

Approach for Detecting *Phytophthora infestans* in Peruvian Potato Crops Using EfficientNet-B0, ResNet-50, and MobileNetV2

Yoimer Dávila¹; Steve Castillo¹; Lenis Wong¹

¹ Universidad Peruana de Ciencias Aplicadas, Perú, u20201b973@upc.edu.pe, u202121679@upc.edu.pe, pcsilewo@upc.edu.pe

Abstract—Late blight (*Phytophthora infestans*) is a major threat to potato crops throughout Peru due to its rapid spread and severe impact. Traditional inspection methods are highly subjective, relying heavily on the expertise of farmers and varying field conditions. Due to this challenge, we propose an approach for late blight detection on potato leaves using semantic segmentation with a U-Net architecture and three different encoders: ResNet-50, EfficientNet-B0, and MobileNetV2. The approach is developed in four phases: (i) dataset acquisition from International Potato Center, (ii) dataset preprocessing, (iii) model training, and (iv) performance evaluation. All models focus on processing potato leaf images to generate two segmentation outputs: the full leaf and the lesion-affected regions. Results show that ResNet-50 achieved the best performance, with IoU values of 0.94 for leaf segmentation and 0.89 for lesion segmentation, demonstrating higher stability and accuracy compared to the other two models.

Keywords—late blight, *phytophthora infestans*, potato, semantic segmentation, U-Net, ResNet-50.

Enfoque para la Detección de *Phytophthora infestans* en Cultivos de Papa en Perú Aplicando EfficientNet-B0, ResNet-50 y MobileNetV2

Yoimer Dávila¹; Steve Castillo¹; Lenis Wong¹

¹ Universidad Peruana de Ciencias Aplicadas, Perú, u20201b973@upc.edu.pe, u202121679@upc.edu.pe, pcsilewo@upc.edu.pe

Resumen– El tizón tardío (*Phytophthora infestans*) es una gran amenaza para los cultivos de papa en todo el Perú, debido a su rápida propagación y a su gran impacto. Sabemos que los métodos de revisión tradicionales son muy subjetivos y que dependen mayormente de la experiencia de los agricultores expertos y de las diferentes condiciones de campo. Por ello, se propone un enfoque para la detección del tizón tardío en hojas de papa usando la segmentación semántica utilizando la arquitectura U-Net con tres codificadores diferentes: ResNet-50, EfficientNet-B0 y MobileNetV2. El enfoque se desarrolla en 4 fases: (i) la adquisición del conjunto de datos provenientes del Centro Internacional de la Papa, (ii) el preprocesamiento del conjunto de datos, (iii) el entrenamiento de los modelos y (iv) la evaluación de su desempeño. Todos los modelos se enfocan en el uso de imágenes de hojas de papa con dos resultados de segmentación: hoja completa y áreas de lesión afectadas. Los resultados mostraron que ResNet-50 logró el mejor rendimiento con valores de IoU de 0.94 para la segmentación de hojas y 0.89 para la segmentación de las lesiones, lo que demuestra una estabilidad y precisión superiores en comparación con los otros dos modelos.

Palabras clave-- tizón tardío, *phytophthora infestans*, papa, segmentación semántica, U-Net, ResNet-50.

I. INTRODUCCIÓN

La papa (*Solanum tuberosum*) constituye uno de los cultivos más importantes a nivel mundial, siendo esencial para la seguridad alimentaria y el desarrollo económico de numerosos países, especialmente en regiones andinas como el Perú [1]. Sin embargo, este cultivo enfrenta múltiples amenazas fitosanitarias, entre las cuales destaca el tizón tardío, causado por el hongo *Phytophthora infestans*, considerado la enfermedad más devastadora de la papa [2]. Su rápida propagación puede ocasionar pérdidas de hasta el 100% de la producción, generando un impacto económico considerable y poniendo en riesgo la sostenibilidad de comunidades agrícolas dependientes de este tubérculo [3].

Tradicionalmente, la identificación de *Phytophthora infestans* se ha realizado mediante inspecciones visuales en campo, dependientes de la experiencia del agricultor. Este enfoque resulta limitado debido a la subjetividad y al diagnóstico tardío, cuando el daño ya es irreversible [4]. En este contexto, la falta de herramientas que integren un análisis agronómico preciso dificulta estimar la severidad real de la infección en

campo. Cuantificar exactamente el área foliar afectada es un paso crítico en las prácticas agrícolas para tomar decisiones informadas sobre el manejo fitosanitario, como la aplicación focalizada de fungicidas. La variabilidad climática y la limitada disponibilidad de asistencia técnica agravan aún más el problema, haciendo indispensable el uso de sistemas automatizados de apoyo en la toma de decisiones [5].

Frente a esta problemática, diversos estudios han explorado el uso de técnicas de inteligencia artificial (IA) y visión por computadora para el diagnóstico de enfermedades agrícolas [6]. En particular, modelos de redes neuronales convolucionales (CNN) como ResNet, EfficientNet o Inception han mostrado resultados prometedores en la clasificación de imágenes de cultivos afectados por diferentes patologías [7]. Asimismo, arquitecturas de detección de objetos como Faster R-CNN [7] y MobileNetV2 [8] han demostrado su capacidad para identificar patrones visuales asociados a enfermedades vegetales con altos niveles de precisión. No obstante, aún persiste una brecha en la evaluación comparativa de múltiples arquitecturas en escenarios agrícolas peruanos, lo cual limita la detección temprana de esta enfermedad.

Por ello, este estudio propone un enfoque basado en cuatro arquitecturas de Deep Learning para la detección temprana de *Phytophthora infestans* en cultivos de papa en Perú. Con la finalidad de identificar el modelo más eficiente. La propuesta se articula en cuatro fases: (i) adquisición del dataset (ii) preprocesamiento de datos, (iii) entrenamiento de modelos, y (iv) evaluación de resultados.

Este artículo se organiza de la siguiente manera. En la sección II se presenta la revisión de la literatura. Luego, la sección III describe los materiales y métodos. Posteriormente, la sección IV expone los resultados obtenidos y su respectiva discusión. Finalmente, la sección V presenta las conclusiones y líneas de investigación futura.

II. TRABAJOS RELACIONADOS

En la literatura se han propuesto diversos enfoques orientados a la detección y clasificación de enfermedades en cultivos agrícolas mediante técnicas de Deep Learning. Modelos como EfficientRMT-Net [11], Mask R-CNN [15] y ResNet-50 [11], entre otros (Tabla I). De manera similar, otras

investigaciones han empleado técnicas de *Deep Learning* en otros cultivos como: maíz [18], café [17] y yuca [21]. Por otro lado, se observa una tendencia en los estudios existentes a emplear enfoques de clasificación multiclase centrados en distintas enfermedades, con un uso predominante de imágenes RGB extraídas de *datasets* públicos como *PlantVillage* [11] para el entrenamiento, mientras que para la validación se recurre a imágenes capturadas en condiciones reales.

Respecto a investigaciones específicas sobre la detección de enfermedades en cultivos de papa, en [9] propusieron un modelo de fusión multidimensional *Atrous-CNN* para identificar enfermedades en papa considerando 4 clases: “tizón tardío”, “tizón temprano”, “antracnosis” y “sana” en un *dataset* de 126 puntos hiperespectrales, logrando 99,87 % de precisión. Por su parte, en [10] emplearon *ECLF-CS*, *YOLO v3* y *Fastter R-CNN* para la clasificación de hojas anómalas afectadas con PVY en 1003 imágenes, obteniendo una precisión del 96,7%. Con *EfficientRMT-Net* [11] combinan *ResNet-50* y *ViT* para la detección de tizón tardío en 10800 imágenes, con una precisión del 99,12%. También, la arquitectura *CNN* que [12] propusieron sobre 12912 imágenes logra obtener un 99,53% de precisión en la detección de tizón tardío. Por otra parte, en [13] utilizan *XGBoost* y *LightGBM* para la clasificación en 6 niveles de *Verticillium*: “nula”, “leve”, “moderada”, “severa”, “muy severa” y “terminal” en 212 muestras etiquetadas extraídas de imágenes multiespectrales, alcanzando una precisión del 93%. Por su parte, en [14] emplean *YOLOv5* y un modelo propio llamado *PDDCNN* para la detección de tizón tardío en 4062 imágenes, obteniendo una precisión del 99,75%. En [15] utilizan un marco integrado de tres etapas combinando *Mask R-CNN*, *VGG16* y *U-Net* para la detección de tizón tardío en 1800 imágenes, logrando una precisión del 95,33%. De igual forma, con *PLDPNet* [16] utilizan *U-Net*, *VGG19*, *Inception-V3* y *ViT* para predecir el tizón tardío en 2152 imágenes, con una precisión del 98,66%.

TABLA I
DL EN LA DETECCIÓN DE ENFERMEDADES DE PAPA

Algoritmo	Clase	Dataset	ACC	Ref.
Atrous-CNN	4	126	95%	[9]
ECLF-CS	2	1003	96,7%	[10]
EfficientRMT-Net	3	10800	99,12%	[11]
Arquitectura CNN propuesta	3	12912	99,53%	[12]
XGBoost & LightGBM	6	212	93%	[13]
PDDCNN	3	4062	99,75%	[14]
VGG16	3	1800	95,33%	[15]

Con respecto a las investigaciones sobre la detección de enfermedades en hojas de otros cultivos, [17] utilizan *Mask R-CNN*, *U-Net* & *PSPNet* y *ResNet-50* & *ResNet-101* en un marco integrado para la identificación de minador y roya en “café”, alcanzando una precisión del 99,54%. Con *MMF-Net* [18] se

basaron en los modelos *ResNeXt*, *VGG-16* y *AlexNet*, para la detección temprana de tizón, roya común y mancha gris en el “maíz” en 4188 imágenes y alrededor de 4000 registros ambientales, logrando una precisión del 99,23%. También, [19] en su marco eficiente con sensores IoT y *MobileNet V1* & *V2*, *VGG-16*, *VGG-19* e *Inception* para la detección de tizón bacteriano y mancha marrón en 120 imágenes de “arroz”, alcanza una precisión del 97,27%. Por su parte, [20] integran una arquitectura *CycleGAN* con *VGG19*, *ResNet50* y *EfficientNetB0* para la detección de la mancha y rizado de la “pera” en 3505 imágenes, logrando una precisión del 86,59%. Con *DGCNN* [21] combinan *DGP* con una *CNN* en un *kernel* híbrido para clasificar la mancha marrón y la enfermedad del mosaico de la “yuca” en 1656 puntos de datos espectrales, alcanzando un 90,1% de precisión. Por su parte, el modelo propio *Conv-3 DCNN* [22] desarrollado para la detección de costra y podredumbre negra en la “manzana” en 3171 imágenes, alcanzó una precisión del 98%. También, *Maize-YOLO* [23] combinan *YOLOv7* con los módulos *CSPResNeXt-50* & *VoVGSCSP* para la detección de plagas en el “maíz” en 4533 imágenes, logrando una precisión del 99,7%.

III. MATERIALES Y MÉTODOS

En esta sección se presenta el enfoque propuesto donde se han considerado tres modelos que han tenido mejor rendimiento en la literatura: *EfficientNet-B0*, *ResNet-50* y *MobileNetV2*.

A. Adquisición del dataset

El *dataset* utilizado en esta investigación se obtuvo a partir de una búsqueda exhaustiva de repositorios de datos públicos especializados en agricultura y fitopatología. Como resultado, se seleccionó un conjunto de datos proporcionado por el Centro Internacional de la Papa (CIP), el cual es de acceso público [24]. Este incluye imágenes recolectadas tanto en Perú como en África, que abarcan diferentes condiciones de cultivo y tipos de enfermedades foliares, como tizón tardío, tizón temprano (*Alternaria*) y virus de la papa. Para esta investigación, se seleccionaron 275 imágenes correspondientes al contexto peruano, específicamente clasificadas en hojas afectadas por “tizón tardío” (254 imágenes) y “hojas sanas” (21 imágenes). En la Fig. 1 se muestra los 2 tipos de imágenes seleccionadas del *dataset*: tizón tardío (Fig. 1a) y hojas sanas (Fig. 1b) [24].

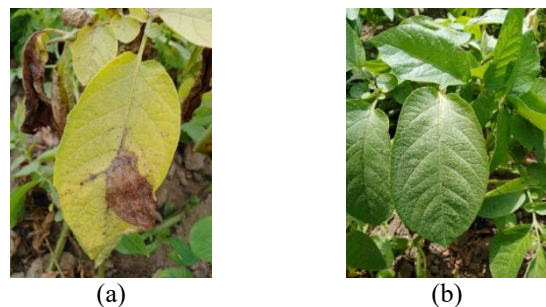


Fig. 1 Tipo de imágenes seleccionadas: tizón tardío (a) y hojas sanas (b)

B. Preprocesamiento de datos

El proceso de preprocesamiento de las imágenes se llevó a cabo en dos etapas principales: la segmentación de las hojas y el aumento de datos. En primer lugar, se abordó la segmentación de las hojas para poder trabajar con una clasificación más detallada, que incluyera los niveles de severidad del tizón tardío según el área que afecta la lesión a la hoja principal. A continuación, se implementó un procedimiento de aumento de datos para equilibrar las clases y mejorar la capacidad de generalización del modelo.

1. Segmentación de las hojas

a) Etiquetado manual utilizando la herramienta *LabelMe*, delimitando las regiones correspondientes a las hojas en todas las imágenes del *dataset*.

b) Creación de máscaras binarias utilizando los archivos de etiquetas generados que delimitan las hojas dentro de cada imagen (Fig. 2).

c) Re-escalamiento de las imágenes a un tamaño de 256x256 píxeles para asegurar la compatibilidad con la arquitectura del modelo de segmentación de lesiones.

d) Aplicación de un modelo segmentación enfocada en identificar las zonas afectadas utilizando un modelo de segmentación *U-Net*, entrenado específicamente para reconocer lesiones de tizón tardío.

e) Almacenamiento de las predicciones generadas sobre las lesiones como mascarar binarias (Fig. 3). Algunas de ellas fueron corregidas utilizando herramientas de edición de imágenes.



(a)



(b)

Fig. 2 Segmentación de la hoja: hoja en fondo (a) y mascara (b)



(a)



(b)

Fig. 3 Tipo de imágenes seleccionadas: tizón tardío (a) y hojas sanas (b)

f) Cálculo del porcentaje de área afectada en cada imagen utilizando las máscaras de hoja y lesión. Con base en este valor,

las imágenes con presencia de tizón tardío se clasificaron en tres niveles de severidad: leve, moderada y severa.

g) Generación de un archivo de etiquetas que incluye el porcentaje de área afectada y la categoría de severidad correspondiente por cada imagen del *dataset*.

2. Aumento de datos

A partir de las etiquetas obtenidas, se siguió el siguiente flujo para aumentar las imágenes: 1) Contar el número de imágenes por clase utilizando la biblioteca *pandas*. 2) Aplicar un algoritmo de balanceo para equilibrar las clases con un límite máximo de diez aumentos por imagen. 3) Determinar las cantidades máximas de imágenes por clase, como se muestra en la Tabla II. 4) Aplicar el aumento de datos para alcanzar el máximo establecido para cada clase. Los aumentos se efectuaron mediante la biblioteca *Albumentations*, utilizando transformaciones como *HorizontalFlip*, *VerticalFlip*, *Affine*, *RandomRotate90*, *ChannelShuffle*, *GaussianBlur*, *Perspective*, *CoarseDropout*, *HueSaturationValue* y *RandomBrightnessContrast*.

TABLA II
CANTIDAD DE IMÁGENES POR CLASE

Id	Clase	Qty.	Aug.	Total
0	Sanas	21	210	231
1	Tizón tardío leve	91	182	273
2	Tizón tardío moderado	123	123	246

C. Entrenamiento de modelos

Para la experimentación, se empleó un entorno en *Kaggle Notebook*, equipado con una *GPU NVIDIA Tesla T4* de 16 GB de *VRAM* y *Python* 3.11.13. El desarrollo de los modelos de aprendizaje profundo se realizó utilizando *Keras* 3.8.0 como *framework*, con *TensorFlow* 2.18.0 como *backend*. Para garantizar la reproducibilidad técnica, el entrenamiento se configuró con un tamaño de lote (batch size) de 16, empleando el optimizador Adam con una tasa de aprendizaje inicial de 0.0001. Al tratar la segmentación de la hoja y la lesión como tareas independientes de clasificación a nivel de píxel, se implementó una función de pérdida híbrida que combina la Entropía Cruzada Binaria (BCE) y la pérdida Dice para mejorar el manejo del desbalance de clases espaciales en las imágenes. Un aspecto crucial de esta configuración es la asignación de pesos diferenciados a las salidas de la red durante la compilación. Se otorgó un factor de 2.0 a la pérdida de la lesión frente a un factor de 1.0 para la hoja, lo que obliga a la arquitectura a penalizar con mayor severidad los errores en la detección de las áreas enfermas y prioriza la sensibilidad en la identificación de *Phytophthora infestans*.

Para este estudio, se seleccionaron los modelos que han demostrado un rendimiento destacado en trabajos previos de la literatura. Esta elección se basó en su capacidad para abordar de manera efectiva el problema planteado, así como en su historial de éxito en tareas similares:

1) *EfficientNet-B0*: Es una arquitectura de redes neuronales donde usa un método escalado compuesto que uniformiza la profundidad, la resolución y el ancho de red para

mejorar el rendimiento usando pocos recursos. Su implementación comúnmente es en *TensorFlow/Keras* y *PyTorch* [25].

2) *ResNet-50*: Es una red neuronal convolucional de 50 capas para DL usando conexiones residuales, también utiliza *TensorFlow/Keras* y *PyTorch*. Se usa comúnmente para la detección de objetos. [26].

3) *MobileNetV2*: Es una red neuronal convolucional diseñada para ser ligera y eficiente, es ideal para usarse en dispositivos con recursos limitados o móviles. Usa *TensorFlow Lite* y *PyTorch Mobile*. [27].

En la Fig. 4 se presenta el modelo propuesto basado en una arquitectura *U-Net* adaptada con *backbones* (encoders) preentrenados para la tarea de segmentación semántica multisalida. En la fase de codificación (encoder), se eliminan las capas superiores de clasificación (include_top=False) y se utilizan los mapas de características intermedios del backbone seleccionando de manera específica las capas de skip connections y el bottleneck según la arquitectura empleada. Esta selección es metodológicamente crítica para la detección fitosanitaria; al concatenar directamente mapas de alta resolución desde el codificador hacia el decodificador, se mitiga drásticamente la degradación espacial generada por las operaciones de agrupamiento (pooling). Consecuentemente, el modelo preserva los detalles finos de la morfología foliar y reconstruye con mayor fidelidad los bordes irregulares y difusos que caracterizan a las lesiones necróticas por *Phytophthora infestans*. Luego, en la fase de decodificación (*decoder*), se aplican operaciones de *upsampling* y concatenación con las *skip connections* correspondientes, seguidas de bloques convolucionales que reconstruyen progresivamente la resolución espacial. Finalmente, el modelo incluye un cabezal de salida compartido que genera dos mapas binarios mediante convoluciones con activación sigmoide: uno para la segmentación de la hoja y otro para la segmentación de la lesión.

En la Fig. 5 se muestra un flujo de segmentación semántica utilizadas para el preprocesamiento de las imágenes, la entrada procesa la imagen y esta genera dos máscaras, donde la primera es la que delimita la región total de la hoja mientras que la segunda identifica específicamente el área donde la hoja presenta la lesión. Este enfoque nos ayuda a que los modelos de clasificación centren su análisis y así eliminando el “ruido” y mejorando así la precisión del diagnóstico.

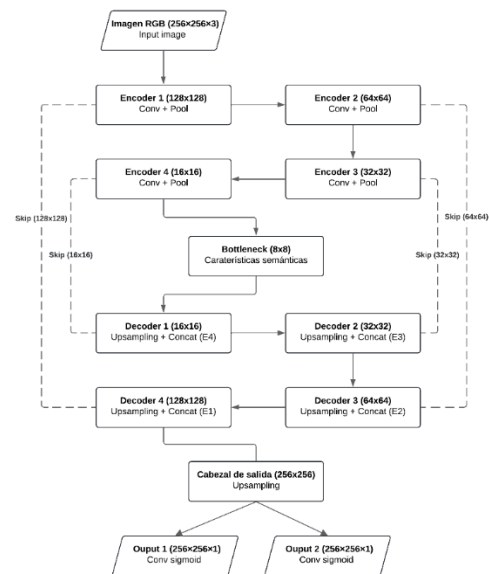


Fig. 4 Arquitectura del modelo *U-Net* con *backbones* preentrenados y doble salida (hoja y lesión)

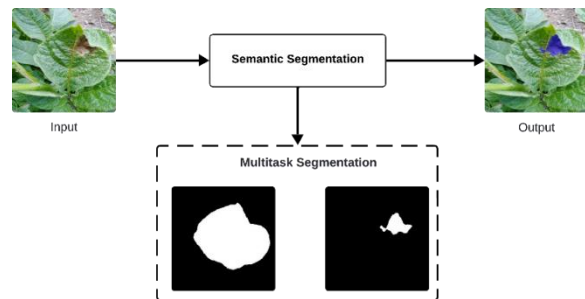


Fig. 5 Proceso de identificación de la hoja y lesión de la papa.

D. Evaluación de desempeño

Las métricas utilizadas para evaluar el desempeño del modelo multisalida de segmentación (hoja y lesión) incluyen: *Precision*, *Recall*, Intersección sobre Unión (*IoU*) y Puntuación *F1*.

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$IoU = \frac{TP}{TP + FN + FP} \quad (3)$$

$$F1 = \frac{2TP}{2TP + FP + FN} \quad (4)$$

Donde, TP es el número de muestras positivas correctamente predichas; TN es el número de muestras negativas correctamente predichas; FP es el número de

muestras negativas incorrectamente predichas como positivas; FN es el número de muestras positivas incorrectamente predichas como negativas.

Estas métricas permiten evaluar la capacidad del modelo para identificar correctamente las regiones objetivo y cuantificar la similitud entre la segmentación predicha y la real. La *Precision* (1) indica la proporción de píxeles predichos como positivos que son realmente positivos, mientras que el *Recall* (2) refleja la capacidad del modelo para detectar todos los píxeles positivos. *IoU* (3) mide la similitud espacial entre segmentaciones predicha y real. Finalmente, la *F1* (4) combina *Precision* (1) y *Recall* (2) mediante su media armónica, ofreciendo un balance entre exactitud y sensibilidad. Todas estas toman valores entre 0 y 1, donde valores más cercanos a 1 indican un mejor desempeño del modelo.

IV. RESULTADOS Y DISCUSIÓN

Para evaluar el desempeño de los *backbones* *ResNet-50*, *EfficientNetB0* y *MobileNetV2* utilizados para la arquitectura *U-Net* multitarea, se analizaron dos escenarios: la “segmentación de la hoja” completa y la “segmentación de la lesión”.

En la Fig. 6 se presenta las métricas *IoU* que se obtuvo de la evolución de los tres modelos para la “segmentación de la hoja”. Donde se evidencia que *ResNet-50* inicia con un rendimiento más bajo respecto a los demás modelos, pero a partir de las épocas 10 a 15 muestra un crecimiento notablemente más acelerado. Este impulso le permite no solo alcanzar, sino también igualar el desempeño que inicialmente mostraban los otros modelos. Con el avance del entrenamiento, todos los modelos tienden a converger en el rango de 0.90 a 0.95; sin embargo, al momento de la estabilización, *ResNet-50* se posiciona con el mejor desempeño general, superando a los otros modelos.

En la Fig. 7 se muestran las métricas *IoU* que se obtuvo para la “segmentación de la lesión” de la hoja. En este caso, aunque *MobileNetV2* muestra un desempeño inicial más alto, el foco relevante es el comportamiento de *ResNet-50*. A lo largo de las primeras épocas mantiene valores moderados, pero alrededor de la época 20-22 experimenta un crecimiento claramente acelerado, alcanzando niveles cercanos a 0.87-0.89. Este aumento marcado contrasta con la inestabilidad observada en *EfficientNetB0*, cuyas oscilaciones tempranas son más pronunciadas. Conforme avanza el entrenamiento, especialmente después de la época 50, las variaciones de *EfficientNetB0* se atenúan y los tres modelos convergen hacia un rendimiento similar en el rango de 0.88-0.89. Aun así, el comportamiento sostenido y el crecimiento consistente de *ResNet-50* lo posicionan como el modelo con la progresión más sólida dentro del proceso evaluado.

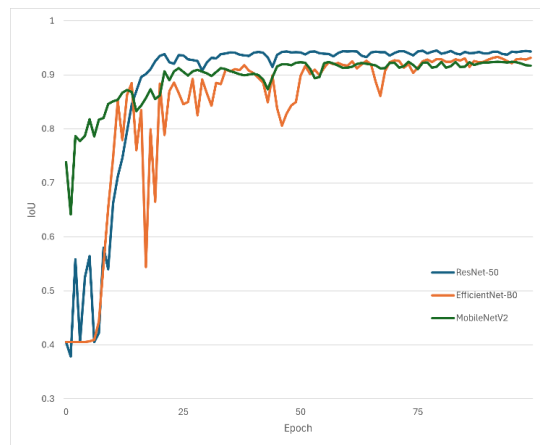


Fig. 6 Métrica *IoU* de cada *backbone* para la segmentación de la hoja.



Fig. 7 Métrica *IoU* de cada *backbone* para la segmentación de la lesión.

Las Tabla III y Tabla IV presentan las métricas definidas de los tres modelos para la “segmentación de la hoja” y la “segmentación de la lesión” en la hoja. En la Tabla III se evidencia que *ResNet-50* mostró el mejor rendimiento global con un *IoU* de 0.94, *Precision* de 0.98, *Recall* de 0.92 y *F1* de 0.92. Esto indica que el modelo es capaz de identificar correctamente la mayoría de los píxeles correspondientes a la hoja con una baja tasa de falsos positivos y una alta capacidad para detectar la región objetivo. En la Tabla IV se observa una disminución en el desempeño general debido a la complejidad morfológica de las lesiones. Sin embargo, *ResNet-50* lidera con un *IoU* de 0.89 y una *Precision* de 0.96. Desde una perspectiva agronómica, esta alta precisión en la segmentación de la lesión es crucial, ya que minimiza los falsos positivos y permite cuantificar con exactitud el porcentaje de área foliar dañada. Esta capacidad técnica se traduce directamente en una herramienta de apoyo para el manejo fitosanitario, posibilitando a los especialistas determinar la severidad de la infección en etapas tempranas y, en consecuencia, optimizar la dosificación de agroquímicos antes de que la cosecha se vea comprometida.

TABLA III
MÉTRICAS DE EVALUACIÓN POR CADA BACKBONE EN SEGMENTACIÓN DE LA HOJA

Backbone	IoU	Precision	Recall	F1
ResNet-50	0.94	0.98	0.92	0.92
EfficientNetB0	0.93	0.98	0.91	0.93
MobileNetV2	0.92	0.95	0.91	0.93

TABLA IV
MÉTRICAS DE EVALUACIÓN POR CADA BACKBONE EN SEGMENTACIÓN DE LA LESIÓN

Backbone	IoU	Precision	Recall	F1
ResNet-50	0.89	0.96	0.80	0.82
EfficientNetB0	0.88	0.95	0.79	0.85
MobileNetV2	0.88	0.95	0.78	0.84

En la Fig. 8 se presenta ejemplos de las segmentaciones generadas por cada modelo evaluado, teniendo en cuenta que es la misma imagen de entrada, se muestran dos máscaras de salida, la hoja completa y la lesión. Se logra ver que los 3 modelos delimitan adecuadamente con algunas variaciones

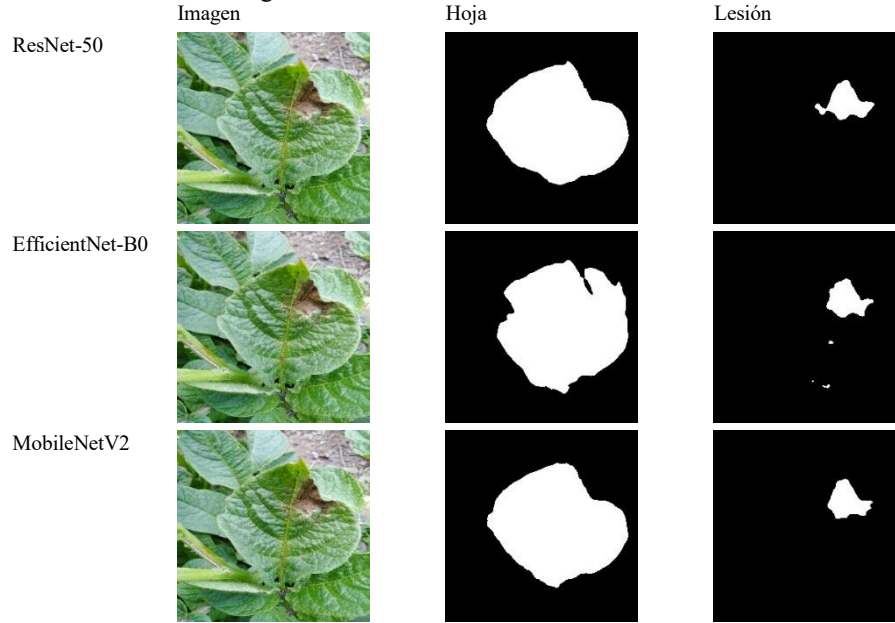


Fig. 8 Ejemplos de segmentación real con cada *backbone*

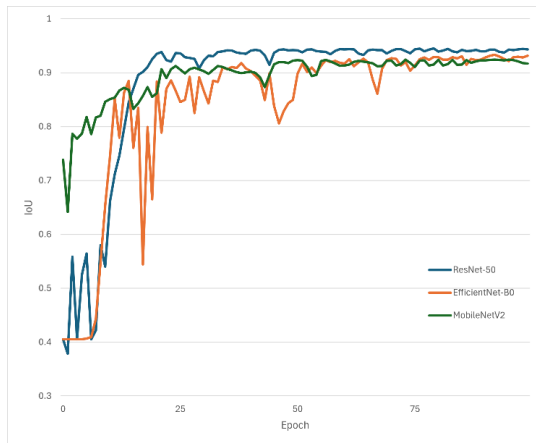


Fig. 9 Comparación de las pérdidas generales de cada *backbone*.

pequeñas, separándola efectivamente de la hoja. Sin embargo, se observa que existe una mayor diferenciación cuando se observan las regiones afectadas. *ResNet-50* muestra una máscara de lesión más detallada y mejor parametrizada.

En la Fig. 9 se muestra la evolución de la pérdida durante el entrenamiento. Al inicio, *MobileNetV2* presenta los valores de pérdida más bajos, mientras que *ResNet-50* y *EfficientNetB0* exhiben pérdidas significativamente mayores. Alrededor de las épocas 22–23, los tres modelos convergen por primera vez. Se observa además que *EfficientNetB0* presenta la mayor variabilidad a lo largo del proceso. A partir de la época 75, los tres modelos alcanzan valores de pérdida prácticamente equivalentes, convergiendo en torno a 0.2.

V. CONCLUSIONES Y TRABAJOS FUTUROS

En este estudio se analizó 3 modelos avanzados (*ResNet-50*, *EfficientNet-B0* y *MobileNetV2*) utilizados como codificadores para la segmentación semántica multitarea de la hoja y lesiones de *Phytophthora infestans* empleando la arquitectura de segmentación semántica *U-Net*. El enfoque se centró en cuatro etapas: (1) la adquisición del conjunto de datos provenientes del CIP, (2) el preprocesamiento del conjunto de datos, (3) el entrenamiento de los modelos y (4) la evaluación de su desempeño.

Para evaluar la eficacia de los distintos codificadores del modelo de segmentación semántica multitarea, se analizaron imágenes de hojas de papa afectadas por *Phytophthora infestans*. Se emplearon métricas como *IoU*, *Precision*, *Recall* y *F1*, lo que permitió comparar el desempeño de cada codificador. Debido a este análisis, fue posible determinar cuál

de ellos logra una segmentación más precisa y consistente de las áreas objetivo en las imágenes.

Los resultados obtenidos demostraron que los codificadores evaluados: *ResNet-50*, *EfficientNet-B0* y *MobileNetV2*, son alternativas viables para la segmentación semántica multitarea de hojas de papa afectadas por *Phytophthora Infestans*, donde *ResNet-50* mostró un desempeño superior en las métricas clave, posicionándose como la opción más precisa y robusta para la identificación y segmentación de las áreas dañadas.

Como trabajos futuros, se recomienda evolucionar esta propuesta hacia un prototipo funcional que permita a los agricultores peruanos obtener un diagnóstico inmediato mediante un aplicativo móvil. Para lograr esto, será indispensable explorar arquitecturas adicionales más recientes y ligeras que optimicen el consumo de recursos en dispositivos móviles. Asimismo, se proyecta ampliar el dataset actual incluyendo un mayor volumen de imágenes capturadas en diversas regiones y condiciones climáticas para mejorar la robustez de la red, culminando con una fase de validación en campo que evalúe el desempeño práctico del modelo en escenarios reales de la mano de los propios agricultores.

AGRADECIMIENTOS

Queremos expresar nuestro más sincero agradecimiento al especialista del CIP por facilitar el acceso al *dataset* utilizado en este estudio. También agradecemos el apoyo a la Dirección de Investigación de la Universidad Peruana de Ciencias Aplicadas, por el apoyo del presente proyecto de investigación a través del incentivo UPC-Expost-2026-1.

REFERENCIAS

- [1] Devaux, A., Kromann, P., & Ortiz, O. (2014). Potatoes for sustainable global food security. *Potato Research*, 57(3), 185–199.
- [2] Fry, W. E. (2008). *Phytophthora infestans*: the plant (and R gene) destroyer. *Molecular Plant Pathology*, 9(3), 385–402.
- [3] Haverkort, A. J., Boonekamp, P. M., Hutten, R., Jacobsen, E., Lotz, L. A. P., Kessel, G. J. T., ... & Visser, R. G. F. (2008). Societal costs of late blight in potato and prospects of durable resistance through cisgenic modification. *Potato Research*, 51, 47–57.
- [4] Gold, C. S., et al. (2019). Limitations of traditional disease detection in agriculture. *Journal of Crop Protection*, 123, 45–53.
- [5] Ortiz, O., et al. (2022). Climate variability and late blight management in Andean potato farming. *Agricultural Systems*, 195, 103–112.
- [6] Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 147, 70–90.
- [7] Tan, M., & Le, Q. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. *Proceedings of ICML*.
- [8] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 28.
- [9] Gao, W., Xiao, Z., & Bao, T. (2023). Detection and identification of Potato-Typical diseases based on multidimensional fusion Atrous-CNN and hyperspectral data. *Applied Sciences*, 13(8), 5023.
- [10] Oishi, Y., Habaragamuwa, H., Zhang, Y., Sugiura, R., Asano, K., Akai, K., Shibata, H., & Fujimoto, T. (2021). Automated abnormal potato plant detection system using deep learning models and portable video cameras. *International Journal of Applied Earth Observation and Geoinformation*, 104, 102509.
- [11] Shaheed, K., Qureshi, I., Abbas, F., Jabbar, S., Abbas, Q., Ahmad, H., & Sajid, M. Z. (2023). EfficientRMT-NeT—An efficient RESNET-50 and Vision Transformers approach for classifying potato plant leaf diseases. *Sensors*, 23(23), 9516.
- [12] Lee, T., Lin, I., Yu, J., Yang, J., & Chang, Y. (2021). High Efficiency Disease Detection for Potato Leaf with Convolutional Neural Network. *SN Computer Science*, 2(4).
- [13] Lizarazo, I., Rodriguez, J. L., Cristancho, O., Olaya, F., Duarte, M., & Prieto, F. (2022). Identification of symptoms related to potato Verticillium wilt from UAV-based multispectral imagery using an ensemble of gradient boosting machines. *Smart Agricultural Technology*, 3, 100138.
- [14] Rashid, J., Khan, I., Ali, G., Almotiri, S. H., AlGhamdi, M. A., & Masood, K. (2021). Multi-Level Deep learning model for potato leaf disease recognition. *Electronics*, 10(17), 2064.
- [15] Li, X., Zhou, Y., Liu, J., Wang, L., Zhang, J., & Fan, X. (2022). The detection method of potato foliage diseases in complex background based on instance segmentation and semantic segmentation. *Frontiers in Plant Science*, 13.
- [16] Arshad, F., Mateen, M., Hayat, S., Wardah, M., Al-Huda, Z., Gu, Y. H., & Al-Antari, M. A. (2023). PLDPNet: End-to-end hybrid deep learning framework for potato leaf disease prediction. *Alexandria Engineering Journal*, 78, 406–418.
- [17] assis, L. M., De Souza, J. E. T., & Krohling, R. A. (2021). A deep learning approach combining instance and semantic segmentation to identify diseases and pests of coffee leaves from in-field images. *Computers and Electronics in Agriculture*, 186, 106191.
- [18] Rashid, R., Aslam, W., Aziz, R., & Aldehim, G. (2024). An early and smart detection of corn plant leaf diseases using IoT and Deep Learning Multi-Models. *IEEE Access*, 12, 23149–23162.
- [19] Rezk, N. G., Hemdan, E. E., Attia, A., El-Sayed, A., & El-Rashidy, M. A. (2023). An efficient IoT based framework for detecting rice disease in smart farming system. *Multimedia Tools and Applications*, 82(29), 45259–45292.
- [20] Alshammari, K., Alshammari, R., Alshammari, A., & Alkhudaydi, T. (2024). An improved pear disease classification approach using cycle generative adversarial network. *Scientific Reports*, 14(1).
- [21] Ahishakiye, E., Mwangi, W., Muriithi, P., Kanobe, F., Owomugisha, G., Taremwia, D., & Nkalubo, L. (2023). Deep Gaussian convolutional neural network model in classification of cassava diseases using spectral data. *The Journal of Supercomputing*, 80(1), 463–485.
- [22] Vishnoi, V. K., Kumar, K., Kumar, B., Mohan, S., & Khan, A. A. (2022). Detection of apple plant diseases using leaf images through convolutional neural network. *IEEE Access*, 11, 6594–6609.
- [23] Yang, S., Xing, Z., Wang, H., Dong, X., Gao, X., Liu, Z., Zhang, X., Li, S., & Zhao, Y. (2023). Maize-YOLO: a new High-Precision and Real-Time method for maize pest detection. *Insects*, 14(3), 278.
- [24] Gamboa, Soledad, Kreuze, Jan., Andrade-Piedra, Jorge. (2019). “Dataset for: Photos of leaves showing symptoms caused by *Phytophthora infestans*, fungi and virus on potato crop,” *International Potato Center*, V1. <https://doi.org/10.21223/BCVIZY>
- [25] Tan, M., & Le, Q. (2019). *EfficientNet: Rethinking model scaling for convolutional neural networks*. PMLR. <https://proceedings.mlr.press/v97/tan19a.html>
- [26] He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep Residual Learning for Image Recognition*. CVPR 2016.
- [27] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). *MobileNetV2: Inverted Residuals and Linear Bottlenecks*. CVPR 2018.