Universidad Internacional de Valencia

Desarrollo de un sistema de diagnóstico de enfermedades en hojas de tomate mediante modelos de aprendizaje profundo

Titulación:
Máster en Big Data y Ciencia de Datos
Curso Académico 2024-2025

Alumno/a: Marín Lucas, Rubén
DNI: 07272889-J
Director/a del TFT:
Ricardo Lebrón Aguilar

Convocatoria:

SEGUNDA

De:
Planeta Formación y Universidades

Universidad Internacional de Valencia

Índice general

Índice de figuras

Índice de cuadros

Lorem ipsum (RESUMEN)

Palabras clave: primero, segundo, tercero

Agradecimientos

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

1. Introducción

1.1. Motivación

Tomate o tomatera (*Solanum lycopersicum*) es una planta herbácea de la familia Solanaceae cultivada en todo el mundo para el cultivo de su fruto, el tomate o jitomate, uno de los ingredientes más universales de ensaladas y salsas en el mundo entero. (Wikipedia contributors, 2025b)

Según los últimos estudios filogenéticos, la planta silvestre de la cual surge el tomate doméstico actual tiene origen en la zona andina del norte de Perú y sur de Ecuador. Su domesticación y diversificación posterior se originó en México.

Los pueblos aztecas y mayas lo usaban en su cocina y fue exportado al resto del mundo a partir de la llegada de los españoles que lo distribuyeron a lo largo de sus colonias en el Caribe y la península ibérica a partir de lo cual pudo llegar al resto de eruopa. También lo llevaron a Filipinas y de allí pudo entrar al continente asiático. (Ing. Agr. Miguel Silva, 2025a)

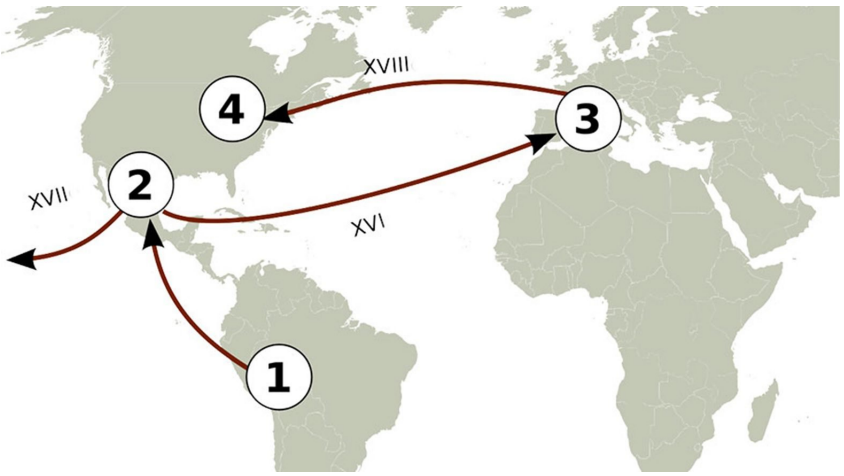

Figura 1.1.: Origen del tomate

La producción mundial de tomate ascendió a más de 186 millones de toneladas en 2022 según los datos de la Organización de las Naciones Unidas para la Alimentación y la Agricultura (FAO). Según esta misma organización esta es la evolución de los 20 países que más han producido hasta 2022: (Wikipedia contributors, 2025a)

Como ya se ha mencionado, el cultivo de tomate es uno de los cultivos hortícolas más importantes a nivel mundial. Sin embargo, su producción se ve amenazada

Cuadro 1.1.: Top 20 países productores de tomates 2022

Titulo				
País	2000	2010	2020	2022
China	22 200	46 760	64 680	68 242
...	...	...	...	...

por una amplia variedad de enfermedades causadas por hongos, bacterias, virus y nematodos. Estas enfermedades pueden provocar una bajada de rendimiento que van desde reducciones parciales hasta la pérdida completa de la cosecha.

Entre las enfermedades más comunes se encuentran:

- Tizón tardío (*Phytophthora infestans*): Puede destruir por completo una plantación si no se controla a tiempo, especialmente en condiciones húmedas y templadas.

Figura 1.2.: Tizón tardío en una planta de tomate

- Tizón temprano (*Alternaria solani*): Produce defoliación progresiva, debilitando la planta y reduciendo el número y calidad de los frutos.
- Fusariosis vascular (*Fusarium oxysporum*): Ataca el sistema vascular, provocando marchitez y muerte de plantas.
- Virus como TYLCV y TSWV: Pueden causar deformaciones severas y reducciones completas en la producción, especialmente cuando se transmiten por vectores como la mosca blanca.

La manifestación simultánea o sucesiva de estas enfermedades es una de las principales causas en la disminución en la productividad del cultivo a escala global.

Figura 1.3.: Tizón temprano en una planta de tomate

Además, muchas de estas enfermedades no solo viven en la planta sino que persisten en el suelo, semillas o herramientas que hayan inteactuado con la planta, lo que dificulta su erradicación y aumenta los costos del tratamiento. (Ing. Agr. Miguel Silva, 2025b)

Dada la magnitud del impacto de estas enfermedades, la detección temprana y precisa de las mismas es crucial. Permite una correcta intervención que minimiza las pérdidas, permitiendo la reducción del uso innecesario de los agroquímicos y mejorando la sostenibilidad. En este contexto, las tecnologías basadas en visión por computadora, sensores remotos e inteligencia artificial ofrecen soluciones eficaces para mejorar el seguimiento y el control sanitario de este cultivo clave.

1.2. Estructura del resto del documento

La documentación de este proyecto se ha desarrollado dividiendo el contenidoen distintos caítulos con el objetivo de facilitar la comprensión por parte del lector. A continuación se enuncia la información presente en cada capítulo del resto del documento:

- Capítulo 2. Objetivos: se describe el objetivo principal del proyecto junto con los objetivos intermedios necesarios para conseguirlo.
- Capítulo 3. Estado del Arte:
- Capítulo 4. Implementación y desarrollo: se explica el preprocesamientode los datos y las técnicas ML aplicadas a los mismos.
- Capítulo 5. Evaluación y resultados: se presentan los resultados obtenidos al aplicar las técnicas ML.
- Capítulo 6. Conclusiones

2. Objetivos

El objetivo general de este proyecto consiste en conseguir un clasificador que a partir de imágenes de hojas de plantas de tomate distinga entre estado saludable y 10 enfermedades distintas.

2.1. Objetivos específicos

1. Analizar dataset de hojas de tomate.
2. Implementar y entrenar modelos CNN para la clasificación.
3. Evaluar la precisión de los modelos y comparar resultados.

3. Estado del arte

En los últimos años la aplicación de técnicas de inteligencia artificial en la agricultura ha cobrado un papel relevante, especialmente en tareas de diagnóstico temprano de enfermedades en cultivos. El uso de aprendizaje profundo permite automatizar la detección de patrones en imágenes, lo cual puede ayudar a los agricultores a tomar decisiones más rápidas y eficientes.

Inicialmente, los métodos empleados para esta tarea incluían algoritmos de aprendizaje supervisado como máquinas de vectores de soporte (SVM), kvecinos más cercanos (KNN) y redes bayesianas. Sin embargo, estos enfoques dependían en gran medida de una segmentación previa precisa y de la extracción manual de características, lo que limitaba su capacidad de generalización y precisión en entornos reales.

Con la llegada de las redes neuronales convolucionales (CNN), se ha producido un cambio significativo en la forma de abordar este problema. Las CNN son capaces de aprender representaciones directamente a partir de los datos de imagen, eliminando la necesidad de ingeniería manual de características. Diversos estudios han demostrado su eficacia para la clasificación de enfermedades en hojas de tomate.

Por ejemplo, una revisión publicada en la Revista de Investigación e Innovación de las Ciencias de la Universidad Tecnológica de Bolívar (Martínez et al., 2024), las técnicas tradicionales de aprendizaje supervisado como SVM, KNN y lógica difusa muestran limitaciones significativas en tareas de detección de enfermedades en imágenes de frutas debido a su dependencia de extracción manual de características y segmentación previa. En contraste, las redes neuronales convolucionales han demostrado una precisión superior, mayor robustez frente a la variabilidad y mayor capacidad de generalización. Esta revisión respalda la elección de CNNs como enfoque principal en este trabajo.

Por otra parte, Valeria Maeda Gutiérrez (2019) (Gutiérrez, 2019) realizó una comparativa entre varias arquitecturas CNN, incluyendo AlexNet, GoogleNet, InceptionV3, ResNet 18 y ResNet 50 aplicadas al conjunto de datos PlantVillage. Todas las arquitecturas consiguieron más del 98 % de precisión y sensibilidad, lo que confirma la ideonidad de las mismas para la tarea que se pretende hacer. Concretamente con GoogleNet consiguió una precisión del 99,3 % y una sensibilidad del 99,1 %

En otra línea, Eduardo A.Huerta-Mora, Víctor González-Huitrón, Héctor Rodríguez-Rangel y Leonel Ernesto Amabilis-Sosa (2024) (A.Huerta-Mora et al., 2024) emplearon la arquitectura VGG16 con técnicas de fine-tuning para el mismo conjunto de datos PlantVillage, obteniendo alrededor del 90 % de sensibilidad y precisión. Este hecho confirma que esta arquitectura también puede ser interesante para el estudia a realizar.

4. Implementación y desarrollo

En este capítulo se presenta tanto el *hardware* como el *software* usados en este proyecto. Además se explica la procedencia y estructura del conjunto de datos que serán usados para el estudio. Finalmente, se desarrolla el preprocesmiento que se realiza a este conjunto de datos junto con los modelos entrenados para conseguir un clasificador.

4.1. Herramientas usadas

Para llevar a cabo este proyecto, se ha usado Google Colab (abreviatura de Google Colaboratory) que se accedía desde el ordenador portátil del autor del documento. Este ordenador es un ASUS TUF Gaming FX505GT que cuenta con las siguientes características:

- 16 GB de RAM con formato DDR4.
- Almacenamiento compuesto por un disco duro con tecnología SSD de 512GB.
- Procesador Intel Core i7-9750H CPU a 2.60 GHz, con 6 procesadores principales y 6 procesadores lógicos.
- Tarjeta gráfica NVIDIA GeForce GTX 1650 con 4GB de RAM.

Google Colab es un servicio gratuito de Google que permite escribir y ejecutar código en la nube sin necesidad de instalar nada en tu equipo. Los recursos que ofrece de forma gratuita varían con el tiempo, pero las características que suele ofrecer son las siguientes:

- GPU NVIDIA Tesla T4 con 16 GB de VRAM ó CPU Intel Xeon con alrededor de 13 GB de RAM.
- Almacenamiento temporal se corresponde con unos 100 GB de espacio en disco.
- La duración de la sesión puede ser de hasta 12 horas, aunque en la práctica podrían terminarse antes según uso y carga del sistema.

Por otra parte el lenguaje de programación usado ha sido Python, un lenguaje que es ampliamente utilizado por científicos de datos. En las últimas décadas Python se ha enriquecido con numerosas librerías relacionadas con técnicas de ML que

facilitan el uso de las mismas. En concreto para este proyecto se ha utilizado la versión 3.12.11 de Python.

En cuanto a las librerías de Python usadas para la implementación, se presentan a continuación:

- NumPy: es una librería que ofrece la posibilidad de crear matrices y vectores multidimensionales y provee además un gran número de operaciones matemáticas de alto nivel.
- Pandas: es una librería que ofrece la estructura de datos llamada DataFrame que facilita la manipulación y el análisis de datos. Es una extensión de la librería NumPy. Ha sido usada para tratar y transformar los datos.
- Plantcv: es una librería de Python de código abierto diseñada específicamente para el análisis de imágenes de plantas. Se ha usado para lectura de las imágenes.
- Tensorflow: es una librería de software de código abierto creada por Google para desarrollar y entrenar modelos de machine learning (ML) y deep learning (DL). Recibe este nombre porque trabaja con tensores, estructuras de datos multidimensionales, como matrices o vectores que fluyen a través de un grafo computacional de operaciones. Se ha usado para crear y entrenar los modelos descritos en este proyecto.
- Seaborn: esta librería permite la visualización de los datos a través de distintos tipos de gráficas. Ha sido usada para realizar los distintos gráficos como las matrices de confusión para evaluar los modelos.

4.2. Procedencia y descripción de los datos

Los datos provienen de la plataforma online de ciencia de datos de Google, Kaggle, que funciona como una mezcla de red social, repositorio de datasets y espacio de competición. En concreto, el conjunto de datos usuado es el llamado "Tomato Leaves Dataset"(***Enlace). Según su descripción en la misma plataforma se trata de un conjunto de datos de más de 20.000 imágenes de hojas de tomate con 11 clases, 10 enfermedades y una clase sana. Estas imágenes se han recopilado tanto en entornos de laboratorio como en entornos naturales.

En concreto se pueden extraer dos directorios que servirán como conjunto de datos para el entrenamiento y conjunto de datos para validación, ambos cuentan con 11 directorios con imágenes dentro. Cada uno de estos subdirectorios representa una de las clases que serán brevemente expuestas a continuación:

Cuadro 4.1.: Clases del conjunto de datos

Conjunto de datos	
Clase	Traducción Clase
Healthy	Saludable
Bacterial_spot	Manchas bacterianas
Early_blight	Tizón precoz
Late_blight	Tizón tardío
Leaf_Mold	Hojas con moho
Powdery_mildew	Moho polvoriento
Septoria_leaf_spot	Hojas manchadas de septoriosis
Spidermite_Two-spotted_spider_mite	Picadura de araña roja de dos manchas
Target_Spot	Punto blanco
Tomato_mosaic_virus	Virus mosaico
Tomato_Yellow_Leaf_Curl_Virus	Virus de la hoja amarilla

4.3. Preprocesado de los datos

Al realizar la carga de datos se tiene un directorio con dos subdirectorios, cada uno de ellos representará un conjunto de datos, uno de datos de entrenamiento (train) y otro de validación (valid). Cada uno de estos directorios contienen a su vez 11 directorios, representando cada una de las clases. Llegados a este punto, se llevan a cabo las primera tareas de exploración de las imágenes. En primer lugar, seleccionando el directorio que contiene los datos de entrenamiento se realiza una función para obtener el número de imágenes existentes por cada tipo de tamaño. Gracias a esta función se sabe que se cuentan con imágenes de variables tamaños, concretamente existen 760 tipos de tamaños distintos. En la siguiente tabla se exponen los 5 tipos de tamaño que más se repiten:

Cuadro 4.2.: Los 5 tamaños de imágenes más comunes

Top 5 tamaños de imágenes	
Tamaño (píxeles)	Nº imágenes
256x256	18942
227x227	4120
640x640	1207
533x800	151
800x600	10

De cara a construir un modelo todas las imágenes tienen que tener el mismo tamaño y debida a esta primera toma de contacto se toma la decisión de transformar todas las imágenes a tamaño de 256x256 píxeles.

En segundo lugar, se realiza una muestra aleatoria de una imagen por clase, para visualizar el tipo de imágenes que se vana tratar.

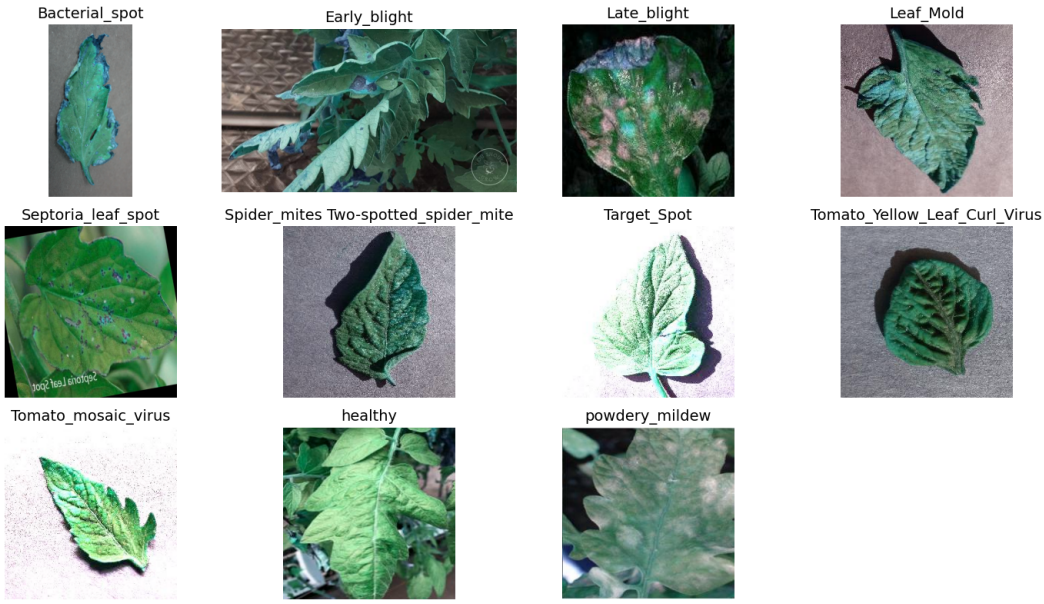

Figura 4.1.: Imagen aleatoria por clase

Se puede observar la gran variedad de imágenes que existen, con distintos brillos, distintos fondos o incluso giradas:

- En cuanto al brillo se puede observar que la imagen de `Target_Spot` tiene mucho más brillo que la de `Tomato_Yellow_Leaf_Curl_Virus`.
- Con respecto al fondo podemos vir distinciones entre la imagen `Bacterial_spot` con un fondo grisáceo plano, la imagen `Late_blight` con fondo completamente negro y la foto `Early_blight` en la que se ve el resto de la planta de tomate, no solo se ve una hoja.
- También se puede destacar la diferencia entre las posiciones de las hojas, algunas como `Bacterial_spot` tienen el tallo abajo, otras como `Leaf_Mold` tienen el tallo arriba y otras como `Septoria_leaf_spot` tienen el tallo horizontal.
- Además se tiene un ejemplo de imagen rotada, concretamente la `Septoria_leaf_spot`

Debido a esta gran variedad de imágenes y de enfermedades se llegó a la conclusión de que no tiene mucho sentido usar funciones especiales de la librería PlantCV, ya que esta está muy orientada al fenotipado clásico (área, forma, color, índices), y esas características a veces no capturan la complejidad de patrones de enfermedades, que suelen ser más sutiles y no lineales. La estrategia que se ha seguido es pasar a un pipeline de deep learning con imágenes preprocesadas de forma estándar. De esta manera el modelo aprenderá por si mismo las características discriminantes en lugar de imponer un conjunto de "features"manuales.

A continuación se dispuso a formar los conjuntos de datos que usará el modelo. Hasta ahora se tienen datos para el entrenamiento del modelo y para la validación del mismo, sin embargo, no se tienen datos para realizar pruebas sobre el modelo resultante. Por lo tanto, se formará un nuevo conjunto de datos de pruebas a partir del cunjunto de entrenamiento, concretamente, seleccionando un 20 % de sus datos.

Para realizar esta tarea se recorre el directorio de datos de entrenamiento y aleatoriamente se seleccionan imágenes de cada subdirectorio (clase) y se añaden a un nuevo directorio que será el de datos de prueba.

Otra tarea importante en cuanto al procesado de los datos es normalizar los mismos. Para ello, se hace uso de ImageDataGenerator de tensorflow. Se usa esta función porque también sirve para aplicar la técnica de data augmentation en los datos de entrenamiento que ayuda a que el modelo resultante del entrenamiento generalice mejor.

Normalizar en valid y train significa...

Train además se le aplica: lista

De esta manera, quedan los siguientes conjuntos de datos: Entrenamiento: Hay 20.686 imágenes utilizadas para entrenar el modelo. Se trata de la mayor parte de los datos, ya que el modelo necesita muchos ejemplos para aprender patrones. Validación: Hay 6.683 imágenes para la validación. Estos datos se utilizan para evaluar el rendimiento del modelo durante el entrenamiento, sin afectar a los parámetros del modelo. Prueba: Hay 5.165 imágenes para la prueba final. Este conjunto de datos se utiliza una vez finalizado el entrenamiento para medir objetivamente el rendimiento del modelo con datos nuevos que nunca se han visto.

Otros datos importantes del conjunto de entrenamiento...

Número de imágenes por clases que se tiene en el conjunto de entrenamiento

Clase 'Bacterial_spot': 2826 imágenes Clase 'Tomato_Yellow_Leaf_Curl_Virus': 2039 imágenes Clase 'Target_Spot': 1827 imágenes Clase 'Leaf_Mold': 2754 imágenes Clase 'Early_blight': 2455 imágenes Clase 'Late_blight': 3113 imágenes Clase 'Spidermite_Two-spotted_spider_mite': 1747 imágenes Clase 'powdery_mildew': 1004 imágenes Clase 'healthy': 3051 imágenes Clase 'Tomato_mosaic_virus': 2153 imágenes Clase 'Septoria_leaf_spot': 2882 imágenes

Eso nos llevará a usar class weights en el modelo

Por otra parte para mayor robustez se usará data augmentation

4.4. Modelado

Explicación de la estructura común de modelado como los callbacks, earlystopping y demás. Breve revisión de CNN en un párrafo Breve resumen de los distintos modelos, Inception, Resnet...

5. Evaluación y resultados

Secciones con cada modelo concreto y sus parámetros y los resultados mostrados con las métricas y una matriz de confusión

6. Conclusiones

Se selecciona el mejor modelo y se muestra una gráfica probando el modelo (desarrollar)

A. Anexo I: Ejemplo de anexo

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

1. Primer elemento.
2. Segundo elemento
3. Tercer elemento.
    a) Primer subelemento.
    b) Segundo subelemento.
        - Primer punto.
        - Segundo punto.

B. Anexo II: Otro ejemplo de anexo

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Bibliografía

A.Huerta-Mora, E., González-Huitrón, V., Rodríguez-Rangel, H., & Amabilis-Sosa, L. E. (2024). Detección de enfermedades foliares conarquitecturas de redes neuronales convolucionales [Consultado el 30 de julio de 2025]. https://rinderesu.com/index.php/rinderesu/article/view/46/50

Gutiérrez, V. M. (2019). Comparación de arquitecturas de redes neuronales convolucionales para la clasificación de enfermedades en tomate [Consultado el 30 de julio de 2025]. https://sedici.unlp.edu.ar/handle/10915/139770

Ing. Agr. Miguel Silva. (2025a, julio). Cultivo de tomate: Cómo se realiza, plagas e importancia. *Agrotendencia TV*. Consultado el 28 de julio de 2025, desde https://agrotendencia.tv/agricultura/cultivos/hortalizas/el-cultivo-de-tomate/#Historia_del_tomate_o_jitomate

Ing. Agr. Miguel Silva. (2025b, julio). Cultivo de tomate: Cómo se realiza, plagas e importancia. *Agrotendencia TV*. Consultado el 28 de julio de 2025, desde https://agrotendencia.tv/agricultura/cultivos/hortalizas/el-cultivo-de-tomate/#Historia_del_tomate_o_jitomate

Martínez, M. Y., Molina, M. M., García, N. M., & López, E. V. (2024). *Técnicas de aprendizaje supervisado para la detección y clasificación de enfermedades y defectos en imágenes de frutas: revisión* [Consultado el 30 de julio de 2025]. https://revistas.utb.edu.ec/index.php/magazine/article/view/2330/1983

Wikipedia contributors. (2025a, julio). Producción mundial del tomate. *Wikipedia*. Consultado el 28 de julio de 2025, desde https://es.wikipedia.org/wiki/Solanum_lycopersicum

Wikipedia contributors. (2025b, julio). *Solanum lycopersicum*. Wikipedia. Consultado el 28 de julio de 2025, desde https://es.wikipedia.org/wiki/Solanum_lycopersicum