Coffee Leaf Diseases Identification and Severity Classification using Deep Learning

Eduardo Lisboa
Instituto de Computação
Universidade Federal de Alagoas
Maceió - Alagoas, Brazil
Email: eall@ic.ufal.br

Givanildo Lima
Instituto de Computação
Universidade Federal de Alagoas
Maceió - Alagoas, Brazil
Email: glnj@ic.ufal.br

Fabiane Queiroz
Centro de Ciências Agrárias
Universidade Federal de Alagoas
Rio Largo - Alagoas, Brazil
Email: fabiane.queiroz@laccan.ufal.br

***Abstract*—In this paper, we propose a method for automatic identification and classification of leaf diseases and pests in the Brazilian Arabica Coffee leaves. We developed a Machine Learning model, trained with the BRACOL public image dataset, to evaluate if a given image of a leaf has a disease or pest — *Miner*, *Phoma*, *Cercospora* and *Rust* — or if it is healthy. We then compared our model with other famous and well-known classification models, and we were able to achieve an accuracy of 98,04%, which greatly exceeds the accuracy of the other methods implemented. In addition, we developed an assessment to perform a classification related to the percentage of each leaf that is affected by the disease, achieving an accuracy of approximately 90%.**

I. Introduction

Machine Learning is the branch of Artificial Intelligence that focuses on the development of applications that learn based on data and enhance their precision with time without the need of being reprogrammed [1]. Machine learning algorithms are “trained” to find patterns and resources in large amounts of data to make decisions and predictions based on new data. The better the algorithm, the more accurate decisions and predictions will be as it processes more data. This property of Machine Learning has the potential of being extremely helpful in situations where it would take a lot more time for a human to do the same because of the large amounts of data a computer can process.

A common problem in Brazilian crops, as well as in crops all around the world, is diseases and pests [2] [3]. All kinds of crops can suffer from such issues, and as the world needs these agricultural products, preventing them is a highly sought subject. Usually, it can be hard to diagnose these diseases and pests in advance because of the similarities between diseases or the size of the insects. Identifying these diseases is fundamental to avoid crop losses [4] [5] [6] [7], treating them as fast as possible. With that in mind, we want to develop a Machine Learning algorithm, more precisely a Deep Neural Network, that could identify whether a plant is healthy or not and its disease severity. That can assist rural producers in identifying and allowing these producers to treat their plants quickly and effectively.

The use of Deep Neural Networks has been proven to be greatly beneficial in increasing the accuracy of classification problems [8]. This paper proposes utilizing a Deep Convolutional Neural Network to create a viable and reasonably precise way of identifying and classifying coffee leaf diseases and their severity. The rest of this paper is structured as follows: The literature review and discussion of related works is done in Section II. Section III describes the adopted method for the development of this work. Results are presented in Section IV and finally, Section V concludes.

II. Related Works

In the literature, when it comes to the use of Neural Networks, it has had many applications such as identifying weeds, classifying land cover, fruit counting, plant leaf stress detection, etc [9] [10] [11] [12] [13]. Specifically, with plant leaf stress, several papers propose methods of determining these stresses.

Among the most common methods of the state-of-art, we can cite the use of texture attributes analysis and texture analysis combined with Deep Learning Techniques [8]. This method is used to recognize diseases such as *Cercospora* and *Rust* in coffee leaves. We can cite either the use of Convolutional Neural Networks (CNNs) to identify diseases with limited input data [14] or Bayesian Generalized Linear Regression (*GBLASSO*) [15]. All these methods can be used not only for coffee leaves but also for pretty much any major crop with visible disease symptoms, like strawberries [13], paddy leaves [11] or soybean [9].

The use of Neural Networks is becoming common over the years because more and more papers are proving it can be used to enhance the results achieved from other methods significantly. As Xiao et al. [13] has shown, by using a Convolutional Neural Network, they have reached an accuracy of 100% on the identification of leaf blight cases in strawberries. Esgario et al. [16] proposed a ResNet50 — a CNN that is 50 layers deep — architecture paired with data augmentation techniques to increase the robustness of the system obtained 95,24% accuracy for biotic stress classification. Another approach to the problem of diseases and pests was proposed by de Oliveira Aparecido et al. [12], using the correlation between infection rates and weather variables, combined with machine learning algorithms.

But, as these methods can bring significantly better results, they also carry some potential disadvantages, shown by Liu and Wang [10]. When using a Neural Network only as a feature extractor, you rely on other classifiers for the classification results. Using a Neural Network directly as the classifier, some portion of the inputs must be accounted for; otherwise, some important characteristics may be pooled out. Generally, only one class of stress is considered per input.

III. Material and Methods

This section presents the materials used in developing this work and explains the implemented methods in detail.

A. The BRACOL Dataset

In this work, the experiments were conducted using the BRACOL Dataset [17], which is in the public domain, consisting of 1747 images of size 2048 x 1024 pixels. Still, due to the corruption of the compressed archive, only 1216 images could be used.

The images in BRACOL are labeled with healthy or diseased and the severity of each disease present, being these diseases *Rust*, *Miner*, *Cercospora* and *Phoma*, and the severity is divided in 5 different levels, being 0 a healthy leaf and four a very compromised leaf. All images were captured in similar conditions of lighting and a single leaf per image. The leaves were collected at different times of the year. Fig 1 shows two examples of the dataset images.

((a)) Diseased Leaf (Rust/Miner)

((b)) Healthy Leaf

Fig. 1. Samples of the BRACOL Image Dataset

B. Leaf Disease Identification

A CNN (Convolutional Neural Network) is a Deep Neural Network specialized in processing data that has a grid-like representation, such as an image. A digital image represents visual data containing a series of pixels arranged in a grid-like fashion that has pixel values denoting how bright and what color each pixel should be. It usually consists of four main operations [18]: *feature extraction*, [19] from an input; *activation function* [20] such as ReLU (Rectified Linear Unit), Sigmoid and Softmax; *pooling* [21], which reduces the dimensions of a layer; and *fully connected layers* (or *dense layers*) [22], which are layers that have every node connected to all nodes on the adjacent layers. These are usually the last layers of a CNN.

In our model, we had one single input, an array of images to be processed. Our proposed method also had ten hidden layers consisting of three 2D convolution layers interspersed with three 2D max-pooling layers, then a dropout layer, a flattening layer, and two dense layers, producing as an output a single array with two items containing the probability of the image being a diseased or a healthy leaf. Fig 2 shows the basic structure of our CNN.

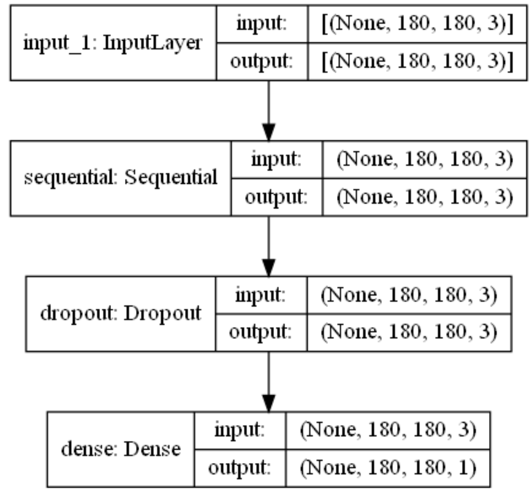


Fig. 2. Model's Basic Structure

C. Severity Classification

The classification was carried out based on the severity of the disease from the percentage of the affected leaf. For this, the BRACOL dataset was divided into three subsets according to the degree of severity of the injury suffered:

- S_0: Class of images considered healthy, with a percentage of affected leaf area in the range of 0-1%;
- S_1: Class of images with medium health, with a percentage of affected leaf area in the range of 1-10%;
- S_2: Image class heavily affected by biotic stress, with a percentage of affected leaf area greater than 10%.

In Fig. 3, it is possible to visualize an example figure of each class and make the understanding of each class more intuitive.

((a)) S_0

((b)) S_1

((c)) S_2

Fig. 3. Samples of the BRACOL Image Dataset with your respective class

After the division, we noticed that these sets were unbalanced since S_1 has a dimension considerably higher than the other classes. A data augmentation technique was applied to solve this problem. From each image of a given set, n other images are generated that can undergo modifications such as a rotation of up to 30°, horizontal or vertical displacement of up to 10%, zoom of up to 10%, vertical or horizontal mirroring, in addition to changes in the range of up to 25% in brightness and saturation. This extended dataset will be used for evaluation (training and validation).

The value of n is selected so that the ratio of the number of images between the sets S_0, $S2$ and S_1 approaches 1. Thus, for a generic set i, n is given by

$$n = \lfloor \#S_i / \#S_1 \rfloor \tag{1}$$

where $\#$ is a set cardinality operator, $\lfloor\rfloor$ is the floor function operator and $i = \{0, 2\}$.

From these labels that defined the classes, the image base was randomly divided into two sets, one for training, containing 80% of the images, and one for testing, containing the other 20% of the base of images.

To carry out the classification step, we chose to use a pre-trained network model, called ResNet-50 [23], which is a 50-layer-deep convolutional neural network. The version used was pre-trained on more than one million images from the ImageNet database. At this stage, the images are resized, which now have a 224x224 dimension.

To carry out the experiment, a number of epochs equal to 20 and a sample size (Batch Size) equal to 32 were adopted. In addition, three dense layers were added, with 1024 neurons each and a layer with 512 neurons, all with ReLU activation functions.

IV. Results and Discussion

In this section, we evaluate the performance of the proposed method in comparison with the texture identification methods *Local Binary Patterns* (LBP), *Gray Level Co-occurrence Matrix* (GLCM) and the neural network *VGG16*, being the latter retrained with our dataset.

A. Leaf Disease Identification: Dicussion

We reduce the neural networks training time resizing the original images to 512 x 512 pixels. The experiments were made using a machine with an Intel® Core™ I5-10600K @ 4.10GHz with 6 cores and 12 threads, two 8GB DIMM DDR4 2400MHz memory sticks, totalizing 16GB of RAM and Windows 10 Pro Operating System and also a NVidia GeForce GTX 1660 Super graphics card, with 6GB of GDDR6 memory.

The algorithms were implemented using Python 3.8.7 programming language, using the libraries TensorFlow 2.4.1 [24], Keras 2.4.3 [25] included in the TensorFlow library for the creation and training of the neural network, scikit-learn 0.24.1 [26] for the accuracy analysis and matplotlib 3.3.4 [27] for the graph modeling.

The parameters used to create the neural network were the image patch size, the batch size and the number of epochs. All of the experiments used the same optimizer, learning rate and loss function, which were Adam, 0.001 and Sparse Categorical Cross-Entropy, respectively. The training of the neural networks were based on minimizing the loss instead of maximizing the accuracy.

When the model was assembled, it was then trained by processing images from the BRACOL Dataset [17], using approximately 80% of the total number of images for training and accuracy measuring, being this 80% further divided in another 80% for training and 20% for validation. After this process, with every epoch and batch size trained, the best model had a training and validation accuracy and loss as seen on Fig 4. The remaining 20% of images that the model has never seen were then processed by the resulting model and used to calculate its accuracy, which will be discussed in the next section.

Table I shows the comparison between the accuracy obtained with every implemented method. The evaluation metrics used were Accuracy, Precision, Recall and F1 scores. The accuracy of the developed neural network is from the model with the best results. As we can see, the proposed method had all the evaluation metrics significantly higher than the other methods implemented, with our neural network reaching 98,04 % accuracy mean and a standard deviation of only 0,0063.

B. Severity Classification: Discussion

In Fig. 5, graphs relating accuracy (Fig. 5(a)) and loss (Fig. 5(b)) in the training and validation stages are presented.

Thus, we built a method capable of distinguishing between the three classes determined with a precision of 90%, being able to determine how affected was the leaf whose image was evaluated.

V. Conclusion

In this paper, we proposed a method of identifying and classifying diseases of pests in Brazilian Arabica coffee leaves using neural networks. The results point out the feasibility of the proposed method, having acquired an accuracy of 97,17%, which outmatches the compared methods.

We also developed a method using transfer learning to estimate the degree of involvement of the leaf from three levels related to the percentage that was affected by the disease. In this approach, we obtained a precision of 90%, a result considered satisfactory.

As future works, we intend to identify the optimal parameters for the neural network, use patches of the original images to further enhance the dataset size as well as making it distinguish which disease or pest is present on the leaf and making it available as a website and/or as a smartphone application.

Acknowledgment

The authors would like to thank the Institutional Scientific Initiation Scholarship Program (*Programa Institucional de Bolsas de Iniciação Científica — PIBIC*) of the Federal University of Alagoas (*Universidade Federal de Alagoas — UFAL*).

TABLE I
ACCURACY RESULTS OF IMPLEMENTED METHODS

Methods	Accuracy	Precision	Recall	F1	Accuracy Mean	Standard Deviation
Co-occurrence Matrix (GLCM)	56,50 %	56,50 %	56,50 %	56,50 %	56,50 %	0
Local Binary Patterns (LBP)	84,75 %	84,75 %	84,75 %	84,75 %	84,75 %	0
Neural Network (VGG16)	83,05 %	83,05 %	83,05 %	83,05 %	83,05 %	0
Proposed Method	**97,18 %**	**98,66 %**	**98,00 %**	**98,33 %**	**98,04 %**	0,0063

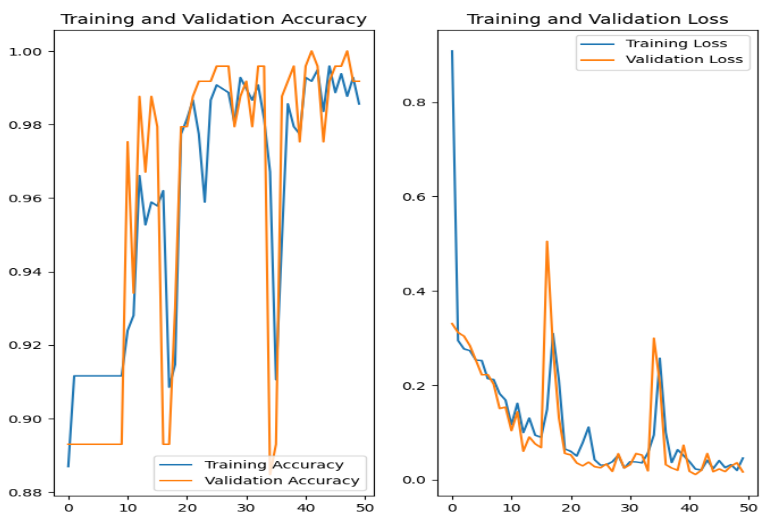
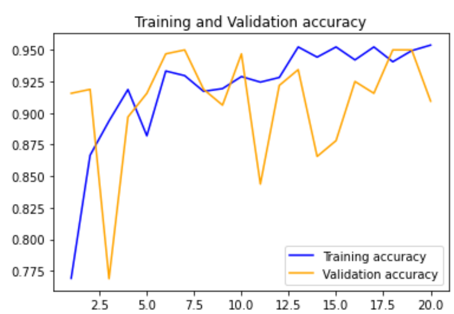
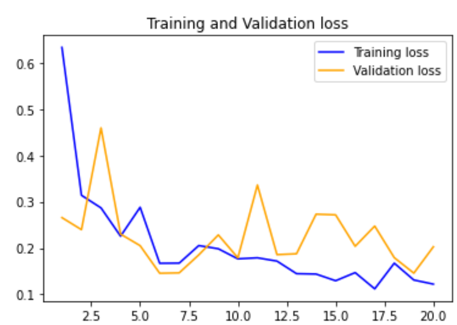


Fig. 4. Training and Validation Accuracy and Loss of best results



((a)) Accuracy



((b)) Loss

Fig. 5. Accuracy and Loss in the training and validation stages

REFERENCES

[1] IBM, "Machine learning," https://www.ibm.com/br-pt/cloud/learn/machine-learningtoc-machine-le-CLH1KWJn, 2020.

[2] J. Hoff, "Pest problem: Farmers dealing with extreme drought now fighting off insect infestations," https://www.kare11.com/article/weather/pest-problem-farmers-dealing-with-extreme-drought-now-fighting-off-insect-infestations/89-06763123-73de-4e25-bd81-feb4fbc4575a, 2021.

[3] P. Silvestre, "Milho: seca e ataque de pragas são os principais problemas de agricultores em mt," https://www.canalrural.com.br/noticias/milho-seca-ataque-pragas-problemas-agricultores-mt/, 2021.

[4] S. Trimble, "Detecting plant root diseases pests," https://cid-inc.com/blog/detecting-plant-root-diseases-pests/, 2019.

[5] C. Rural, "Nova praga de pastagem é registrada em ms pela embrapa," https://www.canalrural.com.br/noticias/nova-praga-de-pastagem-e-registrada-em-ms-pela-embrapa/, 2021.

[6] R. P. News, "Produção de cana pode diminuir até 60% com incidência de pragas e doenças agrícolas," https://revistarpanews.com.br/producao-de-cana-pode-diminuir-ate-60-com-incidencia-de-pragas-e-doencas-agricolas/, 2021.

[7] H. Andrade, "Produção de sorgo sofre com praga do pulgão na região oeste do rn e safra deve ser 40% menor," https://g1.globo.com/rn/rio-grande-do-norte/noticia/2021/05/23/producao-de-sorgo-sofre-com-praga-do-pulgao-na-regiao-oeste-do-rn-e-safra-deve-ser-40percent-menor.ghtml, 2021.

[8] L. X. Boa Sorte, C. T. Ferraz, F. Fambrini, R. dos Reis Goulart, and

J. H. Saito, "Coffee leaf disease recognition based on deep learning and texture attributes," *Procedia Computer Science*, vol. 159, pp. 135–144, 2019, knowledge-Based and Intelligent Information Engineering Systems: Proceedings of the 23rd International Conference KES2019.

[9] A. Karlekar and A. Seal, "Soynet: Soybean leaf diseases classification," *Computers and Electronics in Agriculture*, vol. 172, p. 105342, 2020.

[10] J. Liu and X. Wang, "Plant diseases and pests detection based on deep learning: a review," *Plant Methods*, vol. 17, p. 22, 2021.

[11] S. Nalini, N. Krishnaraj, T. J. K. Vinothkumar, A. S. F. Britto, K. Subramaniam, and C. Bharatiraja, "Paddy leaf disease detection using an optimized deep neural network," *Computers, Materials & Continua*, vol. 68, no. 1, pp. 1117–1128, 2021.

[12] L. E. de Oliveira Aparecido, G. de Souza Rolim, J. R. da Silva Cabral De Moraes, C. T. S. Costa, and P. S. de Souza, "Machine learning algorithms for forecasting the incidence of coffea arabica pests and diseases," *International Journal of Biometeorology*, vol. 64, no. 4, pp. 671–688, Apr 2020.

[13] J.-R. Xiao, P.-C. Chung, H.-Y. Wu, Q.-H. Phan, J.-L. A. Yeh, and M. T.-K. Hou, "Detection of strawberry diseases using a convolutional neural network," *Plants*, vol. 10, no. 1, 2021.

[14] A. Afifi, A. Alhumam, and A. Abdelwahab, "Convolutional neural network for automatic identification of plant diseases with limited data," *Plants*, vol. 10, no. 1, 2021.

[15] I. C. d. e. a. SOUSA, "Genomic prediction of leaf rust resistance to arabica coffee using machine learning algorithms," *Scientia Agricola*, vol. 78, 2021.

[16] J. G. Esgario, R. A. Krohling, and J. A. Ventura, "Deep learning for classification and severity estimation of coffee leaf biotic stress," *Computers and Electronics in Agriculture*, vol. 169, p. 105162, 2020.

[17] R. Krohling, J. Esgario, and J. A. Ventura, "Bracol - a brazilian arabica coffee leaf images dataset to identification and quantification of coffee diseases and pests," https://www.doi.org/10.17632/yy2k5y8mxg.1, 2019.

[18] V. Sharma, "Deep learning – introduction to convolutional neural networks," https://vinodsblog.com/2018/10/15/everything-you-need-to-know-about-convolutional-neural-networks/, 2018.

[19] S. Chatterjee, "What is feature extraction? feature extraction in image processing," https://www.mygreatlearning.com/blog/feature-extraction-in-image-processing/, 2020.

[20] S. Sharma, "Activation functions in neural networks," https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6, 2017.

[21] J. Brownlee, "A gentle introduction to pooling layers for convolutional neural networks," https://www.geeksforgeeks.org/cnn-introduction-to-pooling-layer/, Machine Learning Mastery, 2019.

[22] A. Moawad, "Dense layers explained in a simple way," https://medium.com/datathings/dense-layers-explained-in-a-simple-way-62fe1db0ed75, Medium, 2019.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[24] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," https://www.tensorflow.org/, 2015.

[25] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[26] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[27] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.