AI

MDPI

Article

Integration of YOLOv8 Small and MobileNet V3 Large for Efficient Bird Detection and Classification on Mobile Devices

Axel Frederick Félix-Jiménez [1,†], Vania Stephany Sánchez-Lee [1,†], Héctor Alejandro Acuña-Cid [1,*], Isaul Ibarra-Belmonte [2], Efraín Arredondo-Morales [1] and Eduardo Ahumada-Tello [3]

[1] Instituto Politécnico Nacional, Unidad Profesional Interdisciplinaria de Ingeniería Campus Zacatecas (UPIIZ), Academia de Ciencias de la Computación, Zacatecas 98160, Mexico; afelixj2000@alumno.ipn.mx (A.F.F.-J.); vsanchezl2000@alumno.ipn.mx (V.S.S.-L.); earredondo@ipn.mx (E.A.-M.)
[2] Centro de Investigación en Matemáticas (CIMAT), Unidad Zacatecas, Departamento de Ingeniería de Software, Zacatecas 98160, Mexico; isaul.ibarra@cimat.mx
[3] Facultad de Contaduría y Administración, Universidad Autónoma de Baja California, Tijuana 22424, Mexico; eahumada@uabc.edu.mx
* Correspondence: hacunac@ipn.mx; Tel.: +52-492-176-2903
† These authors contributed equally to this work.

Abstract: Background: Bird species identification and classification are crucial for biodiversity research, conservation initiatives, and ecological monitoring. However, conventional identification techniques used by biologists are time-consuming and susceptible to human error. The integration of deep learning models offers a promising alternative to automate and enhance species recognition processes. Methods: This study explores the use of deep learning for bird species identification in the city of Zacatecas. Specifically, we implement YOLOv8 Small for real-time detection and MobileNet V3 for classification. The models were trained and tested on a dataset comprising five bird species: Vermilion Flycatcher, Pine Flycatcher, Mexican Chickadee, Arizona Woodpecker, and Striped Sparrow. The evaluation metrics included precision, recall, and computational efficiency. Results: The findings demonstrate that both models achieve high accuracy in species identification. YOLOv8 Small excels in real-time detection, making it suitable for dynamic monitoring scenarios, while MobileNet V3 provides a lightweight yet efficient classification solution. These results highlight the potential of artificial intelligence to enhance ornithological research by improving monitoring accuracy and reducing manual identification efforts.

Keywords: YOLOv8 Small; MobileNet V3; TensorFlowLite; bird classification; automated sampling; mobile app; digital color optical imaging; spatial resolution; illuminance; chromaticity

Academic Editor: Arslan Munir

Received: 27 January 2025
Revised: 21 February 2025
Accepted: 6 March 2025
Published: 13 March 2025

Citation: Félix-Jiménez, A.F.; Sánchez-Lee, V.S.; Acuña-Cid, H.A.; Ibarra-Belmonte, I.; Arredondo-Morales, E.; Ahumada-Tello, E. Integration of YOLOv8 Small and MobileNet V3 Large for Efficient Bird Detection and Classification on Mobile Devices. *AI* **2025**, *6*, 57. https://doi.org/10.3390/ai6030057

1. Introduction

Bird monitoring plays a fundamental role in biodiversity research, ecological conservation, and environmental management. Understanding bird populations, migratory patterns, and habitat changes provides valuable insights into ecosystem health and climate change effects. Traditionally, biologists have relied on manual observation techniques and standardized protocols to conduct bird surveys. Currently, researchers in the biology area of the Autonomous University of Zacatecas (UAZ) carry out their bird monitoring following a manual that standardizes the process. However, this approach is very time-consuming, as it involves the installation of research equipment that the same manual specifies as necessary [1]. This procedure can take several hours to complete a single survey, involving setting up research equipment, such as mist nets and acoustic sensors, to collect data on bird species. While these traditional methods are widely used, they present significant

limitations, including long processing times, susceptibility to human error, and the need for extensive field resources. In many cases, accurate bird identification requires expert knowledge, making these processes inaccessible to non-specialists.

In this context, there are technologies that could optimize and simplify this process. In recent years, advances in computer vision and deep learning have introduced new possibilities for automating bird detection and classification. Various deep learning-based approaches leverage convolutional neural networks (CNNs) to recognize and classify birds from images with high precision [2]. Among the most widely used techniques, YOLO (You Only Look Once) models have gained popularity for object detection due to their real-time performance, while MobileNet architectures are frequently utilized for classification tasks, particularly in mobile and resource-limited environments. YOLOv5 has been used in a number of studies for bird detection using transfer learning approaches with models like VGG19, Inception V3, and EfficientNet to accurately categorize bird species [3]. Even though these methods have shown a great deal of success, they are not feasible for field use on mobile devices since they frequently require either huge training datasets, significant computational resources, or cloud-based processing [4].

In consideration of these difficulties, a portable, real-time, and effective system that can properly identify and categorize bird species and be implemented in environments with limited resources is required. By combining YOLOv8 Small for real-time bird identification and MobileNet V3 Large for classification, this study fills these deficiencies and provides a portable yet reliable ornithological research tool. The implementation of YOLOv8 Small, together with its Ultralytics library, is particularly convenient due to its compatibility with mobile devices, allowing for the more efficient identification of birds in images [5]. In addition, the use of the MobileNet V3 neural network complements this system for the classification of detected birds, allowing it to distinguish between different species accurately and quickly [6]. The proposed system is optimized for mobile devices, allowing researchers to conduct bird monitoring more efficiently without requiring external high-performance computing resources. By combining real-time object detection with a specialized classification model, this approach enables accurate species identification directly from smartphone cameras, reducing the dependency on traditional, time-intensive field methods. In order to test the performance of the system, five different species of birds from the Zacatecas region were chosen, including the Vermilion Flycatcher, Pine Flycatcher, Mexican Chickadee, Arizona Woodpecker, and Striped Sparrow, which were chosen taking into consideration their ecological relevance, distinctive visual traits, and regional distribution [7]. The results of this investigation show that MobileNet V3 Large achieved 93% classification accuracy and YOLOv8 Small achieved 89.67% detection accuracy, proving the usefulness and dependability of the system. By using deep learning to automate bird identification, this research advances the creation of scalable, accessible, and real-time solutions for ecological research, biodiversity monitoring, and conservation initiatives.

2. Related Work

The detection and classification of birds have been areas of interest in projects due to their relevance in environmental monitoring [4]. These works employ advanced techniques such as convolutional neural networks (CNNs) and transfer learning, as well as their application in portable devices, such as mobile phones and drones, with the aim of being tested in natural environments.

However, existing studies face challenges such as the need for more representative datasets, precise bird identification, and model optimization for resource-limited devices. Next, in Table 1, a comparison of the approaches employed in recent studies for bird detection and classification is presented.

Table 1. Comparison of related studies in bird detection and classification.

Article	Detection Method	Classification	Advantages	Limitations
Effectiveness of Inception V3 and MobileNet V2 Models in Classifying Bird Species Based on Physical Characteristics [3]	Not applied	Inception V3, MobileNet V2	MobileNet V2 achieves higher accuracy (94.93%) compared to Inception V3 (91.16%)	Limited to classification tasks; detection methods not explored
Bird Detection and Species Classification: Using YOLOv5 and Deep Transfer Learning Models [4]	YOLOv5	VGG19, Inception V3, EfficientNetB3	Robust detection; accurate classification of species	Requires large datasets and high computational cost
Acoustic detection of regionally rare bird species through deep convolutional neural networks [8]	DCNNs adapted	Not applied	Acoustic generalization through data augmentation	Does not address images; focuses on audio only
DeepVision: Enhanced Drone Detection and Recognition in Visible Imagery through Deep Learning Networks [9]	YOLO, Faster R-CNN	Not applied	Efficient use of UAVs for habitat monitoring	Difficulties with detection of objects smaller than 40×40 pixels and in motion
Automating Bird Detection Based on Webcam Captured Images using Deep Learning [10]	Faster R-CNN, SSD	Not applied	Learning transfer with pretrained sets	Does not classify species; limited to webcams
Bird monitoring using the smartphone (iOS) application Videography for motion detection [11]	Movement with smartphones	Not applied	Use of accessible hardware	Limited to close detection
Advanced Computer Vision Methods for Tracking Wild Birds from Drone Footage [12]	YOLOv7, YOLOv8	Not applied	Optimization for small objects with drones	High computational complexity
Analysis of DenseNet-MobileNet-CNN Models on Image Classification using Bird Species Data [13]	Not applied	DenseNet, MobileNet, traditional CNN	MobileNet outperforms DenseNet and traditional CNN in accuracy and real-time performance	Focuses solely on classification; detection not addressed

According to the analysis of the information in Table 1, it is observed that the majority of the projects carried out focus on species' identification but not on their classification. Of the six projects analyzed, the one titled "Bird Detection and Species Classification: Using YOLOv5 and Deep Transfer Learning Models" stands out, as it implements both an identification model and a classification model, using technologies such as YOLOv5, Inception V3, and EfficientNetB3 [4]. Focusing on their advantages and limitations, it is concluded that these implementations offer robust detection and precise species classification. However, to achieve the high levels of accuracy reported, it is necessary to have an extensive dataset, in addition to facing high costs associated with training neural network models.

Regarding the projects analyzed in Table 1 that use YOLO technology, most are limited to the "identification" of objects of interest without addressing their "classification". Moreover, these implementations are often integrated into devices such as drones, UAVs (unmanned aerial vehicles), and web cameras. Even object identification models like YOLOv5 are widely used in object detection across different fields, from manufacturing to species identification in nature [14]. Additionally, in investigations where high accuracy and efficiency are required in classifying classes, some of the most used deep learning models are Inception V3 and MobileNet, both of which have been proven to be effective in image recognition tasks. On the other hand, the article "Bird Monitoring Using the Smartphone (iOS) Application Videography for Motion Detection" is related to our approach of using neural networks on mobile devices for bird identification. This article describes the use of an application called "Videography", which employs an algorithm designed to detect pixel changes in the camera's field of view [11].

Alternative Technologies

In recent years, transformer-based models have revolutionized the field of computer vision, introducing an alternative approach to traditional convolution-based architectures. Unlike conventional methods, transformers use self-attention mechanisms, allowing them to capture global relationships within an image without relying exclusively on local operations. This has led to the development of optimized architectures for computer vision, such as the Swin Transformer, MobileViT, and EfficientFormer, each designed to improve feature representation in detection and classification tasks. Table 2 presents a comparison of some of the main transformer models applied to computer vision, highlighting their advantages, limitations, and areas of application.

Table 2. Comparison of transformer-based object detection models.

Model	Type	Advantages	Limitations	Application Area
Swin Transformer [15]	Vision transformer (ViT)	Strong feature representation; scalable	Computationally expensive for mobile devices	Object detection; medical imaging
MobileViT [16]	Lightweight transformer	Optimized for mobile; better spatial efficiency	Less accurate than CNNs for some tasks	Real-time object detection on mobiles
EfficientFormer [17]	Hybrid transformer–CNN	Balances accuracy and efficiency	Still relatively new; less field-tested	Mobile AI; real-time applications

The Swin Transformer is a computer vision architecture that introduces a hierarchical approach using sliding windows to compute attention, allowing it to capture relationships at different scales and reduce computational complexity [15]. MobileViT combines the advantages of convolutional networks and transformers, designing a parameter-efficient architecture that integrates MobileNet blocks with self-attention mechanisms, achieving a balance between precision and efficiency [16]. EfficientFormer is a vision transformer that achieves speeds comparable to MobileNet, maintaining high performance in image classification tasks, thanks to an optimized design that reduces computational complexity [17].

These architectures have demonstrated outstanding performance in object recognition, image segmentation, and scene classification applications, especially in domains where a detailed analysis of the image at a global level is required. Despite the advances in transformer-based vision models, their application in mobile devices and in real-time still presents significant challenges in terms of computational efficiency and inference speed [18]. Transformers require complex self-attention calculations, which involve higher memory and processing consumption, making them less ideal for implementation on mobile phones with limited

resources. Therefore, for the purpose of this study, it was determined that we would work with convolutional neural networks (CNNs) such as MobileNet and YOLO [19].

3. Materials and Methods

The objective of this project is to meet the need of researchers in the field of biology to automate the process of bird sampling. To achieve this, the aim was to enable a mid-high-range Android mobile device to identify and classify bird species. The target species for classification included the Vermilion Flycatcher, Pine Flycatcher, Mexican Chickadee, Arizona Woodpecker, and Striped Sparrow.

To select the most suitable model or neural network for bird identification, a study was conducted that concluded that YOLOv8 was the most optimal option for its implementation on mobile devices. According to the article titled "A Review on YOLOv8 and Its Advancement", the compact and efficient design of YOLOv8 facilitates its adaptation to mid-range or low-end hardware platforms [20]. Likewise, this research highlights that scaled versions, such as YOLOv8s Small, are specifically designed to operate in resource-limited environments, such as smartphones and embedded devices.

For this reason, it was decided to implement YOLOv8 Small as the model for bird identification. The process involves using YOLOv8 Small to detect a bird in an image, cropping the area where the bird is located, and sending it to the MobileNet V3 classification model. This workflow, represented in Figure 1, aims to optimize the performance of the classification model by providing a cropped image solely of the area containing the bird.

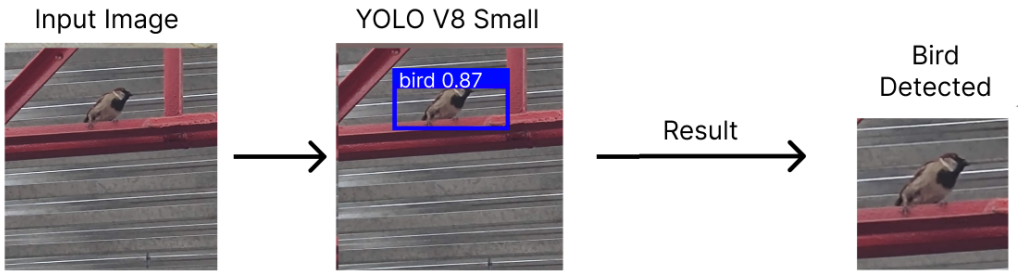

Figure 1. Graphical description of bird identification process using YOLOv8 Small.

In this way, the need to analyze external elements in the image is reduced, allowing the model to focus exclusively on the bird, which speeds up and simplifies recognition.

A preliminary investigation was conducted with the aim of selecting the most suitable classification model for this project, focused on identifying five specific bird species. During the process, three neural network models were evaluated: ResNet34, Inception V3, and MobileNet V3. In this study, their performance was evaluated using key metrics such as accuracy, validation loss, and accuracy on the validation set to compare the performance of each model classifying bird images, in addition to analyzing their computational requirements to determine the feasibility of their implementation in different environments.

According to the results in Table 3, the Inception V3 model achieved the highest accuracy with a 99.87% accuracy and a validation loss of 0.0013, making it the most effective for the classification task. However, this model has high computational requirements, making it more suitable for environments with powerful GPUs and high-capacity servers. On the other hand, MobileNet V3 achieved an accuracy of 94.46% and a validation loss of 0.1936, standing out for its efficiency and low resource consumption, making it the best option for implementations in mobile devices and embedded systems. These results lead us to the conclusion that the best model to adopt depends on the situation: MobileNet V3 is the most effective option for mobile apps and low computational cost, while Inception V3 is best for maximum precision in high-performance settings.

Table 3. Comparison of neural network models in terms of validation loss, accuracy on validation set, and final accuracy.

Model	Validation Loss	Accuracy on Validation Set	Final Accuracy
Inception V3	0.0013	99.87%	99.87%
ResNet34	0.8271	85.67%	85.67%
MobileNet V3	0.1936	94.46%	94.46%

However, this study highlighted that the design of MobileNet V3 is specially optimized for implementation on mobile devices, making it an ideal candidate for this project. The implementation of the model allowed, from a photograph or image captured with a mobile phone or similar device, the classification of the bird into one of the following species, as illustrated in Figure 2: Vermilion Flycatcher, Pine Flycatcher, Mexican Chickadee, Arizona Woodpecker, and Striped Sparrow.

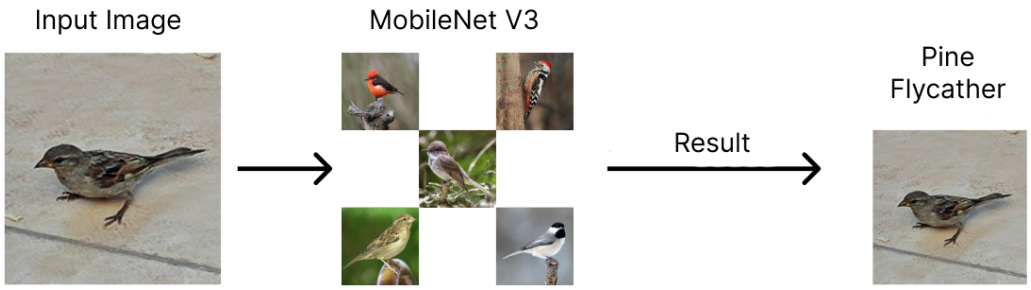

Figure 2. Graphical description of bird grading process using MobileNet V3.

The following details the process followed to implement the technologies in this project, including the methods used for data preparation, model selection, training strategies, and result evaluation, as well as the challenges encountered and the strategies employed to overcome them.

3.1. Overview of YOLOv8 Small and MobileNet V3

3.1.1. YOLOv8 Small

YOLOv8 (You Only Look Once version 8) is a deep learning model specializing in object detection, image classification, and instance segmentation tasks. Developed by Ultralytics, it represents a significant evolution from previous YOLO versions, incorporating architectural optimizations that enhance accuracy and reduce inference times. Its design is particularly appealing for implementations on devices with limited computational resources, such as embedded systems and mid-to-high-end mobile devices [21].

The YOLOv8 architecture is organized into three main components: the backbone, the neck, and the head. The backbone functions as a feature extractor using an optimized convolutional network, leveraging modern techniques like the SiLU activation function to ensure the capture of relevant patterns in images. The neck employs an FPN-PANet module that combines features extracted at various levels, which is crucial for multi-scale object detection. Finally, the head performs the final predictions, generating the bounding box coordinates and classifying the detected objects (Figure 3). This design incorporates an anchor-free approach, reducing complexity and improving the model's generalization capabilities [22].

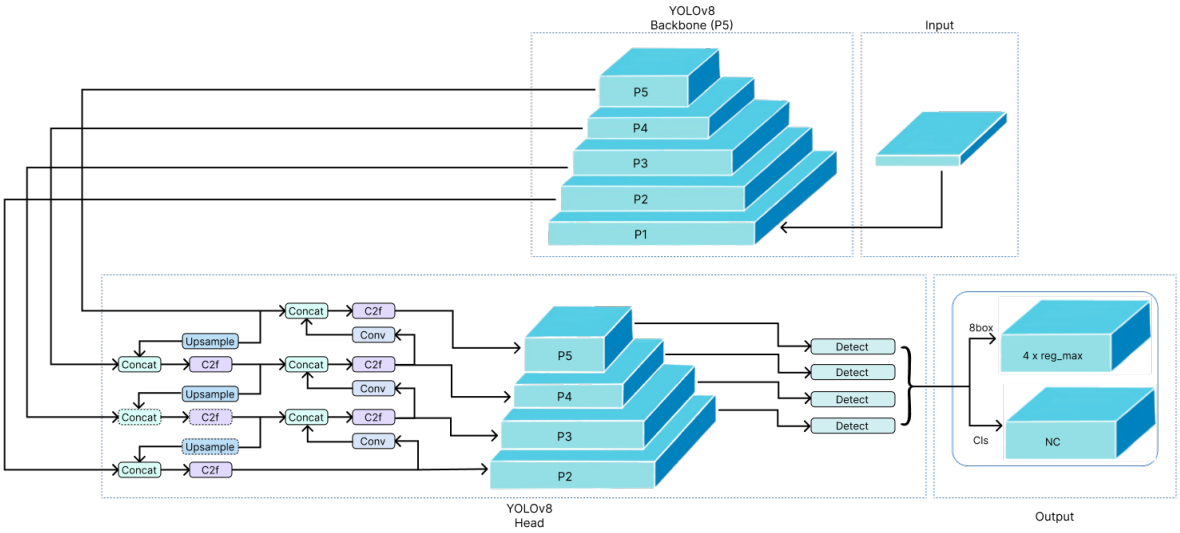

Figure 3. YOLOv8 architecture.

The Small version of YOLOv8 is compact and designed specifically for applications with limited computational resources, such as mid-to-high-end smartphones. It strikes a balance between accuracy and efficiency, ensuring seamless integration into less powerful hardware without sacrificing performance [23].

Due to its compact and efficient design, YOLOv8 Small proves to be an excellent choice for devices with constrained computational resources. Its capability to achieve a balance between accuracy and performance renders it highly suitable for a wide range of applications, effectively meeting the demands of mobile and portable systems.

3.1.2. MobileNet V3 Large

MobileNet V3 is a convolutional neural network architecture optimized to maximize efficiency on devices with limited computational resources, such as mobile phones and embedded systems. This architecture combines advanced network architecture search techniques with innovative design improvements, achieving an outstanding balance between precision and computational performance [24].

One of the key aspects of MobileNet V3 is the use of depthwise separable convolutions, a technique that significantly reduces the number of parameters and computational operations compared to traditional convolutions. Additionally, the architecture integrates inverted residual blocks alongside "squeeze-and-excitation" modules Figure 4. These modules are essential for optimizing the information flow within the network, emphasizing the most relevant features in processed images [25].

Another notable element is the "Hard-Swish" activation function, an efficient variant of Swish, specifically designed to enhance performance on mobile hardware. This design not only maximizes computational efficiency but also improves the network's ability to handle complex tasks without significantly increasing computational costs Figure 4. Furthermore, the model was fine-tuned through a combination of advanced techniques, such as the NetAdapt algorithm and hardware-aware network search, to optimize its performance on mobile device CPUs, achieving an ideal balance between precision and latency [25].

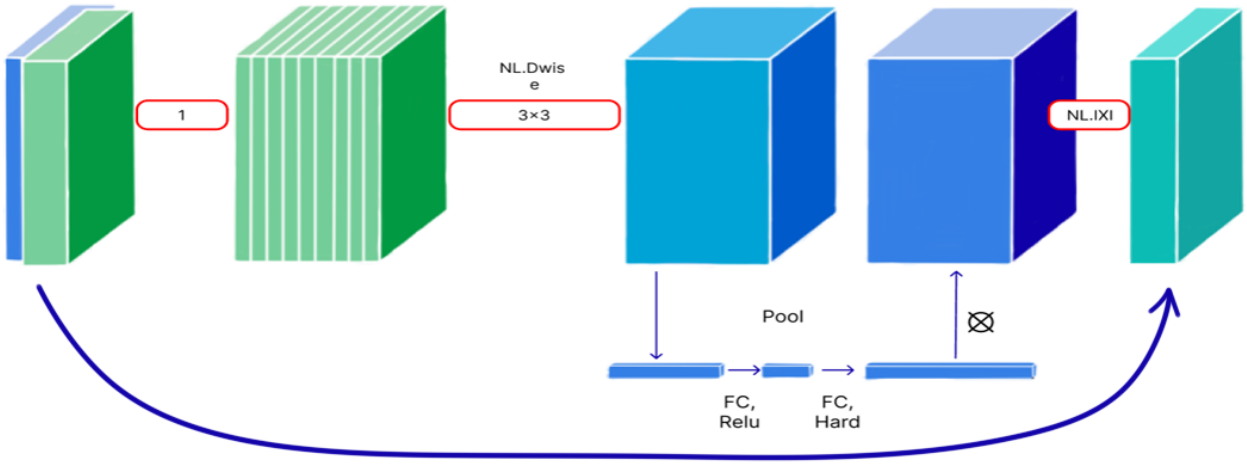

Figure 4. MobileNet V3 architecture.

In the context of this project, the MobileNet V3 Large variant was selected due to its ability to handle high-resolution images and perform complex classifications with high precision. This variant is specifically designed to operate on resource-limited hardware, ensuring an optimal balance between processing speed and energy efficiency, which are essential characteristics for mobile devices [26].

MobileNet V3 Large also stands out for its compatibility with TensorFlow Lite, facilitating its integration into embedded devices and smartphones. This compatibility enables the model to be implemented efficiently, maintaining robust performance and a real-time user experience. By incorporating MobileNet V3 Large into the proposed system, an accurate classification of bird species is ensured, meeting the portability and efficiency objectives set for the project [27].

3.2. Preliminary Experiments with MobileNet

Before conducting the definitive training sessions with MobileNet V3, a series of exploratory experiments were performed exclusively on this model to evaluate its capability to classify bird species under conditions that simulated the variability of photographs captured with mobile devices. These preliminary experiments focused solely on the classification model, as the identification model already had a pre-existing and well-prepared dataset for bird detection. However, no publicly available dataset contained sufficient samples of the specific bird species targeted in this study for classification purposes. Therefore, it was necessary to conduct these preliminary experiments to assess the model's ability to generalize effectively and optimize its training process. The primary objective of these experiments was to determine whether MobileNet V3 could achieve an accuracy threshold of 85%, which is considered comparable to human-level performance in object classification tasks [28].

For these exploratory trainings, a dataset comprising 5000 images was utilized, evenly distributed across five species (1000 images per species). Approximately half of these images contained noise or visual interference, while the other half consisted of high-quality, noise-free photographs. This setup was designed to emulate real-world conditions, where image quality captured by mobile phones can vary significantly.

The results of these exploratory experiments are illustrated in Figure 5. In the upper chart, the precision and loss values obtained during the training sessions are shown. Across the seven attempts, the model exhibited variability in its performance, failing to meet the desired precision threshold. The lower chart presents the results of the validation dataset, revealing further challenges in the model's generalization capability.

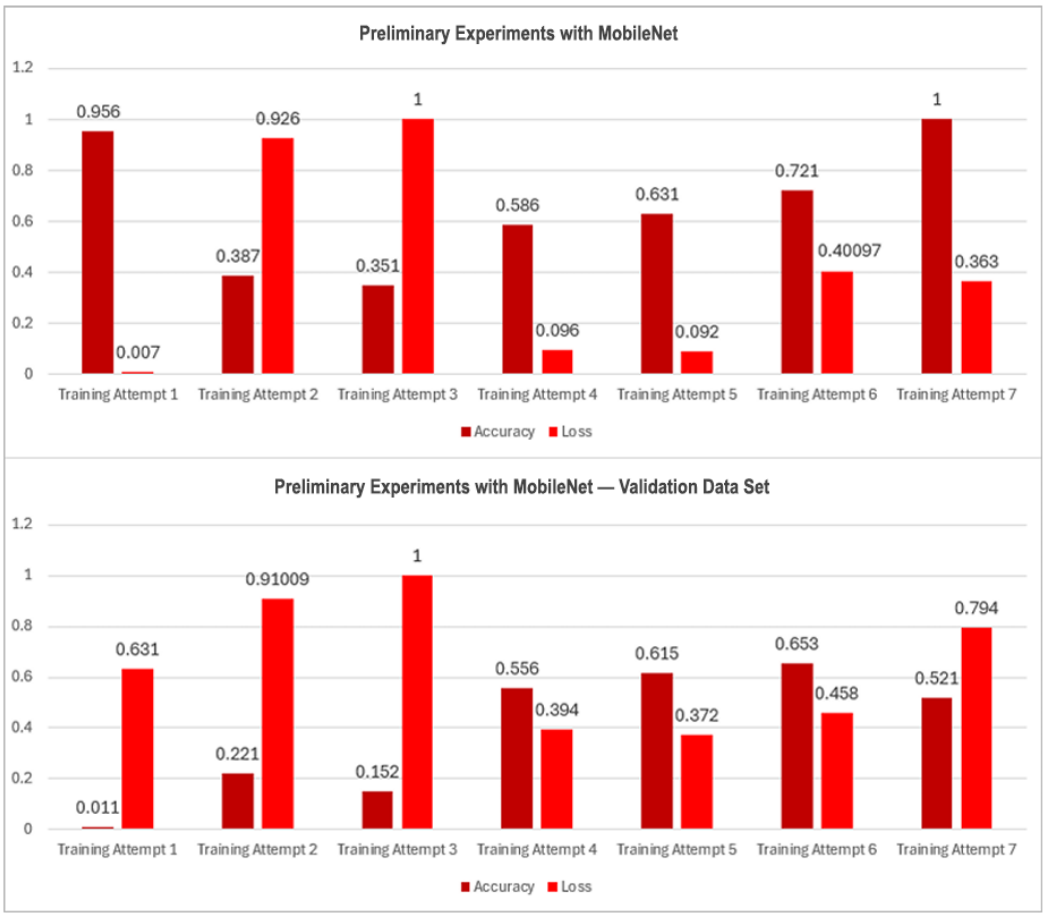

Figure 5. Results of preliminary experiments with MobileNet. The upper chart shows precision and loss on the training dataset, while the lower chart displays results on the validation dataset.

To further analyze the impact of different training configurations and assess their effect on model performance, we tested various combinations of dropout regularization, transfer learning, data augmentation, and layer unfreeze strategies Table 4 presents the configuration settings for each of the seven training attempts.

Table 4. Training configurations applied in preliminary experiments with MobileNet.

Training Attempt	Dropout	Transfer Learning	Data Augmentation	Layer Unfreeze (False)	Layer Unfreeze (Last 20 Layers)	Layer Unfreeze (True)
1	✓	✓	✗	✓	✗	✗
2	✗	✓	✓	✓	✗	✗
3	✓	✓	✓	✗	✓	✗
4	✗	✓	✗	✓	✗	✗
5	✓	✓	✓	✗	✓	✗
6	✓	✓	✓	✗	✗	✓
7	✗	✓	✓	✗	✗	✓

The results from these training sessions provided valuable insights into the effects of each configuration on model performance. The observations included the following:

- **High training accuracy but poor generalization.** Several attempts, such as Training Attempt 1 (accuracy: 0.956) and Training Attempt 7 (accuracy: 1.00), showed exceptionally high accuracy on the training dataset. However, these results did not translate to the validation dataset, where accuracy dropped dramatically (0.011 and 0.521, respectively), indicating severe overfitting.

- **Limited impact of data augmentation.** Training Attempts 3, 5, 6, and 7 incorporated data augmentation, but the results did not show significant improvement in validation accuracy. For instance, Training Attempt 3, which included all enhancement techniques, yielded a validation accuracy of only 0.151, confirming that augmentation alone was insufficient to address model limitations.
- **Layer unfreezing did not enhance performance.** Fully unfreezing layers in Training Attempts 6 and 7 led to increased instability, with high training accuracy (0.721 and 1.00) but suboptimal validation results (0.653 and 0.521). Even partially unfreezing layers (Training Attempts 3 and 5) failed to produce a breakthrough in generalization, suggesting that fine-tuning the model did not help in the presence of dataset noise.
- **Dropout regularization had minimal effect.** While dropout was expected to mitigate overfitting, the results indicate that its presence did not consistently contribute to improved validation accuracy. Training Attempt 2, which lacked dropout, still performed similarly to attempts that incorporated it, with a validation accuracy of 0.221.

These findings suggest that, despite variations in hyperparameters and techniques, none of the tested configurations led to a meaningful improvement in generalization. The model struggled to adapt to the variability of mobile-captured images, reinforcing the need for more robust dataset refinement and architectural adjustments before proceeding with definitive training.

3.3. Dataset Preparation

3.3.1. Dataset for YOLOv8 Small

For the training of the YOLOv8 neural network, the dataset titled "Bird Detection Computer Vision Project" [29], available on the Roboflow platform, was chosen. This dataset was selected due to its wide variety of images representing birds in different environments, resulting in a more robust detection model.

The percentages of division for training (84%), validation (11%), and testing (6%) were maintained according to the original data set configuration. This division follows the recommendations of the deep learning literature, where it is suggested to allocate more than 80% of the data for training to ensure an adequate generalization capacity of the model [30]. No modifications were made to this division to avoid introducing biases and to take advantage of the optimal distribution designed by the dataset authors.

The dataset contains a total of 14,162 bird images, distributed across the training, validation, and test folders. Table 5 shows the details of this division.

Table 5. YOLO dataset splitting. Distribution of images for training, validation and testing.

Dataset Split	Percentage	Number of Images
Training Set	84%	11,850
Validation Set	11%	1521
Testing Set	6%	791

3.3.2. Dataset for MobileNet V3

The use of MobileNet V3 was for the classification of birds identified through an image taken by a cell phone camera or uploaded from it. As already mentioned, there were 5 birds to be classified (Vermilion Flycatcher, Pine Flycatcher, Mexican Chickadee, Arizona Woodpecker, and Striped Sparrow), as shown in Figure 6

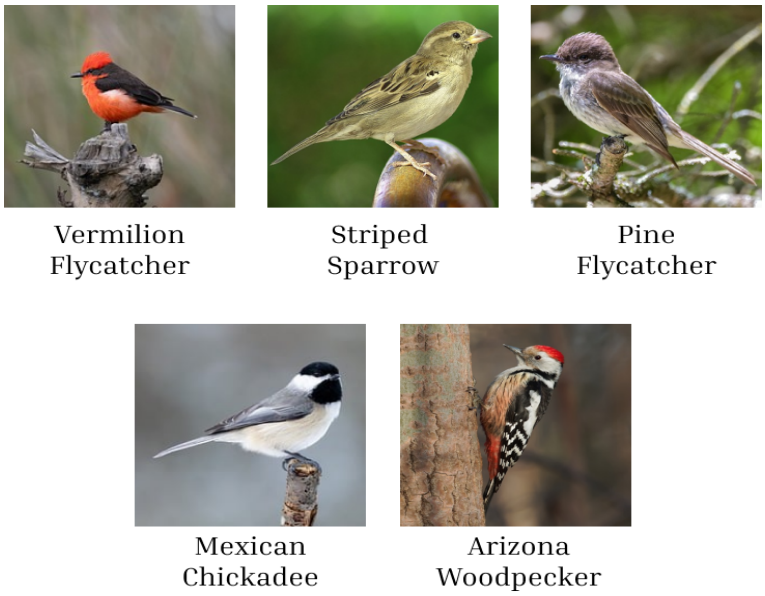

Figure 6. Birds to be classified. Examples of the five target species.

In order to train the model, it was necessary to obtain a collection of images for every class of bird. These images were obtained from public-access pages that provide photography and publications of bird photographs by enthusiasts and professionals, such as iNaturalist or eBird. The process of creating the dataset followed an iterative approach to improve its quality and representativeness. The three generated versions are described below:

- **Basic Dataset:** Composed of 2500 images per class (12,500 in total). This initial dataset lacked advanced preprocessing techniques, resulting in low diversity and limitations in the model's generalization capacity.
- **Extended Dataset:** Increased to 5000 images per class (25,000 in total), including *data augmentation* such as rotations, cropping, and brightness adjustments. Although this improved diversity, overfitting was observed in certain classes.
- **Balanced and Clean Dataset:** Consolidated to 3000 images per class (15,000 in total), selecting the highest-quality images and eliminating redundant ones. This adjustment balanced efficiency and representativeness, optimizing the model's performance.

The progression of the number of images in each iteration is illustrated in Figure 7.

Several picture preprocessing methods were used on the dataset to increase the model's generalization and classification accuracy. These techniques were designed to simulate real-world conditions and improve the robustness of the model against common variations in image capture, such as occlusions, lighting changes, and different perspectives. To improve the dataset's diversity and the model's capacity to accurately categorize bird species, the following preprocessing techniques were used:

- **Images with noise:** contain interferences or external visual patterns that can hinder identification, simulating real conditions.
- **Rotated images:** modified through angular rotations to increase diversity and improve the model's robustness against changes in orientation.
- **Zoomed images:** scaled to simulate approaches and improve the model's ability to identify objects at different distances.
- **Background-free images:** processed to remove the background environment and focus solely on the main object, which is, in this case, the bird.

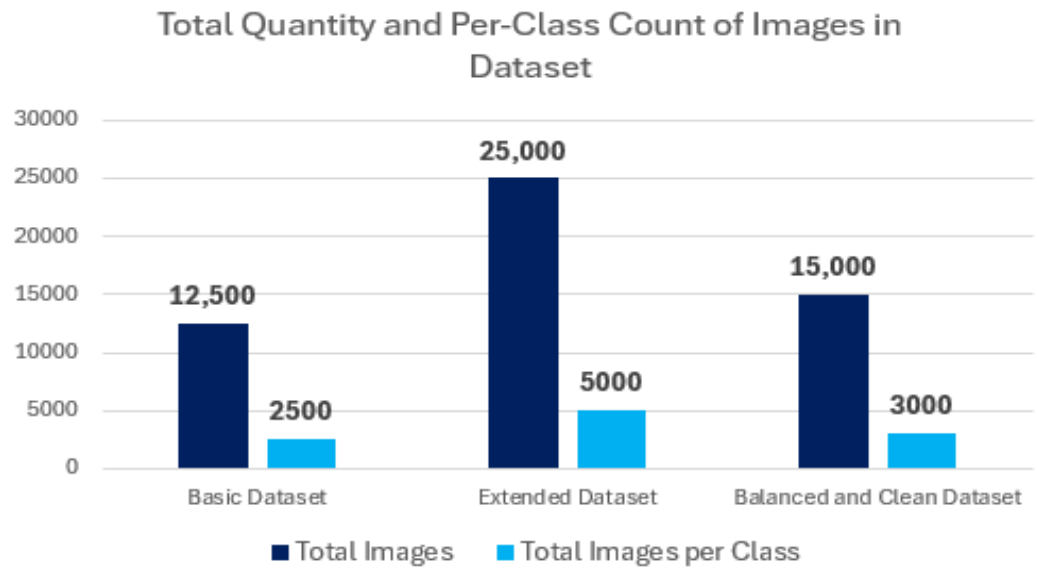

Figure 7. Number of images. Progression of number of images per class in final dataset.

The evolution of the dataset allowed for a progressive improvement in data quality, starting with basic images in the *Basic Dataset* and adding techniques such as rotations, cropping, and zoom in the *Extended Dataset*. Finally, the *Balanced and Clean Dataset* prioritized images without backgrounds and of higher quality, consolidating a robust set for model training. These improvements are summarized in Table 6.

Table 6. A comparison of characteristics in the datasets. Progressive improvements in the quality of the data used.

Characteristics	Basic Dataset	Extended Dataset	Balanced and Clean Dataset
Noisy images	✓	✓	✓
Rotated images		✓	
Zoomed images		✓	
Backgroundless images		✓	✓
Higher-quality images			✓

The table shows the progressive features applied to the datasets used in the training.

3.4. YOLOv8 Small Training

The configuration used for the model was as follows:

- **Model used:** YOLOv8 Small (`yolov8s.pt`), which balances precision and speed, ideal for mobile devices [30].
- **Image size:** 640 pixels, for a balance between resolution and computational efficiency.
- **Batch size:** 16 images, suitable for the hardware used.

As already mentioned, for the training of YOLOv8 Small, the dataset titled "Bird Detection Computer Vision Project" was exclusively used, as this dataset was selected for its quality and representativeness, specifically designed for bird detection tasks in various environments. No additional modifications were made to the dataset, such as (*data augmentation*) techniques or specific preprocessing, as the original dataset included an adequate variety of scenarios and configurations. This allowed for a direct focus on optimizing the YOLOv8 Small model using the dataset's default parameters.

Throughout the project, different numbers of training epochs were experimented with to optimize the model's accuracy and efficiency. In an initial phase, to minimize the

consumption of computational resources, the *Early Stopping* technique was used, configured with a limit of 10 epochs. This parameter stopped the training if the model did not show improvements in its performance after 10 consecutive epochs, thus saving time and resources. During this stage, the training ended at epoch 67, achieving an accuracy of 88.48%.

Subsequently, a second experiment was conducted with the aim of exploring whether the model could achieve greater accuracy. To achieve this, *Early Stopping* was disabled and a limit of 100 training epochs was set. The results obtained were slightly favorable, achieving an accuracy of 89.68%. However, when analyzing the impact of additional training, it was observed that the model did not show significant improvement after the first 67 epochs. This suggests that, although prolonged training allowed for a slight improvement in metrics such as *accuracy* and *mAP@50-95*, the model had already reached optimal performance in earlier stages.

As presented in Table 7, the model trained for 100 epochs outperformed the one trained for 67 epochs in key metrics, such as *accuracy*, which reached a value of 89.68%, and *mAP@50-95*, with 60.59%. These metrics confirm that the selected model met this project's objectives by offering solid and efficient performance in bird identification.

Table 7. Comparison of metrics between YOLOv8 Small (100 and 67 epochs). Key metrics for two YOLOv8 Small configurations.

Metrics	100 Epochs	67 Epochs
Fitness	0.6372	0.6331
Accuracy (B)	0.8968	0.8848
Recall (B)	0.8591	0.8625
mAP@50 (B)	0.9185	0.9169
mAP@50-95 (B)	0.6059	0.6015

The table shows a comparison between the key metrics obtained in two configurations of YOLOv8 Small (100 and 67 epochs).

Therefore, the YOLOv8 Small model trained with 100 epochs was selected as the final configuration due to its balance between accuracy and efficiency. This choice predisposed a robust performance for bird identification in controlled environments, meeting the objectives set in this project.

3.5. MobileNet V3 Training

To evaluate the performance of MobileNet V3, three experiments were conducted, each using a different version of the dataset (Basic Dataset, Expanded Dataset, Balanced and Clean Dataset). Each experiment allowed testing different configurations and adjusting both the quantity and quality of the data to optimize the model.

The three trainings were designed to explore how the characteristics of the dataset and the training configurations influenced the model's performance. The following describes the key techniques and configurations used in the experiments:

- **Learning Transfer:** A technique in deep learning that uses a pretrained model on one task as a starting point for another related task. For example, a model trained on millions of images can be fine-tuned on a smaller new dataset, leveraging the patterns already learned.
- **Data Augmentation:** This involves increasing the quantity and diversity of training data by applying transformations to the existing data, such as rotations, cropping, brightness adjustments, or adding noise. This improves the model's generalization by simulating variations that could be encountered in the real world.

- **Dropout:** Regularization technique in neural networks that involves randomly deactivating some neurons during training. This prevents overfitting and improves the model's generalization capacity.
- **Layer Defrosting:** This is the process of unfreezing the layers of a previously trained model (frozen to keep their weights fixed) to allow them to be adjusted during training on a new task. This is common in transfer learning to adapt pretrained models to specific data.

Based on Table 8, which shows the different metrics implemented during the model training. Three distinct trainings were carried out, each using a different dataset. The Basic Dataset was used for the first training, the Extended Dataset for the second, and the Balanced and Clean Dataset for the third. Also, to obtain better results from the final models, different configurations were used for every training for every dataset.

Table 8. Comparison of configurations in trainings. Techniques used in three MobileNet V3 trainings.

Characteristics	Training 1	Training 2	Training 3
Learning transfer	✓	✓	✓
Data augmentation	✓	✓	✓
Dropout		✓	✓
Layer defrosting			✓

The table shows the configurations and techniques applied in the training sessions conducted with MobileNet V3.

It is important to mention that in the three training sessions conducted, each one had a different number of epochs. As shown in Figure 8, Training 1 involved 28 epochs, Training 2 had 54 epochs, and Training 3 was completed in 26 epochs. These varying numbers of epochs reflected the experimental nature of the training sessions, where the goal was to evaluate the model's performance in different stages. This strategy helped determine the most effective training configuration based on the number of epochs and its corresponding impact on the model's accuracy.

Figure 8. Number of epochs. Progression of number of training epochs.

Each version of the dataset (Basic Dataset, Expanded Dataset, Balanced and Clean Dataset) corresponded to a different training. As mentioned earlier, the third dataset was the one that achieved the best results along with its configuration. With just 26 training epochs, an accuracy of 100% and a loss of 0.0058 were achieved. Additionally, outstanding results were achieved on the validation set, with an accuracy of 92.86% and a loss of 0.3595.

In contrast, the training conducted with the first and second datasets yielded regular results in the training set but exhibited overfitting issues, as the results in the validation set were significantly inferior, as shown in Table 9.

Table 9. Training results. Accuracy and loss in three trainings of MobileNet V3.

Training	Accuracy	Loss	Validation—Accuracy	Validation—Loss
Training 1	0.98	0.012	0.0012321	2.0108
Training 2	0.65	0.73	0.005461	0.00324323
Training 3	1.00	0.0058	0.9286	0.3595

The table shows the results obtained in the training sessions conducted with MobileNet V3.

These training sessions revealed that the best configuration for the classification of the five birds was the one in which the model layers were unfrozen. The unfrozen layers allowed the model to specifically adapt to the dataset it was trained on, as by default, MobileNet V3 used the pretrained weights from ImageNet as a starting point [31]. Additionally, the model was able to better understand the traits of each bird species and recognize them from photographs in a real-world setting by learning from higher-quality images and backgroundless images. Training 3 stood out from the rest, producing the model with the best performance. For this reason, this model was selected as the most suitable to be implemented in this project.

3.6. Integrated System

The integrated system combines the capabilities of YOLOv8 Small and MobileNet V3 Large to efficiently identify and classify birds on mobile devices. This system follows an optimized workflow, as described in Figure 9, where each component fulfills a specific role to ensure accurate results.

The process begins with the user, who can upload or take a photograph directly from the mobile application. This image is sent to the YOLOv8 Small model, which performs bird detection in the image. If the model identifies a bird, it generates a bounded region in the image that contains the detected object. This step allows the system to crop only the area of interest, removing irrelevant elements in the image, so that the classification model analyzes only the part of interest in the image and not irrelevant elements.

The cropped region is then sent to the MobileNet V3 Large model, specializing in bird classification. This model evaluates the region of interest and classifies the bird into one of the five previously defined species: Vermilion Flycatcher, Pine Flycatcher, Mexican Chickadee, Arizona Woodpecker, and Striped Sparrow.

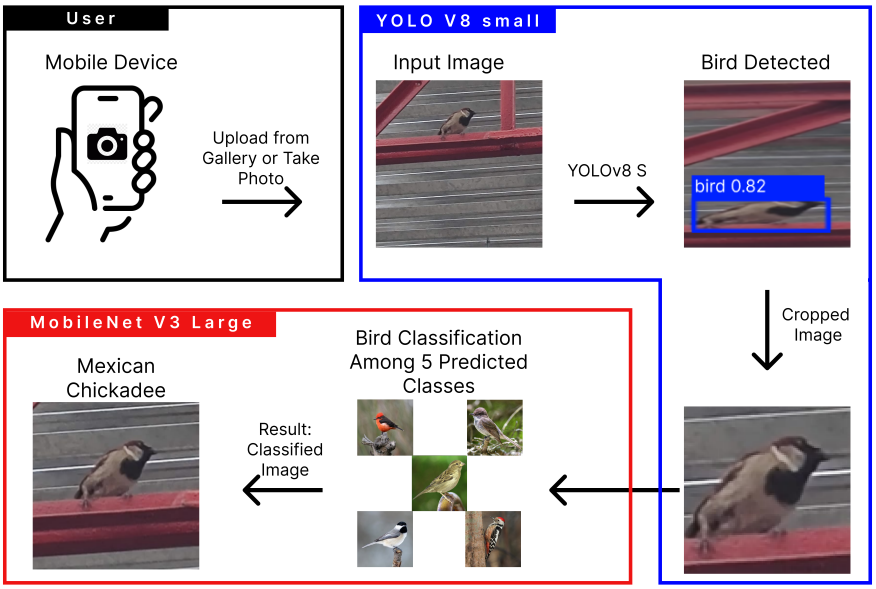

Figure 9. Workflow. Integrated process from capture and identification of bird to its classification.

4. Results

4.1. Results Obtained from MobileNet V3 Large

As mentioned earlier, the configuration and dataset used in Training 3 were selected, as it achieved the best results. The results obtained during this training are presented in Table 10.

Table 10. Final results: MobileNet V3 Large. Accuracy and loss after 26 training epochs.

Epochs	Accuracy	Loss	Accuracy_val	Loss_val
26	1.0000	0.0058	0.9286	0.3595

The table presents the final accuracy and loss results obtained with MobileNet V3 Large after 26 training epochs.

As can be observed, the results obtained demonstrate outstanding performance, with an accuracy of 100% with a minimal loss of only 0.5 in the training set. These findings highlight the efficiency and robustness of the model during the training phase, showcasing its exceptional ability to correctly classify all training data with virtually no errors. Furthermore, the metrics obtained in the validation set reinforce the reliability of the model, with an accuracy of 92% accompanied by a loss of 35. This behavior indicates that the model generalized effectively to unseen data, with no evidence of overfitting or significant degradation in performance when applied to data outside the training set.

Additionally, to rigorously validate the model's performance and assess its applicability in broader scenarios, a confusion matrix was generated using an entirely **new dataset**, distinct from those used in the training and validation phases. This additional dataset comprised a total of 15,000 images, evenly distributed across 3000 images per class. The inclusion of this dataset, as illustrated in Figure 10, aimed to evaluate the model's ability to maintain robust performance when exposed to a wider variety of data and real-world scenarios. Furthermore, since the training and validation datasets were balanced across species, the model's classification accuracy was not biased toward more frequently observed birds in real-world conditions. This balance ensured consistent model performance, regardless of natural variations in species occurrence, reinforcing the validity of the results and providing a comprehensive evaluation of the model's consistency, efficacy, and reliability.

Of the five classes found in the confusion matrix, we find the following results:

1. **Mexican Chikadee:** correct: 2338; errors: 662.
2. **Arizona Woodpecker:** correct: 2895; errors: 105.
3. **Vermilion Flycatcher:** correct: 2939; errors: 83.
4. **Pine Flycatcher:** correct: 2992; errors: 8.
5. **Striped Sparrow:** correct: 2865; errors: 135.

Most of the classes performed outstandingly when classifying the new dataset. However, specific patterns were observed in the model's confusions:

The **Mexican Chickadee** had the highest number of errors (662), frequently being confused with the **Striped Sparrow** and the **Pine Flycatcher**. These confusions were mainly due to similarities in size and plumage coloration patterns. In particular, the Mexican Chickadee and the Striped Sparrow share small and compact bodies, with predominantly brown or dark gray plumage on their wings and back. Additionally, the Striped Sparrow has muted tones that, under certain lighting conditions, can resemble the darker areas of the Mexican Chickadee. On the other hand, the Pine Flycatcher has similar body proportions and light tones on the chest that, in certain images, can be confused with the tones of the Mexican Chickadee, especially in situations where the lighting does not highlight the differences in head coloration Figure 11.

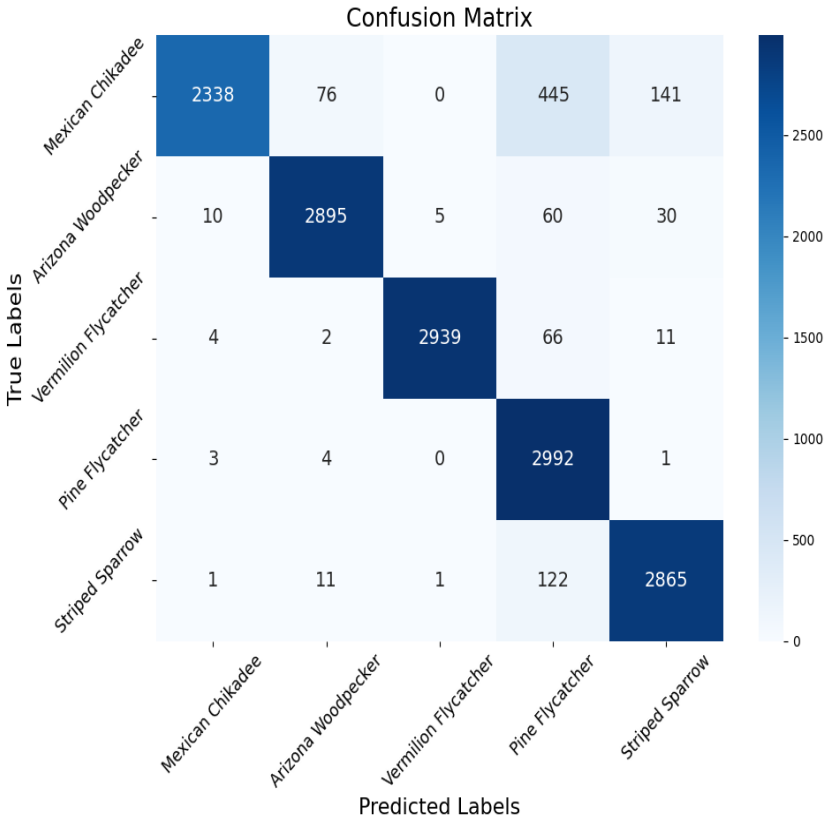

Figure 10. Confusion matrix. Evaluation of the model in five classes of birds.

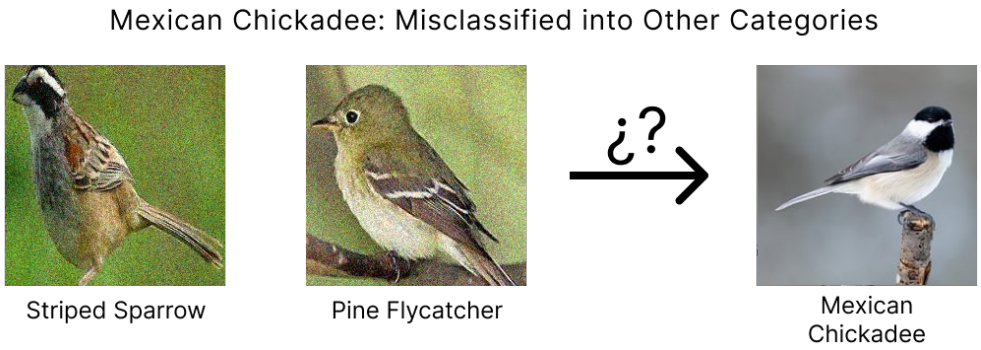

Figure 11. Misclassification of Mexican Chickadee class.

The **Vermillion Flycatcher** was occasionally confused with the **Pine Flycatcher** and the **Stripped Sparrow**. These confusions were mainly due to the fact that all these species share small and compact bodies, with plumages that include light and brown tones in various parts of the body. The Vermillion Flycatcher, which is distinguished by reddish tones in specific parts of its plumage, may not be correctly identified if these distinctive areas are not well visible or if the image has low resolution. Likewise, the Striped Sparrow and the Vermillion Flycatcher have light tones on their chest and lower parts, which can make it difficult to distinguish them in images with uniform lighting (Figure 12).

Figure 12. Misclassification of Vermilion Flycatcher class.

The **Pine Flycatcher** showed a very high performance, with only eight errors, but presented confusions with the **Arizona Woodpecker** and the **Stripped Sparrow**. These confusions were partly due to the similarities in body proportions and the brown or beige tones that both share. The Arizona Woodpecker, for example, has dorsal plumage that can resemble that of the Pine Flycatcher at angles where its characteristic red crest is not visible. Likewise, the Stripped Sparrow and the Pine Flycatcher have similar sizes and dark tones in parts of their plumage, which can be a factor contributing to classification errors (Figure 13).

Figure 13. Misclassification of Pine Flycatcher class.

The **Arizona Woodpecker** was mainly confused with the **Pine Flycatcher** and the **Striped Sparrow**. These confusions may have been due to the fact that, at certain angles, the distinctive features of the Woodpecker, such as its red crest and long beak, are not visible, making it appear similar to the Flycatcher, which has brown or beige plumage on its back. On the other hand, the Striped Sparrow and the Arizona Woodpecker share compact bodies and similar proportions, with plumages that can appear dark in certain areas depending on the lighting conditions. These similarities, combined with variations in lighting or capture angles, can explain the confusions observed in Figure 14.

Figure 14. Misclassification of Arizona Woodpecker class.

The **Striped Sparrow** showed minor confusions with the **Pine Flycatcher** and the **Arizona Woodpecker**. These confusions were related to the similar color patterns in the dorsal plumage. Both the Striped Sparrow and the Pine Flycatcher have light or brown plumage on their undersides and small sizes. Moreover, the similar body proportions of these species could lead the model to misclassify them in certain cases. Regarding the Arizona Woodpecker, the Striped Sparrow, and this one, they share brownish tones in the dorsal plumage that, in the absence of clear details like the long beak of the Woodpecker, could confuse the model (Figure 15).

Figure 15. Misclassification of Striped Sparrow class.

In addition, based on the confusion matrix generated with the new dataset, the primary metrics used to evaluate the model's performance, including precision, recall, the F1-score, and support, were calculated. These metrics provide a comprehensive assessment of the model's classification capabilities across the five evaluated classes. Overall, the model achieved a global accuracy of 93% when classifying the 15,000 images, highlighting its ability to generalize effectively to new and diverse data.

The results reveal consistent performance across most metrics, with precision and recall values averaging 0.93 across all classes, as shown in Table 11. The F1-score, which balances precision and recall, also demonstrated robust values, underscoring the model's effectiveness in maintaining a reliable classification across all categories. Furthermore, the support values indicate that the model was evaluated on a balanced dataset, which strengthens the validity of the results and ensures that the metrics were not biased by imbalanced class distributions.

Table 11. Results of confusion matrix. Precision and recall of MobileNet V3 in five classes.

Class	Accuracy	Recall	F1-Score	Support
Mexican Chickadee	0.99	0.78	0.87	3000
Arizona Woodpecker	0.97	0.96	0.97	3000
Vermilion Flycatcher	1.00	0.97	0.99	3022
Pine Flycatcher	0.81	1.00	0.90	3000
Striped Sparrow	0.94	0.95	0.95	3000
Precision		0.93 (Accuracy)		
Macro-Average	0.94	0.93	0.93	15,022
Weighted Average	0.94	0.93	0.93	15,022

The table shows the key metrics obtained from the confusion matrix for the five evaluated classes.

While the aggregated metrics, such as the macro- and weighted averages, confirm the model's robustness, there are opportunities for further optimization in certain classes. These opportunities primarily arise in cases where slight discrepancies between precision and recall are observed, suggesting potential improvements in the fine-tuning of feature extraction or class-specific thresholds. Nevertheless, the overall performance met-

rics indicate a high degree of reliability and effectiveness in addressing the multiclass classification problem.

The detailed results presented in Table 11 highlight the utility of the model as a robust tool for real-world classification tasks. The combination of high accuracy, balanced precision, and recall across all classes demonstrates its potential for deployment in diverse and complex scenarios.

4.2. Results Obtained from YOLOv8 Small

The dataset used to train the YOLOv8 Small neural network included a validation folder specifically designed to evaluate the model's performance. The metrics obtained from these tests reflected an 89.67% precision, indicating a high level of accuracy in identifying birds in the validation images. Likewise, a recall of 85.91% was achieved, demonstrating the model's excellent ability to capture the majority of relevant instances. Moreover, the mAP@50 (91.85%) and mAP@50-95 (60.59%) metrics confirmed that the model maintained solid performance, even under stricter evaluation thresholds.

As we observed in Table 12, YOLOv8 Small showed good performance when identifying birds in the validation dataset, thus fulfilling its designated task.

Table 12. Results of YOLOv8 Small model. Key metrics for bird identification.

Parameter	Value
Fitness	0.637
Accuracy	0.897
Recall	0.859
mAP@50	0.919
mAP@50-95	0.606
Speed (seconds)	
Preprocessing	0.133
Inference	0.839
Postprocessing	0.828

The table presents the key metrics and processing times of the YOLOv8 Small model when identifying birds.

4.3. Handling Edge Cases

During real-world testing, various challenging scenarios were analyzed to evaluate the model's performance, including partially occluded birds and extreme capture angles. These situations are common in fieldwork, where birds may be partially covered by vegetation or photographed from non-optimal perspectives.

Regarding partial occlusions, the YOLOv8 Small model exhibited reliable performance when at least 50% of the bird's body remained visible. However, as the obstruction exceeded this threshold, the confidence in detections dropped significantly, increasing the false negative rate. The model's ability to correctly identify birds strongly depended on the visibility of key anatomical features, such as plumage patterns, beak shape, and body proportions. When these distinguishing characteristics were blocked, the likelihood of a misclassification increased.

In terms of extreme angles, no significant classification issues were observed, as the MobileNet V3 model was trained with images captured from multiple perspectives. However, a decline in precision was noted when the photograph was taken from below, as natural lighting tends to darken the bird's underside, making it difficult to identify distinguishing features. As observed in Figure 16, both models were tested in different environments.

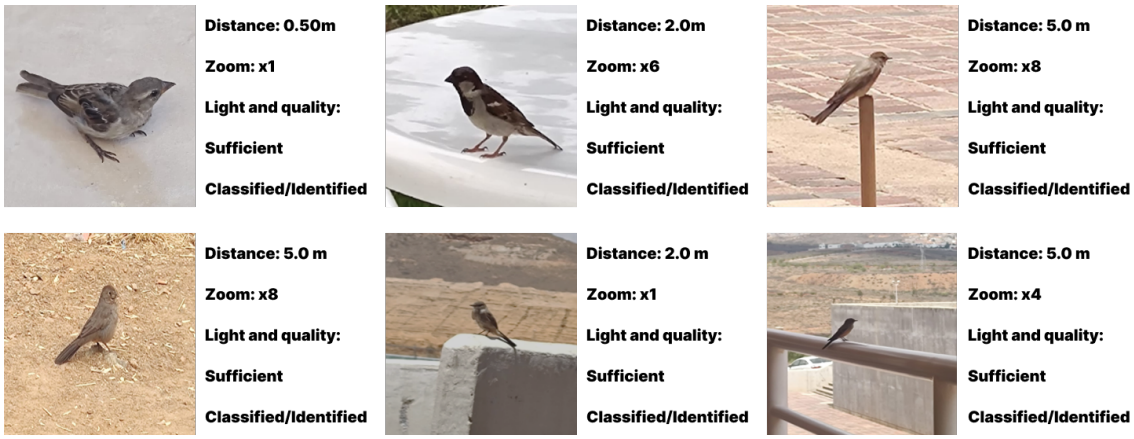

Figure 16. Model performance evaluation under challenging scenarios

Additionally, an experiment was conducted to determine the optimal range for capturing images using a mid-range smartphone camera. The results are summarized as follows (Figure 17):

- **Minimum detection distance:** 0.13 m. Below this threshold, the camera struggled to focus properly.
- **Recommended detection distance:** 3 m. The model maintained high accuracy regardless of zoom level.
- **Maximum detection distance:** 5 m. Achievable with 8× digital zoom. Beyond this distance, classification accuracy declined significantly due to image resolution limitations.

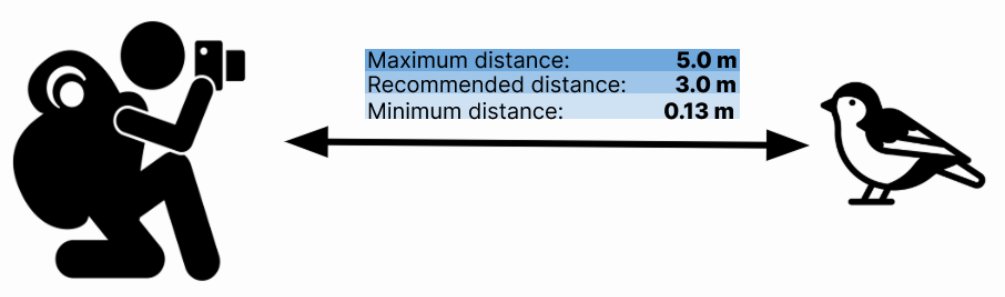

Figure 17. Optimal detection distances for image capture using a mid-range smartphone camera.

These findings highlight the importance of providing users with guidance on optimal image capture conditions to maximize detection accuracy. Future improvements could include pose estimation techniques or occlusion-based training to enhance model robustness in challenging scenarios.

4.4. Test in Real Environments

4.4.1. General Evaluation of the System

The results obtained with YOLOv8 Small and MobileNet V3 in controlled environments were subsequently validated in real-world tests through a movil application. According to these experiments, the integration of both technologies enables a balance between efficiency and precision, accommodating the computing constraints of mobile devices.

On one hand, YOLOv8 Small, with an accuracy of 89.67%, proved to be highly effective in detecting the presence of birds in various environmental conditions.The model maintained a strong identification rate, even with a little fall in recognition confidence, when the bird was partially obscured or had a difficult posture.

On the other hand, MobileNet V3, with an accuracy of 93%, managed to correctly classify most species when the prior detection by YOLOv8 Small was accurate. The model

generally maintained a high generalization capability in photographs taken by users in the field, although it did exhibit some small issues with visually similar species, such as the Striped Sparrow and the Mexican Chickadee.

By combining YOLOv8 Small with MobileNet V3, the system was able to optimize processing on low-resource smartphones by removing irrelevant photos prior to categorization. This integrated strategy prevents the classifier from analyzing photos with irrelevant objects or without birds, hence reducing the amount of incorrect predictions compared to other systems that solely employ classification models.

As well, TensorFlow Lite's implementation allowed quick inference; on mid-range mobile devices, processing durations for each image were less than one second. These findings demonstrate that the suggested method works well for classifying and identifying birds in the field without the requirement for specialized tools.

4.4.2. Comparison with Traditional Methods

Bird monitoring based on the traditional methods used by biologists from the Autonomous University of Zacatecas (UAZ) requires a considerable amount of time depending on the technique employed. Methods such as fixed-radius point counts can take between 15 to 20 min per point (0.25 to 0.33 h), while linear transects can extend up to 2 h per route. In more advanced methods, such as mist netting, the duration of monitoring ranges from 4 to 6 h, due to the need for installation and handling of the birds. Additionally, the analysis of acoustic recordings can extend over several days (Figure 18).

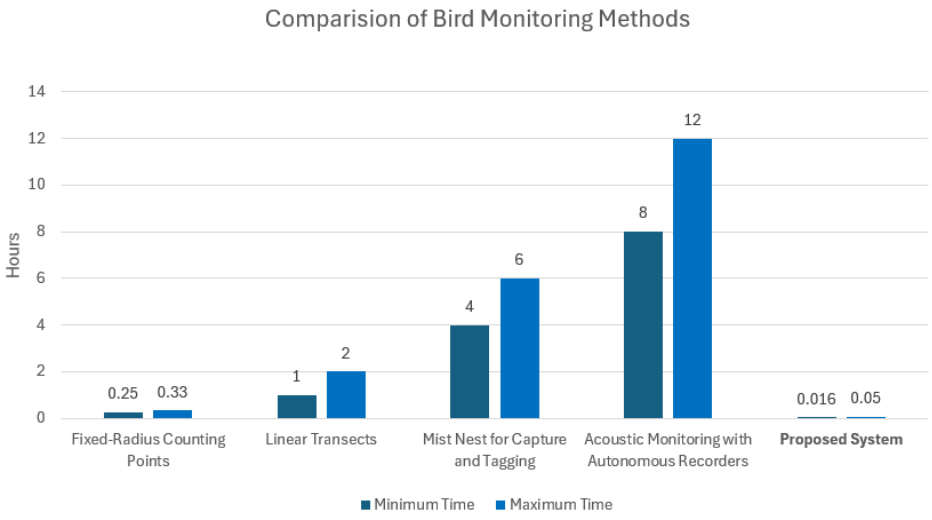

Figure 18. Comparison of the time required for different birdwatching methods.

In contrast, the system proposed in this study allows for the identification and classification of birds in a matter of seconds or minutes using a mobile device, taking from 1 to 3 min (0.016 to 0.05 h) to identify and classify the image obtained. The integration of YOLOv8 Small for detection and MobileNet V3 Large for classification significantly reduces the time required for identification, facilitating fieldwork and providing a portable and easily accessible tool for researchers. This optimization not only improves the efficiency of data collection but also allows for the increased frequency and coverage of monitoring, benefiting biodiversity and conservation studies.

4.4.3. Testing in Real Conditions

As a final step, the models were prepared and optimized for use on mobile devices using TensorFlow Lite. This tool is responsible for optimizing the models by reducing their

file size, eliminating unnecessary weights and operations. In this way, the model becomes lighter and more efficient, making it easier to implement it in mobile environments with limited resources [32].

The following section presents the tests conducted in real environments, demonstrating the functionality of the models and their integration into the application "IdBird". The application is designed to classify birds using advanced technologies like YOLOv8 Small and MobileNet V3 Large, both of which were converted to TensorFlow Lite for mobile compatibility.

The conducted tests revealed several key functionalities of the application. As illustrated in Figure 19a, when the classification model (MobileNet) processed an image of a bird that did not belong to the five predefined classes (e.g., a pigeon), it correctly generated an on-screen notification stating "Bird Not Found!". Additionally, in these cases, the input field for the bird's colloquial name in the application interface remained empty, accurately reflecting the absence of a valid classification.

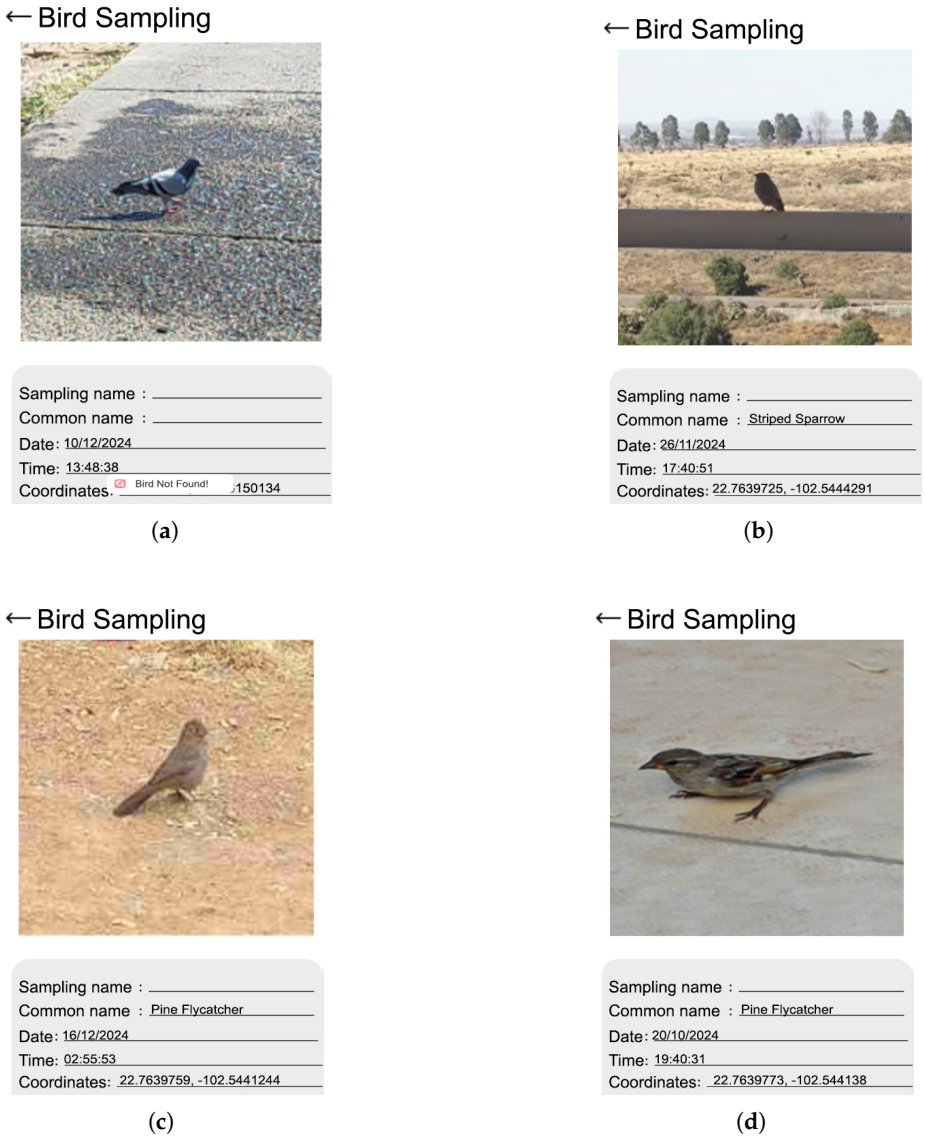

Figure 19. Tests in real environments. Examples of classification with the "IdBird" application. (**a**) Unidentified bird, (**b**) Striped Sparrow, (**c**) Pine Flycatcher, (**d**) Pine Flycatcher.

Conversely, when the model successfully identified a bird from one of the predefined classes, the application displayed the corresponding information. Figure 19b–d demon-

strate the accurate identification of species such as the "Pine Flycatcher" and "Striped Sparrow". In these instances, the application's interface automatically populated the colloquial name field with the detected bird's name, confirming the seamless integration of the model's output with the application's functionality.

These results underscore the effectiveness of the implemented models, which integrate YOLOv8 Small for object detection and MobileNet V3 Large for classification. Despite the computational limitations of mobile devices, the models achieved accurate and reliable performance in real-world scenarios, demonstrating their suitability for resource-constrained environments.

5. Discussion

When comparing this work with the projects reviewed in the state of the art, significant differences are identified that highlight the contributions of this proposal. For example, some works, such as those using YOLOv5 along with classification models like VGG19 and Inception V3, offer robust approaches for detection and classification. However, these solutions often rely on large datasets and require high computational resources, which makes their implementation on mobile devices difficult.

On the other hand, projects that employ YOLO or Faster R-CNN in drones or stationary cameras are effective for detection in large habitats, but they do not include classification models, which limits their scope to object identification without differentiating species. This system, by integrating YOLOv8 Small for detection and MobileNet V3 Large for classification, simultaneously addresses both problems, successfully identifying birds and classifying them into five specific species. This combination makes the solution more comprehensive and practical for environmental research.

A key aspect is the incorporation of YOLOv8 Small, a lighter and optimized version for mobile devices compared to YOLOv5 or YOLOv7 [33], which allows for a workflow more adaptable to devices with limited resources. Moreover, the integration of a classification model strengthens the system, expanding its capabilities compared to solutions that only implement detection.

The Extensibility of the System

This study focused on the identification and classification of five bird species specific to the Zacatecas region due to their ecological importance and significant presence in previous biodiversity studies. This selection enabled an efficient implementation of YOLOv8 Small and MobileNet V3 Large models in mobile devices, ensuring high accuracy in environments with limited computational resources.

While the system was optimized for these five species, its architecture is flexible enough to be extended to a broader range of species. The generalization capability of MobileNet V3 Large, in particular, can be improved through fine-tuning techniques on more diverse datasets, increasing the model's representativeness and enabling its application in biodiversity studies in other regions.

To extend the system, the following strategies could be applied:

- Incorporating more species into the classification dataset while ensuring class balance to prevent biases in prediction.
- Leveraging transfer learning with pretrained models and using architectures previously trained on extensive datasets such as iNaturalist or NABirds.
- Implementing data augmentation techniques to simulate variations in lighting, capture angles, and occlusions that may appear in field images.
- Optimizing more advanced models such as EfficientNet or vision transformers (ViTs) and evaluating their feasibility on mobile devices.

From the outset, the system design incorporated transfer learning by implementing MobileNet V3 Large with pretrained weights from ImageNet. This approach accelerated model convergence and improved its generalization capability from a relatively small initial dataset. As detailed in the training section, alternative models such as ResNet34 and Inception V3 were evaluated, but MobileNet V3 was selected due to its balance between accuracy and efficiency on mobile devices.

Additionally, the YOLOv8 Small model benefited from the use of pretrained weights on general object detection datasets, facilitating adaptation to bird detection without requiring training from scratch. The combination of these approaches ensures that the system can adapt to new species with minimal adjustments.

Although this study focused on five species to validate the system's effectiveness in a controlled context, its modular design enables scalability for broader biodiversity monitoring. Through fine-tuning on more diverse datasets and the application of transfer learning with more robust models, the system could be expanded to address more complex ecological scenarios and encompass a broader range of species in conservation studies.

6. Conclusions

This study demonstrates the feasibility and efficiency of integrating YOLOv8 Small and MobileNet V3 Large for the identification and classification of birds on mobile devices. Unlike traditional ornithological monitoring methods, which require prolonged installation and analysis times, the developed system allows for near-real-time identification with high precision, optimizing fieldwork for researchers and biologists. The obtained results highlight the robustness of the model in different environments and lighting conditions, achieving an accuracy of 89.67% in detection and 93% in classification. The combination of both architectures not only improves processing efficiency on devices with limited resources but also allows for precise identification without the need to rely on specialized infrastructure. Moreover, optimization with TensorFlow Lite allows it to run on mid-range mobile devices; this portability drastically reduces sampling and processing times, facilitating the identification of species in an agile, accessible, and efficient manner for researchers and conservationists. The presented solution transforms ornithological monitoring and biodiversity conservation by providing an innovative, precise, and easy-to-implement tool. Its combination of artificial intelligence, portability, and efficiency makes it a technology with high growth potential and application in large-scale studies, contributing to the development of autonomous and accessible monitoring systems for the scientific community.

Author Contributions: Conceptualization, A.F.F.-J., V.S.S.-L. and I.I.-B.; methodology, I.I.-B.; software, A.F.F.-J. and V.S.S.-L.; validation, A.F.F.-J. and V.S.S.-L.; formal analysis, A.F.F.-J. and V.S.S.-L.; investigation, A.F.F.-J. and V.S.S.-L.; resources, H.A.A.-C.; data curation, A.F.F.-J. and V.S.S.-L.; writing—original draft preparation, A.F.F.-J. and V.S.S.-L.; writing—review and editing, H.A.A.-C., I.I.-B. and E.A.-M.; visualization, A.F.F.-J. and V.S.S.-L.; supervision, H.A.A.-C., I.I.-B. and E.A.-M.; project administration, H.A.A.-C. and E.A.-T.; funding acquisition, H.A.A.-C., E.A.-M. and E.A.-T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available upon request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Pérez-Valadez, N. Adiciones a la avifauna del estado de Zacatecas. *Huitzil* **2016**, *17*, 175–183. [CrossRef]
2. Rust, N.; Jannuzi, B. Identifying Objects and Remembering Images: Insights From Deep Neural Networks. *Curr. Dir. Psychol. Sci.* **2022**, *31*, 316–323. [CrossRef]
3. Riyadi, S.; Salsabila, A.S.; Dewi, A.R.P. Effectiveness of Inception V3 and MobileNet V2 Models in Classifying Bird Species Based on Physical Characteristics. In Proceedings of the 2024 IEEE 14th Symposium on Computer Applications & Industrial Electronics (ISCAIE), Penang, Malaysia, 24–25 May 2024; pp. 265–269. [CrossRef]
4. Vo, H.-T.; Thien, N.; Mui, K. Bird Detection and Species Classification: Using YOLOv5 and Deep Transfer Learning Models. *Int. J. Adv. Comput. Sci. Appl.* **2023**, *14*, 939–947. . [CrossRef]
5. Ma, S.; Lu, H.; Liu, J.; Zhu, Y.; Sang, P. LAYN: Lightweight Multi-Scale Attention YOLOv8 Network for Small Object Detection. *IEEE Access* **2024**, *12*, 29294–29307. [CrossRef]
6. Lee, V.; Jiménez, A.; Belmonte, I.; González, E. Image Recognition System for Bird Sampling in the City of Zacatecas. In Proceedings of the 2023 12th International Conference on Software Process Improvement (CIMPS), Cuernavaca, Mexico, 18–20 October 2023; pp. 126–135. [CrossRef]
7. Dixit, N.; Sharma, A.; Sharma, S.K.; Kumar, R. Influence Analysis of Image Feature Selection Techniques Over Deep Learning Models. *Int. J. Emerg. Trends Eng. Res.* **2022**, *10*, 380–386. [CrossRef]
8. Zhong, M.; Taylor, R.; Bates, N.; Christey, D.; Basnet, H.; Flippin, J.; Palkovitz, S.; Dodhia, R.; Ferres, J. Acoustic Detection of Regionally Rare Bird Species through Deep Convolutional Neural Networks. *Ecol. Inform.* **2021**, *64*, 101333. [CrossRef]
9. Al Dawasari, H.J.; Bilal, M.; Moinuddin, M.; Arshad, K.; Assaleh, K. DeepVision: Enhanced Drone Detection and Recognition in Visible Imagery through Deep Learning Networks. *Sensors* **2023**, *23*, 8711. [CrossRef]
10. Mirugwe, A.; Nyirenda, J.; Dufourq, E. Automating Bird Detection Based on Webcam Captured Images Using Deep Learning. In Proceedings of the 43rd Conference of the South African Institute of Computer Scientists and Information Technologists. *EPiC Ser. Comput.* **2022**, *85*, 62–76. [CrossRef]
11. Steen, R. Bird Monitoring Using the Smartphone (iOS) Application Videography for Motion Detection. *Bird Study* **2017**, *64*, 62–69. [CrossRef]
12. Mpouziotas, D.; Karvelis, P.; Stylios, C. Advanced Computer Vision Methods for Tracking Wild Birds from Drone Footage. *Drones* **2024**, *8*, 259. [CrossRef]
13. Reddy, M.Y.; Ahmed, M.S.; Nagumothu, A.; Kavitha, M. Analysis of DenseNet-MobileNet-CNN Models on Image Classification using Bird Species Data. In Proceedings of the 2023 International Conference on Disruptive Technologies (ICDT), Greater Noida, India, 11–12 May 2023; pp. 536–542. [CrossRef]
14. Zendehdel, N.; Chen, H.; Leu, M.C. Real-time tool detection in smart manufacturing using You-Only-Look-Once (YOLO)v5. *Manuf. Lett.* **2023**, *35*, 1052–1059. [CrossRef]
15. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. *arXiv* **2021**, arXiv:2103.14030.
16. Mehta, S.; Rastegari, M. MobileViT: Light-weight, General-purpose, and Mobile-friendly Vision Transformer. *arXiv* **2021**, arXiv:2110.02178.
17. Li, Y.; Yuan, G.; Wen, Y.; Hu, J.; Evangelidis, G.; Tulyakov, S.; Wang, Y.; Ren, J. EfficientFormer: Vision Transformers at MobileNet Speed. *arXiv* **2022**, arXiv:2206.01191.
18. Li, Y.; Hu, J.; Wen, Y.; Evangelidis, G.; Salahi, K.; Wang, Y.; Tulyakov, S.; Ren, J. Rethinking Vision Transformers for MobileNet Size and Speed. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 1–6 October 2023. https://arxiv.org/abs/2212.08059. [CrossRef]
19. Lika, S.; Pernando, Y.; Kurniawan, A. Lightweight Deep Learning for Object Detection on Mobile Device. *Bull. Inform. Data Sci.* **2023**, *2*, 106. [CrossRef]
20. Mupparaju, S.; Thotakura, S.R.; Venkata Rami Reddy, C. A Review on YOLOv8 and Its Advancements. In *Data Intelligence and Cognitive Informatics*; Springer: New York, NY, USA, 2024; pp. 529–545. [CrossRef]
21. Fedorov, V. Railway Infrastructure Instance Segmentation Based on Convolutional Neural Networks. In Proceedings of the 2023 International Russian Automation Conference (RusAutoCon), Sochi, Russia, 10–16 September 2023; pp. 443–447. https://doi.org10.1109/RusAutoCon58002.2023.10272908.
22. Dhakal, S.; Sigdel, S.; Paudel, S.P.; Ranabhat, S.K.; Lamichhane, N. Mero Nagarikta: Advanced Nepali Citizenship Data Extractor with Deep Learning-Powered Text Detection and OCR Project. *arXiv* **2024**. [CrossRef]
23. Ma, M.; Pang, H. SP-YOLOv8s: An Improved YOLOv8s Model for Remote Sensing Image Tiny Object Detection. *Appl. Sci.* **2023**, *13*, 8161. [CrossRef]
24. Howard, A.; Pang, R.; Adam, H.; Le, Q.; Sandler, M.; Chen, B.; Wang, W.; Chen, L.-C.; Tan, M.; Chu, G.; et al. Searching for MobileNetV3. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324. [CrossRef]

25. Pang, D.; Guan, Z.; Luo, T.; Su, W.; Dou, R. Real-Time Detection of Road Manhole Covers with a Deep Learning Model. *Sci. Rep.* **2023**, *13*, 16479. [CrossRef]
26. Qian, S.; Ning, C.; Hu, Y. MobileNetV3 for Image Classification. In Proceedings of the 2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), Nanchang, China, 26–28 March 2021; pp. 490–497. [CrossRef]
27. Ahmed, S.; Bons, M. Edge Computed NILM: A Phone-Based Implementation Using MobileNet Compressed by TensorFlow Lite. In Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring, Virtual Event, 18 November 2020. [CrossRef]
28. Wilson, R.; Shenhav, A.; Straccia, M.; Cohen, J. The Eighty-Five Percent Rule for Optimal Learning. *Nat. Commun.* **2019**, *10*, 4646. [CrossRef]
29. Bird Detection Avisent. Bird-Detection Dataset. Roboflow Universe 2024. Available online: https://universe.roboflow.com/birddetectionavisent/bird-detection-kzmpu (accessed on 5 January 2025).
30. Khalili, B.; Smyth, A.W. SOD-YOLOv8—Enhancing YOLOv8 for Small Object Detection in Aerial Imagery and Traffic Scenes. *Sensors* **2024**, *24*, 6209. [CrossRef]
31. Ejaz, M. VGG16 and MobileNet Performance Evaluation on Edge Device in Self-Driving Car Technology. In Proceedings of the 2024 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS), Shah Alam, Malaysia, 29 June 2024; pp. 12–17. [CrossRef]
32. Liu, L.; Ke, C.; Lin, H.; Xu, H. Research on Pedestrian Detection Algorithm Based on MobileNet-YoLo. *Comput. Intell. Neurosci.* **2022**, *2022*, 8924027. [CrossRef] [PubMed]
33. Lou, H.; Duan, X.; Guo, J.; Liu, H.; Gu, J.; Bi, L.; Chen, H. DC-YOLOv8: Small-Size Object Detection Algorithm Based on Camera Sensor. *Electronics* **2023**, *12*, *2323. [CrossRef]*