Wing-Based Identification of Dengue Vector Mosquitoes Using Convolutional Neural Networks

Lucas Ferreira Quintão Moreira, Agnaldo José da Rocha Reis, and Alan Kardek Rêgo Segundo

***Abstract*—Given the increasing number of dengue cases in Brazil, accurately identifying vector mosquitoes is essential for effective control and prevention strategies. This study proposes a classification model capable of distinguishing *Aedes aegypti* and *Aedes albopictus* based on wing images from the WingBank database of the Butantan Institute. Two convolutional neural network architectures, MobileNetV2 and EfficientNet-B0, were evaluated. Data augmentation was applied to address the limited number of samples. The models were tested on a Cortex-A processor, demonstrating that high-accuracy mosquito classification can be achieved on embedded devices. These results highlight the potential of the proposed approach to support scalable, real-time mosquito monitoring and vector control systems.**

Link to graphical and video abstracts, and to code: https://latamt.ieeer9.org/index.php/transactions/article/view/10106

***Index Terms*—Dengue, MobileNetV2, EfficientNet-B0, Classification, Edge Impulse.**

I. Introduction

BRAZIL reported a record number of dengue-related deaths in 2024, accompanied by an increase in cases compared to 2023, according to data from the Ministry of Health's Arbovirus Panel. These findings highlight why *Aedes aegypti*, the primary vector of diseases such as dengue, yellow fever, chikungunya, and Zika, is regarded as one of the most concerning mosquitoes by the World Health Organization (WHO).

Aedes aegypti originates from Egypt and is currently distributed across tropical and subtropical regions, including the Americas, Southeast Asia, and India [1]. The species is estimated to have arrived during the colonial period, between the 16th and 19th centuries. In contrast to *Aedes albopictus*, a secondary vector more commonly associated with rural environments, *Aedes aegypti* is predominantly found in urban areas, where it breeds in water storage containers. Overall, the seasonal dynamics of this vector are strongly associated with climatic variations, including increases in temperature, changes in rainfall patterns, and relative humidity, which are characteristic of the Brazilian summer season [2].

The associate editor coordinating the review of this manuscript and approving it for publication was Ruth Aguilar (*Corresponding author: Lucas Ferreira Quintao Moreira*).

This work was supported by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brazil (CAPES), Finance Code 001, and the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

Lucas Ferreira Quintão Moreira, A. J. D. R. Reis, and A. K. R. Segundo are with the Federal University of Ouro Preto, Ouro Preto, Minas Gerais, Brazil (e-mails: lucas.quintao@aluno.ufop.edu.br, reis@ufop.edu.br, and alankardek@ufop.edu.br).

In large cities, this mosquito demonstrates high adaptability to urban environments, with larvae frequently detected in gutters, tires, empty bottles, and other sites where water accumulates. The species displays diurnal activity, depositing its eggs mainly at dawn and in the afternoon [3]. Demographic growth combined with intensive urbanization has contributed to the rising incidence of the disease. At the same time, dengue control poses increasing challenges, since the vector shows resistance to most current control strategies.

The primary control strategy against *Aedes aegypti* in Brazil consists of identifying and eliminating water reservoirs that serve as breeding sites [4]. To reinforce this approach, the federal government has promoted public educational campaigns aimed at encouraging the removal of potential mosquito habitats. Chemical control also plays a central role, relying on the application of insecticidal products to target both larvae and adult mosquitoes. Nevertheless, shortcomings in the identification process may result in ineffective control measures.

Various technologies have been employed to support the control of *Aedes aegypti* and *Aedes albopictus*, including selective infestation monitoring, the use of biological control agents, molecular techniques for population management, and integrated applications of these methods [5]. Building on recent advances in Artificial Intelligence (AI), this study proposes an alternative approach for identifying dengue vector mosquitoes. Specifically, Convolutional Neural Networks (CNNs) are applied to classify *Aedes aegypti* and *Aedes albopictus* using wing images.

Initially, the fundamental concepts and principles of the project were presented, followed by the testing and evaluation of the results. Two convolutional neural network models, MobileNetV2 and EfficientNet-B0, were developed using the Edge Impulse platform, which provides a user-friendly interface for building AI models. This platform was employed to train and test both models.

The main goal of this research is to assess and compare the performance of MobileNetV2 and EfficientNet-B0 in identifying dengue vector mosquitoes using wing images. Both models were tested on a platform designed for devices with limited memory and processing capacity, and a comparative analysis of their classification accuracy was conducted to evaluate their effectiveness. The choice of these AI models is motivated by the goal of implementing a dengue classifier on a smartphone in the near future. Therefore, the model must be both accurate and lightweight, with minimal computational requirements.

This paper is organized as follows. Section II reviews the research that inspired the project, providing details on relevant

Fig. 1. Mapping of water tanks and swimming pools from aerial imagery. Source: [8].

studies in this field. Section III presents the theoretical discussion of the MobileNetV2 and EfficientNetB0 architectures, describing their main characteristics. Section IV discusses the dataset used, the data augmentation techniques applied, the model development and configuration process. Section V presents the results and discussions regarding the model performance. Finally, Section VI provides the conclusions and future work.

II. Literature Review

One of the earliest works evaluated in this study is the article proposed by [7], which introduces the application of YOLO (You Only Look Once) for identifying potential breeding sites of *Aedes aegypti*. The study analyzed images of mosquito breeding sites collected using drones. Unmanned Aerial Vehicles (UAVs) facilitate the inspection of hard-to-access areas while requiring fewer human and financial resources compared to other aircraft-based methods. The YOLOv4 architecture, released in 2020, was employed, enabling real-time recognition of up to 80 objects in images and videos. Training was conducted on a dataset comprising approximately 500 images from the metropolitan region of São Paulo. This approach demonstrated significant results in terms of precision, as reported in [7].

Another similar study, conducted by researchers from the Federal University of Minas Gerais (UFMG), the University of São Paulo (USP), and the University of Sheffield (United Kingdom), improved AI-based software to identify dengue breeding sites from aerial images [8] (Fig. 1). The research employed the Premise Condition Index (PCI) to map risk areas and guide control efforts. The PCI evaluates building and backyard conditions, as well as shading, using data collected from approximately 200 city blocks in Campinas, São Paulo. A deep neural network was trained with these images to predict building conditions based on facade visualization, resulting in a computational model named PCINet.

At the Federal University of Pernambuco (UFPE), an automatic mosquito egg counting system was proposed using deep learning methods [9]. The study utilized modern ovitraps, which traditionally require considerable human intervention. Ovitraps are dark containers with wide openings, partially filled with water, containing a rough wooden pallet installed vertically inside. To automate the device, a CNN was trained on a dataset of *Aedes aegypti* egg images. YOLOv4 was employed, achieving significant overall accuracy for the task

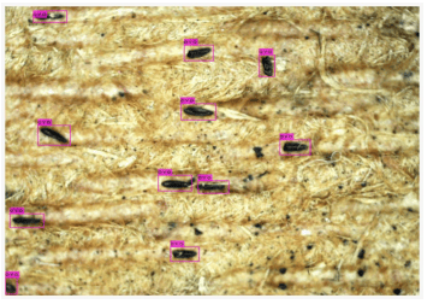


Fig. 2. Recognition of dengue mosquito eggs using YOLOv4. Source: [9].

(Fig. 2) and further demonstrating the potential of computer vision techniques in dengue control.

Similarly, a study conducted in the city of Natal, Rio Grande do Norte (RN), employed a deep learning algorithm to predict future dengue incidence based on the previous week's reported cases or the number of eggs detected in ovitraps [10]. The analysis used data from 397 ovitraps distributed throughout the city. The application of this model yielded valuable insights for planning public health interventions.

On the other hand, the use of AI for mosquito species identification based on wing morphology serves as an important foundation. A notable example is a master's research project conducted at the University of São Paulo (USP), which applied AI techniques to classify mosquito species using wing morphometric features [11]. The analysis focused on key species, including *Aedes*, *Anopheles*, and *Culex*. Traditional identification methods rely on morphological characteristics and taxonomic keys, enabling the application of Wing Geometric Morphometry (WGM). By using artificial neural networks, the researchers were able to recognize patterns in the wing images and classify the specimens according to their morphometric features [11].

A total of 180 wing images from six mosquito species belonging to the genera *Culex*, *Aedes*, *Wyeomia*, and *Anopheles* were analyzed. Anatomical landmarks were extracted and used for morphometric-based image classification. The automation process and digital image processing were implemented in Python using the Random Forest algorithm. Model performance was evaluated by calculating the Mahalanobis distance between predicted and reference landmarks, resulting in the correct classification of at least one image of *Culex quinquefasciatus*.

In addition, another study explored the use of deep learning models to classify mosquito species characterized by high inter-species similarity and intra-species variation [12]. To achieve this, a dataset of 3,600 images from 8 mosquito species was constructed, including variations in posture and deformation, in order to explore the feasibility of transferring general features learned from generic datasets to mosquito classification. In this context, the implemented CNN achieved more than 97% classification accuracy with appropriate data augmentation techniques were applied [12].

The same approach was observed in [13], where a CNN was developed to identify *Aedes* species using wing images exclu-

sively, facilitating standardized image capture and reducing the complexity of CNN implementation. In this case, a dataset was built with 1,155 wing images of *Aedes* and 554 wing images of non-*Aedes* mosquitoes. Image processing and data augmentation techniques were applied using the PyTorch deep-learning framework with both grayscale and RGB images. The model achieved a mean macro F1 score of 90% for grayscale images and 91% for RGB images [13].

Additionally, investigations in signal processing provide an alternative approach for mosquito identification. One study proposed an algorithm to detect dengue-infected *Aedes aegypti* mosquitoes using their wingbeat signals as input, employing a Recurrent Neural Network (RNN) based on Long Short-Term Memory (LSTM) [14]. In this case, the methodology identified approximately 5% of the input wingbeat signals as uncertain and achieved a classification accuracy of 94.87% on the remaining data.

Therefore, recent studies on the general identification of mosquitoes show that the use of CNNs provides an innovative and distinctive approach to public health challenges. This work advances research by comparing the performance of two state-of-the-art architectures, MobileNetV2 and EfficientNet-B0, both proposed by the Edge Impulse platform.

III. Theoretical Foundations

Computer vision is a field of science that enables machine vision, allowing computers to perceive the world and extract meaningful information from images captured by cameras, sensors, or scanners [15]. This capability makes it possible to automate tasks, such as facial recognition, which previously required extensive manual annotation. With technological advances, image processing has become more efficient, and CNNs have emerged as one of the most effective architectures for image recognition. Unlike dense multilayer networks, CNNs operate on local receptive fields, enhancing their performance in computer vision applications [16].

Local receptive fields in CNNs allow pattern recognition in different regions of an image while reducing dimensionality by decreasing the number of parameters and improving computational efficiency. Instead of matrix multiplication, CNNs perform convolution operations, where kernels transform the input image into feature maps. In this context, architectures such as ResNet, based on residual learning, use skip connections to provide direct information flow between shallow and deep layers, thereby improving performance in tasks like mosquito wing image classification [17].

Another relevant architecture is DenseNet [18], characterized by dense connections in which each layer receives feature maps from all previous ones. This structure promotes feature reuse, resulting in more compact models and reducing overfitting risks. Nevertheless, given the focus on lightweight solutions for embedded applications, this study emphasizes architectures specifically designed for efficiency, such as MobileNetV2 [19] and EfficientNet-B0 [20], both available through the Edge Impulse platform, which will be discussed in the sequel.

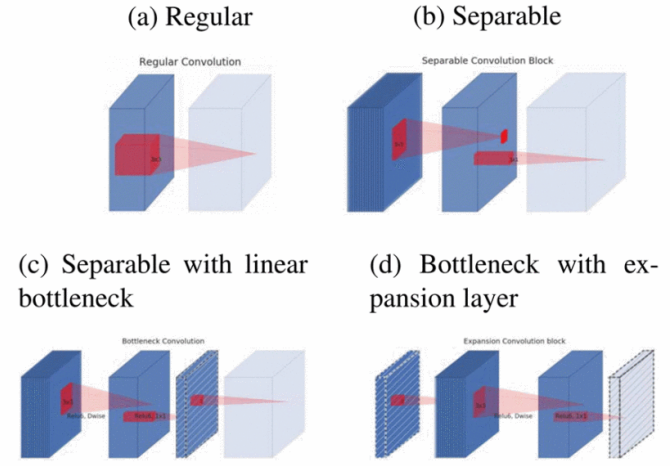


Fig. 3. Evolution of separable convolution blocks. Diagonally hatched regions indicate layers without nonlinearities. The final shaded layer marks the beginning of the next block. Note that 3d and 3c are equivalent when stacked. Source: [19].

A. MobileNetV2

MobileNetV2 employs Depthwise Separable Convolutions (DSC) to improve efficiency and portability. It also mitigates information loss in nonlinear layers by adopting linear bottlenecks and introducing inverted residuals, which help to preserve feature information [21]. The fundamentals of MobileNetV2 are presented below:

1) **Depthwise Separable Convolutions**. The main idea is to replace a full convolutional operator with a factorized version that splits the convolution into two separate layers [19]. In this approach, the first layer performs a lightweight operation by applying a single convolutional filter per input channel, while the second layer is responsible for building new features by computing linear combinations of the input channels.
2) **Linear Bottlenecks**. Linear bottlenecks are layers that project high-dimensional activations into a lower-dimensional subspace before applying ReLU. This helps preserve important information that might otherwise be lost when ReLU zeros out channels, allowing subsequent linear operations to maintain efficient representations [21].
3) **Inverted Residuals**. The bottleneck blocks are similar to residual blocks, that is, they consist of an input, several bottleneck layers, and an expansion layer [19]. The objective is to retain most of the essential information, while the expansion layer applies a non-linear transformation.

The Fig. 3 shows all the points described above. In general, this structure in the model uses bottleneck depthwise separable convolutions with residuals, expanding and compressing representations to preserve information, remain lightweight, and allow flexible filtering for specific classification tasks via transfer learning [21].

B. EfficientNet-B0

Another approach evaluated in this study is EfficientNet, specifically EfficientNet-B0. This network builds deep neural networks using inverted residual blocks, which are then combined in a Stacking Block by sequentially stacking several inverted residual blocks [22]. At the end of the network, a Global Average Pooling layer is applied, followed by a Fully Connected layer, which reduces the spatial dimensions to a $1 \times 1 \times N$ tensor, where N is the number of channels.

In this context, Edge Impulse is an online platform that streamlines the machine learning workflow, supporting the development of models for edge devices. It provides tools for data collection, model training, performance evaluation, and deployment on embedded systems [23]. The platform also allows users to customize solutions by adjusting hyperparameters, modifying layers, and applying data augmentation techniques. Moreover, it includes pre-configured architectures and classification models, such as MobileNetV2 and EfficientNet-B0, which are the focus of evaluation in this study.

Using Python-based methodologies, it is possible to design a model from scratch; however, this approach can be complex and less accessible to individuals without programming expertise. In contrast, the Edge Impulse platform provides a more intuitive workflow, simplifying model construction and interpretation while also streamlining deployment on embedded devices, enabling the development of a classifier for dengue vector mosquitoes as described in this project.

IV. Material and Methods

A. Dataset

The Butantan Institute is the largest producer of serums and vaccines in Latin America and a global reference for efficiency and quality. Among its activities, it stands out for producing hyperimmune serums against the venom of poisonous animals, bacterial toxins, and the rabies virus. It is also responsible for the largest national volume of vaccine antigens. Furthermore, it promotes studies and research in biology and biomedicine, with the mission of researching, developing, manufacturing, and providing products and services for public health [24].

In 2024, researchers from the Butantan Institute released a mosquito wing dataset to support the monitoring of dengue vectors and other arboviruses. Called WingBank [25], this platform contains 14,000 images from 80 species, with 2,100 images available for public access. One reason wings are so valuable for this kind of research is their ability to reveal genetic variability within species, which makes it possible to assess their evolutionary potential. In this sense, wing shape and appearance reflect the genetic characteristics of the mosquito species.

During the data collection process, it was observed that some images on the platform (originally 2,100 available) were not accessible for download, resulting in a final set of 1,261 images across the selected species. Table I presents the number of images and their proportional representation for each species class in the WingBank platform. Although the final dataset remained unbalanced across species, manual selection during data collection sought to preserve a balanced proportion of male and female wing images within each species, ensuring adequate traceability.

TABLE I
Number of Images and Percentage Distribution of each Species Class in the WingBank Dataset

Species	Number of Images	Percentage (%)
Aedes aegypti	448	35.53
Aedes albopictus	194	15.38
Aedes scapularis	112	8.88
Anopheles albitarsis	15	1.19
Anopheles arthuri	7	0.56
Anopheles cruzii	290	23.00
Anopheles homunculus	49	3.89
Anopheles strodei	25	1.98
Anopheles triannulatus	48	3.81
Culex nigripalpus	73	5.78

Although the primary focus of this study is on *Aedes aegypti* and *Aedes albopictus*, the other species present in the dataset are also considered to evaluate the classifier's behavior in a broader context. Species with low percentage representation, such as *Anopheles albitarsis*, *Anopheles arthuri*, and *Anopheles strodei*, were excluded to ensure sufficient data for training, resulting in a dataset comprising 1,214 images. The distribution of these images is shown in Fig. 4.

Considering the distribution, species such as *Culex nigripalpus* and *Aedes scapularis* form well-defined and clearly separated clusters, with little or no overlap with other classes. In contrast, *Anopheles cruzii*, *Anopheles homunculus*, and *Anopheles triannulatus* are more challenging, as it is difficult to draw clear boundaries between them. Regarding *Aedes aegypti* and *Aedes albopictus*, a moderate issue exists because these two species are very close and exhibit significant overlap.

This difficulty arises because both species belong to the same genus (*Aedes*) and therefore share many morphological characteristics, making classification non-trivial. As a result, the model may occasionally misclassify *Aedes aegypti* as *Aedes albopictus* and vice versa. Additionally, the noticeable class imbalance will be addressed through the application of data augmentation techniques.

B. Data Augmentation

Data augmentation techniques play an important role in improving the performance of deep learning models, including convolutional neural networks. When databases present limitations such as class imbalance, this strategy of generating synthetic samples from real data becomes a promising approach [26]. Techniques like flipping, rotation, translation, and resizing are commonly applied to improve the effectiveness of image classification tasks.

In this context, the following basic techniques are applied to illustrate the process in the selected classes: flipping, rotation, and resizing. Since dengue females are directly responsible for disease transmission, they are used as the primary example.

1) **Flipping**: this technique consists of mirroring the image along a specific axis, generating a new version of the original. Fig. 5a and Fig. 5b present the results of vertical

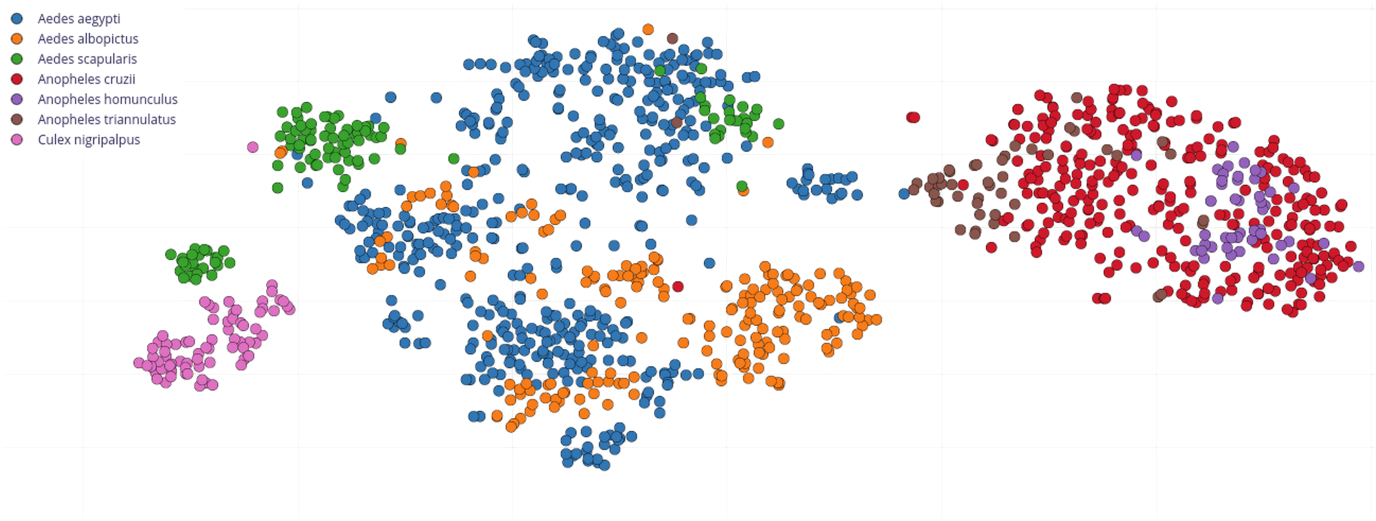
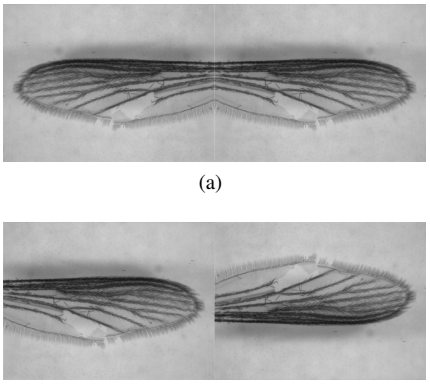


Fig. 4. Distribution of the selected classes after excluding species with low percentage relevance. Source: Author.



(a)

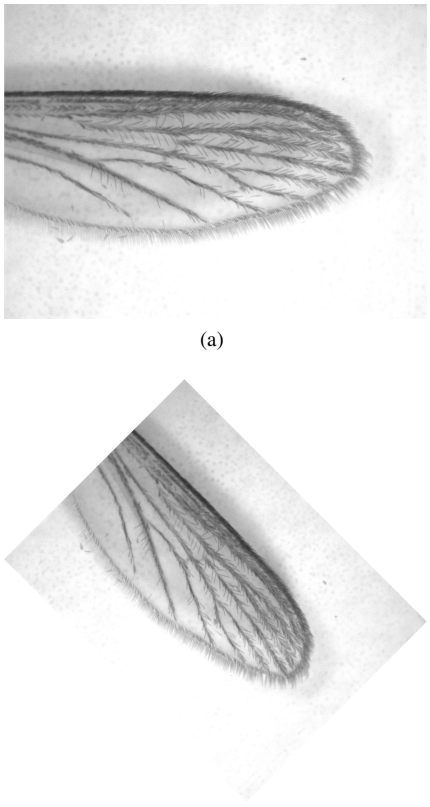
(b)

Fig. 5. Horizontal flipping of a left wing of the female *Aedes aegypti* (a). Vertical flipping of the right wing of a female *Aedes aegypti* (b). Source: Author.

and horizontal flipping applied to the left wing of a female *Aedes aegypti*.

2) **Rotation**: this technique consists of rotating an image by a specific angle around its center. In this example, a $45°$ rotation was applied to the right wing of an *Aedes scapularis* (Fig. 6a), and the result is shown in Fig. 6b. A $45°$ rotation is particularly relevant as it simulates the range of orientations that may occur in practical scenarios.
3) **Resizing**: this technique consists of adjusting the image dimensions (height and width), modifying its resolution and physical size while preserving the essential characteristics. In this example, resizing the wing of *Anopheles cruzii* (Fig. 7a) produces the result shown in Fig. 7b. The final resolution of 800 × 600 pixels was empirically, ensuring adequate detail to capture the relevant morphological features of the wings.

Edge Impulse allows users to manually select and configure data augmentation techniques to improve class balance and increase dataset diversity. In the following sections, the model



(a)

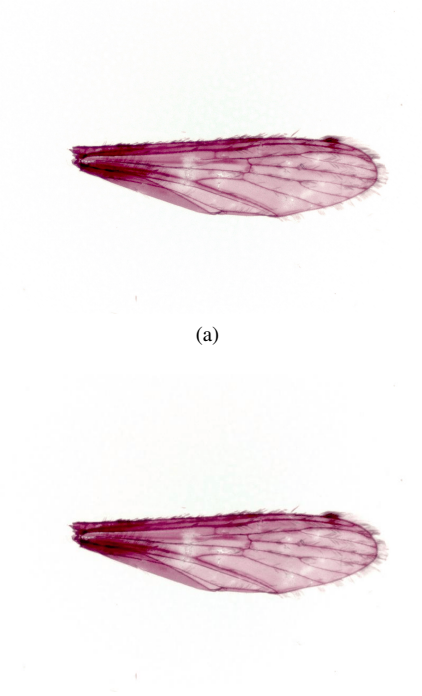
(b)

Fig. 6. Right wing of a female *Aedes scapularis* (a). $45°$ rotation of the right wing of a female *Aedes scapularis* (b). Source: Author.

implementation is presented, including the data split and relevant platform configurations. To evaluate the effectiveness of the chosen augmentation techniques, the model will be tested both with and without their application.

C. Model Development and Configuration

A flowchart (Fig. 8) was developed to detail the model implementation process. In general, the data is first collected



(a)

(b)

Fig. 7. Left wing of a female *Anopheles cruzii* (a). Resizing of the right wing of a female *Anopheles cruzii* from 2,083 × 1,544 to 800 × 600 pixels (b). Source: Author.

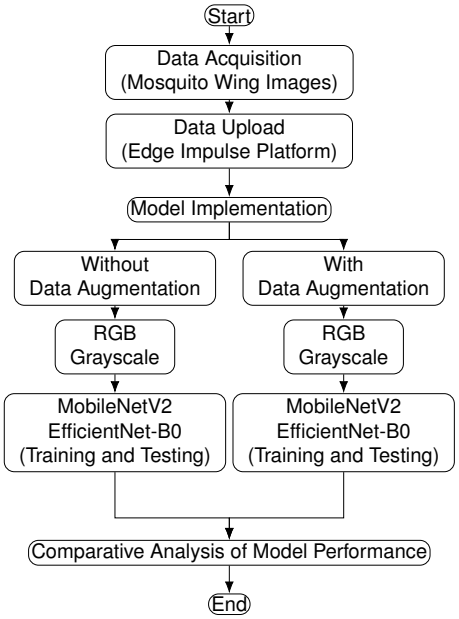


Fig. 8. Flowchart of model development. Source: Author.

from the WingBank platform and then uploaded to Edge Impulse. At this stage, the model implementation begins, including the configuration for the selected embedded system, the division of data into training and testing sets, and the setup of the Transfer Learning framework. The model is then evaluated under scenarios with and without data augmentation. Finally, the results are compared using relevant performance metrics.

TABLE II
TARGET DEVICE CONFIGURATION FOR MOBILE DEPLOYMENT

Parameter	Configuration
Processor family	ARM Cortex-A (ARMv8-A)
Clock rate	2 GHz
RAM	4 GB
ROM	32 GB
Latency	100 ms

As pointed out earlier, Edge Impulse enables the development of models for embedded systems. In this work, a processor from the Cortex-A family was selected, with its specifications detailed in Table II. These processors are commonly employed in mobile devices, offering a realistic and efficient platform for testing purposes. The reported latency of 100 ms corresponds to an empirical measurement obtained while running the CNN models on this processor under standard testing conditions. It is worth emphasizing that the hardware choice in this study serves solely to evaluate model performance, rather than to implement the complete embedded solution at this moment.

The images of mosquito wings were resized to a 96×96 format (width × height) using the fit-to-shortest-axis method, where the outer regions of the longer axis are cropped and the image is interpolated to the target size while preserving proportions. This resolution was chosen because it preserves key morphological features of the wings, such as vein patterns and spots, which are critical for species classification. It also reduces computational cost and is compatible with the evaluated architectures (MobileNetV2 and EfficientNet-B0) on the Edge Impulse platform, which is optimized for low-resource devices. Color depth (grayscale or RGB) was selected according to each evaluation scenario described in the flowchart [13].

Thus, the dataset was organized on the platform, with 80% allocated for training and 20% for testing. During the training phase, 20% of the training data was automatically set aside for validation. These splits ensure that the model is properly trained and evaluated. To enhance the reliability of the evaluation, a 4-fold cross-validation approach was adopted, maintaining the 80/20 proportion in each fold.

According to the Edge Impulse ebook, data is typically split by random sampling according to the specified proportions. However, because the dataset contains multiple subgroups, some of which are underrepresented, stratified sampling was employed. In this technique, the splits are performed individually within each subgroup and then combined. This approach helps prevent similar or identical specimens from being shared between training and validation sets, ensuring a more reliable evaluation during the 4-fold cross-validation process.

Transfer learning was applied to leverage knowledge from a pre-trained model and improve learning on a new task. In this approach, suppose there are two distributions, P1 and P2. Knowledge gained from P1 is used to enhance learning on P2, under the assumption that many of the features learned from P1 remain relevant for P2 [27]. Alternatively, the models could be trained from scratch, without relying on pre-trained

TABLE III
HYPERPARAMETERS USED FOR TRAINING THE MODELS, INCLUDING THE NUMBER OF EPOCHS (N), LEARNING RATE (ζ), AND BATCH SIZE (B)

Models	Epochs (N)	Learning Rate (ζ)	Batch size (B)
MobileNetV2	30	0.001	32
EfficientNet-B0	40	0.0005	32

weights, which would require learning all features directly from the target dataset. In this study, the pre-trained models were adapted to the new task using transfer learning.

Regarding the models, some configurations are defined directly on the platform. The MobileNetV2 architecture has 7 neurons in the final layer, corresponding to the 7 target classes, and a Dropout rate of 0.1. For the EfficientNet-B0 model, an additional dense layer with 64 neurons is added before the final classification layer to enhance feature extraction, while applying the same Dropout rate as in MobileNetV2 [28]. The number of neurons in the additional layer was chosen based on standard practice.

V. RESULTS AND DISCUSSION

A. Training

The models were trained using only the Central Processing Unit (CPU). Therefore, specific configurations were defined for the training process, particularly regarding the following hyperparameters:

1) **Epochs** (N): refers to a complete pass through the training dataset. An inappropriate choice of this value can lead to overfitting, reducing the model's ability to generalize to unseen data.
2) ***Learning Rate*** (ζ): defines the rate at which the model's weights are updated in each iteration, that is, the speed at which the model learns.
3) ***Batch size*** (B): indicates the number of training examples processed in each iteration.

Several hyperparameter combinations were explored in preliminary experiments, including variations in the number of epochs and learning rates. The configuration presented in Table III was selected as it provided the best balance between stable convergence and adequate generalization, as confirmed by the training loss and accuracy trends shown in Fig. 9. A 4-fold cross-validation approach was applied to rigorously assess this configuration, and Early Stopping was employed to further prevent overfitting.

Early Stopping [29] monitors the model's performance on a validation set during training and stops the training process when performance ceases to improve for a predefined number of epochs, thereby preventing the model from overfitting the training data. In this study, an early stopping patience of 8 epochs was applied, based on preliminary experiments that indicated this value provided stable convergence while preventing overfitting.

According to the learning curves, the two models exhibited distinct behaviors. EfficientNetB0 converged very rapidly in training loss and later showed an increase and larger fluctuations in the validation loss, indicating overfitting under the initial training schedule. Early stopping, however, mitigated this behavior by halting training before the divergence became more pronounced, leading to improved generalization. In contrast, MobileNetV2 displayed a more gradual and stable decrease in both training and validation loss, with only mild oscillations throughout training. The next section presents the model's performance according to standard evaluation metrics.

B. Model Performance

Accuracy (1) is one of the main evaluation metrics, providing a percentage that measures the proportion of correct predictions relative to the total number of observations, where TP denotes True Positives, TN True Negatives, FP False Positives, and FN False Negatives. In addition to Accuracy, other metrics such as Precision (2), Recall (3), and F1-Score (4) was also considered to comprehensively assess the model's performance [30].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1\text{-}Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4}$$

The performances of the tested models, considering all evaluation metrics and the scenarios described in this work, are summarized in Table IV and Table V.

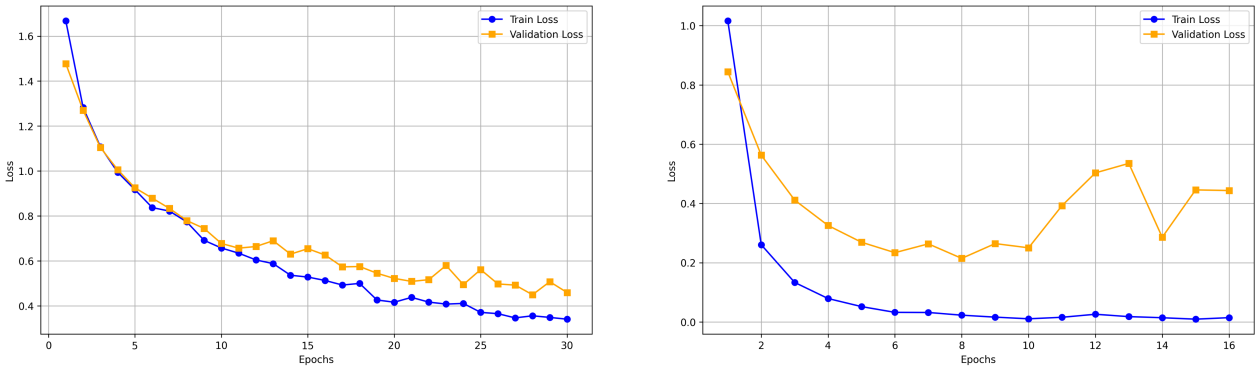
C. Discussions

The models' results presented in the tables provide valuable insights:

1) Both models perform well on the current tasks, with EfficientNet-B0 demonstrating better performance than MobileNetV2 according to the evaluated metrics.
2) Data augmentation had no significant impact on the results, suggesting that dataset imbalance is not a critical factor for either model. Additionally, models trained with RGB inputs outperformed those trained with grayscale images.
3) The performance observed with 4-fold cross-validation demonstrates that both models are not overfitting the data and generalize well across each subset, indicating good robustness.

To provide a detailed analysis of the classification performance for the *Aedes aegypti* and *Aedes albopictus* classes, confusion matrices for the models (Fig. 10 and Fig. 11) were generated.

The confusion matrix highlights several of the previously discussed observations. For instance, *Culex nigripalpus* and *Aedes scapularis* achieved high classification accuracy in both models, as evidenced by the prominent values along the main diagonal, which represent correct predictions for each class. In contrast, a higher number of misclassifications persisted



(a) (b)

Fig. 9. Training and validation loss curves for the two evaluated architectures. (a) MobileNetV2. (b) EfficientNetB0. These plots illustrate the typical learning behavior observed during model training. Source: Author.

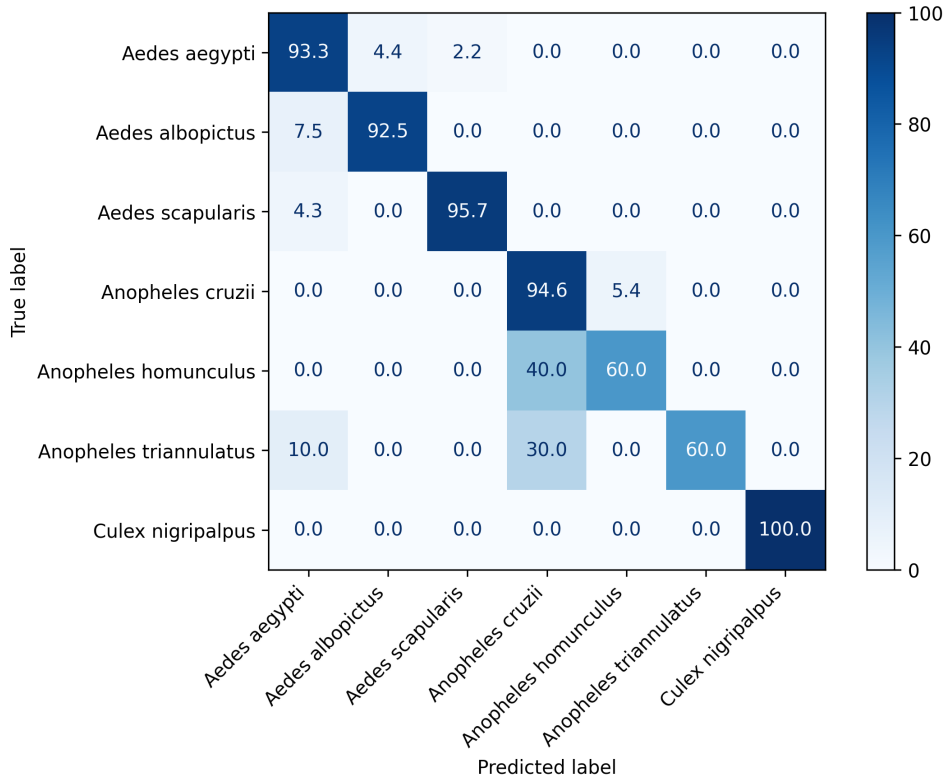


Fig. 10. Confusion matrix generated using the MobileNetV2 model without data augmentation, with RGB inputs. Source: Author.

for *Anopheles cruzii*, *Anopheles homunculus*, and *Anopheles triannulatus*. Regarding *Aedes aegypti* and *Aedes albopictus*, both models were able to clearly distinguish between these two species.

Among the tested scenarios for EfficientNet-B0, the model achieved the highest performance in this experiment. As shown in Table VI, the major classes reached high F1-scores ($\geq$ 0.91), including *Aedes aegypti* (0.94), *Aedes albopictus* (0.91), *Aedes scapularis* (0.94), and *Anopheles cruzii* (0.91). Performance decreased for minority classes with low support, such as *Anopheles homunculus* and *Anopheles triannulatus*, which is expected under data scarcity and could be improved with additional samples. Even with this imbalance, the macro-average F1-score remains high (0.87), indicating that the model is still able to capture minority classes.

A paired t-test was conducted to assess whether data augmentation produced measurable improvements in model performance across the simulated scenarios. Table VII reports the p-values obtained from comparing the F1-scores of each model and input type with and without augmentation. All p-values exceeded the 0.05 threshold, indicating that the differences observed were not statistically significant under the tested conditions.

Misclassifications, such as an image of *Aedes albopictus*

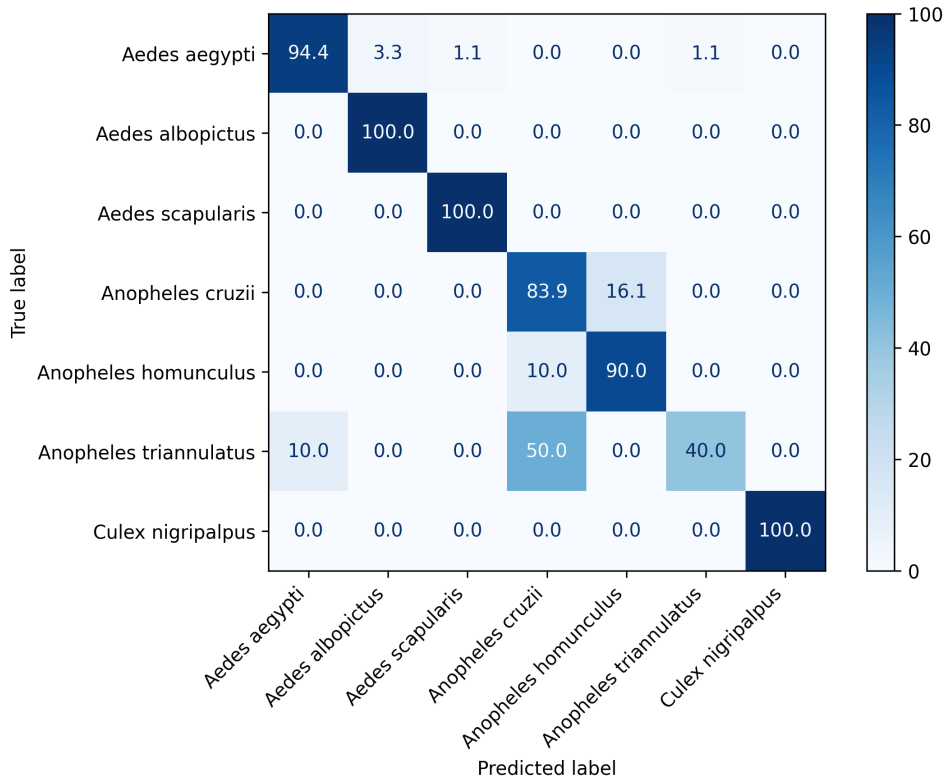


Fig. 11. Confusion matrix generated using the EfficientNet-B0 model without data augmentation, with RGB inputs. Source: Author.

being incorrectly labeled as *Aedes aegypti* (or vice versa), are mainly attributed to the morphological similarities between the two species. Morphological identification involves evaluating multiple characteristics, including color and scale patterns, head and thorax structures, abdomen, and maxillary palps [31], which makes it a complex and time-consuming process.

In contrast, analyzing wing images directly reduces assessment time, minimizes human bias, and allows for large-scale processing. This task is inherently repetitive, which further limits its speed compared to the CNN-based approach proposed in this study. For instance, some studies report a correct identification rate of 81% achieved by human specialists [32], highlighting the advantages of adopting CNNs for morphological classification tasks. Furthermore, according to [13], mounted wing images require minimal preprocessing and are easier to standardize, allowing for simpler CNN architectures and reducing both image variability and dataset size for effective training.

VI. Conclusion

This study classified dengue vector mosquitoes, *Aedes aegypti* and *Aedes albopictus*, using wing images from the WingBank database. EfficientNet-B0 outperformed MobileNetV2, achieving $91.80 \pm 2.27\%$ accuracy with data augmentation, demonstrating superior performance on RGB inputs. The models were successfully tested on a Cortex-A processor, showing that high-accuracy mosquito classification is feasible on embedded devices, supporting scalable vector monitoring systems.

However, some limitations of the proposed approach should be noted. Misclassifications, such as labeling *Aedes aegypti*, the primary dengue vector in urban areas, as *Aedes albopictus*, may lead to underestimating the actual transmission risk, potentially affecting public health alert systems. Moreover, the relatively small dataset limits the representation of morphological diversity, increasing the risk of overfitting and potentially reducing the model's generalization to unseen samples.

Even considering the dataset's limitations, this study demonstrates strong potential for application in public health strategies. The proposed approach can be extended to mosquito species beyond dengue vectors, supporting broader epidemiological surveillance. By enabling continuous monitoring of mosquito populations, the model may provide valuable insights into vector biology and transmission dynamics, thereby informing more effective vector control policies and related preventive actions.

The model also shows promise for practical deployment in real field conditions. Its evaluated architectures have relatively low computational requirements, making them suitable for integration into mobile devices, remote sensing platforms, or automated trapping systems. Such implementations could provide real-time species identification directly in endemic regions, reducing reliance on specialized personnel and accelerating decision-making in vector control operations.

Acknowledgments

The authors would like to acknowledge the support of the Coordenação de Aperfeiçoamento de Pessoal de Nível Supe-

TABLE IV
PERFORMANCE METRICS OF MOBILENETV2 AND EFFICIENTNET-B0 MODELS WITHOUT DATA AUGMENTATION, EVALUATED ON RGB AND GRAYSCALE INPUTS

Scenario	Metrics (%)	Fold 1	Fold 2	Fold 3	Fold 4	Mean ± Std
MobileNetV2						
RGB	Accuracy	90.16	83.61	87.70	88.11	87.39± 2.38
	F1-Score	91.00	86.00	91.00	91.00	89.75± 2.17
	Recall	91.00	86.00	91.00	91.00	89.75± 2.17
	Precision	91.00	86.00	92.00	91.00	90.00± 2.35
Grayscale	Accuracy	84.02	79.10	84.02	86.89	83.51± 2.80
	F1-Score	87.00	83.00	86.00	89.00	86.25± 2.17
	Recall	88.00	85.00	88.00	90.00	87.75± 1.79
	Precision	87.00	86.00	89.00	90.00	88.00± 1.58
EfficientNet-B0						
RGB	Accuracy	90.57	92.62	92.62	90.57	91.59± 1.03
	F1-Score	91.00	95.00	94.00	93.00	93.25± 1.48
	Recall	91.00	95.00	94.00	93.00	93.25± 1.48
	Precision	93.00	95.00	94.00	94.00	94.00± 0.71
Grayscale	Accuracy	89.34	87.30	88.93	90.98	89.14± 1.31
	F1-Score	92.00	89.00	91.00	93.00	91.25± 1.48
	Recall	92.00	89.00	91.00	93.00	94.00± 0.71
	Precision	92.00	90.00	92.00	93.00	91.75± 1.09

TABLE V
PERFORMANCE METRICS OF MOBILENETV2 AND EFFICIENTNET-B0 MODELS WITH DATA AUGMENTATION, EVALUATED ON RGB AND GRAYSCALE INPUTS

Scenario	Metrics (%)	Fold 1	Fold 2	Fold 3	Fold 4	Mean ± Std
MobileNetV2						
RGB	Accuracy	87.70	82.79	87.30	82.79	85.15± 2.36
	F1-Score	92.00	85.00	88.00	88.00	88.25± 2.49
	Recall	92.00	86.00	89.00	89.00	89.00± 2.12
	Precision	92.00	86.00	88.00	89.00	88.75± 2.15
Grayscale	Accuracy	83.61	75.41	76.64	70.49	76.54± 4.69
	F1-Score	86.00	83.00	82.00	78.00	82.25± 2.86
	Recall	88.00	85.00	85.00	80.00	84.50± 2.87
	Precision	89.00	83.00	82.00	78.00	83.00± 3.94
EfficientNet-B0						
RGB	Accuracy	93.44	91.80	88.10	93.85	91.80± 2.27
	F1-Score	94.00	94.00	88.00	95.00	92.75± 2.77
	Recall	94.00	94.00	88.00	95.00	92.75± 2.77
	Precision	94.00	94.00	88.00	96.00	93.00± 3.00
Grayscale	Accuracy	89.75	85.66	85.25	89.75	87.60± 2.15
	F1-Score	91.00	89.00	89.00	90.00	89.75± 0.83
	Recall	91.00	89.00	89.00	90.00	93.00± 3.00
	Precision	91.00	89.00	91.00	89.00	90.00± 1.00

TABLE VI
PER-CLASS CLASSIFICATION METRICS AND MACRO-AVERAGE FOR THE TEST SET

Class	Precision	Recall	F1-score	Support
Aedes aegypti	0.94	0.93	0.94	90
Aedes albopictus	0.90	0.92	0.91	40
Aedes scapularis	0.92	0.96	0.94	23
Anopheles cruzii	0.88	0.95	0.91	56
Anopheles homunculus	0.67	0.60	0.63	10
Anopheles triannulatus	1.00	0.60	0.75	10
Culex nigripalpus	1.00	1.00	1.00	15
Macro avg	0.90	0.85	0.87	244

TABLE VII
PAIRED T-TEST RESULTS FOR F1-SCORES WITH AND WITHOUT DATA AUGMENTATION

Model	Input	p-value
MobileNetV2	RGB	0.22
MobileNetV2	Grayscale	0.21
EfficientNet-B0	RGB	0.82
EfficientNet-B0	Grayscale	0.10

rior – Brazil (CAPES), Finance Code 001, the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), and the Department of Control and Automation Engineering at the Federal University of Ouro Preto (UFOP).

REFERENCES

[1] M. U. G. Kraemer *et al.*, "The global distribution of the arbovirus vectors *Aedes aegypti* and *Ae. albopictus*," *eLife*, vol. 4, 2015, doi: 10.7554/eLife.08347.

[2] D. V. Viana and E. Ignotti, "The occurrence of dengue and meteorological variations in Brazil: A systematic review," *Revista Brasileira de Epidemiologia*, vol. 16, no. 2, pp. 240–256, 2013, doi: 10.1590/S1415-790X2013000200002.

[3] J. F. Day, "Mosquito oviposition behavior and vector control," *Insects*, vol. 7, no. 4, p. 65, 2016, doi: 10.3390/insects7040065.

[4] A. L. S. A. Zara *et al.*, "Strategies for controlling *Aedes aegypti*: a review," *Epidemiologia e Serviços de Saúde*, vol. 25, no. 2, pp. 391–404, 2016, doi: 10.5123/S1679-49742016000200017.

[5] M. A. M. Sallum, R. G. Obando, N. Carrejo, et al., "Identification keys to the *Anopheles* mosquitoes of South America (Diptera: Culicidae). II. Fourth-instar larvae," *Parasites & Vectors*, vol. 13, p. 582, 2020, doi: 10.1186/s13071-020-04299-5.

[6] A. Richter, B. E. Boudinot, F. H. Garcia, J. Billen, E. P. Economo, and R. G. Beutel, "Wonderfully weird: the head anatomy of the armadillo ant, Tatuidris tatusia (Hymenoptera: Formicidae: Agroecomyrmecinae), with evolutionary implications," *Myrmecological News*, vol. 33, pp. 35–75, 2023, doi: 10.25849/myrmecol.news_033:035.

[7] G. A. Lima, "Computer vision system for automatic identification of potential *Aedes aegypti* mosquito breeding sites from drone-acquired images," *RISTI - Iberian Journal of Information Systems and Technologies*, pp. 93–109, 2021, doi: 10.17013/risti.43.93109.

[8] C. Laranjeira et al., "Automatic mapping of high-risk urban areas for *Aedes aegypti* infestation based on building facade image analysis," *medRxiv*, 2023, doi: 10.1101/2023.11.30.23298876.

[9] C. J. de Santana, A. C. A. Firmo, R. F. A. P. de Oliveira, P. J. B. Lins, G. A. de Lima, and R. A. de Lima, "A solution for counting *Aedes aegypti* and *Aedes albopictus* eggs in paddles from ovitraps using deep learning," *IEEE Latin America Transactions*, vol. 17, no. 12, pp. 1987–1994, Dec. 2019, doi: 10.1109/TLA.2019.9011543.

[10] I. Sanchez-Gendriz, G. F. de Souza, I. G. M. de Andrade *et al.*, "Data-driven computational intelligence applied to dengue outbreak forecasting: a case study at the city of Natal, RN-Brazil scale," *Scientific Reports*, vol. 12, p. 6550, 2022, doi: 10.1038/s41598-022-10512-5.

[11] V. R. de Lima, M. C. C. de Morais, and K. Kirchgatter, "Integrating artificial intelligence and wing geometric morphometry to automate mosquito classification," *Acta Tropica*, vol. 249, p. 107089, Jan. 2024, doi: 10.1016/j.actatropica.2023.107089.

[12] J. Park, D. I. Kim, B. Choi, *et al.*, "Classification and Morphological Analysis of Vector Mosquitoes using Deep Convolutional Neural Networks," *Scientific Reports*, vol. 10, p. 1012, Jan. 2020, doi: 10.1038/s41598-020-57875-1.

[13] F. G. Sauer, M. Werny, K. Nolte, *et al.*, "A convolutional neural network to identify mosquito species (Diptera: Culicidae) of the genus Aedes by wing images," *Scientific Reports*, vol. 14, p. 3094, Feb. 2024, doi: 10.1038/s41598-024-53631-x.

[14] I. Torres, M. Nakano, J. A. Cime-Castillo, E. Escamilla-Hernandez, O. Lopez-Garcia, and H. Lanz Mendoza, "Dengue-Infected Mosquito Detection with Uncertainty Evaluation based on Monte Carlo Dropout," *IEEE Latin America Transactions*, vol. 23, no. 2, pp. 135-143, Feb. 2025, doi: 10.1109/TLA.2025.10851361.

[15] N. Manakitsa *et al.*, "A review of machine learning and deep learning for object detection, semantic segmentation, and human action recognition in machine and robotic vision," *Technologies*, vol. 12, no. 2, p. 15, 2024, doi: 10.3390/technologies12020015.

[16] A. Khan, A. Sohail, U. Zahoora, and A. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artificial Intelligence Review*, vol. 53, pp. 5455–5516, Dec. 2020, doi: 10.1007/s10462-020-09825-6.

[17] J. Bobulski and M. Kubanek, "Waste classification system using image processing and convolutional neural networks," in *Advances in Computational Intelligence (IWANN 2019)*, I. Rojas, G. Joya, and A. Catala, Eds. Cham: Springer, 2019, vol. 11507, pp. 319–331, doi: 10.1007/978-3-030-20518-8_30.

[18] Y. Zhu and S. Newsam, "DenseNet for dense flow," in *2017 IEEE International Conference on Image Processing (ICIP)*, Beijing, China, pp. 790–794, 2017, doi: 10.1109/ICIP.2017.8296389.

[19] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.

[20] B. Koonce, "EfficientNet," in *Convolutional Neural Networks with Swift for TensorFlow*, Apress, Berkeley, CA, 2021, doi: 10.1007/978-1-4842-6168-2_10.

[21] K. Dong, C. Zhou, Y. Ruan, and Y. Li, "MobileNetV2 Model for Image Classification," in *2020 2nd International Conference on Information Technology and Computer Application (ITCA)*, Guangzhou, China, pp. 476–480, 2020, doi: 10.1109/ITCA52113.2020.00106.

[22] K. Kansal, T. B. Chandra, and A. Singh, "ResNet-50 vs. EfficientNet-B0: Multi-Centric Classification of Various Lung Abnormalities Using Deep Learning," *Procedia Computer Science*, vol. 235, pp. 70-80, 2024, doi: 10.1016/j.procs.2024.04.007.

[23] S. Hymel, C. Banbury, D. Situnayake, A. Elium, C. Ward, M. Kelcey, *et al.*, "Edge Impulse: An MLOps platform for tiny machine learning," *arXiv preprint arXiv:2212.03332*, 2022, doi: 10.48550/arXiv.2212.03332.

[24] M. De Franco and J. Kalil, "The Butantan Institute: History and future perspectives," *PLoS Neglected Tropical Diseases*, vol. 8, no. 7, p. e2862, 2014, doi: 10.1371/journal.pntd.0002862.

[25] F. Virginio, V. Domingues, L. C. G. da Silva, L. Andrade, K. R. Braghetto, and L. Suesdek, "WingBank: A wing image database of mosquitoes," *Frontiers in Ecology and Evolution*, vol. 9, p. 660941, 2021, doi: 10.3389/fevo.2021.660941.

[26] B. B. Das, S. K. Ram, K. S. Babu *et al.*, "Person identification using autoencoder-CNN approach with multitask-based EEG biometric," *Multimedia Tools and Applications*, vol. 83, pp. 83205–83225, 2024, doi: 10.1007/s11042-024-18693-z.

[27] R. P. da Silva Neto and A. O. de Carvalho Filho, "Automatic classification of breast lesions using transfer learning," *IEEE Latin America Transactions*, vol. 17, no. 12, pp. 1964–1969, Dec. 2019, doi: 10.1109/TLA.2019.9011540.

[28] M. Alkanan and Y. Gulzar, "Enhanced corn seed disease classification: leveraging MobileNetV2 with feature augmentation and transfer learning," *Frontiers in Applied Mathematics and Statistics*, vol. 9, 2024, doi: 10.3389/fams.2023.1320177.

[29] L. Prechelt, "Early Stopping – But When?," in Neural Networks: Tricks of the Trade, G. B. Orr and K. R. Müller, Eds., Berlin, Heidelberg: Springer, 1998, vol. 1524, pp. 55–69, doi: 10.1007/3-540-49430-8_3.

[30] B. J. Bipin Nair, B. Arjun, S. Abhishek, N. M. Abhinav, and V. Madhavan, "Classification of Indian Medicinal Flowers using MobileNetV2," in 2024 11th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2024, pp. 1512–1518, doi: 10.23919/INDIACom61295.2024.10498274.

[31] A. Richter, B. E. Boudinot, F. H. Garcia, J. Billen, E. P. Economo, and R. G. Beutel, "Wonderfully weird: the head anatomy of the armadillo ant, Tatuidris tatusia (Hymenoptera: Formicidae: Agroecomyrmecinae), with evolutionary implications," Myrmecological News, vol. 33, pp. 35–75, 2023, doi: 10.25849/myrmecol.news 033:035.

[32] F. Jourdain, M. Picard, T. Sulesco, *et al.*, "Identification of mosquitoes (Diptera: Culicidae): an external quality assessment of medical entomology laboratories in the MediLabSecure Network," Parasites and Vectors, vol. 11, p. 553, 2018, doi: 10.1186/s13071-018-3127-7.

Lucas Ferreira Quintão Moreira received his B.S. degree in Control and Automation Engineering from the School of Mines, Ouro Preto, Brazil, with an academic exchange at the Faculty of Engineering, University of Porto (FEUP), Portugal. He is currently pursuing the M.Sc. degree in Electronic and Computer Engineering, with emphasis on Systems and Control, at the Aeronautics Institute of Technology (ITA), Brazil. His research interests include Control Systems and Artificial Intelligence, with applications in Bioengineering. During his undergraduate studies, he conducted research involving Control and AI techniques applied to Agriculture (Irrigation) and Bioengineering (Dengue Control).

Agnaldo J. Rocha Reis received his degree in Electrical Engineering from the Pontifical Catholic University of Minas Gerais (1996), a Master's degree in Electrical Engineering from the Federal School of Engineering of Itajubá (1999), a D.Sc. in Electrical Engineering from the Federal University of Itajubá (2003) and a Post-Doctorate in Electrical Engineering from the Federal University of Minas Gerais (2014-2015). Currently, he is an Associate Professor at the Department of Control and Automation Engineering at the Federal University of Ouro Preto (UFOP) and was the Academic Coordinator of the Postgraduate Program in Instrumentation, Control and Automation of Mining Processes - PROFICAM (UFOP/ITV-Vale Agreement) (2019/2021). He was also the Vice-President of the Brazilian Society of Computational Intelligence (2009-2011).

Alan Kardek Rêgo Segundo received his B.Sc. degree in Control and Automation Engineering from the Federal University of Ouro Preto (UFOP), Ouro Preto, Brazil, in 2008, and the M.Sc. and D.Sc. degrees in Agricultural Engineering from the Federal University of Viçosa (UFV), Viçosa, Brazil, in 2010, and 2014, respectively. He is currently a Reseacher with the UFOP and the Vale Technological Institute (ITV), Ouro Preto, Brazil.