An Object Detection Method for Native Seedling Identification in Caatinga and Atlantic Forest

Arysson S. Oliveira
University of Pernambuco
Recife, Brazil
aso1@ecomp.poli.br

Rodrigo Monteiro
University of Pernambuco
Recife, Brazil
rodrigo.monteiro@poli.br

Regina Aguiar
University of Pernambuco
Petrolina, Brazil
regina.aguiar@upe.br

Márcia Macedo
University of Pernambuco
Recife, Brazil
marcia.macedo@upe.br

Carmelo J. A. Bastos-Filho
University of Pernambuco
Recife, Brazil
carmelofilho@ieee.org

***Abstract*—Plants are essential for agriculture, environmental preservation, and ecosystem restoration. In biomes like the Caatinga and Atlantic Forest in Pernambuco, Brazil, seedling planting is a key strategy for recovering degraded areas. However, identifying plant species at the seedling stage is a challenging task. This work applies object detection models, specifically from the YOLOv11n family, to identify pinhão-bravo and pau-ferro seedlings, analyzing the impact of dataset size and training epochs. Results show that increasing the training dataset significantly improves performance, with data augmentation further enhancing model robustness, though without statistically significant differences. Conversely, increasing the number of epochs beyond 25 brought minimal gains. The findings demonstrate that object detection-based models are practical tools for supporting seedling monitoring in reforestation projects.**

***Index Terms*—Plant Seedling Identification, Object Detection, Computer Vision, Machine Learning, Deep Learning, environment.**

I. Introduction

Plants were among the first organisms to colonize the terrestrial environment, establishing themselves as foundational components of terrestrial ecosystems. They represent a critical element of the planet's natural capital by providing essential ecosystem services, which are typically classified into four categories: provisioning, regulating, supporting, and cultural services [1].

Within these ecosystems, plants perform numerous ecological functions, including serving as sources of food and habitat, participating in biogeochemical cycles, regulating climate, and purifying water. Additionally, they are utilized for medicinal purposes, supply fibers for textile production and raw materials for construction, and shape natural landscapes that serve recreational and touristic purposes. These contributions not only enhance quality of life but also reinforce cultural identity [1].

However, the continued provision of such ecosystem services is increasingly threatened by deforestation. The restoration of terrestrial ecosystems through seedling planting initiatives must prioritize the use of native species, whose correct identification is essential. Nevertheless, this identification process poses significant challenges, as classical taxonomy primarily relies on the morphological features of mature plants [2].

Traditional morphological identification of tree species depends on the observable physical attributes of leaves, flowers, fruits, seeds, bark patterns, and growth habits. For many native species, comprehensive morphological descriptions across early developmental stages are lacking. Seedlings, for example, are generally characterized by roots, stems, and leaves, and those deemed appropriate for planting typically exhibit a minimum height of 30 cm, a stem diameter of no less than 3 mm, and a formation period of at least three months [3].

To address the challenge of ecological restoration, the Government of the State of Pernambuco launched the Plantar Juntos program in 2024. This initiative aims to expand native vegetation cover through the reforestation and recovery of degraded areas within the Atlantic Forest and Caatinga biomes. The program targets the planting of four million trees by 2026, thereby contributing to ecological restoration, biodiversity conservation, and climate change mitigation [4]. The Plantar Juntos program also integrates a technological platform to georeference seedlings, monitor the achievement of reforestation goals, and systematize data gathered by citizens, institutions, and partner agencies.

In this context, technological solutions based on machine learning and computer vision have the potential to support the identification of plant species at early development stages. The success of these models depends on the availability of sufficiently large and diverse datasets, which remains a challenge for native species in biodiversity hotspots like the Caatinga and Atlantic Forest. Although there are important initiatives, such as iNaturalist and Pl@ntNet, which offer extensive collections of labeled plant images, these datasets are often biased towards species from regions with higher user engagement, resulting in limited representation of local species from ecosystems like the Caatinga. This data scarcity is one of the main challenges that motivates this work, particularly the development of a specialized dataset focused on native species to improve model performance in these unique biomes.

This paper aims to test the performance of detection models for species identification by varying the dataset size and the number of training epochs. Additionally, it includes the construction of a dedicated dataset focused on native species, addressing the lack of labeled data for these ecosystems. By varying the number of images and epochs, the goal is to analyze the learning behavior of the models, identify performance saturation points, and understand how dataset size impacts accuracy and generalization.

II. Related works

This section presents a review of previous works related to plant species identification using deep learning, with an emphasis on object detection models, datasets, and their application to real-world plant imagery.

There is a wide variety of image datasets focused on recognizing plant species. Some of these datasets were created in laboratory settings under controlled conditions, such as the PlantVillage dataset [5]. However, most real-world applications, such as agriculture and environmental monitoring, require "in-the-wild" conditions, featuring images of plants in open environments that may include multiple objects and background variability. This characteristic is crucial for ensuring the generalization capability of deep learning models.

Depending on the task, it is possible to find datasets that offer both a wide diversity of species and a sufficient number of samples for training deep learning models. The most significant providers of such datasets include iNaturalist [6], Pl@ntNet [7], and LifeCLEF [8]. These datasets are compiled from images submitted by thousands of users, captured using various devices, including cameras and smartphones. Furthermore, the images are taken under diverse conditions, including variations in lighting, background, and geographic location.

Deep learning models are highly relevant tools for plant species identification, as they enable the automation of complex visual recognition tasks with high accuracy. In agricultural applications, for example, one of the most common uses is the detection and classification of weed species. Accurate weed identification is crucial for precision farming, enabling targeted herbicide applications and minimizing environmental impact. These models can process high-resolution images to distinguish between crop plants and invasive species, even under variable lighting conditions, occlusion, or complex backgrounds. They have also been successfully integrated into unmanned aerial vehicles (UAVs), smart sprayers, and mobile devices, allowing for real-time decision-making in the field. This characteristic also makes deep learning approaches suitable for applications such as biodiversity monitoring in remote and inaccessible areas [9].

Among the main applications of deep learning in plant species recognition are image classification and object detection. Classification involves assigning a label or category to an image based on its visual characteristics. It was one of the first computer vision problems to benefit from significant and large-scale advances enabled by deep learning techniques [9].

Several studies have reported high performance in plant image classification using well-established convolutional neural network (CNN) architectures, with ResNet, VGG, DenseNet, Inception, EfficientNet, Transformer-based models, MobileNet, and Xception being among the most frequently used [10], [11], [11]. These models extract features through convolutional layers and typically use fully connected (FC) layers followed by a softmax function for final class prediction. In other studies, traditional classifiers such as support vector machines (SVM), multilayer perceptrons (MLP), and random forests (RF) have been used instead of softmax, employing the convolutional layers as feature extractors [12]–[14].

Object detection differs from image classification in that it addresses not only the classification of objects but also their localization within the image. In plant image analysis, the most frequently used detection models include YOLO, Faster R-CNN, CenterNet, SSD, RetinaNet, and Transformer-based approaches [9]. These models are typically classified into region-based and region-free. Region-based models, such as Faster R-CNN, extract feature maps, generate region proposals using techniques like anchor boxes (often optimized via K-means clustering), and apply RoI pooling and classification layers to determine object categories and locations [15]–[19]. Although highly accurate, these models tend to be computationally intensive. On the other hand, region-free models, such as YOLO, treat detection as a regression problem, predicting object classes and bounding boxes directly from grid-based divisions of the image. YOLO variants such as YOLOv4 and YOLOv9 are popular due to their efficiency, with enhancements like CSPNet enabling lightweight deployment on edge devices [20]–[23].

Despite the availability of diverse datasets and deep learning approaches for plant species identification, they present certain limitations regarding their intended application in the present work. One of the main limitations concerns the compatibility between the plant species in these datasets and those found in the semi-arid region of Pernambuco. The datasets are often composed of images of plant species that are not native to Pernambuco, making it challenging to develop plant identification solutions that perform well in the local context. Another limitation involves the annotation formats used in these datasets, which may not align with the requirements of the deep learning models to be employed, thereby restricting their usability.

III. Methodology

Figure 1 presents the flowchart that summarizes the methodology adopted in this work. The subsequent subsections describe each step in detail, explaining the processes and decisions involved throughout the development.

A. Dataset creation and labeling

Due to the limitations of existing datasets, which primarily contain images of widely known or local plant species, it was necessary to create a custom dataset. This need was significant

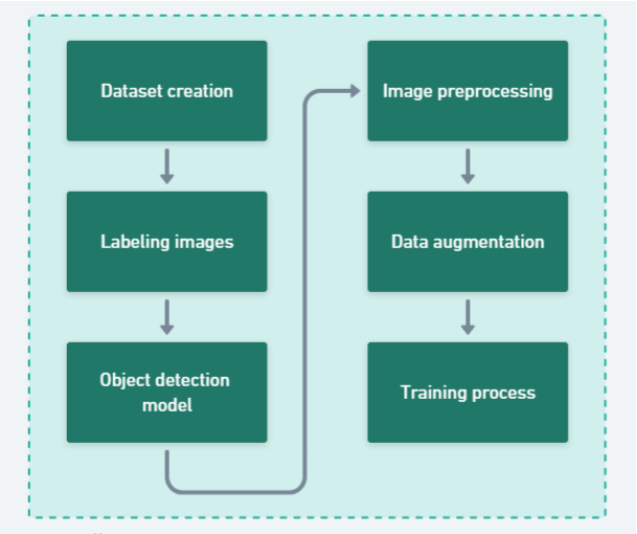

Fig. 1. Methodology flowchart

for the development of a specialized dataset focused on native species of the Caatinga and Atlantic Forest biomes.

The first step to create the dataset was to perform the acquisition of images of the seedlings using a rotating platform on which the seedlings were placed, and photographs were taken in different positions; 15 images were taken of each seedling:

- Four images with a camera in the front position and platform in positions 0°, 90° 180°, and 270°;
- Four images with camera tilted, at approximately 30°, and platform in positions 30°, 120°, 210° and 300°;
- Four images with camera tilted, at approximately 60°, and platform in positions 60°, 150°, 240° and 330°;
- Three images with a camera in the upper position.

The resulting dataset is composed of images of two native plant species: pinhão bravo (*Jatropha mollissima (Pohl) Baill.*), shown in Figure 2, with 224 images, and pau ferro (*Libidibia ferrea (Mart. ex Tul.) L.P.Queiroz*) as shown in Figure 3, with 271 images, totaling 495 images. These two species were initially selected for their ecological relevance and the frequency with which they are used in reforestation actions in the state of Pernambuco. The choice was also made to validate the process of automatic identification of plant species using Artificial Intelligence, allowing controlled tests to be carried out with a balanced number of samples and distinct morphological characteristics.

The labeling process was carried out using the Roboflow platform [25], which facilitates the marking of bounding boxes used during detection training. We decided to identify plants by the treetops of the seedlings. This approach was chosen because the treetop is the most visually distinguishable part of the seedling in aerial and field images, allowing for more accurate localization and minimizing ambiguities related to stem or shadow detection. The labeled dataset was then exported in YOLO format for use in the subsequent model training phase.

Fig. 2. Example of pinhao bravo with bounding-boxes

Fig. 3. Example of pau-ferro with bounding-boxes

B. Model for leaf detection

The leaf detection model used in the experiments was YOLOv11, pre-trained on the MS COCO dataset. YOLOv11 is the latest version of the series of real-time object detection models developed by Ultralytics. This version introduces architectural improvements, including C3K2 blocks, the Spatial Pyramid Pooling Fast (SPPF) module, and the Convolutional block with Parallel Spatial Attention (C2PSA) attention mechanism. These innovations enhance feature extraction and computational efficiency, leading to improved accuracy and inference speed. Compared to YOLOv8, also developed by Ultralytics, YOLOv11 achieves a higher mean average precision (mAP) on the COCO dataset while using 22% fewer parameters, making it more efficient without compromising accuracy [24].

The YOLOv11 Nano (YOLOv11n) is the smallest and fastest version of the YOLOv11 family, designed specifically for devices with limited computing resources, such as edge devices. YOLOv11n has only 2.6 million parameters, delivering a mAP@50-95 accuracy of approximately 0.4 on the COCO dataset while maintaining relatively low latency, making it ideal for real-time applications in compute-constrained environments. Despite its small size, YOLOv11n maintains competitive performance, making it a practical choice for object detection tasks on devices with limited compute resources, such as species detection in *Plantar Juntos*.

C. Images preprocessing

The database images were resized to 640x640 pixels, which is a commonly used size to simplify input to YOLO family models, such as YOLOv11n, standardizing all images regardless of their original resolution. While geometric distortion may slightly alter the shape of objects, this approach improves training speed and avoids the additional computational cost associated with zero padding.

D. Data augmentation

Data augmentation is the process of artificially generating new data from existing data. It is widely used in training new machine learning models, especially when it comes to deep learning models, which require extensive and diverse datasets for training. However, providing sufficiently diverse real-world datasets can be a challenge due to data acquisition difficulties, regulations, and other limitations. Data augmentation artificially increases the data set by making small changes to the original data, bringing benefits to the training process of machine learning-based models [26].

In our experiment, four techniques of data augmentation were used in conjunction:

- Noise addition consists of applying disturbances to the images, such as salt and pepper noise, to improve the model's robustness to imperfections and variations in real-world conditions. A value of 5% was used in this experiment.
- Rotation is a data augmentation technique that improves the model's ability to detect objects in different orientations. In this experiment, rotations from -45° to +45° were applied.
- Flip is a simple augmentation technique that inverts images horizontally or vertically to improve the model's invariance to object position. This experiment applied flips with a 50% probability for both directions.
- Crop is an augmentation technique that removes random portions of the image to simulate different framings and distances. In this experiment, crops ranged from 0% to 20% of the image.

E. Training process

The training process adopted in this research followed two stages. The first stage involved a systematic approach to assess the impact of dataset size on the performance of object detection models, while the second stage aimed to evaluate the impact of the number of epochs during training on model performance.

In the first stage, five different training scenarios were defined, each consisting of 86, 174, 261, and 347 original images, as well as a fifth scenario containing 797 images obtained through data augmentation techniques, including rotation, noise addition, mirroring, and cropping. This strategy allowed us to analyze the evolution of the models' performance as the database was expanded, both with real data and synthetic data.

For the second stage, only two scenarios were defined, comprising 347 original images and 797 images generated through data augmentation from the first dataset. In this scenario, five epochs of model training were considered: 10, 25, 50, 100, and 200 epochs. This strategy enabled us to determine the optimal number of epochs required for sufficient training, thereby minimizing wasted training time.

In all scenarios, we trained 10 independent models, totaling 100 experiments, which enabled us to evaluate the variability of results due to randomness in sample selection, weight initialization, and optimization dynamics. The data were divided as follows: 80% for training, 10% for validation, and 10% for testing. The training was performed using a pre-trained model from the MS COCO database, utilizing weights previously adjusted for generic object detection tasks, which accelerates convergence and improves performance even on smaller databases.

To control overfitting and optimize computational resources, the early stopping technique was adopted, with patience of 50 epochs, interrupting training if there was no improvement in the validation metric after this period. For the first test scenario, where the number of images varied, the maximum number of epochs was limited to 500, ensuring consistency across scenarios. Additionally, an adaptive batch size was employed, automatically adjusted according to the available GPU's capacity, ensuring greater flexibility in utilizing hardware resources.

This protocol allowed a robust comparison of the different scenarios, considering both the mean and the variability of performance metrics, such as *Recall*, *Precision*, *mAP@50* and *mAP@50-95*, faithfully reflecting the influence of the size and diversity of the training base on the generalization of the models. All training was carried out on the Google Colab platform, which supports NVIDIA Tesla T4 GPUs, providing an environment with accessible and suitable computational resources for experiments involving detection models based on deep learning.

F. Evaluation metrics

The performance evaluation of the models was conducted based on four metrics widely used in object detection tasks: Recall, Precision, mAP@50, and mAP@50-95.

- Recall: Measures the model's ability to detect all relevant objects. It is the ratio of true positives to true positives plus false negatives. High Recall means fewer missed objects.
- Precision: Indicates how many detections are correct. It is the ratio of true positives to true positives plus false positives. High Precision means fewer false detections.
- mAP@50: Represents the average Precision with Intersection over Union (IoU) greater than 50%, showing the model's overall detection quality at this overlap threshold.
- mAP@50-95: A stricter metric averaging precision across IoU thresholds from 50% to 95%, providing a more detailed evaluation of detection accuracy.

IV. Results

This section presents the results obtained for the training process in 2 scenarios. One scenario involves varying the number of images in the dataset, and the second scenario involves applying a variation in the number of epochs during the training process.

A. Results for image amount variation

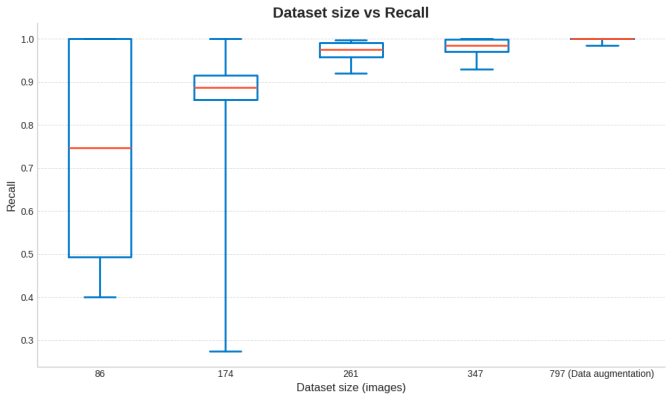

Fig. 4. Results of Dataset size versus recall

The results shown in Figure 4 indicate a progressive improvement in the models' performance in terms of Recall as the number of images in the training base increases. In the scenario with 86 images, Recall varied between 0.4 and 1, while the model trained with 174 images presented an even greater variation, from 0.25 to 1, evidencing high instability in the results. With 261 images, Recall stabilized in a range between 0.9 and 1, a performance that was maintained in the scenario with 347 images, although with a visually smaller dispersion, indicating greater consistency in the model's behavior.

Using a small number of training images can result in a significant variation in Recall values due to the database's insufficient representativeness. It means that the model may not learn adequately to identify relevant patterns, resulting in unstable performance that is sensitive to the randomness of the training samples, as observed in Figure 4. As a consequence, the model may either fail to identify present objects (low Recall) or occasionally get it right by coincidence, which justifies the high dispersion observed in the scenarios with 86 and 174 images. As the training base is expanded, the model receives more examples, becoming more robust and reliable in identifying target objects.

The application of data augmentation allowed the expansion of the training base to 797 images, resulting in an even more consistent performance, with Recall values ranging from 0.95 to 1. Furthermore, the use of the Wilcoxon statistical test does not guarantee that using data augmentation provides better performance than the scenario with 347 images. On the other hand, the scenarios with 261 and 347 images presented statistically equivalent performance, suggesting a saturation point from which the addition of real images, without increasing variability, does not result in additional significant gains.

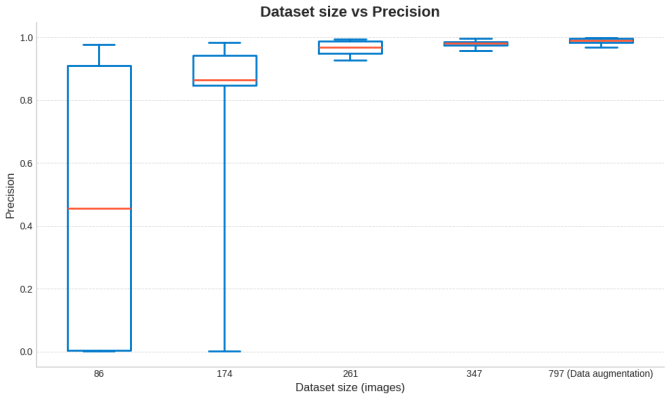

Fig. 5. Results of Dataset size versus precision

In Figure 5, an improvement in Precision performance can be observed as the number of images in the training base increases. In the scenarios with 86 and 174 images, Precision varied widely between 0 and 1, indicating high variability in the performance of the trained models. This large dispersion suggests that, with small datasets, the model is more influenced by the specific selection of samples, which can lead to instability in learning and poor generalization. In contrast, the models trained with 261 and 347 images presented Precision between 0.9 and 1, with visually smaller dispersion in the case of 347 images, indicating greater consistency and reliability in the results. These findings reinforce the importance of a more robust training base to achieve more accurate and stable models.

As occurred in the Recall results, with few examples, the model has a greater chance of memorizing specific patterns instead of learning general rules, which can lead to false positives or erratic behavior in new data. It results in a very unstable Accuracy, which can vary dramatically between different runs. Furthermore, any imbalance or noise in the limited number of available images can significantly impact learning, thereby compromising the model's ability to accurately differentiate between true and false positives. As the number of images increases, Accuracy tends to stabilize, indicating greater confidence in the model's discriminative ability.

Additionally, the use of data augmentation led to even more consistent performance. In this scenario, Precision varied between 0.95 and 1, outperforming the other scenarios in terms of stability. However, the Wilcoxon test does not guarantee that using data augmentation provides better performance than the scenario with 347 images.

The results presented in Figure 6 indicate a growing trend in the value of the mAP@50 metric as the number of training images increases. The model trained with 86 images presented values between 0.8 and 0.95, while with 174 images, the results varied between 0.85 and 1. The models trained with 261 and 347 images achieved values between 0.95 and 1, demonstrating similar average performance and suggesting possible model saturation. However, the model trained with

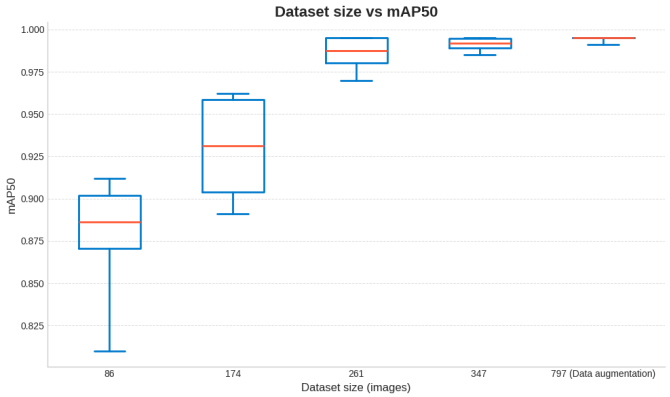

Fig. 6. Results of Dataset size versus mAP50

347 images presented an even smaller dispersion of results, indicating greater consistency in performance.

With the use of data augmentation, the mAP@50 value ranged from 0.99 to 1, which corresponded to the scenario with the least variation in results. This behavior suggests that the synthetic augmentation strategy employed in the database contributed to the greater stability and robustness of the model. However, despite the apparent improvement, the results of the Wilcoxon test did not indicate a statistically significant difference between this scenario and the model trained with 347 images. It suggests that, although the use of data augmentation provides slightly better and more stable performance, it cannot be said with statistical confidence that it outperforms the scenario with 347 original images.

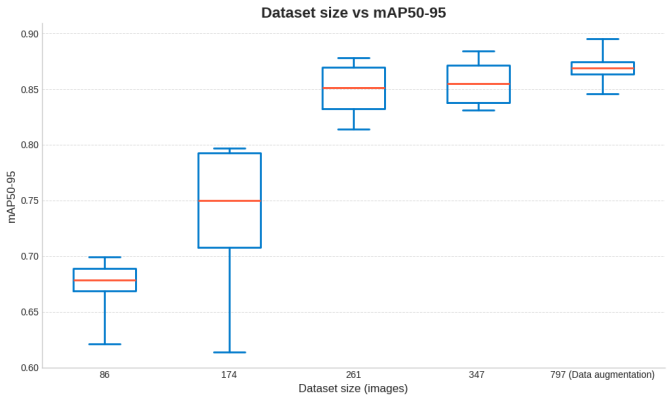

Fig. 7. Results of Dataset size versus mAP50-95

The results presented in Figure 7 indicate a trend of improvement in the models' performance as the number of images used during training increases, as measured by the mAP@50-95 metric. It was observed that the model trained with 86 images presented metric values between 0.6 and 0.7, while with 174 images, the results varied between 0.6 and 0.8. For the scenarios with 261 and 347 images, the values were between 0.8 and 0.9, demonstrating similar performance and suggesting a possible saturation of the model; that is, the increase in the training set did not result in significant additional performance gains.

In the scenario with data augmentation, the mAP@50-95 value ranged from 0.85 to 0.9, indicating a minor variation among the results of all the tested conditions. This stability indicates that the introduction of synthetic images contributed to greater consistency in the model's behavior, even without expanding the base with new original images. However, the results of the Wilcoxon test do not guarantee that this scenario is statistically superior to the model trained with 347 images, reinforcing the idea that, although data augmentation can improve performance stability, the gains in terms of the metric average are not statistically significant when compared to an already robust base.

B. Results for epochs amount variation

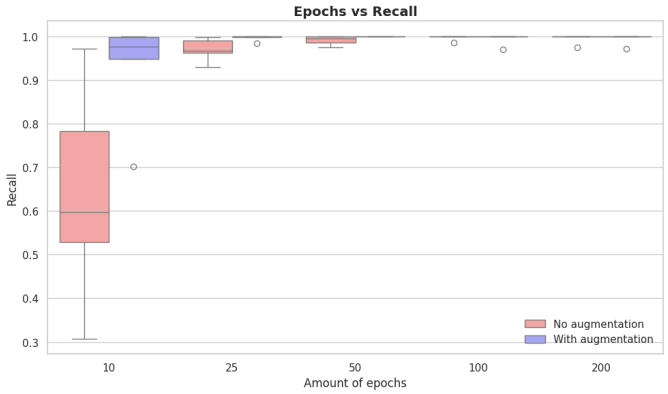

Fig. 8. Results of epochs versus recall

Figure 10 indicates that by increasing the number of epochs, we have a significant improvement in Recall. When training the model without data augmentation for only 10 epochs, the Recall varies between 0.307 and 0.907, indicating high instability in the results. However, for the model with data augmentation, training with only 10 epochs already shows that the Recall varied between 0.7 and 1, indicating high stability with a small number of epochs.

When training with 25 epochs for the dataset without data augmentation, the variation decreases significantly, remaining around 0.92 and 0.99, showing even greater stability for the model. For the other epochs, there is practically no variation in Recall. It shows that the model had already stabilized its learning by around 25 epochs, both for the model trained without data augmentation and for the model trained using data augmentation.

In Figure 9, although the boxplot suggests considerable variation in the results, it is important to note that the precision values are consistently high, ranging from 0.92 to 1.0. It indicates that both models — with and without data augmentation — achieved satisfactory precision even with a reduced number of epochs. After 10 epochs, both models already exhibit good precision levels, demonstrating that the models quickly learn to distinguish the classes. However, it is possible to observe

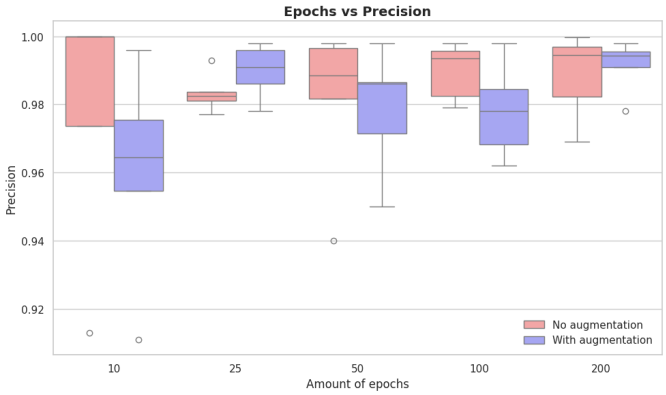

Fig. 9. Results of epochs versus precision

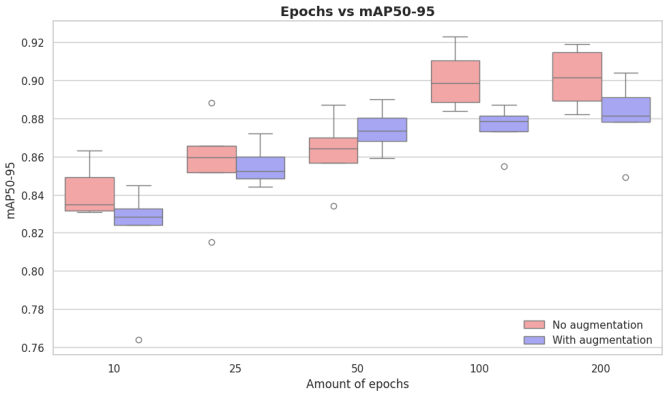

Fig. 11. Results of epochs versus mAP50-95

that the model trained without augmentation shows slightly less variability in most scenarios. In contrast, the model with augmentation presents a wider spread in some cases, especially at 50 and 100 epochs. It suggests that while augmentation introduces more diversity in the training data, potentially improving generalization, it may also lead to greater variance in precision during training.

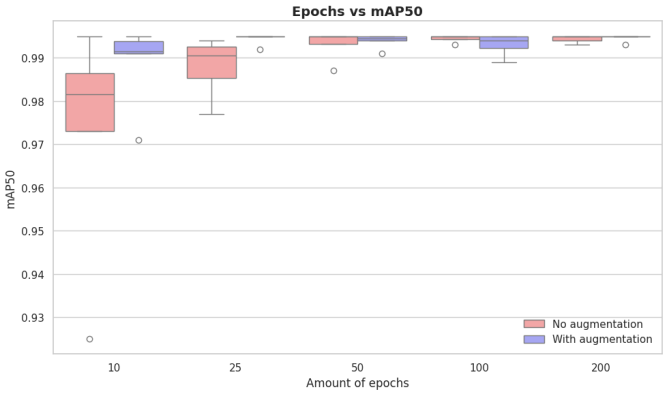

Fig. 10. Results of epochs versus mAP50

The results obtained from the map50 are shown in Figure 10. It is observed that both training strategies achieved high mAP50 values across all configurations. However, when using data augmentation, the models demonstrated greater stability, particularly in scenarios with a low number of epochs. For instance, at 10 epochs, the model with augmentation reached a higher median mAP50 with less variability compared to the model trained without augmentation, which exhibited a wider distribution and lower minimum values, including some outliers below 0.93. As the number of epochs increased, the differences between the two strategies diminished. From 50 epochs onward, both configurations converged to similar performance levels, with mAP50 values consistently above 0.99 and low variance.

In Figure 11, it is possible to observe the behavior of the models in terms of mAP50-95, a more stringent and comprehensive metric for evaluating object detection performance. Unlike what was observed for precision and mAP50, the difference between the models with and without data augmentation becomes more noticeable.

The model trained without augmentation consistently achieves higher mAP50-95 values as the number of epochs increases. This difference becomes particularly evident after 100 epochs, where the no-augmentation model reaches values close to 0.91, while the model with augmentation stabilizes around 0.88. For this dataset, data augmentation may introduce additional variability that slightly hinders the model's ability to generalize across more challenging IoU thresholds. Additionally, it is noticeable that the model without augmentation not only achieves higher mAP50-95 but also shows a slight reduction in variability with more epochs, indicating more stable learning. On the other hand, the model with augmentation demonstrates more consistent, albeit lower, performance across all epoch settings.

V. Conclusion

In this work, it was possible to compare the performance of detection models for identifying the species *pinhão-bravo* and *pau-ferro*, varying the dataset size and the number of training epochs.

A consistent improvement in the results was observed as the volume of training data increased, with satisfactory performance, especially in scenarios where 100% of the images from the training set were used. Additionally, the application of data augmentation techniques contributed to the model's robustness, yielding slightly better means and standard deviations than those obtained with the complete set. However, this difference was not statistically significant according to the Wilcoxon test. On the other hand, increasing the number of epochs did not yield significant improvements in the model's learning; around 25 epochs were already sufficient for the model to achieve quite satisfactory results.

These results indicate that object detection models are promising for practical applications, such as monitoring seedling planting in reforestation projects, particularly in the state of Pernambuco, within the *Plantar Juntos* project. The combination of good performance with the possibility of

optimization via data augmentation can contribute to automated environmental monitoring systems, allowing greater scale and accuracy in data collection in reforestation areas. These findings underscore the potential of artificial intelligence as a strategic tool for the sustainable management of native vegetation and the evaluation of public policies aimed at environmental recovery.

For future work, we will explore alternative models and experiment with different labeling strategies, such as labeling by individual leaves. Additionally, we plan to expand the approach to include other plant species, aiming to improve the model's generalization and robustness across diverse scenarios.

ACKNOWLEDGMENT

We would like to express our gratitude to the Secretariat of Environment and Sustainability (SEMAS) of the State Government of Pernambuco, the University of Pernambuco, the National Council for Scientific and Technological Development (CNPq), and Coordination for the Improvement of Higher Education Personnel (CAPES).

REFERENCES

[1] G. T. Miller and S. E. Spoolman, *Environmental Science*, 16th ed. Boston, MA, USA: Cengage Learning, 2019.

[2] P. H. Raven, R. F. Evert, and S. E. Eichhorn, *Biology of Plants*, 8th ed. New York, NY, USA: W.H. Freeman and Company, 2014.

[3] H. Lorenzi, *Árvores brasileiras: manual de identificação e cultivo de plantas arbóreas nativas do Brasil*, 6th ed., vol. 1. Nova Odessa, SP, Brazil: Instituto Plantarum, 2016.

[4] Secretaria de Meio Ambiente e Sustentabilidade de Pernambuco, "Plantar Juntos," *SEMAS Pernambuco*. [Online]. Available: https://semas.pe.gov.br/plantar-juntos/. [Accessed: May 27, 2025].

[5] D. Hughes *et al.*, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," *arXiv preprint* arXiv:1511.08060, 2015.

[6] G. Van Horn *et al.*, "The iNaturalist species classification and detection dataset," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pp. 8769–8778, 2018.

[7] C. Garcin, A. Joly, P. Bonnet, J.-C. Lombardo, A. Affouard, M. Chouet, M. Servajean, T. Lorieul, and J. Salmon, "Pl@ntNet-300K: a plant image dataset with high label ambiguity and a long-tailed distribution," in *Proc. 35th Conf. Neural Information Processing Systems (NeurIPS)*, 2021.

[8] A. Joly *et al.*, "Interactive plant identification based on social image data," *Multimedia Tools and Applications*, vol. 75, no. 19, pp. 11877–11904, Oct. 2016.

[9] J. Xu, Y. Chen, and X. Liu, "Plant image recognition with deep learning: A review," *Comput. Electron. Agric.*, vol. 212, p. 108072, 2023.

[10] J. Too, L. Yujian, S. Njuki, and L. Yingchun, "A comparative study of fine-tuning deep learning models for plant disease identification," *Comput. Electron. Agric.*, vol. 161, pp. 272–279, 2019.

[11] M. Conciatori, *et al.*, "Plant species classification and biodiversity estimation from UAV images with deep learning," *Remote Sens.*, vol. 16, no. 19, p. 3654, 2024.

[12] A. Mathur and G. M. Foody, "Multiclass and binary SVM classification: Implications for training and classification users," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 2, pp. 241–245, 2008.

[13] P. K. Sethy, S. Behera, and S. Ratha, "Detection of plant leaf disease using image segmentation and SVM classifier," in *Proc. Int. Conf. Emerg. Trends Inf. Technol. Eng. (ic-ETITE)*, 2020.

[14] C. Xie, Y. Wang, J. Zhang, X. Chen, and Y. Yang, "Multi-level learning features-based identification of tomato leaf diseases," *IEEE Access*, vol. 9, pp. 144080–144092, 2021.

[15] Z. Gao, X. Zhang, and Y. Liu, "Improved anchor box generation for object detection using K-means clustering," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2020.

[16] Y. Wang and J. Liu, "Optimizing anchor box dimensions in object detection with K-means++ clustering," *J. Imaging*, vol. 7, no. 1, p. 8, 2021.

[17] P. Kaur, H. Singh, and R. Kaur, "Improving plant object detection with optimized anchor proportions," *Multimedia Tools Appl.*, vol. 81, pp. 18315–18334, 2022.

[18] Y. Mu, *et al.*, "A Faster R-CNN-based model for the identification of weed seedling," *Agronomy*, vol. 12, no. 11, p. 2867, 2022.

[19] T. Abbas, *et al.*, "Deep neural networks for automatic flower species localization and recognition," *Comput. Intell. Neurosci.*, vol. 2022, no. 1, p. 9359353, 2022.

[20] P. Zhang and D. Li, "YOLO-VOLO-LS: A novel method for variety identification of early lettuce seedlings," *Front. Plant Sci.*, vol. 13, p. 806878, 2022.

[21] Y. Zhang, *et al.*. He, "Real-time strawberry detection using deep neural networks on embedded system (RTSD-Net): An edge AI application," *Comput. Electron. Agric.*, vol. 192, p. 106586, 2022.

[22] Y. Huang, *et al.*, "YOLO-IAPs: A rapid detection method for invasive alien plants in the wild based on improved YOLOv9," *Agriculture*, vol. 14, no. 12, p. 2201, 2024.

[23] Y. Wang, X. Lin, Z. Xiang, and W.-H. Su, "VM-YOLO: YOLO with VMamba for strawberry flowers detection," *Plants*, vol. 14, no. 3, p. 468, 2025.

[24] R. Khanam and M. Hussain, "YOLOv11: An Overview of the Key Architectural Enhancements," *arXiv preprint arXiv:2410.17725*, Oct. 2024.

[25] Roboflow, "Roboflow: Computer vision tools for developers and enterprises," Roboflow, 2025. [Online]. Available: https://roboflow.com/. [Accessed: 4-Jun-2025].

[26] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, Jul. 2019.