

# Coffee Plant Leaf Disease Detection for Digital Agriculture

Laís Dib Albuquerque   [ Amazonas State University | [lda.snf20@uea.edu.br](mailto:lda.snf20@uea.edu.br) ]

Eloá B. Guedes  [ Amazonas State University | [ebgcosta@uea.edu.br](mailto:ebgcosta@uea.edu.br) ]

 Research Group on Intelligent Systems, Amazonas State University, Av. Darcy Vargas, 1200, Manaus, Amazonas

Received: 30 October 2023 • Accepted: 05 March 2024 • Published: 18 March 2024

**Abstract:** In an effort to advance Digital Agriculture, this paper provides a comparative assessment of Artificial Neural Networks for intelligent detection of a major biotic stress factors in coffee cultivation. Through a multi-class Computer Vision task, the superior performance of Convolutional Neural Networks, notably the ShuffleNet architecture, was discerned, further substantiated by statistical analyses. This model's performance, akin to state-of-the-art solutions, was achieved with reduced training data and parameter requirements. Robustness was affirmed through external validation using alternative datasets. This contribution directly enhances coffee plantations' quality and supports the development of Edge Computing devices for Agricultural IoT.

**Keywords:** Deep Learning, Digital Agriculture, Computer Vision, Artificial Neural Networks, Coffee

## 1 Introduction

Per the Food and Agriculture Organization (FAO) of the United Nations (UN), coffee stands as a highly consumed global beverage and one of the most traded commodities. Its market is expanding, driven by rising consumption in emerging economies, growing interest in specialty coffee, and product innovations in developed nations. Additionally, coffee plays a role in advancing Sustainable Development Goals by fostering income generation, rural employment, and poverty reduction [FAO, 2023]

Brazil holds the top position globally in the international coffee production market. In 2022, Brazil produced a total of 50,920.1 thousand sacks of coffee (each weighing 60 kg), marking a percentage increase of 6.7% compared to the previous year. This production volume constitutes approximately 32% of the global market share. The coffee cultivation area in Brazil for the year 2022 encompassed 2.24 Mi ha, with 64.78% of the productive area dedicated to the *Coffea arabica* species and the remainder to the *Coffea Canephora* variety, commonly referred to as robusta [CONAB, 2022].

Brazilian coffee production is acknowledged as one of the most socially and environmentally conscious in the world, with a commitment to ensuring sustainable coffee production. The high quality and diversity of coffee crops establish Brazil as a reliable supplier capable of meeting the demands of the most discerning buyers in both local and international markets [Brazil, 2022]. In the context of high-quality coffee production, a complex set of factors must be carefully considered, controlled, and monitored. Among these factors, temperature, soil density, rainfall patterns, wind exposure, humidity, as well as the presence of pests and diseases stand out. Regarding the latter two factors, the importance of precise and early diagnosis is emphasized, with the aim of facilitating effective and efficient decision-making that minimizes environmental and economic impacts. The incidence of fungi, bacteria, nematodes, and viruses can reduce production up to 20%, making timely intervention crucial [Mesquita *et al.*, 2016].

Taking into consideration the demands of the agricultural

sector, Digital Agriculture has been significantly gaining ground. It is characterized by the use and development of technological solutions within the field of Informatics, aimed at enhancing production and achieving higher quality and productivity [Tang *et al.*, 2002]. In Brazil, in particular, Image Processing and Computer Vision (CV) techniques are now explicitly recognized as integral components of the strategic axes to be developed and implemented over the next two decades. [Embrapa, 2014].

Recently, Machine Learning (ML), a subfield of Artificial Intelligence that encompasses the development and utilization of computational models and methods capable of learning from data and making inferences for previously unseen instances [Faceli *et al.*, 2021], has substantially advanced the state of the art in Computer Vision (CV) problems [Khan *et al.*, 2018]. These results have a positive impact on the development of solutions for Agriculture, such as pest identification, soil type determination, plant recognition, automatic seed inspection, fruit counting, among others [Kamilaris and Prenafeta-Boldú, 2018]. The ML applications in this context directly contribute to the objectives of ensuring global food security, given the perspective of up to a 70% increase in food production demand by 2050 due to rapid urbanization, population growth, reduced available planting space, among other factors [Sharma *et al.*, 2021].

With the aim of contributing to the body of intelligent solutions for Digital Agriculture, particularly considering the role of coffee for Brazil in this context, the objective of this paper is to showcase the experimental results derived from the implementation of two Artificial Neural Networks (ANNs) approaches addressing the problem at hand, which is conceptualized as a CV task. It is an extension of previous work on the matter [Albuquerque and Guedes, 2023]. The analysis of the results aligns with the current state of the art, highlighting the superior performance of Deep Learning (DL) techniques in comparison to the traditional approach of feature extraction followed by classification using Feedforward Multilayer Perceptron (MLP) ANNs. The main contributions of this work are:

1. A comparative analysis of two approaches utilizing ANNs for the intelligent detection of coffee plant leaf diseases, taking into account a prevalent biotic stress factors impacting crop health;
2. An experimental design prioritizing reproducibility through the utilization of realistic, varied and expert-labeled experimental data, enabling assessment of the performance of ANNs;
3. The identification of the most suitable ANNs for the specific task based on statistical tests, which were carried out through repeated experiments to reduce the impact of stochastic fluctuations;
4. The external validation of the proposed model across three supplementary datasets to assess its generalization and robustness.

The remainder of this paper is organized as follows: Section 2 provides an analysis of related works available in the literature for the same problem at hand, specifically those employing ANNs in the proposed solution, highlighting recent advancements in the state of the art; Materials and methods employed are expounded upon in Section 3, encompassing details regarding experimental data, approaches, models, parametrization, experimental setup, model selection metrics, and evaluation; Section 4 presents the obtained results, along with a comparative analysis and their correlation with recent literature; An extension of the validation of the proposed solution is conducted in Section 5, wherein three additional datasets are considered; Lastly, concluding remarks are presented in Section 6.

## 2 Related Work

The exploration of pertinent literature involved an examination of Google Scholar outcomes spanning from the year 2021 to 2023. The inclusion criteria focused on studies utilizing methods and techniques rooted in Computer Vision and/or Artificial Intelligence for the automated diagnosis of diseases affecting coffee plant leaves. The overarching goal was to acquire an exhaustive comprehension of the subject matter, elucidate the current state-of-the-art methodologies, as well as highlighting gaps and trends in addressing the problem under consideration.

The work by Dias and Saito [2021] identified coffee leaf anomalies using the JSEG algorithm, a non-supervised segmentation method based on textures and color regions. The experimental data came from *Robusta Coffee Leaf images dataset* (RoCoLe dataset), comprising healthy and unhealthy samples, where the latter case is divided into four levels of rust and red mite presence [Parraga-Alava *et al.*, 2019]. Experiments carried out with 50 samples (3 % of available data) were mainly focused on finding best parameters for JSEG using different scales. They concluded that smaller scales, such as  $9 \times 9$  px and  $17 \times 17$  px, are most suited for the problem because their homogeneity favours classification. However, that work does not show the experimental results, although indicating that further research with MLP ANNs is encouraged.

Işık and Eskicioglu [2022] considered the classification of unhealthy-only coffee leaves using CNNs, a specialized kind

of ANNs for processing high-dimensional data, employing convolutional filters to integrate spatial context, facilitating the extraction of discriminative features [Goodfellow *et al.*, 2016]. The experimental dataset, comprising 542 samples collected under controlled conditions from a Kaggle dataset, exhibited a distribution of 47.41 % affected by rust and the remainder by mine infestation. The authors systematically investigated six distinct filtering processes, namely RGB filtering, Histogram Equalization, Contrast Limited Adaptive Histogram Equalization, Gaussian Blur, Morphology Close and Morphology Gradient, as pre-processing steps for CNN input. Their rationale was grounded in the potential of these techniques to enhance Deep Learning model performance, obviating the necessity for additional procedures like data augmentation. The experimental framework encompassed an 80 %/20 % holdout cross-validation strategy, employing eight diverse CNN architectures (VGG-16, Xception, InceptionV3, DenseNet121, DenseNet169, AlexNet, MobileNetV2 and ResNet50). Hyper-parameter exploration considered batch size and optimizer settings, with each model undergoing training for 30 epochs in each run. Analysis of the results culminated in the authors' conclusion that the evaluated CNNs exhibited substantial success, with an average classification accuracy exceeding 95 % in the context of coffee leaf classification.

Taking into account four major diseases that affect coffee crops, Novtahaning *et al.* [2022] proposed an ensemble method designed to enhance detection accuracy. Their approach hinged on the utilization of three well-established CNN architectures (VGG-16, EfficientNetB0, and ResNet152) which were pre-trained with ImageNet dataset weights; experimental data from the literature comprising 1300 examples (260 for each disease and also for a healthy class) [Esgario *et al.*, 2020a]; image pre-processing and data augmentation techniques; and a bagging-based voting strategy. After carrying out a 80 %/20 % holdout cross-validation strategy, their ensemble approach boasted an accuracy rate of 97.3 % and a  $F_1$ -Score of 95.1 %. Analysis of performance metrics suggests that the proposed model achieves state-of-the-art results for the addressed problem, demonstrating significant contributions to the field. However, this effectiveness comes at the cost of a high number of parameters, exceeding 76 Mi. A key trade-off exists in CNNs between model complexity and computational efficiency. While an increased number of parameters often correlates with improved performance and accuracy, it concomitantly leads to extended training and inference times, alongside escalating energy consumption. This presents a significant challenge for real-world applications, directly impacting hardware requirements and deployment feasibility.

The study by AUFAR *et al.* [2023] utilized JMuBEN and JMuBEN2 datasets which contains 58 550 samples from four diseases and also from healthy leaves [Jepkoech *et al.*, 2021]. The authors approached the problem as a multi-class classification task, partitioning the dataset into 10 subsets and conducting a holdout cross-validation on each subset, with 80 % of the data for training, 10 % for training validation, and 10 % for testing purposes. They employed the CNN architectures ResNet50, InceptionResNetV2, MobileNetV2, and DenseNet169. The results obtained revealed an experimen-

tal accuracy of 100% for the InceptionResNetV2 (55.9 Mi parameters, 449 layers) and for DenseNet169 (14.3 Mi parameters, 338 layers). However, it is noteworthy that the authors did not provide measures of dispersion in their experimental assessments.

In reviewing the body of literature related to the subject matter, some observations emerge: (i) a range of publicly available datasets exists for addressing the coffee leaf disease classification problem. However, they exhibit significant variability in terms of the quantity of images and the representation of diseases; (ii) CNNs have emerged as the predominant approach for coffee leaf disease classification, consistently yielding classification accuracy rates over 95%; (iii) a comprehensive statistical evaluation of experimental outcomes is notably absent from the literature; and (iv) the computational cost regarding the number of CNNs parameters was not discussed as a potential drawback since it may impose practical limitations when deploying such models on resource-constrained platforms, such as mobile devices or Unmanned Aerial Vehicles (UAVs). Given the growing interest in Edge Computing and mobile applications for Digital Agriculture, understanding and addressing these computational constraints becomes imperative.

### 3 Material and Methods

The proposed solution in this work aims to address the classification of coffee leaf diseases from images as a multi-class classification task through Supervised Learning. The experimental data utilized, the approaches for model preparation and parameterization, the computational environment employed for conducting experiments, as well as the selection and evaluation of models, are detailed in the following subsections.

#### 3.1 Experimental Data: Overview and Preparation

The experimental data utilized in this study originates from the JMuBEN and JMuBEN2 datasets, comprising a total of 58,550 images depicting *Coffea arabica* leaves collected in the Kenyan region employing a digital camera. The labeling of these images was performed by a qualified pathologist and encompasses five distinct classes: (i) ‘Healthy,’ representing leaves exhibiting no discernible pathological features; (ii) ‘Cescospora,’ which denotes a fungal disease caused by *Cercospora coffeicola*. The presence of Cescospora is identifiable through the observation of circular grey spots with tan or white centers on the leaf surface; (iii) ‘Rust,’ caused by the fungus *Hemileia vastatrix*, is characterized by the manifestation of chlorotic patches on the upper leaf surface, concomitant with the development of rust pustules on the underside of the leaf; (iv) ‘Phoma,’ induced by the fungus *Phoma costaricensis*, manifests in leaves that undergo a progressive browning and withering process, starting from the leaf tip and extending towards its periphery; and finally, (v) ‘Miner,’ resulting from the activity of larvae belonging to *Leucoptera coffeella*, is typified by the presence of distinctive yellow trails beneath the epidermal layer of the coffee leaf [Jepkoech *et al.*,

2021]. It is imperative to underscore that each class is characterized by visually distinguishable attributes, as depicted in Figure 1.

The image data underwent a series of pre-processing steps conducted by the dataset providers, encompassing: (i) the application of noise filtering and contrast stretching techniques, strategically employed to enhance overall image quality; (ii) a cropping operation, performed to isolate the central square portion of each image, with the explicit objective of accentuating the region of interest within the leaf specimen; and (iii) the implementation of data augmentation methodologies, including rotations (180° counterclockwise) and flipping (horizontal and vertical) [Jepkoech *et al.*, 2021]. Notably, with respect to data augmentation, the authors did not disclose either the initial quantity of images preceding the application of these techniques or any specific filename conventions employed to distinguish between original and artificially generated images. Upon examining the distribution of instances across classes, as delineated in the histogram presented in Figure 2, it is important to draw attention to the inherent class imbalance within the dataset. This observation has prompted the utilization of performance metrics tailored to accommodate this specific data imbalance scenario.

The experimental data described was used in experiments with no further data augmentation than that already provided by authors and all images were resized to 128 × 128 px.

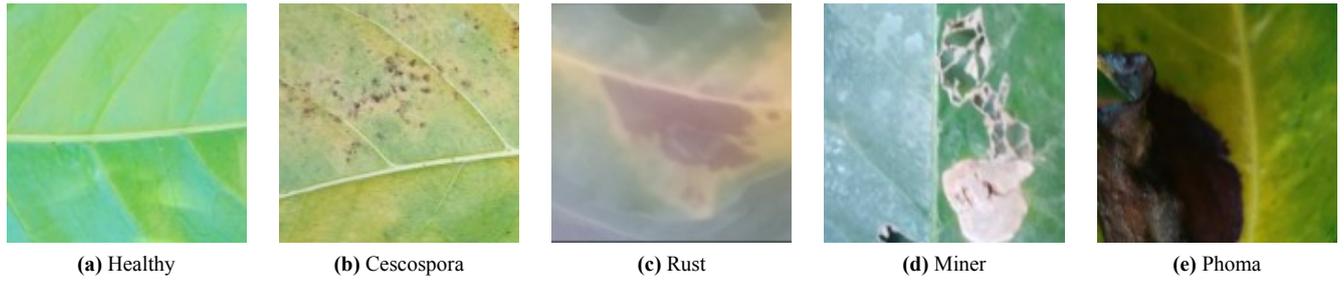
#### 3.2 Approaches, Models and Parametrization

ANNs served as the primary Machine Learning model of focus in this work. This choice was predicated upon their massively parallel distributed architecture, their capacity for learning and therefore generalization, as well as their capabilities in handling nonlinearity, adaptivity, and fault tolerance [Haykin, 2008]. Within the framework of this model, we have explored two distinct approaches: a traditional approach rooted in Computer Vision feature extraction methods, and a contemporary approach following the recent advancements in DL.

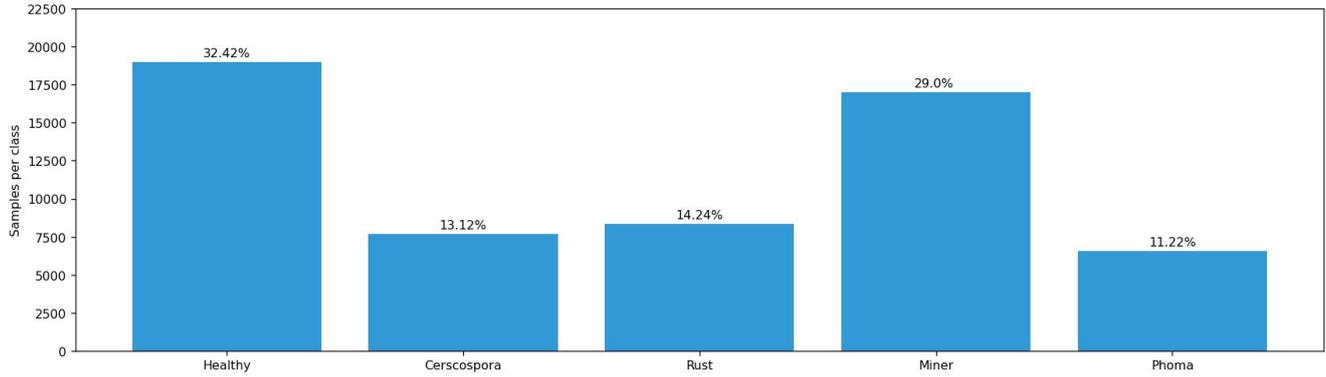
##### 3.2.1 Traditional Approach

This study was grounded in the pipeline of Computer Vision methods, wherein the initial step involves the extraction of image features, which are subsequently employed to train ML algorithms [Prince, 2012].

Drawing from pertinent work within the realm of classification in Agriculture [Santos *et al.*, 2019; VijayaLakshmi and Mohan, 2016; Rehman *et al.*, 2021; Xian and Ngadiran, 2021], the first step was performed with Haralick Textural Features which is defined as a set of statistical measures used to characterize the texture or spatial patterns within an image [Haralick *et al.*, 1973]. Such features are computed by analyzing the Gray-Level Co-occurrence Matrix (GLCM) of an image, a two-dimensional matrix in which each element  $P(i, j)$  represents the frequency of occurrence of a pair of pixels (where  $i$  and  $j$  are the gray levels) in a spatial relation separated by distance  $\delta$  and angle  $\alpha$ . Let  $G$  be an image texture with the size of  $M \times N$ . Each element can be calculated by counting the number of relationships with the following



**Figure 1.** Randomly drawn examples from each class from JMuBEN and JMuBEN2 datasets.  
Source: [Jepkoech *et al.*, 2021].



**Figure 2.** Distribution of instances per class in the JMuBEN and JMuBEN2 datasets.

equation:

$$P(i, j) = \{G(m, n) = i, G(m + \delta, n + \delta) = j\} \text{ for each } \alpha, \quad (1)$$

where  $m = 0, 1, \dots, M - 1$ ,  $n = 0, 1, \dots, N - 1$  and  $\alpha = 0^\circ, \dots, 360^\circ$ . Consider the following notation as hereby established:

- $p(i, j)$ :  $(i, j)$ -th entry in a normalized gray-tone spatial-dependence matrix, i.e.,  $P(i, j) / \sum_i \sum_j P(i, j)$ ;
- $p_x(i)$ :  $i$ -th entry in the marginal-probability matrix obtained by summing the rows of  $p(i, j)$ , i.e.,  $\sum_j P(i, j)$ ;
- $N_g$ : Number of distinct gray levels in the quantized image;
- $p_y(j) = \sum_i p(i, j)$ ;
- $p_{x+y}(k) = \sum_i \sum_{j: i+j=k} p(i, j)$ ,  $k = 2, 3, \dots, 2N_g$ . This term represents the joint probability of occurrence of pixel intensity values  $x$  and  $y$  at a certain spatial relationship within an image.;
- $p_{x-y}(k) = \sum_i \sum_{j: |i-j|=k} p(i, j)$ ,  $k = 0, 1, \dots, N_g - 1$ . It aims to characterize the joint probability of encountering pixel pairs with a particular intensity difference  $(x - y)$  within a given spatial relationship.;
- $\mu_x, \mu_y, \sigma_x$  e  $\sigma_y$  are the means and standard deviation of  $p_x$  and  $p_y$ ;

From the GLCM, various statistical measurements, rooted in Information Theory, are derived to describe the texture properties of the image. In the scope of this work, the following Haralick features were used. Some intuition behind them is provided:

1. **Energy (Angular Second Moment).** Measures the local homogeneity or uniformity of the image texture.

Higher values indicate smoother textures.

$$f_1 = \sum_i \sum_j \{p(i, j)\}^2; \quad (2)$$

2. **Contrast.** Measures the local variation in pixel intensities. Higher values indicate greater contrast between neighboring pixels.

$$f_2 = \sum_{n=0}^{N_g-1} n^2 \left\{ \sum_i \sum_{j: |i-j|=n} p(i, j) \right\}; \quad (3)$$

3. **Correlation.** Describes the linear dependency between pixel pairs. It can be an indicator of how repetitive or periodic a texture is.

$$f_3 = \frac{\sum_i \sum_j (ij) p_{ij} - \mu_x \mu_y}{\sigma_x \sigma_y}; \quad (4)$$

4. **Variance.** Provides information about the degree of variation in pixel pair values within a particular spatial relationship in the image.

$$f_4 = \sum_i \sum_j (i - \mu)^2 p(i, j); \quad (5)$$

5. **Inverse Difference Moment.** Measures the local homogeneity or closeness of pixel intensities in the specified spatial relationship. It is an indicator of how uniform or regular the texture appears in the image.

$$f_5 = \sum_i \sum_j \frac{1}{1 + (i - j)^2} p(i, j); \quad (6)$$

6. **Sum Average.** Characterizes the average sum of the gray-level values of the pixel pairs in the GLCM, offering insights into the overall brightness or intensity characteristics of the texture.

$$f_6 = \sum_{i=2}^{2N_g} i p_{x+y}(i); \quad (7)$$

7. **Sum Variance.** Characterizes the variance or spread in the product of the row and column indices of pixel pairs within the specified spatial relationship in the image. It provides information about the variability of the products of gray levels, taking into account their spatial distribution. A higher value indicates that the products of gray levels in the texture tend to vary more widely or are more dispersed.

$$f_7 = \sum_{i=2}^{2N_g} (1 - f_8)^2 p_{x+y}(i); \quad (8)$$

8. **Sum Entropy.** Measures the randomness or disorder in the distribution of the sums of gray levels of pixel pairs within the specified spatial relationship in the image. It provides information about the complexity or irregularity of the texture pattern.

$$f_8 = - \sum_{i=2}^{2N_g} p_{x+y}(i) \log\{p_{x+y}(i)\}; \quad (9)$$

9. **Entropy.** Quantifies the randomness or disorder of texture. Higher entropy values suggest more complex or irregular textures.

$$f_9 = - \sum_i \sum_j p(i, j) \log(p(i, j)); \quad (10)$$

10. **Difference Variance.** Measures how much the gray-level differences between neighboring pixels vary within the image. A higher value indicates that these differences are more variable or dispersed, suggesting a more complex or heterogeneous texture.

$$f_{10} = \text{variance of } p_{x-y}; \quad (11)$$

11. **Difference Entropy.** Measures how unpredictable or irregular the differences between neighboring pixel intensities are in the image. A higher value indicates that these intensity differences are more random or diverse, suggesting a more complex or irregular texture.

$$f_{11} = \sum_{i=0}^{N_g-1} p_{x-y}(i) \log\{p_{x-y}(i)\} \quad (12)$$

12. **Information Measures of Correlation.** It measures the similarity of the joint probability distribution of pixel pairs to a product of their marginal distributions. High values indicate stronger correlation between pixel pairs in the GLCM.

$$f_{12} = \frac{f_9 - HXY1}{\max(HX, HY)}, \quad (13)$$

$$f_{13} = (1 - \exp[-2(HXY2 - f_9)]^{\frac{1}{2}}), \quad (14)$$

where  $HX$  is the entropy of  $p_x$ ,  $HY$  is the entropy of  $p_y$ ,  $HXY = - \sum_i \sum_j p(i, j) \log(p(i, j))$ ,  $HXY1 = - \sum_i \sum_j p(i, j) \log\{p_x(i)p_y(j)\}$  and  $HXY2 = - \sum_i \sum_j p_x(i)p_y(j) \log\{p_x(i)p_y(j)\}$  [Hung *et al.*, 2019; Mirjalili and Hardeberg, 2022; Haralick *et al.*, 1973, Appendix A].

Haralick features, a cornerstone in texture analysis, draw heavily upon principles established by Information Theory which studies the transmission, processing, extraction, and utilization of information [Cover and Thomas, 2006]. The effectiveness of Information Theory in solving various CV and pattern recognition problems, including image matching, clustering, segmentation, saliency detection, and feature selection, is well-documented. The connection between Haralick features and Information Theory, emphasizes how quantitative measures derived from the GLCM offer valuable insights into the spatial distribution and relationships of pixel intensities within an image. These insights pave the way for effectively characterizing and analyzing the intricate texture patterns inherent in digital images [Escolano *et al.*, 2009].

Following the extraction of Haralick features, each input image was represented by 13 numerical attributes, subsequently employed for training MLP ANNs. The choice of this ML model stemmed from its aptitude for generalization within the Supervised Learning Paradigm, its effectiveness in handling non-linear separable problems, and its inherent resilience to data noise [Haykin, 2008].

The task of identifying the optimal MLP architecture for a specific scenario remains a prominent unsolved challenge in the field of Machine Learning. Recognizing this open problem, we chose to employ the Geometric Pyramid rule-of-thumb, a well-established heuristic documented in the literature, that offers a practical alternative to the computationally expensive and time-consuming process of exhaustive architecture search [Palit and Popovic, 2005]. Taking into account that  $N_i = 13$  is the number of input nodes,  $N_o = 5$  is the number of output nodes, corresponding to the number of classes of the classification task, the goal is to obtain  $N_h$ , the number of hidden neurons, according to the following assignment:

$$N_h = \alpha \left[ \sqrt{N_i N_o} \right] = \alpha \left[ \sqrt{13 \cdot 5} \right] = 8\alpha. \quad (15)$$

Let  $0.5 \leq \alpha \leq 2$  and  $\lceil \cdot \rceil$  denote the closest integer function, then:

$$4 \leq N_h \leq 16. \quad (16)$$

We began by evaluating single-hidden-layer MLPs, where  $(i)$  represents an architecture with  $i$  hidden neurons. Based on the obtained values for  $N_h$ , we selected 12 configurations ranging from 4 to 16 hidden neurons for further analysis. Acknowledging the limitations of single-hidden-layer MLPs, we also explored double hidden-layer architectures. Here,  $(i, j)$  denotes an MLP with  $i$  neurons in the first hidden layer and  $j$  neurons in the second. Utilizing a random sampling approach, we selected 4 different values for  $N_h$  within the previously investigated range to create these double-hidden-layer configurations. This resulted in a total of 49 MLP architectures. For all of them, we employed the ReLU (Rectified

Linear Unit) activation function and trained them using the Adam optimizer [Kingma and Ba, 2015] for 300 epochs.

### 3.2.2 Contemporary Approach

This approach was grounded in recent advancements in DL wherein multidimensional images serve as input to CNNs. These networks autonomously perform large-scale hierarchical learning of feature extractor parameters, enabling them to proficiently classify intricate patterns within the input data [Goodfellow *et al.*, 2016]. Noteworthy advantages of this approach encompass the elimination of human intervention in the feature extraction or selection process and the availability of multiple canonical CNN architectures tailored to the realm of CV [Khan *et al.*, 2018].

For the following considered CNN architectures listed below, the images were resized to  $128 \times 128$  pixels, and the values of the three color channels were normalized.

1. **MobileNetV2**. It is regarded as a lightweight CNN, designed for deployment on mobile and embedded devices. Its simplified architecture employs depthwise separable convolutions in the initial layers to reduce computational overhead [Howard *et al.*, 2018];
2. **ShuffleNet**. Also designed specifically for mobile devices with limited computational power, it employs group convolutions, where each of the multiple convolutions covers a portion of the input channels, and channel shuffling, randomly mixing the output channels of group convolutions. This strategy significantly reduces computational cost without sacrificing performance [Zhang *et al.*, 2018];
3. **VGG-16**. One of the most popular CNN models due to its simplicity and performance in large-scale object classification tasks in Computer Vision. It consists of 16 layers in depth, employs  $3 \times 3$  convolutional filters for feature extraction, incorporates max-pooling layers for dimensionality reduction, and includes fully connected final layers for classification. The use of smaller filters results in fewer parameters and facilitates parallelization for hardware acceleration during training [Simonyan and Zisserman, 2014];
4. **InceptionV3**. With the convolutional layers processed in parallel, this architecture enables information processing along the depth dimension of the input. The considered version employs convolution factorization to reduce the number of parameters without compromising efficiency in Computer Vision tasks [Szegedy *et al.*, 2015];
5. **EfficientNetV2S**. With a parameter count similar to InceptionV3, it represents a contemporary architecture that strikes a balance between depth and the number of parameters, simultaneously optimizing training speed and classification efficiency [Tan and Le, 2019];

For all selected CNN architectures, the hyperparameters were employed based on the following criteria: (i) weights were initialized randomly, without transfer learning or leveraging weights from another task; (ii) training considered a maximum of 300 epochs; (iii) the Early Stopping technique

was adopted with a patience of 30 epochs to prevent overfitting, monitoring metrics on the validation set; (iv) the Model Checkpoint technique was used to monitor accuracy on the validation set and save to disk the weight set that provided the best generalization; (v) the initial learning rate was set to  $10^{-4}$ ; (vi) the ReLU activation function was used; (vii) the Adam optimizer was employed [Kingma and Ba, 2015]; and (viii) the choice of the batch size hyperparameter was determined empirically for each architecture, balancing the number of parameters and the utilization of available computational resources, including main memory. The dense and final layers of all mentioned architectures were resized to accommodate a number of neurons compatible with the problem's classes.

### 3.3 Experimental Setup

Python, in conjunction with the Keras, TensorFlow, and Sci-Kit Learn frameworks, served as the primary tools for training and evaluating the proposed models [Van Rossum and Drake, 2009; Chollet *et al.*, 2015; Abadi *et al.*, 2015; Pedregosa *et al.*, 2011]. The Mahotas framework, specifically, was employed to perform the Haralick features extraction [Coelho, 2013]. Implementations were executed on a computational system equipped with an Intel® Core™ i5-7400 CPU running at a clock speed of 3 GHz, supported by 24 GB of primary memory, 2 TB of secondary memory and 2 NVIDIA GTX 1650 GPUs with 4 GB VRAM each to promote hardware speedup when training CNNs.

### 3.4 Model Selection and Evaluation

To assess the selected models' performance, three distinct repetitions of holdout cross-validation were employed. In each repetition, 60% of the available data were allocated for training, 10% for validation, and 30% for testing, with the latter partition used for performance evaluation. The mean values of the following performance metrics across the three repetitions summarized this assessment, where  $C$  represents the set of classes in the problem.

$$\text{Accuracy} = \frac{1}{|C|} \sum_{c \in C} \left( \frac{\text{TP}_c + \text{FP}_c}{\text{TP}_c + \text{FP}_c + \text{FN}_c + \text{FP}_c} \right) \quad (17)$$

$$\text{Precision} = \frac{1}{|C|} \sum_{c \in C} \left( \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c} \right) \quad (18)$$

$$\text{Recall} = \frac{1}{|C|} \sum_{c \in C} \left( \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c} \right) \quad (19)$$

$$F_1\text{-Score} = 2 \left( \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \right) \quad (20)$$

In Eqs. (17)-(20), the acronyms denote the four potential outcomes of a binary classification task, namely: TP (True Positive) represents the count of correct classifications for the positive class; TN (True Negative) corresponds to the correct classifications for the negative class; FP (False Positive) signifies Type I Errors; and FN (False Negative) indicates Type II Errors.

Neural network training exhibits inherent stochasticity due to factors like weight initialization and batch composition, necessitating multiple experimental repetitions to mitigate expected fluctuations. Balancing statistical rigor with computational feasibility, we conducted 3 repetitions in our experiments. While this may not entirely eliminate stochastic effects, it acknowledges their presence and incorporates their potential influence into the results. Consequently, the metrics presented in Eqs. (17)-(20) will be reported in terms of both average and standard deviation. Moreover, as depicted in Figure 3, this Venn diagram elucidates the extent of sample overlap observed across the experimental repetitions. Notably, it underscores the diverse conditions in which these experiments were executed, revealing that approximately 13.95 % of the samples consistently resided within the test partition across all experiments.



Figure 3. Venn diagram for samples in experimental repetitions.

In the traditional approach, MLPs were ranked based on their average  $F_1$ -Score, and the top 5 architectures will be presented and discussed. The number of selected models matches the quantity of CNNs evaluated in the contemporary approach.

For the best performing models, regardless of their approach, statistical tests will be conducted to ascertain whether the samples comes from the same distribution, i.e., the models would be deemed equivalent for the proposed classification task under the  $F_1$ -Score. In the absence of ensuring that the data followed a normal distribution, we resorted to the use of the non-parametric Kruskal-Wallis  $H$  test, with a confidence level of 95 % ( $\alpha = 0.05$ ), to test the null hypothesis ( $H_0$ ) that these models are equivalent against the alternative hypothesis ( $H_A$ ) suggesting otherwise [Walpole *et al.*, 2012].

In addition to evaluating task performance, data on the number of parameters, training time and epochs of all models on the experiments were gathered. For the CNNs, measurement of the maximum Giga Floating-Point Operations Per Second (GFLOPS) over the repetitions was conducted. These metrics were examined with the aim of quantifying the computational processing power necessary for these mod-

els, enabling further evaluations related to emergent research fields, such as Green AI [Schwartz *et al.*, 2020] and Edge Computing [Cao *et al.*, 2020].

## 4 Results and Discussion

The computational experiments were conducted following the proposed methodology, and the summary of the results is presented in Tables 1 and 2. These tables represent the performance of the top 5 MLPs and CNNs in terms of the average and standard deviation across 3 repetitions.

By listing the  $F_1$ -Score as the reference performance metric, it is noteworthy that the traditional approach falls short of the contemporary approach, as depicted in Figure 4. This observation holds particular practical significance, as the former heavily relies on human intervention and expertise for feature extraction and architecture design. When examining the results obtained with the contemporary approach, it is first noteworthy to highlight the performance degradation of MobileNetV2 when compared to the results reported in the work by Aufar *et al.* [2023]. These results are indicative of underfitting for this model with respect to the learning task. This observation suggests that the model's effectiveness is heavily influenced by the availability of extensive training data and might call for targeted fine-tuning efforts to ensure consistent performance in dealing with specific tasks. These considerations may pose challenges when applying the model to broader tasks within the Digital Agriculture domain.

Aside from the MobileNetV2, the other architectures in the contemporary approach achieved an average  $F_1$ -Score exceeding 99.8 %. All of them made use of early stopping technique during training, requiring fewer than 300 epochs to achieve convergence, as shown in Table 3. Given the remarkably similar performance exhibited by all the CNNs within the contemporary approach during the experiments, a non-parametric Kruskal-Wallis  $H$  test was carried out following the conditions outlined in Section 3.4. The outcome of this analysis yielded a  $p$ -value of 0.1012, which does not fall below the significance threshold of  $\alpha = 0.05$  ( $p \not< \alpha$ ), indicating that rejecting the null hypothesis is not feasible. Hence, all the CNNs assessed, with the exception of MobileNetV2, demonstrate comparable performance in this specific task. In comparison to the results obtained by Aufar *et al.* [2023], equivalent performance was achieved, but with smaller architectures. The InceptionV3 and ShuffleNet models, for example, have 57.24 % and 97.54 % fewer parameters, respectively, than InceptionResNetV2. While InceptionV3 has more parameters than DenseNet169, it has lower depth, which may impact training time. ShuffleNet, in particular, is smaller than DenseNet169 in both dimensions. In comparison to this related work, all CNNs were trained with a notably reduced amount of data, resulting in a data reduction of approximately 33.50 %.

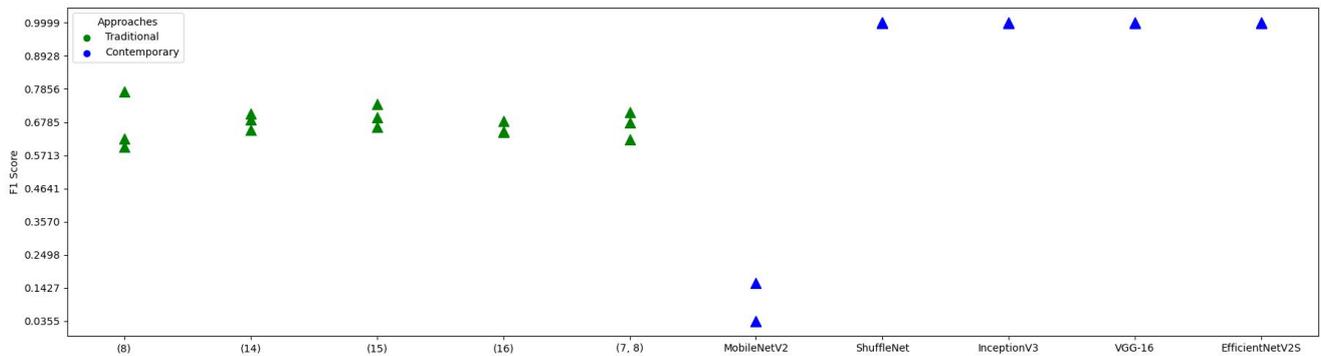
In the endeavor to distinguish among the examined CNNs, ShuffleNet proved to be the most fitting choice for serving as the reference solution for the problem investigated in this study. The confusion matrices for the three rounds of testing are depicted in Fig 5. This decision was driven by its lower

**Table 1.** Experimental results for the traditional approach.

Architectures	Precision	Recall	F <sub>1</sub> -Score	Accuracy
(8)	0.6686 ± 0.1056	0.7223 ± 0.0619	0.6672 ± 0.0966	0.7223 ± 0.0619
(14)	0.7127 ± 0.0757	0.7329 ± 0.0235	0.6823 ± 0.0270	0.7329 ± 0.0235
(15)	0.6927 ± 0.0421	0.7280 ± 0.0258	0.6984 ± 0.0367	0.7280 ± 0.0258
(16)	0.6828 ± 0.0370	0.7236 ± 0.0142	0.6589 ± 0.0197	0.7236 ± 0.0142
(7, 8)	0.6758 ± 0.0793	0.7168 ± 0.0407	0.6700 ± 0.0435	0.7168 ± 0.0407

**Table 2.** Experimental results for the contemporary approach.

Architectures	Precision	Recall	F <sub>1</sub> -Score	Accuracy
MobileNetV2	0.0486 ± 0.0400	0.2030 ± 0.0857	0.0766 ± 0.0581	0.2030 ± 0.0857
ShuffleNet	0.9995 ± 0.0003	0.9995 ± 0.0003	0.9995 ± 0.0003	0.9995 ± 0.0003
InceptionV3	0.9990 ± 0.0005	0.9990 ± 0.0005	0.9990 ± 0.0005	0.9990 ± 0.0005
VGG-16	0.9987 ± 0.0009	0.9987 ± 0.0009	0.9987 ± 0.0009	0.9987 ± 0.0009
EfficientNetV2S	0.9988 ± 0.0009	0.9988 ± 0.0009	0.9988 ± 0.0009	0.9988 ± 0.0009

**Figure 4.** F<sub>1</sub>-Score for each test repetition per model.

parameter count and diminished parameters when compared to all other CNNs analyzed. Furthermore, this preference is reinforced by the computational processing metrics, which are detailed in Table 3, collected during the experiments.

ShuffleNet is a CNN explicitly tailored for devices with constrained computational resources typically ranging from 10 to 150 MFLOPs [Zhang *et al.*, 2018]. Its compact nature, characterized by a reduced number of parameters and layers, implies a diminished computational cost during both training and inference. Consequently, ShuffleNet is well-suited for integration within solutions adhering to the Agricultural Internet of Things (IoT) paradigm which seeks to enable intelligent identification, positioning, tracking, monitoring, and management of agricultural entities and processes [Quy *et al.*, 2022].

Within this context, the proposed CNN can find practical application, particularly in technologies like UAVs, which play a crucial role in the remote surveillance of coffee plantations with respect to the targeted pathologies. Such utilization aligns with the principles of Edge Computing, an approach that extends cloud computing capabilities to the periphery of a network, encompassing IoT devices. Edge Computing is renowned for its advantageous features, including low latency, efficient data management, reduced bandwidth

consumption, and scalability [Hassan *et al.*, 2018].

Furthermore, ShuffleNet has demonstrated its utility in addressing various Computer Vision (CV) problems relevant to the Digital Agriculture domain, including the classification of weeds in agricultural fields [Carvalho *et al.*, 2019], real-time navigation and detection in the context of apple-picking robots [Ji *et al.*, 2022], and the formulation of a methodology for estimating maize nitrogen grading [Sun *et al.*, 2023].

Although the experimental results obtained on a diverse and realistic dataset labeled by experts have exhibited promising metrics that endorse the adoption of CNNs for the detection of coffee leaf diseases, it is imperative to acknowledge potential threats to the validity of these findings. A comparative analysis between the proposed solution, utilizing ShuffleNet and having experimental accuracy of  $99.93 \pm 0.03\%$ , and the work of Aufar *et al.* [2023], which reports a experimental accuracy of  $100\%$  using Inception-ResNetV2, reveals a compelling performance by both approaches in addressing a real-world Computer Vision task. A visual inspection of dataset samples, as depicted in Figures 6-8, provides preliminary insights into this matter.

The dataset providers have reported the utilization of data augmentation techniques. Data augmentation is considered as a preprocessing step applied exclusively to the training

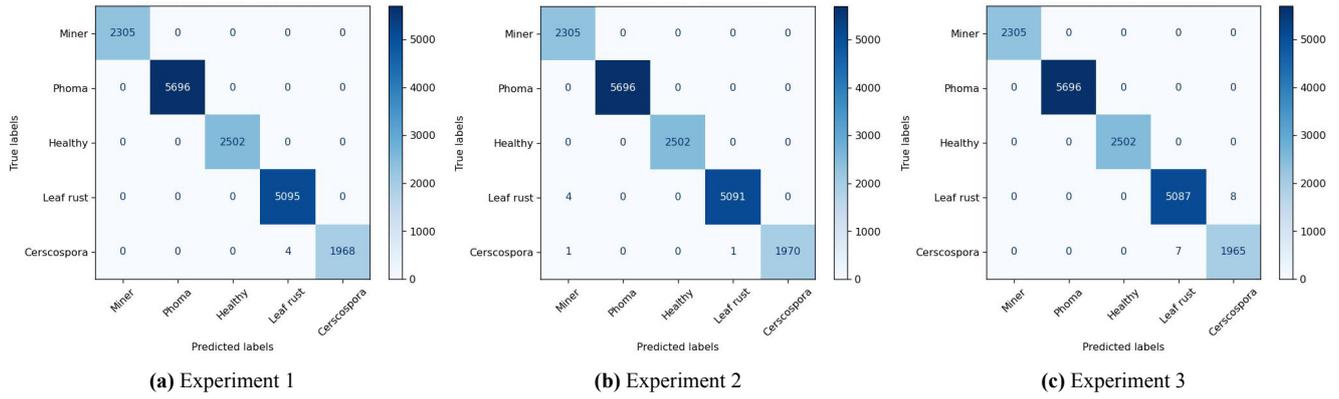


Figure 5. Confusion matrices for ShuffleNet on the testing data of each experimental repetition.

Table 3. Computing processing metrics for the contemporary approach.

Architectures	Training Time	Parameters	GFLOPS	Epochs
MobileNetV2	209 ± 18 s	2.91 Mi	0.237248	36 ± 02
ShuffleNet	314 ± 37 s	1.37 Mi	0.238612	69 ± 08
InceptionV3	512 ± 80 s	22.85 Mi	0.261484	70 ± 11
VGG-16	1364 ± 97 s	67.15 Mi	0.328626	92 ± 06
EfficientNetV2S	544 ± 71 s	22.85 Mi	0.351497	74 ± 09

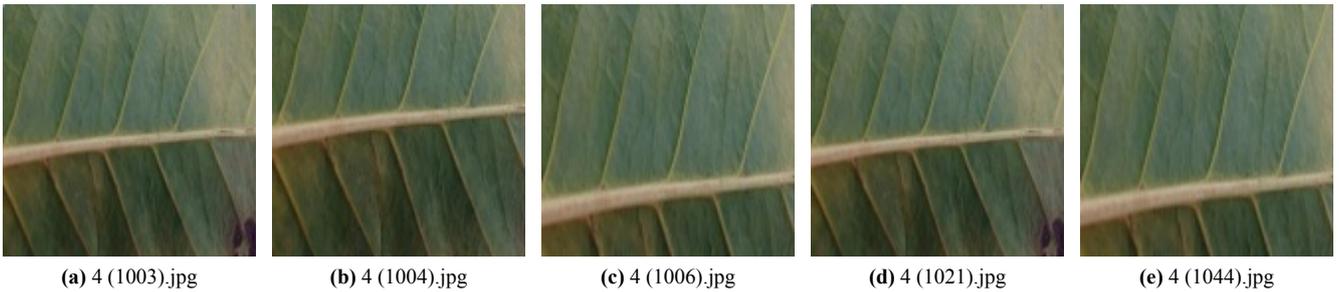


Figure 6. Samples from Cercospora class. Source: [Jepkoech *et al.*, 2021].



Figure 7. Samples from Miner class. Source: [Jepkoech *et al.*, 2021].

set, primarily to encourage model regularization [Goodfellow *et al.*, 2016]. Regrettably, the authors of the dataset have not provided explicit instructions or information regarding how the dataset should be partitioned to prevent augmented data from inadvertently infiltrating the test set. This situation, despite the incorporation of countermeasures against overfitting, such as early stopping, has the potential to impede the model's generalization capacity significantly. In the event that the issue is not attributed to data augmentation, a visual examination suggests that the data collection process may not

have adequately encompassed sample diversity as typically encountered in real-world scenarios. This apparent lack of diversity substantially reduces the complexity of the learning task for CNNs, given their inherent capacity for handling features that are invariant under translation, rotation, scale, illumination and color [Goodfellow *et al.*, 2016].

Consequently, a more comprehensive investigation is warranted to substantiate the observed high performance. In this context, the subsequent section will present the results of an extended validation encompassing three distinct datasets, all

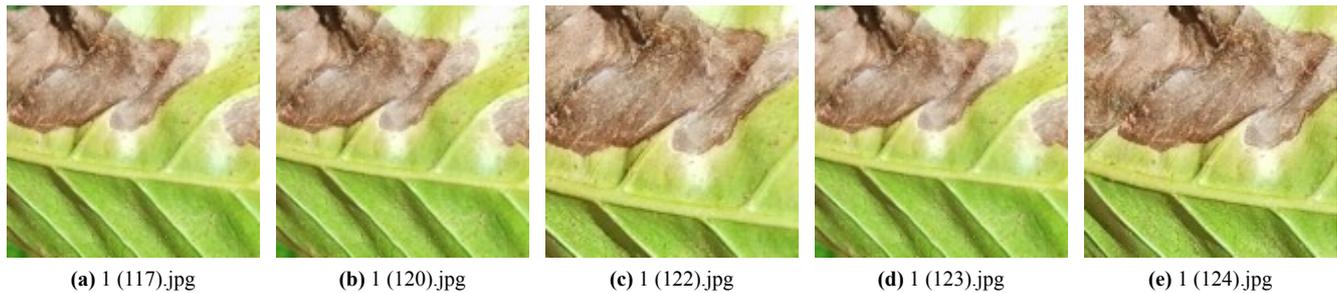


Figure 8. Samples from Phoma class. Source: [Jepkoech *et al.*, 2021].

centered on the task of coffee plant leaf disease detection.

## 5 External Validation

According to Ho *et al.* [2020], external validation is critical for establishing ML model quality. It involves the use of independently derived datasets (hence, external), to validate the performance of a model that was trained on initial input data. The reason this method is useful is because a well-trained model, that's good at capturing important information, is robust and will continue to exhibit good results even when repeatedly challenged with new data.

In this section, an examination is undertaken to explore a divergent external validation approach, the primary aim of which is to determine the extent of generalizability within the feature set. This assessment was pursued by using three distinct coffee leaf diseases datasets based on the criterion that they had been utilized in at least one prior study involving DL methods. The performance of ShuffleNet was analyzed under two conditions: one adhering to the identical experimental parametrization detailed in previous sections (named *A*), and the other without the application of Early Stopping regularization (named *B*). This choice was motivated by the consideration that these datasets offer a smaller volume of training data compared to JMUBEN and JMUBEN2 datasets. Subsequent subsections provide a comprehensive presentation of these datasets and the results derived from the generalization evaluation.

### 5.1 BRACOL Dataset

The *Brazilian Arabica Coffee Leaf* (BRACOL) dataset was created to assess Deep Learning algorithms for identifying coffee tree biotic stresses and healthy samples [Krohling, 2019]. Smartphone-captured images of coffee leaves' abaxial (lower) sides were collected throughout the year and expert-labeled, then partitioned into training, validation, and test sets. The Symptom dataset, comprising 2209 images, was generated by isolating single stress conditions from original images. Illustrative samples from the BRACOL Symptom dataset are presented in Figure 9.

As it can be seen, BRACOL Symptoms datasets comprise the same classes as JMUBEN and JMUBEN2 datasets, but the main difference relies on the number of samples which is 185.45% smaller than the latter. Moreover, there are also differences on the distribution of samples per class, as shown in Figure 10.



Figure 9. Randomly drawn examples from each class from BRACOL Symptom dataset. Source: [Krohling, 2019].



Figure 10. Samples per class on BRACOL Symptom dataset. Source: [Krohling, 2019].

The proponents of BRACOL Symptoms dataset already provided a stratified partition of samples to carry out a hold-out cross-validation: 70% for train, 15% for train validation and 15% for test. Prior work on literature evaluated different CNN architectures on such multi-classification Supervised Learning CV task, where best results were observed for ResNet50 (25 Mi parameters, 50 layers) trained for 80 epochs using the SGD optimizer, data augmentation and pre-trained weights from ImageNet dataset [Esgario *et al.*, 2020b].

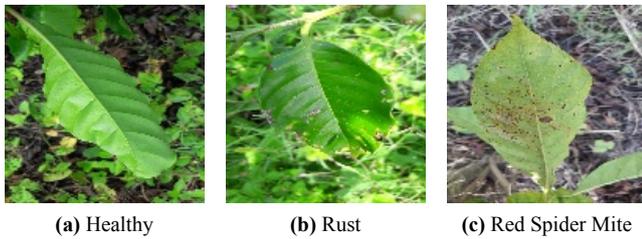
The results obtained on evaluating the ShuffleNet on the BRACOL Symptom dataset are shown in Table 4 in contrast with the related work from literature. Upon examining the external validation results, it is evident that the model proposed in this study exhibits commendable generalization capabilities, despite being trained on a notably smaller dataset in comparison to the original experiments. On contrasting the results with related work, ShuffleNet falls short in comparison with ResNet50, an  $F_1$ -Score reduction of 11.21%, probably due to the smaller number of parameters (94.52% less parameters), weights initialization strategy and data augmentation procedures.

**Table 4.** Experimental results on BRACOL Symptom dataset.

Architectures	Precision	Recall	F <sub>1</sub> -Score	Accuracy
ShuffleNet A	0.8309	0.8268	0.8244	0.8268
ShuffleNet B	0.8635	0.8626	0.8605	0.8626
ResNet50	0.9685	0.9699	0.9692	0.9707

## 5.2 RoCoLe Dataset

The RoCoLe dataset contains imagery of upper and back side of coffee leaves collected from *Coffea canephora* species, also known as robusta coffee, showing healthy samples and also examples affected by Rust (*Hemileia vastatrix*) and Red Spider Mite (*Tetranychus urticae*), as shown in Fig. 11. The dataset contains 1560 images collected from a crop in Ecuador with 390 coffee plants [Parraga-Alava *et al.*, 2019]. The number of samples per class for this dataset are depicted in Fig. 12.



**Figure 11.** Randomly drawn examples from each class from RoCoLe dataset.

Source: [Parraga-Alava *et al.*, 2019].



**Figure 12.** Samples per class on RoCoLe dataset. Source: [Parraga-Alava *et al.*, 2019].

It can be noticed that the images in the RoCoLe dataset were captured under particularly demanding conditions for automated detection, as they exhibit variations in background, leaf positioning, illumination, and other factors. In order to compare with related work, it was performed a 80%/20% holdout cross-validation strategy, but with 10% of overall data for training reserved for validation.

Based on the results presented in Table 5, it is evident that ShuffleNet consistently delivered equiparable performance in the external validation process for this dataset, as observed in both experiments *A* and *B*. Nevertheless, the outcomes did not surpass the results obtained in the study by Işık and Eskicioglu [2022]. The aforementioned authors explored eight different CNN architectures, and five of them achieved an experimental accuracy of 100% on the RoCoLe dataset. MobileNetV2 was selected as their reference architecture due to its smaller size, although it is important to note that pre-processing the images incurred computational costs in their work. This disparity in results may be attributed to variations in coffee species, data availability, image capture conditions, and the specific pathologies considered.

**Table 5.** Experimental results on RoCoLe dataset.

Architectures	Precision	Recall	F <sub>1</sub> -Score	Accuracy
ShuffleNet A	0.5469	0.5841	0.5488	0.5841
ShuffleNet B	0.5777	0.5873	0.5801	0.5873
MobileNetV2	1.000	1.000	1.000	1.0000

## 5.3 Rust and Miner in Coffee Dataset

The third dataset considered in this external validation contains examples of unhealthy-only arabica coffee leaves affected by rust and miner [Brito Silva *et al.*, 2020], as shown in Fig. 13. It comprises 257 images affected by miner and 258 images affected by rust. Samples were randomly stratified partitioned into training (70%), validation (10%) and test (20%), according to the same strategy used in the previous external validation scenarios.



**Figure 13.** Randomly drawn examples from each class. Source: [Brito Silva *et al.*, 2020].

Results on ShuffleNet for Rust and Miner in Coffee Dataset are shown in Table 6. Related work on the same dataset considered a CV detection task [Carneiro *et al.*, 2021] and, thus, cannot be directly compared.

**Table 6.** Experimental results on Rust and Miner in Coffee dataset.

Architecture	Precision	Recall	F <sub>1</sub> -Score	Accuracy
ShuffleNet A	0.2730	0.5225	0.3586	0.5225
ShuffleNet B	0.9371	0.9369	0.9369	0.9369

The third external validation scenario provides insights into the differences between the training strategies *A* and *B* and their impact on the final performance results. It underscores the resilience of ShuffleNet in addressing the validation task, even when faced with a scarcity of training data. However, it suggests that enhanced generalization outcomes might be achievable with additional refinements in fine-tuning. An alternative approach that could potentially enhance performance during external validation involves utilizing pre-trained weights from the initial task.

## 6 Conclusion and Future Work

The aim of this work was to compare the performance of two approaches using ANNs in the context of the CV multi-class classification problem of coffee leaf diseases. To achieve this, a realistic dataset and an experimental scenario with cross-validation and repetitions were considered. The results obtained, supported by statistical tests, demonstrated that CNNs proved to be more suitable for this task. The

ShuffleNet architecture, listed as the reference solution, was trained with less data and has significantly fewer parameters than other related works in the literature. This was corroborated by processing cost metrics collected during the experiments. In order to provide a more stringent assessment of the reference solution, a divergent external validation was conducted, using three other datasets for the same problem. The experimental results obtained favorably corroborated the robustness of the ShuffleNet architecture for this task, even when considering a different coffee species.

This article contributes to the body of solutions for Digital Agriculture, specifically focusing on coffee farming, aiming to address the challenges associated with the detection of a major biotic stress factors affecting this cultivation. The proposed solution leverages emerging Deep Learning techniques, while also considering the computational cost involved, which may facilitate efforts for its *in situ* adoption within the context of Edge Computing for Agricultural IoT.

The proposed solution has limitations that need to be addressed. First, the primary databases used, JMuBEN and JMuBEN2, are from Kenya. In Brazil and other regions, the considered pathologies may manifest differently due to distinct environmental conditions. Additionally, there may be other pathologies not included in these databases. Therefore, a more in-depth analysis, particularly involving experts in the field, is necessary to fully utilize the proposed solution. This may lead to adaptations such as retraining or transfer learning. Another limitation arises from the nature of plant foliar disease image databases. The examples in these databases, which are essential for the development of Machine and Deep Learning models, only consider situations where the disease characteristics are already sufficiently developed. This prevents early interventions that could minimize damage. This latter aspect, in particular, should be considered when proposing new databases for this domain, favoring the development and applications of Digital Agriculture.

Future work will involve deploying the proposed solution on a low-cost, single-board computer (e.g., Raspberry Pi or Jetson Nano) for real-world field testing on coffee crops in collaboration with an agricultural specialist. This deployment aims to assess the solution's performance under practical conditions. Furthermore, we emphasize the need for future research to evaluate the processing power requirements of these CNNs for achieving energy-sustainable solutions. Such evaluations will pave the way for advancements in Green AI and Edge Computing, fostering the development of resource-efficient models suited to resource-constrained environments.

## Declarations

## Acknowledgements

The authors would like to express their gratitude for the financial support provided by the National Council for Scientific and Technological Development (CNPq) and the Research Support Foundation of the State of Amazonas (FAPEAM), respectively, through the PAIC programs for the years 2021/2022 and 2022/2023. They also

acknowledge the material support received from the Laboratory of Intelligent Systems (LSI) at the Amazonas State University (UEA).

## Authors' Contributions

Both authors have contributed equally to this manuscript.

## Competing interests

The authors declare they do not have competing interests.

## Availability of data and materials

The datasets generated and/or analysed during the current study are available <https://github.com/el1oa/jis-2023>

## References

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.
- Albuquerque, L. and Guedes, E. B. (2023). Um comparativo de abordagens com redes neurais artificiais para detecção inteligente de patologias na folha do café. In *Anais do XIV Workshop de Computação Aplicada à Gestão do Meio Ambiente e Recursos Naturais*, pages 131–140, Porto Alegre, RS, Brasil. SBC. DOI: 10.5753/wcama.2023.229712.
- Aufar, Y., Abdilllah, M. H., and Romadoni, J. (2023). Web-based CNN application for Arabica Coffee leaf disease prediction in smart agriculture. *J. RESTI (Rekayasa Sist. Dan Teknol. Inf.)*, 7(1):71–79.
- Brazil (2022). Brazilian Ministry of Agriculture, Livestock, and Supply – Brazilian Agribusiness Overview – Coffee in Brazil. Available at <https://www.gov.br/agricultura/pt-br/assuntos/politica-agricola/cafe/cafeicultura-brasileira>. Accessed in October, 2023.
- Brito Silva, L., Cavalcante Carneiro, A. L., and Silveira Almeida Renaud Faulin, M. (2020). Rust (*Hemileia vastatrix*) and leaf miner (*Leucoptera coffeella*) in coffee crop (*Coffea arabica*).
- Cao, K., Liu, Y., Meng, G., and Sun, Q. (2020). An overview on edge computing research. *IEEE Access*, 8:85714–85728. DOI: 10.1109/ACCESS.2020.2991734.
- Carneiro, A. L. C., Silva, L. B., and Faulin, M. S. A. R. (2021). Artificial intelligence for detection and quantification of rust and leaf miner in coffee crop.
- Carvalho, V., Guedes, E., and Salame, M. (2019). Classificação de ervas daninhas em culturas agrícolas com comitês de redes neurais convolucionais. In *Anais do XVI*

- Encontro Nacional de Inteligência Artificial e Computacional*, pages 60–71, Porto Alegre, RS, Brasil. SBC. DOI: 10.5753/eniac.2019.9272.
- Chollet, F. *et al.* (2015). Keras. <https://keras.io>.
- Coelho, L. P. (2013). Mahotas: Open source software for scriptable computer vision. *Journal of Open Research Software*. DOI: 10.5334/jors.ac.
- CONAB (2022). Companhia Nacional de Abastecimento – 4º Levantamento da Safra de Café. ISSN: 2318-7913. Available at <http://www.conab.gov.br/>. Accessed in October, 2023.
- Cover, T. M. and Thomas, J. A. (2006). *Elements of Information Theory*. John Wiley & Sons, Nashville, TN, 2 edition.
- Dias, J. and Saito, J. (2021). Coffee plant image segmentation and disease detection using JSEG algorithm. In *Anais do XVII Workshop de Visão Computacional*, pages 42–47, Porto Alegre, RS, Brasil. SBC. DOI: 10.5753/wvc.2021.18887.
- Embrapa (2014). *Visão 2014–2034 – O Futuro do Desenvolvimento Tecnológico da Agricultura Brasileira (Síntese)*. Embrapa, Distrito Federal, Brasil.
- Escolano, F., Suau, P., and Bonev, B. (2009). *Information Theory in Computer Vision and Pattern Recognition*. Springer, London, England, 2009 edition.
- Esgario, J. G. M., Krohling, R. A., and Ventura, J. A. (2020a). Deep learning for classification and severity estimation of coffee leaf biotic stress. *Computers and Electronics in Agriculture*, 169:105162. DOI: <https://doi.org/10.1016/j.compag.2019.105162>.
- Esgario, J. G. M., Krohling, R. A., and Ventura, J. A. (2020b). Deep learning for classification and severity estimation of coffee leaf biotic stress. *Computers and Electronics in Agriculture*, 169:105162. DOI: 10.1016/j.compag.2019.105162.
- Faceli, K., Lorena, A. C., Gama, J., and Carvalho, A. C. P. L. F. (2021). *Inteligência Artificial – Uma abordagem de aprendizado de máquina*. Editora LTC, Rio de Janeiro, 2 edition.
- FAO (2023). Food and Agriculture Organization of the United Nations – Coffee. Available at <https://www.fao.org/markets-and-trade/commodities/coffee/en/>. Accessed in October, 2023.
- Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. The MIT Press, Cambridge.
- Haralick, R. M., Shanmugam, K., and Dinstein, I. (1973). Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3(6):610–621.
- Hassan, N., Gillani, S., Ahmed, E., Yaqoob, I., and Imran, M. (2018). The role of edge computing in internet of things. *IEEE Communications Magazine*, 56(11):110–115. DOI: 10.1109/mcom.2018.1700906.
- Haykin, S. (2008). *Neural Networks and Learning Machines*. Pearson Prentice-Hall, USA, 3 edition.
- Ho, S. Y., Phua, K., Wong, L., and Bin Goh, W. W. (2020). Extensions of the external validation for checking learned model interpretability and generalizability. *Patterns*, 1(8):100129. DOI: <https://doi.org/10.1016/j.patter.2020.100129>.
- Howard, A., Sandler, M., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). Mobilenetv2: Inverted Residuals and Linear Bottlenecks. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4510–4520. Available at: <https://arxiv.org/abs/1801.04381>.
- Hung, C.-C., Song, E., and Lan, Y. (2019). *Image Texture Analysis – Foundations, Models and Algorithms*. Springer, Switzerland.
- İşik, A. H. and Eskicioglu, Ö. C. (2022). Classification of coffee leaf diseases through image processing techniques. In *Artificial Intelligence and Smart Agriculture Applications*, pages 233–252. Auerbach Publications, USA.
- Jepkoech, J., Mugo, D. M., Kenduiywo, B. K., and Too, E. C. (2021). Arabica coffee leaf images dataset for coffee leaf disease detection and classification. *Data in Brief*, 36:107–142.
- Ji, W., Pan, Y., Xu, B., and Wang, J. (2022). A real-time apple targets detection method for picking robot based on ShufflenetV2-YOLOX. *Agriculture*, 12(6):856. DOI: 10.3390/agriculture12060856.
- Kamilaris, A. and Prenafeta-Boldú, F. X. (2018). Deep Learning in Agriculture: A Survey. *Computers and Electronics in Agriculture*, 147:70–90.
- Khan, S., Rahmani, H., Shah, S. A. A., and Bennamoun, M. (2018). *A Guide to Convolutional Neural Networks for Computer Vision*. Synthesis Lectures on Computer Vision. Morgan & Claypool, San Rafael, California, USA, 1a edition.
- Kingma, D. P. and Ba, J. L. (2015). Adam: a method for stochastic optimization. In *Proceedings of 3rd International Conference on Learning Representations*, San Diego, CA. ArXiv. DOI: <https://doi.org/10.48550/arXiv.1412.6980>.
- Krohling, R. A. (2019). BRACOL - a brazilian arabica coffee leaf images dataset to identification and quantification of coffee diseases and pests.
- Mesquita, C. M., Rezende, J. E., Carvalho, J. S., Júnior, M. A. F., Moraes, N. C., Dias, P. T., Carvalho, R. M., and Araújo, W. G. (2016). *Manual do Café – Distúrbios Fisiológicos, Pragas e Doenças do Cafeeiro*. EMATER, Belo Horizonte, 1 edition.
- Mirjalili, F. and Hardeberg, J. Y. (2022). On the quantification of visual texture complexity. *Journal of Imaging*, 8(9). DOI: 10.3390/jimaging8090248.
- Novtahaning, D., Shah, H. A., and Kang, J.-M. (2022). Deep learning ensemble-based automated and high-performing recognition of coffee leaf disease. *Agriculture*, 12(11). DOI: 10.3390/agriculture12111909.
- Palit, A. K. and Popovic, D. (2005). *Computational Intelligence in Time Series Forecasting*. Springer-Verlag, London.
- Parraga-Alava, J., Cusme, K., Loor, A., and Santander, E. (2019). RoCoLe: A robusta coffee leaf images dataset for evaluation of machine learning based methods in plant diseases recognition. *Data in Brief*, 25(104414).
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P.,

- Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- Prince, S. (2012). *Computer Vision: Models, Learning, and Inference*. Cambridge University Press, Cambridge.
- Quy, V. K., Hau, N. V., Anh, D. V., Quy, N. M., Ban, N. T., Lanza, S., Randazzo, G., and Muzirafuti, A. (2022). IoT-enabled smart agriculture: Architecture, applications, and challenges. *Applied Sciences*, 12(7):3396. DOI: 10.3390/app12073396.
- Rehman, A., Tariq, Z., din Memon, S. u., Zaib, A., Khan, M. U., and Aziz, S. (2021). Cucumber leaf disease classification using local tri-directional patterns and haralick features. In *2021 International Conference on Artificial Intelligence (ICAI)*, pages 258–263, Islamabad, Pakistan. IEEE. DOI: 10.1109/ICAI52203.2021.9445237.
- Santos, F., Canuto, A., Bedregal, B., Palmeira, E., and Silva, I. (2019). Supervised methods applied to the construction of a vision system for the classification of cocoa beans in the cut-test. In *Anais do XVI Encontro Nacional de Inteligência Artificial e Computacional*, pages 72–83, Porto Alegre, RS, Brasil. SBC. Available at <http://dx.doi.org/10.5753/eniac.2019.9273>. Accessed at March 18, 2024.
- Schwartz, R., Dodge, J., Smith, N. A., and Etzioni, O. (2020). Green ai. *Commun. ACM*, 63(12):54–63. DOI: 10.1145/3381831.
- Sharma, A., Jain, A., Gupta, P., and Chowdary, V. (2021). Machine learning applications for precision agriculture: A comprehensive review. *IEEE Access*, 9:4843–4873.
- Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. Available at <https://arxiv.org/abs/1409.1556>. Accessed at March 18, 2024.
- Sun, W., Fu, B., and Zhang, Z. (2023). Maize nitrogen grading estimation method based on UAV images and an improved shufflenet network. *Agronomy*, 13(8):1974. DOI: 10.3390/agronomy13081974.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 1–9, Massachusetts, USA. IEEE.
- Tan, M. and Le, Q. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 6105–6114.
- Tang, S., Zhu, Q., Zhou, X., Liu, S., and Wu, M. (2002). A Conception of Digital Agriculture. In *IEEE International Geoscience and Remote Sensing Symposium*, pages 3026–3028, Canada. IEEE.
- Van Rossum, G. and Drake, F. L. (2009). *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA.
- VijayaLakshmi, B. and Mohan, V. (2016). Kernel-based PSO and FRVM: An automatic plant leaf type detection using texture, shape, and color features. *Computers and Electronics in Agriculture*, 125:99–112. DOI: <https://doi.org/10.1016/j.compag.2016.04.033>.
- Walpole, R. E., Myers, R. H., Myers, S. L., and Ye, K. (2012). *Probability & Statistics for Engineers & Scientists*. Prentice Hall, United States, 9 edition.
- Xian, T. S. and Ngadiran, R. (2021). Plant diseases classification using machine learning. *Journal of Physics: Conference Series*, 1962(1):012024. DOI: 10.1088/1742-6596/1962/1/012024.
- Zhang, X., Zhou, X., Lin, M., and Sun, J. (2018). Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6848–6856, USA. IEEE.