# Machine Learning for Monitoring the Spread of Carmine Cochineal on Different Accessions of Forage Palm Plants

João Vieira
*Agricultural School of Jundiai*
*Federal University of Rio Grande do Norte (UFRN)*
Macaiba, Brazil
joao.vieira.712@ufrn.edu.br

Josenalde Oliveira
*Agricultural School of Jundiai*
*Federal University of Rio Grande do Norte (UFRN)*
Macaiba, Brazil
josenalde.oliveira@ufrn.br

Marcone Chagas
*Agricultural Research Company of Rio Grande do Norte (EMPARN)*
Parnamirim, Brazil
*Embrapa Cotton*
Campina Grande, Brazil
conna1656@gmail.com

***Abstract*—Entomologists and experts on the mitigation and control of agricultural pests need to evaluate the resistance of different species and accessions within the same plant species, which is usually carried out manually with visual inspection. This paper presents an automatic approach for carmine cochineal identification and counting based on machine learning algorithms. A dataset is created from actual experiments and through a pipeline for feature engineering, getting both morphological and color features. The models Gradient Boosting, XGBoost, Stochastic Gradient Descent (SGD), Multilayer Perceptron Neural Network (MLP) and Logistic Regression are evaluated in a binary classification task. XGBoost and Gradient Boosting outperform the other models, with an accuracy of $93\%$ on the test set, with a good generalization property and without the need of deep learning or more complex models. Moreover, the solution can be embedded into a software to aid experts on these agronomic comparisons for proper pest management.**

***Index Terms*—supervised learning, image processing, agricultural pests, emsemble models, artificial intelligence, entomology**

## I. Introduction

The multispecies plant genus *Opuntia* plays a significant nutritional role not only in Brazil but also in many countries in the Near East and North Africa (NENA) region (18 member countries of the Food and Agriculture Organization of the United Nations (FAO)) and other arid regions throughout the world [1]. However there is a major concern on its vulnerability to pest attacks on the field, such as those from carmine cochineal *(Dactylopius Opuntiae)*. Its bright red carmine dye has economic importance for agroindustry [2], however, for most producers, this pest is a significant threat to their production if the infestation is not properly treated and controlled. Outbreaks can decimate orchards, completely destroying them in a matter of months, devastating the livelihoods that depend on them [1]. In Brazil, it has decimated extensive areas of the Northeast [3]. The pest is a red scale insect and its presence is most noticeable on the plant when the nymphs secrete a white wax over their bodies (Figure 1b), turning the cactus white in color. Figure 1 presents the four growing states [4].

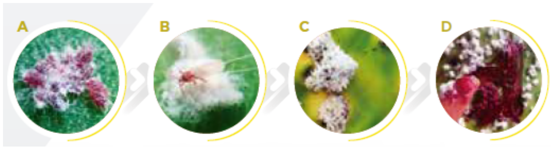

Fig. 1. (A) colony of fixed nymphs in 1st instar. (B) adult male on 2nd instar nymph. (C) colonies of adult females with male cocoons. (D) carmine from adult females. Source: [4], Marcone César Mendonça das Chagas (EMPARN/Embrapa Algodão)

Entomologists dedicated to their study and characterization have been conducting research to select the most resistive assertions of forage palm plants (*Opuntia*) to carmine cochineal [3], [5]. However, these experiments in greenhouses or even in the laboratory environment often rely on visual inspections to obtain an approximate insect count. The higher the count, the higher the suitability of that plant accession to the insect presence, and, therefore, that particular plant accession may be disregarded as a resistive option for the growers. An example of such an evaluation can be seen in Figure 2 where the infected susceptible sample lies in the center of the greenhouse, with surrounding samples of each accession. In Figure 3 it is shown another setup of an experiment conducted in a laboratory.

Periodically, an expert inspects each sample and quantifies the presence of the insects on the plant surface. This presence is associated to the identification of a white wax with a tuft-like aspect (Figure 4) where the cochineal lies under it. This method is often time-consuming, rely heavily on subjective

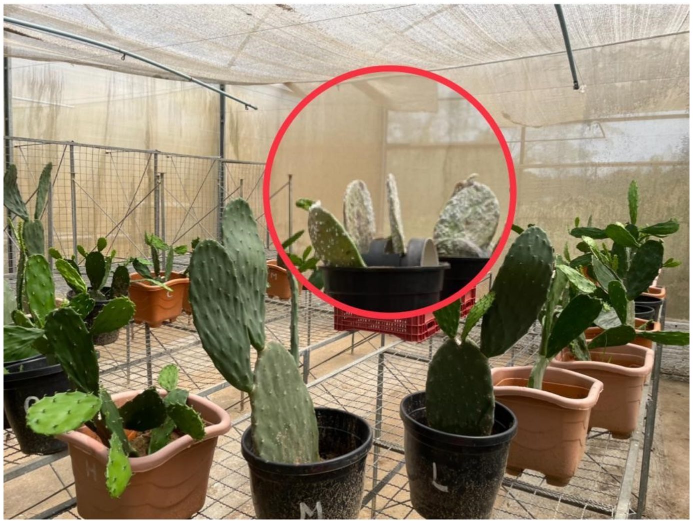

Fig. 2. A setup within a greenhouse for evaluating palm plant (*Opuntia*) resistance to carmine cochineal. At the middle the susceptible sample is highlighted. Source: Marcone César Mendonça das Chagas (Empresa de Pesquisa Agropecuária do RN - EMPARN/Embrapa Algodão)

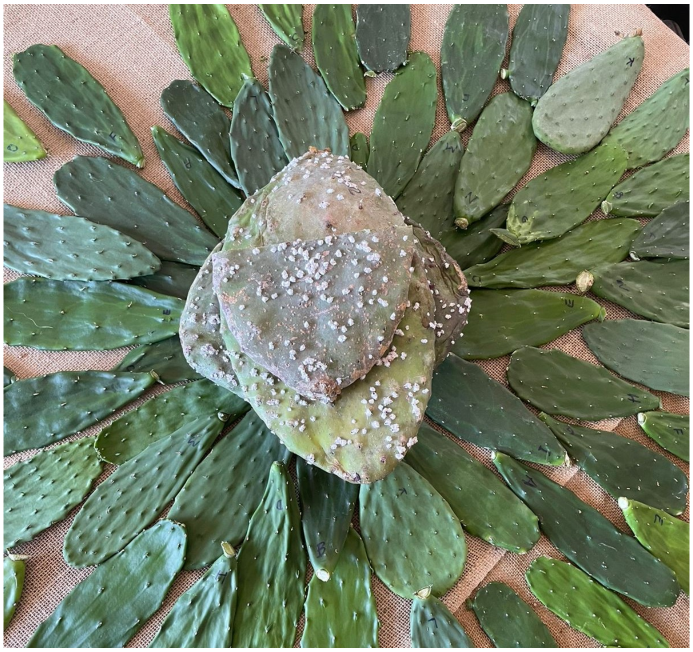

Fig. 3. Another setup within a laboratory facility for evaluating palm plant (*Opuntia*) resistance to carmine cochineal. At the middle the susceptible (infected) sample is highlighted. Source: Marcone César Mendonça das Chagas (EMPARN/Embrapa Algodão)

assessments and it is prone to inaccuracies. Considering the morphological aspects and color-based features of the carmine cochineal, a pest identification task can improve this visual counting process, thus separating the white wax from the thorns and other non-insect elements.

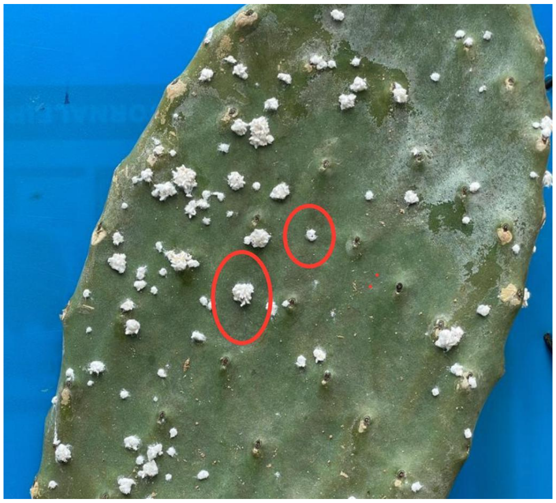

Fig. 4. An *Opuntia* sample with carmine cochineal (white points) throughout its surface. The blue background was added at the image collection phase in order to improve the image segmentation. Source: Marcone César Mendonça das Chagas (EMPARN/Embrapa Algodão)

In agriculture, several applications of machine learning algorithms for pest identification and classification are being investigated and actually used [6], [7], [8], [9], [10], [11], [12], leading to image-based identification frameworks using Convolutional Neural Networks (CNN) [13] or ensemble of deep learning models [14], mostly applied when a variety of pests is expected [16]. Some researches state the transformation that AI is bringing tho this field [17]. Indeed, accuracies higher than 90% are usual and these results underscore the transformative potential of deep learning for early detection and agricultural diagnostics [15], although there is no widespread availability of imageset for agricultural pests. Known sources as Agricultural Pest dataset [19] does not include species of cochineal and AgriPest [20] is domain-specific. More recently, [21] and [22] shared their dataset with images of healthy and damaged cacti under open field conditions. Usually, researches need to collect images and build a new dataset or fine-tune transfer learning models with new data, where usually some method of data augmentation is necessary. There is an increasing research interest on this topic [18].

On the other hand, machine learning classification algorithms classified as non-deep maintain their foundational interest and applications, not only when intended to be executed on low performance devices (tinyML) [24], but also providing higher interpretability and accuracy to outputs. In [23] twelve baseline models were trained for two common pests in rice, evaluated using F1 scores and AUC values due to the imbalanced nature of the dataset, and then reduced to four models for hyperparameter tuning, Random Forest, Balanced Random Forest, XGBoost and CatBoost, which outperformed the others. Again, accuracies compatible with deep learning techniques were achieved.

In this work, it is presented the evaluation of six classification models (Stochastic Gradient Descent, Gradient Boosting, XGBoost, Random Forest, Logistic Regression and Multilayer Perceptron) for identification of carmine cochineal white wax on palm samples. This choice is based on the diverse nature of each algorithm and wide application in similar problems. As far as author's knowledge there is no previous related works for carmine cochineal, although a deep learning approach had been applied to scale cochineal *(Diaspis echinocact)*, another pest [25].

This work is organized as follows: section II presents the feature engineering steps to get characteristics from the images and build the dataset for training the classification models. Section III presents the results and discussion of the application of all six models, highlighting their comparisons regarding the main metrics used for classification tasks. Section IV concludes with the main findings and intended work for the near future.

## II. Feature Engineering

This section describes the data gathering, with a digital image processing pipeline to get metrics from each image pixel corresponding to target contours. These metrics from color and morphological aspects feed a dataset which is manually labeled as cochineal (1) and non cochineal (0).

### *A. Feature Engineering: image pre-processing*

A total of 109 raw images representing different palm samples (labeled accessions) were acquired by the researcher Marcone César Mendonça das Chagas (EMPARN/Embrapa Algodão) from october to december 2023 during the experiment (Figures 2 and 3) using an iPhone 12 camera with $1512 \times 2016$ pixels resolution. Some of these images represent the same sample in different periods, since the objective is to understand the resistance to the target pest. Four acquisition sessions at oct,$29^{th}$, nov,$7^{th}$, nov,$12^{th}$ and dec,$26^{th}$ comprised the growing period from the egg to the second instar [4]. First, the images were captured under natural lighting, with a uniform blue background, the palm centered, and the presence of an identifying label. Subsequently, tests were conducted with different artificial light sources to improve the results. The first variation tested was yellow artificial lighting, which resulted in significantly lower performance, as the algorithm relies on information from the white spectral range to identify the pest, and the yellow hue compromised this distinction. Next, white artificial lighting was used, which yielded the best results, as it more clearly highlighted the whitish elements while maintaining the visual fidelity necessary for automatic recognition. Although the image database is still small, its variability naturally increases due to the characteristics of the host plant. During the growing cycle, the plant shoots may present morphological changes, such as curvature, wilting, or senescence, which contributes to greater diversity in the dataset. It is also worth noting that these insects do not follow a standard morphological development pattern, which in itself introduces significant variations.These input images were submitted to the OpenCV pipeline in Figure 5 and the output contours can be seen with a colored border in Figure 6. Besides OpenCV library, the software was developed using the Google Colab platform with the Python language 3.12.7 with NumPy, Matplotlib and common machine learning libraries Pandas, Scikit-learn, Imbalanced-learn and XGBoost.

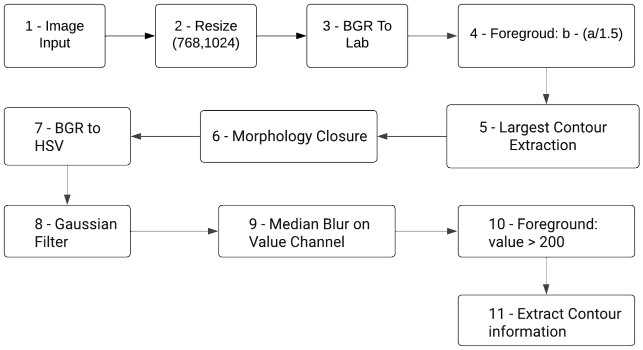
1 - Image Input
2 - Resize (768,1024)
3 - BGR To Lab
4 - Foregroud: b - (a/1.5)
5 - Largest Contour Extraction
6 - Morphology Closure
7 - BGR to HSV
8 - Gaussian Filter
9 - Median Blur on Value Channel
10 - Foreground: value > 200
11 - Extract Contour information

Fig. 5. Stages of the OpenCV digital image processing for feature engineering

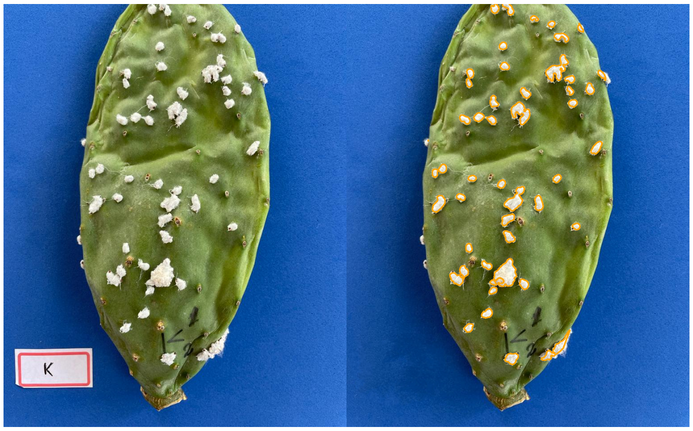

Fig. 6. (left) Image of an *Opuntia* sample (label K) with carmine cochineal under the white waxes and yellowish spines throughout its surface. (right) The pipeline output with detected contours. Source: Marcone César Mendonça das Chagas (EMPARN/Embrapa Algodão)

As a first result, the aggregation of these contours gives the approximate infestation area relative to the total surface area. For each of the 109 processed images, all detected contours were automatically cropped and thus it was generated 8149 images for getting contour information.

Besides measures of **aspect_ratio:** ratio between the width and height of the bounding rectangle of each object, **extent**: ratio between the contour area and the bounding rectangle area, **solidity**: ratio between the contour area and the convex hull area and **equivalent_diameter**: equivalent diameter of a circle with the same area, the last step in Figure 6 - extract contour information - retrieves for each contour the following color and morphological features (Table I):

TABLE I
DESCRIPTION OF FEATURES RETRIEVED FROM CONTOURS FOR COCHINEAL DETECTION

| Feature | Description |
|---|---|
| area | Area of each object |
| perimeter | Perimeter of each object |
| centroids_x | X-coordinate of the centroid of each object |
| centroids_y | Y-coordinate of the centroid of each object |
| r_mean | Mean value of the red (R) channel |
| r_min | Minimum value of the red (R) channel |
| r_max | Maximum value of the red (R) channel |
| r_std | Standard deviation of the red (R) channel |
| g_mean | Mean value of the green (G) channel |
| g_min | Minimum value of the green (G) channel |
| g_max | Maximum value of the green (G) channel |
| g_std | Standard deviation of the green (G) channel |
| b_mean | Mean value of the blue (B) channel |
| b_min | Minimum value of the blue (B) channel |
| b_max | Maximum value of the blue (B) channel |
| b_std | Standard deviation of the blue (B) channel |
| width | Width of the bounding rectangle |
| height | Height of the bounding rectangle |
| angle | Angle of the bounding rectangle |
| radius | Radius of the enclosing circle |

These features compose a dataset with $8149$ rows. Each row is manually labeled as cochineal or non cochineal, therefore constructing a dataset for a binary classification task. After dropping rows with $area = 0$, the unbalanced dataset has $class_0$ : $649$ samples and $class_1$ : $7235$ samples, which means that $91\%$ represent contours with the presence of cochineal. Here it is used the Synthetic Minority Oversampling Technique (SMOTE) for increasing the minority class and get a balanced scenario for the training phase, an usual solution for pest management systems [26]. With SMOTE the replacement of minority class instances is avoided in favor of the generation of synthetic examples. The described method randomly selects a nearest neighbor of a minority instance and linearly generates synthetic examples based on the original instance and a nearest neighbor.

## III. RESULTS AND DISCUSSION

Each of the six prospective models were subject to the pipeline in Figure 7, using its default hyperparameters. The dataset was divided into 70% for training set and 30% for the testing set. The training portion was used in a cross-validation process, in which, at each iteration, four folds were used for training and one fold for validation. It is noteworthy that the SMOTE is applied only to the train set and the validation set seen by the $10-$fold cross validation keeps its original distribution. When this methodology is neglected, the quality metrics may be biased. After, using only the training dataset, the best hyperparameters for each model were found with grid search. Finally, the models fitted with their best parameters were evaluated using the test dataset, which remained separate throughout the process. The ROC Curve and AUC are presented in Figure 8. The usual metrics accuracy (ACC), precision (P), recall (R), Matthews Correlation coefficient (MCC), $F_1$ and Area Under Curve (AUC) are reported in Table II.

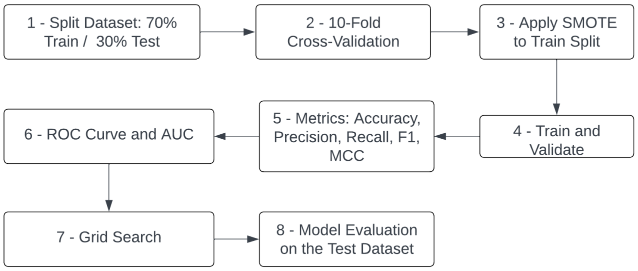
1 - Split Dataset: 70% Train / 30% Test
2 - 10-Fold Cross-Validation
3 - Apply SMOTE to Train Split
4 - Train and Validate
5 - Metrics: Accuracy, Precision, Recall, F1, MCC
6 - ROC Curve and AUC
7 - Grid Search
8 - Model Evaluation on the Test Dataset

Fig. 7. Pipeline for the selection of potential classification models for tuning and improvement

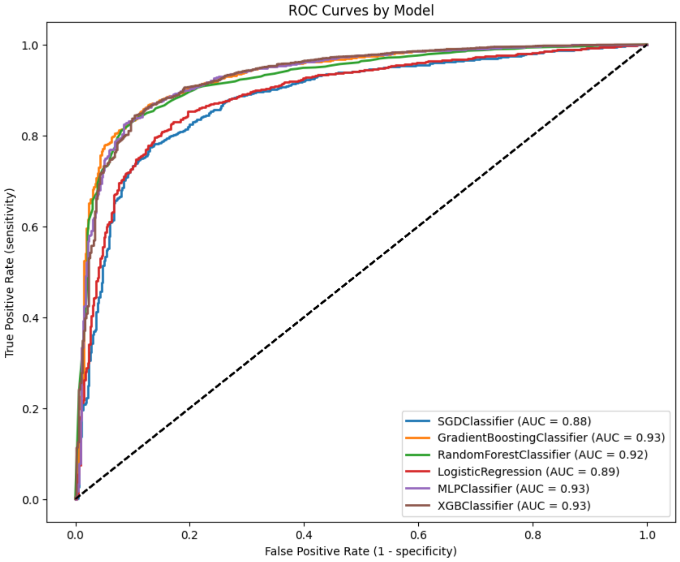

Fig. 8. ROC curves for six classification models

From Table II, four models were selected for fine tuning with grid search, based on accuracy, MCC, $F_1$, AUC and the better balance between precision and recall. Indeed, these metrics are better balanced for the XGBoost classifier, with a difference only in the third decimal place.

Running the grid search stage in Figure 7 for the classifiers Gradient Boosting, Random Forest, MLP and XGBoost and the accuracy as the score to be maximized, Table III presents the best parameters found and the improved ACC for all models (compare column ACC in Table II and Table III)

There is an expected similarity between XGBoost and Gradient Boosting. It is then necessary to evaluate each model on the test set without SMOTE, since it represents data not seen during training and validating phases and thus gives an intuition on the model capacity for generalization under a production software environment. These results are presented in Table IV. The difference between test ACC and average validation ACC confirms Gradient Boosting (GB) and XGBoost (XGB) as the best choices for this particular problem. How these models handle false positives and false negatives can be inferred from Table VI to Table VIII. Particularly for GB and

TABLE II
MODEL PERFORMANCE METRICS (MEAN VALUES)

| Model | ACC | P | R | MCC | $F_1$ | AUC |
|---|---|---|---|---|---|---|
| SGDClassifier | 0.84 | 0.98 | 0.80 | 0.44 | 0.89 | 0.88 |
| **Gradient Boosting** | 0.90 | 0.97 | 0.91 | 0.55 | 0.94 | 0.93 |
| **Random Forest** | 0.91 | 0.97 | 0.94 | 0.54 | 0.95 | 0.92 |
| Logistic Regression | 0.85 | 0.98 | 0.86 | 0.46 | 0.91 | 0.89 |
| **MLP** | 0.87 | 0.98 | 0.88 | 0.53 | 0.92 | 0.93 |
| **XGBoost** | 0.93 | 0.962 | 0.964 | 0.56 | 0.96 | 0.93 |

TABLE III
BEST HYPERPARAMETERS AND ACCURACY (ACC) FOR EACH SELECTED MODEL

| Model | Best Parameters | ACC |
|---|---|---|
| Random Forest | {max_depth: None, max_features: 'sqrt', min_samples_leaf: 1, min_samples_split: 2, n_estimators: 300} | 0.95 |
| MLP | {activation: 'relu', alpha: 0.0001, hidden_layer_sizes: (100, 50), learning_rate_init: 0.01, solver: 'adam'} | 0.93 |
| Gradient Boosting | {learning_rate: 0.1, max_depth: 5, min_samples_leaf: 2, min_samples_split: 5, n_estimators: 200, subsample: 0.8} | 0.96 |
| XGBoost | {colsample_bytree: 1, gamma: 0, learning_rate: 0.1, max_depth: 7, n_estimators: 200, subsample: 0.8} | 0.97 |

XGB, the difference is that XGB classifies less false negatives and increases the true positives (Table VIII).

TABLE IV
PERFORMANCE METRICS FOR EACH MODEL ON THE TEST SET

| Model | ACC | P (Precision) | R (Recall) | $F_1$ Score |
|---|---|---|---|---|
| Random Forest | 0.91 | 0.72 | 0.79 | 0.91 |
| MLP | 0.89 | 0.68 | 0.82 | 0.72 |
| Gradient Boosting | 0.93 | 0.76 | 0.78 | 0.77 |
| XGBoost | 0.93 | 0.77 | 0.78 | 0.78 |

TABLE V
CONFUSION MATRIX - RANDOM FOREST

| | Predict | |
|---|---|---|
| **True** | Negative | Positive |
| Negative | 120 | 67 |
| Positive | 136 | 2046 |

TABLE VI
CONFUSION MATRIX - MULTI LAYER PERCEPTRON (MLP)

| | Predict | |
|---|---|---|
| **True** | Negative | Positive |
| Negative | 140 | 47 |
| Positive | 225 | 1954 |

TABLE VII
CONFUSION MATRIX - GRADIENT BOOSTING

| | Predict | |
|---|---|---|
| **True** | Negative | Positive |
| Negative | 111 | 76 |
| Positive | 86 | 2093 |

TABLE VIII
CONFUSION MATRIX - XGBOOST

| | Predict | |
|---|---|---|
| **True** | Negative | Positive |
| Negative | 111 | 76 |
| Positive | 79 | 2100 |

## IV. CONCLUSIONS AND FUTURE WORKS

This work presented the problem of classifying the presence of carmine cochineal on forage palm (Opuntia sp.), in order to evaluate the spread of this agricultural pest in different accessions. After the construction of the dataset from actual images collected during experiments, it was evaluated six machine learning classification models, based on gradient with and without boosting techniques, a simple multilayer perceptron neural network and a tree ensemble model. The XGBoost and Gradient Boosting presented similar performance with good generalization properties when applied to the test set. The XGBoost is an optimized (and faster) version o Gradient Boosting, with parallel processing, tree-pruning and built-in regularization to avoid overfitting. The final accuracy on the train set (97%) and on the test set (93%) exceed the known percentage of 91% of the samples with cochineal at the original dataset, therefore aggregating value and confidence of using this model. Besides it provided the best balance between precision and recall. For this particular problem of identifying and counting a single pest species, there is no need of deep learning algorithms.

As future works this model will be embedded in a mobile application for fast classification and counting of the carmine cochineal during the experiments, since the images are directly collected from the smartphone of the expert. Another work is to consider not only static images but also video capture with prompt output to the user.

## V. ACKNOWLEDGEMENTS

The authors thank the Empresa de Pesquisa Agropecuária do Rio Grande do Norte (EMPARN) and Embrapa Algodão for the partnership, and Marcone César Mendonça das Chagas (EMPARN/Embrapa Algodão) for providing the images used in this work.

# References

[1] FAO. Status of cochineal and Opuntia spp. production in the Near East North Africa region 2022: a perspective from Jordan, Lebanon, Morocco, the Syrian Arab Republic and Tunisia. Rome, FAO, 2022, doi: https://doi.org/10.4060/cc3256en.

[2] G.M.O. Manzo, H.E.M. Flores, J.O.R. López and L.Portillo. Carmine red from cochineal (*Dactylopius coccus*), a natural dye: a review. Ciencia Nicolaita, n. 93, pp. 26–34, 2025.

[3] M.C. Batista, R. Nascimento, I.V.B. Almeida, L.T.V. Medeiros, J.T.A. Souza, J.P.O. Santos, P.H.A. Cartaxo and J.R.E.S. Araujo. Production and selection of accessions of Opuntia spp. with resistance to false carmine cochineal. Comunicata Scientiae, vol. 13: e3869, 2022, doi: https://doi.org/10.14295/CS.v13.3869.

[4] M.C.M. Chagas, E.C.S. Silva, S.M. Nascimento, G.F.C. Lima and T.C.C Lima. Cochonilha do carmim na palma forrageira: conheca a praga e as estrategias de controle. EMPARN, 2018.

[5] G.W. Teklu, K.M. Ayimut, F.A. Abera, Y.G. Egziabher, W. Gerima and I. Fitiwi. Evaluation of Opuntia species for their resistance to carmine cochineal (Dactylopius coccus) insect in Tigray, Northern Ethiopia. Haseltonia vol. 31, n. 1, pp. 80–88, 2024.

[6] A.S. Muhammad, S. Mandala and P.H. Gunawan. IOT-based pest detection in maize plants using machine learning. In: 2023 International Conference on Data Science and Its Applications, pp. 254–258, 2023.

[7] A. Kumar, F. Shaik, B. Yashwitha, P. Vidyavathi, A.U. Maheswari and P. Spurthi. Intelligence pest detection and control in agriculture using computer vision and deep learning. In: 2025 International Conference on Intelligent Computing and Control Systems pp. 1147–1150, 2025.

[8] D. Jani, M.I. Patel, R. Gajjar and N. Gajjar. Comparative analysis of machine learning models for pest detection on Raspberry Pi. In: 2023 7th International Conference on Trends in Electronics and Informatics, pp. 991–995, 2023.

[9] S.B. Vemulapalli, D.P.R. Pillagolla, G. Vempala and S.M. Ravuri. Image-based pest detection and identification system for agriculture. In: 2024 4th International Conference on Pervasive Computing and Social Networking, pp. 989–993, 2024.

[10] P. Ramadoss, V. Ananth, M. Navaneetha and U. Oviya. E-xpert bot-guidance and pest detection for smart agriculture using AI. In: IEEE 12th International Conference on Communication Systems and Network Technologies, pp. 797–802, 2023.

[11] P. Bhatt, A.K. Mishra and N. Tripathi. Real-time machine learning for precision agriculture: targeted weed detection and adaptive agrochemical application. In: International Conference on Cybernation and Computation, pp. 63–67, 2024.

[12] H. Alaa, K. Waleed, M.S.M. Tarek, H. Sobeah and M.A. Salam. An intelligent approach for detecting palm trees diseases using image processing and machine learning. International Journal of Advanced Computer Science and Applications, vol. 11, N. 7, pp. 434–441, 2020.

[13] R. Wayama, Y. Sasaki, S. Kagiwada, N. Iwasaki and H. Iyatomi. Investigation to answer three key questions concerning plant pest identification and development of a practical identification framework. Computers and Electronics in Agriculture, vol. 222: e109021, 2024, doi: https://doi.org/10.1016/j.compag.2024.109021.

[14] M. Khanramaki, E.A. Asli-Ardeh and E. Kozegar. Citrus pests classification using an ensemble of deep learning models. Computers and Electronics in Agriculture, vol. 186: e106192, 2021, doi: https://doi.org/10.1016/j.compag.2021.106192.

[15] M. Shoaib, A. Sadeghi-Niaraki, F. Ali, R. Hussain and S.K. Khalid. Leveraging deep learning for plant disease and pest detection: a comprehensive review and future directions. Front. Plant Sci., vol. 16: e1538163, 2025, doi: 10.3389/fpls.2025.1538163.

[16] P.L.O. Costa, T.M.O. Costa, L.F.R. Moreira, L.H.F. Pinto, J.F. Mari. Classification of agricultural pests through digital images using deep learning. Revista de Informática Teórica e Aplicada, vol. 32, n. 1, pp. 18–25, 2025.

[17] E.V. Madhuri, J.S. Rupali, S.P. Sharan, N.S. Pooja, G.S. Sujatha, D.P. Singh, K. Ahmad, A. Kumar and R. Prabha. Transforming pest management with artificial intelligence technologies: the future of crop protection. Journal of Crop Health, vol. 77:e48, 2025, doi: https://doi.org/10.1007/s10343-025-01109-9.

[18] M. Du, F. Wang, Y. Wang, K. Li, W. Hou, L. Liu, Y. He and Y. Wang. Improving long-tailed pest classification using diffusion model-based data augmentation. Computers and Electronics in Agriculture, vol. 234: e110244, 2025, doi: https://doi.org/10.1016/j.compag.2025.110244.

[19] D. Javale, A. Dere, S. Gavas, A. Teke and A. Khan. Dataset for agriculture pest images, 2024, doi: 10.21227/xmfd-k187.

[20] R. Wang, L. Liu, C. Xie, P. Yang, R. Li and M. Zhou. AgriPest: a large-scale domain-specific benchmark dataset for practical agricultural pest detection in the wild. Sensors, vol. 21: e1601, n. 5, 2021, doi: https://doi.org/10.3390/s21051601.

[21] A. Benali and I. Jdey. Cactus dataset, vol. 1 Mendeley Data, 2024, doi: 10.17632/jgt7ghdmg5.1.

[22] A. Berka, A. Hafiane, Y. Es-Saady, M. Hajji, R. Canals and R. Bouharround. CactiViT: Image-based smartphone application and transformer network for diagnosis of cactus cochineal. Artificial Intelligence in Agriculture, vol. 9, pp. 12–21, 2023.

[23] D. Wadhwa and K. Malik. A generalizable and interpretable model for early warning of pest-induced crop diseases using environmental data. Computers and Electronics in Agriculture, vol. 227: e109472, 2024, doi: https://doi.org/10.1016/j.compag.2024.109472.

[24] J.B. Oliveira, A.M.S. Araujo, L.R.L. Texeira, E.C.S. Silva, L.E.A.S. Santana, T.C. Rodrigues and M.C.M. Chagas. Uma Proposta de Sistema Embarcado para Contagem Automatizada de Cochonilhas de Escama em Laboratório. In: XV Simpósio Brasileiro de Automação Inteligente - SBAI, pp. 2063–2068, 2021.

[25] N. Daskalakis and J.B. Oliveira. Segmentação por Instâncias de Estágios de Desenvolvimento de Cochonilha de Escama com Mask R-CNN. In: 14. Congresso Brasileiro de Agroinformática, pp. 408–413, 2023.

[26] A. Longo, M. Rizzi and C. Guaragnella. Improving classification performance by addressing dataset imbalance: a case study for pest management. Applied Sciences, vol. 15: e5385, 2025, doi: https://doi.org/10.3390/app15105385.