

# UNIVERSIDAD TÉCNICA DEL NORTE



**Facultad de Ingeniería en Ciencias Aplicadas**

**Carrera de Ingeniería en Software**

## **DETECCIÓN AUTOMÁTICA DE MALEZAS EN PLANTACIONES DE MAÍZ Y/O PAPA UTILIZANDO IMÁGENES ADQUIRIDAS POR DRON Y RED NEURONAL CONVOLUCIONAL YOLO**

Trabajo de grado previo a la obtención del título de Ingeniero en Software ante la  
ilustre Universidad Técnica del Norte.

Autor:

Bryan Andres Ortega Llanos

Director:

PhD. Iván Danilo García Santillán

Ibarra – Ecuador

2024



# UNIVERSIDAD TÉCNICA DEL NORTE

## BIBLIOTECA UNIVERSITARIA

### AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

#### 1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

<b>Datos de contacto</b>			
<b>CEDULA DE IDENTIDAD:</b>	100397978-6		
<b>APELLIDOS Y NOMBRES:</b>	Ortega Llanos Bryan Andres		
<b>DIRECCIÓN:</b>	Av. 13 de Abril, Ibarra		
<b>EMAIL:</b>	<a href="mailto:baortegal@utn.edu.ec">baortegal@utn.edu.ec</a>		
<b>TELEFONO FIJO:</b>	62600035	<b>TELEFONO MOVIL:</b>	960283499

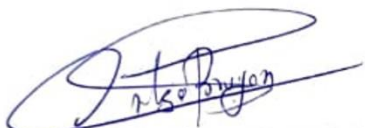
<b>DATOS DE LA OBRA</b>	
<b>TÍTULO:</b>	DETECCIÓN AUTOMÁTICA DE MALEZAS EN PLANTACIONES DE MAÍZ Y/O PAPA UTILIZANDO IMÁGENES ADQUIRIDAS POR DRON Y RED NEURONAL CONVOLUCIONAL YOLO
<b>AUTOR(ES):</b>	Ortega Llanos Bryan Andres
<b>FECHA:</b>	05/03/2024
<b>PROGRAMA:</b>	Pregrado
<b>TÍTULO POR EL QUE OPTA:</b>	Ingeniero en Software.
<b>DIRECTOR:</b>	PhD. García Santillán Iván Danilo.
<b>ASESOR 1:</b>	PhD. Pusedá Chulde Marco Remigio.

## 2. CONSTANCIA

El autor(es) manifiesta que la obra objeto de la presente autorización es original y se la desarrollo, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es(son) el(los) titular(es) de los derechos patrimoniales, que asume(n) la responsabilidad sobre el contenido de ésta, y saldrá(n) en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 5 días del mes de marzo de 2024.

EL AUTOR:

A handwritten signature in blue ink, appearing to read 'Ortega Llanos Bryan Andres', is written over a large, light blue oval scribble.

Nombre: Ortega Llanos Bryan Andres

C.I: 1003979786



UNIVERSIDAD TÉCNICA DEL NORTE  
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

Carrera de Software

CERTIFICACIÓN DIRECTOR DEL TRABAJO DE TITULACIÓN

En mi calidad de Tutor de Trabajo de Grado presentado por el egresado, **Bryan Andres Ortega Llanos** para optar por el título de Ingeniero de Software, cuyo tema es: "**DETECCIÓN AUTOMÁTICA DE MALEZAS EN PLANTACIONES DE MAÍZ Y/O PAPA UTILIZANDO IMÁGENES ADQUIRIDAS POR DRON Y RED NEURONAL CONVOLUCIONAL YOLO**". Considero que el presente trabajo reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del tribunal examinador.

En la ciudad de Ibarra, a los 5 días del mes de marzo de 2024.

A handwritten signature in blue ink, appearing to read 'Iván Danilo García Santillán', written over a horizontal line.

PhD. García Santillán Iván Danilo  
DIRECTOR DE TRABAJO DE GRADO

## **DEDICATORIA**

Con un gran cariño y esfuerzo este trabajo va dedicado a mis padres y hermanos, cuya influencia y apoyo han sido fundamentales en mi trayectoria hacia el éxito profesional.

Y un mérito especial a mí mismo por tener el coraje de cambiar de carrera. Aunque no fue una decisión tomada en el momento ideal, estoy feliz de haberlo hecho, ya que me condujo a mi verdadera pasión.

## **AGRADECIMIENTO**

Quiero expresar mi más profunda gratitud a mi familia: a mi padre y mi madre, cuyo amor y guía incondicional han sido mi sostén; a mis hermanos, Dylan y Jorge, por su constante apoyo y camaradería; a mi abuelita Merceditas, cuya sabiduría y ternura han enriquecido mi vida; y a mi novia Nathy, por su amor y aliento incesante. Una mención especial para mi sobrinita Zahid, cuya presencia ha traído una alegría inmensurable a nuestra familia.

Mi agradecimiento se extiende también a los profesores de la Universidad Técnica del Norte, cuya dedicación y enseñanzas han sido fundamentales en mi formación académica y profesional. Cada uno de ellos ha contribuido significativamente a mi crecimiento y desarrollo en esta etapa crucial de mi vida.

Un agradecimiento especial a los PhD. García Santillán Iván Danilo y Pusedá Chulde Marco Remigio por su orientación y apoyo excepcionales en mi trabajo de titulación. Su sabiduría y mentoría han sido clave en la culminación exitosa de este proyecto.

## TABLA DE CONTENIDOS

CAPÍTULO 1 .....	19
1.1. Fundamentación teórica.....	19
1.1.1. Malezas .....	19
1.1.2. Control de malezas .....	19
1.1.3. Malezas en maíz y papas .....	20
1.1.4. Imágenes por dron .....	22
1.1.5. Dron.....	22
1.1.6. Condiciones de vuelo.....	23
1.2. Redes neuronales convolucionales .....	23
1.2.1. Inteligencia Artificial (IA) .....	23
1.2.2. Visión Artificial .....	24
1.2.3. Redes neuronales convolucionales (CNN).....	25
1.2.4. Componentes, arquitectura y técnicas de entrenamiento.....	26
1.2.5. Aprendizaje supervisado y no supervisado, inicialización de pesos, regularización y optimizadores .....	27
1.2.6. Herramientas y bibliotecas para trabajar con redes neuronales convolucionales .....	27
1.2.7. Red neuronal convolucional YOLO .....	28
1.2.8. YOLOv8.....	30
1.2.9. Metodología de desarrollo.....	32
1.2.10. Metodología KDD .....	32
1.2.11. Metodología Kanban .....	33
1.3. Aplicación Web.....	34
1.3.1. Detalles del sistema .....	34
1.3.2. Lenguaje de programación .....	35

1.4.	Trabajos relacionados .....	37
1.5.	Preguntas Científicas .....	40
CAPÍTULO 2.....		42
Desarrollo del proyecto .....		42
2.1.	Metodología y herramientas utilizadas .....	42
2.1.1.	Metodología KDD y Kanban .....	42
2.1.2.	Herramientas de Desarrollo y Evaluación .....	44
2.2.	Elaboración del dataset.....	47
2.2.1.	Adquisición de datos .....	47
2.2.2.	Preprocesamiento de imágenes .....	51
2.2.3.	Etiquetado de imágenes.....	54
2.3.	Elaboración del dataset.....	57
2.4.	Implementación y Evaluación del Modelo .....	57
2.4.1.	Establecimiento del entorno de entrenamiento .....	58
2.4.2.	Configuración de hiperparámetros .....	60
2.4.3.	Entrenamiento del modelo .....	61
2.5.	Desarrollo de la aplicación Web .....	62
2.5.1.	Pruebas y cumplimiento de la norma ISO 25010.....	63
CAPÍTULO 3.....		64
Resultados y validaciones .....		64
3.1.	Métricas del modelo YOLOv8.....	64
3.1.1.	Línea de tiempo del modelo entrenado .....	64
3.2.	Comparación del modelo con métodos tradicionales .....	71
3.2.1.	Inferencias y resultados .....	71
3.2.2.	Métodos tradicionales .....	74
3.2.3.	Uso del modelo de detección de malezas con Dron .....	74
3.2.4.	Comparativa .....	74



<b>3.3.</b>	Pruebas estadísticas.....	76
3.3.1.	T – Student en tiempos de detección de malezas .....	76
3.3.2.	T – Student en detección de malezas con YOLO y drones .....	77
3.3.3.	T – Student en detección de papas con YOLO vs experto .....	79
3.3.4.	T – Student en detección de plantas con YOLO vs experto .....	82
<b>3.4.</b>	Nivel de Infestación.....	85
<b>3.5.</b>	Discusión.....	86
<b>3.6.</b>	Conclusiones.....	87
<b>3.7.</b>	Recomendaciones.....	88
<b>3.8.</b>	Bibliografía.....	89

## ÍNDICE DE FIGURAS

Etapas del maíz (Zimdahl, 2018). .....	20
Etapas de la papa.....	21
Condiciones de vuelo, adaptado de (Volpato et al., 2021).....	23
Ramas de la IA (Neota Logic,2015).....	24
Ejemplo de detección por computadora (Appen, 2019). .....	25
Estructura de una CNN (Manav Mandal, 2021).....	26
Arquitectura YOLOV8 (Schcolnik-Elias et al., 2023). .....	31
Comparación de rendimiento YOLO (Ultralytics, 2023). .....	32
Etapas del aplicativo web (Elaboración Propia). .....	35
Tareas programadas con Kanban.....	44
Dron DJI Mavic 2 Pro .....	45
Recursos ofrecidos en Colab Pro .....	46
Implementación del modelo con Gradio .....	46
Planta de papa con 30 días de sembrado .....	48
Plantación de papas.....	48
Planes de vuelo DronDeploy.....	49
Fotografía a 9 m de altura .....	50
Parámetros de vuelo .....	50
Plantas a etiquetar.....	51
Información imagen original .....	52
Obstáculos presentes en la imagen.....	53
Tomas fuera de la plantación .....	54
Toma desde altura muy alta.....	54
Toma desde altura muy baja.....	54
Uso de herramienta roboflow .....	55
Etiquetado por bounding box .....	56

Formato de etiquetado YOLO .....	56
Formato de exportación Roboflow .....	56
Estadísticas dataset .....	57
Estructura entorno de entrenamiento .....	59
Aplicación en Gradio .....	62
Alojamiento en Google Colab.....	62
Inferencia en imagen.....	63
Métricas en entrenamiento.....	65
Métricas en validación .....	66
Matriz de confusión .....	67
Grafica Precisión vs Confianza .....	67
Grafica Recall vs Confianza .....	68
Resultados del modelo.....	69
Curva F1 - Confidence .....	70
Grafica de mAP de las 5 clases detectadas.....	71
Metricas en Validación .....	71

## ÍNDICE DE TABLAS

Tabla 1 Principales malezas en plantaciones de maíz en la sierra. ....	20
Tabla 2 Principales malezas en plantaciones de papa de la sierra. ....	21
Tabla 3 Versiones YOLO .....	29
Tabla 4 Especificaciones técnicas del dron .....	44
Tabla 5 Herramientas para el desarrollo.....	47
Tabla 6 Información de los terrenos.....	49
Tabla 7 Cantidad de imágenes obtenidas. ....	52
Tabla 8 Banco de imágenes depuradas .....	54
Tabla 9 Recursos en Google Colab con Colab Pro.....	59
Tabla 10 Hiperparámetros usados.....	61
Tabla 11 Resultado de los experimentos variando los hiperparámetros .....	64
Tabla 12 Tiempos de detección de malezas .....	75
Tabla 13 Muestra de comparación de modelo vs expertos.....	77
Tabla 14 Ejemplo de comparación de modelo vs expertos.....	80
Tabla 15 Ejemplo de comparación de modelo vs expertos.....	82
Tabla 16 Niveles de afectación de los terrenos.....	85
Tabla 17 Valores de mAP (mean Average Precision) de trabajos relacionados .....	86

## RESUMEN

La tarea de identificar y cuantificar malezas representa un reto significativo en el campo de la agricultura, donde estas plantas no deseadas compiten por recursos esenciales con los cultivos, afectando su rendimiento y calidad. Tradicionalmente, este proceso ha sido complejo y subjetivo, influenciado por múltiples factores como el tamaño del terreno, la experiencia del agricultor y las condiciones ambientales.

En este proyecto, se introduce una aplicación web avanzada que utiliza YOLOv8, una técnica de deep learning, para la detección precisa de malezas mediante el análisis de imágenes tomadas por drones. Esta herramienta busca transformar el enfoque convencional, ofreciendo un método rápido y objetivo para la evaluación de malezas, facilitando así decisiones más efectivas en su manejo.

El primer capítulo del estudio se centra en explorar las características distintivas entre cultivos y malezas, examinando diversos tipos de malezas y las aplicaciones potenciales del deep learning en su detección. Se investigan las arquitecturas de algoritmos actuales y se analizan los retos existentes en este ámbito, estableciendo un contexto comparativo con investigaciones anteriores.

En el segundo capítulo, se describe en detalle el proceso de recolección de datos y la elección de YOLOv8 como el algoritmo principal. Se discute la evolución del algoritmo para optimizar la precisión de las predicciones, la implementación de la aplicación web a través de Gradio y las metodologías aplicadas durante todo el desarrollo.

El tercer capítulo se dedica a la evaluación de los resultados de las distintas versiones del algoritmo, culminando en la selección del modelo más efectivo para su integración en la aplicación web. Se mide la eficiencia de la solución tecnológica utilizando las métricas establecidas por la ISO 25010 y se emplea el análisis estadístico t de Student para confirmar la mejora en la eficiencia del proceso de identificación y cuantificación de malezas con la ayuda de la aplicación.

Finalmente, en la sección de discusión, se comparan los hallazgos de este estudio con los de investigaciones previas, evaluando la innovación y el impacto de esta nueva herramienta en el ámbito de la gestión agrícola.

## **ABSTRACT**

The identification and quantification of weeds present a significant challenge in agriculture, where these unwanted plants compete for essential resources with crops, impacting their yield and quality. Traditionally, this process has been complex and subjective, influenced by multiple factors such as the size of the land, the farmer's experience, and environmental conditions.

In this project, we introduce an advanced web application that utilizes YOLOv8, a deep learning technique, for the precise detection of weeds through the analysis of images captured by drones. This tool aims to transform the conventional approach, offering a quick and objective method for weed assessment, thereby facilitating more effective decisions in their management.

The first chapter of the study focuses on exploring the distinctive features between crops and weeds, examining various types of weeds and the potential applications of deep learning in their detection. Current algorithm architectures are investigated, and existing challenges in this field are analyzed, establishing a comparative context with previous research.

In the second chapter, the process of data collection and the choice of YOLOv8 as the primary algorithm are detailed. The evolution of the algorithm to optimize prediction accuracy, the implementation of the web application through Gradio, and the methodologies applied throughout the development are discussed.

The third chapter is dedicated to evaluating the results of the different versions of the algorithm, culminating in the selection of the most effective model for integration into the web application. The efficiency of the technological solution is measured using the metrics established by ISO 25010, and the Student's t-test statistical analysis is used to confirm the improvement in the efficiency of the weed identification and quantification process with the aid of the application.

Finally, in the discussion section, the findings of this study are compared with those of previous research, evaluating the innovation and impact of this new tool in the field of agricultural management.

## INTRODUCCIÓN

**TEMA:** Detección automática de malezas en plantaciones de maíz y/o papa utilizando imágenes adquiridas por dron y red neuronal convolucional YOLO

### PROBLEMÁTICA:

Dentro de la agricultura existen factores que reducen la velocidad y calidad del producto a obtener, entre plagas, enfermedades o plantas invasoras, estas últimas llamadas también malezas, son consideradas plantas de diferente especie a la esperada de cosechar y causantes de cambios desfavorables tanto en absorción de recursos como también reducción de la calidad del terreno. Una rápida detección de estas permiten una maximización del rendimiento del cultivo objetivo y hasta un menor uso de recursos importantes como son la luz, agua, fertilizantes y dinero en general, ya que de lo contrario se originan problemas como el uso indiscriminado de agroquímicos los cuales llegan a contaminar la tierra y aguas subterráneas, por ende puede impactar a la salud de las personas, plantas y animales del medio ambiente, también se llega a provocar erosión del suelo por parte de tractores al hacer uso de maquinaria pesada para la rápida aplicación de herbicidas en las plantaciones.

### Árbol de problemas



### Objetivo General

Identificar y cuantificar malezas existentes en plantaciones de maíz y/o papa haciendo uso de imágenes obtenidas por dron y la red neuronal YOLO

## Objetivos Específicos

- Desarrollar un marco teórico dirigido a la identificación de malezas a través de redes neuronales convolucionales
- Elaborar un modelo de IA que detecte malezas en cultivos de papa y/o maíz mediante el uso de imágenes tomadas por drones e implementarlo en una aplicación web evaluando la Eficiencia de desempeño con la norma ISO 25010
- Validar los resultados obtenidos junto a un especialista bajo métricas estadísticas y de IA

## Alcance

El objetivo principal de este proyecto es desarrollar una solución para detectar malezas en cultivos de maíz y/o papa mediante el uso de drones y técnicas de inteligencia artificial ya que en la actualidad esta toma largos periodos de tiempo, e implantación de fumigación tardíos. La solución final será de gran utilidad para los agricultores, ya que les permitirá identificar las malezas de manera rápida y precisa, mejorando la eficiencia en el control de malezas en sus cultivos. Para alcanzar este objetivo, se ha planteado una arquitectura que nos ayudara a comprender de una mejor manera el proyecto a desarrollar.

## Diagrama del proyecto



## Metodología

La metodología se desarrollará en base a los 3 objetivos específicos planteados. Objetivo 1: Se llevará a cabo una revisión exhaustiva y metódica de la literatura existente sobre técnicas de aprendizaje automático aplicadas a la detección de malezas en cultivos



mediante una SLR (Systematic Literature Review). El objetivo es identificar, evaluar e interpretar investigaciones disponibles y poner en contexto el presente trabajo a ser realizado para dar respuestas a un listado de preguntas de investigación propuestas. Algunas preguntas de investigación que se abordarán y guiarán este estudio son las siguientes: 11

- ¿Qué drones y cámaras están usando para la tarea de detección automática de malas hierbas?
- ¿Cuál es la altura y velocidad adecuada del dron para capturar imágenes y regiones de interés (ROIs) de alta resolución (px/cm)?
- ¿Qué espacio de color y formato de imágenes resulta adecuado?
- ¿En qué bandas del espectro electromagnético (visible, infrarrojo, ultravioleta) están trabajando?
- ¿Qué cultivos y tipos de malezas se están detectando y dónde?
- ¿Qué conjunto de datos se encuentran disponibles públicamente para el entrenamiento de un modelo de DL (Deep Learning)?
- ¿Cómo crear y anotar un conjunto de datos propio?
- ¿Qué arquitectura de CNN se usan para la detección automática de malas hierbas?
- ¿Qué hardware específico o servicios en la nube resulta adecuado para entrenar modelos de DL?
- ¿Qué métricas y gráficas se usan para validar el rendimiento de los modelos de DL?
- ¿Cómo afinar un modelo de DL para mejorar el rendimiento?

Para el desarrollo de la SLR se debe tener en cuenta sus tres etapas: la planificación, ejecución e informe se revisarán artículos y casos de estudio de CNNs que utilicen imágenes de drones, diferentes arquitecturas de CNN para determinar la más adecuada para el entrenamiento de un modelo de DL , también se revisarán herramientas y frameworks de programación para DL, planificaciones inteligentes de vuelo en drones finalmente se revisaran métricas y gráficas de rendimiento para validar modelos de DL.

Objetivo 2: Para cumplir con este objetivo, utilizaremos el dron DJI Mavic 2 Pro de la carrera de software para capturar imágenes de los campos de maíz y/o papa en diferentes etapas de crecimiento y condiciones climáticas en la finca experimental "La Pradera", propiedad de la Universidad Técnica del Norte en Ibarra, Ecuador. Las imágenes serán

revisadas y etiquetadas manualmente para identificar las malezas y las plantas. Se procesarán las imágenes para eliminar ruido y distorsión antes de entrenar un modelo de detección de objetos (YOLO) con las imágenes previamente etiquetadas. Finalmente, se desplegará el modelo entrenado en una plataforma cloud siguiendo la norma ISO/IEC 25010 para su uso por parte de los usuarios finales a través de una aplicación Web creada en Angular. Este proceso permitirá determinar la cantidad de malezas en la plantación y si es necesario o no tratarlas de acuerdo con su densidad.

Objetivo 3: Se llevará a cabo un análisis de resultados utilizando métricas estadísticas e inteligencia artificial para evaluar la eficiencia y viabilidad de su uso en comparación con los métodos actuales. Se compararán los resultados obtenidos con el dron con los obtenidos por métodos tradicionales para determinar la precisión y rapidez de detección de malezas. Además, se evaluará el impacto económico y ambiental de ambos métodos para determinar la viabilidad de su adopción a largo plazo. Este análisis permitirá identificar las fortalezas y debilidades del uso del dron para la detección de malezas y ayudará a mejorar la eficiencia y sostenibilidad del sistema en el futuro.

### **Justificación**

Es necesario llevar a cabo un análisis comparativo entre el uso del dron para la detección de malezas y los métodos actuales debido a que la agricultura es un sector clave para la economía global y la seguridad alimentaria, y la detección temprana y control de malezas juega un papel crucial en el éxito de un cultivo (Soledad et al., 2022). Los métodos actuales de detección de malezas, como la inspección manual y el uso de herbicidas, pueden ser costosos, tardados y tener impactos negativos en el medio ambiente (Ismael et al., 2022). El uso del dron para la detección de malezas proporciona una alternativa más eficiente y sostenible. El dron puede cubrir grandes extensiones de terreno de manera rápida y precisa, reduciendo el tiempo y el costo de inspección. Además, al utilizar tecnología de visión artificial para identificar malezas, se puede minimizar el uso de herbicidas, lo que es positivo tanto para el medio ambiente como para la salud humana.

# CAPÍTULO 1

## Marco Teórico

### 1.1. Fundamentación teórica

#### 1.1.1. Malezas

La maleza es un término amplio que se refiere a cualquier planta no deseada que crece en un lugar determinado, como un campo de cultivo, un jardín o incluso en un camino. Las malezas se propagan rápidamente y pueden ser difíciles de controlar, lo que las convierte en una amenaza para la salud y el bienestar de los cultivos y del ecosistema en general.

Además de competir por los recursos necesarios para el crecimiento de los cultivos, como el agua, los nutrientes y la luz solar, las malezas también pueden tener un impacto negativo en la calidad del suelo, reducir su capacidad de retener agua y nutrientes, y aumentar la erosión. También pueden actuar como hospedadores de plagas y enfermedades, lo que puede resultar en una disminución de la productividad de los cultivos (Ong et al., 2023).

#### 1.1.2. Control de malezas

La estrategia convencional para controlar las malezas en los cultivos implica el uso de herbicidas o la exploración manual, siendo la aplicación uniforme de herbicidas en todo el campo una técnica comúnmente empleada. Sin embargo, esta técnica puede resultar ineficiente debido a que una porción significativa del químico rociado no llega a las plantas no deseadas. Además, su uso puede generar residuos y contaminar el medio ambiente, aumentando así el riesgo de sobredosis, lo que a su vez incrementa la probabilidad de encontrar residuos de agroquímicos en los cultivos. Esto puede afectar negativamente la salud humana, ya que los residuos de agroquímicos pueden superar los límites permisibles (Reyes et al., 2016). En los cultivos de hileras, el control tecnificado de las malezas se realiza mediante la aplicación mecánica o manual de herbicidas. Los sistemas mecánicos son eficaces para eliminar las malezas que crecen entre las hileras de cultivo, pero no dentro de las líneas de siembra (Utstumo et al., 2018). Por lo tanto, la eliminación manual de hierbas muy cerca de las plantas se vuelve necesaria, lo que puede resultar más costoso que el procedimiento mecánico y no garantiza una eliminación completa de las

malezas.

### 1.1.3. Malezas en maíz y papas

## 2. Maíz

En las plantaciones de maíz, la presencia de malezas puede afectar significativamente el crecimiento y la producción del cultivo (Fig.1) (Zimdahl, 2018). Existen diferentes tipos de malezas que pueden estar presentes en los campos de maíz, y su aparición varía dependiendo de la etapa del cultivo y las condiciones ambientales (Soltani et al., 2016) .



Figura 1 Etapas del maíz (Zimdahl, 2018).

Las malezas anuales de verano, como el amaranto (*Amaranthus* spp.), la quinua (*Chenopodium album*) y el pasto de la jungla (*Echinochloa colona*), emergen durante la temporada de crecimiento del maíz y pueden ser más visibles durante las etapas iniciales de desarrollo del cultivo (Zambrano et al., 2021). Estas malezas compiten con el maíz por recursos como agua, nutrientes y luz, lo que puede afectar negativamente el rendimiento del cultivo (Zimdahl, 2018).

Por otro lado, las malezas perennes, como el pasto de trigo (*Elymus repens*) y el sorgo de alepo (*Sorghum halepense*), pueden establecerse antes o después de la siembra del maíz y persistir durante todo el ciclo de cultivo (William G. Johnson et al., 2000). Estas malezas pueden ser más visibles en las etapas intermedias y finales del desarrollo del maíz, ya que su crecimiento puede ser más lento al principio, pero luego acelerarse (Kruger et al., 2010).

Tabla 1 Principales malezas en plantaciones de maíz en la sierra.

Tipo de maleza	Nombre de la Maleza	Nombre Científico
Hoja ancha	Amaranto	<i>Amaranthus</i> spp.
	Verdolaga	<i>Portulaca oleracea</i>
	Quinua	<i>Chenopodium quinoa</i>
Hoja delgada	Sorgo de Alepo	<i>Sorghum halepense</i>

	Capín colorado	<i>Digitaria sanguinalis</i>
	Pasto Bermuda	<i>Cynodon dactylon</i>

(Zambrano et al., 2021)

## Papas

Existen diferentes tipos de malezas que pueden estar presentes en los campos de papas, y su aparición varía dependiendo de la etapa del cultivo (Fig. 2) y las condiciones ambientales (Jabran et al., 2023).

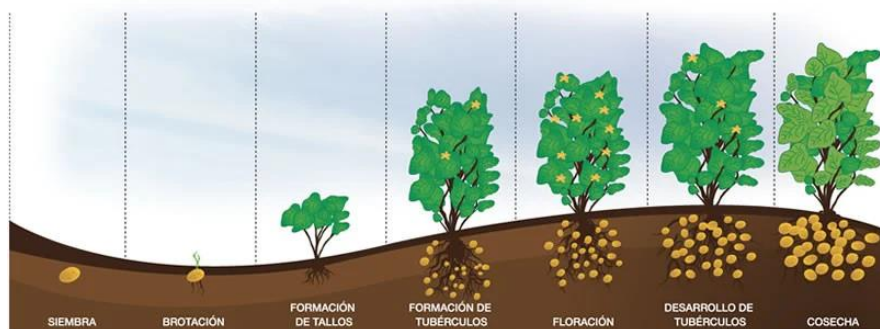


Figura 2 Etapas de la papa

Las malezas anuales de verano, como el amaranto (*Amaranthus* spp.), la quinua (*Chenopodium album*) y el pasto de la jungla (*Echinochloa colona*), emergen durante la temporada de crecimiento de las papas y pueden ser más visibles durante las etapas iniciales de desarrollo del cultivo. Estas malezas compiten con las papas por recursos esenciales, lo que puede afectar negativamente el rendimiento del cultivo (Zimdahl, 2018).

Según estudios realizados por el INIAP (Instituto nacional de investigación Agropecuaria), menciona las siguientes malezas como las principales en la región sierra:

Tabla 2 Principales malezas en plantaciones de papa de la sierra.

Tipo de Hoja	Nombre de la Maleza	Nombre Científico
Hoja Ancha	Lengua de vaca	<i>Rumex crispus</i>
	Diente de león	<i>Taraxacum officinale</i>
	Rábano	<i>Raphanus sativus</i>
	Corazón herido	<i>Corchorus olitorius</i>
Hoja Delgada	Berbenita	<i>Verbena officinalis</i>

	Kikuyo	Pennisetum clandestinum
--	--------	-------------------------

(Zambrano et al., 2021)

#### 1.1.4. Imágenes por dron

En los últimos años, la tecnología de drones ha sido utilizada para la identificación y monitoreo de malezas en plantaciones. Las imágenes de plantaciones tomadas por drones pueden ser analizadas con técnicas de procesamiento de imágenes y aprendizaje automático para identificar las malezas presentes en la plantación. Esto puede permitir una mejor gestión de las malezas y una reducción en el uso de herbicidas, lo que a su vez puede ser beneficioso para el medio ambiente y para la salud humana.

Según (Li et al., 2014), el uso de drones para la identificación de malezas en la agricultura puede mejorar la eficiencia y la precisión del monitoreo de malezas. Además, pueden reducir los costos y los riesgos asociados con el uso de métodos tradicionales de identificación de malezas. Otros estudios, como el de (López & Granados, 2016), también han demostrado que el uso de imágenes de drones puede ser una herramienta eficaz para la detección y el seguimiento de malezas en cultivos como la soja y el maíz. Estas investigaciones sugieren que la tecnología de drones y el procesamiento de imágenes pueden ser una herramienta valiosa para la gestión de malezas en la agricultura.

#### 1.1.5. Dron

Los drones utilizados para la identificación de malezas en plantaciones pueden ser de diferentes tipos, pero los más comunes son los drones multi rotores y los drones de ala fija. Los drones multi rotores son más maniobrables y pueden volar en espacios más reducidos, mientras que los drones de ala fija tienen mayor autonomía y pueden cubrir grandes áreas en un solo vuelo (Sishodia et al., 2020).

En cuanto a los sensores, los más utilizados para la identificación de malezas son los sensores ópticos y los sensores hiper espectrales. Los sensores ópticos pueden capturar imágenes RGB (rojo-verde-azul) de alta resolución, lo que permite una identificación visual de las malezas. Los sensores hiper espectrales, por otro lado, pueden capturar información en múltiples bandas espectrales, lo

que permite la identificación de las malezas por su firma espectral (del Cerro et al., 2021).

### 1.1.6. Condiciones de vuelo

La altura de vuelo recomendada para la captura de imágenes de plantaciones con drones suele estar en el rango de 5,10 y 30 metros, dependiendo del tamaño de la plantación y la resolución deseada de las imágenes (Volpato et al., 2021). Para la identificación de malezas, se recomienda una altura de vuelo lo suficientemente baja para obtener imágenes detalladas de la plantación, pero lo suficientemente alta para evitar obstáculos y riesgos de colisión con la vegetación.

En cuanto a la velocidad de vuelo, es recomendable mantener una velocidad constante y adecuada para garantizar la calidad de las imágenes. Para drones multi rotores, la velocidad de crucero recomendada se encuentra entre 2 y 4 metros por segundo (Fig.3), mientras que, para drones de ala fija, la velocidad de crucero recomendada es de alrededor de 10 a 15 metros por segundo (Volpato et al., 2021).

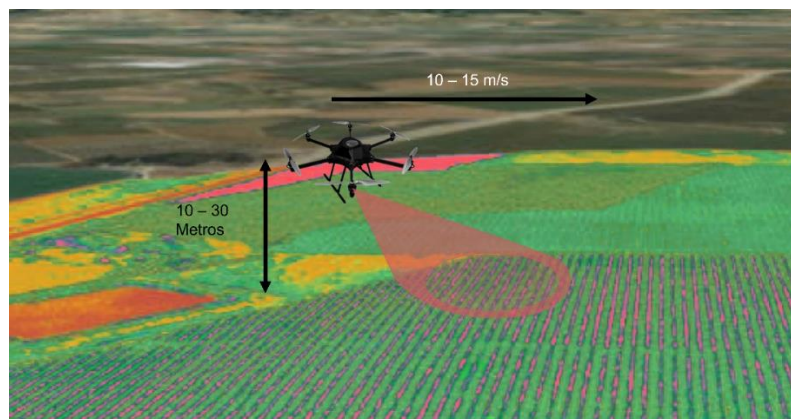


Figura 3 Condiciones de vuelo, adaptado de (Volpato et al., 2021).

## 1.2. Redes neuronales convolucionales

### 1.2.1. Inteligencia Artificial (IA)

La inteligencia artificial (IA) es un campo interdisciplinario de la ciencia que busca desarrollar sistemas y algoritmos capaces de imitar la cognición humana, lo que incluye el aprendizaje, el razonamiento, la percepción y la interacción en lenguaje natural (Ben Goertzel, 2021).

La IA se divide en varias subáreas, como el aprendizaje automático (machine learning), el aprendizaje profundo (deep learning), la robótica, la visión por computadora y el procesamiento del lenguaje natural (PNL), entre otras. En los últimos años, la IA ha experimentado avances significativos, impulsados principalmente por el desarrollo de algoritmos más sofisticados, la disponibilidad de grandes conjuntos de datos y el aumento en capacidad de cómputo (Brown et al., 2020).

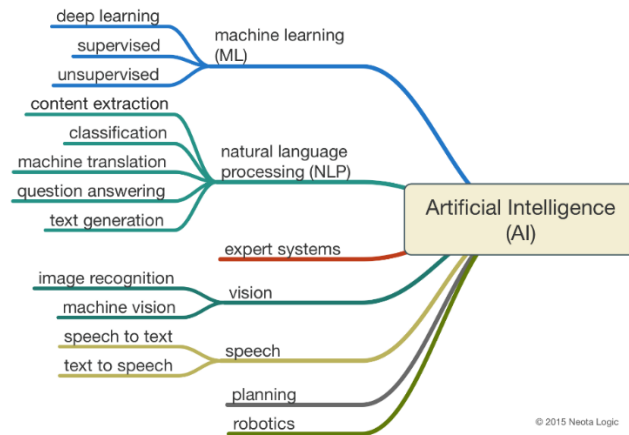


Figura 4 Ramas de la IA(Inteligencia Artificial) (Neota Logic,2015).

### 1.2.2. Visión Artificial

La visión artificial, también conocida como visión por computadora, es una rama de la inteligencia artificial que se enfoca en el desarrollo de técnicas y algoritmos que permiten a las máquinas interpretar y comprender información visual a partir de imágenes y videos (Szeliski, 2022). Esto implica la capacidad de identificar objetos, clasificarlos, reconocer patrones y formas como podemos observar en la (Fig.5), así como estimar la posición, movimiento y orientación de los objetos en el entorno (Howse & Minichino, 2020). La visión artificial se aplica en una amplia gama de campos, como la robótica, la vigilancia, la medicina, la agricultura, la industria automotriz y la realidad virtual, entre otros.



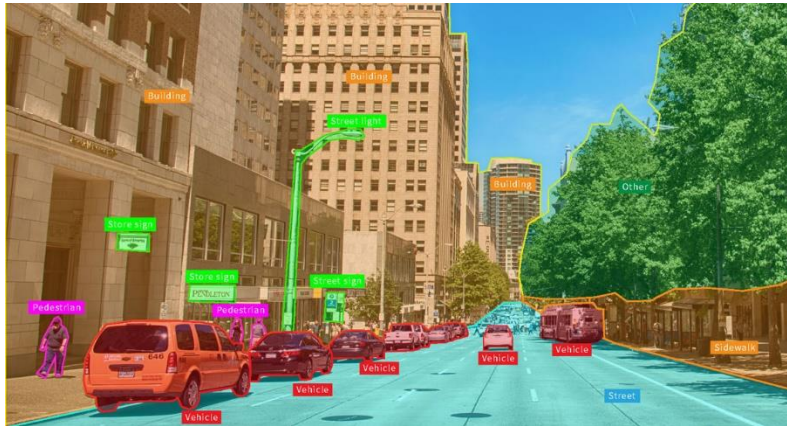


Figura 5 Ejemplo de detección por computadora (Appen, 2019).

Los avances en la visión artificial han sido impulsados en gran medida por el progreso en el aprendizaje profundo y las redes neuronales convolucionales (CNN), que han demostrado un rendimiento excepcional en tareas de reconocimiento y clasificación de imágenes (Krizhevsky et al., 2017). Además, la disponibilidad de grandes conjuntos de datos etiquetados y el aumento en capacidad de cómputo han contribuido significativamente a la rápida evolución de esta disciplina.

### 1.2.3. Redes neuronales convolucionales (CNN)

Las redes neuronales convolucionales (CNN, por sus siglas en inglés) son un tipo de modelo de aprendizaje profundo especialmente diseñado para el análisis y procesamiento de imágenes y señales (Lecun et al., 2015). Estas redes están compuestas por múltiples capas, incluyendo capas convolucionales, que aplican filtros para detectar características locales en los datos de entrada, y capas de agrupación, que reducen la dimensionalidad de los datos, mejorando así la eficiencia computacional y la invariancia a la escala y la posición (Ian Goodfellow & Yoshua Bengio, 2016).

La arquitectura de las CNN permite aprender jerarquías de características, desde las más simples hasta las más complejas, adaptándose a diversas aplicaciones, como la clasificación de imágenes, detección de objetos y procesamiento de lenguaje natural (Krizhevsky et al., 2021). Su capacidad para aprender de manera automática características relevantes ha llevado a un desempeño sobresaliente en diversas tareas y ha sido clave en el avance del campo de la inteligencia artificial (Lecun et al., 2015).

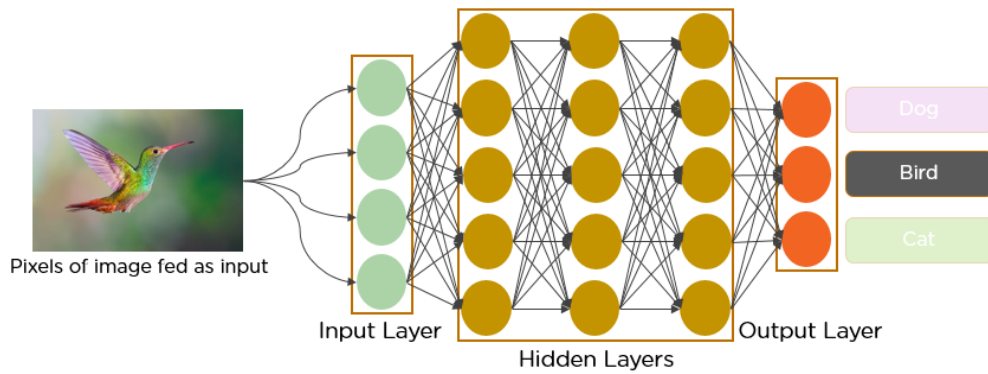


Figura 6 Estructura de una red neuronal convolucional (Manav Mandal, 2021)

#### 1.2.4. Componentes, arquitectura y técnicas de entrenamiento

Las redes neuronales convolucionales (CNN) han experimentado un gran avance en los últimos años, debido en gran parte a la mejora en sus componentes y arquitecturas (Zoph, 2020). Las capas convolucionales son fundamentales en las CNN, ya que aplican filtros para detectar características locales en los datos de entrada, como bordes, texturas y patrones (Zhang, 2019). Las capas de agrupación, por otro lado, reducen la dimensionalidad de los datos, lo que contribuye a mejorar la eficiencia computacional y la invariancia a la escala y la posición (Zhang, 2019).

### 3. Funciones de activación y normalización

Las funciones de activación juegan un papel crucial en las redes neuronales convolucionales, ya que introducen no linealidades en el modelo, permitiendo que la red aprenda representaciones más complejas y sofisticadas de los datos (Ramachandran, Zoph, & Le, 2019). Algunas funciones de activación populares incluyen ReLU (rectified linear unit), Leaky ReLU, ELU (exponential linear unit) y Swish (Ramachandran et al., 2019). Estas funciones de activación ayudan a evitar problemas como el desvanecimiento y explosión del gradiente, facilitando el entrenamiento de modelos más profundos y eficientes (He, 2020).

La normalización es otra técnica importante en las CNN, que tiene como objetivo estabilizar y acelerar el proceso de entrenamiento. La normalización por lotes (Batch Normalization) es una técnica comúnmente utilizada, que normaliza la entrada de cada capa en función de la media y la varianza del lote actual (Ioffe & Szegedy, 2019). Esta técnica no solo ayuda a mejorar la convergencia y la

estabilidad del entrenamiento, sino que también permite utilizar tasas de aprendizaje más altas y reduce la sensibilidad a la inicialización de los pesos (Ioffe & Szegedy, 2019).

### **1.2.5. Aprendizaje supervisado y no supervisado, inicialización de pesos, regularización y optimizadores**

El aprendizaje supervisado es un enfoque común en las redes neuronales convolucionales, donde se utiliza un conjunto de datos etiquetados para entrenar y ajustar los parámetros del modelo (Caron, 2021). El aprendizaje no supervisado, por otro lado, no requiere etiquetas y explora estructuras y patrones en los datos mediante técnicas como el autoencoder y el aprendizaje contrastivo (Krishnan & Isola, 2020). Ambos enfoques pueden ser aplicados en diferentes contextos y pueden complementarse para mejorar la eficiencia y el rendimiento de las CNN.

La inicialización de pesos, la regularización y los optimizadores son técnicas esenciales en el entrenamiento de las CNN. La inicialización adecuada de los pesos puede acelerar la convergencia y mejorar el rendimiento del modelo (Xiao, 2020). La regularización, como el abandono (dropout), ayuda a prevenir el sobreajuste y mejora la generalización del modelo (Gu & Liu, 2019). Los optimizadores, como el descenso de gradiente estocástico y Adam, ajustan los parámetros del modelo para minimizar la función de pérdida durante el entrenamiento, influyendo en la velocidad de convergencia y la calidad de la solución final (Kingma & Ba, 2020).

### **1.2.6. Herramientas y bibliotecas para trabajar con redes neuronales convolucionales**

#### **TensorFlow, Keras, PyTorch y otros marcos de trabajo de aprendizaje profundo**

Varias bibliotecas y marcos de trabajo de código abierto están disponibles para facilitar el desarrollo y la implementación de redes neuronales convolucionales. TensorFlow, desarrollado por Google Brain, es un popular marco de aprendizaje profundo que ofrece una amplia gama de herramientas y API para crear, entrenar y desplegar modelos de CNN (TensorFlow, 2023). Keras, una interfaz de alto nivel para TensorFlow, permite construir y entrenar

modelos de CNN de manera rápida y sencilla con un enfoque en la simplicidad y la facilidad de uso (Keras, 2023)

PyTorch, desarrollado por Facebook AI Research, es otro marco de trabajo ampliamente utilizado para el aprendizaje profundo y las redes neuronales convolucionales (PyTorch, 2023). Ofrece un enfoque dinámico para la construcción de grafos computacionales y proporciona una amplia gama de herramientas y utilidades para facilitar la investigación y el desarrollo en aprendizaje profundo. Estos marcos de trabajo y bibliotecas simplifican enormemente el proceso de diseño, entrenamiento, evaluación e implementación de redes neuronales convolucionales en una variedad de aplicaciones y dominios.

### **Conjuntos de datos y desafíos en la investigación de CNN**

Los conjuntos de datos y los desafíos en la investigación de redes neuronales convolucionales son fundamentales para impulsar el desarrollo y la innovación en el campo. Existen numerosos conjuntos de datos etiquetados disponibles para entrenar y evaluar modelos de CNN en tareas específicas. Algunos de los conjuntos de datos populares incluyen ImageNet (ImageNet, 2021) que contiene millones de imágenes etiquetadas en miles de categorías y ha sido un catalizador para el avance en la clasificación de imágenes; y COCO (Lin et al., 2021), que ofrece un conjunto de datos más diverso y desafiante para tareas como la detección de objetos, la segmentación y la descripción de imágenes.

Los desafíos y competencias, como el Desafío de clasificación visual a gran escala de ImageNet (ILSVRC) (Russakovsky et al., 2021) y la competencia COCO, ofrecen oportunidades para que investigadores y profesionales evalúen sus modelos y técnicas en entornos competitivos y compartan sus avances con la comunidad científica. Estos desafíos y competencias impulsan la innovación, promueven la colaboración y contribuyen al rápido desarrollo de las redes neuronales convolucionales y su aplicabilidad en una amplia gama de tareas y dominios.

#### **1.2.7. Red neuronal convolucional YOLO**

La CNN YOLO (You Only Look Once) es un algoritmo de detección de objetos en tiempo real que ha revolucionado el campo de la visión por computadora. Este algoritmo presenta una innovadora manera de detectar objetos en imágenes y video, mejorando la precisión y la velocidad en comparación con otros métodos tradicionales (Redmon & Farhadi, 2018). Las versiones de YOLO han venido variando con el paso del tiempo.

*Tabla 3 Versiones YOLO*

<b>Versión</b>	<b>Año</b>	<b>Desarrollador</b>
YOLO	2015	Universidad de Washington
YOLO9000	2016	Universidad de Washington
YOLOv2	2017	Universidad de Washington
Fast YOLO	2017	Universidad de Washington
YOLOv3	2018	Universidad de Washington
YOLOv4	2020	Ultralytics
YOLOv5	2020	Ultralytics
YOLOv6	2021	Megvii
YOLOv7	2022	Ultralytics
YOLOv8	2023	Ultralytics

Redmon y Farhadi (2018) presentaron YOLOv3, una versión mejorada del algoritmo YOLO que supera a sus predecesores en términos de precisión y velocidad. YOLOv3 combina varias técnicas, como la detección de objetos a múltiples escalas y la utilización de funciones de activación como Leaky ReLU, para mejorar la detección de objetos pequeños y aumentar la precisión general.

En 2020, Bochkovskiy, Wang y Liao presentaron YOLOv4, una nueva versión que mejoró aún más la velocidad y el rendimiento de la detección de objetos en tiempo real. YOLOv4 incorporó mejoras en la arquitectura de la red neuronal, como la utilización de la red CSPDarknet53 y módulos SPP (Spatial Pyramid Pooling), que aumentan la eficiencia y la precisión en la detección de objetos (Bochkovskiy & Liao, 2020).

Wang, Liao y Bochkovskiy (2021) propusieron YOLOv5, otra versión mejorada que optimiza la eficiencia y el rendimiento en aplicaciones de detección de objetos en tiempo real. YOLOv5 incluye mejoras en la arquitectura, como la implementación de EfficientDet-D0 y la adición de módulos PANet (Path Aggregation Network), que permiten un mejor rendimiento en la detección y clasificación de objetos.

### 1.2.8. YOLOv8

YOLOv8 representa un avance significativo en la serie de algoritmos YOLO para la detección de objetos, destacándose por su capacidad para manejar diversas aplicaciones en tiempo real y entornos desafiantes. Basado en las referencias proporcionadas, aquí está un resumen consolidado que destaca sus características y aplicaciones:

**Arquitectura de YOLOv8:** YOLOv8 utiliza CSPDarknet53 como su arquitectura principal. Esta red neuronal profunda extrae características en múltiples resoluciones, lo que permite detectar objetos a diferentes escalas y con distintos niveles de detalle. Además, YOLOv8 introduce módulos repetidos y múltiples cabezales de detección en su arquitectura, lo que mejora la precisión en la detección de objetos (Terven & Cordova-Esparza, 2023).

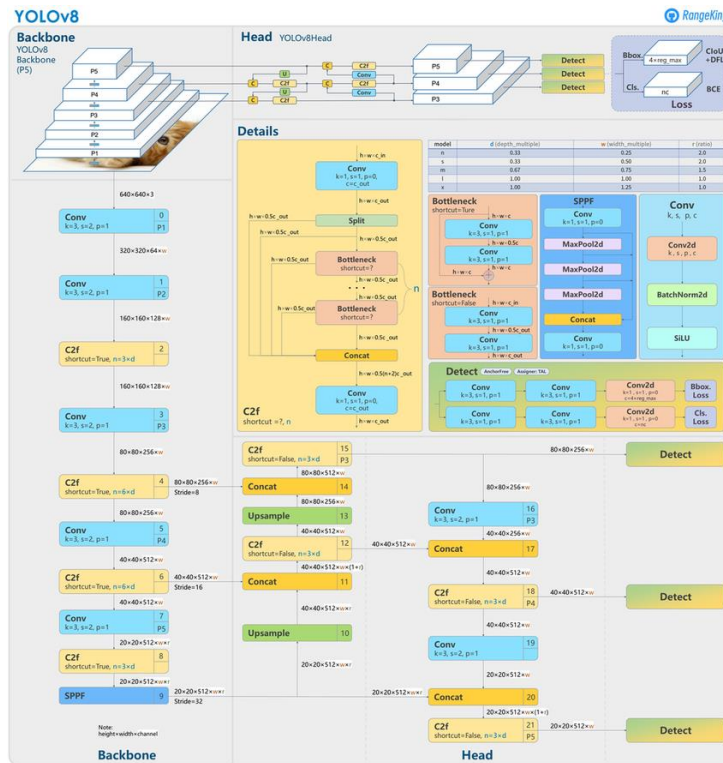


Figura 7 Arquitectura YOLOv8 (Schcolnik-Elias et al., 2023).

**Función de Pérdida y Reglas de Actualización:** YOLOv8 utiliza una función de pérdida generalizada que incorpora pesos de pérdida individuales y un término de regularización con decaimiento de peso. Esta función de pérdida incluye la pérdida Clou (Complete Intersection over Union) para la regresión de cuadros delimitadores y la entropía cruzada binaria para la clasificación (Reis et al., 2023).

**Detección sin Anclas en YOLOv8:** A diferencia de los modelos anteriores de YOLO, YOLOv8 es un modelo sin anclas, lo que significa que predice directamente el centro de un objeto en lugar de un desplazamiento desde un cuadro delimitador conocido. Esto reduce el número de predicciones de cuadros delimitadores y acelera el proceso de Supresión No Máxima (NMS), un paso de posprocesamiento complicado que filtra las detecciones candidatas después de la inferencia (Xiao et al., 2023).

**Mejoras en la Precisión y Entrenamiento de YOLOv8:** La investigación de YOLOv8 se motivó principalmente por la evaluación empírica en el benchmark COCO. Cada ajuste en la red y la rutina de entrenamiento se validó mediante nuevos experimentos para verificar su efecto en el modelado de COCO. Además, YOLOv8 implementa la augmentación de mosaico durante el entrenamiento, pero esta se desactiva en las últimas diez épocas de

entrenamiento para evitar la degradación del rendimiento (Ultralytics, 2023).

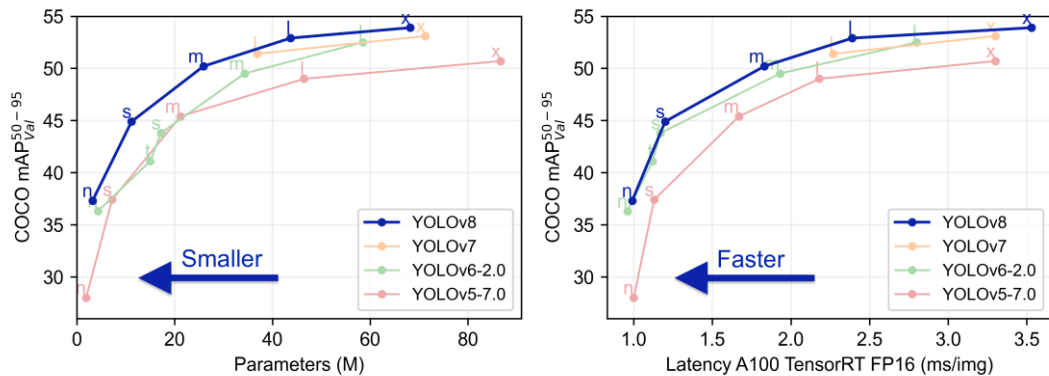


Figura 8 Comparación de rendimiento YOLO (Ultralytics, 2023).

En resumen, la CNN YOLO ha evolucionado significativamente desde su primera versión, mejorando continuamente su rendimiento y eficiencia en la detección de objetos en tiempo real. Estas mejoras han sido posibles gracias a la incorporación de nuevas técnicas y arquitecturas en las diferentes versiones de YOLO, lo que ha llevado a una mayor precisión y velocidad en la detección de objetos (Fig.8).

### 1.2.9. Metodología de desarrollo

La metodología de desarrollo de software es un conjunto estructurado de prácticas, procesos y técnicas que sirven como guía para la creación y mantenimiento de sistemas de software. Estas metodologías abarcan enfoques y actividades que abordan el diseño, la implementación, la prueba y la documentación de software, facilitando la colaboración y la comunicación efectiva entre los miembros del equipo de desarrollo y garantizando la entrega exitosa de proyectos de software de alta calidad (Sommerville, 2016). En el desarrollo de un proyecto de investigación es importante usar metodologías que ayuden a procesar y extraer información mediante una serie de pasos, para esta investigación se implementará la metodología KDD.

### 1.2.10. Metodología KDD

La metodología KDD (Knowledge Discovery in Databases) es un proceso integral para descubrir patrones útiles y conocimiento relevante a partir de grandes conjuntos de datos. La metodología KDD es ampliamente utilizada en el campo de la minería de datos y ha sido objeto de numerosas investigaciones y aplicaciones en diferentes dominios (Fayyad, Piatetsky-Shapiro, 1996).



La metodología KDD sigue un proceso iterativo que consta de varias etapas (Wu et al., 2014):

- **Selección de datos:** En esta etapa, se identifican y seleccionan las fuentes de datos relevantes para el análisis. La calidad de los datos seleccionados es crucial para obtener resultados precisos y significativos.
- **Preprocesamiento de datos:** Los datos seleccionados pueden contener ruido, inconsistencias o valores faltantes. En esta etapa, se limpian y transforman los datos para prepararlos para el análisis. Las técnicas comunes incluyen la imputación de valores faltantes, la eliminación de duplicados y la normalización de los datos.
- **Transformación de datos:** Los datos preprocesados se transforman en un formato adecuado para el análisis mediante la minería de datos. Esto puede incluir la creación de nuevas variables, la discretización de variables continuas o la aplicación de técnicas de reducción de dimensionalidad.
- **Minería de datos:** Esta es la etapa central del proceso KDD, en la que se aplican algoritmos y técnicas de minería de datos para identificar patrones y conocimiento relevante en los datos transformados. Los enfoques comunes incluyen clasificación, agrupamiento, regresión y detección de anomalías.
- **Interpretación y evaluación:** Los resultados de la minería de datos deben ser interpretados y evaluados para determinar su validez y utilidad. Esto puede implicar la validación cruzada de los resultados, la comparación con conocimientos previos o la evaluación de su relevancia en el contexto del problema en cuestión.
- **Consolidación del conocimiento:** El conocimiento adquirido se integra en el dominio de aplicación para mejorar la toma de decisiones o para refinar el modelo de minería de datos.

### 1.2.11. Metodología Kanban

Kanban es una metodología de desarrollo y gestión de proyectos ágil que se

originó en la industria de la fabricación y ha sido adoptada con éxito en el ámbito del desarrollo de software (Ahmad, 2013). La metodología Kanban se centra en la visualización del flujo de trabajo, la limitación del trabajo en curso (WIP) y la mejora continua del proceso, lo que permite a los equipos de desarrollo identificar y eliminar cuellos de botella, reducir el tiempo de ciclo y aumentar la eficiencia general del proceso.

El enfoque visual de Kanban implica el uso de un tablero Kanban que divide el proceso de desarrollo en columnas, cada una representando una etapa diferente del flujo de trabajo, como "por hacer", "en curso" y "terminado". Las tareas o elementos de trabajo se representan como tarjetas que se mueven a través de las columnas a medida que avanzan en el proceso de desarrollo (Ahmad, 2013). Al visualizar el flujo de trabajo de esta manera, los equipos pueden identificar fácilmente los bloqueos y áreas de mejora, facilitando la comunicación y la colaboración entre los miembros del equipo.

La limitación del trabajo en curso es otro principio clave de la metodología Kanban que busca minimizar el multitasking y garantizar que los miembros del equipo se centren en las tareas prioritarias. Al establecer límites de WIP para cada etapa del flujo de trabajo, Kanban asegura que el equipo no asuma más trabajo del que puede manejar, lo que reduce el tiempo de ciclo y mejora la calidad del producto entregado (Ahmad, 2013).

### 1.3. Aplicación Web

#### 1.3.1. Detalles del sistema

El enfoque propuesto para lograr una detección temprana de malezas se basa en la combinación de imágenes capturadas por un dron y un sistema de predicción implementado en una aplicación web. Esta aplicación albergará un modelo YOLO previamente entrenado, encargado de analizar las imágenes proporcionadas por el dron. Así, se podrá calcular la densidad de malezas presentes en la plantación de manera eficiente y precisa. Para llevar a cabo este proceso, es necesario seguir los pasos que se describen a continuación:

1. Crear una base de datos que contenga imágenes capturadas por un dron, las cuales serán utilizadas para el entrenamiento del modelo de inteligencia

artificial.

2. Etiquetar adecuadamente cada una de las imágenes con el fin de facilitar el aprendizaje supervisado del modelo.
3. Una vez completado el entrenamiento del modelo YOLO y asegurada su precisión, se procederá a desarrollar una aplicación web.
4. Se realizará el despliegue en Google Colab haciendo uso de la herramienta Gradio.
5. Posteriormente, se ingresarán nuevas imágenes al aplicativo para ser procesadas por el modelo entrenado para identificar la cantidad de malezas presentes.
6. Finalmente, se calculará la densidad de malezas en la plantación, utilizando la información proporcionada por el análisis de las imágenes y determinará la eficiencia y beneficios frente a métodos tradicionales.

De esta manera, se establece un proceso claro y estructurado para llevar a cabo el análisis de las imágenes obtenidas por el dron y la identificación de malezas en la plantación.



Figura 9 Etapas del aplicativo web (Elaboración Propia).

### 1.3.2. Lenguaje de programación

Python es un lenguaje de programación de alto nivel que ha ganado gran popularidad en diversas áreas científicas y técnicas, incluyendo inteligencia artificial, big data, aprendizaje profundo y desarrollo web (Guttag, 2016; McKinney, 2018; Müller & Guido, 2016). Su versatilidad, legibilidad y amplia biblioteca de recursos lo convierten en una excelente opción para manejar grandes volúmenes de datos y abordar problemas científicos complejos (Guttag, 2016).

Algunas características clave de Python que contribuyen a su éxito en la comunidad científica incluyen:

**Lenguaje interpretado:** A diferencia de los lenguajes compilados, Python utiliza un intérprete para ejecutar el código fuente directamente sin necesidad de compilarlo previamente a código máquina (Guttag, 2016). Esto facilita la depuración y agiliza el proceso de desarrollo.

**Multiplataforma:** Python es compatible con una amplia gama de sistemas operativos, incluyendo Windows, macOS, Unix y Linux, lo que permite a los desarrolladores trabajar en diferentes entornos sin problemas (McKinney, 2018).

**Multiparadigma:** Python admite varios paradigmas de programación, como la programación orientada a objetos, estructurada, imperativa y funcional (Müller & Guido, 2016). Esto permite a los desarrolladores utilizar el enfoque más apropiado para abordar un problema específico.

**Tipado dinámico:** En Python, no es necesario declarar explícitamente el tipo de dato de una variable, lo que facilita la escritura y modificación del código (Guttag, 2016). Además, las variables pueden cambiar de tipo en cualquier momento durante la ejecución del programa.

Durante el desarrollo del proyecto, se emplearán diversas librerías de Python, cada una con funciones específicas y aplicaciones relevantes para el trabajo en cuestión. A continuación, se describen las librerías seleccionadas:

**Sys:** Este módulo, incluido en el intérprete de Python, proporciona acceso a funciones y variables del sistema que permiten interactuar con el entorno de ejecución (Python Software Foundation, 2021a).

**Os:** Esta librería permite interactuar con el sistema operativo y gestionar archivos y estructuras de directorios de manera compatible con diversas plataformas (Python Software Foundation, 2021b).

**Math:** Este módulo ofrece funciones matemáticas de alto nivel, como operaciones trigonométricas y logaritmos, basadas en el estándar C (Python Software Foundation, 2021c).

**Numpy:** Es una librería especializada en el manejo de matrices y arreglos multidimensionales, ideal para realizar cálculos numéricos y análisis de datos eficientes (Harris et al., 2020).

**Tensorflow:** Esta plataforma de código abierto facilita la creación, entrenamiento y ejecución de redes neuronales profundas mediante aprendizaje automático, y es especialmente útil para la clasificación y reconocimiento de imágenes (Abadi et al., 2016).

**Keras:** Es una biblioteca de código abierto para la creación de redes neuronales que permite utilizar diversos frameworks de aprendizaje automático (Chollet, 2015).

**Matplotlib:** Esta librería multiplataforma permite crear gráficos en 2D de alta calidad e interactivos, siendo ampliamente utilizada en proyectos de ciencia de datos (Hunter, 2007).

#### 1.4. Trabajos relacionados

Listado de artículos científicos relacionados al tema propuesto:

	Tipo de cultivo	Título del artículo	Tipo de modelo	Referencia
1	Caña de azúcar	Weed Detection in Sugarcane Fields Using Image Processing Techniques	Método tradicional de procesamiento de imágenes	(Sheeraz, 2021)
2	Maíz	2.CNN feature based graph convolutional network for weed and crop recognition in smart farming	Modelo Faster R-CNN	(Honghua, 2020)
3	Maíz	CNN Based Automated Weed Detection System Using UAV Imagery	Modelo Faster R-CNN	(Mohd, 2022)
4	Sésamo	Weed detection in sesame fields using a YOLO model with an enhanced attention mechanism and feature fusion	Modelo YOLOv3	(Jiqing, 2022)
5	Pimiento	Deep convolutional neural network models for weed detection in polyhouse grown bell peppers	Modelos AlexNet, GoogLeNet, InceptionV3 y Xception	

### 1. Weeds Detection and Classification using Convolutional Long-Short-Term Memory

Resumen:

Los autores utilizaron imágenes de nueve tipos de malezas para entrenar y evaluar cuatro modelos basados en redes neuronales convolucionales y memoria a largo plazo, que tienen como objetivo detectar y clasificar las malezas de forma automática y precisa. El modelo que obtuvo el mejor rendimiento fue el CNN-LSTM apilado bidireccional, que logró una precisión media del 99.36% y un valor F1 del 99.37%.

### 2. CNN feature based graph convolutional network for weed and crop recognition in smart farming

Resumen:

Los autores utilizaron un conjunto de datos de 10.000 imágenes aéreas de campos de maíz con diferentes tipos de malezas y cultivos. Las imágenes se dividieron en 80% para entrenamiento y 20% para prueba. Los autores extrajeron características visuales de las imágenes usando una red neuronal convolucional pre-entrenada (VGG16) y luego aplicaron una red convolucional gráfica para capturar las relaciones espaciales entre los píxeles. Los autores compararon el rendimiento de su método con otros cuatro métodos basados en redes neuronales convolucionales y redes neuronales recurrentes. El método propuesto obtuvo la mayor precisión media (98.76%) y el menor tiempo de ejecución (0.12 segundos por imagen).

### 3. CNN Based Automated Weed Detection System Using UAV Imagery

Resumen:

En el estudio analizado, se aplica una red neuronal convolucional (CNN) para categorizar segmentos de imágenes obtenidas por vehículos aéreos no tripulados (UAV) en cuatro clases distintas: suelo, soja, pasto y hoja ancha. Los investigadores trabajaron con un conjunto de datos compuesto por 4400 imágenes y 15336 segmentos, y optimizaron los

parámetros del modelo CNN utilizando validación cruzada. El modelo CNN logró una precisión global del 99.44% y una satisfacción del usuario del 98.75% en la detección de malezas.

En relación con el enfoque propuesto para la detección de malezas en plantaciones de maíz usando drones y CNN YOLO, se pueden identificar similitudes y diferencias con el estudio mencionado. Ambos métodos emplean drones y CNN para identificar malezas en cultivos, pero el estudio se enfoca en la soja, mientras que el enfoque propuesto se centra en el maíz. Adicionalmente, el estudio utiliza un modelo CNN genérico, mientras que en el enfoque propuesto se implementa un modelo CNN específico llamado YOLO (You Only Look Once), que resulta más rápido y eficiente en la detección de objetos en tiempo real. Por lo tanto, el enfoque propuesto podría presentar ventajas en términos de velocidad y rendimiento en comparación con el estudio analizado.

#### 4. Weed detection in sesame fields using a YOLO model with an enhanced attention mechanism and feature fusion

El presente artículo aborda el impacto negativo de la aplicación indiscriminada de herbicidas en el medio ambiente y la economía, proponiendo el uso de métodos de visión por computadora, específicamente la detección de objetos, para mejorar el manejo de malezas y la aplicación específica de herbicidas. Para ello, los autores crearon una base de datos anotada con 374 imágenes en color RGB, organizadas en clases de malezas monocotiledóneas y dicotiledóneas, obtenidas mediante un sistema aéreo no tripulado (UAS) en parcelas de maíz y soja en Indiana. Tras anotar manualmente 25,560 instancias de malezas, se dividió la base de datos en cuatro subconjuntos para entrenar el modelo de aprendizaje profundo YOLOv3 en cinco experimentos independientes. El mejor resultado se obtuvo con el conjunto de imágenes tomadas a 10 m de altura, alcanzando una precisión media (AP) del 91,48% y 86,13% para malezas monocotiledóneas y dicotiledóneas respectivamente, con un umbral de intersección sobre unión (IoU) del 25%. Este estudio destaca la necesidad de desarrollar bases de datos anotadas de malezas para evaluar modelos de aprendizaje profundo y confirma resultados de

investigaciones previas sobre la identificación de malezas en condiciones de campo mediante la detección de objetos.

5. Deep convolutional neural network models for weed detection in polyhouse grown bell peppers

Resumen:

El artículo presenta un procedimiento para localizar *Colchicum autumnale* en imágenes de alta resolución de drones en el rango de luz visible. La metodología emplea redes neuronales convolucionales para detectar la posición de las flores, utilizando imágenes etiquetadas manualmente mejoradas mediante la ampliación de imágenes. La detección se evaluó en sitios de pastizales no incluidos en el entrenamiento, logrando detectar el 88,6% de las flores en el conjunto de datos de prueba, lo que indica su aplicabilidad en el control específico de malezas.

### 1.5. Preguntas Científicas

1. **¿Qué tipos de drones y cámaras se emplean en la detección automática de malas hierbas?**

Los drones empleados para esta tarea se clasifican en dos categorías según su forma de vuelo: multi rotores y de ala fija.

2. **¿Cuál es la altura y velocidad óptimas del dron para obtener imágenes y regiones de interés (ROIs) de alta resolución (px/cm)?**

La altura sugerida varía en función de la calidad de la cámara, siendo 5, 10 y 15 metros las alturas recomendadas. Para drones multi rotor, se recomienda una velocidad de crucero de 2 a 4 m/s.

3. **¿Qué espacio de color y formato de imágenes son apropiados?**

El espacio de color RGB es el adecuado para la captura de imágenes, y se recomienda guardar las imágenes en formatos JPG o PNG.

4. **¿En qué bandas del espectro electromagnético (visible, infrarrojo, ultravioleta) se opera?**

El dron utilizado cuenta con una cámara 4K que captura imágenes en el espectro visible.

5. **¿Qué cultivos y tipos de malezas se detectan y en qué ubicación?**



Las imágenes provienen de plantaciones de papa en el sector de Canchaguano, en la parroquia Montufar, provincia del Carchi, Ecuador.

**6. ¿Qué conjuntos de datos públicos están disponibles para entrenar un modelo de aprendizaje profundo (Deep Learning)?**

Existen diversos conjuntos de datos públicos, como MNIST, CIFAR-10, CIFAR-100, ImageNet, COCO, Stanford Dogs and Cats y UCI Machine Learning Repository, utilizados en clasificación de imágenes, detección de objetos, segmentación semántica y otras tareas de visión computacional.

**7. ¿Cómo crear y anotar un conjunto de datos propio?**

- Obtención de imágenes: Es necesario capturar imágenes de plantaciones de papa y/o maíz con un dron, siguiendo los parámetros óptimos de vuelo para obtener imágenes claras de plantas y malezas.
- Etiquetado: Se deben preparar las imágenes para identificar plantas y malezas con mayor facilidad, y luego realizar el etiquetado manual, clasificando áreas específicas de interés, en este caso, plantas y malezas.

**8. ¿Qué arquitecturas de redes neuronales convolucionales (CNN) se emplean en la detección automática de malas hierbas?**

Algunas de las CNN más utilizadas son YOLO, Faster R-CNN y Mask R-CNN.

**9. ¿Qué hardware específico o servicios en la nube son adecuados para entrenar modelos de aprendizaje profundo?**

Los servicios en la nube más comunes son Azure, Google Cloud y Amazon Web Services.

**10. ¿Qué métricas y gráficas se utilizan para validar el rendimiento de los modelos de aprendizaje profundo?**

Las métricas más comunes incluyen matriz de confusión, F1 score, curva ROC, precisión (accuracy), exhaustividad (recall) y curva PR.

**11. ¿Cómo ajustar un modelo de aprendizaje profundo para mejorar su rendimiento?**

Para optimizar los resultados, se pueden aplicar técnicas como el ajuste de hiper parámetros, preprocesamiento de datos, aumento de datos de entrenamiento y regularización.

## CAPÍTULO 2

### Desarrollo del proyecto

Esta tesis se centra en el desarrollo de un sistema de detección de malezas en plantaciones de papa utilizando un dron y la red neuronal convolucional YOLO. Para ello, se han recopilado imágenes mediante el uso de dron, se realiza el etiquetado manual y se lleva a cabo el preprocesamiento de datos para crear un dataset adecuado para el entrenamiento del modelo. A continuación, se implementa la red neuronal convolucional YOLOv8 para lograr una detección y clasificación precisa de las malezas en las imágenes. Además, se lleva a cabo la implementación del sistema en la nube, lo que permite su acceso remoto y escalabilidad. Mediante la infraestructura en la nube, los agricultores pueden utilizar esta solución de detección de malezas desde cualquier ubicación, a través de una interfaz en línea. Asimismo, se adoptará la metodología Kanban para la organización y gestión de las tareas del proyecto, y se verificará el cumplimiento de la norma ISO 25010, en particular en el aspecto de eficiencia, durante el despliegue del sistema en la nube, asegurando así un rendimiento óptimo y un uso eficiente de los recursos (Ahmad et al., 2013; Corceiro et al., 2023; Ravello et al., 2014).

#### 2.1. Metodología y herramientas utilizadas

Para garantizar un desarrollo sistemático y coherente de este proyecto, se adoptarán metodologías probadas y se hará uso de una variedad de herramientas específicas. Las metodologías que se aplicarán incluyen el proceso de Descubrimiento de Conocimiento en Bases de Datos (KDD) y la metodología Kanban, ambas elegidas por su relevancia y eficacia para este tipo de proyecto. Por otro lado, se utilizarán herramientas específicas para la recopilación y etiquetado de datos, desarrollo y evaluación del modelo, y para la gestión del proyecto. Esta combinación de metodologías y herramientas permitirá un avance sistemático y eficaz del proyecto.

##### 2.1.1. Metodología KDD y Kanban

El proceso KDD es un método ampliamente aceptado que se utilizará en este

contexto para la detección de objetos. Este proceso se compone de cinco etapas principales: selección de datos, preprocesamiento, transformación, detección de objetos y evaluación (Mariscal et al., 2010).

En el contexto de este proyecto, la selección de datos implica la elección de las imágenes adquiridas mediante el dron que son representativas y de alta calidad. El preprocesamiento de datos consiste en la extracción de las imágenes capturadas en los vuelos del dron, la limpieza de las imágenes y su etiquetación para su posterior uso en el entrenamiento del modelo.

La transformación de datos involucra la etiquetación de las imágenes, que se realiza utilizando la herramienta Roboflow. La detección de objetos se lleva a cabo mediante el uso de la red neuronal convolucional YOLOv8 para el entrenamiento del modelo. Este proceso implica la identificación y clasificación de objetos específicos, en este caso las malezas en plantaciones de papa, dentro de las imágenes (Roboflow, 2023).

La etapa final, la evaluación, involucra la comprobación de la eficacia del modelo mediante pruebas con imágenes fuera del conjunto de datos de entrenamiento. Esta evaluación permitirá verificar la precisión y fiabilidad del modelo para detectar y clasificar las malezas en diferentes condiciones de iluminación y perspectiva.

La metodología Kanban será utilizada para organizar y gestionar las tareas del proyecto. Este enfoque implica la visualización de todas las tareas en un tablero Kanban, lo que permite un seguimiento fácil y efectivo del progreso del proyecto. Además, Kanban se enfoca en limitar la cantidad de trabajo en progreso, lo que ayuda a evitar el sobre-apego y facilita un enfoque claro en las tareas prioritarias (Medeiros et al., 2018).

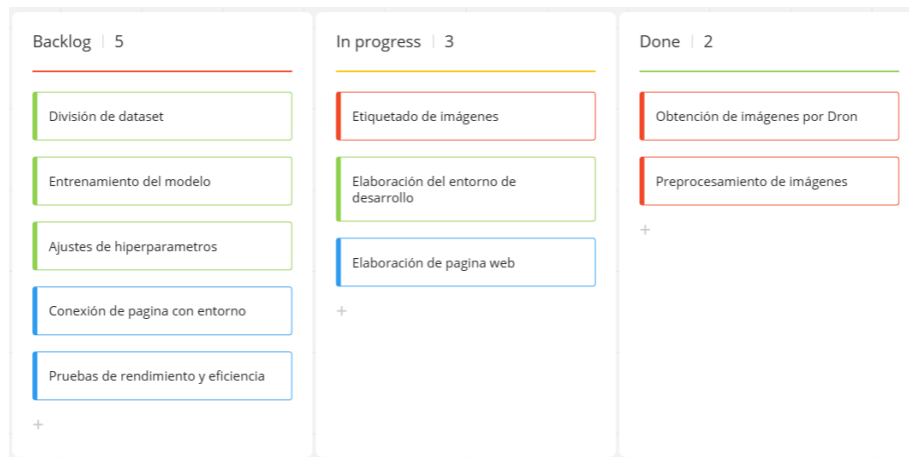


Figura 10 Tareas programadas con Kanban

## 2.1.2. Herramientas de Desarrollo y Evaluación

Durante el desarrollo de este proyecto, se utilizarán diversas herramientas que desempeñarán roles clave en las diferentes etapas. Para la recopilación de datos, se empleará un dron para adquirir imágenes de la plantación de papa, mientras que Python será utilizado para crear scripts que facilitarán la extracción de imágenes a partir de los videos capturados. Posteriormente, se utilizará la herramienta Roboflow para etiquetar las imágenes de manera eficiente.

Tabla 4 Especificaciones técnicas del dron

Componente	Información
<b>Sensor</b>	1" CMOS
<b>Píxeles efectivos</b>	20 millones
<b>Lente FOV</b>	Aprox. 77°
<b>Tamaño de imagen</b>	5472×3648
<b>Resolución de video</b>	4K: 3840×2160 24/25/30p
	FHD: 1920×1080 24/25/30/48/50/60p
<b>Tasa de bits</b>	Máx. 100Mbps
<b>Peso al despegar</b>	907 g

<b>Dimensiones</b>	Doblada: 214 × 91 × 84 mm
	Desplegada: 322 × 242 × 84 mm
<b>Distancia diagonal</b>	354 mm
<b>Velocidad máxima</b>	Ascenso: 5 m/s (modo S)
	Descenso: 3 m/s (modo P)
<b>Velocidad máxima</b>	72 km/h (modo S)
<b>Altitud máxima</b>	6000 m
<b>Tiempo máximo de</b>	vuelo: 31 min (a 25 km/h)
<b>Resistencia máxima</b>	al viento: 29–38 km/h
<b>Frecuencia de</b>	operación:
<b>operación</b>	2.400 - 2.483 GHz
	5.725 - 5.850 GHz
<b>Rango de precisión</b>	Vertical: ± 0,1 m (visual)
<b>de desplazamiento</b>	Horizontal: ± 0,3 m (visual)
<b>Almacenamiento</b>	interno: 8GB



*Figura 11 Dron DJI Mavic 2 Pro*

Fuente: (DJI, 2018)

En cuanto al desarrollo del modelo, se utilizará Google Collab como el entorno de desarrollo para escribir y gestionar el código necesario para el entrenamiento de la red neuronal convolucional YOLOv8. Además, las pruebas de entrenamiento y detección serán realizadas en esta misma plataforma ya que permite contratar recursos potentes de GPU, CPU y memoria RAM mediante el plan Colab PRO (Google, 2024).

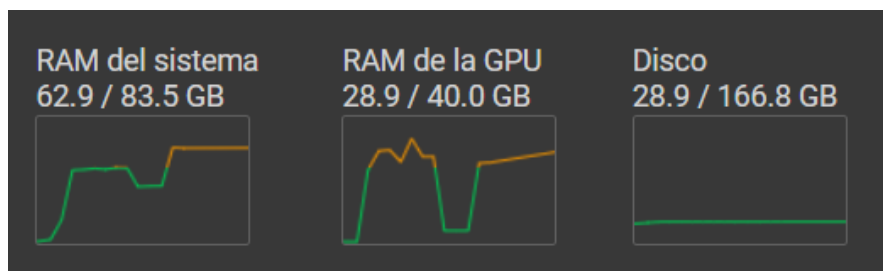


Figura 12 Recursos ofrecidos en Colab Pro

Para la creación de la interfaz de la aplicación web, se empleó Gradio, un marco de trabajo popular para el desarrollo de aplicaciones web enfocadas a inteligencia artificial y machine learning, lo cual permitirá construir una interfaz interactiva y fácil de usar sin tener que crear un backend aparte ni una conexión API ya que se construye en un solo ambiente (Fig.13) tanto la lógica del modelo, su implementación y la interfaz (Gradio, 2024).



Figura 13 Implementación del modelo con Gradio

En cuanto a la gestión del proyecto, se implementó la metodología Kanban utilizando la herramienta Miro, una plataforma de colaboración visual en línea. Esto permitirá visualizar y rastrear el progreso de las tareas de manera clara y organizada, facilitando la gestión eficiente del proyecto.

Adicional a esto se hizo uso de Visual Studio Code para la elaboración de scripts que permitan cortar las imágenes y sus etiquetas en tamaños específicos haciendo uso del lenguaje Python en varias dimensiones requeridas debió a que la imagen original obtenida tiene una resolución grande de 5742x3648 pixeles.

Finalmente, para evaluar el rendimiento del modelo, se utilizarán métricas proporcionadas por YOLOv8 y se verificará el cumplimiento de la norma ISO 25010, que se encarga de evaluar la calidad del software.

*Tabla 5 Herramientas para el desarrollo*

<b>Herramienta</b>	<b>Función</b>
Dron	Recopilación de imágenes de la plantación de papa
Python	Creación de scripts para la extracción de imágenes
Roboflow	Etiquetado de imágenes
Visual Studio Code	Entorno de desarrollo integrado para la codificación del modelo
Google Colab	Plataforma para el entrenamiento del modelo de aprendizaje profundo
Gradio	Desarrollo de la interfaz de la aplicación web
Miro	Implementación de la metodología Kanban y seguimiento del progreso del proyecto
YOLOv8	Proporcionar métricas de rendimiento para la evaluación del modelo
Norma ISO 25010	Evaluación de la calidad del software

## 2.2. Elaboración del dataset

### 2.2.1. Adquisición de datos

La adquisición de datos para este estudio se llevó a cabo mediante un proceso sistemático y cuidadoso para asegurar que los datos fueran representativos y de alta calidad. Para este fin, se seleccionó varias plantaciones de papas que tuvieran entre 2 semanas de sembrado hasta 5 semanas, ya que

es durante este período que tanto las plantas de papa como las malezas son claramente visibles. Esta elección se hizo para garantizar la visibilidad tanto de plantas de papa como de malezas en las imágenes recogidas (Manuel Pumisacho & Stephen Sherwood, 2002).



*Figura 14 Planta de papa con 30 días de sembrado*

Además, se tuvo especial cuidado en seleccionar los terrenos para que fueran relativamente planos. Esta condición es esencial para asegurar que los píxeles en las imágenes capturadas representen el mismo tamaño en todas las tomas y en todas las áreas de la plantación. Esta precaución permite que los datos de entrenamiento sean consistentes y representativos de las condiciones de campo (Robayo Ordoñez & Santillán, 2022).



*Figura 15 Plantación de papas*

Las plantaciones seleccionadas se encuentran en el cantón Espejo y Montufar. Las imágenes fueron adquiridas entre las 10 am y 2 pm para tener varios tipos de iluminación a 9 m de altura y velocidad de 1 m/s, valores mínimos que son permitidos programar en el software de vuelo controlado Dron Deploy.

A esta altura se obtiene una resolución de 1.0 cm/px

*Fechas en las que se capturaron las imágenes en los terrenos elegidos*



Tabla 6 Información de los terrenos

Terreno	Fecha
Terreno 1 - San Gabriel, comunidad de Chutan Bajo, 77o51'03"W (Propiedad del padre del Ing. Marco Pusdá).	20/05/2023.
	28/05/2023.
	04/06/2023.
Terreno 2 - San Gabriel, 77o48'403"W (Propiedad anónima).	04/06/2023.
Terreno 3 - Cuesaca, Avenida de la Prehistoria, 77o52'40"W (Propiedad del tío del estudiante Kevin Vinueza).	17/06/2023.
Terreno 4 - El Ángel, barrio San Vicente, 0o38'21"N (Propiedad anónima).	16/12/2023.
Terreno 5 - El Ángel, barrio San Vicente, 77o55'59"W (Propiedad anónima).	16/12/2023.
Terreno 6 - El Ángel, barrio San Vicente, 77o55'59"W (Propiedad anónima).	16-12-2023.
Terreno 7 - El Ángel, barrio San Vicente, 0o38'13"N (Propiedad anónima).	16-12-2023.



Figura 16 Planes de vuelo DronDeploy



Figura 17 Fotografía a 9 m de altura

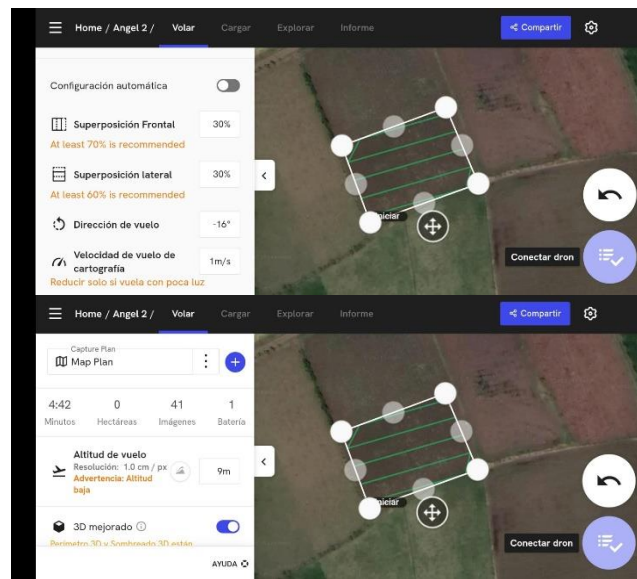


Figura 18 Parámetros de vuelo

Además de la recopilación de imágenes, se tomaron fotografías de las malezas presentes en el terreno para su posterior etiquetación: kikuyo (*Pennisetum clandestinum*), lengua de vaca (*Rumex crispus*) y rábano (*Raphanus raphanistrum*). Las hojas de las plantas de papa también fueron fotografiadas y medidas. Estas fotografías se tomarán en tres ocasiones, con una semana de intervalo entre cada toma, proporcionando datos para analizar el crecimiento y desarrollo de las plantas y las malezas a lo largo del tiempo (Edifarm, 2018).

Planta	Imagen
--------	--------






Lengua de vaca	
Kikuyo	
Diente de Leon	
Papa	
Otras (Agujilla, Berbenita)	

Figura 19 Plantas a etiquetar

### 2.2.2. Preprocesamiento de imágenes

El primer paso en el preprocesamiento de las imágenes fue extraer las imágenes tomados durante los vuelos del dron desde la tarjeta SD. Las imágenes extraídas del dron tienen una resolución superior a 4K (5742x3648). Esta resolución alta es particularmente útil para nuestro propósito ya que, debido al tamaño relativamente pequeño de las plantas y las malezas en comparación con toda la imagen, la alta resolución permite etiquetarlas con precisión. Cabe destacar que la red la red YOLOv8 permite redimensionar automáticamente las imágenes a un tamaño específico dado por el usuario,

dándonos la posibilidad de no realizar ninguna redimensión de la imagen previo. Esto significa que una imagen 4K de 3840x2160 la podemos redimensionar al tamaño que se desee, pero teniendo en cuenta que mientras más pequeña se la haga a la imagen existirá mayor pérdida de datos aunque el tiempo de entrenamiento se reduzca. Este proceso permite que el modelo maneje las imágenes dentro de sus limitaciones de memoria sin distorsionar la relación de aspecto original(Ultralytics, 2023a). Pero en nuestro proyecto se probaron ambas soluciones, sin recortar y recortando la imagen en tamaños de 400x400; 600x600; 800x800; 1080x1080; 2048x2048 y usando redimensión de YOLOv8 se probó en dimensiones de 1024x1024 2048x2048 y 3072x3072.

Imagen	
Id. de imagen	
Dimensiones	5472 x 3648
Ancho	5472 píxeles
Alto	3648 píxeles
Resolución horizontal	72 ppp
Resolución vertical	72 ppp
Profundidad en bits	24
Compresión	
Unidad de resolución	2
Representación del color	sRGB
Bits comprimidos/píxel	

Figura 20 Información imagen original

Tabla 7 Cantidad de imágenes obtenidas.

Terreno	Fecha y hora	Número de imágenes originales
Terreno 1 - San Gabriel, comunidad de Chutan Bajo, 77o51'03"W (Propiedad del padre del Ing. Marco Pusdá).	20/05/2023, 9:10 - 9:55.	539.
	28/05/2023, 10:07 - 10:42.	409.
	04/06/2023, 11:25 - 12:38.	926.
Terreno 2 - San Gabriel, 77o48'403"W (Propiedad anónima).	04/06/2023, 14:12 - 14:20.	197.
Terreno 3 - Cuesaca, Avenida de la Prehistoria, 77o52'40"W (Propiedad del tío del estudiante Kevin Vinueza).	17/06/2023, 8:58 - 9:20.	209.
Terreno 4 - El Ángel, barrio San Vicente, 0o38'21"N (Propiedad anónima).	16/12/2023, 10:23 - 10:28.	61.

Terreno 5 - El Ángel, barrio San Vicente, 77°55'59"W (Propiedad anónima).	16/12/2023, 10:52 - 10:55.	28.
Terreno 6 - El Ángel, barrio San Vicente, 77°55'59"W (Propiedad anónima).	16-12-2023, 11:18 - 11:25.	53.
Terreno 7 - El Ángel, barrio San Vicente, 0°38'13"N (Propiedad anónima).	16-12-2023, 11:51 - 11:55.	25.
Total.		2447.

Para el recorte de imágenes se empleo un script en Python que permita cortar las imágenes en tamaños específicos dados por el usuario haciendo uso de la librería de OpenCV que nos permite realizar una basta variación en imágenes como recortes, redimensión, edición de la iluminación, etc (OpenCV, 2024).

En el proceso de preprocesamiento, adicionalmente se llevo a cabo una limpieza de las imágenes. Se eliminaron las imágenes que no contribuirían eficazmente al entrenamiento del modelo. Estas incluyen imágenes capturadas al inicio y al final del vuelo, cuando el dron estaba despegando o aterrizando, así como aquellas donde la cámara estaba obstruida por árboles, salía de la plantación o las líneas de cultivo no estaban paralelas al vuelo. También se descartaron las imágenes borrosas.



*Figura 21 Obstáculos presentes en la imagen*





Figura 22 Tomas fuera de la plantación



Figura 23 Toma desde altura muy alta



Figura 24 Toma desde altura muy baja

Tabla 8 Banco de imágenes depuradas

<b>Criterio de eliminación</b>	<b>Número de imágenes útiles originales</b>
Eliminación de todas las imágenes adquiridas en el terreno 1 y 3.	364.
Eliminación de aquellas que presentaban desenfoque.	274.
Imágenes finales.	274.

### 2.2.3. Etiquetado de imágenes

Las imágenes adquiridas y preprocesadas fueron etiquetadas de acuerdo

con la variedad y cantidad de malezas presentes en las imágenes para poder obtener datos relevantes junto a las plantas de papa. Este proceso de etiquetado es crucial para el entrenamiento del modelo de red neuronal convolucional, ya que proporciona a la red la información que necesita para aprender a distinguir entre las clases de plantas etiquetadas. (Boronat, 2022; Dang et al., n.d.).

Para el proceso de etiquetado, se eligió utilizar la herramienta Roboflow debido a su versatilidad y portabilidad. Roboflow es una herramienta de anotación de imágenes online que permite la fácil creación y gestión de conjuntos de datos de aprendizaje automático. Su interfaz intuitiva permite a los usuarios seleccionar y clasificar objetos dentro de las imágenes, mientras que su naturaleza online facilita la colaboración entre los equipos y la compatibilidad entre diferentes sistemas y formatos de archivo (Roboflow, 2023).



Figura 25 Uso de herramienta roboflow

La red neuronal convolucional YOLOv8 requiere un formato de etiquetado específico. Cada objeto en la imagen necesita ser encerrado en un cuadro delimitador y etiquetado con su correspondiente clase (en este caso, planta de papa o maleza). Los cuadros delimitadores se definen por las coordenadas (x, y) del centro del cuadro y su anchura y altura, todos normalizados a la unidad respecto a las dimensiones de la imagen.



Figura 26 Etiquetado por bounding box

```

COCO_val2014_000000013992.txt
0 0.558293 0.899820 0.041387 0.069440
0 0.507853 0.907060 0.026053 0.056480
0 0.690680 0.846350 0.011973 0.019940
0 0.705880 0.841850 0.014320 0.024180

```

Figura 27 Formato de etiquetado YOLO

Fuente: (Iñigo Martínez, 2022)

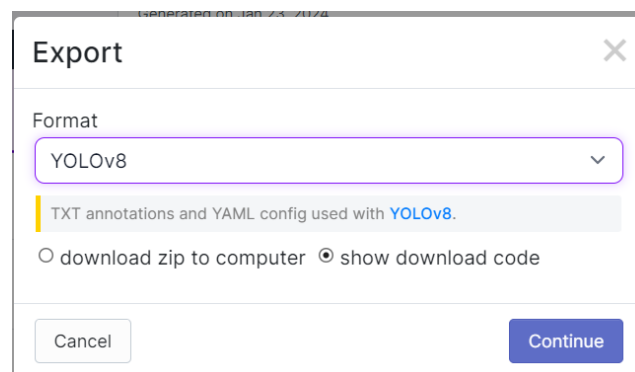


Figura 28 Formato de exportación Roboflow

En cuanto a cómo se almacenan los archivos de etiquetas, cada imagen tiene un archivo de texto correspondiente con el mismo nombre de archivo. Dentro de este archivo de texto, cada objeto en la imagen tiene su propia línea. Cada línea incluye la clase del objeto (un número entero), seguido de las coordenadas x, y, anchura y altura del cuadro delimitador (Reis et al., 2023b).



En resumen, este proceso de etiquetado crea una sólida base de datos de entrenamiento para la red YOLO, permitiéndole aprender a distinguir y localizar con precisión las plantas de papa y las malezas en las imágenes de la plantación.

### 2.3. Elaboración del dataset

Una vez finalizado el proceso de etiquetado del conjunto de datos compuesto por 274 imágenes, cada una con una resolución de 5472x3648 se obtuvo las estadísticas del dataset, permitiéndonos ver a nivel general la cantidad de imágenes, etiquetas y número de clases etiquetadas.

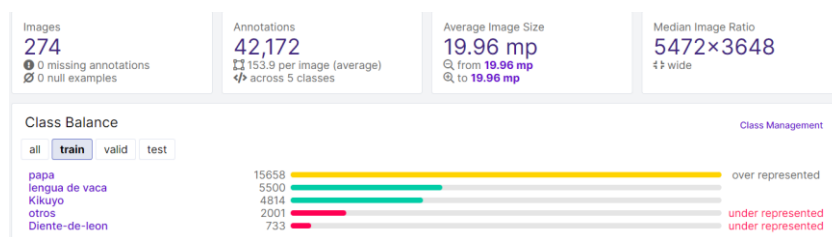


Figura 29 Estadísticas dataset

Para evitar pérdida de información al momento de recortar las imágenes usando la herramienta de Roboflow debió a que está comprime la imagen hasta un 40%.

### 2.4. Implementación y Evaluación del Modelo

Para la implementación del modelo, se optó por utilizar Google Colab como plataforma de entrenamiento. Google Colab ofrece recursos computacionales potentes y gratuitos, como GPUs, que son esenciales para el entrenamiento eficiente de modelos de red neuronal convolucional. El dataset de imágenes y sus etiquetas, así como el modelo entrenado, los resultados, las imágenes de prueba y las métricas proporcionadas por YOLOv8, se almacenarán en una carpeta de Google Drive. Esta elección permite un fácil acceso y gestión de todos los archivos y datos necesarios para el proyecto.

El dataset se dividirá en un 80% para el entrenamiento y un 20% para la validación del modelo. Estos porcentajes son comunes en aprendizaje automático, ya que proporcionan un equilibrio entre tener suficientes datos para entrenar el modelo y asegurar que hay una cantidad significativa de datos que el modelo no ha visto antes para su validación. Durante el entrenamiento, se

variarán las épocas y otros hiperparámetros, como el tamaño de redimensionamiento y el batch, para encontrar la configuración más efectiva. Finalmente, el modelo se probará con imágenes fuera del dataset de entrenamiento para verificar su eficacia y generalización a nuevas imágenes de plantaciones de papa.

#### **2.4.1. Establecimiento del entorno de entrenamiento**

El establecimiento de un entorno de entrenamiento adecuado es fundamental para el desarrollo y el rendimiento exitoso de cualquier modelo de aprendizaje automático. En este caso, se decidió utilizar Google Colab como plataforma de entrenamiento debido a su capacidad para proporcionar recursos computacionales potentes y gratuitos, incluyendo el uso de GPUs (Alnajjar, 2021). Estos recursos son esenciales para el entrenamiento eficiente de modelos de redes neuronales convolucionales como YOLOv8 (Aiman et al., 2023).

Además, Google Drive se utiliza en conjunto con Google Colab para almacenar y administrar todos los archivos y datos necesarios para el proyecto. En particular, se crea una estructura de directorios específica dentro de Google Drive. Esta estructura incluye directorios separados para las imágenes de entrenamiento y validación, así como para los archivos de texto que contienen las etiquetas correspondientes. Adicional a esto, se establece una carpeta dedicada para almacenar las imágenes que se utilizarán en las pruebas posteriores.

Una vez que se ha establecido el entorno de trabajo y la estructura de directorios, se realiza la instalación de la biblioteca Ultralytics en el entorno de Google Colab. Esta biblioteca es fundamental ya que proporciona la implementación de YOLOv8 que se utilizará para el entrenamiento del modelo (Ultralytics, 2023a).

Tras la configuración de la biblioteca Ultralytics, se procede a transferir las imágenes del dataset, que se encuentran almacenadas en Google Drive, al entorno de Colab. En este proceso, se realiza una división de los datos en dos subconjuntos: un 80% de las imágenes se destinan al entrenamiento y el 20% restante se utiliza para la validación del modelo. Estos subconjuntos se copian

en sus respectivos directorios de acuerdo con la estructura de directorios establecida previamente(Orosco & Luna, 2021).

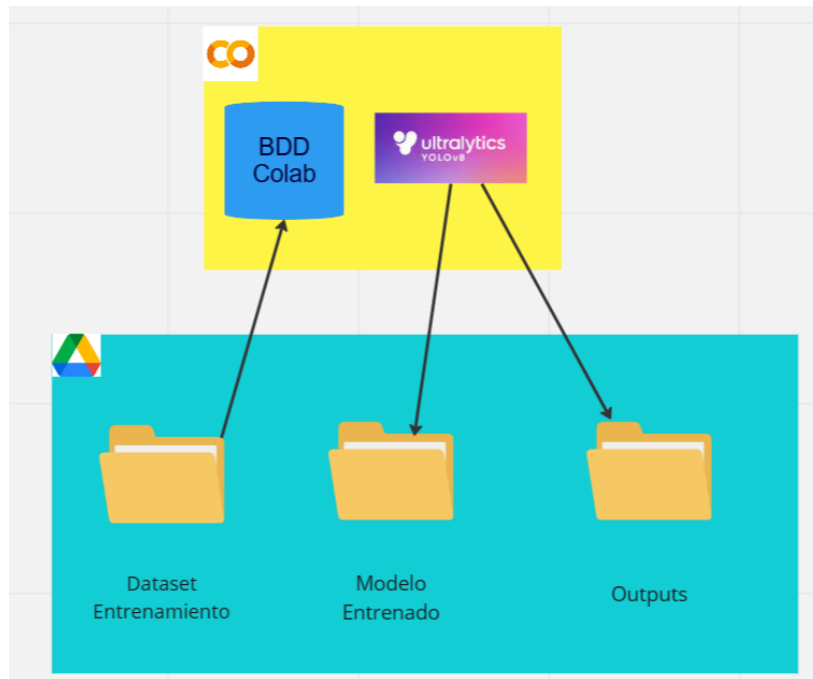


Figura 30 Estructura entorno de entrenamiento

El último paso en la configuración del entorno de entrenamiento es la creación de un archivo YAML. Este archivo es crucial para el proceso de entrenamiento ya que YOLOv8 lo utiliza para localizar el dataset y sus respectivas etiquetas.

Respecto a los recursos informáticos, se utilizan los siguientes:

Recurso	Versión
<b>Ultralytics YOLO</b>	v8.0.109
<b>Python</b>	3.10.11
<b>PyTorch</b>	2.0.1+cu118
<b>GPU</b>	A100 40 GB
<b>RAM</b>	80 GB

Tabla 9 Recursos en Google Colab con Colab Pro

Esta configuración de recursos y la cuidadosa organización del entorno de entrenamiento aseguran que el proceso de entrenamiento pueda llevarse a cabo

de manera eficiente y ordenada, lo cual es esencial para el desarrollo exitoso del modelo de detección de malezas.

#### 2.4.2. Configuración de hiperparámetros

En el proceso de entrenamiento del modelo YOLOv8 para la detección de objetos, se ha enfocado en una selección cuidadosa de hiperparámetros para optimizar el rendimiento y la precisión. A pesar de que YOLOv8 ofrece una amplia gama de hiperparámetros ajustables, se ha decidido concentrar los esfuerzos en los más influyentes, incluyendo algunos adicionales para mejorar aún más la eficiencia del modelo.

1. **Épocas (epochs):** Se ajustará el número de épocas para equilibrar el aprendizaje profundo y evitar el sobreajuste.
2. **Tamaño del Lote (batch):** Se configurará el tamaño de lote adecuado, clave para la eficiencia computacional y la actualización efectiva de los parámetros.
3. **Tamaño de la Imagen (imgsz):** Se determinará un tamaño óptimo para la detección precisa en diversas escalas.
4. **Modelo Preentrenado (pretrained):** Se usará un modelo preentrenado para aprovechar conocimientos previos y mejorar la precisión.
5. **Optimizador:** Se seleccionará el optimizador más adecuado para una convergencia eficaz y rápida.
6. **Tasa de Aprendizaje Inicial (lr0):** Se establecerá cuidadosamente para permitir ajustes precisos en los pesos del modelo.

Además, se incorporarán tres hiperparámetros adicionales para fortalecer el entrenamiento:

7. **Decaimiento de Peso (weight\_decay):** Este parámetro ayudará a regularizar el modelo, contribuyendo a prevenir el sobreajuste.
8. **Precisión Mixta Automática (amp):** Se empleará para acelerar el entrenamiento, manteniendo la precisión mediante la utilización de precisión mixta.
9. **Validación (val):** Fundamental para evaluar el rendimiento del modelo en datos no vistos y asegurar una buena generalización.

Estos hiperparámetros han sido cuidadosamente seleccionados para

aprovechar al máximo las capacidades de YOLOv8, asegurando un modelo altamente eficiente y preciso en la detección de objetos(Jocher Glenn, 2024).

### 2.4.3. Entrenamiento del modelo

En el entrenamiento del modelo YOLOv8 para detección de objetos, se realizó una serie de experimentos modificando diversos hiperparámetros. Estos ajustes tuvieron impactos significativos en el rendimiento y los tiempos de entrenamiento del modelo. Las variaciones en los hiperparámetros clave, como el número de épocas, tamaño del lote, tamaño de la imagen, tasa de aprendizaje, y el uso de modelos pre entrenados, mostraron cómo cada uno de estos factores puede influir en la precisión y eficiencia del modelo. La experimentación cuidadosa ayudó a encontrar el equilibrio adecuado para lograr un rendimiento óptimo del modelo YOLOv8(Vincent & Jidesh, 2023).

Tabla 10 Hiperparámetros usados

Hiperparámetro	Descripción	Impacto en el Entrenamiento
Número de Épocas	Número de veces que el modelo procesa el conjunto de datos entero	Afecta la precisión y el tiempo
Tamaño del Lote	Número de muestras procesadas antes de actualizar el modelo	Incide en la memoria y velocidad
Tamaño de la Imagen	Dimensiones de las imágenes de entrada	Influye en la precisión y el tiempo
Tasa de Aprendizaje	Velocidad a la que el modelo aprende	Clave para la convergencia y precisión
Modelo Pre entrenado	Uso de un modelo ya entrenado con datos similares	Reduce el tiempo para alcanzar precisión
Optimizador	Algoritmo para actualizar los pesos del modelo	Afecta la velocidad y estabilidad

## 2.5. Desarrollo de la aplicación Web

El desarrollo e integración web son partes cruciales de esta tesis, ya que proporcionan una interfaz fácil de usar para los usuarios y permiten que la solución de detección de malezas sea accesible desde cualquier ubicación. Para esto, se desarrolló una aplicación web usando la herramienta Gradio (Gradio, 2024) permitiendo que los usuarios carguen las imágenes tomadas por el dron y obteniendo la cantidad de afectación de malezas que existe en el terreno. La aplicación web se diseñó de manera intuitiva y amigable para el usuario, con funciones claras y una navegación sencilla. Fig. 31.

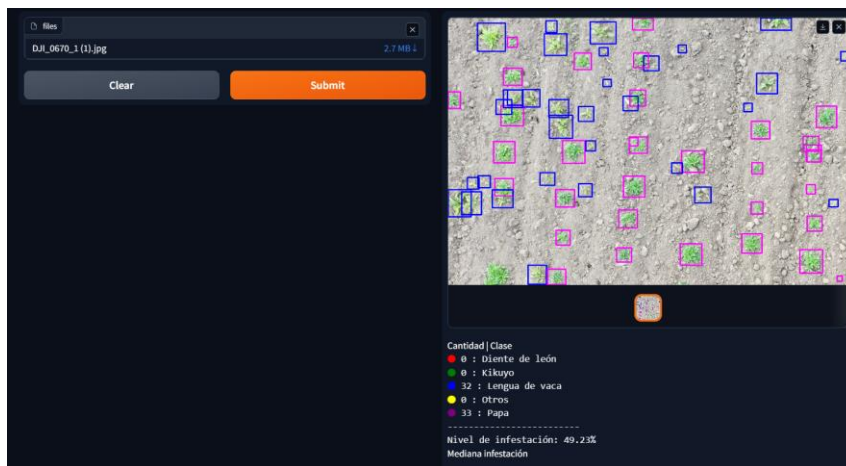


Figura 31 Aplicación en Gradio

Esta aplicación se aloja en un entorno de Google Colab (Google, 2024) y permitiendo crear un servicio web para un acceso sencillo la cual se conecta automáticamente con la conexión api que se crea en el entorno. Fig.32

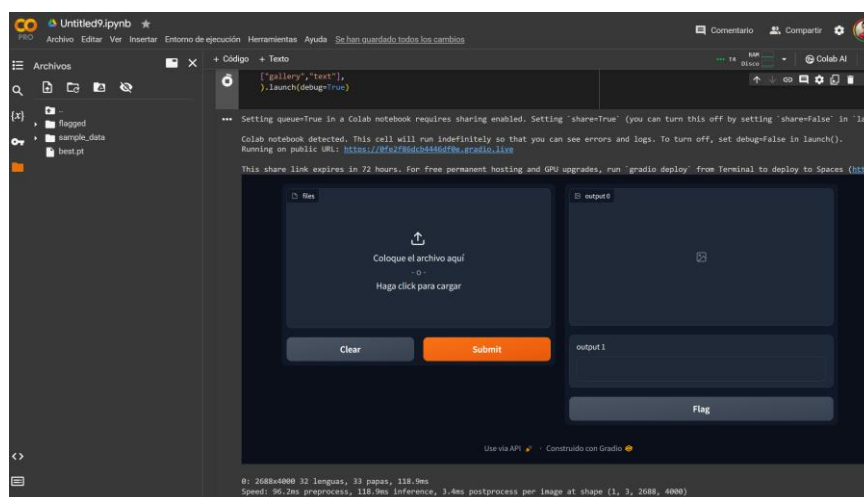
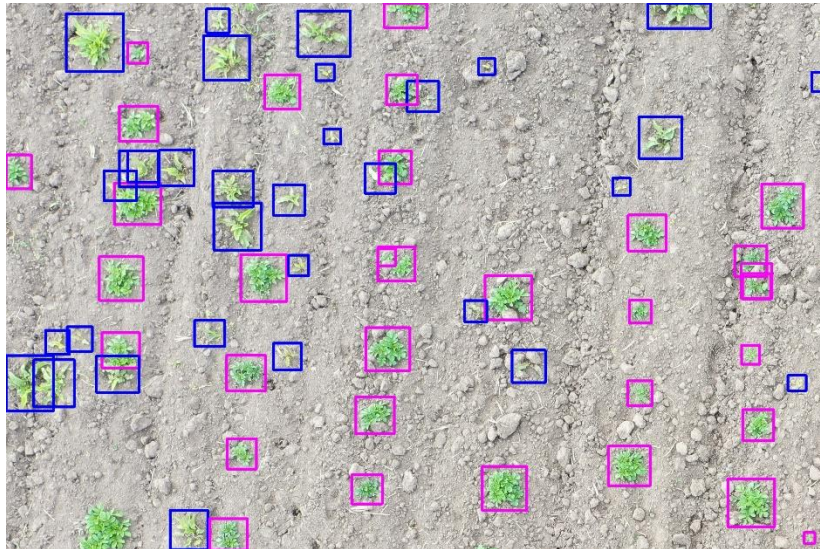


Figura 32 Alojamiento en Google Colab



*Figura 33 Inferencia en imagen*

### **2.5.1. Pruebas y cumplimiento de la norma ISO 25010**

Se llevaron a cabo pruebas exhaustivas para evaluar la capacidad de detección de nuestro sistema y garantizar que pueda identificar de manera precisa y confiable las malezas en las imágenes proporcionadas por los usuarios. Estas pruebas nos permitieron validar la efectividad del modelo YOLOv8 utilizado y asegurarnos de que ofrece resultados de calidad para los usuarios.

Además, se realizó un análisis de eficiencia para evaluar el cumplimiento de la norma ISO 25010 en este aspecto específico. En esta evaluación, nos centramos en la relación entre el rendimiento del sistema y los recursos utilizados en diferentes condiciones. Se realizaron mediciones de tiempo de respuesta, consumo de memoria y otros factores relevantes para evaluar la eficiencia del sistema (ISO, 2005).

Se implementaron medidas para optimizar la eficiencia del sistema, asegurando que el procesamiento de imágenes y la generación de resultados se realicen de manera rápida y eficiente. Esto se logró minimizando el uso de recursos de computación necesarios y mejorando la capacidad de respuesta del sistema. Al cumplir con los criterios de eficiencia establecidos en la norma ISO 25010, buscamos brindar a los usuarios una experiencia eficiente y maximizar el rendimiento del sistema en términos de tiempo y recursos utilizados.

## CAPÍTULO 3

### Resultados y validaciones

Los resultados y graficas presentados en este capítulo fueron obtenidos mediante la herramienta Tensorboard que viene incluida dentro de la biblioteca de Ultralytics que es la que nos permite importar el modelo YOLOv8, presentándonos los resultados de cada entrenamiento junto a la mejor y última época del entrenamiento.

#### 3.1. Métricas del modelo YOLOv8

##### 3.1.1. Línea de tiempo del modelo entrenado

A lo largo de este proceso, se realizaron varios entrenamientos del modelo c, ajustando cuidadosamente los hiperparámetros en cada etapa se obtuvieron diversos resultados (Tabla 8). Este enfoque permitió la creación de un modelo inicial y un final, sobre el cual se realizaron las comparaciones. De todas las versiones del modelo entrenado, se seleccionó el más eficiente en función de su rendimiento y precisión.

*Tabla 11 Resultado de los experimentos variando los hiperparámetros*

Version del modelo	Precisión	Recall	Map	Hiperparámetros cambiados				Tiempo(s)	Recursos	
				Tamaño de imagen	Batchsize	Epocas	Optimizador		GPU	RAM
1	0,074	0,07	0,15	400x400	16	5	Adam	500	11%	19%
2	0,469	0,07	0,16	400x400	20	5	Adam	309	60%	
3	0,812	0,11	0,17	400x400	20	10	SGD	443	60%	
4	0,815	0,14	0,19	400x400	20	15	Adam	637	62%	
5	0,618	0,17	0,20	600x600	20	10	Adam	550	64%	
6	0,627	0,21	0,21	600x600	20	15	Adam	700	65%	
7	0,471	0,23	0,22	600x600	20	20	Adam	850	80%	50%
8	0,451	0,18	0,21	800x800	20	10	Adam	350	95%	70%
9	0,461	0,26	0,26	800x800	20	15	Adam	739	95%	70%
10	0,32	0,29	0,29	800x800	20	20	Adam		95%	70%
11	0,31	0,24	0,24	1080x1080	20	10	Adam	3 min	15 GB	18 GB
12	0,26	0,26	0,26	1080x1080	20	15	Adam	2 min	15 GB	18 GB
13	0,31	0,29	0,30	1080x1080	20	20	Adam	3 min	15 GB	18 GB
14	-----	-----	-----	2048x2048	20	10	Adam	-----	Sobrepaso	Sobrepaso
15	-----	-----	-----	2048x2048	10	10	Adam	-----	Sobrepaso	Sobrepaso
16	0,53	0,33	0,34	2048x2047	5	10	Adam	6 min	27 GB	36 GB
17	0,55	0,33	0,34	2048x2048	5	15	Adam	7 min	27 GB	36 GB
18	0,56	0,33	0,36	2048x2048	5	20	Adam	8 min	27 GB	36 GB
19	-----	-----	-----	2048x2048	5	10	Adam	-----	40 GB	39 GB
20	0,58	0,32	0,34	2048x2050	1	10	Adam	-----	70 GB	39 GB
21	0,56	0,33	0,33	2048x2051	1	15	Adam	11 min	70 GB	39 GB
22	-----	-----	-----	2048x2052	1	20	Adam	-----	Sobrepaso	Sobrepaso
23	0,39	0,44	0,39	5220x5220	1	10	Adam	25 min	75	40 GB
24	0,42	0,47	0,41	5220x5220	1	15	Adam	38 min	75	40 GB
25	0,51	0,5	0,48	5220x5220	1	30	Adam	50 min	75	40 GB

En este apartado, se analizaron las métricas clave utilizadas para evaluar el rendimiento del modelo de detección de malezas en plantaciones de papa a través del uso de drones. Estas métricas se dividieron en tres categorías: entrenamiento, validación y resultados finales, incluyendo precisión, recall y mAP.



## Fase de Entrenamiento

En la fase de entrenamiento, se analizaron meticulosamente las métricas de pérdida del modelo. Estas incluyeron:

- Pérdida de caja, que muestra qué tan bien el modelo predice la ubicación y el tamaño de las cajas delimitadoras de las malezas.
- Pérdida de clasificación, que indica la precisión con la que el modelo distingue las malezas de otros objetos.
- Pérdida de distribución de puntos de anclaje, que refleja la precisión en la predicción de puntos específicos dentro de las cajas delimitadoras.

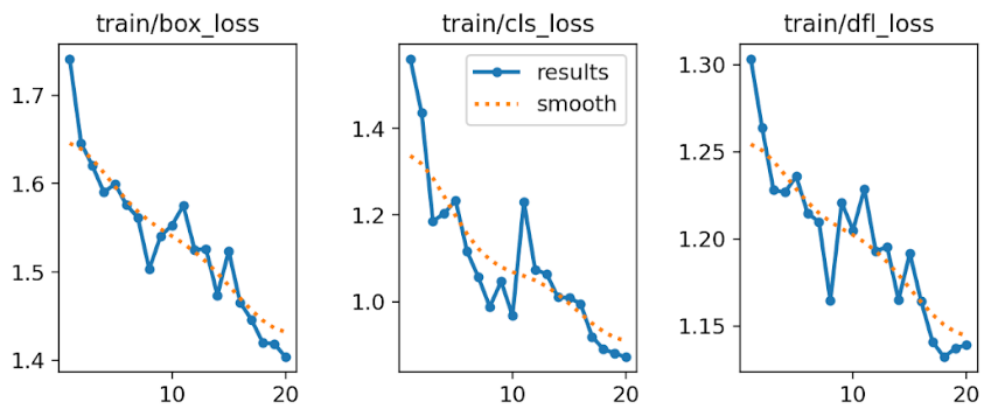


Figura 34 Métricas en entrenamiento

Estas métricas fueron fundamentales para ajustar y calibrar el modelo, enfocándose en la precisión tanto en la localización como en la identificación de las malezas(Wang et al., 2023).

## Fase de Validación

Durante la fase de validación, se evaluó el modelo utilizando las mismas métricas de pérdida pero en un conjunto de datos no utilizado en el entrenamiento. Este proceso fue esencial para comprobar si el modelo podía generalizar su aprendizaje a nuevas situaciones, un aspecto clave para evitar el sobreajuste. Es decir, se aseguró que el modelo no solo memorizara los datos de entrenamiento, sino que también fuera capaz de aplicar su aprendizaje a nuevas imágenes de malezas(Niu et al., 2023).

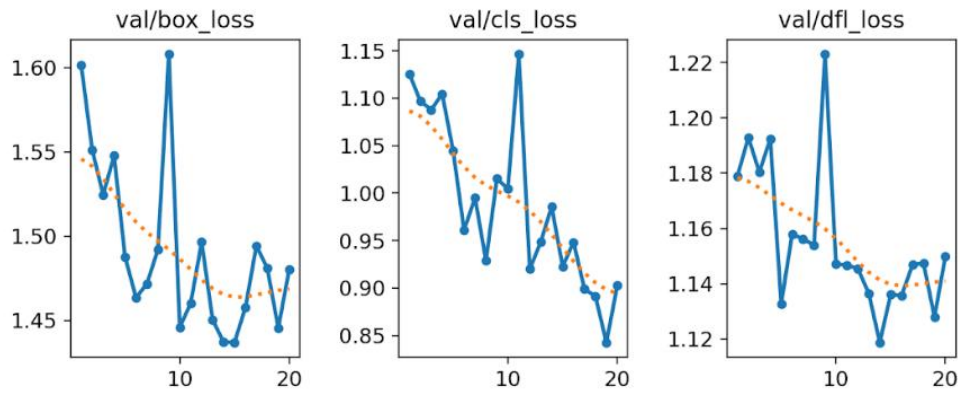


Figura 35 Métricas en validación

### Fase de Resultados

En la fase de resultados, se enfocó en métricas clave como la precisión, el recall, el mAP, matriz de confusión y la curva F1-Confidence.

La matriz de confusión es una herramienta poderosa para visualizar el rendimiento del modelo, mostrando la cantidad de verdaderos positivos, falsos positivos, verdaderos negativos y falsos negativos. Esta matriz ayuda a entender mejor en qué situaciones el modelo tiene éxito y en cuáles falla el modelo.

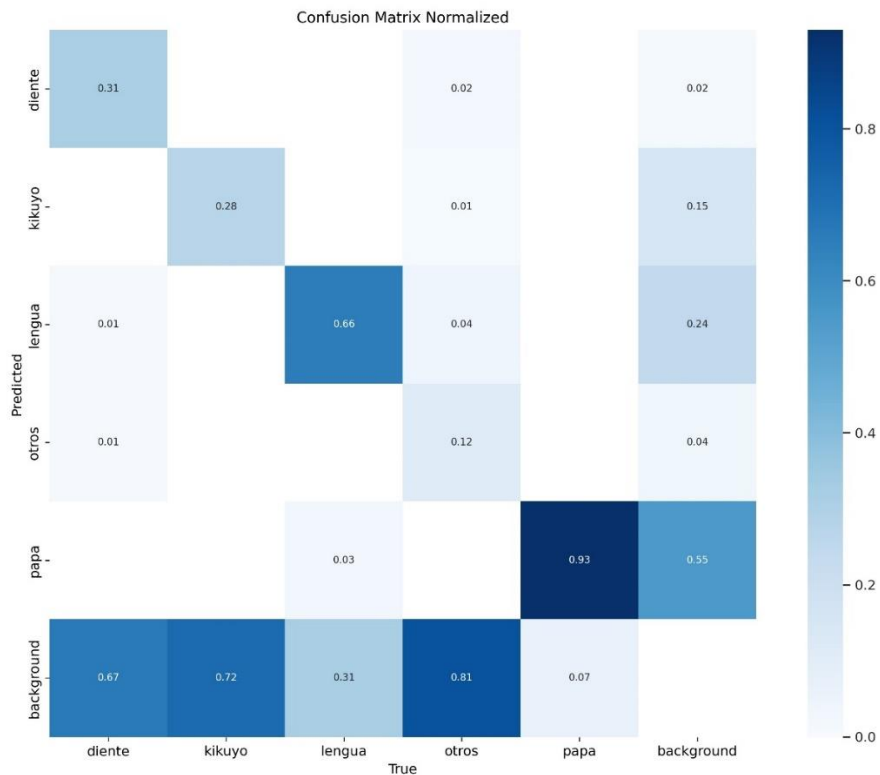


Figura 36 Matriz de confusión

La precisión mostró el porcentaje de detecciones de malezas que fueron correctas, mientras que el recall reflejó qué proporción de todas las malezas presentes fue detectada por el modelo.

### Precisión

Una alta precisión significa que el modelo tiene un buen desempeño en identificar correctamente las malezas, minimizando la identificación incorrecta de objetos que no son malezas.

$$\text{Precisión} = \frac{\text{Verdaderos Positivos (VP)}}{\text{Verdaderos Positivos (VP)} + \text{Falsos Positivos (FP)}}$$

En esta ecuación:

- **Verdaderos Positivos (VP)** se refiere a los casos donde el modelo identifica correctamente la clase objetivo (por ejemplo, malezas).
- **Falsos Positivos (FP)** son situaciones donde el modelo identifica erróneamente la clase objetivo en objetos que no pertenecen a ella.

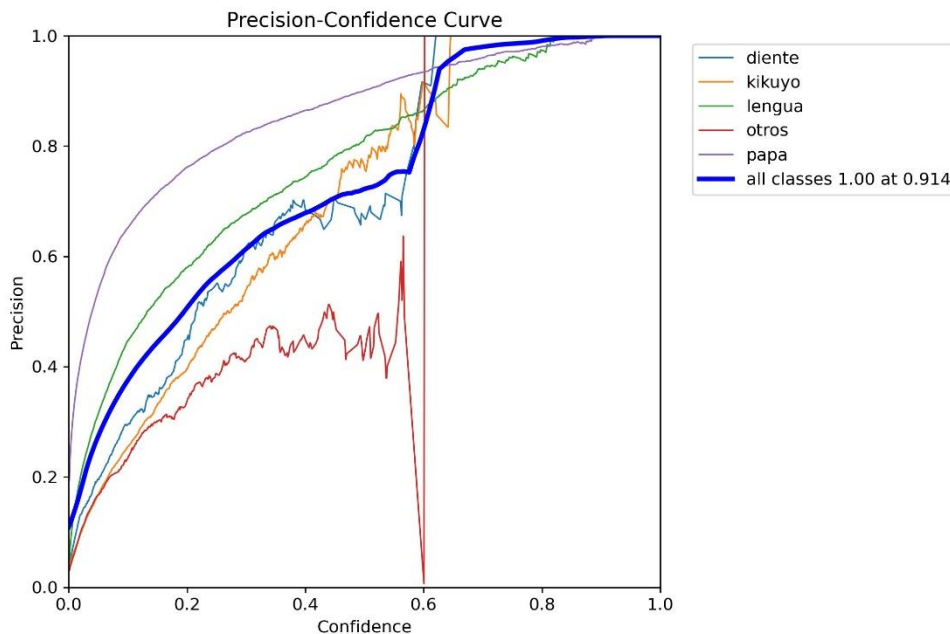


Figura 37 Grafica Precisión vs Confianza

### Recall

Un alto valor de recall implicaría que el modelo es efectivo en identificar la mayoría de las malezas presentes, minimizando el riesgo de pasarlas por alto.

$$\text{Recall} = \frac{\text{Verdaderos Positivos (VP)}}{\text{Verdaderos Positivos (VP)} + \text{Falsos Negativos (FN)}}$$

En esta fórmula:

- **Verdaderos Positivos (VP)** son los casos en que el modelo identifica correctamente la clase de interés (por ejemplo, malezas en una plantación de papa).
- **Falsos Negativos (FN)** son los casos en que el modelo no identifica la clase de interés, a pesar de que realmente está presente en la imagen.

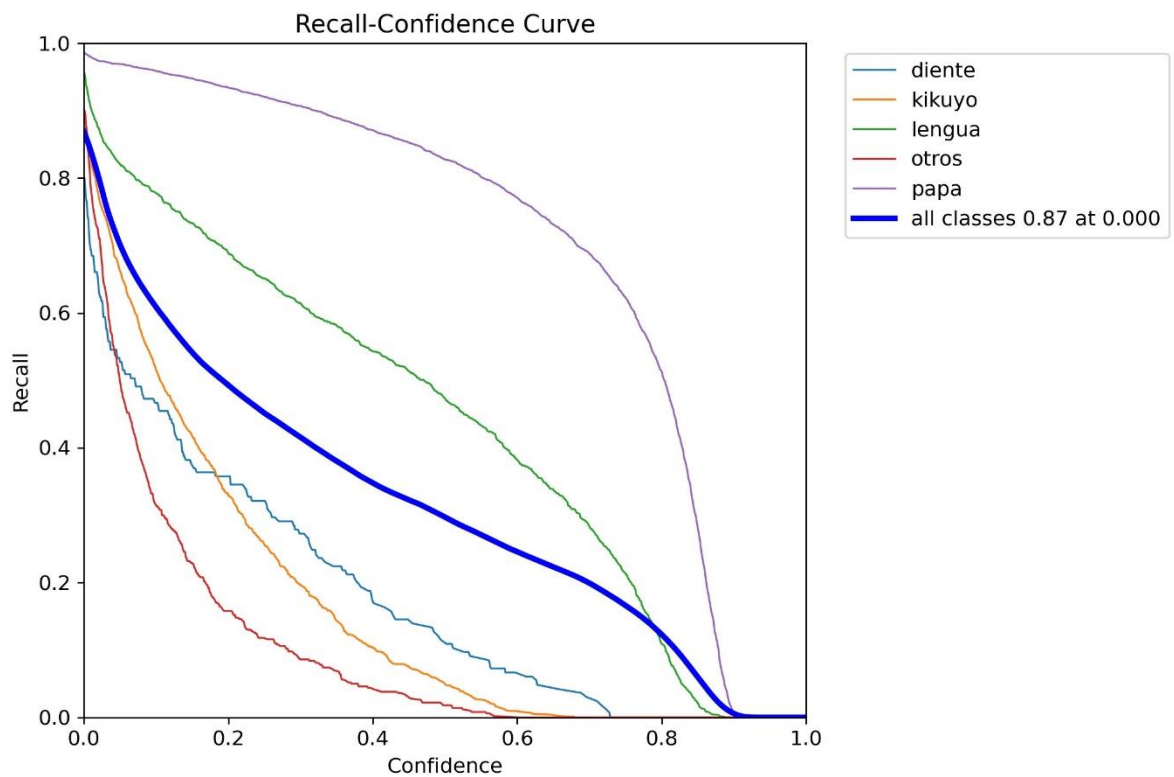


Figura 38 Grafica Recall vs Confianza

Ambas métricas son fundamentales para entender la efectividad del modelo en la identificación correcta de malezas y en minimizar los falsos positivos. Por último, el mAP proporcionó una evaluación exhaustiva al combinar la precisión y el recall en varios umbrales de confianza, dando una visión integral del rendimiento del modelo bajo diferentes condiciones. Esta métrica fue vital para confirmar la aplicabilidad del modelo en la detección práctica de malezas en

ambientes agrícolas reales (Zhu et al., 2023).

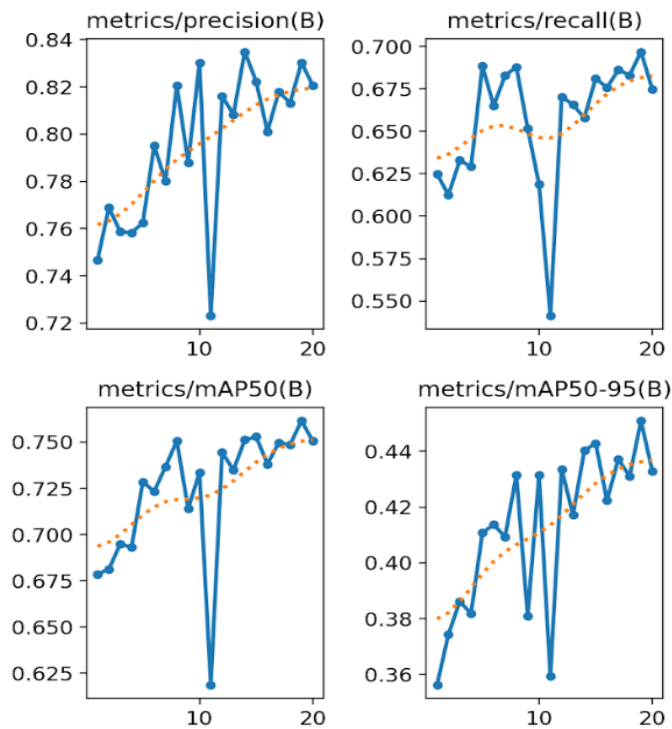


Figura 39 Resultados del modelo

Una métrica importante en esta fase fue la inclusión de la curva F1-Confidence, que representó cómo la Puntuación F1 del modelo varió con diferentes niveles de confianza. Esta curva es esencial para encontrar el equilibrio óptimo entre precisión y recall, asegurando que el modelo no solo detecte la mayoría de las malezas (alto recall) sino que también haga estas detecciones de manera precisa (alta precisión). La adecuada calibración de esta curva es vital para la aplicación práctica del modelo en la detección de malezas en entornos agrícolas (Demirel et al., 2023).

La fórmula del F1 Score es:

$$F1 \text{ Score} = 2 \times \frac{\text{Precisión} \times \text{Recall}}{\text{Precisión} + \text{Recall}}$$

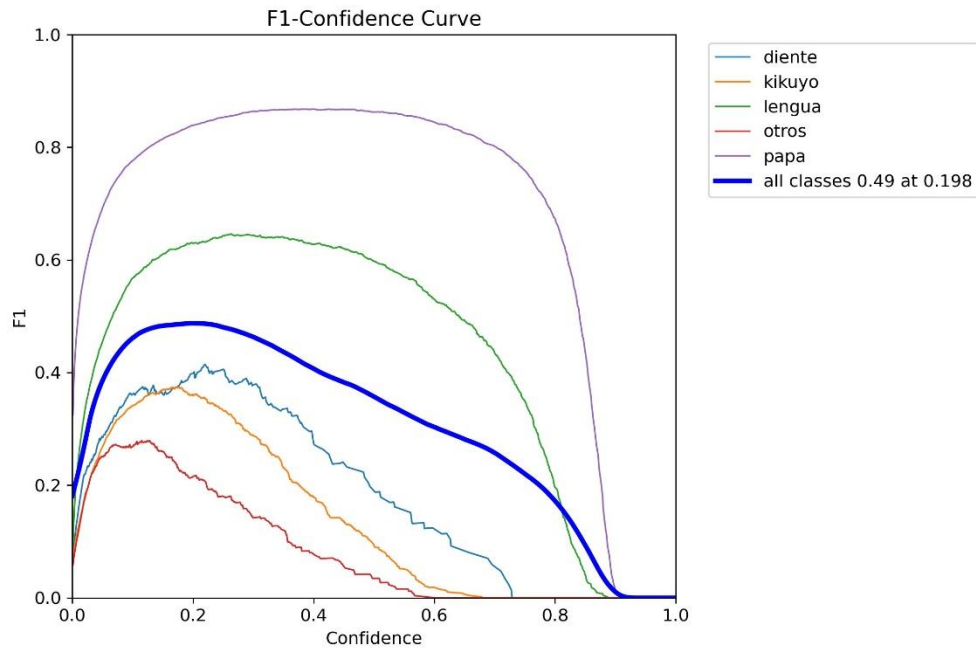


Figura 40 Curva F1 - Confidence

Para finalizar, es crucial distinguir entre la Puntuación F1 y el mAP, ya que ambas ofrecen perspectivas diferentes del rendimiento del modelo. Mientras la Puntuación F1 combina precisión y recall en un solo valor, enfocándose en el equilibrio entre estos dos aspectos para una clase o conjunto de clases, el mAP evalúa el rendimiento del modelo a través de varios umbrales y clases, siendo más adecuado para escenarios complejos de detección de objetos. La Puntuación F1 es ideal para situaciones con clases desbalanceadas, proporcionando una medida resistente a desequilibrios. Por otro lado, el mAP ofrece una visión detallada y exhaustiva, crucial para entender cómo el modelo se comporta en la detección de múltiples objetos o clases en una misma imagen.

Se ha considerado el mAP como medida de comparación con los modelos de trabajos similares en el campo de la detección con inteligencia artificial ya que ésta nos permite saber que tan bueno es nuestro modelo al momento de detectar la mayor cantidad de malezas o planta. Fig.41. y una vez que se encontró una planta, saber con certeza que clase es la encontrada, ya que esta medida logra combinar el recall y la precisión.

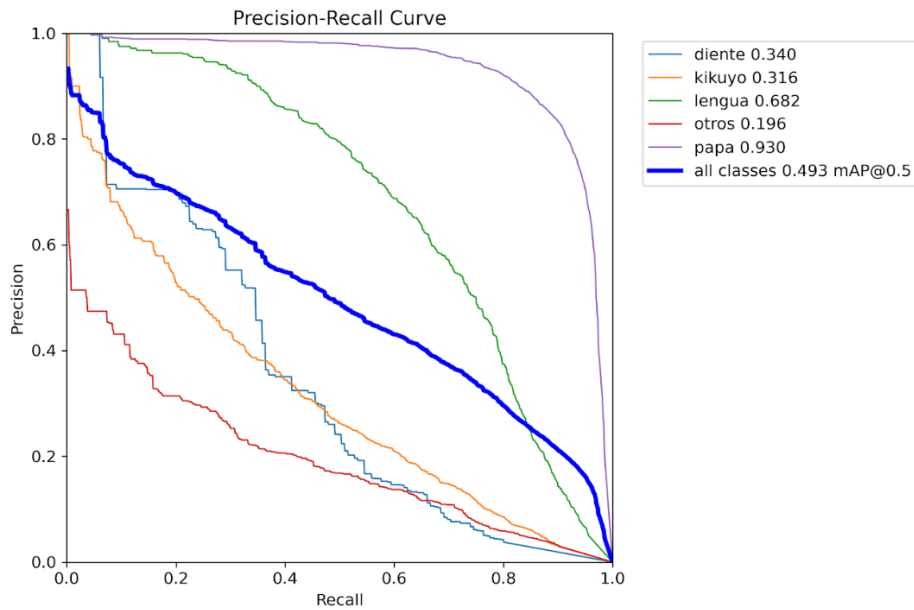


Figura 41 Grafica de mAP ( mean Average Precision) de las 5 clases detectadas.

```

Model summary (fused): 168 layers, 3006623 parameters, 0 gradients, 8.1 GFLOPs
  Class  Images  Instances  Box(P)  R    mAP50  mAP50-95) :
  all    493      7758      0.5     0.496 0.493  0.281
  diente 493      165       0.447  0.358 0.34  0.169
  kikuyo 493      1160      0.391  0.334 0.316 0.13
  lengua 493      1340      0.577  0.694 0.682 0.391
  otros  493      500       0.327  0.158 0.196 0.1
  papa   493      4593      0.759  0.935 0.93  0.615
    
```

Figura 42 Metricas en Validación

### 3.2. Comparación del modelo con métodos tradicionales

En este apartado se realiza una comparación detallada entre el método tradicional empleado por los agricultores para la detección de malezas en plantaciones de papa y el método propuesto que utiliza drones y el modelo de detección YOLOv8 alojado en una aplicación web.

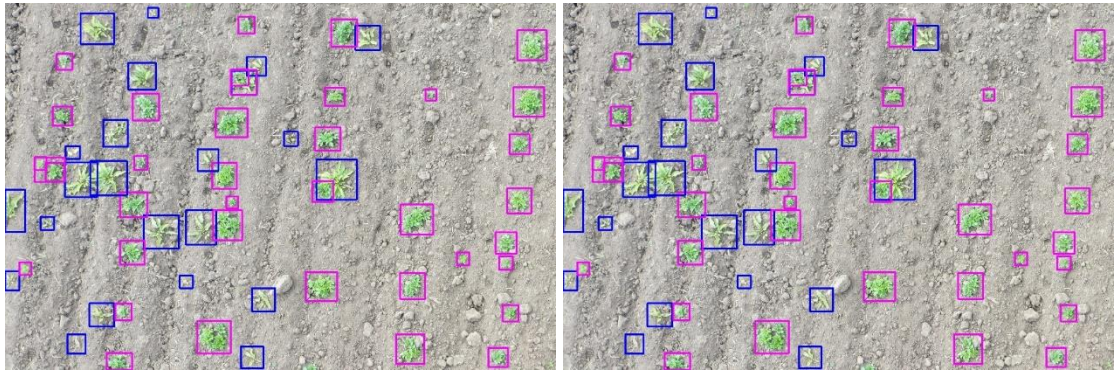
#### 3.2.1. Inferencias y resultados

Con el modelo entrenado y desplegado en la aplicación web se procedió a realizar inferencias en algunas imágenes del dataset en la cual se dibujo los recuadros limitadores correspondientes a las etiquetas de dicha imagen y se comparo con el la inferencia realizada por la aplicación web.

Prueba 1

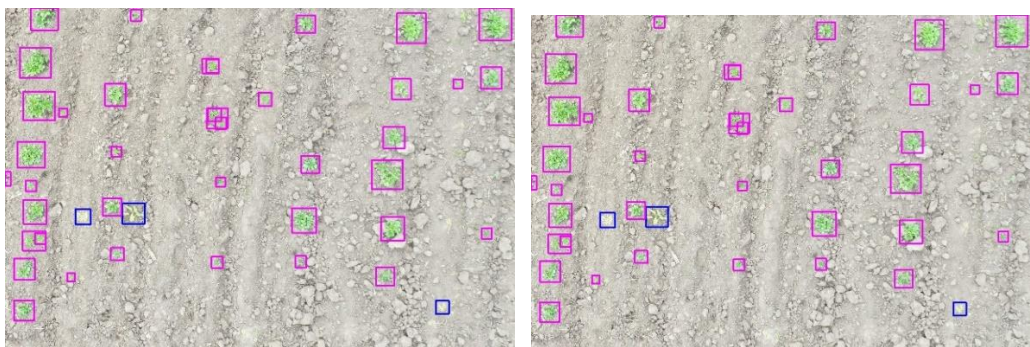
Imágenes etiquetadas	Imágenes con inferencia (Etiquetas correctas)	Resultado Modelo vs Imágenes Etiquetadas

Lenguas de vaca: 11 Papa: 22 Plantas totales: 33	Lenguas de vaca: 22 Papa: 35 Plantas totales: 57	24 etiquetas adicionales correctas
--	--	--



### Prueba 2

<b>Imágenes etiquetadas</b>	<b>Imágenes con inferencia (Etiquetas correctas)</b>	<b>Resultado Modelo vs Imágenes Etiquetadas</b>
Lenguas de vaca: 1 Papa: 17 Plantas totales: 18	Lenguas de vaca: 3 Papa: 29 Plantas totales: 32	24 etiquetas correctas más de las etiquetas





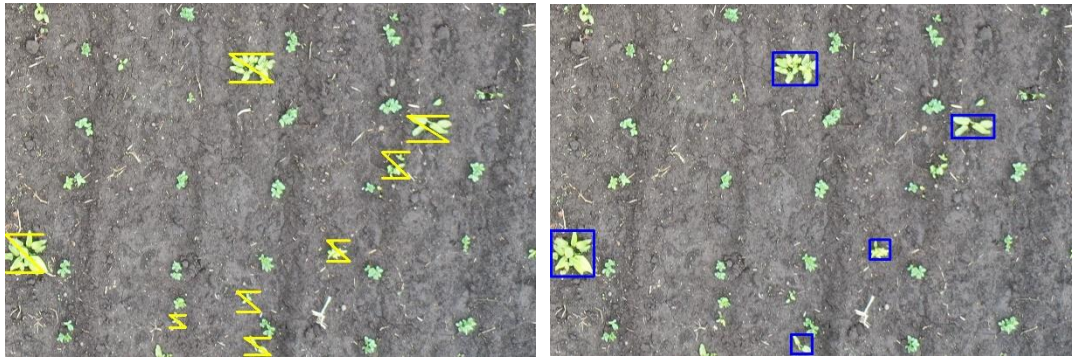
Prueba 3

<b>Imágenes etiquetadas</b>	<b>Imágenes con inferencia</b>	<b>Resultado Modelo vs Imágenes Etiquetadas</b>
Lenguas de vaca: 7 Kikuyo: 2 Plantas totales: 9	Lenguas de vaca: 6 Kikuyo: 1 Plantas totales: 7	2 etiquetas menos que las predichas



Prueba 4

<b>Imágenes etiquetadas</b>	<b>Imágenes con inferencia</b>	<b>Resultado Modelo vs Imágenes Etiquetadas</b>
Lenguas de vaca: 8 Plantas totales: 8	Lenguas de vaca: 5 Plantas totales: 5	3 etiquetas menos que las predichas



*Figura 43 Comparación de imágenes etiquetadas*

Con estas pruebas y comparaciones podemos observar que en ciertas plantaciones en donde existen las variedades de papa y lengua de vaca, la eficiencia del modelo incrementa y donde existen kikuyo, y diente de león la precisión baja aunque de manera leve en comparación al incremento.

### **3.2.2. Métodos tradicionales**

Estos métodos consisten en caminar entre las líneas del cultivo y realizar una estimación visual de la presencia de malezas. Este proceso, aunque intuitivo y basado en la experiencia, puede ser subjetivo y varía significativamente en precisión y eficiencia, dependiendo del observador. Además, se destaca el tiempo considerable que requiere este método, así como su posible impacto físico en los agricultores debido a la necesidad de recorrer extensas áreas a pie.

### **3.2.3. Uso del modelo de detección de malezas con Dron**

Por otro lado, el uso del modelo y el dron inicia con una planificación y ejecución del vuelo del dron, que permite capturar imágenes aéreas de las plantaciones. Posteriormente, estas imágenes se cargan en una aplicación web que aloja el modelo YOLOv8 para la detección y conteo de malezas. Este enfoque se caracteriza por su capacidad para cubrir grandes áreas en un tiempo reducido, ofreciendo una visión más amplia y detallada del estado de las malezas en la plantación.

### **3.2.4. Comparativa**

Se llevó a cabo una comparativa entre el método tradicional de detección de malezas en plantaciones de papa, que consiste en la inspección visual directa por parte de agricultores, y un enfoque moderno que integra el uso de drones y

el modelo YOLOv8. En términos de eficiencia temporal, el método tradicional mostró limitaciones significativas, especialmente para grandes áreas de cultivo, debido a la necesidad de recorrer físicamente el terreno, lo cual resulta en un proceso más lento y laborioso. Por otro lado, el método que emplea drones y YOLOv8 destacó por su rapidez y eficacia en la recolección y análisis de datos, permitiendo cubrir amplias áreas en un tiempo reducido y ofreciendo resultados de forma casi instantánea tras la captura de imágenes.

Cabe recalcar que, en el estudio realizado, se midió el tiempo que un agricultor requiere para evaluar la infestación de malezas en un terreno de cultivo de papa. Los resultados indicaron que, en promedio, el agricultor invertía aproximadamente 5 segundos para analizar cada metro cuadrado del terreno. Este dato proporciona una referencia importante para comparar la eficiencia de métodos tradicionales con técnicas automatizadas, como el uso de drones y modelos de visión por computadora, en la detección y gestión de malezas (Tabla 9).

*Tabla 12 Tiempos de detección de malezas*

<b>Identificación del Terreno</b>	<b>Extensión (m<sup>2</sup>)</b>	<b>Método</b>	<b>Tiempo Total</b>	<b>Diferencia</b>
Terreno 1	800	Tradicional (Visual)	66:00 min	53:12 min
		Dron + YOLOv8	17:12 min	
Terreno 2	650	Tradicional (Visual)	54:00 min	42:30 min
		Dron + YOLOv8	17:30 min	
Terreno 3	730	Tradicional (Visual)	60:00 min	42:30 min
		Dron + YOLOv8	17:30 min	
Terreno 4	500	Tradicional (Visual)	41:35 min	31:10 min
		Dron + YOLOv8	15:15 min	
Terreno 5	1600	Tradicional (Visual)	133:20 min	93:20 min
		Dron + YOLOv8	45:00 min	

### 3.3. Pruebas estadísticas

#### 3.3.1. T – Student en tiempos de detección de malezas

##### Resumen

Se evaluó la eficiencia entre dos métodos de inspección en plantaciones de papa, comparando el método tradicional y el uso de YOLO con drones.

##### Desarrollo

Utilizando la prueba T de Student para muestras relacionadas y el coeficiente de correlación de Pearson, se calcularon las siguientes estadísticas:

##### Hipótesis

H0: No hay diferencia estadísticamente significativa en el tiempo de detección entre el método tradicional comparando con el modelo

H1: Si hay diferencia estadísticamente significativa en el tiempo de detección entre el método tradicional comparando con el modelo

Los cálculos se basaron en las fórmulas estándar para la prueba T de Student y la correlación de Pearson.

##### Resultados Clave

##### Media y Varianza

Estadística	Método Tradicional	Uso de YOLO y Dron
Media (minutos)	70.8	22.2
Varianza (minutos <sup>2</sup> )	1347.2	178.2

##### Prueba T de Student

Estadística	Valor
Valor t	0.01
Valor p	0.05

### Correlación de Pearson

Estadística	Valor
Coeficiente r	0.9796
Valor p	0.05

### Conclusión

Los datos muestran una diferencia significativa en los tiempos de inspección a favor del uso de YOLO y drones ( $p = 0.0100$ ,  $\alpha = 0.05$ ). La alta correlación ( $r = 0.9796$ ) indica una relación consistente entre los tamaños de terreno inspeccionados con ambos métodos. **Los resultados justifican la implementación del método YOLO con drones por su mayor eficiencia.**

### 3.3.2. T – Student en detección de malezas con YOLO y drones

#### Datos

Para la comparación se usó comparación entre la cantidad de etiquetas correctas realizadas por el modelo y la cantidad de etiquetas realizadas manualmente como se observa en la Tabla 13.

Tabla 13 Muestra de comparación de modelo vs expertos

Nombre del Archivo	Cantidad de Malezas Etiquetadas	Cantidad de Malezas predichas
DJI_0009.txt	30	13
DJI_0018.txt	34	13
DJI_0023.txt	33	20
DJI_0031.txt	14	7
DJI_0037.txt	18	9
DJI_0040.txt	27	19
DJI_0053.txt	63	17
DJI_0060.txt	20	6
DJI_0639.txt	7	7
DJI_0641.txt	13	13
DJI_0645.txt	27	27
DJI_0647.txt	17	17
DJI_0650.txt	8	8
DJI_0664.txt	13	13
DJI_0665.txt	17	17

DJI_0666.txt	23	23
DJI_0673.txt	42	42
DJI_0674.txt	44	44

**Resumen** Este estudio evaluó la eficacia en la detección de malezas en plantaciones de papa, comparando las anotaciones manuales con las detecciones realizadas por el modelo YOLO implementado en drones.

### Hipotesis

H0: No hay diferencia estadísticamente significativa en el número de malezas detectadas por el modelo y las etiquetadas manualmente

H1: Si hay diferencia estadísticamente significativa en el número de malezas detectadas por el modelo y las etiquetadas manualmente

**Desarrollo** Se utilizó la prueba t de Student para muestras independientes para comparar las diferencias en la cantidad de malezas detectadas entre ambos métodos. Las estadísticas clave incluyeron:

Se emplearon las fórmulas estándar para la prueba t de Student y la correlación de Pearson.

### Resultados Clave

#### Media y Varianza

Estadística	Anotaciones Manuales	Detecciones YOLO
Media (cantidad de malezas)	22,39	17,97
Varianza (cantidad de malezas <sup>2</sup> )	304,94	249,87
Observaciones	54	54

### Prueba T de Student

Estadística	Valor
Valor t	1,22038878
Valor p	0,26585671
Alfa ( $\alpha$ )	0.05

### Correlación de Pearson

#### Tabla de correlación

- **Muy débil: 0.00 a 0.19**
- **Débil: 0.20 a 0.39**
- **Moderada: 0.40 a 0.59**
- **Fuerte: 0.60 a 0.79**
- **Muy fuerte: 0.80 a 1.00**

Estadística	Valor
Coefficiente r	0.80

### Conclusión

La regla de decisión

Si  $p - \text{value}(0.265) > 0.05$  entonces no se rechaza  $H_0$

Este resultado indica que no hay una diferencia estadísticamente significativa entre las cantidades de malezas y lo que hemos supuesto como papas. El valor p es mayor que 0.05, lo que sugiere que el modelo ha otorgado buenos resultados los cuales se pueden comparar a un experto en el área. La correlación entre las dos variables es 0.80, lo que indica una fuerte correlación positiva. Esto sugiere que a medida que aumenta la cantidad de malezas etiquetadas, también tiende a aumentar la cantidad de malezas predichas por el modelo, y viceversa.

### 3.3.3. T – Student en detección de papas con YOLO vs experto

#### Datos

Para la comparación se uso comparación entre la cantidad de etiquetas de

papa correctas realizadas por el modelo y la cantidad de etiquetas de papa realizadas manualmente como se observa en la Tabla 13.

Tabla 14 Ejemplo de comparación de modelo vs expertos

Nombre del Archivo	Cantidad de Papas Etiquetadas	Cantidad de Papas predichas
DJI_0009.txt	30	13
DJI_0018.txt	34	13
DJI_0023.txt	33	20
DJI_0031.txt	14	7
DJI_0037.txt	18	9
DJI_0040.txt	27	19
DJI_0053.txt	63	17
DJI_0060.txt	20	6
DJI_0639.txt	7	7
DJI_0641.txt	13	13
DJI_0645.txt	27	27
DJI_0647.txt	17	17
DJI_0650.txt	8	8
DJI_0664.txt	13	13
DJI_0665.txt	17	17
DJI_0666.txt	23	23
DJI_0673.txt	42	42
DJI_0674.txt	44	44

**Resumen** Este estudio evaluó la eficacia en la detección de malezas en plantaciones de papa, comparando las anotaciones manuales con las detecciones realizadas por el modelo YOLO implementado en drones.

### Hipotesis

H0: No hay diferencia estadísticamente significativa en el número de papas detectadas por el modelo y las etiquetadas manualmente

H1: Si hay diferencia estadísticamente significativa en el número de malezas papas por el modelo y las etiquetadas manualmente

**Desarrollo** Se utilizó la prueba t de Student para muestras independientes para comparar las diferencias en la cantidad de papas detectadas entre ambos métodos. Las estadísticas clave incluyeron:



Se emplearon las fórmulas estándar para la prueba t de Student y la correlación de Pearson.

### Resultados Clave

#### Media y Varianza

Estadística	Anotaciones Manuales	Detecciones YOLO
Media (cantidad de malezas)	126,282609	114,521739
Varianza (cantidad de malezas^2)	5427,98502	7298,96618
Observaciones	54	54

#### Prueba T de Student

Estadística	Valor
Valor t	0,74366491
Valor p	0,44604754
Alfa ( $\alpha$ )	0.05

#### Correlación de Pearson

##### Tabla de correlación

- **Muy débil: 0.00 a 0.19**
- **Débil: 0.20 a 0.39**
- **Moderada: 0.40 a 0.59**
- **Fuerte: 0.60 a 0.79**
- **Muy fuerte: 0.80 a 1.00**

<b>Estadística</b>	<b>Valor</b>
Coeficiente r	0.998

### **Conclusión**

La regla de decisión

Si  $p - \text{value}(0.446) > 0.05$  entonces no se rechaza  $H_0$

Este resultado indica que no hay una diferencia estadísticamente significativa entre las cantidades de malezas etiquetadas manualmente y las papas detectadas por el modelo. El valor p es mayor que 0.05, lo que sugiere que el modelo ha otorgado buenos resultados los cuales se pueden comparar a un experto en el área. La correlación entre las dos variables es 0.989, lo que indica una muy fuerte correlación positiva. Esto sugiere que a medida que aumenta la cantidad de papas etiquetadas, también tiende a aumentar la cantidad de papas predichas por el modelo, y viceversa.

### **3.3.4. T – Student en detección de plantas con YOLO vs experto**

#### **Datos**

Para la comparación se uso comparación entre la cantidad de etiquetas de plantas correctas realizadas por el modelo y la cantidad de etiquetas de plantas realizadas manualmente como se observa en la Tabla 13.

*Tabla 15 Ejemplo de comparación de modelo vs expertos*

<b>Nombre del Archivo</b>	<b>Cantidad de Plantas Etiquetadas</b>	<b>Cantidad de Plantas predichas</b>
DJI_0053.txt	63	17
DJI_0060.txt	20	6
DJI_0639.txt	57	57
DJI_0641.txt	85	85
DJI_0645.txt	163	163
DJI_0647.txt	164	164
DJI_0650.txt	172	172
DJI_0664.txt	187	187
DJI_0665.txt	184	184
DJI_0666.txt	201	201
DJI_0673.txt	166	166
DJI_0674.txt	177	177

DJI_0675.txt	176	176
DJI_0677.txt	212	212
DJI_0680.txt	242	242
DJI_0681.txt	254	254
DJI_0711.txt	284	284
DJI_0730.txt	144	144

**Resumen** Este estudio evaluó la eficacia en la detección de malezas en plantaciones de papa, comparando las anotaciones manuales con las detecciones realizadas por el modelo YOLO implementado en drones.

### Hipotesis

H0: No hay diferencia estadísticamente significativa en el número de papas detectadas por el modelo y las etiquetadas manualmente

H1: Si hay diferencia estadísticamente significativa en el número de malezas papas por el modelo y las etiquetadas manualmente

**Desarrollo** Se utilizó la prueba t de Student para muestras independientes para comparar las diferencias en la cantidad de papas detectadas entre ambos métodos. Las estadísticas clave incluyeron:

Se emplearon las fórmulas estándar para la prueba t de Student y la correlación de Pearson.

### Resultados Clave

#### Media y Varianza

Estadística	Anotaciones Manuales	Detecciones YOLO
Media (cantidad de malezas)	84,1886792	82,9622642

Varianza (cantidad de malezas^2)	6427,771408	6291,152395
Observaciones	54	54

### Prueba T de Student

Estadística	Valor
Valor t	0,74366491
Valor p	0,2604754
Alfa ( $\alpha$ )	0.05

### Correlación de Pearson

#### Tabla de correlación

- **Muy débil: 0.00 a 0.19**
- **Débil: 0.20 a 0.39**
- **Moderada: 0.40 a 0.59**
- **Fuerte: 0.60 a 0.79**
- **Muy fuerte: 0.80 a 1.00**

Estadística	Valor
Coefficiente r	0.985

### Conclusión

La regla de decisión

Si  $p - \text{value}(0.260) > 0.05$  entonces no se rechaza  $H_0$

Este resultado indica que no hay una diferencia estadísticamente significativa entre las cantidades de plantas etiquetadas manualmente y las plantas detectadas por el modelo. El valor p es mayor que 0.05, lo que sugiere que el modelo ha otorgado buenos resultados los cuales se pueden comparar a

un experto en el área. La correlación entre las dos variables es 0.985, lo que indica una muy fuerte correlación positiva. Esto sugiere que a medida que aumenta la cantidad de plantas etiquetadas, también tiende a aumentar la cantidad de plantas predichas por el modelo, y viceversa.

### 3.4. Nivel de Infestación

En cuanto a la precisión para medir el nivel de infestación, el método tradicional presenta variabilidad y subjetividad, dependiendo en gran medida de la experiencia y percepción del agricultor. Contrariamente, el uso de drones junto con YOLOv8 proporciona un enfoque más objetivo y consistente para la detección de malezas, con la capacidad de procesar imágenes y clasificar malezas de manera automatizada y uniforme (Tabla 13).

Para el nivel de infestación se hizo uso de la siguiente fórmula:

$$\text{Índice de infestación de malezas} = \left( \frac{\text{Número de Malezas por imagen}}{\text{Número total de plantas por imagen (Malezas + Papas)}} \right) \times 100$$

- **Bajo nivel de infestación:** Un IIM de hasta 20% podría considerarse bajo, indicando que las malezas están presentes, pero probablemente no afectan significativamente el rendimiento del cultivo.
- **Medio nivel de infestación:** Un IIM de 20% a 50% podría ser considerado medio, sugiriendo que las malezas podrían estar comenzando a competir con el cultivo por recursos y podrían afectar el rendimiento si no se manejan adecuadamente.
- **Alto nivel de infestación:** Un IIM de más de 50% sería considerado alto, lo que indica una fuerte competencia con el cultivo y un posible impacto negativo significativo en el rendimiento (Leguizamón, 2005).

Tabla 16 Niveles de afectación de los terrenos

Identificación del Terreno	Extensión (m2)	Método	Nivel de afectación
Terreno 1	800	Tradicional (Visual)	Bajo
		Dron + YOLOv8	Medio
Terreno 2	650	Tradicional (Visual)	Bajo

		Dron + YOLOv8	Bajo
Terreno 3	730	Tradicional (Visual)	Medio
		Dron + YOLOv8	Alto
Terreno 4	500	Tradicional (Visual)	Alto
		Dron + YOLOv8	Alto
Terreno 5	1600	Tradicional (Visual)	Alto
		Dron + YOLOv8	Medio

### 3.5. Discusión

En la discusión sobre el modelo YOLOv8 utilizado en la tesis para la detección de malezas en plantaciones de papa, se resalta su rendimiento en comparación con otros modelos como YOLOv7, YOLOv5m, YOLOv4, YOLOv3, Faster R-CNN y SSD. Siendo YOLOv8 el que obtuvo el mAP más alto en detección general con un score de 67.73 obtenido del promedio entre detección de papas y malezas, se destaca en la detección específica de malezas, mostrando una especialización y eficacia en este contexto. Importante mencionar, YOLOv8 ha sido evaluado en cinco métricas de mAP, con dos resultados sobresalientes de 93 (papa) y 80 (lengua de vaca). Fig.42, indicando una alta precisión en ciertos aspectos específicos de la detección. Esta especialización y el alto rendimiento en métricas adicionales sugieren que YOLOv8 es una opción eficiente y adaptada para la tarea específica de detectar malezas en plantaciones de papa, sobre los otros modelos con los que se compara Tabla 15 teniendo en cuenta además que se hizo un estudio de multi clasificación en comparación con los trabajos relacionados que usaron el mismo dataset de Remolacha dulce (Sugar beet) en el cual se toma en cuenta solamente la planta y la maleza de forma Binaria. Teniendo concordancia con la comparación previamente obtenida en la Fig.8 de comparación de modelos YOLO.

*Tabla 17 Valores de mAP (mean Average Precision) de trabajos relacionados*

Model	Referencia	Backbone	mAP Score	mAP Plant	mAP Weed
-------	------------	----------	-----------	-----------	----------

YOLOv7	(Gallo et al., 2023)	DN-53	61.0	<b>74.1</b>	<b>48.0</b>
YOLOv5m	(Jocher Glenn, 2021.)	DN-53	51.0	67.5	34.6
YOLOv4	(Bochkovskiy et al., 2020)	DN-53	41.2	59.5	23.0
YOLOv3	(Redmon & Farhadi, 2018)	DN-53	50.4	66.3	34.6
F-RCNN	(Fan et al., 2016)	R-50 FPN	42.4	62.2	22.6
F-RCNN	(Fan et al., 2016)	R-101 FPN	42.2	62.2	22.3
F-RCNN	(Fan et al., 2016)	Rx-101 FPN	43.2	62.8	23.6
SSD	(Liu et al., 2015)	VGG-16 Bi-Real	37.4	54	20.8
<b>YOLOv8(Modelo usado)</b>		CNN ConvNeXt	67.73	<b>93</b>	<b>42,5</b>

(Gallo et al., 2023)

### 3.6. Conclusiones

1. **Desafíos y Ajustes en la Recolección de Datos del Dataset:** Se enfrentaron desafíos significativos relacionados con la calidad del dataset, que incluyeron la necesidad de eliminar imágenes debido a desenfoco, variaciones en la altura del vuelo debido al terreno irregular, y la densidad de malezas. La reducción de la velocidad del dron y el ajuste de las técnicas de captura de imágenes demostraron ser esenciales para mejorar la calidad de las imágenes.
2. **Optimización del Procesamiento de Imágenes con YOLOv8:** A través del uso de YOLOv8, se logró un equilibrio entre la resolución de las imágenes y la eficiencia del procesamiento. Aumentar la resolución mejoró la precisión del modelo (mAP), pero también aumentó el tiempo y los recursos necesarios. El recorte de imágenes y el aumento del tamaño del lote (batch size) fueron estrategias efectivas para mantener la calidad sin comprometer excesivamente los recursos.
3. **Ventajas de la Eficiencia de YOLOv8 en la Detección de Objetos:** YOLOv8, como una red neuronal convolucional rápida, se destacó por su capacidad de procesar imágenes en una sola pasada, lo que lo hace significativamente más rápido que otros modelos que requieren múltiples pasadas. Esta característica fue crucial para manejar eficientemente el gran volumen de datos de imagen, lo

que es esencial en aplicaciones de monitoreo agrícola con drones.

4. La existencia de desenfoque en su mayoría se debía debido a la velocidad del dron al ser superior a 2 m/s, ya que al tomar la fotografía no lograba enfocar de una correcta manera la imagen.

### 3.7. Recomendaciones

- Selección Óptima de Condiciones de Iluminación para la Captura de Fotografías:

Se recomienda tomar fotografías tanto en condiciones de buena luz como en días nublados. Esto contribuye a incrementar la diversidad en el dataset, lo cual es crucial para el entrenamiento de un modelo robusto y preciso. La variedad en las condiciones de iluminación ayuda a que el modelo sea más adaptable y efectivo bajo diferentes escenarios de iluminación en la práctica real.

- Monitoreo Constante de las Malezas para Prevenir la Sobrepoblación:

Es importante realizar un seguimiento continuo de las malezas en las plantaciones. Una sobrepoblación de malezas no solo dificulta su identificación y clasificación por parte del modelo, sino que también puede afectar negativamente a la plantación al competir por nutrientes esenciales. Mantener un control adecuado de las malezas es vital para la efectividad del modelo y la salud de la plantación.

- Utilización de Herramientas de Pago para Mejorar el Entrenamiento del Modelo:

Se aconseja considerar el uso de plataformas de pago como Google Colab Pro para el entrenamiento de los modelos. Estas plataformas ofrecen acceso a hardware especializado, como GPUs de alto rendimiento, que pueden acelerar significativamente el proceso de entrenamiento y manejar datasets de mayor tamaño de manera más eficiente.

- **Implementación de Gradio para la Inferencia de Modelos de IA:**

Por último, se recomienda el uso de Gradio como una herramienta versátil y fácil de usar para la inferencia de modelos de inteligencia artificial



entrenados. Gradio facilita la visualización y el uso práctico de los modelos de IA, permitiendo una interacción sencilla y efectiva con el modelo para usuarios finales o investigadores.

Estas recomendaciones proporcionan una guía para abordar aspectos críticos en la replicación o extensión del proyecto, desde la recopilación de datos hasta el entrenamiento y la implementación del modelo. Su implementación puede mejorar significativamente la precisión, eficiencia y aplicabilidad del modelo en escenarios de detección de malezas en la agricultura.

### 3.8. Bibliografía

Ahmad, M. O., Markkula, J., & Oivo, M. (2013). Kanban in software development: A systematic literature review. *Proceedings - 39th Euromicro Conference Series on Software Engineering and Advanced Applications, SEAA 2013*, 9–16. <https://doi.org/10.1109/SEAA.2013.28>

Aiman, N., Norizan, A., Razali, M., Tomari, M., Nurshazwani, W., & Zakaria, W. (2023). Object Detection Using YOLO for Quadraped Robot Manipulation. *Evolution in Electrical and Electronic Engineering*, 4(1), 329–336. <https://doi.org/10.30880/eeee.2023.04.01.039>

Alnajjar, M. (2021). *Image-Based Detection Using Deep Learning and Google Colab*. <http://dspace.alazhar.edu.ps/xmlui/handle/123456789/2681>

Ben Goertzel. (2021). *Artificial General Intelligence: 14th International Conference (14th International ...)*. Springer. [https://books.google.com.ec/books?hl=es&lr=&id=LzFYEAAAQBAJ&oi=fnd&pg=PR5&dq=Hutter,+M.+\(2022\).+Artificial+general+intelligence:+Foundations,+achievements,+and+challenges.+Springer+Nature.&ots=3D4-N-zJL3&sig=YE0KmydOk2VrW9tnE6ZOdJqNyyw#v=onepage&q&f=false](https://books.google.com.ec/books?hl=es&lr=&id=LzFYEAAAQBAJ&oi=fnd&pg=PR5&dq=Hutter,+M.+(2022).+Artificial+general+intelligence:+Foundations,+achievements,+and+challenges.+Springer+Nature.&ots=3D4-N-zJL3&sig=YE0KmydOk2VrW9tnE6ZOdJqNyyw#v=onepage&q&f=false)

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). *YOLOv4: Optimal Speed and Accuracy of Object Detection*. <https://arxiv.org/abs/2004.10934v1>

Boronat, C. D. J. (2022). *Estudio y comparación de las versiones de YOLO para la aplicación al tenis en tiempo real*. <https://idus.us.es/handle/11441/140536>

- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodei, D. (2020). Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems*, 33, 1877–1901. <https://commoncrawl.org/the-data/>
- Corceiro, A., Alibabaei, K. ;, Assunção, E. ;, & Gaspar, P. D. (2023). Methods for Detecting and Classifying Weeds, Diseases and Fruits Using AI to Improve the Sustainability of Agricultural Crops: A Review. *Processes* 2023, Vol. 11, Page 1263, 11(4), 1263. <https://doi.org/10.3390/PR11041263>
- Dang, F., Chen, D., Lu, Y., ... Z. L.-2022 A. A., & 2022, undefined. (n.d.). DeepCottonWeeds (DCW): a novel benchmark of YOLO object detectors for weed detection in cotton production systems. *Elibrary.Asabe.Org*. <https://doi.org/10.13031/aim.202200214>
- del Cerro, J., Ulloa, C. C., Barrientos, A., & de León Rivas, J. (2021). Unmanned Aerial Vehicles in Agriculture: A Survey. *Agronomy* 2021, Vol. 11, Page 203, 11(2), 203. <https://doi.org/10.3390/AGRONOMY11020203>
- Demirel, B., Baran, O. B., & Cinbis, R. G. (2023). Meta-tuning Loss Functions and Data Augmentation for Few-shot Object Detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2023-June, 7339–7349. <https://doi.org/10.1109/CVPR52729.2023.00709>
- DJI. (2018). *Mavic 2 Pro*. <https://www.dji.com/mavic-2>
- Edifarm. (2018). *Manual de Papa paso a paso* (Carlos J. Falconi Borja, AGROQUIM CÍA. LTDA, & Universidad Politécnica Salesiana, Eds.; Primera Edición, Vol. 1).
- Fan, Q., Brown, L., & Smith, J. (2016). A closer look at Faster R-CNN for vehicle detection. *IEEE Intelligent Vehicles Symposium, Proceedings, 2016-August*, 124–129. <https://doi.org/10.1109/IVS.2016.7535375>
- Gallo, I., Rehman, A. U., Dehkordi, R. H., Landro, N., La Grassa, R., & Boschetti, M. (2023). Deep Object Detection of Crop Weeds: Performance of YOLOv7 on a Real Case Dataset from UAV Images. *Remote Sensing*, 15(2), 539. <https://doi.org/10.3390/RS15020539/S1>

- Google. (2024). *Google Colab*.  
<https://research.google.com/colaboratory/intl/es/faq.html>
- Gradio. (2024). *Gradio Interface Docs*. <https://www.gradio.app/docs/interface>
- Howse, J., & Minichino, J. (2020). *Joseph Howse, Joe Minichino - Learning OpenCV 4 Computer Vision with Python 3*.  
<https://www.oreilly.com/library/view/learning-opencv-4/9781789531619/>
- Ian Goodfellow, & Yoshua Bengio. (2016). *Deep Learning*.  
<https://www.deeplearningbook.org/>
- ImageNet. (2021). *ImageNet*. <https://image-net.org/index.php>
- Iñigo Martínez. (2022). *Entrene su detector de objetos personalizado con YOLO*.  
<https://morioh.com/p/f04ccdbc3381>
- ISO. (2005). *ISO 25010*. <https://iso25000.com/index.php/normas-iso-25000/iso-25010>
- Jabran, K., Ahmad, T., Siddiqui, A. O., Üremiş, İ., & Doğan, M. N. (2023). Weed management in potato. *Potato Production Worldwide*, 121–131.  
<https://doi.org/10.1016/B978-0-12-822925-5.00013-X>
- Jocher Glenn. (n.d.). *ultralytics/yolov5: v5.0 - YOLOv5-P6 1280 models, AWS, Supervise.ly and YouTube integrations*.  
<https://doi.org/10.5281/ZENODO.4679653>
- Jocher Glenn. (2024). *Hyperparameter Tuning - Ultralytics YOLOv8 Docs*.  
<https://docs.ultralytics.com/guides/hyperparameter-tuning/#steps-involved>
- Keras. (2023). *Developer guides*. <https://keras.io/guides/>
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (n.d.). *ImageNet Classification with Deep Convolutional Neural Networks*. Retrieved March 20, 2023, from <http://code.google.com/p/cuda-convnet/>
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84–90. <https://doi.org/10.1145/3065386>
- Kruger, G. R., Davis, V. M., Weller, S. C., & Johnson, W. G. (2010). Control of Horseweed (*Conyza canadensis*) with Growth Regulator Herbicides. *Weed Technology*, 24(4), 425–429. <https://doi.org/10.1614/WT-D-10-00022.1>
- Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature* 2015

521:7553, 521(7553), 436–444. <https://doi.org/10.1038/nature14539>

Leguizamón. (2005). *El monitoreo de malezas en el campo - Facultad de Ciencias Agrarias - UNR*.  
<https://fcagr.unr.edu.ar/Extension/Agromensajes/17/1AM17.htm>

Li, L., Zhang, Q., & Huang, D. (2014). A Review of Imaging Techniques for Plant Phenotyping. *Sensors 2014, Vol. 14, Pages 20078-20111*, 14(11), 20078–20111. <https://doi.org/10.3390/S141120078>

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2015). SSD: Single Shot MultiBox Detector. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9905 LNCS, 21–37. [https://doi.org/10.1007/978-3-319-46448-0\\_2](https://doi.org/10.1007/978-3-319-46448-0_2)

López-Granados, F., Torres-Sánchez, J., Serrano-Pérez, A., de Castro, A. I., Mesas-Carrascosa, F. J., & Peña, J. M. (2016). Early season weed mapping in sunflower using UAV technology: variability of herbicide treatment maps against weed thresholds. *Precision Agriculture*, 17(2), 183–199. <https://doi.org/10.1007/S11119-015-9415-8/METRICS>

Manav Mandal. (2021). *CNN for Deep Learning | Convolutional Neural Networks*. <https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/>

Manuel Pumisacho, & Stephen Sherwood. (2002). *El cultivo de la papa en Ecuador* (Manuel Pumisacho & Stephen Sherwood, Eds.; 2002nd ed.). INIAP-CIP.  
[https://books.google.es/books?hl=es&lr=&id=QJciG\\_CWNqgC&oi=fnd&pg=PA13&dq=crecimiento+de+la+papa+ecuador&ots=y5G2gvkyZk&sig=QfFs8lgxBNZhhovtrCSWrjeUny4#v=onepage&q=crecimiento%20de%20la%20papa%20ecuador&f=false](https://books.google.es/books?hl=es&lr=&id=QJciG_CWNqgC&oi=fnd&pg=PA13&dq=crecimiento+de+la+papa+ecuador&ots=y5G2gvkyZk&sig=QfFs8lgxBNZhhovtrCSWrjeUny4#v=onepage&q=crecimiento%20de%20la%20papa%20ecuador&f=false)

Mariscal, G., Marbán, Ó., & Fernández, C. (2010). A survey of data mining and knowledge discovery process models and methodologies. *The Knowledge Engineering Review*, 25(2), 137–166. <https://doi.org/10.1017/S0269888910000032>

Medeiros, P. S., Santos, D., Beltrão, A. C., Pedraça De Souza, B., & Travassos, G. H. (2018). On the benefits and challenges of using kanban in software

- engineering: a structured synthesis study. *Journal of Software Engineering Research and Development* 2018 6:1, 6(1), 1–29. <https://doi.org/10.1186/S40411-018-0057-1>
- Niu, L., Liu, J., Yuan, Z., Yang, D., Wang, X., & Liu, W. (2023). *Improving Post-Training Quantization on Object Detection with Task Loss-Guided Lp Metric*. <https://arxiv.org/abs/2304.09785v3>
- Ong, P., Teo, K. S., & Sia, C. K. (2023). UAV-based weed detection in Chinese cabbage using deep learning. *Smart Agricultural Technology*, 4. <https://doi.org/10.1016/J.ATECH.2023.100181>
- OpenCV. (2024). *OpenCV: OpenCV modules*. <https://docs.opencv.org/4.x/>
- Orosco, R. C., & Luna, A. D. (2021). *Diseño de robot agrícola para fumigación automática de malezas en terrenos llanos usando visión inteligente FASTER R-CNN en plataforma Python*. <http://repositorio.urp.edu.pe/handle/20.500.14138/4553>
- PyTorch. (2023). *PyTorch documentation — PyTorch 2.2 documentation*. <https://pytorch.org/docs/stable/index.html>
- Ravanello, A., Desharnais, J. M., Villalpando, L. E. B., April, A., & Gherbi, A. (2014). Performance measurement for cloud computing applications using ISO 25010 standard characteristics. *Proceedings - 2014 Joint Conference of the International Workshop on Software Measurement, IWSM 2014 and the International Conference on Software Process and Product Measurement, Mensura 2014*, 41–49. <https://doi.org/10.1109/IWSM.MENSURA.2014.33>
- Redmon, J., & Farhadi, A. (2018). *YOLOv3: An Incremental Improvement*. <https://arxiv.org/abs/1804.02767v1>
- Reis, D., Kupec, J., Hong, J., & Daoudi, A. (2023a). *Real-Time Flying Object Detection with YOLOv8*. <https://arxiv.org/abs/2305.09972v1>
- Reis, D., Kupec, J., Hong, J., & Daoudi, A. (2023b). *Real-Time Flying Object Detection with YOLOv8*. <https://arxiv.org/abs/2305.09972v1>
- Reyes, Y., Vergara, I., Torres, O., Díaz Lagos, M., & González Jimenez, E. E. (2016). Contaminación por metales pesados: Implicaciones en salud, ambiente y seguridad alimentaria. *Ingeniería Investigación y Desarrollo: I2+D, ISSN-e 2422-4324, ISSN 1900-771X, Vol. 16, N° 2, 2016 (Ejemplar*

*Dedicado a: Julio-Diciembre 2016*), Págs. 66-77, 16(2), 66–77.  
<https://dialnet.unirioja.es/servlet/articulo?codigo=6096110&info=resumen&idioma=ENG>

Robayo Ordoñez, A., & Santillán, D. G. (2022). *Construcción de un sistema de visión por computadora para la detección de maleza y líneas de cultivo utilizando imágenes provenientes de un Dron*.  
<http://repositorio.utn.edu.ec/handle/123456789/11950>

Roboflow. (2023). *Introduction - Roboflow Docs*. <https://docs.roboflow.com/>

Scholnik-Elias, A., Martínez-Díaz, S., Luna-Taylor, J. E., & Castro-Liera, I. (2023). Detección de armas tipo pistola mediante el uso de redes convolucionales con una arquitectura tipo YOLO y estereoscopía. *Pädi Boletín Científico de Ciencias Básicas e Ingenierías Del ICBI*, 11(Especial2), 196–204.  
<https://doi.org/10.29057/ICBI.V11IESPECIAL2.10727>

Sishodia, R. P., Ray, R. L., & Singh, S. K. (2020). Applications of Remote Sensing in Precision Agriculture: A Review. *Remote Sensing 2020*, Vol. 12, Page 3136, 12(19), 3136. <https://doi.org/10.3390/RS12193136>

Soltani, N., Nurse, R. E., & Sikkema, P. H. (2016). Response of glyphosate-resistant soybean to dicamba spray tank contamination during vegetative and reproductive growth stages. *Canadian Journal of Plant Science*, 96(1), 160–164.  
[https://doi.org/10.1139/CJPS-2015-0175/ASSET/IMAGES/LARGE/CJPS\\_2015-0175F1.JPEG](https://doi.org/10.1139/CJPS-2015-0175/ASSET/IMAGES/LARGE/CJPS_2015-0175F1.JPEG)

Szeliski, R. (2022). *Computer Vision*. <https://doi.org/10.1007/978-3-030-34372-9>

TensorFlow. (2023). *TensorFlow Core*. <https://www.tensorflow.org/guide?hl=es-419>

Terven, J. R., & Cordova-Esparza, D. M. (2023). *A Comprehensive Review of YOLO: From YOLOv1 and Beyond*. <https://arxiv.org/abs/2304.00501v5>

Ultralytics. (2023a). *Ultralytics YOLOv8 Docs*.  
<https://docs.ultralytics.com/tasks/detect/#models>

Ultralytics. (2023b). *YOLOv8 - Ultralytics YOLOv8 Docs*.  
<https://docs.ultralytics.com/models/yolov8/#performance-metrics>

Utstumo, T., Urdal, F., Brevik, A., Dørum, J., Netland, J., Overskeid, Ø., Berge,

- T. W., & Gravdahl, J. T. (2018). Robotic in-row weed control in vegetables. *Comput. Electron. Agric.*, *154*, 36–45. <https://doi.org/10.1016/J.COMPAG.2018.08.043>
- Vincent, A. M., & Jidesh, P. (2023). An improved hyperparameter optimization framework for AutoML systems using evolutionary algorithms. *Scientific Reports* *2023 13:1*, *13(1)*, 1–19. <https://doi.org/10.1038/s41598-023-32027-3>
- Volpato, L., Pinto, F., González-Pérez, L., Thompson, I. G., Borém, A., Reynolds, M., Gérard, B., Molero, G., & Rodrigues, F. A. (2021). High Throughput Field Phenotyping for Plant Height Using UAV-Based RGB Imagery in Wheat Breeding Lines: Feasibility and Validation. *Frontiers in Plant Science*, *12*, 185. <https://doi.org/10.3389/FPLS.2021.591587/BIBTEX>
- Wang, X., Kong, L., Zhang, Z., Wang, H., & Lu, X. (2023). Keypoint regression strategy and angle loss based YOLO for object detection. *Scientific Reports* *2023 13:1*, *13(1)*, 1–10. <https://doi.org/10.1038/s41598-023-47398-w>
- William G. Johnson, Pauley R. Bradley, Stephen E. Hart, Michelle L. Buesinger, & Raymond E. Massey. (2000). Efficacy and Economics of Weed Management in Glyphosate-Resistant Corn (*Zea mays*) on JSTOR. *Cambridge University Press*. <https://www.jstor.org/stable/3988506>
- Xiao, X., Feng, X., Xiao, X., & Feng, X. (2023). Multi-Object Pedestrian Tracking Using Improved YOLOv8 and OC-SORT. *Senso*, *23(20)*, 8439. <https://doi.org/10.3390/S23208439>
- Zambrano, J. L., Velásquez Carrera, J. S., Peñaherrera Mafla, D. F., Sangoquiza Caiza, C. A., Cartagena Ayala, Y. E., Villacrés Poveda, C. E., Garcés, S., Ortíz Calle, R., León Ruiz, J., Campaña, D., López Guerrero, V. A., Asaquibay, C., Nieto Beltrán, M., Sanmartín Mesias, G. del R., Pintado, P., LastNameYáñez G., C., & Racines Jaramillo, M. R. (2021). *Guía para la producción sustentable de maíz en la Sierra ecuatoriana*. <http://repositorio.iniap.gob.ec/handle/41000/5796>
- Zhu, Y., Ruan, Y., Xiong, Z., Guo, S., & Technology, M. (2023). *Edge Wasserstein Distance Loss for Oriented Object Detection*. <https://arxiv.org/abs/2312.07048v1>
- Zimdahl, R. L. (2018). *Fundamentals of weed science*. 735.

## ANEXOS

**Anexo 1:** Conjunto de imágenes que forman el dataset: [Imágenes papas](#)

**Anexo 2:** Scripts usados en el proyecto [Programacion](#)



NOMBRE DEL TRABAJO

Trabajo\_de\_grado\_Ortega\_Bryan\_v1.pdf

AUTOR

Bryan Ortega

RECUENTO DE PALABRAS

20563 Words

RECUENTO DE CARACTERES

120202 Characters

RECUENTO DE PÁGINAS

97 Pages

TAMAÑO DEL ARCHIVO

5.2MB

FECHA DE ENTREGA

Mar 5, 2024 8:01 AM GMT-5

FECHA DEL INFORME

Mar 5, 2024 8:03 AM GMT-5

**● 8% de similitud general**

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para cada base de datos.

- 6% Base de datos de Internet
- Base de datos de Crossref
- 6% Base de datos de trabajos entregados
- 1% Base de datos de publicaciones
- Base de datos de contenido publicado de Crossref

**● Excluir del Reporte de Similitud**

- Material bibliográfico
- Material citado
- Fuentes excluidas manualmente
- Material citado
- Coincidencia baja (menos de 10 palabras)

