UNIVERSIDAD NACIONAL MICAELA BASTIDAS DE APURÍMAC

FACULTAD DE INGENIERÍA

ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA INFORMÁTICA Y SISTEMAS

Tesis en formato de artículo científico

Detection of Malaria Infections Using Convolutional Neural Networks

Presentado por:

Luis Edison Ñahui Vargas

Para optar el título de Ingeniero Informático y Sistemas

Abancay, Perú

2025

UNIVERSIDAD NACIONAL MICAELA BASTIDAS DE APURÍMAC

FACULTAD DE INGENIERÍA

ESCUELA ACADÉMICO PROFESIONAL DE INGENIERÍA INFORMÁTICA Y SISTEMAS

MODALIDAD DE TESIS EN FORMATO DE ARTÍCULO CIENTÍFICO

Detection of Malaria Infections Using Convolutional Neural Networks

Presentado por **Luis Edison Ñahui Vargas**, para optar el título de Ingeniero Informático y Sistemas

Sustentado y aprobado el 31 de julio de 2025, ante el jurado evaluador:

Presidente: *Dr. Manuel Jesús Ibarra Cabrera*

Primer miembro: *Dra. Hesmeralda Rojas Enríquez*

Segundo miembro: *Mtro. Rafael Ricardo Quispe Merma*

Asesores: *Mag. Mario Aquino Cruz*

"Año de la recuperación y consolidación de la economía peruana"

CONSTANCIA DE ORIGINALIDAD
Nº 161-2025

La Universidad Nacional Micaela Bastidas de Apurímac, a través de la Unidad de Investigación de la Facultad de Ingeniería declara que, la Tesis en modalidad de Artículo Científico titulada en español: **"Detección de Infecciones de Malaria mediante Redes Neuronales Convolucionales"**, e inglés: **"Detection of malaria infections using convolutional neural networks"**, presentado por el Bach: **Luis Edison Ñahui Vargas,** Para optar el Título de **Ingeniero Informático y Sistemas** ; ha sido sometido a un mecanismo de evaluación y verificación de similitud, a través del Software Turnitin, siendo el índice de similitud ACEPTABLE de **(05%)** por lo que, cumple con los criterios de originalidad establecidos por la Universidad.

Abancay, 18 de julio del 2025

Atentamente,

UNIVERSIDAD NACIONAL MICAELA BASTIDAS
FACULTAD DE INGENIERÍA
Dra. Hesmeralda Rojas Enriquez
DIRECTORA DE LA UNIDAD DE INVESTIGACIÓN
FACULTAD DE INGENIERÍA

C. c.
Archivo
REG. N° 530

Agradecimiento

Agradezco profundamente a Dios por regalarme el don de la vida y la fuerza necesaria para alcanzar cada meta. A mi papá Valentín, a mi mamá Yolanda y mi hermano Joel, por ser mi motor constante, por su amor incondicional y su apoyo firme en cada paso de este camino. A mi familia en general, a mis amigos y a todas aquellas personas que, incluso sin conocernos profundamente, compartieron conmigo sus experiencias y consejos, motivándome a seguir creciendo en el ámbito profesional.

Extiendo también mi sincero agradecimiento a cada uno de mis docentes, por su dedicación y por ser guía en mi formación académica. Y, por supuesto, a la Universidad Nacional Micaela Bastidas de Apurímac, por acogerme durante esta etapa crucial de mi vida, brindándome el espacio para aprender, desarrollarme y construir las bases de mi futuro profesional.

***Gracias a todos por formar parte de este logro**.*

Dedicatoria

Dedico este trabajo a Dios, por sostenerme en los momentos de incertidumbre y darme la vida, la fe y la fortaleza para seguir adelante cuando todo parecía cuesta arriba.

A mis padres, pilares inquebrantables de mi existencia, por su amor incondicional, sus sacrificios silenciosos y su ejemplo constante de integridad y esfuerzo. A mi hermano, compañero de vida, por su apoyo sincero y su presencia que siempre me inspira.

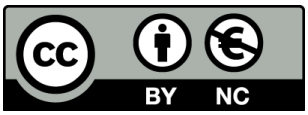
A mi familia en general, que creyó en mí, y a mis amigos verdaderos, que supieron acompañarme con afecto, comprensión y aliento en cada etapa de esta travesía.

A cada docente que sembró en mí no solo conocimiento, sino también valores. Y a aquellas personas que, sin saberlo, marcaron mi vida con un gesto, una palabra o una enseñanza.

Finalmente, dedico este logro a mí mismo: al joven que no se rindió, que cayó, pero se levantó; que dudó, pero siguió. A ese yo que creyó que, con esfuerzo, todo puede transformarse en posibilidad.

Detection of Malaria Infections Using Convolutional Neural Networks

Línea de investigación: Ingeniería informática, industria y sociedad



ÍNDICE

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

(IJACSA) International Journal of Advanced Computer Science and Applications,
Vol. 16, No. 5, 2025

Detection of Malaria Infections Using Convolutional Neural Networks

Luis Edison Ñahui Vargas, Mario Aquino Cruz
Departamento Académico De Informática y Sistemas, Universidad Nacional Micaela Bastidas De Apurímac, Abancay, Perú

Abstract*—Malaria persists as a serious global public health threat, particularly in resource-limited regions where timely and accurate diagnosis is a challenge due to poor medical infrastructure. This study presents a comparative evaluation of three pre-trained convolutional neural network (CNN) architectures—EfficientNetB0, InceptionV3, and ResNet50—for automated detection of Plasmodium-infected blood cells using the Malaria Cell Images Dataset. The models were implemented in Python with TensorFlow and trained in Google Colab Pro with GPU A100 acceleration. Among the models evaluated, ResNet50 proved to be the most balanced, achieving 97% accuracy, a low false positive rate (1.8%) and the shortest training time (2.9 hours), making it a suitable choice for implementation in real-time clinical settings. InceptionV3 obtained the highest sensitivity (98% recall), although with a higher false positive rate (4.0%) and a higher computational demand (6.5 hours). EfficientNetB0 was the fastest model (3.2 hours), showed validation and a higher false negative rate (6.2%). Standard metrics—accuracy, loss, recall, F1-score and confusion matrix—were applied under a non-experimental cross-sectional design, along with regularization and data augmentation techniques to improve generalization and mitigate overfitting. As a main contribution, this research provides reproducible empirical evidence to guide the selection of CNN architectures for malaria diagnosis, especially in resource-limited settings. This systematic comparison between state-of-the-art models, under a single protocol and homogeneous metrics, represents a significant novelty in the literature, guiding the selection of the most appropriate architecture. In addition, a lightweight graphical user interface (GUI) was developed that allows real-time visual testing, reinforcing its application in clinical and educational settings. The findings also suggest that these models, in particular ResNet50, could be adapted for the diagnosis of other parasitic diseases with similar cell morphology, such as leishmaniasis or babesiosis.

Keywords*—Malaria diagnosis; CNN architectures; deep learning; artificial intelligence; plasmodium; clinical decision support; medical imaging

I. INTRODUCTION

Malaria is one of the most prevalent and deadly infectious diseases in tropical and subtropical regions, affecting mainly developing countries. According to the World Health Organization (WHO), approximately 247 million cases of malaria were reported worldwide in 2021, with more than 600,000 deaths, with children under five years of age being the most vulnerable group [1]. By 2023, global cases increased to 263 million, with 597,000 deaths, attributed to factors such as climate change, drug resistance and persistent inequalities in health systems. Sub-Saharan Africa continues to account for 94% of cases, highlighting the magnitude of the problem [2]. In Peru, malaria has been a historical concern, particularly affecting the Amazon regions. Despite the implementation of strategies such as the Plan towards the Elimination of Malaria in Peru 2022-2030, the disease remains a critical public health problem. In 2023, 17,840 cases of malaria were reported, while up to epidemiological week 40 of 2024, nine deaths were reported [3], [4]. Loreto, the most affected region, reported more than 8,000 cases in the same period [5]. Factors such as poverty, inaccessibility of many communities and climatic conditions conducive to the Anopheles mosquito transmitter aggravate the situation, highlighting the need for innovative diagnostic solutions that are accurate, accessible and rapidly implemented. Although significant progress has been made in its prevention and treatment, timely and accurate diagnosis remains a critical challenge, especially in resource-limited areas where access to equipment and trained personnel is insufficient [1]. Traditional malaria diagnosis, based on microscopic analysis of stained blood smears, has several limitations, including the need for technical expertise and the time required to process multiple samples [6]. These limitations not only delay the initiation of appropriate treatment, but can also result in misdiagnosis, exacerbating the burden of disease in already affected communities [7].

Against this backdrop, artificial intelligence (AI) technologies have emerged as promising tools to address the challenges associated with malaria diagnosis. Convolutional neural networks used in image processing and analysis are particularly notable for their ability to identify complex patterns in medical images. These technologies have proven to be effective in the diagnosis of various diseases, including malaria, through automated analysis of blood smears [8]. Some recent studies have also explored the use of advanced architectures such as EfficientNet, ResNet and VGG16 in clinical, mobile and rural settings, demonstrating their practical applicability for automated diagnosis in areas with scarce computational resources. However, the current literature presents a significant gap in terms of systematic comparisons between state-of-the-art CNN models applied to malaria. Most studies focus on individual architectures or use complex methods such as ensembles, which require high computational capacity. Therefore, this research posed the following research question: which of the convolutional neural network models ResNet50, EfficientNetB0 or InceptionV3 demonstrates better performance in the automated detection of malaria infections?

The objective of this research was to comparatively evaluate these architectures, using standard metrics such as accuracy, loss, recall, and F1-Score, in order to identify the most efficient and feasible model for eventual integration into automated

clinical diagnostic systems. This study uses the Malaria Cell Images Dataset, composed of 27,558 images categorized as "infected" and "uninfected" [9]. The development and training of the models was carried out using Google Colab Pro, using Python programming tools and libraries such as TensorFlow and Keras.

The main contribution of this research lies in providing reproducible and comparative empirical evidence on the performance of three widely recognized architectures in the field of deep learning, in a controlled but replicable environment. The results will guide technical decisions for the implementation of practical AI-assisted diagnostic solutions, especially in endemic regions with limited medical infrastructure. This research not only explores the technical capabilities of CNNs, but also highlights their potential to be adapted to the detection of other parasitic diseases of similar morphology, such as leishmaniasis or babesiosis, thus extending their relevance beyond the specific context of malaria.

II. Related Work

In recent years, several research works have evidenced the potential of convolutional neural networks (CNNs) for automated malaria diagnosis. For example, Zhao et al. [10] proposed in 2020 a solution for low-cost mobile devices, achieving 96.5% accuracy with VGG16. Vizcaino Gispert [11], also in 2020, used Faster R-CNN, improving their results with pre-trained weights. Subsequently, in 2021, Sierra Segovia et al. [12] applied CNN for both malaria and COVID-19, obtaining an accuracy of 99.33% in the latter. That same year, Ferreras Extremo [13] used an EfficientNet assembly, achieving an accuracy of 98.29%. More recently, Marín Calvo [14] in 2022 reported 95.58% with data augmentation. Finally, Meza-Bautista et al. [15] in 2024 performed a comparison of CNN architectures and concluded that EfficientNetB0 offered the best performance (97.12%). In this context of advances, our research focused on a systematic and controlled comparison of ResNet50, EfficientNetB0 and InceptionV3. This study provides quantitative evidence on which offered the best balance between diagnostic accuracy, computational efficiency and clinical applicability, filling a gap in the literature regarding direct and comprehensive comparisons of these architectures in the field of malaria diagnosis.

III. Methodology

A. Research Design

The research design is non-experimental cross-sectional, which implies that the performance of different convolutional neural network (CNN) models in detecting malaria infections at a specific time was evaluated, without manipulating the study variables [16]. This approach allowed an objective analysis of performance metrics, such as accuracy, recall and F1-score, using a previously established data set [17].

B. Participants

The population of this study consisted of a set of 27,560 labeled blood smear images obtained from public databases such as Kaggle [18], which provides a set of images used in scientific research to train and evaluate classification models.

The sample was divided into three subsets: 60% for training, 30% for validation and 10% for testing, ensuring that each subset has a similar distribution of images, which minimizes bias in model evaluation [19]. Fig. 1 and Fig. 2 show representative images of blood smears used in the study, while Fig. 3, 4 and 5 illustrate the architectures of the convolutional neural networks used: ResNet50, EfficientNetB0 and InceptionV3, respectively. Fig. 1 presents a malaria-infected blood smear image from the dataset [18], the presence of parasites is evident, with multiple organisms within the red blood cells, indicating an active and severe infection.

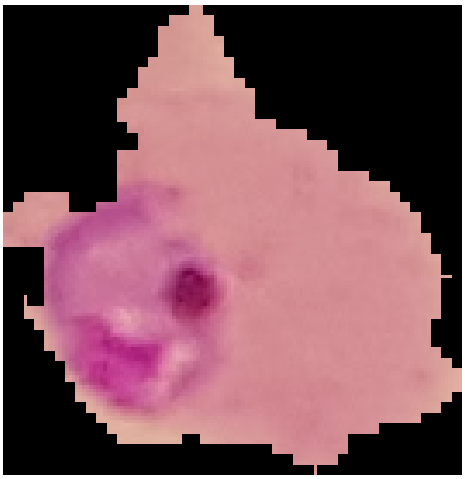


Fig. 1. Malaria-infected blood smear.

Fig. 2 shows a blood smear that is not infected by malaria, image from the dataset [18], the staining is uniform and clear, with no signs of parasites present in the red blood cells.

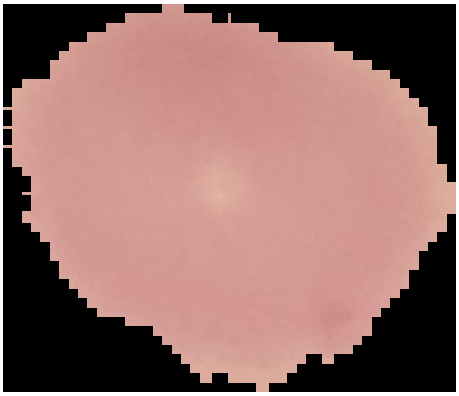


Fig. 2. Blood smear that is not infected with malaria.

Design of the architectures: The following figures illustrate the pretrained convolutional neural network architectures used.

The ResNet50 model (see Fig. 3) is a 50-layer deep convolutional neural network, distinguished by its innovative residual or "skip connections." These connections are crucial for enabling information and gradients to flow directly across multiple layers, effectively mitigating the vanishing gradient problem in very deep architectures and significantly aiding training.The architecture is structured into five primary stages, each built with residual blocks. It begins with an initial 7x7

convolutional layer followed by max-pooling. The subsequent four stages employ "bottleneck blocks," each comprising a sequence of 1x1, 3x3, and 1x1 convolutional layers. The 1x1 layers manage channel dimensionality, while the 3x3 focuses on feature extraction. The residual connection in each block adds the original input to the output of these layers, allowing the model to learn identity functions. Specifically, the architecture includes 3 residual blocks in the first stage (conv2_x), 4 in the second (conv3_x), 6 in the third (conv4_x), and 3 in the fourth (conv5_x). The final stages involve a global average pooling layer and a dense layer with SoftMax activation for binary classification (e.g., 'infected' or 'uninfected'). This robust hierarchical structure renders ResNet50 exceptionally effective for complex image classification tasks [20], [21], [22].

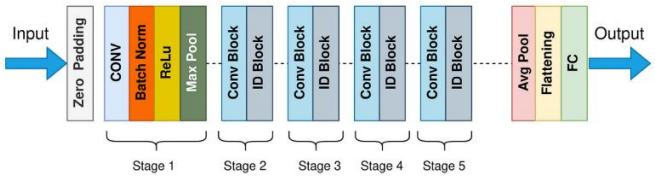


Fig. 3. Architecture of the ResNet50 model based on transfer of learning, taken from [23]. This diagram illustrates the residual connections that facilitate gradient flow through deep layers, organized into five main stages with "bottleneck" blocks. While the dimensions and number of filters for the layers are depicted in the diagram.

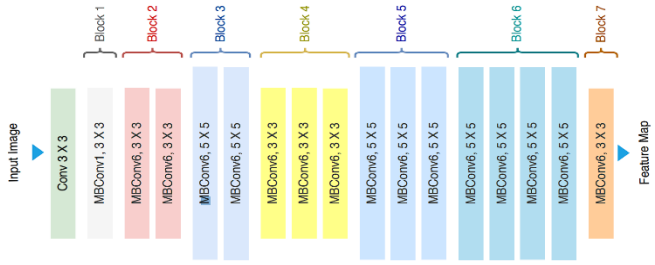


Fig. 4. The Architecture of the EfficientNetB0 model, extracted from [25]. This diagram illustrates the network's structure based on compound scaling and its primary building block, the Mobile Inverted Bottleneck Convolution (MBConv) block, which incorporates Squeeze-and-Excitation networks. While the detailed parameters and operations within each block are depicted.

Following ResNet50, EfficientNetB0 (see Fig. 4) represents another state-of-the-art CNN architecture, notable for its compound scaling method. This innovative approach uniformly scales network depth, width, and resolution using fixed coefficients, allowing superior performance with fewer parameters and lower computational cost. The core building block is the Mobile Inverted Bottleneck Convolution (MBConv) block, adapted from MobileNetV2. These MBConv blocks integrate depthwise separable convolutions, significantly reducing computational expense, along with an inverted bottleneck structure. Crucially, each MBConv block also includes a Squeeze-and-Excitation (SE) network, which adaptively recalibrates channel-wise feature responses by learning channel-wise attention, further enhancing the model's representational power. EfficientNetB0 is organized into multiple stages, each consisting of several MBConv blocks. The network begins with an initial convolutional layer, followed by compound-scaled MBConv stages for feature extraction. A global average pooling layer and a final dense layer with softmax activation are then used for classification. Despite its relatively compact size, EfficientNetB0 is designed for high efficiency and accuracy, making it a compelling choice for deployment in resource-constrained environments where computational budget is a significant concern [24].

Finally, Finally, InceptionV3 (see Fig. 5) is another powerful pre-trained CNN architecture explored in this study, designed for high computational efficiency and accuracy by optimizing resource use. Its core innovation lies in the Inception modules, which enable the network to perform multiple parallel convolutions with different kernel sizes (1x1, 3x3, 5x5) and max-pooling operations on the same input. This parallel processing allows the model to capture features at various scales simultaneously, providing richer data representation. A key aspect of InceptionV3's design is the strategic factorization of larger convolutions into smaller ones (e.g., replacing a 5x5 convolution with two 3x3s, or a 3x3 into 1x3 and 3x1). This significantly reduces parameters and computational cost while preserving or improving representational capacity. Additionally, InceptionV3 incorporates batch normalization in auxiliary classifiers and uses label smoothing during training for regularization and overfitting prevention. The InceptionV3 architecture consists of multiple stacked Inception modules, with interleaved pooling layers to reduce spatial dimensions. It begins with initial convolutional layers and max-pooling, followed by Inception modules that extract abstract features. Similar to other models, the final part employs a global average pooling layer and a dense layer with softmax activation for binary classification. This design makes InceptionV3 robust and efficient, particularly where capturing multi-scale features is critical [26].

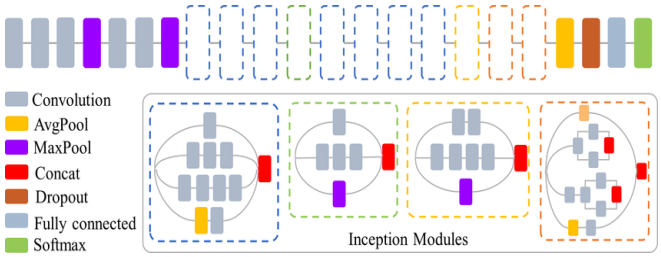


Fig. 5. The Architecture of the InceptionV3 model, extracted from [27]. This diagram illustrates the network's structure, highlighting the use of Inception modules that perform parallel convolutions with different kernel sizes to capture multi-scale features. While the specific details of the factorization and layer parameters are shown.

C. Instruments and Techniques

For the implementation of this study, Python was used as the programming language, taking advantage of its versatility and the vast support of specialized machine learning libraries [28]. Model training was conducted on Google Colab Pro, a development environment that provided access to cloud computing resources, including an A100 GPU, which was essential to handle the computational complexity [29]. The TensorFlow and Keras libraries were used, which are fundamental for the construction and training of neural networks, facilitating the implementation of deep learning algorithms [30]. In terms of techniques, extensive image preprocessing including pixel value normalization and data augmentation strategies such as rotations, image inversion and

pixel filling were applied to improve the generalizability of the model [19], [31]. For model training, the pre-trained architectures ResNet50, EfficientNetB0 and InceptionV3 were selected and optimized in the Google Colab Pro environment with GPU A100 [20], [26]. Finally, the model was evaluated using standard metrics such as accuracy, recall and F1-score, complemented with a confusion matrix for a detailed analysis of the classification errors [32], [33].

D. Data Analysis

Data analysis was performed by comparing the performance metrics of the trained models. Accuracy, loss, recall and F1-score metrics were calculated for each model using the test set. The confusion matrix was used to identify specific error patterns, allowing a more detailed assessment of the performance of each model [34]. The results were statistically analyzed to determine the effectiveness of each neural network in detecting malaria infections, ensuring rigorous and reproducible analysis of the results [8], [35].

IV. Results

A. Results of the EfficientNetB0 Model

Table I presents the performance metrics obtained during the training, validation and testing phases of the EfficientNetB0 model. The training was initially set up for 100 epochs, but was automatically stopped at epoch 15 due to the EarlyStopping callback, which monitored val_loss with a patience of 5 epochs. The total training time was 3 hours, 14 minutes and 48 seconds, using an A100 GPU.

TABLE I. Performance of EfficientNetB0 in Training, Validation and Testing

Metric	Train	Validation	Test
Accuracy	0.99	0.95	0.95
Loss	0.041	0.187	0.200
Predictive value (Precision)	0.99 (P), 0.99 (U)	0.96 (P), 0.96 (U)	0.97 (P), 0.94 (U)
Recall	0.99 (P), 0.99 (U)	0.95 (P), 0.96 (U)	0.94 (P), 0.97 (U)
F1-Score	0.99	0.95	0.95

- Legend:
- P: Parasitized (infected cells)
- U: Uninfected (uninfected cells)

Fig. 6 shows the evolution of the accuracy during EfficientNetB0 training:

- Train Accuracy: Increased rapidly from 89.17% (epoch 1) to 99.15% (epoch 15), indicating that the model learned the training data efficiently.
- Validation Accuracy: It showed high variability, ranging between 50% and 95.52%, with the best performance in epoch 10 (95.52%)

The gap between “train” and “validation” suggests possible overfitting.

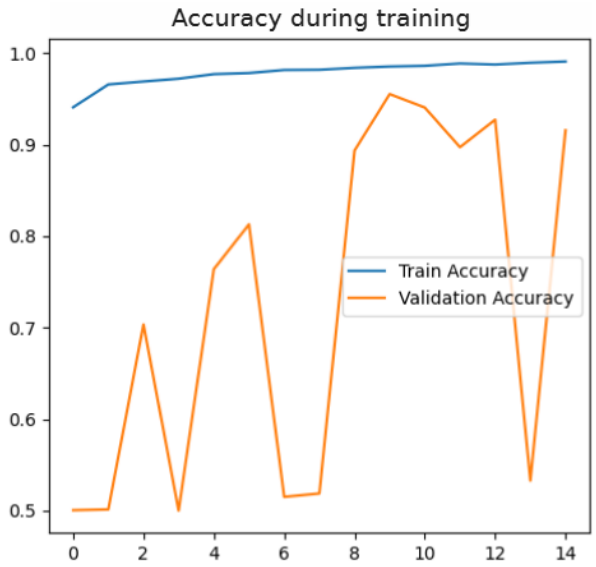


Fig. 6. Accuracy curve during EfficientNetB0 training.

Fig. 7 shows the decrease in loss:

- Train Loss: It decreased steadily from 8.8683 (epoch 1) to 0.0412 (epoch 15).
- Validation Loss: It showed significant fluctuations, with peaks at epochs 4, 7 and 14. The minimum was reached at epoch 10 (0.1865), coinciding with the maximum val_accuracy.

Fig. 7. Loss curve during EfficientNetB0 training.

The confusion matrix Fig. 8 highlights a 6.2% false negative rate (84 parasitized cells not detected), a potential risk in medical applications.

- True Positives (TP): 1294 (93.8% of correctly identified Parasitized).

- False Negatives (FN): 84 (6.2% of Parasitized misclassified as Uninfected).
- False Positives (FP): 43 (3.1% of Uninfected misclassified as Parasitized).
- True Negatives (TN): 1335 (96.9% of correctly identified Uninfected).

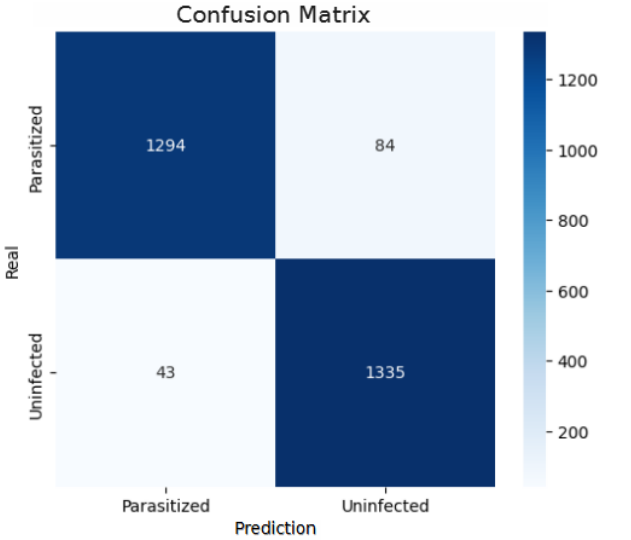


Fig. 8. Confusion matrix of the EfficientNetB0 model in the test set.

B. *Results of the InceptionV3 Model*

Table II presents the performance metrics obtained during the training, validation and testing phases of the InceptionV3 model, which was trained for 16 epochs (out of 100 scheduled) using EarlyStopping with patience for 5 epochs (monitoring val_loss). The total training time was 6 hours, 29 minutes and 44 seconds using an A100 GPU.

TABLE II. PERFORMANCE OF INCEPTIONV3 IN TRAINING, VALIDATION AND TESTING

Metric	Train	Validation	Test
Accuracy	0.99	0.97	0.97
Loss	0.029	0.120	0.125
Predictive value (Precision)	0.99 (P), 0.99 (U)	0.97 (P), 0.97 (U)	0.96 (P), 0.97 (U)
Recall	0.99 (P), 0.99 (U)	0.97 (P), 0.97 (U)	0.98 (P), 0.96 (U)
F1-Score	0.99	0.97	0.97

- Legend:
- P: Parasitized (infected cells)
- U: Uninfected (uninfected cells)

Fig. 9 shows the evolution of Accuracy during InceptionV3 training:

- Train Accuracy: Reached 99.5% at epoch 14, indicating near perfect learning.
- Validation Accuracy: Stabilized at 97% after epoch 3, with minimal fluctuations in epochs 5 and 14.

Reduced gap: 2.27% (vs. 3.63% in EfficientNetB0), suggesting less over-adjustment.

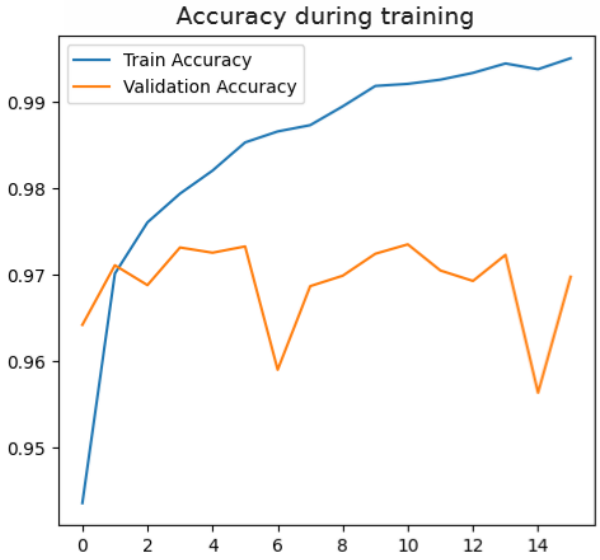


Fig. 9. Accuracy curve during InceptionV3 training.

Fig.10 displays the decrease in loss of the InceptionV3 model:

- Train Loss: Decreased from 10.6204 to 0.0294.
- Validation Loss: It showed a minimum peak at epoch 6 (0.1063).

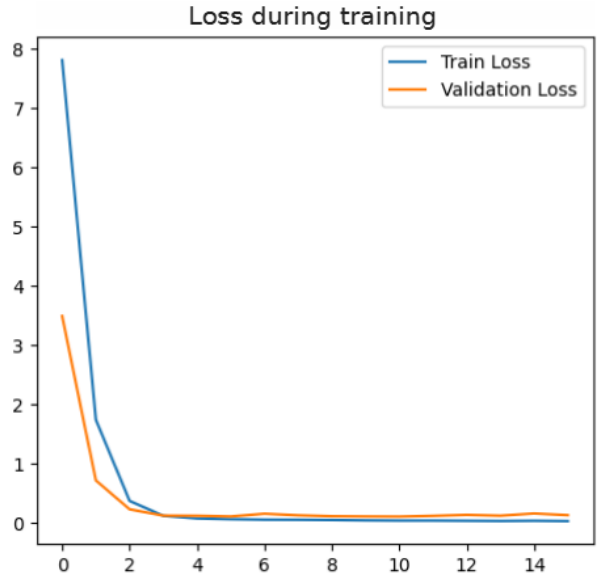


Fig. 10. Loss curve during InceptionV3 training.

Fig. 11 shows the confusion matrix of InceptionV3:

- False negatives (FN): 34 (2.5% of Parasitized misclassified as Uninfected).
- False positives (FP): 55 (4.0% of Uninfected misclassified as Parasitized).

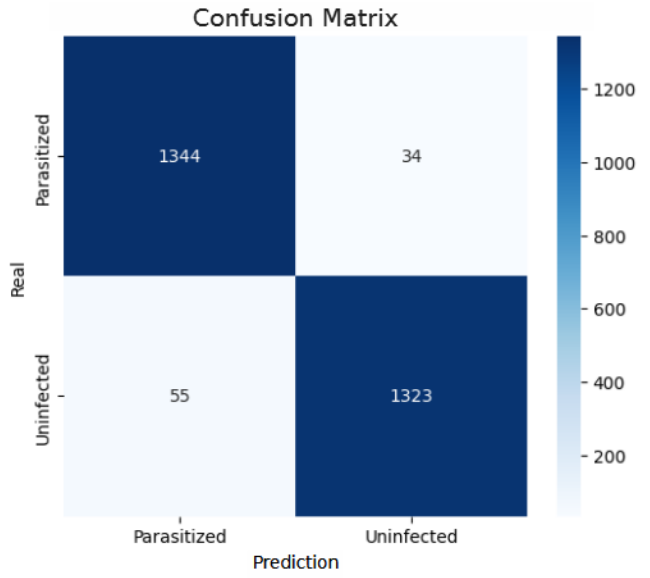


Fig. 11. Confusion matrix of InceptionV3 in the test set.

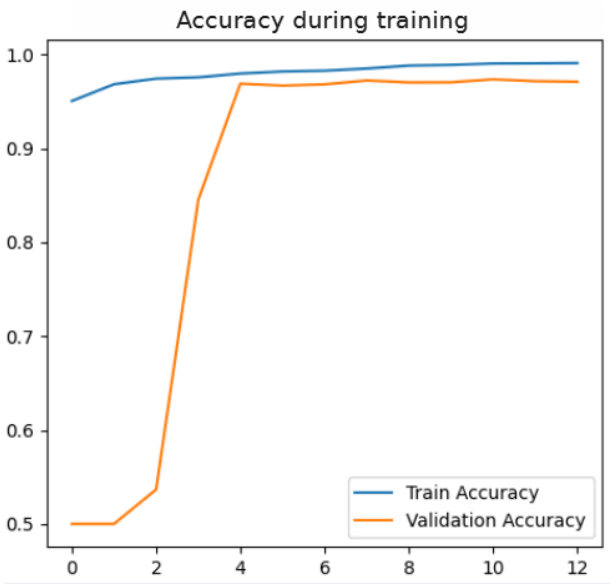


Fig. 12. Accuracy curve during ResNet50 training.

C. *Results of the Resnet50 Model*

Table III presents the performance metrics obtained during the training, validation and testing phases of the ResNet50 model, it was trained for 13 epochs (out of 100 scheduled), using EarlyStopping with 5 epoch patience and validation loss monitoring (val_loss). The total training time was 2 hours, 58 minutes and 35 seconds using an A100 GPU.

TABLE III. PERFORMANCE OF RESNET50 IN TRAINING, VALIDATION AND TESTING

Metric	Train	Validation	Test
Accuracy	0.99	0.97	0.97
Loss	0.037	0.109	0.112
Predictive value (Precision)	0.99 (P), 0.99 (U)	0.97 (P), 0.97 (U)	0.98 (P), 0.97 (U)
Recall	0.99 (P), 0.99 (U)	0.97 (P), 0.97 (U)	0.96 (P), 0.98 (U)
F1-Score	0.99	0.97	0.97

- Legend:
- P: Parasitized (infected cells)
- U: Uninfected (uninfected cells)

Fig. 12 shows the evolution of Accuracy during ResNet50 training:

- Train Accuracy: Reached 99.13% at epoch 13.
- Validation Accuracy: Stabilized at 97% after epoch 5, with minimal fluctuations.

Reduced gap: 1.78% (vs. 2.27% in InceptionV3), suggesting less overfitting.

Fig. 13 shows the loss decrease of the ResNet50 model:

- Train Loss: Decreased from 10.7185 to 0.0372.
- Validation Loss: Showed a minimum in epoch 11 (0.1089).

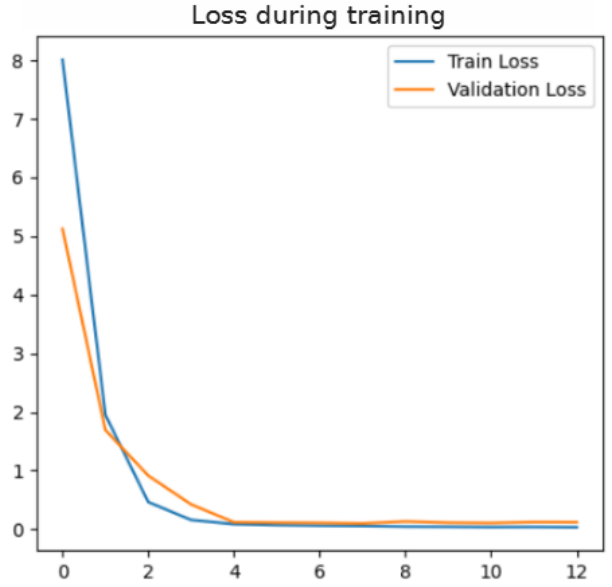


Fig. 13. Loss curve during ResNet50 training.

Fig. 14 shows the ResNet50 confusion matrix:

- False negatives (FN): 49 (3.6%) vs. 34 in InceptionV3. Slight increase, but still better than EfficientNetB0 (84).
- False positives (FP): 25 (1.8%). Better specificity than InceptionV3 (55).

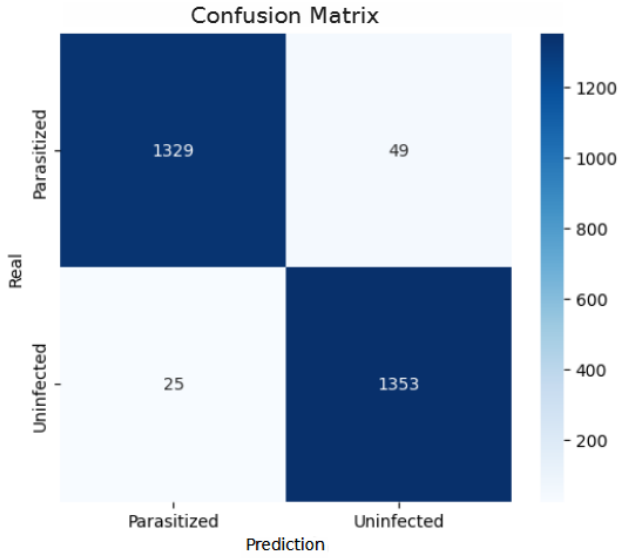


Fig. 14. ResNet50 confusion matrix in test set.

D. *Comparative Table of the Models in the EfficientNetB0, InceptionV3 and ResNet50 Test Set*

ResNet50 combined high accuracy (97%) with the shortest time (2 hours and 58 minutes) and FP (1.8%), ideal for accurate diagnostics. InceptionV3 had better recall for 'Parasitized' (98%), but required more resources (6.5h). Table IV summarizes all the values corresponding to the performance on the test set, the three models evaluated (EfficientNetB0, InceptionV3 and ResNet50) in terms of:

- Accuracy (Test): Overall percentage of hits.
- Precision (P/U): Predictive value for each class (Parasitized/Uninfected).
- Recall (P/U): Sensitivity to detect true cases.
- F1-Score: Balance between precision and recall.
- Training Time: Computational Efficiency.
- FN/FP: They are critical in medical diagnosis

TABLE IV. COMPARATIVE PERFORMANCE OF EFFICIENTNETB0, INCEPTIONV3 AND RESNET50 MODELS ON THE TEST SET

Metric	EfficientNetB0	InceptionV3	ResNet50
Accuracy (Test)	0.95	0.97	0.97
Precision (P/U)	0.97 / 0.94	0.96 / 0.97	0.98 / 0.97
Recall (P/U)	0.94 / 0.97	0.98 / 0.96	0.96 / 0.98
F1-Score	0.95	0.97	0.97
Training Time (h)	3.2	6.5	2.9
False Negatives (FN)	6.2%	2.5%	3.6%
False Positives (FP)	3.1%	4.0%	1.8%

Legend:

- P = Parasitized,
- U = Uninfected.

V. DISCUSSION

The results obtained showed that the three convolutional neural network models evaluated, EfficientNetB0, InceptionV3 and ResNet50, showcased excellent performance metrics in the automated detection of Plasmodium-infected cells. However, the main novelty of this study lay in the direct, systematic and reproducible comparison between these three models in a controlled environment, using a single data stream and uniform metrics, which has not been addressed in previous studies. This methodology allows a more accurate assessment of the clinical applicability of each architecture.

ResNet50 stood out for its balanced performance: it achieved 97% accuracy, 1.8% false positives and the shortest training time (2 hours and 58 minutes), positioning it as the most viable model for real implementations. Although Ferreras Extremo [13] achieved an accuracy of 98.29% using an ensemble of EfficientNet models, his approach demanded cross-validation of 10 iterations, increasing the complexity and computational time. In contrast, our ResNet50 achieved similar results with lower operational burden. With respect to Marín Calvo [14] who reported 95.58% accuracy, our InceptionV3 exceeded that value in recall (98%), which is key to avoid false negatives in medical contexts. Unlike Vizcaino Gispert [11] whose Faster R-CNN model only reached 2.01% in initial detection, our models exceeded 95% accuracy from the first trainings thanks to the use of transfer learning and balanced datasets. In addition, studies such as that of Sierra Segovia et al. [12] showed comparable accuracy, but with less stability between training and validation (5% gap vs. our 1.78% gap in ResNet50). This evidences the positive effect of regularizations such as Dropout and L2 used in our work. Also, although Zhao et al. [10] achieved good accuracy (96.5%) on mobile hardware, our ResNet50 model demonstrated superiority in accuracy with equal or better efficiency. Finally, Meza-Bautista et al. [15] found that EfficientNetB0 achieved 97.12%, although with fine-tuning techniques not applied in our study. Even so, our ResNet50 matched that accuracy with less technical complexity.

From a practical standpoint, ResNet50 reduced false negatives to 3.6%, which is critical in medical diagnostics, and had a specificity of 98%, thus reducing misdiagnosis and unnecessary treatment. InceptionV3 offered higher sensitivity (recall of 98%), but required 6.5 hours of training, which may limit its adoption in resource-poor settings. This analysis also highlights opportunities for future research. It is recommended to validate the models on real clinical images to assess their robustness outside the standardized dataset. In addition, the combination of models (ensembles) that integrate the speed of EfficientNetB0 with the accuracy of ResNet50 could be explored. Another relevant line is the implementation in low-cost devices (such as Raspberry Pi), optimizing models with TensorFlow Lite and quantization. Finally, these architectures could be adapted to other parasitic diseases such as leishmaniasis or babesiosis, expanding their impact on public health. Overall, this research provides clear empirical evidence on the performance of three popular models on a critical health problem. ResNet50, due to its balance between performance, efficiency and simplicity, is emerging as a robust and applicable solution in real clinical scenarios.

VI. Conclusions and Recommendations

This study evaluated and compared the performance of three pre-trained convolutional neural network models (EfficientNetB0, InceptionV3 and ResNet50) for automated detection of malaria-infected cells. The main contribution of this work lies in providing a systematic, reproducible and clinically applicable-oriented comparative analysis using a unified data flow and homogeneous metrics, which represents an approach rarely addressed in previous studies. The EfficientNetB0 model was characterized by its light weight and training speed (300 seconds per epoch), making it a viable alternative for environments with limited computational resources, such as mobile devices or rural areas. However, its performance on the validation set was inconsistent, with significant fluctuations in the accuracy of the validation set (val_accuracy) and a relatively high level of false negatives (6.2%), which compromises its reliability for direct application in clinical settings without additional adjustments. InceptionV3 demonstrated a high detection capability, reaching a recall of 98% for the "Parasitized" class, making it the most sensitive model in the study. Its stability during training was also remarkable, with a gap between training and validation of only 2.27%. However, this model had a higher false positive rate (4.0%) and considerable computational demand, with a total training time of approximately 6.5 hours learning. The ResNet50 model emerged as the most balanced option. It achieved an accuracy of 97%, similar to that of InceptionV3, but with a lower false positive rate (1.8%) and a smaller gap between training and validation metrics (1.78%), which evidences a greater generalization capacity. In addition, its training time was the most efficient among the three models evaluated (2 hours and 58 minutes), which reinforces its feasibility to be implemented in real clinical scenarios. In practical terms, ResNet50 is positioned as the best choice for medical settings where both false negatives and false positives need to be minimized. InceptionV3, on the other hand, may be more suitable in contexts where early detection is a priority and a higher false positive rate can be tolerated. EfficientNetB0 is an interesting alternative for embedded or hardware constrained solutions, although its clinical implementation would require further optimization.

As technical recommendations, we suggest adjusting the decision thresholds in the InceptionV3 and EfficientNetB0 models to values close to 0.4, and to 0.45 in ResNet50, in order to improve sensitivity without excessively compromising specificity. Also, implementing loss functions such as focal loss could help to reduce false negatives, especially in critical clinical classes. To control overfitting, it is recommended to increase regularization by Dropout (between 0.5 and 0.7), apply L2 penalty and use Batch Normalization in dense layers. From the optimization point of view, it is advisable to experiment with algorithms such as AdamW and dynamic learning rate adjustment strategies such as cosine decay. In addition, converting models to optimized formats such as TensorFlow Lite and applying quantization techniques can facilitate their execution on embedded hardware. For future research, it is proposed to validate the models on images from real clinical contexts beyond the standardized dataset, as well as to explore the development of hybrid models (ensembles) that combine efficiency and accuracy. It is also proposed to extend this methodology to the diagnosis of other parasitic diseases with similar cell morphology, such as leishmaniasis or babesiosis, and to implement complete systems in low-cost portable devices -such as Raspberry Pi with digital microscopy-, aimed at environments with limited access to medical infrastructure.

The novelty of this study lies not only in the results obtained, but also in its practical applicability. As part of the materialization of this research, the graphical interface NhAI-Malaria Classifier Fig. 15 was developed, a desktop tool that integrates the trained models (ResNet50, InceptionV3 and EfficientNetB0) to classify images of blood cells with malaria. The interface allows loading images manually or generating random samples, selecting the desired model and visualizing predictions with their associated probability. Designed to be intuitive and accessible, this GUI is ideal for resource-limited clinical settings, medical education or rapid sample validation. Its local operation guarantees data privacy, eliminating dependence on external connections, which makes it especially suitable for areas with low connectivity. Source code and deployment instructions are available at https://github.com/lu-mala/NhAI-Malaria-Classifier.

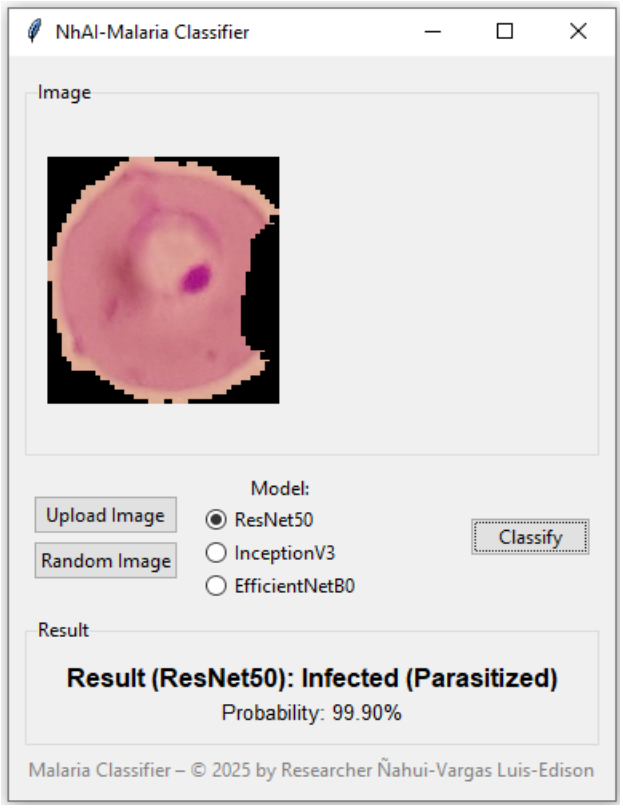


Fig. 15. NhAI-Malaria classifier graphical interface. Users can upload images, select a model and receive predictions with confidence levels.

References

[1] World Health Organization, "World malaria report 2021," World Health Organization. Accessed: Jan. 05, 2025. [Online]. Available: https://www.who.int/teams/global-malaria-programme/reports/world-malaria-report-2021

[2] "World malaria report 2024." Accessed: Jan. 05, 2025. [Online]. Available: https://www.who.int/teams/global-malaria-programme/reports/world-malaria-report-2024

[3] MINSA, "Alerta Epidemiologica," Centro Nacional de Epidemiologia, Prevención y Control de Enfermedades - MINSA. Accessed: Jan. 10, 2025. [Online]. Available: https://www.dge.gob.pe/epipublic/uploads/alertas/alertas_202218_04_141525.pdf

[4] MINSA, "Número de casos de malaria, Perú 2020 – 2024," Centro Nacional de Epidemiologia, Prevención y Control de Enfermedades - MINSA. Accessed: Jan. 07, 2025. [Online]. Available: https://www.dge.gob.pe/portal/docs/vigilancia/sala/2024/SE48/malaria.pdf

[5] D. Valdivia Blume, "Malaria se expande en Loreto: dos muertes y más de 8000 casos reportados," Infobae Perú. Accessed: Jan. 07, 2025. [Online]. Available: https://www.infobae.com/peru/2024/04/28/malaria-se-expande-en-loreto-dos-muertes-y-mas-de-8000-casos-reportados/

[6] World Health Organization, "A framework for malaria elimination," World Health Organization. Accessed: Jan. 10, 2025. [Online]. Available: https://www.who.int/publications/i/item/9789241511988

[7] Malaria No More, "Malaria Elimination Efforts," Malaria No More. Accessed: Jan. 10, 2025. [Online]. Available: https://www.malarianomore.org/news/malaria-know-more-kemri-scientist-dr-damaris-matoke-discusses-her-role-in-malaria-elimination-efforts/

[8] G. Litjens et al., "A survey on deep learning in medical image analysis," Med Image Anal, vol. 42, pp. 60–88, Dec. 2017, doi: 10.1016/J.MEDIA.2017.07.005.

[9] S. Rajaraman, S. Candemir, I. Kim, G. Thoma, and S. Antani, "Visualization and Interpretation of Convolutional Neural Network Predictions in Detecting Pneumonia in Pediatric Chest Radiographs," Applied Sciences 2018, Vol. 8, Page 1715, vol. 8, no. 10, p. 1715, Sep. 2018, doi: 10.3390/APP8101715.

[10] O. S. Zhao et al., "Convolutional neural networks to automate the screening of malaria in low-resource countries," PeerJ, vol. 8, p. e9674, Aug. 2020, doi: 10.7717/PEERJ.9674/FIG-6.

[11] B. Vizcaíno Gispert, "END-TO-END DEEP LEARNING FOR MALARIA DETECTION," Universitat Politècnica de Catalunya, Barcelona, 2020. Accessed: Jan. 03, 2025. [Online]. Available: https://upcommons.upc.edu/bitstream/handle/2117/177615/MemoriaBVG_v6.pdf

[12] P. Sierra Segovia, L. Ortiz Cedeño, and L. Zamacona Gómez, "DIAGNÓSTICO DE MALARIA MEDIANTE EL USO DE REDES NEURONALES CONVOLUCIONALE," UAITIE, 2021. Accessed: Jan. 03, 2025. [Online]. Available: https://premionacionaluaitie.uaitie.es/wp-content/uploads/2022/07/IES-Margarita-Salas-Diagnostico-de-Malaria-mediante-el-uso-de-redes-neuronales-convolucionales.pdf

[13] A. Ferreras Extremo, "Estudio de algoritmos de redes neuronales convolucionales en dataset de imágenes médicas," UNIVERSIDAD DE VALLADOLID, Valladolid, 2021. Accessed: Jan. 03, 2025. [Online]. Available: https://uvadoc.uva.es/handle/10324/47137

[14] H. Marín Calvo, "Diseño, implementación y validación de técnicas de identificación de células infectadas de malaria mediante redes neuronales," UNIVERSITAT POLITÈCNICA DE VALÈNCIA, 2022. Accessed: Jan. 03, 2025. [Online]. Available: https://riunet.upv.es/handle/10251/187062

[15] A. Meza-Bautista, L. E. Ñahui-Vargas, E. Mamani-Vilca, and R. Micaela, "Comparativa de Modelos basados en redes neuronales convolucionales: ResNet-50V2, MobileNetV2 e EfficientNetB0 en la detección de Malaria," Micaela Revista de Investigación - UNAMBA, vol. 5, no. 1, pp. 42–49, Oct. 2024, doi: 10.57166/MICAELA.V5.N1.2024.138.

[16] R. Hernández Sampieri, C. Feránadez Collado, and M. D. P. Baptista Lucio, "Metodología de la investigación," Metodología de la investigación, p. 91, 2014, Accessed: Jan. 05, 2025. [Online]. Available: https://dialnet.unirioja.es/servlet/libro?codigo=775008&info=resumen&idioma=SPA

[17] I. Goodfellow, Y. Bengio, and A. Courville, "Deep Learning," MIT Press. Accessed: Jan. 03, 2025. [Online]. Available: https://www.deeplearningbook.org/

[18] "Malaria Cell Images Dataset," Kaggle. Accessed: Jan. 05, 2025. [Online]. Available: https://www.kaggle.com/datasets/iarunava/cell-images-for-detecting-malaria

[19] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," J Big Data, vol. 6, no. 1, pp. 1–48, Dec. 2019, doi: 10.1186/S40537-019-0197-0/FIGURES/33.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," En Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778, 2016, [Online]. Available: http://image-net.org/challenges/LSVRC/2015/

[21] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," 2009 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 248–255, 2009, doi: 10.1109/CVPR.2009.5206848.

[22] O. Russakovsky et al., "ImageNet Large Scale Visual Recognition Challenge," Int J Comput Vis, vol. 115, no. 3, pp. 211–252, Dec. 2015, doi: 10.1007/S11263-015-0816-Y/FIGURES/16.

[23] S. R. Sannasi Chakravarthy, N. Bharanidharan, C. Vinothini, V. Vinoth Kumar, T. R. Mahesh, and S. Guluwadi, "Adaptive Mish activation and ranger optimizer-based SEA-ResNet50 model with explainable AI for multiclass classification of COVID-19 chest X-ray images," BMC Med Imaging, vol. 24, no. 1, pp. 1–23, Dec. 2024, doi: 10.1186/S12880-024-01394-2/TABLES/4.

[24] M. Tan and Q. V Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," Proceedings of Machine Learning Research, PMLR, pp. 6105–6114, May 24, 2019. Accessed: Jan. 11, 2025. [Online]. Available: https://proceedings.mlr.press/v97/tan19a.html

[25] H. Amin, A. Darwish, A. E. Hassanien, and M. Soliman, "End-to-End Deep Learning Model for Corn Leaf Disease Classification," IEEE Access, vol. 10, pp. 31103–31115, 2022, doi: 10.1109/ACCESS.2022.3159678.

[26] C. Szegedy, V. Vanhoucke, V. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," IEEE conference on computer vision, pp. 2818–2826, 2016, Accessed: Jan. 11, 2025. [Online]. Available: https://www.cv-foundation.org/openaccess/content_cvpr_2016/html/Szegedy_Rethinking_the_Inception_CVPR_2016_paper.html

[27] M. Pino, "LA INNOVACIÓN SOCIAL EN EL MARCO DEL DESARROLLO URBANO SOSTENIBLE: EVALUACIÓN DEL PROYECTO TROPA VERDE EN SANTIAGO DE COMPOSTELA," Entre ciencia e Ingenieria, pp. 41–56, Mar. 2022, doi: 10.22533/AT.ED.4002229035.

[28] G. Van Rossum and F. L. Drake, Python 3 Reference Manual; CreateSpace. 2009. Accessed: Jan. 05, 2025. [Online]. Available: https://www.python.org/

[29] E. Bisong, Building Machine Learning and Deep Learning Models on Google Cloud Platform. Apress, 2019. doi: 10.1007/978-1-4842-4470-8.

[30] M. Abadi et al., {TensorFlow}: A System for {Large-Scale} Machine Learning. 2016. Accessed: Jan. 05, 2025. [Online]. Available: https://tensorflow.org.

[31] P. Y. Simard, D. Steinkraus, and J. C. Platt, "Best practices for convolutional neural networks applied to visual document analysis," Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, vol. 2003-January, pp. 958–963, 2003, doi: 10.1109/ICDAR.2003.1227801.

[32] F. Chollet, "Deep Learning with Python," Manning Publications Co. Accessed: Jan. 04, 2025. [Online]. Available: https://scholar.google.com/scholar?hl=es&as_sdt=0%2C5&q=Deep+Learning+with+Python.+Manning+Publications.&btnG=#d=gs_cit&t=1736136253013&u=%2Fscholar%3Fq%3Dinfo%3AKv3TToGRbtQJ%3Ascholar.google.com%2F%26output%3Dcite%26scirp%3D0%26hl%3Des

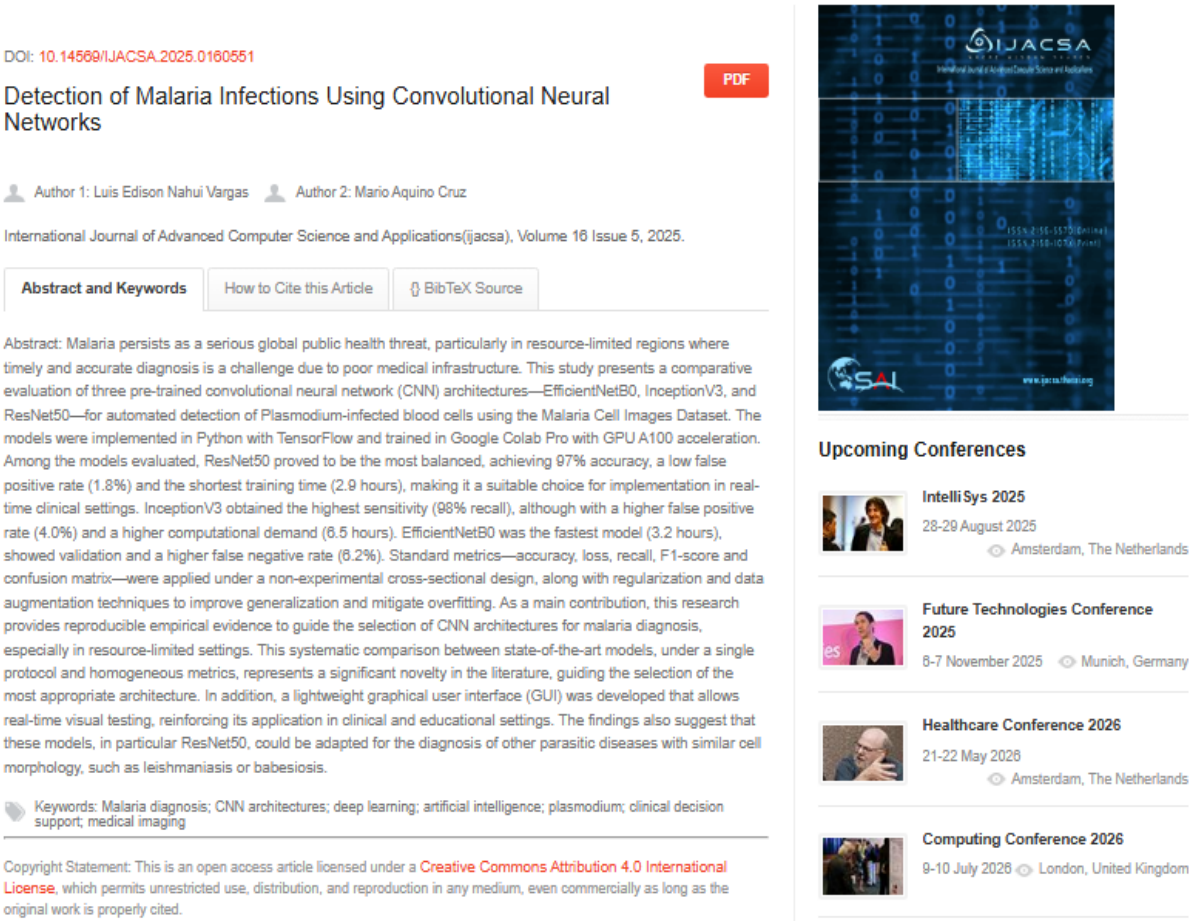
[33] A. A. Taha and A. Hanbury, "Metrics for evaluating 3D medical image segmentation: Analysis, selection, and tool," BMC Med Imaging, vol. 15, no. 1, pp. 1–28, Aug. 2015, doi: 10.1186/S12880-015-0068-X/TABLES/5.

[34] A. Esteva et al., "Dermatologist-level classification of skin cancer with deep neural networks," Nature 2017 542:7639, vol. 542, no. 7639, pp. 115–118, Jan. 2017, doi: 10.1038/nature21056.

[35] P. Rajpurkar et al., "CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning," Nov. 2017, Accessed: Jan. 04, 2025. [Online]. Available: https://arxiv.org/abs/1711.05225v3.

ANEXOS

ANEXO 01. Portada de la publicación de la revista



DOI: 10.14569/IJACSA.2025.0160551

PDF

Detection of Malaria Infections Using Convolutional Neural Networks

Author 1: Luis Edison Nahui Vargas Author 2: Mario Aquino Cruz

International Journal of Advanced Computer Science and Applications(ijacsa), Volume 16 Issue 5, 2025.

Abstract and Keywords | How to Cite this Article | {} BibTeX Source

Abstract: Malaria persists as a serious global public health threat, particularly in resource-limited regions where timely and accurate diagnosis is a challenge due to poor medical infrastructure. This study presents a comparative evaluation of three pre-trained convolutional neural network (CNN) architectures—EfficientNetB0, InceptionV3, and ResNet50—for automated detection of Plasmodium-infected blood cells using the Malaria Cell Images Dataset. The models were implemented in Python with TensorFlow and trained in Google Colab Pro with GPU A100 acceleration. Among the models evaluated, ResNet50 proved to be the most balanced, achieving 97% accuracy, a low false positive rate (1.8%) and the shortest training time (2.9 hours), making it a suitable choice for implementation in real-time clinical settings. InceptionV3 obtained the highest sensitivity (98% recall), although with a higher false positive rate (4.0%) and a higher computational demand (6.5 hours). EfficientNetB0 was the fastest model (3.2 hours), showed validation and a higher false negative rate (6.2%). Standard metrics—accuracy, loss, recall, F1-score and confusion matrix—were applied under a non-experimental cross-sectional design, along with regularization and data augmentation techniques to improve generalization and mitigate overfitting. As a main contribution, this research provides reproducible empirical evidence to guide the selection of CNN architectures for malaria diagnosis, especially in resource-limited settings. This systematic comparison between state-of-the-art models, under a single protocol and homogeneous metrics, represents a significant novelty in the literature, guiding the selection of the most appropriate architecture. In addition, a lightweight graphical user interface (GUI) was developed that allows real-time visual testing, reinforcing its application in clinical and educational settings. The findings also suggest that these models, in particular ResNet50, could be adapted for the diagnosis of other parasitic diseases with similar cell morphology, such as leishmaniasis or babesiosis.

Keywords: Malaria diagnosis; CNN architectures; deep learning; artificial intelligence; plasmodium; clinical decision support; medical imaging

Copyright Statement: This is an open access article licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, even commercially as long as the original work is properly cited.

Upcoming Conferences

IntelliSys 2025
28-29 August 2025 Amsterdam, The Netherlands

Future Technologies Conference 2025
6-7 November 2025 Munich, Germany

Healthcare Conference 2026
21-22 May 2026 Amsterdam, The Netherlands

Computing Conference 2026
9-10 July 2026 London, United Kingdom

Figura 16 — Portada de la publicación de la revista *International Journal of Advanced Computer Science and Applications (IJACSA)*. Disponible en: https://dx.doi.org/10.14569/IJACSA.2025.0160551.