



Arquitectura híbrida EfficientNet y Vision Transformer para detección multiplataforma de patologías agrícolas

Hybrid EfficientNet and Vision Transformer architecture for cross-platform agricultural pathology detection

Erick Rodríguez Ramírez ✉


Facultad de Ciencias de la Computación, Universidad Autónoma de Puebla, Puebla, México
erick.rodriguezra@alumno.buap.mx
ORCID: 0009-0009-2344-4492

Maya Carrillo Ruiz

Facultad de Ciencias de la Computación, Universidad Autónoma de Puebla, Puebla, México
maya.carrillo@correo.buap.mx
ORCID: 0000-0001-6152-456X

Hortensia Carrillo Ruiz

Facultad de Ciencias Biológicas, Universidad Autónoma de Puebla, Puebla, México
hortensia.carrillo@correo.buap.mx
ORCID: 0000-0003-4247-6114

 <https://doi.org/10.36825/RITI.14.33.009>

Recibido: Enero 29, 2026

Aceptado: Abril 08, 2026

Resumen: El diagnóstico temprano de enfermedades foliares es crucial para la seguridad alimentaria; sin embargo, existe una brecha de aplicabilidad (*reality gap*) que limita la viabilidad operativa de los modelos de *Deep Learning* en entornos agrícolas reales debido a su alta complejidad computacional y sensibilidad al ruido visual. El objetivo de este estudio es diseñar, validar y desplegar una arquitectura híbrida y ligera (AgroScan) capaz de operar como una herramienta de soporte de decisiones en tiempo real frente a 49 enfermedades distribuidas en 10 cultivos de alto impacto. La metodología propone un proceso de inferencia de dos etapas: un modelo “portero” de filtrado binario para mitigar el ruido de fondo visual, seguido de un clasificador entrenado con 98,000 imágenes, que fusiona la eficiencia paramétrica de EfficientNetB0 con la capacidad de correlación espacial de un módulo *Transformer (Multi-Head Self-Attention)*. Entrenado y evaluado sobre 98,000 imágenes. La arquitectura híbrida alcanzó una exactitud global del 94.69% (IC 95%: [94.28%, 95.11%]) y un F1-Score de 94.68% (IC 95%: [94.22%, 95.08%]). Los intervalos de confianza se calcularon con remuestreo no paramétrico sobre el conjunto de prueba. El aporte principal del estudio radica en la demostración empírica de que la atención global incrementa la exactitud en el diagnóstico de enfermedades manteniendo una latencia de inferencia (9.45 ms por imagen). Finalmente, la implementación de la arquitectura mediante una plataforma cliente-servidor accesible a través de aplicaciones web y móviles, ha demostrado su robustez operativa como herramienta viable de apoyo agrícola.

Palabras clave: *Deep Learning, Vision Transformers, EfficientNet, Enfermedades Foliares, Agricultura de Precisión, Reality Gap.*

Abstract: The early diagnosis of foliar diseases is crucial for food security; however, there is an applicability gap (reality gap) that limits the operational viability of Deep Learning models in real agricultural environments due to their high computational complexity and sensitivity to visual noise. The objective of this study is to design, validate, and deploy a lightweight, hybrid architecture (AgroScan) capable of operating as a real-time decision support tool against 49 diseases distributed across 10 high-impact crops. The methodology proposes a two-stage inference process: a binary filtering "gatekeeper" model to mitigate visual background noise, followed by a classifier trained on 98,000 images, which fuses the parametric efficiency of EfficientNetB0 with the spatial correlation capability of a Transformer module (Multi-Head Self-Attention). Trained and evaluated on 98,000 images. The hybrid architecture achieved an overall accuracy of 94.69% (95% CI: [94.28%, 95.11%]) and an F1-Score of 94.68% (95% CI: [94.22%, 95.08%]). Confidence intervals were calculated using non-parametric resampling on the test set. The main contribution of the study lies in the empirical demonstration that global attention increases accuracy in disease diagnosis while maintaining an inference latency (9.45 ms per image). Finally, the implementation of the architecture through a client-server platform, accessible via web and mobile applications, has demonstrated its operational robustness as a viable agricultural support tool.

Keywords: *Deep Learning, Vision Transformers, EfficientNet, Foliar Diseases, Precision Agriculture, Reality Gap.*

1. Introducción

México cultiva aproximadamente 280 especies cíclicas y 199 perennes. La Ley de Desarrollo Rural Sustentable (artículo 179) clasifica siete cultivos como estratégicos: maíz, caña de azúcar, frijol, trigo, arroz, sorgo y café [1]. Según la FAO (2011), la seguridad alimentaria requiere garantizar la disponibilidad de estos recursos [2]. Sin embargo, las variaciones climáticas y la incidencia de patógenos constituyen la principal causa de merma agrícola. Las enfermedades foliares degradan los tejidos fotosintéticos y limitan la asimilación de nutrientes; su detección tardía genera daños irreversibles y pérdidas de rendimiento de hasta un 40%, como se documenta en las epidemias de roya en maíz.

El monitoreo manual de estas enfermedades exige una inversión intensiva de tiempo y recursos. Como alternativa, las Redes Neuronales Convolucionales (CNN) logran diagnosticar enfermedades con alta exactitud en entornos controlados [3]. No obstante, las arquitecturas convolucionales presentan dificultades para capturar el contexto global en imágenes de campo, provocando una caída de rendimiento operativo (*reality gap*) [4]. Para mitigar esta deficiencia y reducir la confusión entre enfermedades con sintomatología visual similar, investigaciones recientes integran mecanismos de atención (*Vision Transformers*) configurando arquitecturas híbridas [5].

A pesar de la eficacia teórica de los modelos híbridos, persiste una brecha de implementación. Su alta demanda computacional restringe su ejecución en los dispositivos móviles comerciales accesibles para el agricultor. El problema central de esta investigación radica en transferir la exactitud diagnóstica de las redes profundas al entorno agrícola real, superando las limitantes de hardware. En consecuencia, el objetivo de este trabajo es desarrollar un sistema multiplataforma basado en una arquitectura híbrida (*EfficientNet* y *Vision Transformer*) capaz de detectar 49 enfermedades en 10 cultivos.

Las contribuciones principales de esta investigación son tres:

- El diseño de una arquitectura híbrida para la extracción simultánea de texturas locales y dependencias globales.
- La implementación de una plataforma de software distribuida (*React Native* y *Flask*) que delega la carga de inferencia a un servidor centralizado.
- La validación empírica del modelo mediante pruebas in-situ (imágenes tomadas en condiciones no controladas), evaluando así su operatividad frente a obstrucciones y ruido de fondo.

2. Marco Teórico

En AgroScan son importantes algunos conceptos de inteligencia artificial e ingeniería de software. El primero son las Redes Neuronales Convolucionales (CNN), que se han convertido en un estándar para extraer características jerárquicas (bordes, texturas y formas complejas) mediante operaciones matemáticas. En este trabajo se utiliza la versión más pequeña y rápida de la familia de CNN llamada *EfficientNet*. Esta familia de CNN optimiza la relación entre precisión y costo computacional mediante el escalado compuesto de la red [6] que consiste en escalar de manera uniforme el ancho, la profundidad y la resolución de la red. Dado que las CNN limitan la comprensión del contexto global de la imagen, en nuestra propuesta se integra *Vision Transformers* (ViT) [7], un modelo de aprendizaje profundo, cuyo núcleo, el mecanismo de Atención Multicabezal (*Multi-Head Self-Attention*), relaciona parches distantes de la imagen. Esto otorga al sistema la capacidad de procesar el contexto global de la imagen (por ejemplo, la distribución dispersa de manchas en la superficie foliar). Esta visión global es fundamental para discriminar entre enfermedades que presentan lesiones locales similares, pero con patrones de distribución distintos. Por otro lado, la transferencia de aprendizaje (*Transfer Learning*) se utiliza para optimizar el entrenamiento, pues permite reutilizando el conocimiento de modelos entrenados previamente en conjuntos masivos (ImageNet, con 1.2 millones de imágenes). Esta técnica es útil en el dominio agrícola, pues reduce la necesidad de recolectar grandes bases de datos, ya que permite adaptar el modelo previamente entrenado mediante el ajuste fino (*Fine-Tuning*) de los pesos sinápticos para una tarea específica. Finalmente, para el despliegue del sistema se utiliza una arquitectura Cliente-Servidor comunicada vía API REST, utilizando el protocolo HTTP y el intercambio de datos en formato JSON. Esta separación estructural delega la inferencia del modelo (de alta demanda computacional) a un servidor centralizado. Simultáneamente, la aplicación móvil funciona como un cliente ligero encargado exclusivamente de capturar la imagen y mostrar el diagnóstico de las enfermedades, garantizando la escalabilidad y el mantenimiento del sistema.

3. Estado del arte

El desarrollo de sistemas automáticos para la detección de enfermedades agrícolas ha evolucionado desde pruebas de laboratorio hasta arquitecturas orientadas a la eficiencia operativa. A continuación, se presenta una revisión cronológica y temática.

El trabajo base se consolidó entre 2016 y 2019. Investigaciones como la de Mohanty *et al.* [8] utilizaron Redes Neuronales Convolucionales (CNN) sobre el conjunto *PlantVillage*, logrando una exactitud superior al 99% en 14 cultivos. Desde entonces, diversos estudios han utilizado ampliamente las CNN para la detección de enfermedades, destacando su capacidad para extraer características visuales de manera automática [9].

A pesar del éxito en laboratorio, la literatura identifica barreras operativas debido a la variabilidad de iluminación y fondos en condiciones de campo. Para operar en entornos complejos, la investigación se dirigió hacia la fusión de arquitecturas. Propuestas validan la combinación de CNN con Transformers (ConvViT) [10] y enfoques multispectrales [11] para capturar dependencias globales y mitigar el sobreajuste.

Más recientemente, para elevar la eficiencia frente a la variabilidad agrícola, investigaciones de 2025 proponen optimizar las CNN integrando convoluciones de profundidad (*Depthwise Convolutions*) y bloques *Squeeze-and-Excitation* [12]. Asimismo, estudios paralelos emplean ensambles de *Transformers* [13] para lograr diagnósticos eficaces bajo ruido visual severo, superando las limitaciones locales de las CNN convencionales.

Existe una disociación en la literatura: los modelos híbridos suelen limitarse a simulaciones teóricas en servidores, mientras que las implementaciones móviles utilizan arquitecturas simplificadas para asegurar la velocidad de inferencia. Este trabajo aborda dicha brecha integrando una red híbrida en un ecosistema cliente-servidor. A diferencia de enfoques multitarea, la arquitectura AgroScan dedica su topología (*EfficientNet-B0* y *ViT-B16*) exclusivamente a la clasificación taxonómica fina de enfermedades.

Esta decisión arquitectónica permite alcanzar una exactitud del 94.69% en condiciones controladas y un rendimiento empírico del 88.1% en pruebas in-situ. El modelo captura el contexto global foliar y mitiga confusiones visuales directamente en campo. La Tabla 1 categoriza la literatura revisada frente a la arquitectura desarrollada en este estudio.

Tabla 1. Resumen cronológico comparativo de trabajos previos en el diagnóstico de enfermedades agrícolas.

Referencia	Año	Tipo de estudio	Arquitectura/ Enfoque	Entorno de evaluación	Aporte principal
Mohanty <i>et al.</i> [8]	2016	Propuesta	AlexNet y GoogLeNet (CNN)	PlantVillage (54,306 imágenes, controlado)	Estableció viabilidad técnica con 99.35% de exactitud bajo condiciones ideales de laboratorio.
Saleem <i>et al.</i> (MDPI Plants) [9]	2019	Revisión	Modelos ML y DL (CNN)	N/A (Estudio Documental)	Revisión exhaustiva de arquitecturas <i>Deep Learning</i> aplicadas al diagnóstico.
Li y Li (MDPI Agriculture) [10]	2022	Propuesta	Híbrida (ConvViT)	<i>Dataset</i> de campo	Integración de CNN y <i>Transformer</i> para identificación ligera y eficaz superando a las CNN base.
De Silva y Brown (Sensors) [11]	2023	Propuesta	Híbrida (ViT-CNN)	Imágenes multispectrales	Uso de atención y convoluciones para mitigar sobreajuste en condiciones agrícolas.
Ashurov <i>et al.</i> (Frontiers) [12]	2025	Propuesta	Depthwise CNN con bloques <i>Squeeze-and-Excitation</i>	Imágenes de campo	Optimización de extracción de características con alta exactitud (98.0%) y eficiencia.
Kalpana <i>et al.</i> (Frontiers) [13]	2025	Propuesta	<i>Ensemble Heterogeneous Transformer</i>	Entornos de ruido complejo	Framework de clasificación robusto para identificación simultánea de enfermedades en hojas complejas.
AgroScan (Propuesto)	2026	Propuesta	Híbrida (EfficientNet-B0 + ViT-B16)	Laboratorio (PlantVillage) y Campo In-situ	Clasificación taxonómica fina con validación empírica frente al sesgo de dominio (88.1% exactitud).

Fuente: Elaboración propia.

4. Método Propuesto

Para abordar el problema de la detección de enfermedades, se propone una arquitectura que busca equilibrar la exactitud del diagnóstico y la eficiencia computacional.

4.1. Arquitectura del Modelo Portero

El despliegue in-situ exige procesar imágenes con ruido visual (suelo, herramientas, manos). Evaluar este ruido de fondo directamente en el clasificador principal induce errores y consume recursos computacionales. Para mitigarlo, la arquitectura integra un pipeline de dos etapas que inicia con un "modelo portero" de filtrado binario. Su función exclusiva es discriminar si la entrada corresponde a una hoja (clase positiva) o a un elemento ajeno (clase negativa). Este filtro actúa como barrera primaria; por ejemplo, identifica y rechaza ruido no vegetal (rostro humano), deteniendo la inferencia y solicitando una nueva captura.

Para este filtro, se implementó MobileNetV2 mediante Transferencia de Aprendizaje por su eficiencia paramétrica (*Depthwise Separable Convolutions*). Al requerir un discriminador auxiliar de alta velocidad sin generar cuellos de botella computacionales, sus 2.26 millones de parámetros resultaron idóneos para dispositivos de borde.

El entrenamiento empleó el conjunto "*Leaf vs Non-Leaf Images*" (103,000 imágenes) [14], bajo una partición estratificada de 80% (entrenamiento), 10% (validación) y 10% (prueba). Se congeló el *backbone* preentrenado en ImageNet, optimizando únicamente la capa densa final (2,562 parámetros) utilizando el optimizador Adam (1×10^{-3}) y la función Entropía Cruzada Categórica. Se incluyeron capas nativas de aumento de datos (*Random Flip* y

Random Rotation) para dotar invariancia espacial. El proceso de entrenamiento requirió 1 hora y 15 minutos para completar el ciclo establecido de 25 épocas.

4.2. Arquitectura del Modelo Híbrido

El sistema de diagnóstico emplea una red neuronal híbrida que procesa la información visual integrando la eficiencia de las redes convolucionales con la capacidad contextual de los transformadores (Figura 1).

Se seleccionó EfficientNet-B0 como extractor de características (*backbone*) tras evaluar variantes como ResNet y MobileNet (ver Tabla 3). Esta arquitectura ofrece un equilibrio entre discriminación espacial y eficiencia paramétrica. Dado que el mecanismo de atención incrementa la complejidad computacional, este extractor garantiza la viabilidad operativa y reduce los tiempos de inferencia dentro de la arquitectura cliente-servidor implementada.

El flujo de tensores inicia con una imagen normalizada de $224 \times 224 \times 3$ canales RGB. El bloque convolucional EfficientNet-B0 procesa esta entrada mediante unidades de convolución móvil invertida (MConv). Estas capas reducen la resolución espacial y aumentan la profundidad, generando un mapa de características de $7 \times 7 \times 1280$. La transición al módulo *Transformer* analiza los 49 cuadrantes espaciales (7×7) mediante un mecanismo de Atención Multicabezal (*Multi-Head Self-Attention*). A diferencia del campo receptivo local de las CNN, esta configuración integra el contexto global de la hoja. Esto permite correlacionar patrones distantes, mitigando la confusión diagnóstica entre enfermedades con sintomatología visual similar. El resultado de la atención se comprime en un vector latente de 1×1280 . Finalmente, este vector alimenta un clasificador denso (*Fully Connected Layer*) con activación Softmax. Esta capa proyecta la información a un vector de salida de 1×49 , entregando la distribución de probabilidades para las clases diagnosticadas.

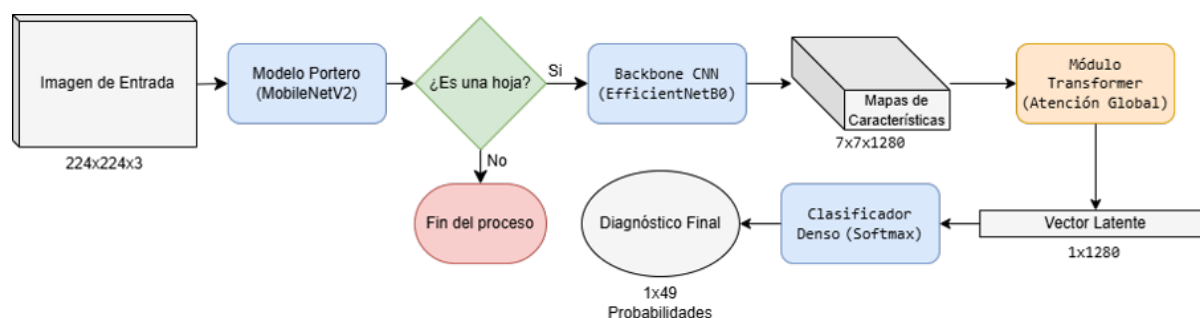


Figura 1. Arquitectura del sistema propuesto. La imagen de entrada es evaluada primero por un modelo portero (MobileNetV2) para el filtrado binario de ruido visual. Una vez validada, es procesada por el *backbone* EfficientNetB0 para generar mapas de características de $7 \times 7 \times 1280$. Estos mapas son ingeridos por el módulo *Transformer*, el cual aplica mecanismos de atención para producir un vector latente de contexto global (1×1280), finalizando en una capa densa que clasifica las 49 enfermedades mediante activación Softmax.

4.3. Arquitectura de Software y Despliegue

Buscando contar con una solución eficiente, se implementó una arquitectura cliente-servidor. Este diseño desacopla la presentación de la lógica de procesamiento. Como se muestra en la Figura 2, el sistema comprende dos entornos interconectados por protocolos web: el Lado del Usuario (*Frontend*) y el Lado del Servidor (*Backend*).

El *Frontend* opera mediante interfaces en React Native (aplicación móvil Android y plataforma web), fungiendo como terminales de adquisición. Al ingresar una imagen, el sistema ejecuta una codificación Base64, serializando la fotografía a texto. Este paquete se transmite mediante una petición HTTP POST, optimizando el consumo de datos en zonas de baja conectividad.

En el *Backend*, una API Flask actúa como puerta de enlace (*gateway*). Los datos se decodifican y la imagen se normaliza a $224 \times 224 \times 3$ píxeles. Este tensor alimenta al modelo híbrido (EfficientNet + ViT) para la inferencia. La clasificación y el nivel de confianza resultantes se empaquetan en un objeto JSON, retornando el diagnóstico al cliente en tiempo real [15].

La implementación centralizada frente al procesamiento local (Edge AI) obedece a limitantes de hardware, ya que el modelo híbrido exige recursos (GPU/TPU) ausentes en dispositivos móviles promedio. Asimismo, este enfoque habilita una estrategia de despliegue continuo (*Continuous Deployment*), permitiendo actualizar el modelo de diagnóstico de enfermedades directamente en el servidor sin requerir descargas adicionales del usuario.

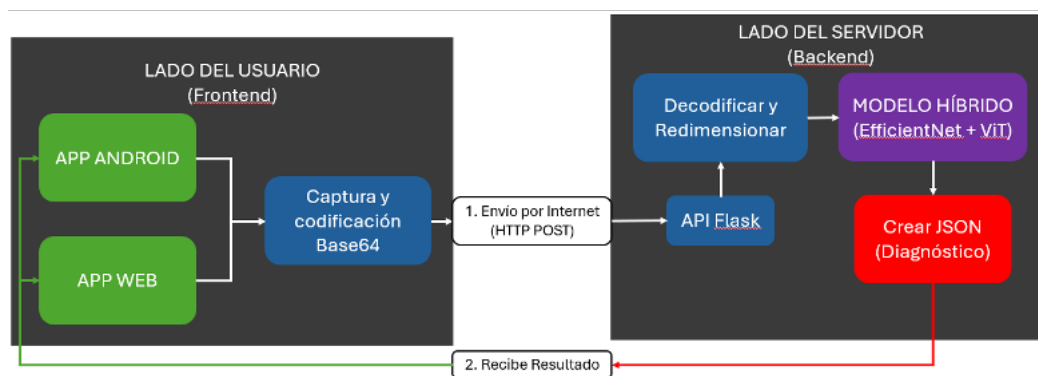


Figura 2. Diagrama de flujo de datos y arquitectura de despliegue del sistema. El lado del usuario (Izquierda) gestiona la captura y serialización de la imagen en formato Base64 a través de los clientes Web y Móvil. El lado del servidor (Derecha) centraliza la lógica de procesamiento: recibe la solicitud HTTP, decodifica el tensor, ejecuta la inferencia con el Modelo Híbrido (EfficientNet + ViT) y retorna el diagnóstico estructurado en un objeto JSON.

5. Datos Utilizados

La calidad del modelo de aprendizaje profundo depende de la representatividad de los datos de entrenamiento. Para este trabajo, se construyó un conjunto de datos heterogéneo (*custom dataset*) mediante la recopilación, filtrado y estandarización de imágenes.

5.1. Composición, origen y procesamiento del dataset

Se construyó un conjunto de 98,000 organizado en 49 clases para 10 cultivos: Maíz, Frijol, Trigo, Naranja, Caña de Azúcar, Tomate, Papa, Café, Pimiento y Chile. La distribución cuantitativa de las enfermedades se detalla en la Tabla 2. Cada cultivo incluye una categoría de control (“Sana”) para establecer una línea base visual que permita al modelo discriminar el tejido vegetal saludable de los primeros estadios de infección, reduciendo la tasa de falsos positivos.

Para maximizar la precisión morfológica, el conjunto integra distintas fuentes. El 70% de las imágenes proviene de PlantVillage [16] (cultivos como Tomate, Papa y Maíz), capturadas en condiciones controladas para facilitar el aprendizaje de las lesiones sin ruido visual. Para mitigar el sesgo de laboratorio, el acervo se complementó con repositorios abiertos de Kaggle y Mendeley Data para cultivos tropicales (Café y Caña de Azúcar), dotando al modelo de capacidad de generalización en escenarios de campo.

El procesamiento de los datos inició con la curaduría manual para eliminar duplicados y muestras desenfocadas. Las imágenes se redimensionaron a 224x224x3 píxeles mediante interpolación bicúbica, preservando las texturas de las lesiones necróticas. Para resolver el desbalance severo (ej. 5,000 muestras de tomate frente a 400 de café), se aplicaron técnicas de aumento de datos (*data augmentation*) exclusivamente en las clases minoritarias.

Este proceso consolidó un conjunto final de 98,000 imágenes, garantizando un balance estricto de 2,000 imágenes para cada una de las 49 enfermedades. La partición se ejecutó mediante un muestreo estratificado: 78,400 imágenes para entrenamiento (80%), 9,800 para validación (10%) y 9,800 para el conjunto de prueba aislado (10%). Finalmente, para suprimir la varianza computacional, se fijó una semilla aleatoria (*random seed*) global de 42, garantizando la reproducibilidad estricta del entorno experimental.

Tabla 2. Distribución cuantitativa del conjunto de datos por cultivo, detallando el recuento original de imágenes, el volumen tras el aumento de datos y la partición estratificada para las fases de entrenamiento, validación y prueba.

Cultivo	Clases	Imágenes originales	Aumento de datos (total balanceado)	Entrenamiento (80%)	Validación (10%)	Prueba (10%)
Café	5	3,224	10,000	8,000	1,000	1,000
Caña de azúcar	5	2,521	10,000	8,000	1,000	1,000
Chile	5	500	10,000	8,000	1,000	1,000
Frijol	3	990	6,000	4,800	600	600
Maíz	4	4,188	8,000	6,400	800	800
Naranja	4	1,021	8,000	6,400	800	800
Papa	8	9,710	16,000	12,800	1,600	1,600
Pimiento	2	2,475	4,000	3,200	400	400
Tomate	10	18,160	20,000	16,000	2,000	2,000
Trigo	3	3,273	6,000	4,800	600	600
Total	49	46,062	98,000	78,400	9,800	9,800

Fuente: Elaboración propia.

5.2. Pre-procesamiento y estrategia de aumento de datos

La heterogeneidad de las fuentes de datos, que combinan imágenes de laboratorio con fotografías de campo tomadas bajo condiciones variables, plantea desafíos relacionados con la consistencia visual. Para homogeneizar la entrada al modelo híbrido y asegurar que la red aprenda patrones reales de las enfermedades y no artefactos de la imagen, se aplicó un proceso de adecuación de tensores.

Primero las imágenes fueron redimensionadas a una resolución espacial de 224x224x3 píxeles mediante interpolación bicúbica, una decisión técnica orientada a preservar la nitidez de los bordes en las lesiones necróticas. Acto seguido, se realizó una normalización matemática de los valores de intensidad de los píxeles, escalando el rango original de enteros [0, 255] al intervalo flotante [0, 1]. Este paso facilita la estabilidad numérica y la convergencia del algoritmo de optimización (descenso de gradiente), y alinea la distribución estadística de los nuevos datos con la de los pesos pre-entrenados de ImageNet utilizados en el *backbone* EfficientNet, favoreciendo la transferencia de aprendizaje.

Paralelamente, para abordar el desbalance de clases previamente mencionado (donde cultivos predominantes superaban las 3,000 muestras y otros apenas alcanzaban las 400), se implementó una estrategia de Aumento de Datos Dirigido (*Targeted Data Augmentation*) para evitar el sobreajuste (*overfitting*). Entrenar con disparidad numérica generaría un sesgo en el modelo, favoreciendo la predicción de las clases mayoritarias. A diferencia de los enfoques que aumentan el conjunto de datos de manera uniforme, en este trabajo se aplicaron transformaciones geométricas y cromáticas exclusivamente a las clases minoritarias, logrando equilibrar la distribución de entrenamiento a 2,000 muestras exactas por clase [17].

Como se ejemplifica en la Figura 3, estas operaciones no alteran la semántica de la lesión, pero incrementan la variabilidad visual. Las transformaciones aplicadas incluyen:

- Invariancia rotacional: Rotaciones aleatorias en un rango de $\pm 30^\circ$ para simular la orientación variable del dispositivo móvil en manos del agricultor.
- Invariancia de escala: Operaciones de zoom aleatorio (rango 0.8 a 1.2) para emular diferentes distancias de captura respecto a la hoja.
- Robustez cromática: Ajustes aleatorios de brillo ($\pm 20^\circ$) para simular condiciones de iluminación natural fluctuantes, desde días nublados hasta sol directo.
- Simetría: Aplicación de volteo horizontal y vertical (*flip*) para duplicar las perspectivas de entrenamiento.

Esta técnica permite incrementar la diversidad efectiva del conjunto de datos sin costo adicional de recolección en campo, induciendo al modelo a aprender las características estructurales invariantes de la enfermedad en lugar de memorizar patrones de píxeles específicos.

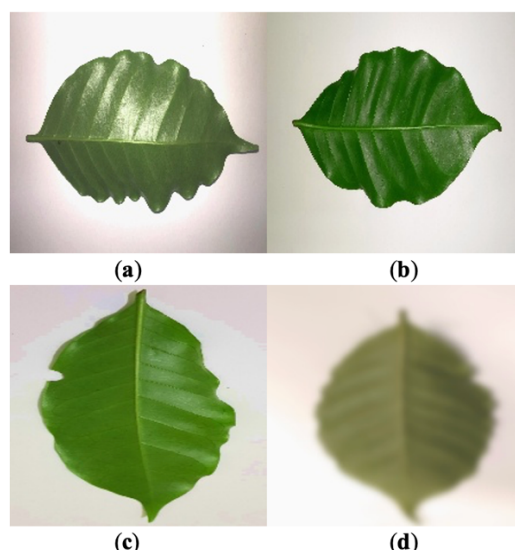


Figura 3. Ilustración de las técnicas de aumento de datos (*data augmentation*) utilizadas para mejorar la robustez del modelo, empleando una muestra representativa de la clase “Café Sano”. (a) Imagen base original capturada en condiciones controladas. (b) Variación con ajuste de intensidad lumínica para simular diferentes horas del día. (c) Rotación aplicada para generar invariancia ante la orientación del dispositivo de captura. (d) Transformación geométrica (escala/reflejo) para robustecer la detección ante variaciones de distancia y posición.

6. Experimentos y resultados

El entrenamiento del modelo se ejecutó en Google Colab Pro utilizando una GPU NVIDIA Tesla T4 (16 GB GDDR6) con una política de precisión mixta (*mixed precision*) [18] MobileNetV2. Esto optimizó el uso de la memoria VRAM y aceleró el cómputo conservando la precisión aritmética del gradiente. A nivel de software, la arquitectura se implementó empleando TensorFlow 2.19.0, PyTorch 2.2.0, Keras y CUDA 12.2.

Como optimizador se seleccionó Adam (*Adaptive Moment Estimation*) para ajustar tasas de aprendizaje individuales en los parámetros de la arquitectura híbrida. Para mitigar el sobreajuste, se implementó una capa de Dropout antes del clasificador final, operando junto con el aumento de datos y prescindiendo de penalización L2. El tamaño de lote (*batch size*) se fijó en 32 imágenes. La función de costo minimizada fue la Entropía Cruzada Categórica (*Categorical Crossentropy*), definida matemáticamente como:

$$L = - \sum_{c=1}^M y_{o,c} \log(p_{o,c})$$

Donde M es el número de clases (49), $y_{o,c}$ es el indicador binario de la clase correcta, y $p_{o,c}$ es la probabilidad predicha.

La estrategia de Transferencia de Aprendizaje se ejecutó en dos etapas. La primera, denominada Fase de Calentamiento (*Warm-up*), entrenó únicamente el módulo *Transformer* y el clasificador denso durante 15 épocas, congelando el *backbone* EfficientNet. Se empleó una tasa de aprendizaje de 1×10^{-3} , con una duración de 270 segundos por época (~1 hora y 10 minutos). Esto estabilizó los pesos del cabezal sin propagar gradientes destructivos hacia la red convolucional. En la segunda etapa, Fase de Ajuste Fino (*Fine-Tuning*), se descongelaron los bloques convolucionales y se redujo la tasa de aprendizaje a 1×10^{-5} . Esta fase abarcó 10 épocas, requiriendo 638 segundos por iteración (~1 hora y 45 minutos), sumando un tiempo total de entrenamiento de 2 horas y 55 minutos. Esta reducción de tasa permitió refinar los filtros para detectar texturas específicas de las enfermedades sin alterar las representaciones aprendidas previamente. La arquitectura final consta de 63,135,700 parámetros totales (59,086,129 entrenables y 4,049,571 fijos). El ciclo de entrenamiento completó las 25 épocas sin detención anticipada (*early stopping*). Se implementó una retrollamada (*ModelCheckpoint*) para monitorear la pérdida de validación (*val_loss*), extrayendo automáticamente los pesos sinápticos de la iteración exacta que minimizó el error, asegurando la evaluación del estado óptimo de generalización.

6.1. Evaluación del Modelo y Métricas de Desempeño

La eficacia del sistema se evaluó utilizando un subconjunto de prueba aislado (10% de los datos originales, 9,800 imágenes) no expuesto durante el entrenamiento, garantizando una medición imparcial de la generalización. Dado el elevado costo computacional de entrenar una arquitectura con más de 63 millones de parámetros, se prescindió de validación cruzada. La estabilidad se garantizó evaluando sobre este extenso conjunto aislado y mediante la estimación de intervalos de confianza, complementado con pruebas in-situ.

El aprendizaje a lo largo de 25 épocas evidenció la convergencia del algoritmo. Como ilustra la

Figura 4, la exactitud de entrenamiento y validación estabilizan su tendencia ascendente alrededor de la época 20. Paralelamente, la pérdida (*categorical crossentropy*) decreció a un mínimo de 0.18. La brecha mínima (< 0.05) entre la pérdida de entrenamiento y validación confirma que el *Dropout* y el aumento de datos dirigido mitigaron el sobreajuste, validando el aprendizaje de patrones estructurales.

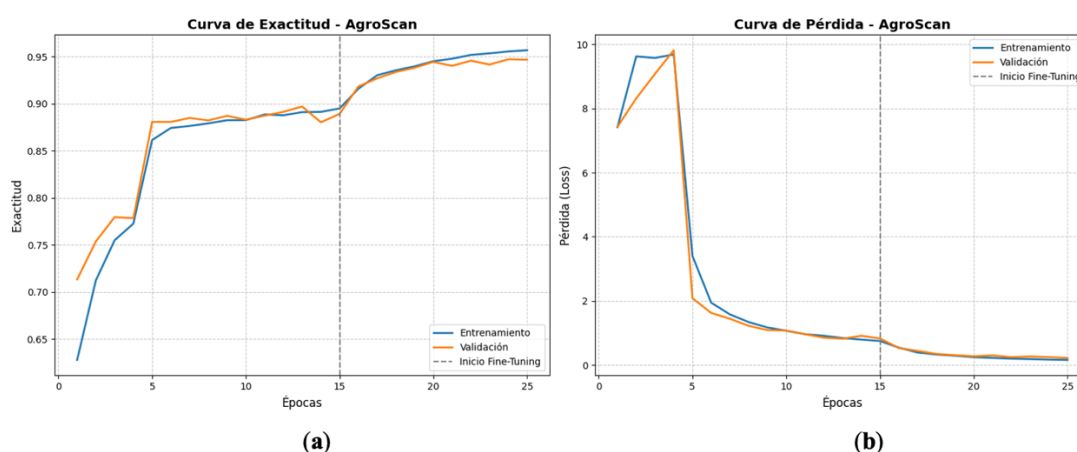


Figura 4. Curvas de aprendizaje del modelo durante el proceso de entrenamiento. (a) Evolución de la exactitud mostrando una tendencia ascendente. (b) Minimización de la función de pérdida. La línea punteada vertical marca el inicio de la fase de Fine-Tuning (época 15), donde se descongelaron las capas del *backbone* para refinar la extracción de características, logrando una convergencia estable sin sobreajuste.

Tras el entrenamiento bajo un entorno determinista (semilla 42), la variabilidad se estimó mediante un procedimiento *bootstrap* no paramétrico ($B = 1000$ muestras), es decir se generaron 1000 muestras con reemplazo del conjunto de pruebas, todas del mismo tamaño que el conjunto de pruebas original. Para calcular el intervalo de confianza se utilizó el método del percentil. El modelo alcanzó una exactitud global del 94.69% (IC 95%: [94.28%, 95.11%]). Aunque nominalmente inferior al 99% reportado en laboratorios, este valor representa un avance empírico en la generalización frente al ruido visual, gracias a los mecanismos de atención global de la arquitectura híbrida.

En la Tabla 3 se compara nuestra propuesta AgroScan con otras arquitecturas. Modelos como MobileNet-V3 y MobileViT presentaron inferencias rápidas (8.36 ms y 5.19 ms) pero saturaron su capacidad espacial, limitando su exactitud ($< 91.5\%$). ResNet-50 mostró confusión de clases por carecer de atención global (90.61%). El modelo ViT-B16 puro se estancó en convergencia ($\sim 60.99\%$) al carecer de extracción local previa. Finalmente, EfficientNet-B0 operó como un extractor local fuerte (92.72%), pero sin el bloque *Transformer* no contextualizó síntomas distribuidos. AgroScan (94.69%) justificó su costo computacional de 9.45 ms por un incremento del 1.97% frente a su *backbone* base y 5.67% frente a MobileViT.

Tabla 3. Rendimiento y tiempo de inferencia de AgroScan frente a arquitecturas de referencia en el conjunto de prueba (9,800 imágenes).

Arquitectura	Exactitud (Accuracy)	Precision (Macro)	Recall (Macro)	F1-Score (Macro)	Tiempo de Inferencia
MobileNet-V3	91.47%	0.9151	0.9144	0.9144	8.36 ms
ResNet-50	90.61%	0.9059	0.9063	0.9061	7.78 ms
ViT-B16 (Puro)	$\sim 60.99\%$	0.6091	0.6097	0.5987	13.45 ms

EfficientNet-B0 (Solo)	92.72%	0.9277	0.9277	0.9263	5.82 ms
MobileViT	89.02%	0.8918	0.8885	0.8902	5.19 ms
AgroScan (Propuesto)	94.69%	0.9478	0.9469	0.9468	9.45 ms

Fuente: Elaboración propia.

AgroScan obtuvo una exactitud de 94.69%, a nivel macro, el sistema registró una Precisión de 94.78%, una Sensibilidad (*Recall*) de 94.69%, y un F1-Score de 94.68% (IC 95% [94.22%, 95.08%]).

Para un análisis granular, la Figura 5 presenta la matriz de confusión sobre las 9,800 imágenes de prueba. Su análisis: a) Invariancia Fenotípica, donde enfermedades con características distintivas (ej. roya del café o tizón de la papa) alcanzan tasas de detección superiores al 97%; b) Mitigación de Confusión, identificando una limitante en el cultivo de maíz con un 4% de error interclase entre mancha foliar y roya por similitud morfológica temprana; y c) Tasa de Falsos Positivos, donde el error crítico de confundir una planta sana con una enferma se mantuvo por debajo del 1.5%.

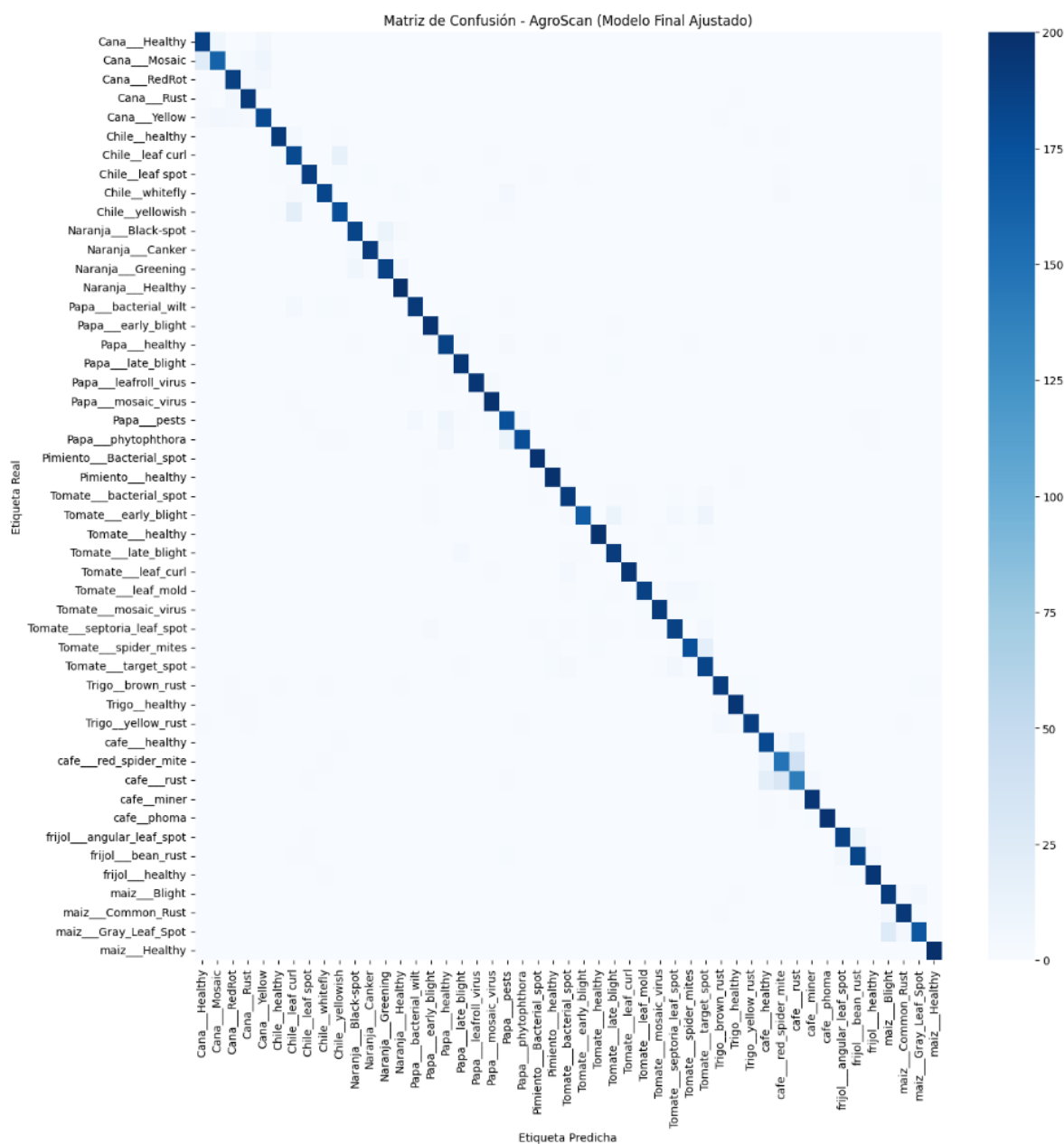


Figura 5. Matriz de confusión normalizada global para las 49 clases objetivo en el conjunto de prueba. La fuerte tendencia diagonal confirma la robustez de la clasificación a través de todos los pares cultivo-enfermedad, reflejando una exactitud superior al 94%.

Finalmente, para evaluar la operatividad frente a la diversidad de enfermedades, la Tabla 4 presenta el reporte de clasificación desglosado por cultivo. Este soporte simétrico de 200 muestras por clase revela la capacidad de generalización del modelo, manteniendo un F1-Score superior al 80% incluso en las clases más retardoras (café, caña de azúcar). La complejidad visual de estas clases radica en la superposición geométrica y el fondo ruidoso que generan las hojas largas y estrechas de la caña de azúcar, así como en la presencia de lesiones pequeñas e irregulares en el café, las cuales son frecuentemente confundibles con reflejos de luz sobre su cutícula cerosa o con necrosis natural. Superar esta barrera fenotípica brinda un respaldo estadístico robusto a las métricas globales del sistema.

Tabla 4. Reporte de Clasificación Exhaustivo por Clase sobre el Conjunto de Prueba Aislado (9,800 imágenes).

Cultivo (N° de clases)	Precision Macro (%)	Recall Macro (%)	F1-Score Promedio (%)	Support
Naranja (4)	97.40	97.38	97.38	800
Pimiento (2)	99.50	99.25	99.38	400
Trigo (3)	96.40	97.50	96.94	600
Tomate (10)	96.84	96.95	96.85	2000
Papa (8)	95.06	96.19	96.04	1600
Maíz (4)	94.65	94.50	94.50	800
Frijol (3)	94.23	94.83	94.46	600
Chile (5)	93.46	91.90	92.62	1000
Caña de azúcar (5)	93.12	92.40	92.50	1000
Café (5)	87.32	87.30	87.30	1000
Promedio Global (49)	94.78	94.69	94.68	9800

Fuente: Elaboración propia.

6.2. Evaluación del pipeline de filtrado binario (modelo portero)

La evaluación sobre las 10,300 imágenes de prueba arrojó una exactitud global del 98.31% (IC 95%: [98.06%, 98.55%]) y un F1-Score macro del 89.83% (IC 95%: [88.32%, 91.22%]), métricas validadas mediante 1,000 iteraciones Bootstrap. Como detalla la Figura 6, destaca una sensibilidad (*Recall*) del 99.64% para la clase “Hoja”, identificando correctamente 9,765 muestras y limitando los falsos negativos a tan solo 35 casos.



Figura 6. Evaluación del modelo 'Portero'. (a) Ejemplo de rechazo de ruido no vegetal. (b) Matriz de confusión sobre 10,300 imágenes de prueba.

No obstante, 139 de 500 muestras atípicas (“No-Hoja”) evadieron el filtro, siendo procesadas por la segunda etapa del sistema. Esta dinámica responde a la Suposición de Mundo Cerrado (*Closed-World Assumption*). El sistema asume axiomáticamente que cualquier objeto validado por el modelo portero corresponde forzosamente a una de las 49 enfermedades.

Este compromiso (*trade-off*) es deliberado: las enfermedades en estadios terminales alteran la morfología foliar, acercando su representación latente al ruido visual. Implementar umbrales de rechazo más restrictivos para bloquear el 100% de la flora atípica incrementaría la tasa de falsos negativos sobre cultivos gravemente enfermos, mermando la utilidad diagnóstica. Se priorizó la sensibilidad, asumiendo la clasificación forzada de elementos ajenos (Figura 7) y delegando la discriminación primaria de objetos evidentes al juicio del agricultor.



Figura 7. Visualización del *trade-off* de Mundo Cerrado. Flora ajena (Cempasúchil) evade el filtro binario y es forzada hacia la categoría “Chile - Leaf Curl” por el clasificador principal.

6.3. Pruebas de integración y validación

Tras la validación en laboratorio, se evaluó la arquitectura en un entorno de despliegue real (*Real-World Testing*). Esta fase enfrentó al sistema a condiciones de campo, incluyendo latencia de red y variabilidad visual no controlada. Se configuró una arquitectura para simular un entorno de producción. El *Frontend* se alojó en Vercel, utilizando su red de entrega de contenido (CDN) para cargar activos estáticos en zonas de conectividad intermitente. Como muestra la Figura 8, esta implementación mantuvo la consistencia visual entre la versión web y la aplicación móvil. El *Backend* de inferencia se ejecutó en un servidor con GPU dedicada, exponiendo sus *endpoints* mediante un túnel HTTPS (ngrok), validando el flujo de datos cifrados desde el dispositivo hasta el motor de clasificación. Posteriormente, se realizó una validación in-situ utilizando un conjunto de prueba de 42 imágenes inéditas capturadas mediante la cámara de un dispositivo móvil, simulando el caso de uso final.

Durante las pruebas, se registraron métricas de rendimiento para validar la viabilidad del sistema en teléfonos de gama media. Dado que la aplicación móvil actúa únicamente como interfaz de adquisición, no requiere cargar tensores ni pesos sinápticos. En consecuencia, el consumo de memoria RAM en el cliente se mantuvo estable entre 60 y 85 MB, dedicados exclusivamente a renderización, activación de cámara y codificación Base64. Al evaluar la latencia de red (*End-to-End*) bajo conectividad celular estándar (3G/4G), el ciclo completo de la petición HTTP POST —transmisión de la imagen, inferencia de 9.45 ms en el servidor y recepción del JSON— registró una latencia promedio de 1.2 a 1.8 segundos. Estas métricas confirman la viabilidad de acceso a diagnósticos precisos sin que el procesamiento local o la memoria limiten al usuario.

El levantamiento de datos se llevó a cabo en parcelas de cultivo en el Estado de Puebla, México. Estas imágenes introdujeron variabilidad ambiental ausente en laboratorio: fluctuaciones de luz solar, sombras proyectadas por el dosel, ruido de fondo (tierra, maleza) y manipulación manual de la hoja.

Frente a este entorno no controlado, el modelo se evaluó mediante un sistema de precisión jerárquica. Se obtuvo una exactitud estricta del 83.3% (35 de 42 muestras), donde la red identificó correctamente tanto la especie del cultivo como la enfermedad específica. Asimismo, se registró una exactitud a nivel de cultivo del 88.1% (37 de 42 muestras), incorporando casos donde se identificó la planta, pero existió confusión en la clasificación de la enfermedad por condiciones de iluminación; únicamente 5 imágenes resultaron en error de clasificación total. La degradación de rendimiento del 11% entre las pruebas de laboratorio (94.69%) y la exactitud de campo (83.3%) corresponde a la brecha de transferencia (*reality gap*) documentada en el despliegue de redes neuronales. Este margen demuestra que la arquitectura no presenta sobreajuste y retiene capacidad diagnóstica operativa.

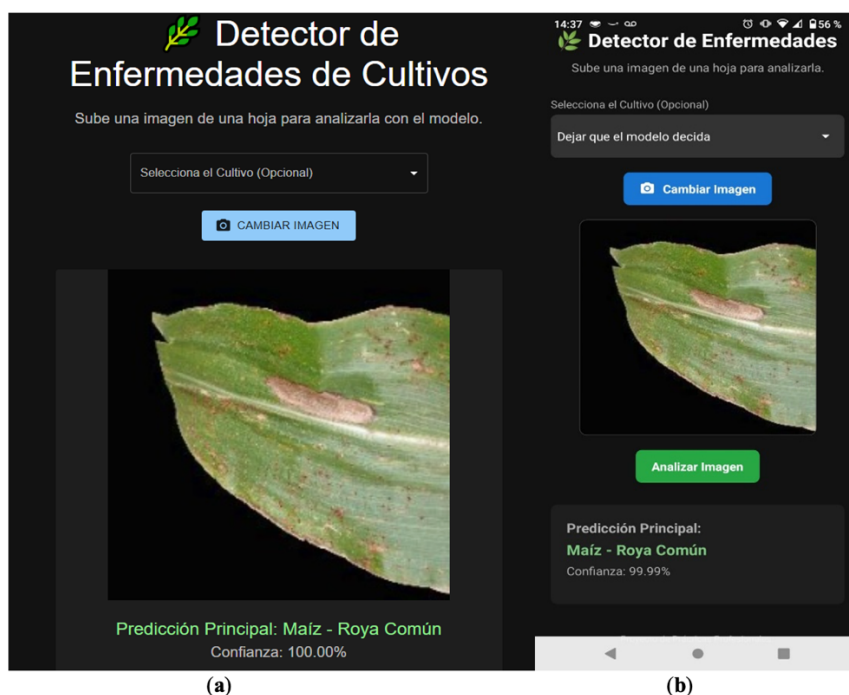


Figura 8. Implementación multiplataforma del sistema de diagnóstico. **(a)** Interfaz de la aplicación Web ejecutándose en un navegador de escritorio. **(b)** Interfaz de la aplicación Móvil (Android) desplegada en tiempo real. Ambas plataformas consumen la misma API REST centralizada, garantizando consistencia en el diagnóstico (confianza >99.9%) independientemente del dispositivo de acceso.



Figura 9. Ejemplos de inferencia en campo ilustrando el comportamiento del modelo frente al ruido ambiental. **(a)** Éxito estricto: Diagnóstico correcto de Maíz Sano con alta confianza (93.76%). **(b)** Caso difícil: Diagnóstico correcto de Maíz Sano, pero con margen de confianza reducido (47.05%) debido a variaciones de iluminación en el campo. **(c)** Confusión de cultivo (Error): Falso diagnóstico de una hoja de maíz clasificada como Caña de Azúcar con Mosaico (58.19%) debido a similitudes de textura, relegando la clase real a una segunda probabilidad (41.57%).

Debido al tamaño reducido de la muestra y a las limitaciones logísticas de recolección, esta evaluación in-situ se define estrictamente como una prueba de concepto preliminar y una evidencia inicial de viabilidad, no como una validación estadística a gran escala. Para brindar transparencia, la totalidad de las 42 muestras correspondieron exclusivamente al cultivo de maíz, abarcando las enfermedades prevalentes identificadas físicamente durante el levantamiento. Si bien el rendimiento demuestra resiliencia ante el entorno no estructurado para esta especie, las conclusiones definitivas sobre la robustez operativa requerirán que incorporen los cultivos restantes soportados por la arquitectura.

Para ilustrar los límites operativos frente a la variabilidad del entorno real, la Figura 9 presenta un desglose visual de las inferencias realizadas. Esta figura presenta el comportamiento de la red neuronal ante el ruido ambiental, categorizando las predicciones en tres escenarios: éxitos estrictos, éxitos con confianza reducida por oclusión o iluminación, y errores derivados de similitudes fenotípicas entre cultivos.

El análisis de estos resultados permite identificar la persistencia de un sesgo de dominio (*domain shift bias*) derivado de la composición del conjunto de entrenamiento, del cual el 70% proviene de condiciones controladas

(*PlantVillage*). Aunque el aumento de datos mitigó este efecto, el modelo mantiene susceptibilidad ante fondos de alta complejidad visual o condiciones de iluminación extrema no representadas en el corpus original.

Finalmente, se reafirma la limitación relacionada con la Suposición de Mundo Cerrado (*Closed-World Assumption*). Dado que el clasificador principal evalúa forzosamente las entradas dentro de las 49 enfermedades aprendidas, el sistema depende de la eficacia del “Modelo Portero” (detallado en la Sección 6.3) para filtrar la flora ajena. Si un objeto fuera de distribución evade este filtro primario, será clasificado erróneamente. Esta decisión de diseño prioriza la minimización de falsos negativos en estadios de infección terminales, delegando la discriminación final de objetos evidentes al juicio visual del agricultor.

7. Conclusiones y Trabajo Futuro

Esta investigación confirma que integrar mecanismos de atención global en arquitecturas convolucionales supera las limitaciones empíricas en el diagnóstico de enfermedades agrícolas. La arquitectura híbrida propuesta logra una extracción complementaria: EfficientNet-B0 identifica patrones texturales locales (necrosis y clorosis), mientras el módulo *Transformer* aporta una comprensión contextual global de la hoja. Esta capacidad de relacionar características distantes permitió alcanzar una exactitud del 94.69%, reduciendo los falsos positivos comunes en redes convencionales.

El aporte ingenieril de esta investigación. Radica en hacer factible, democratizar del acceso a la agricultura de precisión. La arquitectura cliente-servidor desacopla la carga computacional de la interfaz, haciéndola de utilidad en un contexto rural. Para mitigar la conectividad limitada, el sistema serializa las imágenes a Base64, optimizando el ancho de banda en redes 3G/4G. Además, el diseño de una interfaz minimalista centrada en el diagnóstico directo permite a usuarios con dispositivos de gama media operar el sistema sin barreras técnicas.

Las pruebas de campo demostraron la resiliencia operativa del sistema ante ruido visual (sombras, oclusiones y desenfoque). La integración del modelo "portero" para el filtrado primario y la aplicación de aumento de datos (*data augmentation*) dotaron a la herramienta de la robustez necesaria para asistir en tiempo real, operando de manera estructurada bajo la suposición de mundo cerrado (*closed-world assumption*).

El despliegue de esta tecnología plantea consideraciones éticas. Un falso positivo induciría la aplicación innecesaria de agroquímicos (con impactos ambientales y económicos), mientras un falso negativo vulneraría el cultivo. Para mitigar riesgos, AgroScan se conceptualiza estrictamente como un Sistema de Soporte de Decisiones (*Decision Support System*). Actúa como asistente automatizado que complementa el conocimiento empírico del agricultor, asegurando que la validación diagnóstica y la toma de decisiones permanezcan bajo supervisión humana (*human-in-the-loop*).

Como trabajo futuro, se proyecta la migración hacia el procesamiento en el dispositivo (Edge AI). Esto requiere técnicas de cuantización de pesos y poda neuronal para comprimir el modelo, habilitando su ejecución offline en zonas sin internet. Paralelamente, para superar el escepticismo ante modelos de “caja negra”, se incorporarán módulos de Inteligencia Artificial Explicable (XAI). Integrar mapas de calor (Grad-CAM) permitirá al agricultor visualizar las regiones foliares que motivaron el diagnóstico de las enfermedades, aumentando la interpretabilidad.

A nivel de infraestructura, se expandirá el *Backend* para enlazar la clasificación con bases de datos agropecuarias, devolviendo recomendaciones de tratamiento basadas en normativas vigentes y disponibilidad regional de insumos, consolidando un sistema experto integral.

Finalmente, para resolver la susceptibilidad al sesgo de dominio (*domain shift bias*) documentada *in-situ*, es prioritario ejecutar campañas de recolección fotográfica en parcelas. Construir un conjunto de datos nativo permitirá realizar un ajuste fino (*Fine-Tuning*) definitivo, eliminando la dependencia de repositorios de laboratorio y demostrando la necesidad de adaptar las herramientas de inteligencia artificial a las necesidades de los usuarios en entornos reales, una tendencia de adopción tecnológica y aprendizaje documentada recientemente en esta misma revista [19].

8. Referencias

- [1] Secretaría de Agricultura, Ganadería, Desarrollo Rural, Pesca y Alimentación [SAGARPA]. (2017). *Planeación Agrícola Nacional 2017-2030*. Gobierno de México. <https://www.gob.mx/agricultura/acciones-y-programas/planeacion-agricola-nacional-2017-2030-126813>

- [2] Organización de las Naciones Unidas para la Alimentación y la Agricultura [FAO]. (2011). *Seguridad alimentaria y nutricional: Conceptos básicos*. <https://www.fao.org/3/at772s/at772s.pdf>
- [3] IBM. (2023). *¿Qué son las redes neuronales convolucionales?* <https://www.ibm.com/mx-es/topics/convolutional-neural-networks>
- [4] TensorFlow. (2024). *Transferencia de aprendizaje y ajuste fino*. https://www.tensorflow.org/tutorials/images/transfer_learning?hl=es-419
- [5] Hugging Face. (2024). *Vision Transformer (ViT)*. https://huggingface.co/docs/transformers/model_doc/vit
- [6] Keras. (2020). *EfficientNet B0 to B7*. <https://keras.io/api/applications/efficientnet/>
- [7] Google Research. (2020). *Transformers for Image Recognition at Scale*. <https://blog.research.google/2020/12/transformers-for-image-recognition-at.html>
- [8] Mohanty, S. P., Hughes, D. P., Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Frontiers in Plant Science*, 7, 1-10. <https://doi.org/10.3389/fpls.2016.01419>
- [9] Saleem, M. H., Potgieter, J., Arif, K. M. (2019). Plant disease detection and classification by deep learning. *Plants*, 8 (11), 1-22. <https://doi.org/10.3390/plants8110468>
- [10] Li, X., Li, S. (2022). Transformer help CNN see better: A lightweight hybrid apple disease identification model based on Transformers. *Agriculture*, 12 (6), 1-16. <https://doi.org/10.3390/agriculture12060884>
- [11] De Silva, M., Brown, D. (2023). Multispectral plant disease detection with Vision Transformer–Convolutional Neural Network hybrid approaches. *Sensors*, 23 (20), 1-22. <https://doi.org/10.3390/s23208531>
- [12] Ashurov, A. Y., Al-Gaashani, M. S. A. M., Samee, N. A., Alkanhel, R., Atteia, G., Abdallah, H. A., Muthanna, M. S. A. (2025). Enhancing plant disease detection through deep learning: a Depthwise CNN with squeeze and excitation integration and residual skip connections. *Frontiers in Plant Science*, 15, 01-16. <https://doi.org/10.3389/fpls.2024.1505857>
- [13] Kalpana, P., Gera, P., Alabdulkreem, E., Quasim, M. T., Baili, J., Cho, Y., Nam, Y. (2025). An ensemble heterogeneous transformer model for an effective diagnosis of multiple plant diseases. *Frontiers in Plant Science*, 16, 01-22. <https://doi.org/10.3389/fpls.2025.1693095>
- [14] Kaggle. (2023). *Leaf vs Non-Leaf Images Dataset*. Kaggle Open Datasets. <https://www.kaggle.com/datasets/robiulhasanjisan/leaf-vs-non-leaf-images/data>
- [15] Amazon Web Services. (2023). *¿Qué es una API de RESTful?* <https://aws.amazon.com/es/what-is/restful-api/>
- [16] Kaggle. (2019). *PlantVillage Dataset*. Kaggle Open Datasets. <https://www.kaggle.com/datasets/emmarex/plantdisease>
- [17] TensorFlow. (2024). *Aumento de datos (Data augmentation)*. https://www.tensorflow.org/tutorials/images/data_augmentation?hl=es-419
- [18] TensorFlow. (2024). *Precisión combinada (Mixed Precision)*. https://www.tensorflow.org/guide/mixed_precision?hl=es-419
- [19] Lozano Ramirez, M. C. (2025). El profesorado y el uso de la inteligencia artificial (IA) como proceso de aprendizaje. *Revista de Investigación en Tecnologías de la Información (RITI)*, 13 (30), 1-8. <https://doi.org/10.36825/RITI.13.30.001>