

Estimación del crecimiento de las plantas mediante visión artificial

Jhon Jairo Beltrán Rodríguez

Juan Pablo Carvallido Mesa

Trabajo de Grado para Optar al Título de
Ingeniero Electrónico

Director

Jaime Guillermo Barrero Pérez

Magíster en potencia eléctrica

Universidad Industrial de Santander

Facultad de Ingenierías fisicomecánicas

Escuela de Ingenierías

Eléctrica, electrónica y de telecomunicaciones

Bucaramanga

2024

Dedicatoria

Dedico este proyecto de grado con profundo agradecimiento a Dios, quien en los momentos más difíciles y complejos me ha brindado sabiduría y paciencia, guiándome en este camino académico y profesional.

A mi madre, Nhora, cuyo apoyo inquebrantable y fe en mi crecimiento han sido el pilar fundamental de este viaje. Gracias por tus consejos, enseñanzas y el amor constante que me has brindado a lo largo de este proceso.

A mi hija, Annie Sofia, mi fuente de inspiración y motor incansable. Cada día me esfuerzo más gracias al amor y la fortaleza que me brindas. Gracias por ser parte de mí y acompañarme en este desafío.

Finalmente, expresar mi gratitud a mi familia y a la Universidad Industrial de Santander por su apoyo constante en mi desarrollo académico y profesional. Su respaldo ha sido fundamental para alcanzar este logro.

Jhon Jairo Beltrán Rodríguez

Dedico este trabajo de grado a Dios, que toda la honra y la gloria sean para él, cuya guía y amor incondicional han iluminado cada paso de mi camino académico.

A mi querida madre, Alicia Maria, cuyo amor, sacrificio, dedicación y apoyo inquebrantable han sido mi mayor motivación. Tus palabras de aliento, tus consejos y tus abrazos cálidos me han dado la fuerza para seguir adelante en los momentos más difíciles.

A mi padre, Luis Carlos, quien siempre ha sido mi ejemplo de perseverancia. Tú constante apoyo ha sido fundamental en este viaje hacia la realización de mis sueños.

Expreso mi más sincera gratitud a la Universidad industrial de Santander, por brindarme las herramientas y oportunidades para crecer académica y personalmente. Agradezco a mis profesores por su dedicación y conocimientos impartidos, y a mis compañeros por su colaboración y compañerismo.

Juan Pablo Carvallido Mesa

Agradecimientos

A mi querida novia Sheyla, agradezco sus constantes palabras de motivación, apoyo y amor incondicional ya que en todo mi proceso estudiantil me ha acompañado y ayudado a ser una mejor persona. Su perspectiva y apoyo emocional fueron pilares clave en la realización de este proyecto, gracias por ser esa persona tan especial para mí y ayudarme en este logro tan importante.

A mi prima Paula cuyo apoyo incondicional y habilidades en la guía y manejo del equipo de captura fueron esenciales para el éxito de este proyecto.

Al director de trabajo de grado Jaime Guillermo Barrero Perez quiero expresar mi más sincero agradecimiento. Su constante dedicación, orientación y apoyo fueron fundamentales para

la culminación exitosa de este proyecto. Su experiencia y liderazgo guiaron cada paso, y su contribución ha dejado una marca significativa en este logro.

Jhon Jairo Beltrán Rodríguez

Quisiera expresar mi sincero agradecimiento a las siguientes personas, cuyo apoyo y contribuciones fueron fundamentales para la realización de este proyecto de grado:

A mi querida novia, Sara Valentina, por su amor, paciencia y comprensión durante este arduo proceso. Su apoyo incondicional y su ánimo constante fueron una fuente de inspiración para mí. Gracias por estar a mi lado en cada paso del camino y por ser mi roca en los momentos difíciles.

A la señora Rossemary, quien generosamente brindó su apoyo económico para hacer posible la realización de este proyecto. Su generosidad y confianza en mi trabajo fueron un impulso invaluable que me permitió concentrarme en mi investigación y desarrollo.

Al director de mi proyecto de grado, el profesor Jaime Guillermo Barrero Pérez, por su orientación experta, su sabiduría y su dedicación. Su compromiso con la excelencia académica y su liderazgo inspirador fueron fundamentales para el éxito de este proyecto. Agradezco sinceramente sus consejos, comentarios y aliento a lo largo de este proceso.

Juan Pablo Carvallido Mesa

Tabla de Contenido

	Pág.
Introducción	11
1. Objetivos.....	12
1.1 Objetivo General.....	12
1.2 Objetivos Específicos.....	12
2. Marco teórico	13
2.1 Deep learning y redes neuronales	13
2.1.1 Redes Neuronales Convolucionales.....	14
2.1.2 Redes neuronales profundas preentrenadas	16
2.2. Filtros	19
2.2.1. Convolución.....	21
2.2.2. Detectores de borde.....	21
2.3. Rábanos.....	23
3. Metodología	23
3.1. Proceso de filtrado y equipo de captura de imágenes	23
3.2. Detección de la planta.....	25
3.3. Estimación de la altura.....	28
3.3. Comparación del crecimiento	29
4. Resultados.....	30
4.1 Modelo YOLOv8l.....	30
4.2 Estimación del crecimiento.....	33
5. Conclusiones.....	37

6. Recomendaciones 38

Referencias Bibliográficas 39

Lista de Tablas

	Pág.
Tabla 1 Modelos de entrenamiento.....	27
Tabla 2 Parámetros de entrenamiento del modelo.....	28
Tabla 3 Resultados de entrenamiento YOLOv8l.....	33

Lista de Figuras

	Pág
Figura 1 Representación esquemática del modelo matemático de una neurona artificial	14
Figura 2 Red neuronal convolucional con múltiples capas de convolución y agrupación	16
Figura 3 Detección de objetos basados en YOLO	18
Figura 4 Ventana de filtro	20
Figura 5 Kernels para el filtro sobel	24
Figura 6 Comparación imagen filtrada vs original	25
Figura 7 Herramienta de anotación de imagenes graficas “Labelme”	26
Figura 8 Fotos de las medidas manuales.....	29
Figura 9 Validación de etiquetas y predicción del modelo.....	30
Figura 10 Matriz de confusión.....	31
Figura 11 Curva de precisión y recall.....	32
Figura 12 Perdidas de datos del modelo	33
Figura 13 Resultado de la estimación de la altura por software	34
Figura 14 Comparación entre medidas manuales y las de software del primero de marzo.....	35
Figura 15 Comparación entre medidas manuales y las de software del nueve de marzo	35
Figura 16 Comparación entre medidas manuales y las de software del quince de marzo.....	36

Resumen

Título: Estimación del crecimiento de las plantas mediante visión artificial

Autor: Jhon Jairo Beltrán Rodríguez, Juan Pablo Carvallido Mesa

Palabras Clave: Visión por computador, imágenes digitales, Python, Opencv.

Descripción:

El presente proyecto se enfoca en la aplicación de tratamiento digital de imágenes y visión por computador con el objetivo principal de desarrollar una solución actualizada y económica para la estimación del crecimiento de las plantas de rabano. La necesidad de una herramienta eficaz y de fácil acceso para los usuarios finales motiva esta investigación, con la intención de proporcionar una solución práctica y asequible en el ámbito de la agricultura y la investigación botánica, al igual que a lo largo de este trabajo, se exploran y aplican técnicas avanzadas de procesamiento de imágenes en una CPU con un procesador Intel(R) Core(TM) i5-9300H, al igual que algoritmos de visión por computador, aplicados sobre un lote de doce plantas de rabano en un periodo de tiempo de veinticinco días, con el fin de hacer un registro fotográfico de la captura de imagen de las plantas cada dos días, para así, evaluar el crecimiento de las plantas en distintas etapas. La adaptabilidad y simplicidad de la herramienta desarrollada son elementos clave, permitiendo a los usuarios, incluso sin experiencia técnica avanzada, utilizarla de manera efectiva.

* Trabajo de Grado

** Facultad de Ingenierías Fisicomecánicas. Escuela de Ingeniería Eléctrica, Electrónica y Telecomunicaciones (E3T). Mag. Jaime Guillermo Barrero Pérez.

Abstract

Title: PLANT GROWTH ESTIMATION THROUGH COMPUTER VISION

Author(s): Jhon Jairo Beltrán Rodríguez, Juan Pablo Carvallido Mesa

Key Words: Computer Vision, Digital Images, Python, OpenCV

Description: This project focuses on the application of digital image processing and computer vision techniques to develop an up-to-date and cost-effective solution for estimating the growth of radish plants. The need for an efficient and user-friendly tool for end-users motivates this research, aiming to provide a practical and affordable solution in the field of agriculture and botanical research. Throughout this work, advanced image processing techniques on a CPU with an Intel(R) Core(TM) i5-9300H processor and computer vision algorithms were explored and applied to a set of twelve radish plants over a period of twenty-five days. Images of the plants were captured every two days to evaluate their growth at different stages. The adaptability and simplicity of the developed tool are key elements, allowing users, even without advanced technical experience, to use it effectively.

* Degree Work

**Faculty of Physicmechanics Engineering. School of Electrical, Electronic and Telecommunications Engineering (E3T). Mag. Jaime Guillermo Barrero Pérez.

Introducción

En un mundo en constante evolución, la comprensión y cuantificación del crecimiento de las plantas se alza como un imperativo científico y tecnológico (Adnan et al., 2017) y (Sharma et al., 2020). La esencia misma de la vida en la Tierra y la subsistencia de la humanidad dependen en gran medida de la vegetación que cubre nuestro planeta. Sin embargo, la metodología tradicional de medición del crecimiento de las plantas en Colombia se basa en técnicas invasivas y destructivas que involucran la extracción de muestras de tejido vegetal, lo que pone en riesgo la salud de las plantas, disminuye su producción y afecta la biodiversidad local (Ariza, 2020). Debido a esto, la conjunción de la ciencia botánica con los avances de la visión artificial emerge como un hito en la exploración y optimización de este fundamental proceso biológico (Zang F, 2014).

En las últimas décadas, los sistemas de inspección por visión por computadora se han convertido en herramientas importantes en las operaciones agrícolas (Rico-Fernández et al., 2019), un campo interdisciplinario que amalgama la óptica, la informática y la inteligencia artificial como lo es la visión por computador, se posiciona en la vanguardia de la investigación científica y tecnológica, como se puede observar permite la adquisición, análisis y extracción de información precisa a partir de imágenes de plantas en desarrollo (Tabares M, 2022)

En revisión de la literatura, a lo largo de los años se han propuesto diversas técnicas para la estimación del crecimiento de las plantas como ha sido el método de sistema monocular basado en la visión artificial. En Cai & Walker, (2010) se emplea dicho sistema a bordo de un vehículo aéreo no tripulado para lograr capturas de imágenes en la vista superior de los árboles. Otra opción para la estimación del crecimiento mediante visión artificial es la visión estéreo, que consiste en la utilización de dos sensores en paralelo, donde cada sensor toma una imagen al mismo tiempo para ser procesadas (Rovira-Más, Zhang y Reid, 2008). Esta técnica tiene como ventaja principal

proporcionar la profundidad del objeto de estudio en relación a la posición de la cámara considerada como origen.

Teniendo en cuenta esta información y realizado el estudio oportuno, el proyecto de trabajo de grado le apuesta a la necesidad de contar con un prototipo económico, con bajo costo computacional y de instrumentación, el cual permita la estimación del crecimiento de la planta utilizando técnicas de procesamiento digital de imágenes e inteligencia artificial.

1. Objetivos

1.1 Objetivo General

Desarrollar un sistema para la medición del crecimiento de la planta de rábano utilizando técnicas de procesamiento digital de imágenes e inteligencia artificial.

1.2 Objetivos Específicos

Crear una base de datos de imágenes de la planta de rábano etiquetadas con información de referencia, como la altura real, para el entrenamiento y validación de modelos de inteligencia artificial.

Seleccionar y configurar modelos de aprendizaje profundo, como redes neuronales convolucionales (CNN), para la detección de características en las imágenes junto con técnicas de procesamiento digital de imágenes.

Evaluar el desempeño del sistema de visión artificial mediante pruebas en condiciones controladas, comparando las mediciones obtenidas con las mediciones manuales de referencia.

2. Marco teórico

En esta sección se presenta de manera concisa los conceptos que fueron necesarios comprender y abordar para exponer alternativas que brinden respuesta a lo planteado anteriormente.

2.1 *Deep learning* y *redes neuronales*

“El aprendizaje profundo o deep learning se basa en aprender por medio de varias capas de representación, tratando de imitar el cerebro humano a través de una combinación de entradas de datos” (¿Qué Es Deep Learning? | IBM, n.d.). Podemos denominar la profundidad por medio del número de niveles de capas que contenga el modelo, las capas se logran aprender al momento de exponerlas ante el conjunto de datos de entrenamiento, todo esto de manera automática (Cano Avalos, 2022). Todo este sistema biológico de conexiones entre neuronas se le conoce como redes neuronales biológicas.

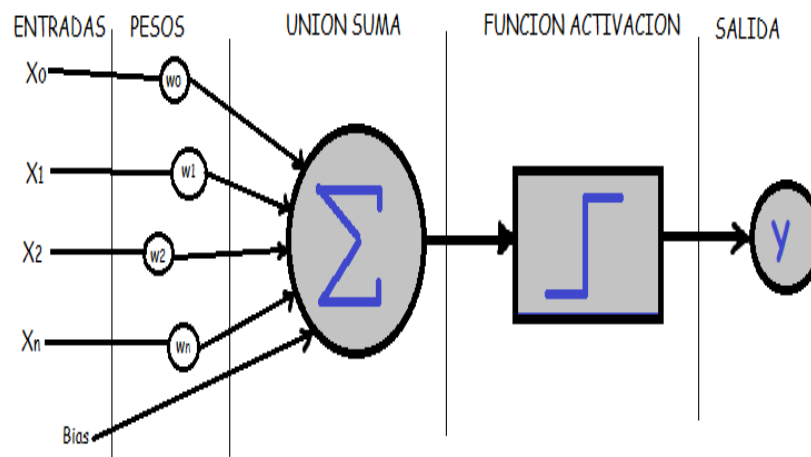
Las redes neuronales logran aprender utilizando una estructura que guarda cierta relación con las neuronas del sistema nervioso humano. “La neurona es una simple unidad procesadora que recibe y combina señales desde y hacia otras neuronas. Si la combinación de entradas es suficientemente fuerte la salida de la neurona se activa.” (Olabe (1998), p. 2-3). Al principio, todas las ponderaciones son aleatorias y las respuestas que resultan de la red son, posiblemente, disparatadas, por ese motivo la red aprende a través del entrenamiento, este entrenamiento se emplea mediante ejemplos para los que se conoce el resultado, y las respuestas que proporciona la red se comparan con los resultados conocidos, seguido a esto la información resultante de esta comparación se pasa hacia atrás a través de la red, cambiando las ponderaciones gradualmente. A medida que progresa el entrenamiento, la red se va haciendo cada vez más precisa en la replicación de resultados conocidos. Una vez entrenada, la red se puede aplicar a casos futuros en los que se

desconoce el resultado. “Las redes neuronales pueden simplemente aprender a resolver problemas, a veces hay problemas en los que sería extremadamente difícil diseñar directamente un sistema convencional o circuito.” (Nielsen, 2015).

La Figura 1. muestra la representación esquemática de un modelo matemático de una neurona artificial, destacando la entrada X_n , peso (w_n), sesgo (bias), función de suma (Σ), función de activación y señal. Las redes neuronales artificiales se componen de este nuevo concepto de neurona.

Figura 1

Representación esquemática del modelo matemático de una neurona artificial



Tomado de: Autor propio

2.1.1 Redes Neuronales Convolucionales

Las redes neuronales convolucionales (CNN) son un tipo de red neuronal artificial, se usan frecuentemente para la clasificación y visión por computadora y están diseñadas específicamente para manejar una variedad de formas en 2D aprovechando los principios del álgebra lineal, como lo afirma (Nielsen, 2015). Estas redes usan arquitectura adaptada a la clasificación de imágenes.

Haciendo que las redes convolucionales sean rápidas de entrenar. Esto, a su vez, nos ayuda a entrenar en profundidad y en muchas capas. Hoy en día, las redes convolucionales profundas se utilizan en la mayoría de los casos para el reconocimiento de imágenes.

Las redes neuronales convolucionales tienen tres tipos principales de capas:

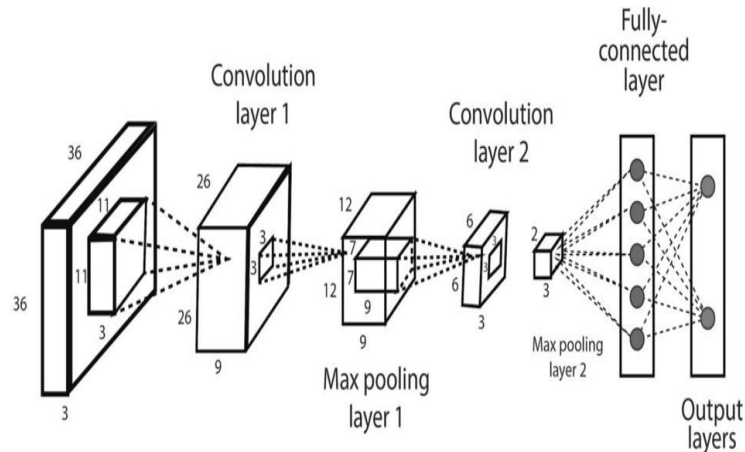
Capa convolucional: Esta capa es el componente básico de la CNN donde se realiza la mayor parte del cálculo; consta principalmente de datos de entrada, filtros y mapas de características. Un detector de características es una matriz bidimensional (2D) de pesos que representa una parte de una imagen. Después de cada operación de convolución, la CNN aplica una transformación de unidad lineal rectificadora (ReLU) al mapa de características.

Capa de agrupación: La agrupación de capas, también conocida como submuestreo, realiza una reducción de dimensionalidad; esto reduce la cantidad de parámetros en la entrada. De manera similar a una capa convolucional, la operación de agrupación barre un filtro sobre toda la imagen, pero la diferencia es que este filtro no tiene pesos.

Capa completamente conectada (FC): Esta capa realiza tareas de clasificación basadas en características extraídas a través de capas anteriores y sus diferentes filtros. Mientras que las capas convolucionales y de agrupación tienden a usar funciones ReLU, las capas FC generalmente utilizan funciones de activación softmax para clasificar adecuadamente la entrada, lo que genera probabilidades entre 0 y 1.

Figura 2

Red neuronal convolucional con múltiples capas de convolución y agrupación.



Tomado de: Sarker, I.H. Machine Learning: Algorithms, Real-World Applications and Research Directions. SN COMPUT. SCI. 2, 160 (2021). <https://doi.org/bibliotecavirtual.uis.edu.co/10.1007/s42979-021-00592-x>

Como se muestra en la Figura 2, la red neuronal convolucional (CNN) mejora el diseño de la red neuronal artificial (ANN) estándar, que consta de capas convolucionales, capas de agrupación y capas completamente conectadas (LeCun et al., 1998).

2.1.2 Redes neuronales profundas preentrenadas

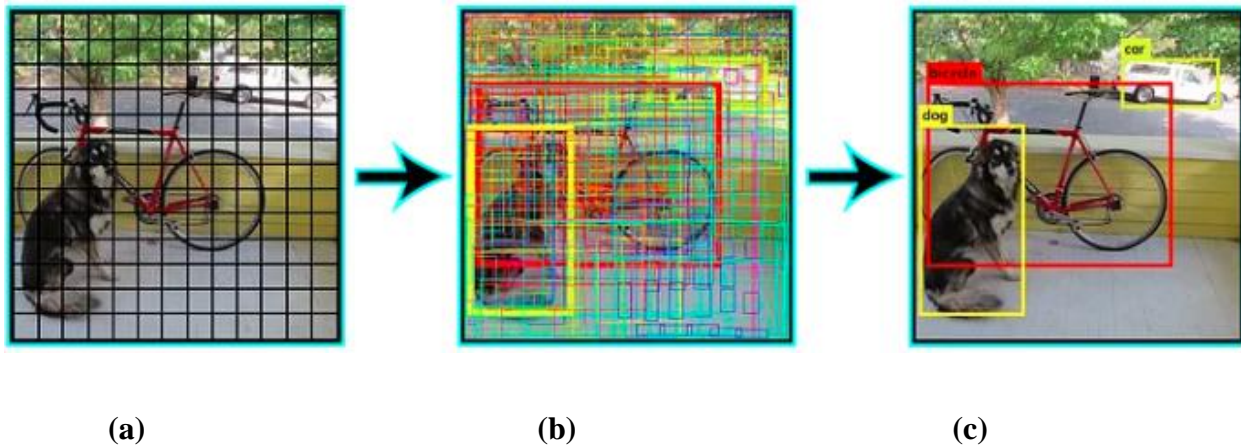
Una prometedora alternativa es el uso de redes preentrenadas, desde campos de aplicación diferentes. Los modelos preentrenados están siendo aplicados de manera exitosa en varias tareas dentro del campo de visión de las computadoras, que puede ser como un generador de características o como una base para el aprendizaje mediante transferencia (Razavian et al., 2014) (Penatti et al., 2015).

De esta manera según (Chanampe et al., s. f.) hay dos estrategias aplicadas a la transferencia de aprendizaje que se pueden identificar, una de ellas es el uso de las redes preentrenadas solamente como extractor de características y la segunda estrategia es hacer pequeños ajustes

(ajuste fino), sobre los pesos de una red preentrenada, para lograr un mejor rendimiento sobre los datos a procesar (Krizhevsky et al., 2012) (He et al., 2016).

Yolo, es uno de los mejores modelos de reconocimiento de objetos debido a su capacidad para detectar objetos en tiempo real con mayor precisión (Supeshala, 2021). Yolo utiliza una red de detección de objetos de una sola etapa las cuales emplea de las redes previamente entrenadas mencionadas anteriormente.

Para llevar a cabo la detección primero divide la imagen en una cuadrícula de $S \times S$ como se muestra en la Figura 3.a. En cada una de las celdas predice N posibles “bounding boxes” o también llamados cuadros de anclaje, los cuales se obtienen utilizando el algoritmo K-Means para agrupar el ancho y el alto de todos los cuadros etiquetados en los datos de entrenamiento (Getting Started with YOLO v2 - MATLAB & Simulink, s. f.). Además, calcula el nivel de certidumbre (o probabilidad) de cada una de ellas, observado en la Figura 3.b. es decir, se calculan $S \times S \times N$ diferentes cajas, la gran mayoría de ellas con un nivel de certidumbre muy bajo. Después de obtener estas predicciones se procede a eliminar las cajas que estén por debajo de un límite. A las cajas restantes se les aplica un paso de “non-max suppression” que sirve para eliminar posibles objetos que fueron detectados por duplicado y así dejar únicamente el más exacto de ellos como se registra en la Figura 3.c.

Figura 3*Detección de objetos basados en YOLO*

Tomado de: (Enrique, A, 2018)

Algunos conceptos esenciales que ayudarán a comprender factores importantes a la hora de entrenar un modelo son:

Datos de entrenamiento: Cuando tenemos un dataset, también llamado agrupación de datos, es importante dividirlo en dos o tres sets diferentes, el primero de ellos es el set de entrenamiento, que se usa para que la red neuronal aprenda patrones sobre los datos.

Datos de validación: Es necesario que este set tenga datos diferentes de los otros dos sets, ya que se quiere ver que tan buena es la red neuronal en datos que nunca ha visto.

Tasa de aprendizaje: Este hiperparámetro indica que tan largo será el camino que tome el algoritmo de optimización.

Función de pérdida: La función de pérdida, también conocida como función de costo, es la función que nos dice que tan buena es la red neuronal, ya que un resultado alto indica que la red neuronal tiene un desempeño pobre y un resultado bajo indica que la red neuronal está haciendo un buen trabajo.

Época: Alude al número de épocas o iteraciones completas que se utilizarán durante el entrenamiento. En cada ciclo todos los datos de entrenamiento pasan por la red neuronal para que esta aprenda sobre ellos, si existen cinco ciclos y dos mil datos, los dos mil datos pasaran cinco veces por la red neuronal.

2.2. Filtros

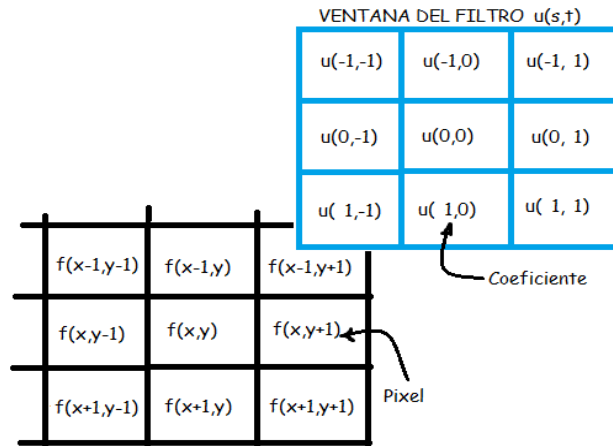
El filtrado es un concepto tomado del procesamiento en el dominio de la frecuencia. Se refiere a pasar, modificar o rechazar componentes de frecuencia específicas de una imagen. Los gráficos ilustran los filtros existentes según las componentes de frecuencia que preservan (Gómez Flórez, 2020).

El filtrado espacial modifica una imagen reemplazando el valor de cada píxel por una función de los valores del píxel y sus vecinos. Un filtro espacial utiliza una ventana (máscara o kernel) que se desplaza píxel por píxel generando una imagen filtrada. El origen del filtro se ubica sobre la coordenada (x,y) del píxel que será modificado. Un filtro espacial puede ser lineal o no lineal.

Un filtro lineal realiza una suma de productos entre una imagen f y una ventana de filtro u . El tamaño de u ($m \times n$) define el vecindario y sus coeficientes determinan la naturaleza del filtro.

Figura 4

Ventana de filtro



Tomado de: Autor propio

Como se puede observar en la figura 4, el pixel modificado es:

$$g(x, y) = u(-1, -1)f(x-1, y-1) + \dots + u(0, 0)f(x, y) + \dots + u(1, 1)f(x+1, y+1) \quad (\text{Ec. 1})$$

Los filtros lineales espaciales más conocidos son:

- Filtro pasa bajos: Generalmente se usa para reducción de ruido.
- Filtro pasa altos: Mayormente utilizado para la detección de bordes.
- Filtro paso banda: Comúnmente usado para el realce de características como las texturas.

Tomando como base los filtros lineales se explicará acerca de los filtros detectores de bordes, debido a que, la característica de borde es una de las características importantes de la imagen, y es una parte importante de procesamiento de imágenes, reconocimiento de patrones y visión por computadora (Edge Detection Based on Improved Sobel Operator).

A lo largo de los años, la segmentación de imágenes ha ido atrayendo cada vez más atención por parte de investigadores y desarrolladores. Se han puesto miles de algoritmos de

segmentación. Los cuales, se pueden dividir en método de umbral de bits, método de detección de bordes y método de crecimiento regional (Numerische Mathematik, 1994)

2.2.1. Convolución

La función de convolución se utiliza para aplicar filtros a los valores de píxel de una imagen. Estos filtros pueden tener varios propósitos, como aumentar la nitidez de la imagen, difuminarla, detectar bordes u otros realces basados en un kernel específico. Los filtros se utilizan para mejorar la calidad general de una imagen al eliminar datos falsos o mejorar las características de los datos existentes (*Función De convolución—ArcMap / Documentación*, n.d.). La convolución implica la aplicación de un kernel móvil o superpuesto, como por ejemplo un kernel 3x3, a la imagen. Este proceso implica calcular el valor de cada píxel en función de la ponderación de sus vecinos dentro de la ventana o vecindad definida por el kernel. En resumen, la función de convolución es una herramienta esencial en el procesamiento de imágenes para realizar diversas operaciones de mejora y realce, esta operación está representada por,

$$g(x_i, y_i) = \sum_{\forall(x,y) \in \Omega_i} h(-x, -y) \Omega_i(x, y) \quad (\text{Ec. 2})$$

Como se puede observar en la ecuación, el kernel denominado $h(x,y)$ está invertido de la forma $h(-x, -y)$, luego multiplicando punto a punto con el vecindario $\Omega_i(x,y)$ y sumando cada uno de sus productos nos da como resultado el píxel de salida $g(x_i, y_i)$, por lo tanto, la imagen de salida va a estar dada por el resultado de cada píxel. Dado que la convolución es un proceso lineal, se cumplen las propiedades de homogeneidad y aditividad.

2.2.2. Detectores de borde

El método de detección de bordes en procesamiento de imágenes es crucial para identificar cambios abruptos en la intensidad de los píxeles, lo que a menudo indica la presencia de bordes o

límites entre objetos. En este contexto, existen varios operadores de detección de bordes, cada uno con sus propias características y aplicaciones.

Borde Diferencial de Primer Orden: Este método se basa en calcular las diferencias de intensidad entre píxeles vecinos para detectar bordes. Es uno de los enfoques más simples pero menos efectivos.

Operador de Roberts: El operador de Roberts utiliza una máscara de convolución para calcular gradientes en dos direcciones: diagonal y antidiagonal. Sin embargo, su efectividad es limitada en comparación con otros operadores más avanzados.

Operador Prewitt: Similar al operador de Roberts, el operador Prewitt calcula gradientes en dirección vertical y horizontal, pero utilizando máscaras de convolución más elaboradas. Aunque mejora la detección de bordes, sigue presentando limitaciones.

Operador LOG (Laplaciano de Gaussiano): El operador LOG combina un suavizado gaussiano con un filtro de Laplaciano para detectar bordes. Esta técnica es más efectiva para identificar bordes de baja intensidad en imágenes con ruido.

Operador Sobel: El operador Sobel es uno de los más utilizados debido a su capacidad para detectar bordes en múltiples direcciones. Sin embargo, el operador clásico de Sobel tiene limitaciones, ya que solo es sensible a las direcciones vertical y horizontal. Para abordar las limitaciones del operador Sobel clásico, se han propuesto diversas mejoras. Por ejemplo, algunos enfoques incluyen la adaptación del operador para detectar bordes en otras direcciones, como la diagonal. Esto permite una detección más completa de los bordes en la imagen y evita la pérdida de información (Zhang & Fang, 2016).

2.3. Rábanos

El Rábano (*Raphanus sativus* L.) es un cultivo de origen asiático, sin embargo, ya se cultivaba desde la antigüedad en Grecia y Egipto. En la época de los griegos y romanos se convirtió en un alimento muy apreciado, fueron éstos últimos quienes extendieron el cultivo por toda Europa, donde se cultivaba el rábano grande, especialmente en la parte norte de Inglaterra. (INFOAGRO, 2009).

Los requerimientos de los rábanos (*raphanus sativus*) de la familia Brassicaceae se basan en que su cultivo sea en clima templado seco, su desarrollo óptimo es alrededor de los 20°C (Fontalvo, 2021).se pueden cultivar en macetas individuales con una profundidad de 10 cm, lo destacado de cultivar rábanos es que es un cultivo de ciclo corto, después de 25 a 45 días de la siembra ya se puede empezar a cosechar rábanos. De preferencia cuando el bulbo o (fruto) esté tierno y ya comience a asomarse.

El riego debe ser abundante sin pasarse a excesivo, ya que la tierra debe permanecer húmeda, pero sin enlagnar, el rábano se puede adaptar a un ambiente encerrado con unas 4 horas de sol hasta en un ambiente exterior en donde está expuesto a los diferentes cambios climáticos.

3. Metodología

3.1. Proceso de filtrado y equipo de captura de imágenes

Antes de empezar con la captura de imágenes se destinó un espacio único para ello, se tuvo en cuenta que hubiese una buena iluminación natural, además de disponer siempre la cámara para la toma de las imágenes, posicionándola a 45 centímetros de la maceta y a una altura del piso de 10 centímetros, en un fondo blanco; Las 12 plantas se colocaron en la misma posición, guiándose de una marca de referencia. Cada imagen fue guardada en Google drive (<https://n9.cl/registrotgg>)

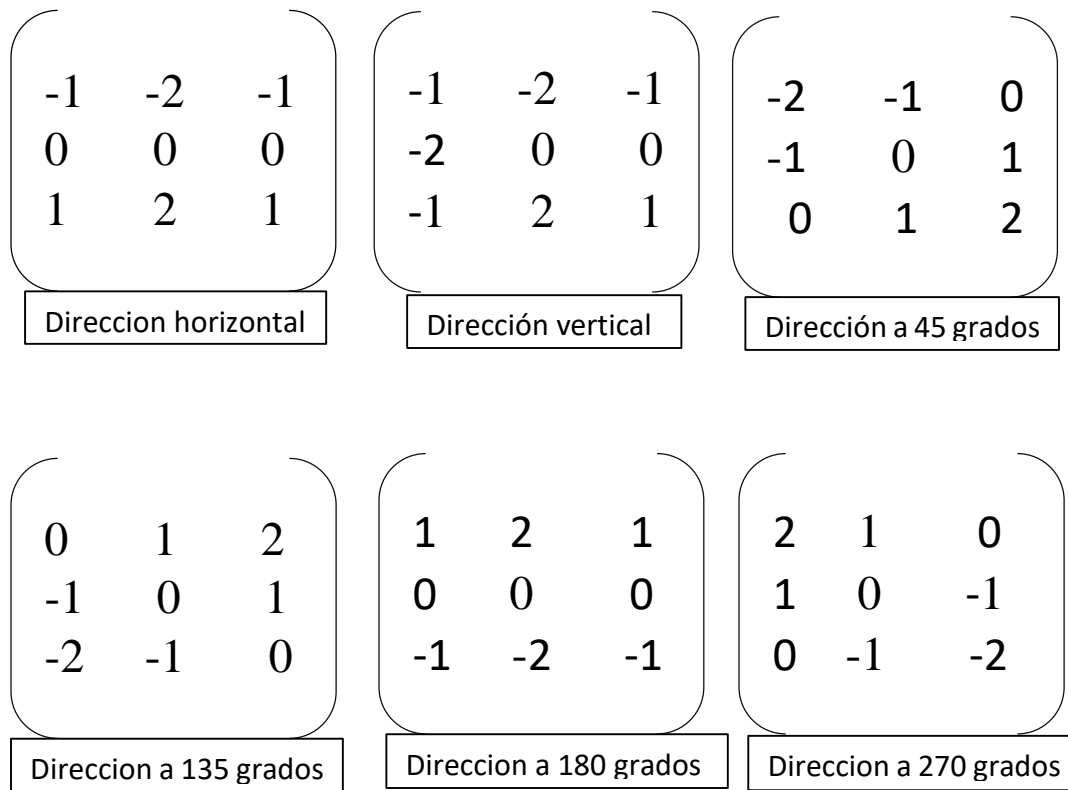
en una carpeta marcada con la fecha correspondiente al día de la toma de fotos, con el fin de obtener una base de datos sólida.

Para la fase de filtrado se decidió usar el filtro sobel ya que es un filtro activo que detecta y resalta los bordes presentes en la imagen, además se usó un filtro bilateral con el fin de lograr atenuar el ruido de entrada presente en las imágenes.

Estos fueron los kernels usados para la detección de bordes en el filtro sobel:

Figura 5

Kernels para el filtro sobel



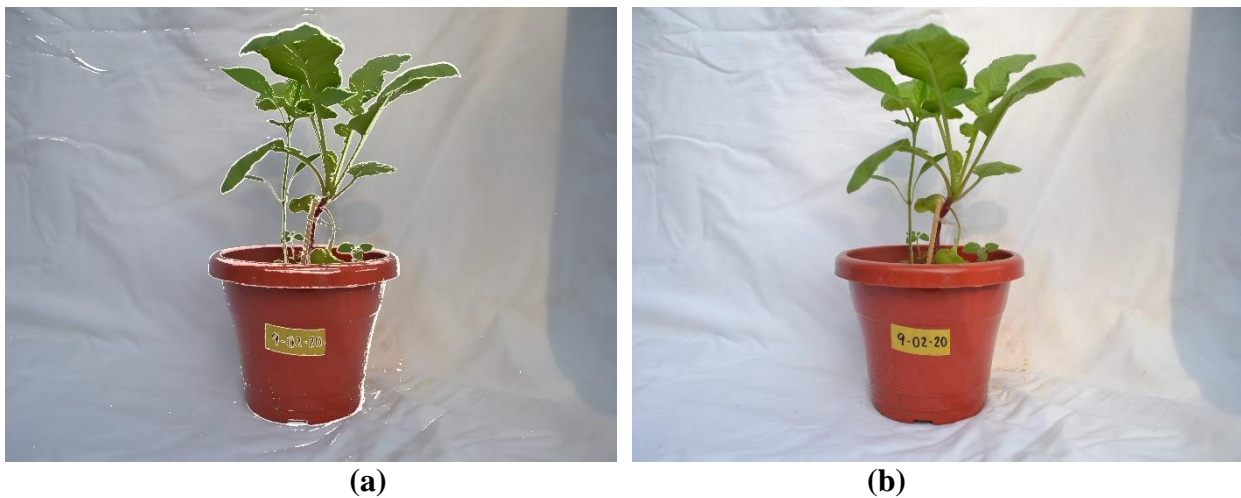
Primero se hizo el cargue de las imágenes desde la carpeta, para posteriormente aplicar el filtro bilateral, logrando una atenuación de ruido y conservación de los bordes, seguido a eso convirtió cada imagen filtrada bilateralmente a un formato de escala de grises pasando de un

formato BGR (Blue-Green-Red) a escala de grises, se puede observar en el siguiente repositorio guardado como filtros.py (https://github.com/juancarvallido/PROYECTO_DE_GRADO).

Después se pasó a convolucionar la imagen con cada kernel de la figura 5, luego se procedió a combinar la imagen resultante de salida con la imagen de entrada para así obtener una imagen resultante en formato BGR con realce en todos los bordes, mostrado en la Figura 6.

Figura 6

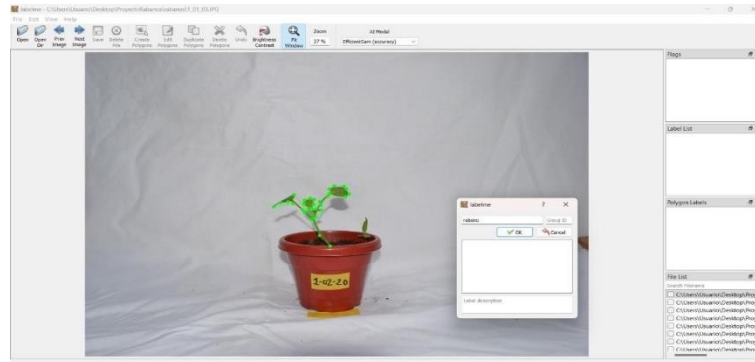
Comparación imagen filtrada vs original



Nota. Imagen (a) es la imagen resultante filtrada con realce y detección de bordes, imagen (b) corresponde a la imagen de entrada

3.2. Detección de la planta

En la segmentación de las plantas de rábano se creó una base de datos (<https://n9.cl/basedatostg>) con 400 fotografías de las plantas de rábano y se realizó mediante el uso de la herramienta de anotación de imágenes gráficas "labelme" para crear máscaras o contornos de las imágenes.

Figura 7*Herramienta de anotación de imágenes gráficas “Labelme”*

Tomado de: Elaboración propia.

Con el uso de la base datos del registro fotográfico tomado a cada una de las 12 plantas, esta herramienta (“labelme”) permitió delimitar manualmente las regiones de interés correspondientes a las plantas de rábano en cada imagen capturada. Durante este proceso, se identificaron los píxeles que representaban las plantas y se crearon máscaras o contornos que los rodeaban, lo que permitió la posterior utilización de estas regiones segmentadas para el entrenamiento del modelo de detección, en dicho entrenamiento se usaron 400 imágenes de las plantas de rábano en la carpeta de training (entrenamiento) y 20 imágenes en la carpeta de val (validación).

Una vez que se obtuvieron las máscaras o contornos de las imágenes utilizando labelme, se procedió al entrenamiento del modelo de red de YOLOv8l, el cual se tuvo en cuenta siguiendo las siguientes características:

Tabla 1

Modelos de entrenamiento

Modelo	Tamaño (Pixel)	mAPval 50-95	Velocidad CPU(ms)	Velocidad A100 (ms)	Parametros (M)	FLOPs (B)
YOLOv8n	640	37.3	80.4	0.99	3.2	8.7
YOLOv8s	640	44.9	128.4	1.20	11.2	28.6
YOLOv8m	640	50.2	234.7	1.83	25.9	78.9
YOLOv8l	640	52.9	375.2	2.39	43.7	165.2
YOLOv8x	640	53.9	479.1	3.53	68.2	257.8

Tomado de: Ultralytics. (2024, February 29). Detect. <https://docs.ultralytics.com/es/tasks/detect/>

Como se puede observar en la tabla 2 la métrica mAP (mean Average Precision) permite cuantificar la precisión de cualquier detector desarrollado por medio de redes neuronales, claro está que entre mayor sea esta métrica, mayor es el gasto computacional, debido a estos parámetros que se tuvieron en cuenta se decidió optar por el modelo de YOLOv8l, con base a que la CPU a utilizar cuenta con un procesador Intel(R) Core(TM) i5-9300H y con un tiempo de inferencia de datos de 104.2ms (milisegundos), esto hace referencia al tiempo que tarda el procesador en realizar una tarea, Se usó el entorno de desarrollo integrado PyCharm, concretamente en el lenguaje de programación Python, lo que permitió identificar automáticamente las plantas en nuevas imágenes. Este enfoque proporcionó una base sólida para la detección precisa de las plantas de rábano en el entorno controlado, facilitando el seguimiento del crecimiento de las plantas a lo largo del tiempo.

Las especificaciones usadas para el entrenamiento del modelo de YOLOv8 fueron:

Tabla 2

Parámetros de entrenamiento del modelo

Parametros	Valor
Dimensión de imagen	640
Época	100
Batch	2
Learning rate	0.01

Nota. Tomado de Ultralytics. (2024, March 30). Train.

<https://docs.ultralytics.com/es/modes/train/#train-settings>

3.3. Estimación de la altura

La estimación de la altura de las plantas de rábano se llevó a cabo considerando los puntos extremos dentro del contorno detectado de cada planta. Estos puntos extremos se identificaron como las coordenadas más altas y bajas dentro del perfil de la planta detectada en cada imagen.

Para lograr esto, primero se detectaron las plantas de rábano utilizando el modelo previamente entrenado. Una vez detectadas, se obtuvo el contorno de cada planta a partir de la detección. Luego se identificaron los puntos extremos dentro de este contorno, las coordenadas más altas y bajas en la dimensión vertical.

Es importante destacar que los puntos extremos se seleccionaron dentro del área delimitada por el contorno detectado de la planta, lo que proporcionó un punto de referencia para la estimación de la altura de la planta en la imagen. Esta metodología aseguró que la altura se calculara específicamente en relación con la planta detectada, evitando incluir áreas externas que no formaran parte de la planta misma.

Una vez identificados los puntos extremos, se calculó la distancia vertical entre ellos para determinar la altura de la planta en píxeles. Posteriormente, se realizó una conversión a unidades de longitud real (centímetros), utilizando una relación preestablecida entre píxeles y centímetros. Si bien se asumió una relación inicial fija, es importante tener en cuenta que esta relación puede variar según las características específicas de las imágenes y podría requerir ajustes adicionales para una precisión óptima.

Este enfoque proporcionó una forma óptima y consistente de medir la altura de las plantas de rábano en las imágenes capturadas, lo que facilitó el seguimiento del crecimiento de las plantas a lo largo del tiempo y la comparación con mediciones manuales realizadas de manera independiente.

3.3. Comparación del crecimiento

La comparación del crecimiento de las plantas de rábano se realizó mediante un enfoque integrado que combinó métodos manuales y automáticos. Utilizando un metro convencional, se realizaron mediciones directas de la altura de las plantas en centímetros.

Figura 8

Fotos de las medidas manuales



Nota. Evidencia de registro manual. Tomado de: Elaboración propia.

Esto proporcionó una referencia sólida para la validación de los resultados obtenidos por el algoritmo de estimación de altura basado en imágenes. Las mediciones manuales se llevaron a cabo de manera cuidadosa y sistemática, garantizando la precisión de los datos recopilados.

A partir del algoritmo de estimación se proporcionaron estimaciones automáticas de la altura de las plantas de rábano, estas estimaciones se compararon con las mediciones manuales para evaluar la precisión y la confiabilidad del enfoque de visión por computadora desarrollado.

4. Resultados

En esta sección se presentan todos los resultados obtenidos de la estimación del crecimiento y el entrenamiento del modelo, con sus respectivas pruebas y gráficas de los valores alcanzados en el entrenamiento.

4.1 Modelo YOLOv8l

Como resultado de los datos de entrenamiento del modelo observados en la tabla 1, el modelo entrenado hace unas validaciones de predicción, con las respectivas imágenes de validación.

Figura 9

Validación de etiquetas y predicción del modelo

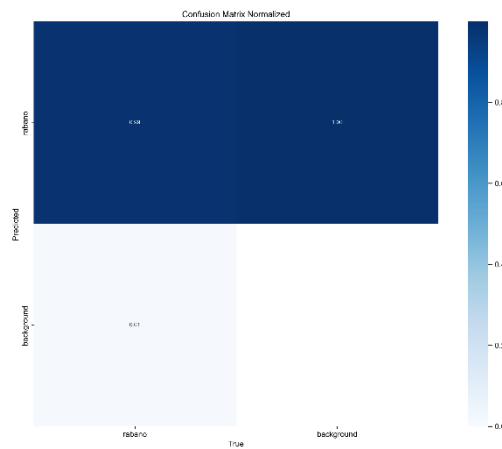


Nota. La figura de la izquierda muestra las etiquetas y en la figura de la derecha se observan las predicciones. *Tomado de:* Elaboración propia.

Estas predicciones, al igual que, el resultado de las validaciones se puede explicar mejor, desde la figura 10 hasta la figura 12, debido a que abarcaremos los temas del resultado de la matriz de confusión, curva de precisión, curva de recall y la pérdida de datos del modelo YOLOv8l.

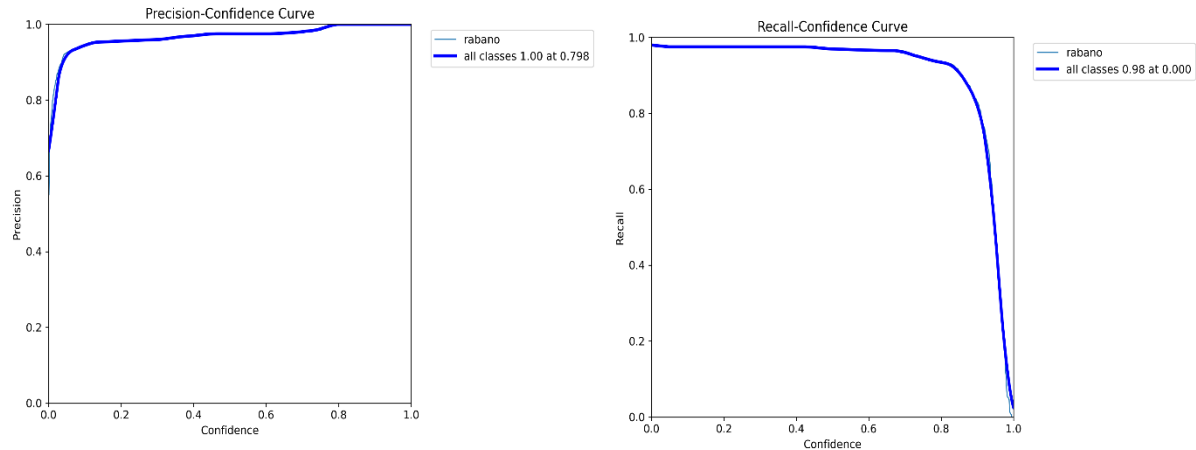
Figura 10

Matriz de confusión



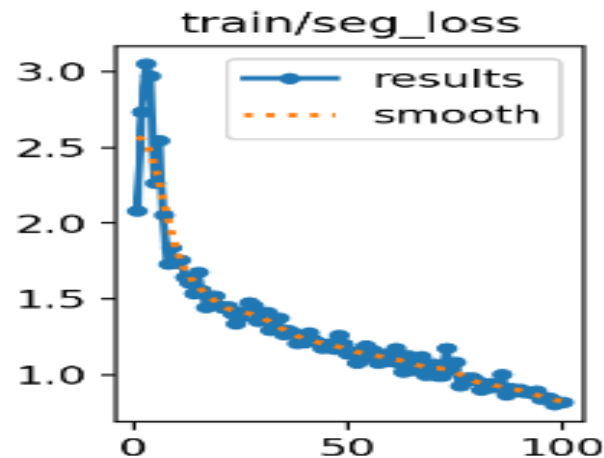
El modelo arrojó una precisión general del 99%, lo que indica que el 99% de los objetos fueron clasificados correctamente.

Algunos otros datos arrojados por el modelo que se pueden evidenciar en la figura 10 son los falsos positivos, hay un 1% de falsos positivos para la clase "rábano", lo que significa que el modelo clasificó erróneamente algunos objetos como rábanos cuando en realidad no lo eran.

Figura 11*Curva de precisión y recall*

Con estas graficas obtenidas del modelo podemos decir que el comportamiento de la precisión vs sensibilidad es bueno, ya que, la precisión nos da una medida de la calidad del modelo propuesto. Indica, que de todas las cajas que devuelve el sistema informático (es decir, de todos los objetos que encuentra), qué fracción de ellas realmente contienen el objeto a detectar. Su valor ideal sería la unidad.

Un detector lo consideramos bueno, si para cuando va aumentando el Recall la precisión se va manteniendo alta (Raneros, 2021), al analizar las dos graficas por ejemplo en el punto de confianza de 0.6 podemos observar como la precisión y el recall se van manteniendo altas acercándose a la unidad.

Figura 12*Perdidas de datos del modelo*

La gráfica muestra que la pérdida de entrenamiento disminuye a medida que se realizan más iteraciones de entrenamiento representadas en el eje x. Esto indica que el modelo está convergiendo a una solución óptima.

Tabla 3

Resultados de entrenamiento YOLOv8l

Información	Valor
Precisión de entrenamiento	0.988
Pérdida de entrenamiento	0.012
Tiempo de entrenamiento(min)	90

4.2 Estimación del crecimiento

Cada altura registrada del software fue obtenida dado el resultado del procesamiento de imágenes que se le hizo a cada planta, a continuación, se muestra el resultado del procesamiento realizado junto con la estimación de la altura por software.

Figura 13

Resultado de la estimación de la altura por software



Se introducirá los resultados de las correspondientes mediciones manuales y las adquiridas por el software

Figura 14

Comparación entre medidas manuales y las de software del primero de marzo

Dia 1 de marzo del 2024				
Planta	Medidas manuales (cm)	Medidas por software (cm)	Comparación (cm)	Desviación
1	7	7.87	0.87	0.37845
2	5	5.66	0.66	0.2178
3	5.5	6	0.5	0.125
4	7	6.95	0.05	0.00125
5	8	8.3	0.3	0.045
6	8.5	7.85	0.65	0.21125
7	7	7.17	0.17	0.01445
8	7.8	7.83	0.03	0.00045
9	7.5	7.9	0.4	0.08
10	4.5	4.51	0.01	5E-05
11	6	6.2	0.2	0.02
12	6.2	6.22	0.02	0.0002
Promedio				0.09115833

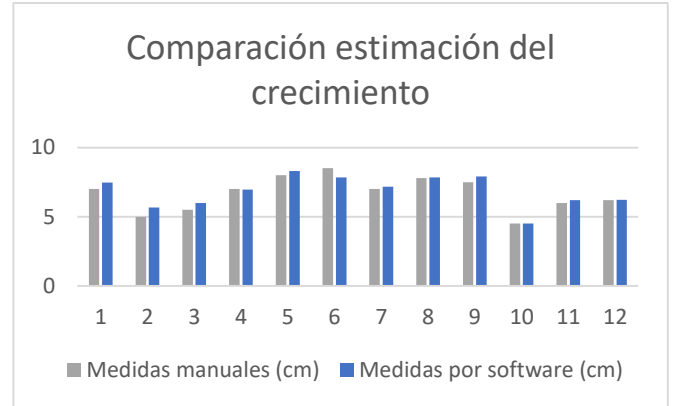


Figura 15

Comparación entre medidas manuales y las de software del nueve de marzo

Dia 9 de marzo del 2024				
Planta	Medidas manuales (cm)	Medidas por software (cm)	Comparación	Desviación
1	11	10.4	0.6	0.18
2	10.5	11.66	1.16	0.6728
3	10	9.5	0.5	0.125
4	10.5	9.75	0.75	0.28125
5	8.5	8.3	0.2	0.02
6	9.5	10.05	0.55	0.15125
7	9.4	9.69	0.29	0.04205
8	9.3	9.8	0.5	0.125
9	11.4	10.5	0.9	0.405
10	7.5	7.22	0.28	0.0392
11	9.8	9.2	0.6	0.18
12	8.9	9.91	1.01	0.51005
Promedio				0.22763333

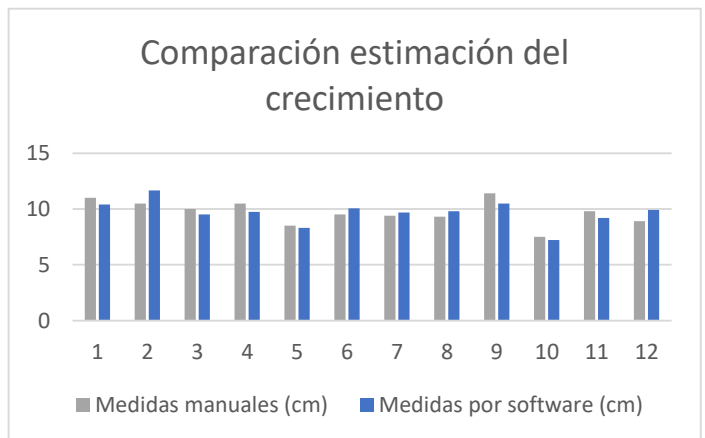
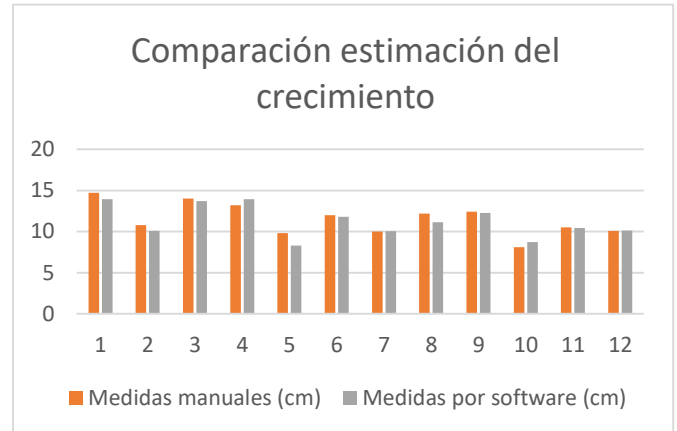


Figura 16

Comparación entre medidas manuales y las de software del quince de marzo

Dia 15 de marzo del 2024				
Planta	Medidas manuales (cm)	Medidas por software (cm)	Comparación (cm)	Desviación
1	14.7	13.93	0.77	0.29645
2	10.8	10.08	0.72	0.2592
3	14	13.7	0.3	0.045
4	13.2	13.92	0.72	0.2592
5	9.8	8.3	1.5	1.125
6	12	11.8	0.2	0.02
7	10	10.03	0.03	0.00045
8	12.2	11.12	1.08	0.5832
9	12.4	12.28	0.12	0.0072
10	8.1	8.74	0.64	0.2048
11	10.5	10.42	0.08	0.0032
12	10.1	10.13	0.03	0.00045
Promedio				0.23367917



Como se evidencian en las figuras 14, 15 y 16, se usó el método de desviación estándar como toma de referencia de la dispersión de los datos y evaluación de la estimación del crecimiento realizada por software vs la toma de datos manuales, al igual que una comparación, representada en centímetros de cada media.

5. Conclusiones

El enfoque desarrollado tiene el potencial de transformar la monitorización del crecimiento de las plantas, permitiendo una gestión agrícola más eficiente y precisa, además de ofrecer oportunidades significativas para avanzar en nuestro entendimiento de la biología vegetal y la sostenibilidad agrícola.

La segmentación manual de las plantas de rábano mediante la herramienta "labelme" demostró ser una estrategia eficaz para la creación de máscaras precisas, esenciales para el entrenamiento del modelo de detección.

Los resultados del modelo YOLOv8l reflejan una capacidad sobresaliente para detectar y clasificar las plantas de rábano, con una precisión general del 99% y una pérdida de entrenamiento mínima, lo que respalda su aplicabilidad en entornos agrícolas y de investigación científica.

La combinación de métodos automáticos y manuales para la medición de la altura permitió una validación rigurosa de los resultados obtenidos por el algoritmo de visión por computadora, garantizando la precisión y confiabilidad del enfoque propuesto.

6. Recomendaciones

Como continuación al trabajo de grado enfocado en la estimación del crecimiento mediante visión artificial, se tiene en cuenta que corresponde a un proyecto de investigación donde convergen diferentes áreas del conocimiento en el cual hay posibilidades de prolongar el estudio realizado como trabajo futuro a otros investigadores.

Primero se sugiere incluir diferentes clases de plantas, con el fin de llevar la estimación del crecimiento a diferentes plantas que se tenga interés, incluir la medición del ancho y del área.

Segundo, se puede plantear que los datos obtenidos de las mediciones por procesamiento de imagen podrían llevarse a cabo en una base de datos para posibles consultas y generación de reportes de producción en el cultivo.

Se puede ampliar y mejorar el modelo para que detecte posibles afectaciones, plagas y mal estado de las plantas en el cultivo, además de integrar un sistema embebido con el fin de tener un sistema más robusto

Referencias Bibliográficas

- Adnan, N., Nordin Shahrina, M., Rahman, I., & Noor, A. (2017). Los efectos de la transferencia de conocimientos en la toma de decisiones de los agricultores hacia prácticas agrícolas sostenibles. *World Journal of Science, Technology, and Sustainable Development*.
- Ariza, A. F. (2022). Análisis de crecimiento destructivo y no destructivo de dos materiales de lulo (*Solanum quitoense*) La Selva y de Castilla en Aguabonita, Cundinamarca (Tesis de título universitario, Universidad Pontificia Javeriana).
- Cai, J., & Walker, R. (2010). Height estimation from monocular image sequences using dynamic programming with explicit occlusions. *IET Computer Vision*, 4(3), 149-161.
- Cano Avalos, J. (2022). Introducción al Deep Learning aplicación en R [Tesis de grado en estadística, Universidad de Sevilla]. <https://idus.us.es/bitstream/handle/11441/142917/GE%20CANO%20AVALOS%2C%20JAVIER.pdf>
- Chanampe, H., Aciar, S., Martín de la Vega, & Molinari Sotomayor, J. L. (s. f.). Modelo de redes neuronales convolucionales profundas para la clasificación de lesiones en ecografías mamarias [Trabajo de investigación, Universidad Nacional de La Rioja]. <https://core.ac.uk/download/pdf/296416305.pdf>
- Enrique, A. (2018, September 9). Detección de objetos con YOLO: implementaciones y como usarlas. Medium. <https://medium.com/@enriqueav/detecci%C3%B3n-de-objetos-con-yolo-implementaciones-y-como-usarlas-c73ca2489246>
- Fontalvo, J. (2021, 7 julio). Universidad veracruzana [Diapositivas]. Proyecto COV-AID. <https://www.uv.mx/hab/files/2021/10/Cultivo-de-Rabano.pdf>

Getting Started with YOLO v2 - MATLAB & Simulink. (s. f.).

<https://www.mathworks.com/help/vision/ug/getting-started-with-yolo-v2.html>

Gómez Flórez, W. (2020). *Análisis de imágenes digitales filtros espaciales suavizantes*

[Diapositivas]. <https://www.tamps.cinvestav.mx/~wgomez/diapositivas/AID/Clase04.pdf>

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Aprendizaje residual profundo para el reconocimiento de imágenes. En Actas de la conferencia IEEE sobre visión por computadora y reconocimiento de patrones (pp. 770–778).

INFOAGRO (Información de la Agricultura). 2009. El cultivo del rábano (en línea). Consultado 24 agosto. 2009. Disponible en <http://www.infoagro.com/hortalizas/rabano.asp>

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Clasificación de ImageNet con redes neuronales convolucionales profundas. En Avances en los sistemas de procesamiento de información neuronal (pp. 1097–1105).

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Aprendizaje basado en gradientes aplicado al reconocimiento de documentos. *Procesamiento IEEE*, 86(11), 2278–324.

Nielsen, M. A. (2015). *Neural networks and deep learning*. Determination Press, San Francisco

Numerische Mathematik. (1994). A geometric model for active contours in image processing. *Numerische Mathematik*, 67(1), 10-11.

Olabe, X. B. (1998). *Redes neuronales artificiales y sus aplicaciones*. Publicaciones de la Escuela de Ingenieros.

Penatti, O. A. B., Nogueira, K., & Dos Santos, J. A. (2015). Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? En IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (pp. 44–51).

¿Qué es Deep Learning? | IBM. (n.d.). <https://www.ibm.com/es-es/topics/deep-learning>

- Razavian, A. S., Azizpour, H., Sullivan, J., & Carlsson, S. (2014). CNN features off-the-shelf: An astounding baseline for recognition. En IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (pp. 512–519).
- Rico-Fernández, M. P., Rios-Cabrera, R., Castelán, M., Guerrero-Reyes, H., & Juárez-Maldonado, A. (2019). A contextualized approach for segmentation of foliage in different crop species. *Computers and Electronics in Agriculture*, 156, 378–386. <https://doi.org/10.1016/j.compag.2018.11.033>
- Rovira-Más, F., Zhang, Q., & Reid, J. F. (2008). Stereovision three-dimensional terrain maps for precision agriculture. *Computers and Electronics in Agriculture*, 60(2), 133-143.
- Raneros, S. R. (2021). *Estudio de la arquitectura YOLO para la detección de objetos mediante deep learning* [TRABAJO FIN DE MÁSTER, UNIVERSIDAD DE VALLADOLID]. <https://uvadoc.uva.es/bitstream/handle/10324/45359/TFMG1316.pdf?sequence=1&isAllowed=y>
- Supeshala, C. (2021, 15 diciembre). YOLO V4 or YOLO V5 or PP-YOLO? Which should I use? | Towards data science. Medium. <https://towardsdatascience.com/yolo-v4-or-yolo-v5-orpp-yolo-dad8e40f7109>
- Sharma, R., Kamble, S. S., Gunasekaran, A., Kumar, V., & Kumar, A. (2020). Una revisión sistemática de la literatura sobre aplicaciones de aprendizaje automático para el desempeño de la cadena de suministro agrícola sostenible. *Research in Computer Science*, 119, páginas.(Sharma, R. et al.,2020)
- Tabares, M. (2022). *PhytoVision: Sistema de medición de crecimiento in vitro de plántulas de Lactuca sativa L.* (Tesis de pregrado, Universidad Tecnológica de Pereira). Recuperado de

<https://repositorio.utp.edu.co/server/api/core/bitstreams/e3d64b6b-7749-4fce-a53f-1d6962658501/content>

Ultralytics. (2024, February 29). *Detect*. <https://docs.ultralytics.com/es/tasks/detect/>

Zhang, F., Fu, L.S. (2014). Application of computer vision technology in agricultural field. In *Applied Mechanics and Materials* (Vol. 462, pp. 72-76).

Zhang, C.-C., & Fang, J.-D. (Eds.). (2016). Edge detection based on improved SOBEL operator (Vol. 52). Published by Atlantis Press. <https://doi.org/10.2991/ceis-16.2016.25>

Función de convolución—ArcMap | Documentación. (n.d.). <https://desktop.arcgis.com/es/arcmap/latest/manage-data/raster-and-images/convolution-function.htm>