

Evaluación comparativa de arquitecturas convolucionales para el diagnóstico de enfermedades en cultivos de hortalizas en invernadero en dispositivos de bajo costo

Jimenez A. Manuela, Cardona J. Santiago, Cañón D. Juan Jose Gabriel y Salazar. Juan Camilo

Abstract—Con el avance de las tecnologías los sectores como la agricultura se empiezan a ver beneficiados para poder darle solución a problemáticas constantes, como lo es la detección temprana de enfermedades de cultivos en invernaderos, esta solución se ha convertido importante ya que, siempre se ha dependido de la inspección visual por parte de expertos agrónomos, un proceso que resulta subjetivo, costoso y difícil de escalar en entornos productivos extensos. El desarrollo de técnicas de visión por computador y aprendizaje profundo ha permitido automatizar este tipo de diagnóstico, ofreciendo herramientas más rápidas, objetivas y reproducibles.

Dentro de estas técnicas, las redes neuronales convolucionales (CNN) han mostrado una notable capacidad para extraer características visuales complejas —como color, textura y morfología— directamente desde las imágenes, sin requerir etapas manuales de segmentación o ingeniería de características. Sin embargo implementar estos modelos actuales generan unos desafíos y limitaciones que requieren de una unidad de procesamiento gráfico GPU bien consolidada. Por ende es necesario identificar arquitecturas que no requieran tanto consumo de máquina, y mantenga el equilibrio entre precisión y eficiencia computacional. Este trabajo busca abordar la problemática mediante la comparación de diferentes configuraciones arquitectónicas de modelos CNN, evaluando su desempeño en términos de precisión, tiempo de inferencia y tamaño de modelo bajo condiciones controladas.

El artículo se enfoca en cultivos de tomate, utilizando el conjunto de datos *PlantVillage*. Se experimenta con variantes arquitectónicas que incluyen cambios en la profundidad, densidad de capas, tipo de convolución y normalización por lotes.

Los resultados permiten identificar configuraciones que ofrecen un desempeño competitivo con una fracción del costo computacional, evidenciando la viabilidad de implementar modelos CNN ligeros en sistemas embebidos para agricultura de precisión.

Index Terms—Aprendizaje profundo, Redes neuronales convolucionales (CNN), Diagnóstico foliar, Agricultura de precisión, *PlantVillage*, Dispositivos de bajo costo.

I. ESTADO DEL ARTE

Las enfermedades de las plantas representan una de las principales amenazas para la seguridad alimentaria mundial. Se estima que provocan aproximadamente un 16 % de pérdida del rendimiento global de los cultivos [4].

A. Deep Learning

La evolución de los enfoques de visión por computadora en agricultura ha seguido un recorrido progresivo desde las redes neuronales convolucionales (CNN) hacia los más recientes

transformadores de visión (ViT). Las CNN se consolidaron como la arquitectura dominante para la clasificación de hojas y lesiones debido a su alta precisión y eficiencia computacional. Por ejemplo, Afifi *et al.* emplearon las arquitecturas ResNet-18, ResNet-34 y ResNet-50 para identificar enfermedades en hojas de café utilizando los conjuntos de datos *PlantVillage* y *Coffee Leaf*, alcanzando una precisión del 99 % [6]. Otros trabajos han extendido el uso del aprendizaje profundo para la estimación de severidad del daño foliar, logrando cuantificar el estrés biótico en hojas de café con alta correlación entre predicción y daño real [6].

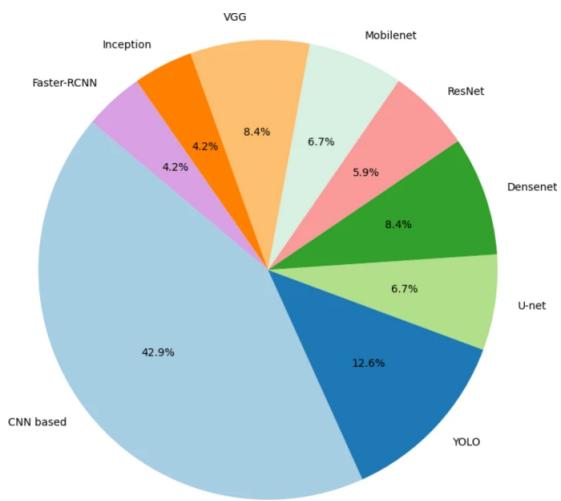


Fig. 1. Algoritmos usados en estudios para detectar enfermedades. Sacado de [6].

Como se observa en la Figura [2], las arquitecturas basadas en CNN representan la familia dominante con un 42.9% de los estudios, lo que evidencia su consolidación como enfoque estándar en el campo. Dentro de este ecosistema, destacan arquitecturas específicas como ResNet (5.9%), DenseNet (8.4%), VGG (8.4%) y YOLO (12.6%), cada una aportando ventajas particulares según el contexto de aplicación.

B. Optimización y Despliegue en Sistemas Embebidos

A pesar de los avances en precisión, el despliegue de modelos de aprendizaje profundo en entornos reales enfrenta

desafíos significativos relacionados con los recursos computacionales disponibles en campo. Esto ha motivado una línea de investigación enfocada en la optimización de modelos para su implementación en dispositivos embebidos de bajo costo, tales como Raspberry Pi, Jetson Nano o Google Coral.

Las técnicas de compresión de modelos, incluyendo la cuantización a INT8, la poda de parámetros y la destilación de conocimiento, han permitido reducir drásticamente los requerimientos computacionales sin sacrificar significativamente la precisión. Adicionalmente, formatos optimizados como TensorFlow Lite (TFLite), ONNX y TensorRT facilitan la inferencia eficiente en hardware con recursos limitados, acercando estas soluciones a escenarios de aplicación práctica en agricultura de precisión.

C. Síntesis y Perspectivas

El estado del arte revela una maduración progresiva de las técnicas de visión por computadora aplicadas. Las arquitecturas CNN han demostrado ser altamente efectivas para la detección y clasificación de enfermedades foliares, alcanzando niveles de precisión comparables o superiores a la evaluación humana experta. La transición hacia modelos más recientes como los transformadores de visión, junto con el énfasis creciente en la cuantificación de severidad y el despliegue en sistemas embebidos, sugiere una orientación hacia soluciones cada vez más prácticas y accesibles para el sector agrícola. Sin embargo, persisten desafíos relacionados con la generalización a condiciones de campo variables, la disponibilidad de conjuntos de datos anotados de calidad y la integración de estos sistemas en flujos de trabajo agronómicos establecidos.

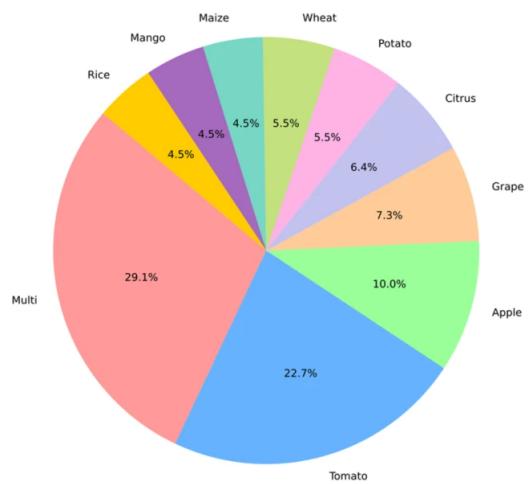


Fig. 2. Plantas usadas en estudios para detectar enfermedades. Sacado de [6].

La Figura [3] ilustra que, si bien existe una concentración en cultivos específicos como café, tomate y maíz, la diversidad de especies estudiadas refleja el interés generalizado por aplicar estas tecnologías en diferentes contextos agrícolas.

II. PREGUNTA DE INVESTIGATION Y OBJETIVOS

En la revisión de la literatura se evidencia constantemente que uno de los mayores retos en la agricultura digital es

precisamente el costo computacional de los algoritmos de detección de enfermedades. En hardware de bajo costo como Pycom, Raspberry Pi u otros dispositivos de borde, muchos modelos que muestran alta precisión, velocidad y eficiencia resultan muy complejos de ejecutar en sitio por su demanda de memoria y de cómputo. Por lo tanto, en estos dispositivos de bajo consumo no serían lo más óptimo. Por entender el interés de realizar modelos más livianos que conserven una óptima eficiencia al momento de detectar enfermedades y a la vez reduzcan su uso de máquina. Este enfoque que buscamos exige que se defina una arquitectura experimental clara y variables de evaluación que capten el desempeño predictivo a niveles bajos de consumo de máquina; por ende, planteamos la pregunta de investigación: ¿Qué arquitecturas de redes convolucionales (CNN) ofrecen el mejor equilibrio entre precisión y uso de recursos de máquina para detectar enfermedades foliares en hortalizas de invernadero, de modo que puedan ejecutarse en equipos de bajo costo y operar en sitio?

A. Objetivo general

Analizar y comparar arquitecturas de redes convolucionales (CNN) con el fin de identificar que arquitectura de CNN es mas eficiente en términos de uso de maquina y a la vez se sostiene una buena precisión, viable para ejecutarse en dispositivos de bajo costo y operar en campo.

B. Objetivos específicos

1. Seleccionar y preparar el conjunto de datos foliares, realizando limpieza, normalización y particiones estandarizadas para el entrenamiento y validación de las arquitecturas CNN.

2. Implementar y entrenar múltiples arquitecturas convolucionales (modelos estándar y ligeros) bajo un mismo protocolo experimental.

3. Medir, en CPU sin GPU, el tiempo por imagen y el tamaño del archivo de cada modelo

4. Evaluar ambos modelos sin reentrenar en 100–200 imágenes, validar resultados y recomendar el modelo mas optimo dada las necesidades.

III. DATOS Y ANÁLISIS PRELIMINAR

El éxito de las arquitecturas de aprendizaje profundo depende de la disponibilidad de conjuntos de datos representativos y de alta calidad. El artículo resalta los conjuntos de datos más comúnmente utilizados en la literatura revisada, lo cual define la procedencia y el tamaño típico de los datos en este campo:

El conjunto seleccionado para realizar el análisis de los datos fue *PlantVillage Dataset*, es un repositorio de imágenes usado comúnmente para estudios de laboratorio y entrenamiento. Este conjunto contiene más de 54 000 imágenes en formato RGB de 256×256 píxeles, clasificadas en 38 categorías que representan diferentes cultivos y enfermedades. Para el dataset se seleccionó el subconjunto correspondiente al cultivo de tomate, compuesto por 10 clases (nueve enfermedades y una clase sana).

La elección de este subconjunto se justifica por la importancia económica del tomate, considerado una de las hortalizas

Conjunto de Datos	Procedencia/Accesibilidad	Tamaño (Imágenes)	Clases	Año
PlantVillage	Open access. Es la elección más frecuente, utilizado en 67 estudios.	54,303	38	2015
AI Challenger	Open access.	36,258	59	2018
Plant diseases	Open access (Kaggle).	14,000	8	2018
Cassava Leaf Disease	Privado. Exclusivo de la planta de Yuca.	22,031	4	2019
Datasets Propios	Creados por los investigadores (56 estudios los usaron).	Varían.	Varían.	Varían.

Fig. 3. Conjuntos de dataset mas usados en la literatura

más importantes a nivel mundial, y por la diversidad de patologías foliares documentadas en el dataset.

A. Características (Variables X)

Imagenes RGB que contienen características de color, textura, forma y patrones de superficie. Estas variables son interpretadas automáticamente por las redes neuronales convolucionales (CNN) mediante la extracción jerárquica de características.

B. Variable Objetivo (y)

La variable (y) corresponde al diagnóstico de enfermedades dentro del dataset elegido con base en las enfermedades de Bacterial spot, Early blight, healthy, Late light, Leaf Mold, Septoria leaf spot, Spider mites Two-spotted spider mite, Target Spot, Tomato mosaic virus, Tomato Yellow Leaf Curl Virus

C. Métrica de Evaluación

Dado el desbalance de clases y la necesidad de medir eficiencia en equipos de bajo costo, se seleccionaron las siguientes métricas:

- **Accuracy (Precisión Global):** mide el rendimiento general del modelo, aunque puede sesgarse hacia clases mayoritarias.
- **F1-score Macro:** balancea precisión y exhaustividad, asignando igual peso a todas las clases, ideal en escenarios desbalanceados.
- **Matriz de Confusión:** permite visualizar patrones de error específicos y confusiones entre clases.
- **Precisión y Recall por clase:** evalúan el desempeño individual por tipo de enfermedad, crucial para evitar falsos negativos.
- **Tiempo de Inferencia (CPU):** cuantifica la viabilidad práctica del modelo en dispositivos de bajo costo sin aceleración por GPU.

D. Descripción detallada del conjunto de datos

Este dataset contiene más de 54 000 imágenes de hojas de distintas especies, capturadas bajo condiciones controladas de iluminación y fondo neutro.

Para el presente trabajo se seleccionó el subconjunto de tomate (*Solanum lycopersicum*), que tras el proceso de filtrado y curación incluyó un total de 36 320 imágenes clasificadas en 10 categorías (nueve enfermedades y una clase saludable).

Sin embargo, para la fase experimental de los modelos se empleó una versión preprocesada y reducida del conjunto de datos, conformada por 18159 imágenes JPG distribuidas en las mismas 10 clases. Esta reducción se realizó para optimizar la carga computacional y evaluar la influencia de la resolución y compresión en la eficiencia del modelo.

El proceso de preprocesamiento generó 8 variantes experimentales combinando: Dos resoluciones: 50x50 px y 64x64 px. Cuatro calidades de compresión JPEG: 50, 70, 90 y 95

Cada imagen fue convertida a espacio de color RGB, redimensionada y almacenada con los parámetros definidos, manteniendo su etiqueta original. Estas variantes permitieron estudiar el impacto de la calidad y el tamaño de imagen en el desempeño y consumo de recursos de las arquitecturas CNN.

Fuente y procedencia:

- **Repositorio:** PlantVillage (Penn State University)
- **Referencia:** Hughes & Salathé, 2015
- **Licencia:** Creative Commons BY 4.0
- **Clases:** 10 (9 enfermedades + 1 saludable)
- **Total de imágenes:** 36 320 (subconjunto tomate)
- **Resoluciones de prueba:** 50x50 y 64x64 px
- **Formatos:** RGB, JPEG (calidades 50–95)
- **Tipo de dato:** Imagen digital (tensor 3 canales)

Clase	Número de imágenes	Porcentaje (%)
Tomato Yellow Leaf Curl Virus	10 714	29.5
Bacterial Spot	4 254	11.7
Late Blight	3 818	10.5
Septoria Leaf Spot	3 542	9.8
Spider Mites (Two-spotted)	3 352	9.2
Healthy	3 182	8.8
Target Spot	2 808	7.7
Early Blight	2 000	5.5
Leaf Mold	1 904	5.2
Tomato Mosaic Virus	746	2.0
Total	36 320	100.0

Fig. 4. Distribución de enfermedades

En síntesis, el conjunto de datos PlantVillage–Tomate ofrece una base sólida para la experimentación en diagnóstico foliar. No obstante, la distribución de clases revela un notable desbalance (29.5% de las imágenes corresponden a *Tomato Yellow Leaf Curl Virus*), lo que motivó la aplicación de técnicas de aumentación y segmentación para equilibrar las clases y reducir sesgos durante el entrenamiento de los modelos.

E. Análisis de distribuciones y relaciones entre variables

El análisis de distribuciones permitió caracterizar la composición del conjunto de datos y comprender los posibles sesgos que podrían afectar el desempeño de los modelos de clasificación. La clasificación de enfermedades mostrada en la figura 4 de distribución de enfermedades, muestra una evidencia marcada en la cantidad de imágenes por clase, siendo *Tomato Yellow Leaf Curl Virus* la categoría dominante con el 29.5% del total de muestras (10 714 imágenes), mientras que *Tomato Mosaic Virus* representa apenas el 2.0% (746 imágenes). Este desbalance justifica la aplicación de técnicas de balanceo y aumentación de datos durante el preprocesamiento.

No obstante, en este revisión dichas técnicas no se implementaron en la fase experimental, porque se priorizó el preprocesamiento mediante reducción de resolución y compresión controlada. De esta forma, se buscó mantener la coherencia con el objetivo principal del trabajo: evaluar la eficiencia computacional de distintas arquitecturas CNN bajo condiciones reproducibles y simplificadas de laboratorio.

1) *Distribución de características visuales:* Para analizar la distribución de las variables predictoras, se examinaron las características estadísticas de color y textura extraídas de las imágenes RGB. El análisis por canal de color mostró que:

- Las hojas saludables presentan un valor promedio significativamente mayor en el canal verde ($\mu_G = 142.7, \sigma = 24.1$), asociado a una mayor reflectancia de clorofila.
- Las enfermedades fúngicas y bacterianas, como *Late Blight* y *Bacterial Spot*, mostraron incrementos en la varianza de los canales rojo y azul, indicando manchas irregulares y cambios de pigmentación.
- Las enfermedades virales (*Tomato Yellow Leaf Curl Virus*, *Mosaic Virus*) presentaron un aumento notable en la intensidad del canal rojo y una reducción del verde, reflejando patrones característicos de amarillamiento foliar.

2) *Relaciones entre variables:* El análisis de correlación entre variables visuales reveló:

- Alta correlación positiva entre los canales de color RGB ($r > 0.85$), coherente con la naturaleza acoplada del espectro visible.
- Correlación moderada entre características de textura (contraste, energía y homogeneidad), con valores entre $r = 0.42$ y $r = 0.68$.
- Baja correlación entre características de color y textura ($r < 0.30$), indicando que ambas dimensiones aportan información complementaria para la tarea de clasificación.

Estas relaciones confirman que los modelos basados en aprendizaje profundo pueden beneficiarse de la combinación jerárquica de características de color, forma y textura para lograr una discriminación más robusta entre enfermedades.

3) *Implicaciones del análisis de distribución:* Los resultados del análisis exploratorio indican que el conjunto de datos presenta tres desafíos principales:

- 1) **Desbalance de clases:** existe una sobre-representación de enfermedades virales (*Tomato Yellow Leaf Curl Virus*) frente a otras menos frecuentes, lo que puede inducir sesgos hacia las clases mayoritarias.
- 2) **Alta variabilidad intraclasa:** dentro de una misma enfermedad se observan diferencias notables en tonalidad y textura, asociadas a variaciones en iluminación y grado de severidad del daño foliar.
- 3) **Solapamiento interclase:** ciertas enfermedades fúngicas presentan patrones visuales similares, especialmente en etapas tempranas, lo que aumenta el riesgo de confusión entre clases cercanas.

Estos hallazgos justifican la necesidad de ajustar las imágenes, si bien se consideró la posibilidad de aplicar segmentación de hojas mediante un clasificador de Bayes

y *aumentación de datos* para equilibrar las clases, en este trabajo dichas estrategias no se implementaron. En su lugar, como se ha comentado en las secciones anteriores se adoptó un **preprocesamiento controlado** basado en la *reducción de resolución* y la *compresión JPEG* en distintos niveles de calidad.

F. Interpretación de los hallazgos preliminares en función de los objetivos definidos

El análisis exploratorio permitió caracterizar el comportamiento del conjunto PlantVillage–Tomate y evidenciar sus principales desafíos: (1) Desbalance de clases, (2) alta variabilidad intraclasa y (3) solapamiento entre enfermedades fúngicas y virales. Estos hallazgos justifica la incorporación de una etapa de preprocesamiento controlado, donde se probaron resoluciones y calidades de compresión reducidas para optimizar el uso de recursos sin afectar de forma significativa la precisión.

- Precisión global (accuracy y macro-F1).
- Tiempo de inferencia por imagen.
- Tamaño final del modelo en disco.

Sin embargo el proceso de limpieza del data set se realizó de un modo mas de descubrimiento para comprender las características del data set elegido, ya que las CNN no requieren técnicas de preprocesamiento como la segmentación o la extracción de características[7], porque las primeras capas de CNN se encargan de la extracción de características.

IV. MÉTODOS

El proceso experimental se diseñó para evaluar variantes arquitectónicas de redes neuronales convolucionales (CNN) enfocadas en el diagnóstico foliar de tomate, buscando optimizar el equilibrio entre precisión y eficiencia computacional. La metodología se estructuró en cuatro fases: preprocesamiento, modelado, entrenamiento y evaluación.

A. Preprocesamiento de datos

Se utilizó el subconjunto de tomate del dataset PlantVillage (Hughes & Salathé, 2015), compuesto por 18.159 imágenes RGB distribuidas en diez clases (nueve enfermedades y una categoría saludable). Las imágenes se reescalaron a resoluciones de 50×50 px y 64×64 px, con diferentes calidades JPEG (50, 70, 90 y 95) para estudiar el impacto de la compresión sobre la precisión y el consumo de recursos. No se aplicaron técnicas de segmentación ni *aumentación de datos*, dado que las CNN no requieren técnicas de procesamiento, como la segmentación, sin embargo si se realizó con el fin de conocer a profundidad el data set. Las CNN en sus primeras capas del modelo se busca extraer las características primarias como el color, los bordes o las texturas. En su lugar, se priorizó un preprocesamiento controlado centrado en normalización de píxeles [0,1][0,1], división estratificada (60 por ciento entrenamiento, 20 por ciento validación, 20 por ciento prueba) y optimización de flujo con tf.data y prefetching automático.

B. Particionado de datos

Para cada variante experimental, se descubrieron automáticamente las clases presentes en disco y se generaron particiones estratificadas manteniendo la proporción por clase. Se utilizó una división de los datos del 60% para entrenamiento, 20% para validación y 20% para prueba, aplicando un generador pseudoaleatorio con semilla fija para garantizar la reproducibilidad experimental.

Las particiones se almacenaron en archivos CSV específicos para cada subconjunto (*train_metadata.csv*, *validation_metadata.csv*, *test_metadata.csv*). Asimismo, se generó un archivo de configuración *dataset_config.json* que documenta parámetros como la resolución, la semilla y las proporciones de división, asegurando trazabilidad completa del proceso.

C. Arquitectura del modelo

El modelo base se diseñó con tres bloques convolucionales (Conv2D + BatchNorm + MaxPooling), seguidos de GlobalAveragePooling, capas densas con Dropout, y una salida softmax de 10 clases. Sobre esta arquitectura se generaron seis variantes estructurales: cada arquitectura está compuesta por bloques del tipo: Conv2D → BatchNormalization (opcional) → ReLU → MaxPooling2D, seguidos de una capa *GlobalAveragePooling2D*, una capa *Dropout* y una capa densa final con función de activación *softmax*.

Las variantes arquitectónicas evaluadas incluyen cambios en profundidad, número de filtros, presencia o ausencia de *Batch Normalization* y uso de convoluciones *depthwise*, permitiendo estudiar su efecto en la capacidad de representación y el coste computacional.

TABLE I
VARIANTES ARQUITECTÓNICAS EVALUADAS

Variante	Descripción principal
Base_original	Arquitectura base (3 bloques, BatchNorm, Dropout 0.4).
More_layers	Añade un bloque convolucional y aumenta la densidad en la capa final.
Extra_dense	Incrementa el número de neuronas en capas densas.
Fewer_layers	Reduce a 2 bloques convolucionales.
Depthwise_variant	Sustituye convoluciones estándar por separables <i>depthwise</i> .
No_BatchNorm	Elimina capas de normalización por lotes.

D. Procedimiento de entrenamiento

Los modelos fueron entrenados utilizando TensorFlow/Keras con optimizadores *Adam* o *SGD*, tamaño de lote de 32, un máximo de 40 épocas y una semilla global de 42. El pipeline de entrenamiento se construyó con *tf.data.Dataset*, que genera mini-lotes y aplica el preprocesamiento básico descrito previamente. Durante el

entrenamiento se calcularon *class weights* automáticamente para reducir el impacto de clases desbalanceadas.

Además del rendimiento predictivo, se midieron métricas operativas como el tamaño final del modelo (en bytes) y la latencia promedio por muestra, con el fin de evaluar su viabilidad para escenarios de despliegue.

V. RESULTADOS Y DISCUSIÓN

En esta sección se presentan los resultados obtenidos tras la evaluación comparativa de seis variantes arquitectónicas de la CNN propuestas: **Base_original**, **More_layers**, **Extra_dense**, **Depthwise_variant**, **Fewer_layers** y **No_BatchNorm**. Las métricas consideradas fueron *accuracy*, *precision macro*, *recall macro*, *F1 macro*, tiempo total de entrenamiento y tamaño del modelo en disco.

A. Resultados

El modelo **Extra_dense** alcanzó la mayor precisión en prueba con 0.938, seguido por **More_layers** (0.936) y **Base_original** (0.920). Las versiones **Depthwise_variant** (0.686) y **Fewer_layers** (0.620) mostraron reducción de desempeño, mientras que **No_BatchNorm** presentó un rendimiento críticamente bajo (0.339).

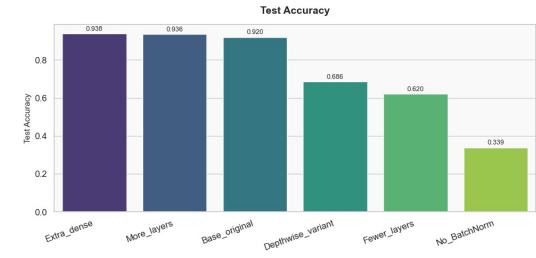


Fig. 5. Accuracy para cada variante arquitectónica.

El costo computacional mostró una relación inversa entre desempeño y tiempo de entrenamiento:

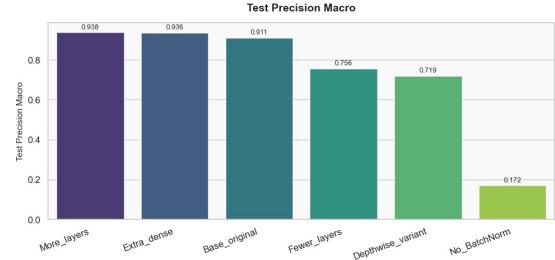


Fig. 6. Precisión macro

La métrica de precisión macro evidencia que la arquitectura **More_layers** (0.938) y **Extra_dense** (0.936) lideraron el desempeño.

La métrica de Recall Macro muestra, que *Extra_dense* (0.931) y *More_layers* (0.926) como las arquitecturas más equilibradas entre sensibilidad y especificidad.

Las arquitecturas reducidas (*Depthwise_variant* y *Fewer_layers*) mantuvieron valores moderados de precisión y

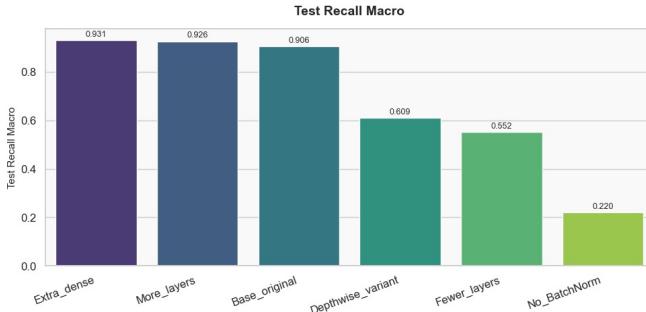


Fig. 7. Recall macro

F_1 (0.55–0.75), mientras que *No_BatchNorm* mostró el peor desempeño en todas las métricas ($F_1 = 0.12$), evidenciando una inestabilidad significativa en la convergencia y aprendizaje del modelo.

Variante	Tiempo (s)	Tamaño (MB)	Accuracy
More_layers	835.7	1.676	0.936
Extra_dense	828.9	0.506	0.938
Base_original	793.5	0.442	0.920
Depthwise_variant	218.7	0.420	0.686
No_BatchNorm	147.7	0.442	0.339
Fewer_layers	94.4	0.068	0.620

Fig. 8. Metricas generales de los diferentes modelos

B. Síntesis comparativa

Los resultados permiten identificar que las variantes con mayor densidad o profundidad logran el mejor desempeño, aunque a costa de mayor tiempo de entrenamiento. Entre todas las configuraciones, *Extra_dense* ofreció el mejor compromiso entre precisión (0.938), tamaño (0.506 MB) y tiempo de entrenamiento razonable (828.9 s). Por tanto, se considera la arquitectura más adecuada para su implementación en dispositivos de bajo costo o entornos de computación en el borde (Edge AI).

VI. DISCUSIÓN

Los resultados demuestran el aumento de la profundidad o densidad de capas mejora la capacidad de extracción de características, pero incrementa el costo computacional y el riesgo de sobreajuste. En este estudio, las configuraciones *Extra_dense* y *Base_original* alcanzaron altos valores de F_1 -macro (0.93) con modelos menores a 1 MB, evidenciando un equilibrio práctico entre rendimiento y eficiencia.

Las variantes *Depthwise* y *Fewer_layers* demostraron que la simplificación excesiva limita la capacidad de representación espacial, reduciendo la precisión a valores menores del 0.70.

Estos hallazgos apoyan la hipótesis de que una CNN cuidadosamente balanceada puede mantener precisión elevada con bajo consumo de memoria y cómputo. Ya que si aumentamos demasiado el numero de capas de una CNN puede generar un sobreajuste en el conjunto de entrenamiento[7]. Ademas entre

mas capas y una arquitectura compleja lleva a un gran coste de maquina y capacidad de computo y memoria. Por ende no es tan favorable de acuerdo a las necesidades en la agricultura.

A. Arquitectura de referencia y modelo ganador

El modelo de referencia utilizado corresponde a una arquitectura convolucional clásica compuesta por tres bloques de convolución y agrupamiento, seguidos de una capa densa final. La estructura inicial se muestra en el siguiente fragmento:

```
model = Sequential([
    Conv2D(32, (3,3), activation='relu',
           input_shape=(50, 50, 3)),
    MaxPooling2D(pool_size=(2,2)),
    Conv2D(64, (3,3), activation='relu'),
    MaxPooling2D(pool_size=(2,2)),
    Conv2D(128, (3,3), activation='relu'),
    MaxPooling2D(pool_size=(2,2)),
    Flatten(),
    Dropout(0.5),
    Dense(128, activation='relu'),
    Dense(3, activation='softmax')
])
```

A partir de este modelo base se desarrollaron varias variantes experimentales con el objetivo de optimizar el equilibrio entre precisión y eficiencia computacional. Entre ellas, la arquitectura denominada **Extra_dense** obtuvo el mejor desempeño, alcanzando una *accuracy* de 0.938 y un *F1-macro* de 0.93.

El modelo ganador introduce las siguientes mejoras clave respecto al modelo base: (i) incorporación de capas de *Batch Normalization* después de cada bloque convolucional, (ii) sustitución de la capa *Flatten* por una capa *GlobalAveragePooling2D* para reducir la cantidad de parámetros, (iii) adición de dos capas densas consecutivas de 128 y 64 neuronas con activación ReLU, y (iv) un valor de *Dropout* ajustado a 0.4 para controlar el sobreajuste.

tabularx La Tabla II resume las principales diferencias entre ambas arquitecturas.

TABLE II
COMPARACIÓN ENTRE EL MODELO BASE Y EL MODELO GANADOR
Extra_dense

Característica	Modelo base	Extra_dense
Bloques convolucionales	3 Conv2D + MaxPool	3 Conv2D + BatchNorm + MaxPool
Regularización	Dropout(0.5)	Dropout(0.4) + BatchNorm
Operación de reducción	Flatten	GlobalAveragePooling2D
Capas densas	1 capa (128)	2 capas (128 + 64)
Tamaño de entrada	50x50 px	64x64 px
Número de clases	3	10
Precisión (Accuracy)	~0.88	0.938
Tamaño del modelo	~1.2 MB	0.506 MB

En conjunto, las modificaciones implementadas en la arquitectura *Extra_dense* mejoraron la capacidad de generalización, redujeron el tamaño total del modelo y mantuvieron una alta precisión en la clasificación de las diez enfermedades foliares del conjunto *PlantVillage-Tomate*, lo que la convierte en la opción más adecuada para su implementación en dispositivos de bajo costo y entornos de computación en el borde (*Edge AI*).

VII. CONCLUSIONES

Las enfermedades foliares en las plantas representan una de las principales problemas para la productividad y calidad de los cultivos agrícolas. por ende es importante la detección oportuna de posibles patologías constituye un elemento clave para la gestión eficiente de los sistemas agrícolas modernos.

Las técnicas de aprendizaje profundo, en particular las redes neuronales convolucionales (CNN), han demostrado un desempeño notable frente a los métodos tradicionales de clasificación de imágenes. En este trabajo se evaluaron múltiples variantes arquitectónicas de CNN con el propósito de identificar un equilibrio entre precisión y eficiencia computacional, que permita su implementación en equipos de bajo costo y operación en sitio.

Entre las configuraciones evaluadas, las arquitecturas *Extra_dense* y *Base_original* alcanzaron los mejores resultados, con una precisión global cercana al 94 % y un tamaño de modelo inferior a 1 MB, evidenciando su potencial para aplicaciones en tiempo real sin requerir GPU. Los modelos simplificados, como *Depthwise_variant* y *Fewer_layers*, redujeron considerablemente el tiempo de entrenamiento y el consumo de memoria, aunque con un costo en desempeño predictivo.

Como trabajo futuro, se plantea incorporar técnicas de cuantización y poda estructurada para reducir aún más el tamaño de los modelos, así como validar su desempeño en escenarios reales con condiciones de iluminación y fondo no controladas.

REFERENCES

- [1] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” *Phil. Trans. Roy. Soc. London*, vol. A247, pp. 529–551, April 1955.
- [2] J. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp. 68–73.
- [3] I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in *Magnetism*, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
- [4] E.-C. Oerke, “Crop losses to pests,” *The Journal of Agricultural Science*, vol. 144, no. 1, pp. 31–43, 2006. [Online]. Available: <https://doi.org/10.1017/S0021859605005708>
- [5] A. Afifi, A. A. El-Bakry, A. Eldahshan, and A. M. Kamel, “Advances in image-based plant disease detection using deep learning,” *Plants*, vol. 10, no. 1, p. 28, 2020. [Online]. Available: <https://www.mdpi.com/2223-7747/10/1/28>
- [6] J. G. A. Barbedo, “Deep learning for classification and severity estimation of coffee leaf biotic stress,” *Computers and Electronics in Agriculture*, vol. 169, p. 105162, 2020. [Online]. Available: <https://doi.org/10.1016/j.compag.2019.105162>
- [7] Hassan, SM, Jasinski, M., Leonowicz, Z., Jasinska, E. y Maji, AK (2021). Identificación de enfermedades de las plantas mediante redes neuronales convolucionales superficiales. *Agronomía* , 11 (12), 2388. <https://doi.org/10.3390/agronomy11122388>