Maize leaf disease classification using convolutional neural networks and hyperparameter optimization

Erik Lucas da Rocha, Larissa Ferreira Rodrigues, João Fernando Mari
Instituto de Ciências Exatas e Tecnológicas
Universidade Federal de Viçosa - UFV
Caixa Postal 22 - 38.810-000 - Rio Paranaíba - MG - Brasil
Email: {erik.rocha, larissa.f.rodrigues, joaof.mari}@ufv.br

Abstract—**Maize is an important food crop in the world, but several diseases affect the quality and quantity of agricultural production. Identifying these diseases is a very subjective and time-consuming task. The use of computer vision techniques allows automatizing this task and is essential in agricultural applications. In this study, we assess the performance of three state-of-the-art convolutional neural network architectures to classify maize leaf diseases. We apply enhancement methods such as Bayesian hyperparameter optimization, data augmentation, and fine-tuning strategies. We evaluate these CNNs on the maize leaf images from PlantVillage dataset, and all experiments were validated using a five-fold cross-validation procedure over the training and test sets. Our findings include the correlation between the maize leaf classes and the impact of data augmentation in pre-trained models. The results show that maize leaf disease classification reached 97% of accuracy for all CNNs models evaluated. Also, our approach provides new perspectives for the identification of leaf diseases based on computer vision strategies.**

Keywords—*Convolutional neural networks; maize leaf; classification; data augmentation; hyperparameter; Bayesian optimization.*

I. INTRODUCTION

By 2050 the number of people worldwide is expected to be almost 10 billion, driving the farm and food system. However, agricultural production has several limitations related to the degradation of agricultural land, water resources, climate change, and food losses [1].

Maize, popularly known as "corn", is the most produced food crop in the world, exceeding wheat and rice production [2]. Also, maize is a primary food used in several industry sectors to produce food, beverage, and cattle feed. Recently, the number of maize diseases and the degree of harm they cause have increased, mainly due to the degradation of agricultural land and changes in cultivation systems. Among the various diseases that affect maize plantations, leaf disease is one of the most critical and causes scaling down the crop yield and food nutritional value [3].

Visual analysis of patterns in leaves is the procedure used to identify leaf diseases in maize crops, but this process is very subjective and time-consuming. Moreover, the inaccurate identification of maize leaf diseases may lead to the wrong usage of pesticides, which reduces the quality and quantity of maize production, as well as health problems in humans [4].

The most promising technique for overcoming these limitations is the development of automatic systems based on computer vision to reduce losses and increase productivity [5].

Also, these techniques are financially attractive, especially for farms in emerging countries.

With advances in computational resources, deep learning models significantly outperform approaches based on hand-crafted features. In particular, Convolutional Neural Networks (CNN) provide automatic feature extraction from input images and demonstrate effective results in visual recognition tasks [6], [7]. Thus, CNNs can be used to identify maize crop diseases in the early stages, which can help improve the accuracy of plant protection and expand the use of technology in precision agriculture.

This paper identifies a suitable method based on CNNs for automatically classifying maize leaf diseases. Its main contributions are: (i) a comparison of the performance of three state-of-the-art CNN architectures in terms of accuracy, precision, recall, and F1-score; (ii) exploration of these CNNs with fine-tuning training; (iii) use of data augmentation strategies based on random rotations, vertical and horizontal flips, to overcome imbalance between the classes of the dataset.

The main novelty of this study is to find a suitable setup for hyperparameter optimization using Bayesian optimization, as finding the optimal hyperparameters to train CNN architectures is challenging due to the fact that there is no optimum method for the selection of hyperparameters. Also, the Bayesian optimization technique finds the best possible parameter setup faster than grid and random search.

To the best of our knowledge, no other study in the literature realizes such hyperparameter optimization considering different CNN architectures to classify maize leaf diseases. Our results suggest that hyperparameter optimization combined with fine-tuning training tends to be the best performing strategy to classify maize leaf diseases. In fact, our best result achieves an accuracy of 97%, which is very close to the highest accuracy score presented in the literature.

The remaining of this paper is organized as follows. Section II introduces the related work. Section III describes the material and methods. Section IV indicates and discusses the results. Section V presents conclusions and future work.

II. RELATED WORK

Considerable efforts have been dedicated to the development of automatic systems based on computer vision for crop disease identification. Mohanty et al. [5] evaluated two CNN architectures to classify 14 crop species and 26 leaf diseases

from PlantVillage dataset, obtaining 99.35% of accuracy. The same approach was adopted by Sladojevic et al. [8] to identify plant disease from healthy leaves. However, they considered leaf image takes from several datasets and achieved an average of 96.3% accuracy on their experimental analysis. Too et al. [9] performed a comparative analysis to classify 38 categories of plant disease with different pre-trained CNN models and achieved 99.75% of accuracy.

When considering maize leaf disease classification, DeChant et al. [10] proposed an automatic identification of northern leaf blight of maize and achieved 96.7% of accuracy. Zhang et al. [11] improved two CNNs architectures and applied data augmentation technique to classify eight kinds of maize leaf diseases. Lin et al. [12] designed a multi-channel CNN to classify five types of maize diseases using images takes from Shandong Province farming area. Alehegn et al. [13] developed a technique based on color, texture, and morphological features to classify maize leaf diseases taken from Ethiopia farming areas.

Bhatt et al. [14] proposed an approach based on CNNs and adaptive boosting with decision tree-based classifier to classify corn leaf diseases. The model developed by [14] reached an accuracy of 90% using Inception-v2 with Random Forest and the accuracy was improved to 98% using AdaBoost.

Priyadharshini et al. [15], also considered the same dataset considered in this paper. They proposed a method for the maize leaf disease classification using a CNN that shares the basic architecture of LeNet-5, all images were preprocessed using PCA whitening, and achieved an accuracy of 97.89%. Sibiya & Sumbwanyambe [16] proposed a system based on CNN to classify maize leaf diseases and obtained 92.85% of accuracy. Hu et al. [17] tested a model based on pre-trained GoogLeNet to classify maize leaf disease and obtained 97.60% of accuracy.

Recently, Waheed et al. [18] presented an optimized DenseNet to classify corn leaf disease. The authors used grid search to find the optimal hyperparameter values and the model was trained using different sets of hyperparameters. However, the grid search algorithm may have problems such as the curse of dimensionality, and lack of resources to handle the time-consuming operations [19]. In this paper, we used Bayesian optimization, which allows obtaining better results in fewer evaluations compared to grid search [20] [21].

In contrast to all previous works, in this work we consider hyperparameter optimization, data augmentation and training based on fine-tuning.

III. MATERIAL AND METHODS

The main purpose of this work is to provide a method able to classify maize leaf disease images using CNNs improved by Bayesian hyperparameter optimization. Fig. 1 illustrates the steps of the proposed method. It is composed of three main stages: a) the dataset splitting in training, validation and testing sets; b) k-fold cross-validation training of the CNNs architectures using Bayesian optimization for hyperparameters selection; and c) decision-making of the models with the testing phase.
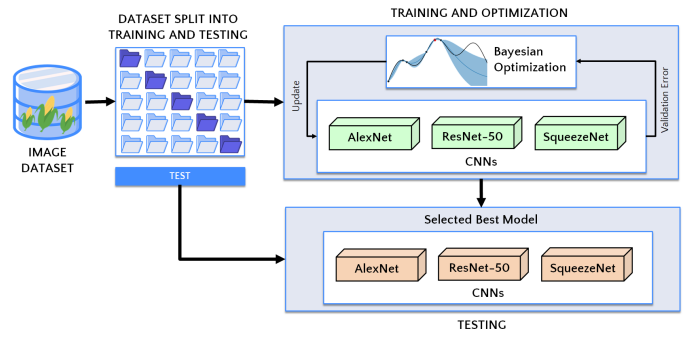

Fig. 1. Steps of proposed method.

A. Image dataset

The images used were taken from PlantVillage [1] dataset [5]. It contains 3852 images of maize leaf, each with a single leaf in evidence, categorized into one of four classes: gray leaf spot (513 images), common rust (1192 images), northern leaf blight (985 images), and healthy (1162 images). To illustrate the dataset, samples from each class are presented in Fig. 2.
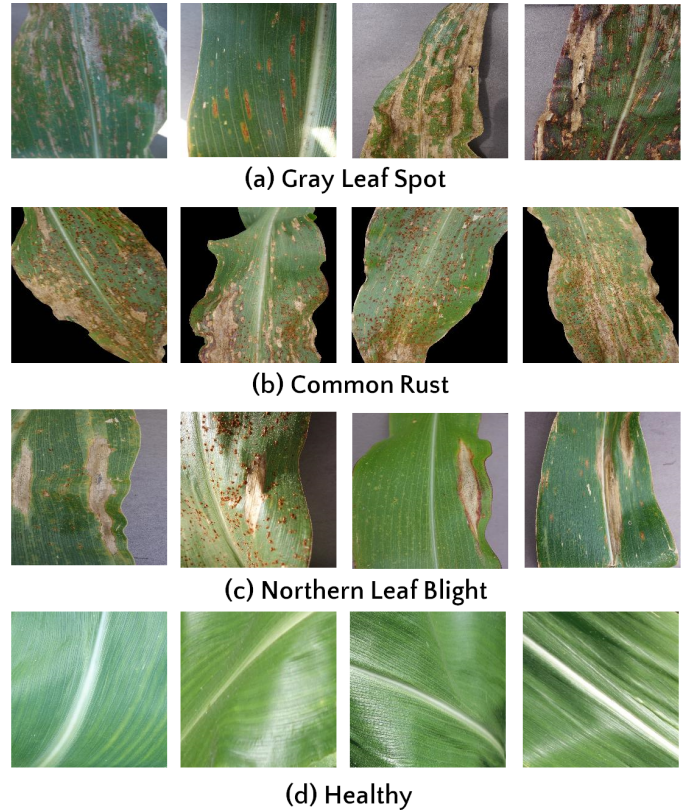

(a) Gray Leaf Spot

(b) Common Rust

(c) Northern Leaf Blight

(d) Healthy

Fig. 2. Examples of images for each class.

B. Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are the state of the art in image classification tasks, were designed to extract visual patterns directly from input images generating feature maps to the next deep layer [6] [7]. In this paper, we tested

---

[1] Available in: https://github.com/spMohanty/PlantVillage-Dataset

three CNN architectures: AlexNet [22], ResNet-50 [23], and SqueezeNet [24].

AlexNet [22] won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012. It is composed of five convolutional layers, three max-pooling layers, and two fully connected layers. Also, this architecture applies dropout regularization to reduce overfitting and ReLU activation to accelerate the training.

ResNet was proposed by He et al. [23] and won the ILSVRC 2015. It is utilizes residual blocks to address the gradient degradation in the training step. The different versions of ResNet have 18,50, 101, 152, and 201 layers. In the present study, we use the ResNet with 50 layers: one convolutional layer of size 7×7, followed by several convolutional layers of size 3×3 and 1×1.

SqueezeNet [24] architecture requires fewer parameters and provides performance equivalent to AlexNet. SqueezeNet is composed mainly of fire blocks that are squeeze convolution layers of size 1×1, which goes into two expand layers, one with a filter size of 1×1 and the other has a size of 3×3. The feature maps obtained from the two expand layers feed into a concatenation layer, being input to the squeeze layer that outputs to the following fire module.

C. Hyperparameter Optimization

Hyperparameters are essential in deep learning algorithms since those parameters define the details of training and affect the performance of the models significantly [25]. The choice of values for the hyperparameters is a crucial task as there is no optimum method for the selection of hyperparameters. The choice of hyperparameters values is represented as an optimization problem, where the objective function is unknown (it is a black-box function) and the hyperparameters are defined as decision variables. The fine-tuned hyperparameters in this paper are as follows:

- Batch size: the batch size is the number of images that will be propagated through the CNN. A large batch size requires less RAM and GPU consumption but could result in a less accurate estimate of the gradient. On the other hand, a smaller batch size requires more RAM and GPU consumption, and fewer groups (batches) will propagate on CNN [26].

- Learning rate: the learning rate defines the level of adjustments of weight connections and network topology, applied at each training epoch, being the main parameter to tuning. This hyperparameter is optimized in order to improve the runtime when using Stochastic Gradient Descent (SGD) optimizer. A high learning rate may sacrifice the accuracy generating a lack of precision. On the other hand, a small learning rate requires more epochs of training to learn the difference between features [25].

- Momentum: the momentum coefficient is necessary to stop the oscillations in the regions of high curvature of the loss function generated by the SGD optimizer [7].

We employed the optimization in order to minimize the objective function, i.e., the cross-entropy function (loss function). The loss function was minimized by SGD optimizer with momentum.

D. Bayesian Optimization

Bayesian optimization [27] is an efficient algorithm composed by four parts: i) an objective function that defines what will be optimized; ii) the performance of the model that varies according to the hyperparameters setup; iii) the hyperparameter space search, which is a list of possible solutions; and iv) optimizer algorithm [20].

The objective function is unknown and is only defined after the setup definition, allocating, training, and testing the model. Therefore, the method considers the objective function as a random function. The Bayesian optimization is called Bayesian because the optimization strategy uses the Bayes' theorem. In this context, given the evidence provided by data D, the posterior probability $P(m|D)$ where m is a model proportional to the probability $P(D|m)$ of overserving D given model m multiplied by the prior probability of $P(m)$, defined in Equation 1.

$$P(m|D) = \frac{P(D|m)P(m)}{P(D)} \qquad (1)$$

In this study, we used Gaussian Process (GP) as the probabilistic model [28]. GP is a model that generates data located throughout some domain (here, the hyperparameters). This method performs a hypothesis about unknown parameters, based on parameters already found. The choices of the Bayesian approach increasing the time for selecting hyperparameters but reduces the time need in the evaluation of the objective function, requiring a less computational cost. Therefore, Bayesian Optimization is high efficiency compared to grid and random searches [20] [21].

E. Models Training

The classification validation was performed by stratified k-fold cross-validation [29]. The dataset was randomly partitioned into six stratified sub-datasets. Of the six sub-datasets, five sub-datasets were used for cross-validation, and a single sub-dataset is retained as the test data for testing the model. The cross-validation process is then repeated five times, with each of the k sub-datasets used exactly once as the validation data. The five validation results were averaged to identify the best model. After identifying the best model, we tested this model using the single sub-dataset retained as the test data.

Fig. 3 shows the overall process of selecting optimal hyperparameters. First, we separate the maize leaf disease dataset into learning and test data. After, the learning data is separated into training and validation data, and k-fold cross-validation based on a Bayesian optimization is carried. It performs the verification of the model on the basis of the preset hyperparameter and the k-fold value. Finally, the model is trained by using the optimal hyperparameters found by Bayesian optimization. Also, in order to analyze the general classification performance, we have chosen the macro-avg evaluation, which makes an averaging calculation by class

and measured using the test data and considering accuracy, precision, recall, and F1-score indices, computed from the confusion matrix [30].
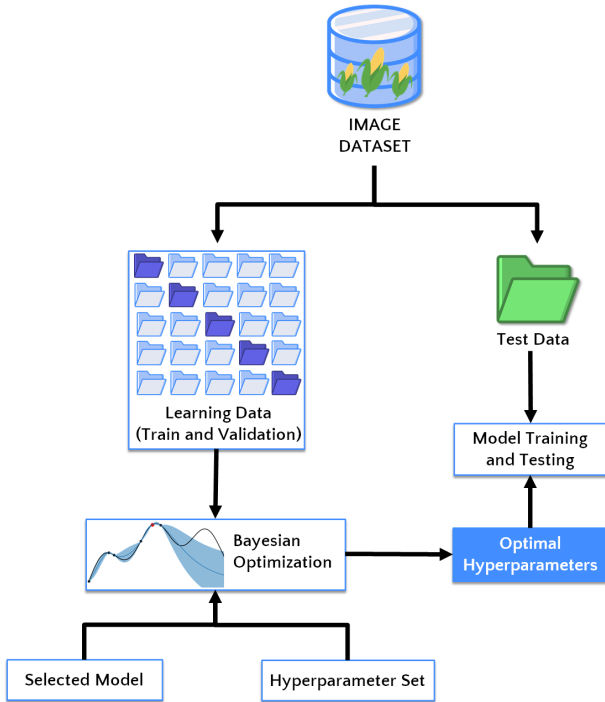

Fig. 3. Process of hyperparameter selection and model evaluation with the k-fold cross-validation.

For the hyperparameters of each CNN model, we consider: (i) batch size; (ii) learning rate; and (iii) momentum (as described in Section III-C). The values used to perform the Bayesian optimization for the hyperparameters are listed in Table I and all hyperparameters are searched considering a uniform distribution.

TABLE I. Hyperparameter search space used for Bayesian optimization.

Hyperparameter	Value
Batch Size	[16, 32]
Learning Rate	[0.001, 0.01]
Momentum	[0, 1]

The best values found through the cross-validated Bayesian optimization process are depicted in Table II.

IV. Results and Discussion

In this section, we compare the model prediction performance across each CNN evaluated.

A. Experimental Environment

All experiments were executed on a machine with an Intel i5 3.00 GHz processor, 16 GB RAM, a GPU NVIDIA GeForce GTX Titan Xp with 12 GB memory, and operating system Ubuntu 16.04.2 LTS. The models were programmed using Python (version 3.6) and PyTorch (version 1.4) deep learning

framework [31] with CUDA version 8.0 and cuDNN 6.0. The hyperparameter optimization algorithm Bayesian was drawn from the *bayesian-optimization*[2] library, version 1.2.0.

According to the process shown in Fig. 3, the five-fold-cross-validation Bayesian optimization was performed and the optimal hyperparameters for each CNN are found through each model tuning. The training was performed using the optimal hyperparameters (see Table II) defined for each CNN and the learning epochs were set to 30. All CNNs evaluated were previously trained using ImageNet dataset [32], adjusting the parameters in the deepest layers. Also, we resized all images to 224×224 pixels to adapt for the input of each CNN architecture evaluated, and we applied data augmentation strategies [22] in the training images by random rotation (considering angles between $0°$ and $360°$), vertical and horizontal flips.

B. Experimental Results

The experiment is performed to compare the performance of each CNN to classify the maize leaf disease dataset. Tables III, IV, and V presents the accuracy, precision, recall, and F1-score obtained when testing each CNN with optimal hyperparameters defined by Bayesian Optimization. Interestingly, the three CNNs achieved 97% of accuracy and this indicates that optimization allowed a better generalization in all models. Also, our best result of 97% accuracy was better or very close to the accuracy scores presented in the literature (92.85% in Sibiya & Sumbwanyambe [16], 97.60% in Hu et al. [17], and 97.89% in Priyadharshini et al. [15], 98% in Bhatt et al. [14]).

We observed that when comparing the overall performance of all four classes, the class gray leaf spot showed slightly lower performance indices. This is due to the imbalance between the classes: the gray leaf spot is the smallest class, about 43% of the size of the largest class (common rust). To overcome the imbalance, we applied data augmentation strategies, which not allowing a significant decreasing in the classification performance.

In addition to classification evaluation, we also analyzed the correlation between each of the four maize leaf classes registered from experimental results considering the testing set, as shown in Fig. 4. In this representation, we observed that there are a few correlations, which are categorized into weak (≤ 0.39) and moderate (≥ 0.40). We will focus our analysis on the moderate correlation, which is of most interest in our study. As can be seen in the Fig. 4, the correlations between the gray leaf spot, common rust, and healthy classes have a moderate intensity of the correlation for ResNet-50 and SqueezeNet.

The correlation between the common rust and gray leaf spot classes indicates that when a leaf is common rust, there is a possibility of classifying it as a gray leaf spot. And, the correlation between the gray leaf spot and healthy leaf indicates that there are situations where gray leaf spot is classified as healthy. This result suggests that there are some cases in which the patterns between the leaves are similar, although the leaves are of different classes.

Although the studies proposed by [14], [15], [16], and [17] use the same dataset considered in our study, they considered a hold-out validation technique, which generates biased sets

---

[2]https://pypi.org/project/bayesian-optimization/

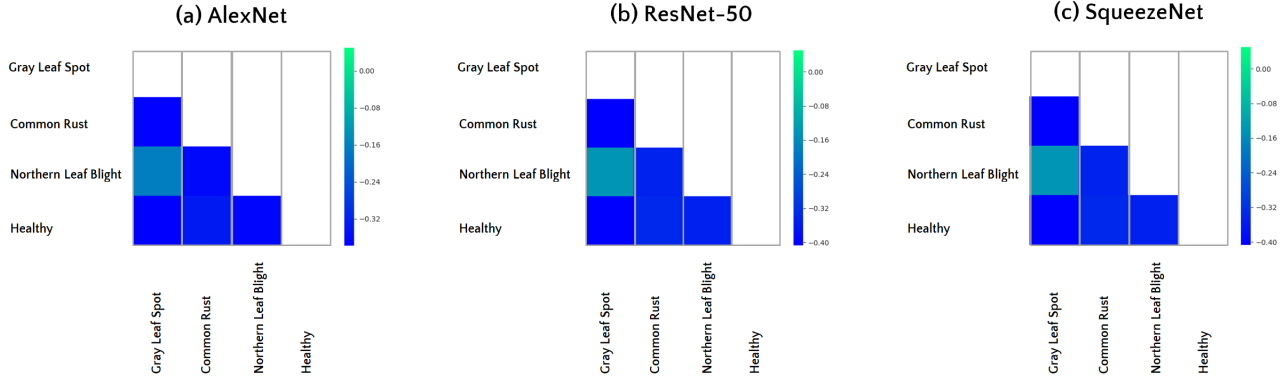| Hyperparameter	AlexNet	ResNet-50	SqueezeNet
Batch size	32	32	18
Learning rate	0.0036931912206946992	0.0042498507802230421	0.0026763419023990384
Momentum	0.13865257470287484	0.4755170748794698	0.34556072704304774

Fig. 4.   The correlation between four maize leaf classes considering each CNN evaluated.

TABLE III.   TESTING PERFORMANCE FOR ALEXNET ARCHITECTURE.

	Precision	Recall	F1-Score
Gray Leaf Spot	91%	85%	88%
Common Rust	100%	99%	100%
Northern Leaf Blight	92%	95%	93%
Healthy	99%	100%	100%
Average	96%	95%	95%
Accuracy		97%	

TABLE IV.   TESTING PERFORMANCE FOR RESNET-50 ARCHITECTURE.

	Precision	Recall	F1-Score
Gray Leaf Spot	86%	93%	89%
Common Rust	100%	99%	100%
Northern leaf blight	96%	91%	93%
Healthy	99%	100%	100%
Average	95%	96%	96%
Accuracy		97%	

TABLE V.   TESTING PERFORMANCE FOR SQUEEZENET ARCHITECTURE.

	Precision	Recall	F1-Score
Gray Leaf Spot	86%	93%	89%
Common Rust	100%	99%	100%
Northern leaf blight	96%	91%	93%
Healthy	99%	100%	100%
Average	95%	96%	96%
Accuracy		97%	

and unexpected values of accuracy. In contrast, we adopted k-fold cross-validation technique to better estimate the accuracy of the studied CNN architectures, which is more robust to outliers and eventual overfitting.

The general quality of our optimized models was estimated using the F1-Score. This metric is an excellent alternative to deal with the imbalance between the classes because it is the harmonic average between precision and recall calculations. Therefore, in terms of F1-Score, the best result was obtained by ResNet-50 and SqueezeNet models (96%).

V.   CONCLUSION

The results presented in this study allow us to conclude that for maize leaf disease classification, the use of CNNs improved through Bayesian hyperparameter optimization is a promising alternative. Based on the comparative analysis, we could conclude that our best result of 97% was better or very close to the accuracy scores presented in the literature. Moreover, it is important to stress that our method is validated using a cross-validation strategy, which is more robust to over-fitting and generates results more reliable. Our results suggest that hyperparameter optimization improved the performance of all CNNs evaluated. The models generated have been able to extract important features about visual patterns of maize leaf. Although the main focus of this study is to classify maize leaf diseases, a classifier system that identifies with high performance a healthy leaf is attractive for farmers to manage the need resources on the crop.

We believe that our approach requires less time investment in a real-world context because the maize leaves can be acquired without the need to place them on a homogeneous background. Thus, this study is suitable for farmers looking for early detection or breeders evaluating the incubation period for a given disease.

As future work, we hope to apply our approach to classify more types of maize leaf and other types of leaf diseases, evaluate further optimization algorithms, and exploit more data augmentation strategies.

ACKNOWLEDGMENTS

REFERENCES

[1] Nikos Alexandratos and Jelle Bruinsma. World agriculture towards 2030/2050: the 2012 revision. Technical report, Jun 2012. FAO Document Repository: http://www.fao.org/3/a-ap106e.pdf.

[2] World Health Organization et al. *The state of food security and nutrition in the world 2018: building climate resilience for food security and nutrition*. Food & Agriculture Org., 2018.

[3] Bekele Shiferaw, Boddupalli M. Prasanna, Jonathan Hellin, and Marianne Bänziger. Crops that feed the world 6. past successes and future challenges to the role played by maize in global food security. *Food Security*, 3(3):307, Aug 2011.

[4] Jayme Garcia Arnal Barbedo. A review on the main challenges in automatic plant disease identification based on visible range images. *Biosystems Engineering*, 144:52 – 60, 2016.

[5] Sharada P. Mohanty, David P. Hughes, and Marcel Salathé. Using deep learning for image-based plant disease detection. *Frontiers in Plant Science*, 7(September):1–10, 2016.

[6] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[7] M. A. Ponti, L. S. F. Ribeiro, T. S. Nazare, T. Bui, and J. Collomosse. Everything you wanted to know about deep learning for computer vision but were afraid to ask. In *2017 30th SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T)*, pages 17–41, Oct 2017.

[8] Srdjan Sladojevic, Marko Arsenovic, Andras Anderla, Dubravko Culibrk, and Darko Stefanovic. Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. *Computational Intelligence and Neuroscience*, 2016, 2016.

[9] Edna Chebet Too, Li Yujian, Sam Njuki, and Liu Yingchun. A comparative study of fine-tuning deep learning models for plant disease identification. *Computers and Electronics in Agriculture*, 161:272 – 279, 2019. BigData and DSS in Agriculture.

[10] Chad DeChant, Tyr Wiesner-Hanks, Siyuan Chen, Ethan L. Stewart, Jason Yosinski, Michael A. Gore, Rebecca J. Nelson, and Hod Lipson. Automated identification of northern leaf blight-infected maize plants from field imagery using deep learning. *Phytopathology®*, 107(11):1426–1432, 2017. PMID: 28653579.

[11] X. Zhang, Y. Qiao, F. Meng, C. Fan, and M. Zhang. Identification of maize leaf diseases using improved deep convolutional neural networks. *IEEE Access*, 6:30370–30377, 2018.

[12] Zhongqi Lin, Shaomin Mu, Aiju Shi, Chao Pang, Xiaoxiao Sun, et al. A novel method of maize leaf disease image identification based on a multichannel convolutional neural network. *Transactions of the ASABE*, 61(5):1461–1474, 2018.

[13] Enquhone Alehegn. Ethiopian maize diseases recognition and classification using support vector machine. *International Journal of Computational Vision and Robotics*, 9(1):90–109, 2019.

[14] Prakruti Bhatt, Sanat Sarangi, Anshul Shivhare, Dineshkumar Singh, and Srinivasu Pappula. Identification of diseases in corn leaves using convolutional neural networks and boosting. In *Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods - Volume 1: ICPRAM,*, pages 894–899. INSTICC, SciTePress, 2019.

[15] Ramar Ahila Priyadharshini, Selvaraj Arivazhagan, Madakannu Arun, and Annamalai Mirnalini. Maize leaf disease classification using deep convolutional neural networks. *Neural Computing and Applications*, 31(12):8887–8895, Dec 2019.

[16] Malusi Sibiya and Mbuyu Sumbwanyambe. A computational procedure for the recognition and classification of maize leaf diseases out of healthy leaves using convolutional neural networks. *AgriEngineering*, 1(1):119–131, 2019.

[17] Rongjie Hu, Shan Zhang, Peng Wang, Guoming Xu, Daoyong Wang, and Yuqi Qian. The identification of corn leaf diseases based on transfer learning and data augmentation. In *Proceedings of the 2020 3rd International Conference on Computer Science and Software Engineering*, CSSE 2020, page 58–65, New York, NY, USA, 2020. Association for Computing Machinery.

[18] Abdul Waheed, Muskan Goyal, Deepak Gupta, Ashish Khanna, Aboul Ella Hassanien, and Hari Mohan Pandey. An optimized dense convolutional neural network model for disease recognition and classification in corn leaf. *Computers and Electronics in Agriculture*, 175:105456, 2020.

[19] Ian Dewancker, Michael McCourt, Scott Clark, Patrick Hayes, Alexandra Johnson, and George Ke. A stratified analysis of bayesian optimization methods, 2016.

[20] James S. Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2546–2554. Curran Associates, Inc., 2011.

[21] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 2951–2959. Curran Associates, Inc., 2012.

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[24] Forrest N. Iandola, Song Han, Matthew W. Moskewicz, Khalid Ashraf, William J. Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5mb model size, 2016.

[25] Yoshua Bengio. Practical recommendations for gradient-based training of deep architectures. In *Neural Networks: Tricks of the Trade: Second Edition*, pages 437–478, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[26] Alex Krizhevsky. One weird trick for parallelizing convolutional neural networks. *CoRR*, abs/1404.5997, 2014.

[27] Martin Pelikan, David E. Goldberg, and Erick Cantú-Paz. Boa: The bayesian optimization algorithm. In *Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation - Volume 1*, GECCO'99, page 525–532, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[28] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[29] Pierre A. Devijver and Josef Kittler. *Pattern Recognition: A Statistical Approach*. Prentice-Hall, 1982.

[30] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification (2Nd Edition)*. Wiley-Interscience, New York, NY, USA, 2000.

[31] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8026–8037. Curran Associates, Inc., 2019.

[32] J. Deng, W. Dong, R. Socher, L. J. Li, Kai Li, and Li Fei-Fei. Imagenet:

A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, June 2009.