos no tripulados (UAV), específicamente imágenes generadas por drones. Luego, se mencionan herramientas adicionales utilizadas en los procesos de soporte de proyectos orientados a IA, como lo son el proceso de etiquetado y la gestión de los datos. Finalmente, se puntualiza en modelos y métricas utilizados en VPC para este tipo de trabajos.

### 2.1. Imágenes generadas por vehículos aéreos

Los UAV han emergido como herramientas fundamentales en la obtención de imágenes aéreas de alta resolución. Su empleo ha permitido avances significativos en diversos campos, entre los que se destacan la agricultura de precisión, la vigilancia ambiental y la inspección de infraestructuras, entre otros [26].

La utilización de UAV en la adquisición de imágenes se fundamenta en varias ventajas inherentes a estos sistemas. En primer lugar, la flexibilidad operativa que ofrecen permite la planificación de vuelos adaptados a necesidades específicas de cada proyecto, garantizando la cobertura de zonas amplias o de difícil acceso (por ejemplo, palmeras en ubicaciones privadas de Montevideo o zonas en donde el acceso a pie es muy costoso). Asimismo, la capacidad de llevar equipamiento especializado (como cámaras multispectrales, sensores térmicos y dispositivos LiDAR) posibilita la obtención de datos con distintos niveles de resolución y en diversas bandas del espectro electromagnético, lo cual es esencial para aplicaciones que requieren análisis detallados y precisos [27].

Otro aspecto relevante es la eficiencia en términos de tiempo y costos. Comparados con métodos tradicionales de monitoreo, los UAV reducen considerablemente los recursos necesarios para este tipo de tareas, al tiempo que minimizan la exposición de personal a ambientes potencialmente peligrosos. Esto, sumado a la posibilidad de repetir vuelos en condiciones controladas, favorece la generación de series temporales de datos que permiten, en conjunción con el monitoreo, el análisis evolutivo de fenómenos específicos (como lo son las plagas).

Los UAV (como los drones) suelen cubrir un área específica; sin embargo, los vuelos de aviones tripulados cubren un área mucho mayor [28]. El servicio de geomática de la IM utiliza estas dos tecnologías para generar imágenes de todo Montevideo. Cada dos o tres años se realiza un vuelo de avión que cubre toda la ciudad. En contraste, los vuelos de drones pueden ser solicitados a demanda, dado que su costo es mucho menor.



La diferencia entre estas tecnologías no solo radica en el costo y área cubierta, sino también en el tipo de cámara que llevan estos vehículos. La resolución espacial de las imágenes generadas dependen de la cámara y la altura a la que vuelan estos vehículos (estas imágenes son procesadas por varios software, uno de ellos es Agisoft [29], que permite generar los ortomosaicos).

### 2.1.1. Imágenes de drones

Las imágenes generadas por drones se destacan por su alta resolución y capacidad para capturar detalles finos del terreno. Estas imágenes suelen tener una mayor resolución (medida en cm/px) en comparación con las generadas por aviones. El DJI Phantom 4 PRO [30] es ampliamente utilizado en aplicaciones de mapeo y fotogrametría debido a su capacidad para generar ortomosaicos con una resolución de hasta 5 cm/px (volando a 100 metros de altura). Otro dron ampliamente utilizado es el DJI MAVIC 3 Enterprise RTK [31]. La integración de tecnología RTK [32] en este dron permite una mayor precisión en la georreferenciación de las imágenes capturadas. A diferencia del DJI Phantom 4 PRO, que necesita puntos de apoyo relevados en el campo tomados con un GPS RTK (en las ejecuciones de vuelos con el DJI Phantom de la IM, los puntos de apoyo fueron tomados con GNSS SinoGNSS T300 Plus [33], utilizando corrección en tiempo real por NTRIP [34] con la base UYMO del Instituto Geográfico Militar uruguayo).

## 2.2. Herramientas de soporte en utilizadas en IA

La cara visible de los proyectos de inteligencia artificial se fundamenta, en general, en los resultados y aplicaciones derivados de modelos de aprendizaje automático. No obstante, al igual que en cualquier proyecto de software, resulta fundamental integrar buenas prácticas de ingeniería de software, lo que implica focalizarse en atributos específicos del sistema, tales como la escalabilidad, la seguridad y la recuperabilidad, entre otros [35]. Para alcanzar estos objetivos, es crucial considerar disciplinas que faciliten su consecución, como la gestión de la configuración o la gestión de datos. A continuación, se exponen algunas herramientas de apoyo a varios de estos procesos.

### 2.2.1. Gestión de código

El control de versiones es fundamental para asegurar la trazabilidad en los proyectos. Git permite mantener un historial detallado de los cambios, facilitando la integración de nuevas funcionalidades y la resolución de conflictos mediante estrategias de *branching* y *merging*. Esta práctica no solo mejora la trazabilidad, sino que brinda transparencia y contribuye a la reproducibilidad del proyecto a lo largo del tiempo. Una de las implementaciones más conocidas es GitHub [36], la cual se ha consolidado como una plataforma de colaboración que posibilita la gestión distribuida de proyectos de software en entornos tanto académicos como empresariales.

### 2.2.2. Gestión de infraestructura

La correcta administración de la infraestructura es esencial para el despliegue y mantenimiento de aplicaciones de inteligencia artificial, tanto en entornos de desarrollo como en producción. Para ello, en la industria se suelen utilizar diversas

herramientas, entre las cuales se encuentran:

- Docker y Docker Compose [37] [38]: estas herramientas permiten la creación de contenedores que encapsulan una aplicación (o varias aplicaciones) y sus dependencias, asegurando un entorno de ejecución uniforme y reproducible en diferentes sistemas. Docker Compose facilita la orquestación de múltiples contenedores, posibilitando la simulación de entornos complejos en local.
- Vagrant [39] [40]: se utiliza para la configuración y despliegue de máquinas virtuales. Esta herramienta ofrece un entorno virtualizado consistente y de fácil replicación, lo que resulta útil para la estandarización del entorno de desarrollo simplemente con un archivo de configuración (*Vagrantfile*).
- OpenShift [41] [42] y Helm [43] [44]: en entornos de producción, OpenShift basado en Kubernetes [45], permite una gestión centralizada y escalable de las aplicaciones, mientras que Helm simplifica la instalación, configuración y actualización de las mismas mediante el uso de paquetes predefinidos.

### 2.2.3. Gestión de usuarios y permisos

La seguridad y la privacidad son aspectos críticos en el desarrollo de aplicaciones de inteligencia artificial. La correcta gestión de usuarios y permisos es esencial para garantizar que solo las personas autorizadas tengan acceso a datos sensibles y funcionalidades específicas del sistema. Para ello, se pueden implementar herramientas compatibles con el protocolo *Lightweight Directory Access Protocol* (LDAP) [46], como lo es *Light LDAP* (LLDAP) [47]. Esta herramienta permite la autenticación y autorización de usuarios, facilitando la gestión centralizada de credenciales y permisos. Además, se pueden establecer políticas de acceso basadas en roles, lo que contribuye a una mayor seguridad y control sobre el sistema.

### 2.2.4. Gestión de etiquetado de datos

La calidad de los modelos de VPC dependen en gran medida de la precisión y consistencia del etiquetado de datos. Para optimizar este proceso, existen herramientas especializadas como CVAT [48], que facilitan la anotación de conjuntos de datos para el entrenamiento y validación de modelos. Su interfaz intuitiva y funcionalidades colaborativas permiten una asignación precisa de etiquetas en imágenes, reduciendo errores y mejorando la calidad de los datos. Entre las funcionalidades de CVAT se encuentra la integración con protocolos LDAP y la gestión de datos utilizando *Object Storage*.

FiftyOne [49], otra herramienta ampliamente utilizada en la industria, permite visualizar, analizar y gestionar los conjuntos de datos etiquetados. Facilita la identificación de inconsistencias y errores en las anotaciones, lo que resulta crucial para la mejora continua de los modelos de aprendizaje.

### 2.2.5. Gestión de repositorio de datos

El manejo eficiente de grandes volúmenes de datos es un aspecto crucial en los proyectos de IA, especialmente las imágenes en proyectos de VPC. Entre las opciones *Open Source* se encuentran:

- MinIO [50]: solución de almacenamiento de objetos compatible con S3 [51] (*S3-compliant*), permite gestionar grandes volúmenes de datos de manera escalable y segura. Su implementación posibilita la integración tanto en entornos de almacenamiento en la nube como en sistemas locales, asegurando la disponibilidad continua de la información. Se integra fácilmente con herramientas de anotaciones como CVAT.
- MongoDB [52]: base de datos NoSQL que se utiliza para almacenar metadatos y gestionar información asociada a los conjuntos de datos. La flexibilidad y escalabilidad de MongoDB facilitan la administración de datos estructurados y no estructurados, permitiendo consultas eficientes y un análisis detallado posterior.

La integración de estas herramientas de soporte permiten establecer un flujo de trabajo sistematizado y robusto. Este enfoque integral asegura la reproducibilidad de los resultados, optimiza la gestión de recursos y facilita tanto el mantenimiento como la evolución del sistema a lo largo del tiempo.

## 2.3. Modelos de visión por computadora

El desarrollo de los modelos de VPC han permitido avances notables en la automatización y análisis de imágenes. Entre las tareas de clasificación y detección de objetos, las CNN se destacan con altos niveles de precisión y eficiencia.

### 2.3.1. Clasificación de objetos

La tarea de clasificación consiste en asignar una etiqueta o categoría a una imagen o a regiones específicas de la misma. Entre los modelos más utilizados en esta área se encuentra la familia *EfficientNet*. Esta familia fue introducida en 2019 por investigadores de Google en el artículo *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks* [53]. Este conjunto de arquitecturas se caracteriza por su innovador enfoque de escalado compuesto, que equilibra de forma simultánea la profundidad, el ancho y la resolución de la red lo que le permite lograr mejores resultados que los enfoques tradicionales.

### 2.3.2. Detección de objetos

La detección de objetos se enfoca no solo en clasificar, sino también en localizar de forma precisa los elementos de interés dentro de una imagen. Esta tarea se ha abordado mediante dos grandes paradigmas: los modelos de dos etapas y los modelos de una etapa.

#### Modelos de dos etapas

Los modelos de dos etapas, representados de manera destacada por *Faster R-CNN*, se caracterizan por un proceso secuencial en el que se generan propuestas de regiones de interés (ROI) para luego clasificarlas y refinar sus límites. La primera etapa utiliza una red de propuestas de regiones (RPN) que sugiere ubicaciones potenciales de objetos. En la segunda etapa, estas propuestas se analizan en detalle mediante un clasificador que asigna una etiqueta y ajusta la caja delimitadora. Esta aproximación, aunque computacionalmente más costosa, ofrece una



alta precisión en la detección, lo que la hace adecuada para aplicaciones donde la exactitud es prioritaria.

### Modelos de una etapa

En contraste, los modelos de una etapa, como YOLO, abordan el problema de la detección en un único paso. Estos modelos dividen la imagen en una cuadrícula y, de forma simultánea, predicen la probabilidad de presencia de un objeto y la localización de su correspondiente caja delimitadora. Esta integración de procesos permite una mayor velocidad de inferencia, haciendo de YOLO una opción idónea para aplicaciones en tiempo real, aunque hay antecedentes del uso de este tipo de modelos para la detección de objetos con una alta eficiencia en la sección 1.2.

## 2.4. Métricas en visión por computadora

La evaluación cuantitativa de los modelos de VPC es esencial para determinar su desempeño y compararlos de manera objetiva. En este contexto, se utilizan diversas métricas que permiten analizar tanto la precisión en la clasificación de imágenes como la efectividad en la detección y localización de objetos. A continuación, se describen las métricas más utilizadas en el campo.

### 2.4.1. Precisión, Recuperación y *F1-Score*

- Precisión (*Precision*): mide la proporción de verdaderos positivos sobre el total de predicciones positivas realizadas por el modelo. Una alta precisión indica que, de todos los objetos identificados, la mayoría corresponde efectivamente a objetos de interés. Se define como:

$$\text{Precisión} = \frac{TP}{TP + FP} \quad (2.1)$$

- Recuperación (*Recall*): representa la proporción de verdaderos positivos detectados sobre el total de objetos de interés presentes en el conjunto de datos. Un valor elevado de *recall* sugiere que el modelo es capaz de capturar la mayoría de los objetos existentes. Se define como:

$$\text{Recuperación} = \frac{TP}{TP + FN} \quad (2.2)$$

- Exactitud (*Accuracy*): mide la proporción de predicciones correctas (tanto verdaderos positivos como verdaderos negativos) sobre el total de predicciones realizadas. Esta métrica es útil en contextos donde las clases están equilibradas, pero puede ser engañosa en conjuntos de datos desbalanceados. Se define como:

$$\text{Exactitud} = \frac{TP + TN}{\text{Total de Predicciones}} \quad (2.3)$$

- *F1-Score*: es la media armónica entre *precision* y *recall*, y proporciona una medida equilibrada cuando se requiere considerar ambas métricas simultáneamente. Se define como:

$$F1 = 2 \cdot \frac{\text{Precisión} \cdot \text{Recuperación}}{\text{Precisión} + \text{Recuperación}} \quad (2.4)$$

Un resumen intuitivo de estas métricas se puede visualizar en la imagen 2.1.

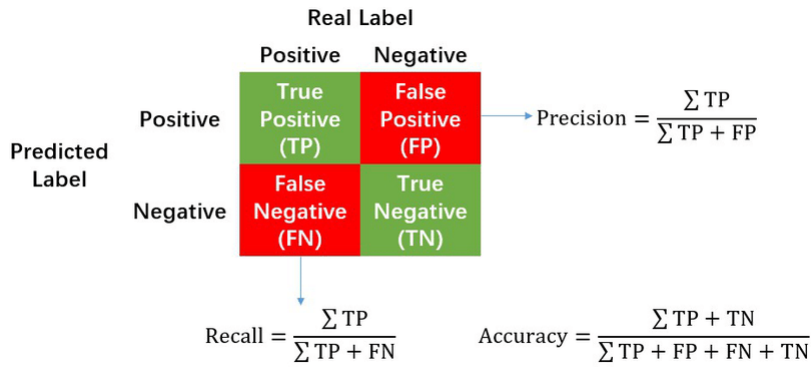


FIGURA 2.1. Matriz de confusion <sup>1</sup>.

#### 2.4.2. Intersección sobre Unión (IOU)

La métrica de IOU evalúa la superposición entre la caja delimitadora predicha y la caja delimitadora real de un objeto. Se calcula como el cociente entre el área de intersección y el área de unión de ambas cajas. Esta métrica es fundamental en tareas de detección, ya que un IOU alto indica una buena coincidencia espacial entre la predicción y la referencia, como se puede intuir en la siguiente formula:

$$IoU = \frac{\text{Área de Intersección}}{\text{Área de Unión}} \quad (2.5)$$

#### 2.4.3. Average Precision (AP) y Mean Average Precision (mAP)

- AP: para cada clase de objeto, se calcula el promedio de la precisión, integrando la curva de precisión-recuperación (ROC). La AP resume el rendimiento del detector en diferentes umbrales de decisión.
- mAP: es la media aritmética de la AP calculada para cada clase. El mAP es ampliamente utilizado en competencias y evaluaciones de modelos de detección de objetos, ya que proporciona una única medida que resume la efectividad global del modelo.

#### 2.4.4. Otras métricas relevantes

Además de las métricas anteriormente descritas, en la práctica se pueden utilizar otros indicadores, dependiendo de la aplicación y la naturaleza de los datos:

<sup>1</sup>Imagen tomada de [https://www.researchgate.net/figure/Calculation-of-Precision-Recall-and-Accuracy-in-the-confusion-matrix\\_fig3\\_336402347](https://www.researchgate.net/figure/Calculation-of-Precision-Recall-and-Accuracy-in-the-confusion-matrix_fig3_336402347)

- Curva ROC y área bajo la curva (AUC): la curva ROC y el AUC se emplean comúnmente en tareas de clasificación para evaluar la capacidad del modelo en discriminar entre clases.
- Error Cuadrático Medio (MSE) y Error Absoluto Medio (MAE): en aplicaciones que requieren regresión, como la estimación de posiciones o dimensiones, se utilizan estas métricas para cuantificar la diferencia entre los valores predichos y los reales.

La correcta selección y combinación de estas métricas permite una evaluación integral de los modelos de VPC, facilitando tanto la identificación de áreas de mejora como la comparación objetiva entre diferentes enfoques o arquitecturas implementadas.



## Capítulo 3

# Diseño e implementación

Todos los capítulos deben comenzar con un breve párrafo introductorio que indique cuál es el contenido que se encontrará al leerlo. La redacción sobre el contenido de la memoria debe hacerse en presente y todo lo referido al proyecto en pasado, siempre de modo impersonal.

### 3.1. Análisis del software

La idea de esta sección es resaltar los problemas encontrados, los criterios utilizados y la justificación de las decisiones que se hayan tomado.

Se puede agregar código o pseudocódigo dentro de un entorno `lstlisting` con el siguiente código:

```
\begin{lstlisting}[caption= "un epígrafe descriptivo"]
  las líneas de código irían aquí...
\end{lstlisting}
```

A modo de ejemplo:

```
1 #define MAX_SENSOR_NUMBER 3
2 #define MAX_ALARM_NUMBER 6
3 #define MAX_ACTUATOR_NUMBER 6
4
5 uint32_t sensorValue[MAX_SENSOR_NUMBER];
6 FunctionalState alarmControl[MAX_ALARM_NUMBER]; //ENABLE or DISABLE
7 state_t alarmState[MAX_ALARM_NUMBER]; //ON or OFF
8 state_t actuatorState[MAX_ACTUATOR_NUMBER]; //ON or OFF
9
10 void vControl() {
11
12     initGlobalVariables();
13
14     period = 500 ms;
15
16     while(1) {
17
18         ticks = xTaskGetTickCount();
19
20         updateSensors();
21
22         updateAlarms();
23
24         controlActuators();
25
26         vTaskDelayUntil(&ticks, period);
27     }
```

28 }

CÓDIGO 3.1. Pseudocódigo del lazo principal de control.

## Capítulo 4

# Ensayos y resultados

Todos los capítulos deben comenzar con un breve párrafo introductorio que indique cuál es el contenido que se encontrará al leerlo. La redacción sobre el contenido de la memoria debe hacerse en presente y todo lo referido al proyecto en pasado, siempre de modo impersonal.

### 4.1. Pruebas funcionales del hardware

La idea de esta sección es explicar cómo se hicieron los ensayos, qué resultados se obtuvieron y analizarlos.





## Capítulo 5

# Conclusiones

Todos los capítulos deben comenzar con un breve párrafo introductorio que indique cuál es el contenido que se encontrará al leerlo. La redacción sobre el contenido de la memoria debe hacerse en presente y todo lo referido al proyecto en pasado, siempre de modo impersonal.

### 5.1. Conclusiones generales

La idea de esta sección es resaltar cuáles son los principales aportes del trabajo realizado y cómo se podría continuar. Debe ser especialmente breve y concisa. Es buena idea usar un listado para enumerar los logros obtenidos.

En esta sección no se deben incluir ni tablas ni gráficos.

Algunas preguntas que pueden servir para completar este capítulo:

- ¿Cuál es el grado de cumplimiento de los requerimientos?
- ¿Cuán fielmente se pudo seguir la planificación original (cronograma incluido)?
- ¿Se manifestó algunos de los riesgos identificados en la planificación? ¿Fue efectivo el plan de mitigación? ¿Se debió aplicar alguna otra acción no contemplada previamente?
- Si se debieron hacer modificaciones a lo planificado ¿Cuáles fueron las causas y los efectos?
- ¿Qué técnicas resultaron útiles para el desarrollo del proyecto y cuáles no tanto?

### 5.2. Próximos pasos

Acá se indica cómo se podría continuar el trabajo más adelante.



# Bibliografía

- [1] Anticimex. *Picudo Rojo | Daños y tratamientos - Anticimex*. URL: <https://www.anticimex.es/picudo-rojo/> (visitado 21-03-2025).
- [2] A. Poplin et al. *Palm Weevils*. Mayo de 2014.
- [3] MGAP. «Información actualizada sobre el picudo rojo de las palmeras a setiembre 2024». En: (). URL: <https://www.gub.uy/ministerio-ganaderia-agricultura-pesca/comunicacion/noticias/informacion-actualizada-sobre-picudo-rojo-palmeras-setiembre-2024> (visitado 21-03-2025).
- [4] Alfonso Arcos. *PICUDO ROJO (Rynchophorus ferrugineus) EN MONTEVIDEO MONITOREO Y CONTROL (Avances)*. Montevideo, ago. de 2024. (Visitado 21-03-2025).
- [5] Intendencia de Montevideo. *Acciones de la Intendencia de Montevideo contra el Picudo Rojo | Portal institucional*. URL: <https://montevideo.gub.uy/noticias/urbanismo-y-obras/acciones-de-la-intendencia-de-montevideo-contr-a-el-picudo-rojo> (visitado 26-03-2025).
- [6] Pedro Hernández Sánchez. «Cirugía especializada en palmeras». es. En: ().
- [7] BMVA. *What is computer vision?* 2017. URL: <https://web.archive.org/web/20170216180225/http://www.bmva.org/visionoverview>.
- [8] A. Torralba, P. Isola y W.T. Freeman. *Foundations of Computer Vision*. Adaptive Computation and Machine Learning series. MIT Press, 2024. ISBN: 978-0-262-37866-6. URL: <https://mitpress.mit.edu/9780262048972/foundations-of-computer-vision/>.
- [9] Xingshuai Dong y Massimiliano L. Cappuccio. *Applications of Computer Vision in Autonomous Vehicles: Methods, Challenges and Future Directions*. arXiv:2311.09093 [cs]. Jun. de 2024. DOI: [10.48550/arXiv.2311.09093](https://doi.org/10.48550/arXiv.2311.09093). URL: <http://arxiv.org/abs/2311.09093> (visitado 21-03-2025).
- [10] Alex Krizhevsky, Ilya Sutskever y Geoffrey E. Hinton. «ImageNet classification with deep convolutional neural networks». en. En: *Communications of the ACM* 60.6 (mayo de 2017), págs. 84-90. ISSN: 0001-0782, 1557-7317. DOI: [10.1145/3065386](https://doi.org/10.1145/3065386). URL: <https://dl.acm.org/doi/10.1145/3065386> (visitado 21-03-2025).
- [11] Joseph Redmon et al. *You Only Look Once: Unified, Real-Time Object Detection*. arXiv:1506.02640 [cs]. Mayo de 2016. DOI: [10.48550/arXiv.1506.02640](https://doi.org/10.48550/arXiv.1506.02640). URL: <http://arxiv.org/abs/1506.02640> (visitado 21-03-2025).
- [12] Mingliang Gao et al. «Recent Advances in Computer Vision: Technologies and Applications». en. En: *Electronics* 13.14 (ene. de 2024). Number: 14 Publisher: Multidisciplinary Digital Publishing Institute, pág. 2734. ISSN: 2079-9292. DOI: [10.3390/electronics13142734](https://doi.org/10.3390/electronics13142734). URL: <https://www.mdpi.com/2079-9292/13/14/2734> (visitado 21-03-2025).
- [13] Manav Sutar. «Convolutional Neural Networks (CNNs): Advancements and Future Trends». En: *Convolutional Neural Networks (CNNs): Advancements and Future Trends* (ene. de 2025). URL: [https://www.academia.edu/128256115/Convolutional\\_Neural\\_Networks\\_CNNs\\_Advancements\\_and\\_Future\\_Trends](https://www.academia.edu/128256115/Convolutional_Neural_Networks_CNNs_Advancements_and_Future_Trends) (visitado 21-03-2025).

- [14] Dima Kagan, Galit Fuhrmann Alpert y Michael Fire. «Automatic large scale detection of red palm weevil infestation using street view images». en. En: *ISPRS Journal of Photogrammetry and Remote Sensing* 182 (dic. de 2021), págs. 122-133. ISSN: 09242716. DOI: [10.1016/j.isprsjprs.2021.10.004](https://doi.org/10.1016/j.isprsjprs.2021.10.004). URL: <https://linkinghub.elsevier.com/retrieve/pii/S0924271621002665> (visitado 21-11-2024).
- [15] Shaoqing Ren et al. *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. arXiv:1506.01497 [cs]. Ene. de 2016. DOI: [10.48550/arXiv.1506.01497](https://doi.org/10.48550/arXiv.1506.01497). URL: <http://arxiv.org/abs/1506.01497> (visitado 21-03-2025).
- [16] Naufal Najiv Zhorif et al. «Implementation of Slicing Aided Hyper Inference (SAHI) in YOLOv8 to Counting Oil Palm Trees Using High-Resolution Aerial Imagery Data». en. En: *International Journal of Advanced Computer Science and Applications* 15.7 (2024). ISSN: 21565570, 2158107X. DOI: [10.14569/IJACSA.2024.0150786](https://doi.org/10.14569/IJACSA.2024.0150786). URL: <http://thesai.org/Publications/ViewPaper?Volume=15&Issue=7&Code=ijacsa&SerialNo=86> (visitado 22-03-2025).
- [17] M. S. Muna et al. «Development of Automatic Counting System for Palm Oil Tree Based on Remote Sensing Imagery». En: *Proceedings of the International Conference on Sustainable Environment, Agriculture and Tourism (ICOSEAT 2022)*. Atlantis Press, ene. de 2023. DOI: [10.2991/978-94-6463-086-2\\_68](https://doi.org/10.2991/978-94-6463-086-2_68).
- [18] H. Wibowo et al. «Large-Scale Oil Palm Trees Detection from High-Resolution Remote Sensing Images Using Deep Learning». En: *Big Data and Cognitive Computing* 6.3 (sep. de 2022). DOI: [10.3390/bdcc6030089](https://doi.org/10.3390/bdcc6030089).
- [19] Adel Ammar, Anis Koubaa y Bilel Benjdira. «Deep-Learning-Based Automated Palm Tree Counting and Geolocation in Large Farms from Aerial Geotagged Images». en. En: *Agronomy* 11.8 (jul. de 2021), pág. 1458. ISSN: 2073-4395. DOI: [10.3390/agronomy11081458](https://doi.org/10.3390/agronomy11081458). URL: <https://www.mdpi.com/2073-4395/11/8/1458> (visitado 22-03-2025).
- [20] D. P. T. Wardana, R. S. Sianturi y R. Fatwa. «Detection of Oil Palm Trees Using Deep Learning Method with High-Resolution Aerial Image Data». En: *ACM International Conference Proceeding Series*. Association for Computing Machinery, 2023, págs. 90-98. DOI: [10.1145/3626641.3626667](https://doi.org/10.1145/3626641.3626667).
- [21] D. Sandya Prasvita, D. Chahyati y A. M. Arymurthy. *Automatic Detection of Oil Palm Growth Rate Status with YOLOv5*.
- [22] Y. Nuwara, W. K. Wong y F. H. Juwono. «Modern Computer Vision for Oil Palm Tree Health Surveillance using YOLOv5». En: *2022 International Conference on Green Energy, Computing and Sustainable Technology, GECOST 2022*. Institute of Electrical y Electronics Engineers Inc., 2022, págs. 404-409. DOI: [10.1109/GECOST55694.2022.10010668](https://doi.org/10.1109/GECOST55694.2022.10010668).
- [23] M. H. Junos et al.
- [24] Stephanie Delalieux et al. «Red Palm Weevil Detection in Date Palm Using Temporal UAV Imagery». en. En: *Remote Sensing* 15.5 (ene. de 2023). Number: 5 Publisher: Multidisciplinary Digital Publishing Institute, pág. 1380. ISSN: 2072-4292. DOI: [10.3390/rs15051380](https://doi.org/10.3390/rs15051380). URL: <https://www.mdpi.com/2072-4292/15/5/1380> (visitado 21-03-2025).
- [25] Intendencia de Montevideo. *Sistema de Información Geográfica*. es. URL: <http://sig.montevideo.gub.uy/> (visitado 26-03-2025).
- [26] Izham Mokhtar et al. *Image Processing Systems for Unmanned Aerial Vehicle: State-of-the-art*. Ago. de 2023. DOI: [10.20944/preprints202308.1855.v1](https://doi.org/10.20944/preprints202308.1855.v1).

- [27] Unmanned System technology. *UAV/Drone Cameras for Commercial, Industrial, and Military Applications*. en-US. URL: <https://www.unmannedsystemstechnology.com/expo/cameras/> (visitado 27-03-2025).
- [28] 2000 Aviation. *Ortoimágenes – 2000 Aviation*. es. URL: <https://www.2000aviation.com/ortoimagenes/> (visitado 27-03-2025).
- [29] Agisoft. *Agisoft Metashape: Agisoft Metashape*. URL: <https://www.agisoft.com/> (visitado 27-03-2025).
- [30] DJI. *Support for Phantom 4 Pro*. en. URL: <https://www.dji.com/global/support/product/photo> (visitado 27-03-2025).
- [31] DJI. *Specs - DJI Mavic 3 Enterprise - DJI Enterprise*. en. URL: <https://enterprise.dji.com/mavic-3-enterprise/photo> (visitado 27-03-2025).
- [32] Wikipedia. *RTK (navegación)*. es. Page Version ID: 161740605. Ago. de 2024. URL: [https://es.wikipedia.org/w/index.php?title=RTK\\_\(navegaci%C3%B3n\)&oldid=161740605](https://es.wikipedia.org/w/index.php?title=RTK_(navegaci%C3%B3n)&oldid=161740605) (visitado 27-03-2025).
- [33] ComNav Technology Ltd. *Receptor GNSS T300 Plus-Receptor GNSS-ComNav Technology Ltd*. zh-CN. URL: <https://www.comnavtech.com/sp/product/receiver/T300Plus.html> (visitado 28-03-2025).
- [34] Wikipedia. *Networked Transport of RTCM via Internet Protocol*. en. Page Version ID: 1277712218. Feb. de 2025. URL: [https://en.wikipedia.org/w/index.php?title=Networked\\_Transport\\_of\\_RTCM\\_via\\_Internet\\_Protocol&oldid=1277712218](https://en.wikipedia.org/w/index.php?title=Networked_Transport_of_RTCM_via_Internet_Protocol&oldid=1277712218) (visitado 27-03-2025).
- [35] Ian Sommerville. *Software Engineering*. Addison-Wesley, mar. de 2015.
- [36] Github. *Build software better, together*. en. URL: <https://github.com> (visitado 27-03-2025).
- [37] Dirk Merkel. «Docker: lightweight Linux containers for consistent development and deployment». En: *Linux journal* 2014.239 (mar. de 2014), págs. 2-. URL: <https://dl.acm.org/citation.cfm?id=2600239.2600241>.
- [38] Docker. *Docker documentation*. en. 0. URL: <https://docs.docker.com/> (visitado 27-03-2025).
- [39] HashiCorp. *Vagrant*. 2023.
- [40] HashiCorp. *Documentation | Vagrant | HashiCorp Developer*. en. URL: <https://developer.hashicorp.com/vagrant/docs> (visitado 27-03-2025).
- [41] Red Hat. *OpenShift*. 2023.
- [42] Red Hat. *Red Hat Product Documentation*. URL: <https://docs.redhat.com/en> (visitado 27-03-2025).
- [43] The Helm Authors. *Helm*. 2023.
- [44] The Helm Authors. *Docs Home*. en. URL: <https://helm.sh/docs/> (visitado 27-03-2025).
- [45] Wikipedia. *Kubernetes*. es. Page Version ID: 166046716. Mar. de 2025. URL: <https://es.wikipedia.org/w/index.php?title=Kubernetes&oldid=166046716> (visitado 27-03-2025).
- [46] *Protocolo ligero de acceso a directorios*. es. Page Version ID: 162588716. Sep. de 2024. URL: [https://es.wikipedia.org/w/index.php?title=Protocolo\\_ligero\\_de\\_acceso\\_a\\_directorios&oldid=162588716](https://es.wikipedia.org/w/index.php?title=Protocolo_ligero_de_acceso_a_directorios&oldid=162588716) (visitado 27-03-2025).
- [47] LLDAP Authors. *LLDAP: Light LDAP implementation*. 2023.
- [48] Boris Sekachev et al. *opencv/cvat: v1.1.0*. Ago. de 2020. DOI: 10.5281/zenodo.4009388. URL: <https://doi.org/10.5281/zenodo.4009388>.
- [49] B. E. Moore y J. J. Corso. «FiftyOne». En: *GitHub. Note: https://github.com/voxel51/fiftyone* (2020).
- [50] MinIO, Inc. *MinIO: High Performance Object Storage*. 2023.

- 
- [51] Amazon. *Amazon S3*. es. Page Version ID: 164024822. Dic. de 2024. URL: [https://es.wikipedia.org/w/index.php?title=Amazon\\_S3&oldid=164024822](https://es.wikipedia.org/w/index.php?title=Amazon_S3&oldid=164024822) (visitado 28-03-2025).
  - [52] MongoDB, Inc. *MongoDB: The Developer Data Platform*. 2023.
  - [53] Mingxing Tan y Quoc V. Le. *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*. arXiv:1905.11946 [cs]. Sep. de 2020. DOI: [10.48550/arXiv.1905.11946](https://doi.org/10.48550/arXiv.1905.11946). URL: <http://arxiv.org/abs/1905.11946> (visitado 27-03-2025).