

Original Research Paper

Experimental Evaluation of Coffee Leaf Disease Classification and Recognition Based on Machine Learning and Deep Learning Algorithms

^{1,2,3}Kacoutchy Jean Ayikpa, ²Diarra Mamadou, ¹Pierre Gouton and ²Kablan Jérôme Adou

¹Laboratoire ImViA, Université Bourgogne Franche-Comté, France

²Laboratoire Mécanique et Informatique, Université Felix Houphouët-Boigny, Cote D'Ivoire

³Unité de Recherche et d'Expertise Numérique, Université Virtuelle de Côte d'Ivoire, Abidjan, Côte d'Ivoire

Article history

Received: 04-09-2022

Revised: 17-11-2022

Accepted :14-11-2022

Corresponding Author:
Kacoutchy Jean Ayikpa
Laboratoire ImViA, Université
Bourgogne Franche-Comté,
France
Email: ayikpajean@yahoo.fr

Abstract: Coffee plant diseases constitute a significant danger to world coffee production, and the greatest challenge is to detect these diseases as early as possible to save the crop. Traditional methods are most often based on visual observations, often with errors in diagnosing diseases. Machine Learning has become a tool that presents itself as an alternative for automatically identifying plant diseases. Our study is to implement a robust method of classification and recognition of coffee leaf diseases using both classical machine learning and deep learning methods, so we set up a custom CNN. These methods were evaluated on the Arabica coffee leaf dataset known as JMuBEN. The results of the classical machine learning methods ranged from 81.03 to 100% and the best performance was obtained with SVM and Random Forest; while the deep learning. In comparison, these provided results between 97.37 and 100% with our CNN custom obtaining receiving accuracy with the lowest loss of 0.013%. Accuracy, precision score, recall, and MCC were employed as performance indicators to support this performance.

Keywords: Machine Learning, Coffee Leaf Diseases, Deep Learning, Computer Vision

Introduction

Agriculture is the sector in which we find the exploitation of the land to obtain crops and also the animal exploitation called breeding. It is confronted at all times with diseases that can jeopardize production. Diseases negatively affect plants and animals and hinder market access for agricultural production. Among these diseases, plant diseases are a real obstacle in terms of yield and quality for producers (Crop Diseases, 2021). There are various categories of plant diseases, including fungal, oomycete, hyphomycete, bacterial, and viral (Lu *et al.*, 2021). Coffee is one of the most consumed products with a world production of over 7 million tons per year (Planetoscope-Statistiques, 2021). Kenya, the second largest African producer of Arabica coffee, is confronted with diseases that attack coffee leaves, which affects its production (Quels sont les meilleurs cafés d'Afrique, 2020). These diseases are cercospora, coffee rust, and phoma (Jepkoech *et al.*, 2021). Identifying diseases and knowing when and how to deal with them effectively is an ongoing challenge for growers. Disease symptoms can appear on the leaves at any stage of the plant's growth.

Early detection and recognition of diseases can minimize crop losses. The process of manual detection of plant diseases is not reliable enough and is very difficult on large farms (Giraddi *et al.*, 2020). Detection using computer vision for plant disease detection will enable experts to take preventive measures for early detection. Detection and classification of coffee leaf disease have important economic and technical significance in agriculture. In recent years, computer vision has been used for object detection and image classification and has made tremendous progress (Jain *et al.*, 2022). We can note several techniques that have been proposed that enable plant disease diagnosis. With a 93% overall accuracy, (Ramcharan *et al.*, 2017) used transfer learning to train a deep learning model to recognize three diseases and two different forms of pest damage. The SSD model, which (Saleem *et al.*, 2020) implemented and showed to have the best average precision of all deep neural networks i.e., 73.07% (Afifi *et al.*, 2020) recommends using Convolutional Neural Networks (CNNs) for their ability to use prior learning weights, three deep

learning architectures are selected such as ResNet18, ResNet34 and ResNet50 to build two basic models, a Triplet network and a Deep Adversarial Metric Learning (DAM) network. These methods achieve 81 to 99% accuracy (Xiao *et al.*, 2020) and propose a convolutional neural network. In this case, ResNet50 uses a method based on two separate data sets that include both source and feature images. For leaf blight affecting the crown, leaf, and fruit, the classification performance is 100%; for grey mold and powdery mildew, it is both 98% (Yang *et al.*, 2021) used a fusion module would merge the image features detected by the convolutive imbalance module and ensure feature extraction from the unbalanced data set. Their findings showed that their method performed better than cutting-edge techniques, with an accuracy of 97.58% on the set of images for rice pests and diseases (Shao *et al.*, 2021) proposed a method that merges the LC-FCN model based on pre-trained models with the watershed algorithm for dense rice image recognition with an accuracy of 89.88% DAWI and Wulan propose research doing a method using wavelets to denoise the images of the data set and then performs the classification of these images with convolution method. The results obtained using the wavelet method and convolutional neural network gave an accuracy of 97% (Dewi and Utomo, 2021; Yu *et al.*, 2019) and propose a new approach for the detection of apple leaf diseases using deep learning considering regions of interest. They designed two sub-networks in the first step. First, the input image is divided into three parts which are the background, the foliage, and the spots indicating diseases on the leaf, which is the region of interest, then they use pre-trained algorithms to train separately with a new set of types containing information about the class. This method named ROI-aware DCNN gives an accuracy of 84.3% Yu and Son in their study proposed an LSA-Net method to recognize the foliar diseases of apple trees. The LSA-Net consists of two sub-networks, the first is a feature segmentation sub-network and the second is a classification, this method gave an accuracy of 89.4% (Yu and Son, 2020). Ahmed and Reddy presented a mobile system for automation to diagnose leaf diseases. The developed system uses convolution neural networks as the underlying deep learning algorithm to classify 38 disease categories and achieves an overall accuracy of 94% (Ahmed and Reddy, 2021; Velásquez *et al.*, 2020) used their work diagnostic model of the state of development of CLR in the crop at the scale of the *Coffea arabica* variety, Caturra, through the use of remote sensing techniques with multi-spectral cameras adapted to

drones and deep learning techniques. Their diagnostic model obtained an F1 score of 77.50%.

Our study consists the first time using classical machine learning algorithms such as K-Nearest Neighbors (K-NN), Support Vector Machine (SVM), Logic Regression, Random Forest, Multi-layer Perceptron Classifier (MLPClassifier), Decision Tree Classifier (DTClassifier), Gaussian Naive Bayes (GNB) and in a second time Convolutional neural networks like MobileNet, VGG-19, Inception-V3, DenseNet-201, ResNet-50, and eventually, our Custom CNN were used to apply deep learning methods more specifically. Finally, a comparison will be done to show which models in our study are the most reliable.

Materials and Methods

Dataset

The database used for our study is the Arabica coffee leaf dataset called JMuBEN, it consists of coffee leaf images from a plantation in Kirinyaga County, Kenya. The first folder includes 7682 images of *Cercospora*, the second contains 8337 images of rust, and the third has 6572 images of *Phoma*. We also added a folder of 8927 healthy images from the JMuBEN2 folder. In sum, the dataset of our study contains 31518 images of coffee leaves divided into four classes, this case, *Phoma*, *Cercospora*, Rust, and healthy (Jepkoech *et al.*, 2021). Figure 1 shows images of different coffee leaf diseases.

Our dataset will be divided as shown in Table 1 below:

Machine Learning

Machine learning is a subfield of artificial intelligence that allows a machine to have a learning capacity, without having been programmed for a specific task. Machine Learning is generally classified into three categories namely (Shobha and Rangaswamy, 2018).

Supervised Learning

In supervised learning, the system is helped in its learning, it is presented with input data and the outputs that we want to have. The system will then generalize what it has learned for the other unlearned data. The algorithm will develop a function that will make an accurate prediction of the output from the input data. Supervised learning is divided into two categories in this case:

- Regression: In this case, the output variable is a category
- Classification: The output variable in this instance is a numerical value

These main algorithms are Random forests, decision trees, the k-Nearest Neighbor (k-NN) method, linear regression, naive Bayesian classification, Support Vector Machine (SVM), logistic regression and gradient boosting, etc.

Table 1: Distribution of our dataset.

Dataset	Training set	Validation set	Test set
100%	60%	20%	20%
31518	18909	6303	6303

Table 2: General description of the parameters

Parameters	Values
Optimizer	SGD
Learning rate	0.0001
Momentum	0.9
Epochs	20
Batch size	32

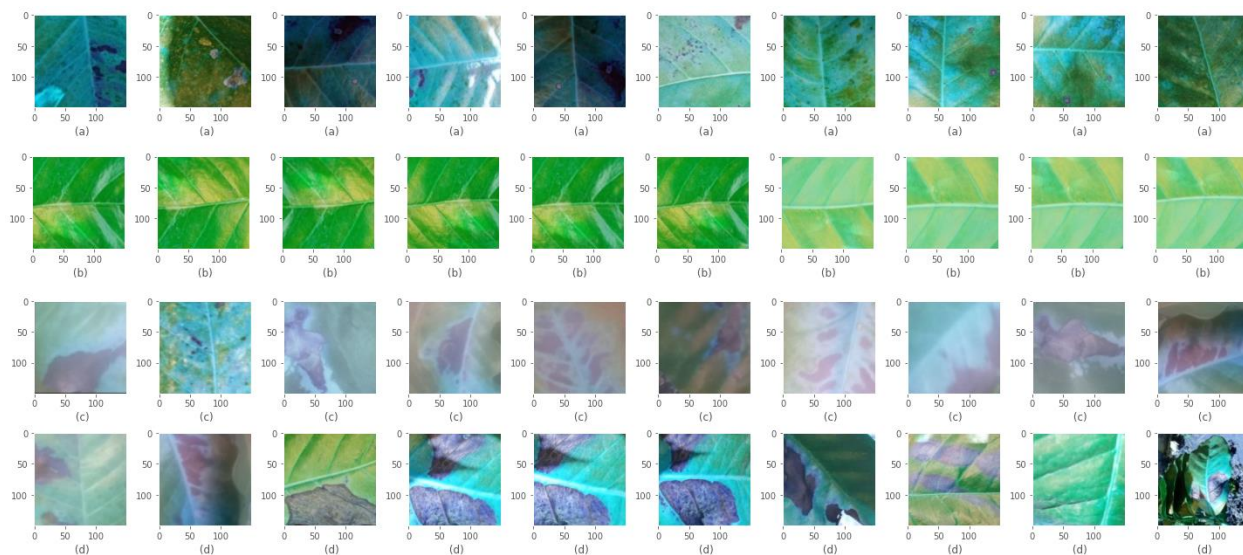


Fig. 1: Image samples of different coffee leaf diseases: (a) Cercospora; (b) Healthy; (c) Leaf rust; (d) Phoma

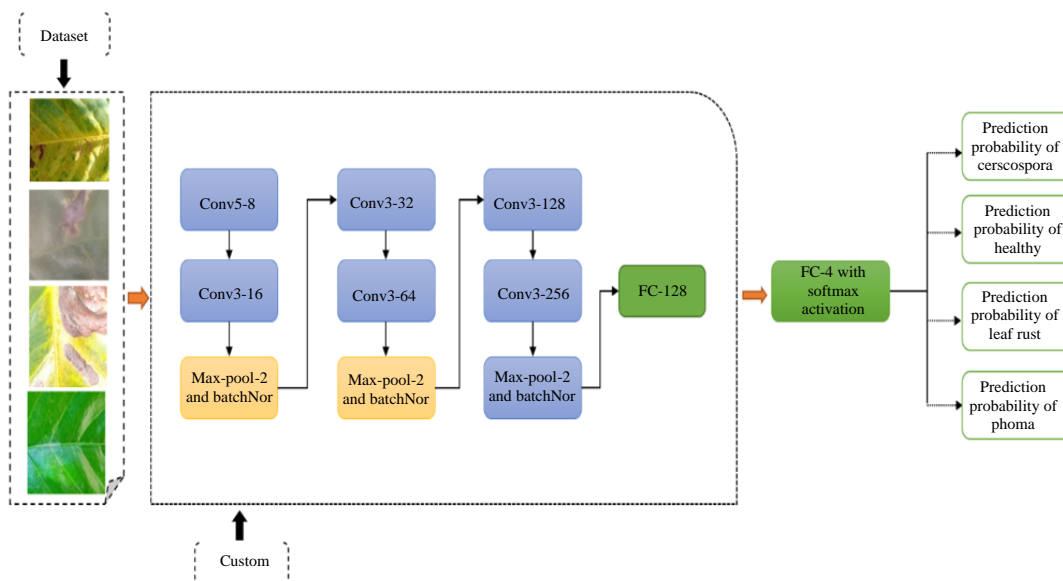


Fig. 2: The architecture of Custom CNN applies to our study

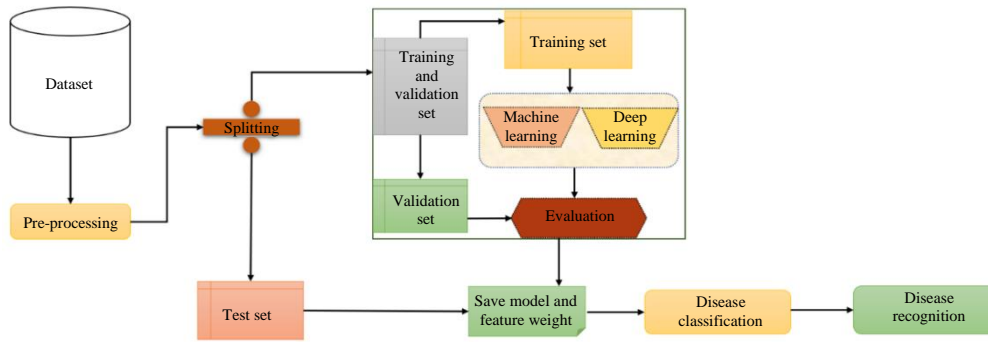


Fig. 3: General architecture of our study method



Fig. 4: K-fold illustrative diagram.

Unsupervised Learning

In unsupervised learning, the system receives no help in determining the structure of the input data. The algorithm will seek to discover features in the data to achieve a certain goal. This approach is also called feature learning. These main algorithms are: K-Means, hierarchical clustering/grouping, and dimensionality reduction

Reinforcement Learning

In reinforcement learning, the system learns by interacting with the surrounding environment. At any given moment, the system obtains the current state and all possible actions. It performs one of the actions and receives a feedback signal that notifies it of its new state and the associated reward. Through iteration, the system should be able to automatically determine the ideal behavior (the one that maximizes the rewards) for a specific context. This type of learning is also called "semi-supervised" in the sense that the reward indicates the right result to achieve

For our study, we will use supervised machine learning algorithms such as Support Vector Machine

(SVM), k-Nearest Neighbors (K-NN), Random Forest (RF), Logistic Regression (LR), Multi-layer Perceptron Classifier (MLP), Decision Tree Classifier (DTClassifier) and Gaussian Naive Bayes (GaussianNB).

Support Vector Machine (SVM)

SVMs are used for classification, regression, and outlier detection. In a high- or infinite-dimensional space, the SVM creates a hyperplane or group of hyperplanes that can be utilized for classification, regression analysis, and other activities. It makes sense that the hyper-plane with the greatest distance from any class's closest training data points will accomplish a good separation, as in general, the higher the margin, the smaller the classifier's generalization error is. (Support Vector Machines, 2021). SVM solves the following equation:

$$\min \frac{1}{2} \omega^T \omega + C \sum_{i=1}^m \zeta_i$$

$$\text{subject to } y_i(\omega^T \Phi(x_i) + b) \geq 1 - \zeta_i$$

$$\zeta_i \geq 0, i=1, \dots, n$$
(1)

K-Nearest Neighbors (K-NN)

K-NN is a non-parametric classification algorithm that is based on training data. It uses distance calculation to measure the similarity of the test data to the data used for training. Subsequently, the test data is classified by a majority vote of the k nearest neighbors of the training set (Akbulut *et al.*, 2017). The distance calculation used is the Euclidean distance, its formula is as follows:

$$De(x, y) = \sqrt{\sum_{j=1}^n (x_j - y_j)^2} \quad (2)$$

When using KNN, we considered 5 clusters.

The RF is an algorithm based on the assembly of independent decision trees. Each tree has an autonomous view of the problem due to a double random draw, namely tree bagging and feature sampling. All three decisions are federated and the decision taken by the RF for test data is then the vote of all trees. The error rate of Random Forest is a function of the correlation between two trees, and the accuracy of each tree (Alam and Vuong, 2013).

Logistic Regression (LR)

The LR is a binomial regression model. The aim is to model as well as possible a simple mathematical model to numerous real observations. Logistic regression is a special case of a generalized linear model to which we apply the sigmoidal function. To produce the logistic regression equation, the maximum likelihood ratio is used to determine the statistical significance of the variables (Kurt *et al.*, 2008).

Multi-layer Perceptron Classifier (MLP Classifier)

MPL Classifier is an artificial neural network algorithm organized into multiple layers within which information flows from the input layer to the output layer only. The MPLClassifier can be defined as a multi-layered directed graph with several hidden layers and fully connected layers. Supervised learning using the backpropagation algorithm is more often used to train the MPLClassifier. The MPLClassifier is an evolution of the single-layer perceptron correcting its weaknesses (Wan *et al.*, 2018).

Decision Tree Classifier (DT Classifier)

DTClassifier is a technique using a decision tree from training data. A decision tree is a predictive model that is a correspondence between observations about an item and conclusions about its target value. In tree structures, leaves represent classifications (also called labels), unleafed nodes are features, and branches represent feature conjunctions that lead to the classification. Decision tree classifiers are also known as multilevel classifiers (Delbarre *et al.*, 2021).

Gaussian Naive Bayes (GNB)

Naïve Bayes is a classifier based on a generative model with a fast learning and testing process. Bayesian classifiers, work based on Bayesian rule and probability theorems. A simplified version of the Bayesian classifier called naive Bayes uses two assumptions. Naive Bayes classification is a case of the naive Bayes method with the assumption of the existence of a Gaussian distribution on the attribute values given the class label (Jahromi and Taheri, 2017). Its general formula is:

$$p(x_i|y) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x_i - \mu_y}{2\sigma^2}\right) \quad (3)$$

Deep Learning

A machine learning method called deep learning uses artificial neural networks. Deep learning is used in many applications ranging from natural language to computer vision processing. There are different types of deep learning architectures namely: Deep Neural Networks (DNN), Deep Belief Networks (DBN), recurrent neural networks (RNN), and Convolutional Neural Networks (CNN). Deep learning has been successfully applied in several research areas, in this case, agriculture, health, education, environment, and health (Santos *et al.*, 2019). In our study, we will use convolutional neural networks. CNN is a deep learning algorithm that is widely used in computer vision, including natural language processing, speech recognition, face recognition, etc. It is composed of three main layers namely the convolution layer, pooling layer, and full layers (Davuluri and Rengaswamy, 2022)

Convolution Layer

The convolution layer is a key element of convolutional neural networks. Its role is to detect the presence of feature points in the images it receives. First of all, it performs convolution filtering. It proceeds by calculating the convolution of each image with each filter. We then obtain an activation map, which locates the characteristic points in the image.

Pooling Layer

Pooling is an operation that consists of reducing the size of the feature points output from the convolution phase. It proceeds by slicing the image into regular cells, then keeping within each cell the maximum value. The most common choices are adjacent cells of size 2×2 pixels that do not overlap, or cells of size 3×3 pixels, spaced from each other by a step of 2 pixels. Generally, this layer is positioned between two convolution layers. The output is the same feature points but smaller in size.

Fully Connected Layer

Fully connected layers receive a vector as input and produce an output as a vector. They apply a linear combination and an activation function to the values. They allow us to classify the images received as input by providing a vector of sizes equivalent to the number of classes in our problem. Each column of the vector indicates the probability of the data belonging to a given class (Aburass *et al.*, 2022).

We will use the following algorithms in the untrained form for convolutional neural networks: MobileNet, VGG-19, Inception-V3, DenseNet-201, ResNet50, and customized CNN.

MobileNet

MobileNet uses depth-separable convolutions. It can reduce the number of parameters without losing accuracy. This allows for lightweight deep neural networks. It is composed of two main layers: Point layers (pw) and deep layers (dw). MobileNet has been made freely available by Google. The deep layers are convoluted with a core size of 3×3 and the point layers are also convoluted with a core size of 1×1 . These layers use the rectified linear unit activation function and the batch normalization algorithm. MobileNet has 19 deep layers (Attallah, 2021).

VGG-19

VGG-19 is a variant of the visual geometry network VGGNet which is a deep neural network with a multilayer. It was created at the University of Oxford. VGG-19 consists of 19 layers which use convolutional layers with a 3×3 size core with max-pooling layers. It has two fully connected layers, each with 4096 nodes, which are then followed by a Softmax layer which together form the classifier (Zheng *et al.*, 2018).

Inception-V3

Inception is a convolutional neural network widely used in classification tasks. The structure of the Inception network is at the heart of the GoogLeNet network. Several versions of Inception exist namely Inception v1 (2014), Inception v2 (2015), Inception v3 (2015), Inception v4 (2016), and Inception-ResNet (2016). The Inception module typically contains three different convolution sizes and maximum pooling. For the network output of the previous layer, the channel is aggregated after the convolution operation, and then the non-network elements are pooled (Sharma, 2022).

Compared to previous versions (Inception v1 and v2), the network structure of Inception v3 uses a convolution kernel division method to divide large-volume integrals into small convolutions. For example, a 3×3 convolution is divided into 3×1 and 1×3 . With the division method, the number of parameters will be reduced, it will speed up the learning speed of the network and the spatial feature can be extracted more efficiently (Dong *et al.*, 2020).

DenseNet-201

DenseNet is short for Dense Convolutional Network and uses fewer parameters than a conventional CNN because it does not learn redundant feature maps. DenseNet layers have 12 filters. DenseNet has four different variants: DenseNet-121, DenseNet-169, DenseNet-201, and DenseNet-264. In our study, we used DenseNet201, each layer has direct access to the original input image and the gradients of the loss function. The computational cost is considerably reduced, which makes DenseNet one of the best choices for image classification (Rahman *et al.*, 2020).

Proposed Custom CNN

We propose a custom CNN that is less complex but very efficient, it is composed of two convolutional layers with a 5×5 size core and four convolutional layers with a 3×3 size core. For each pair of convolutional layers, there is a max-pooling that will allow for a reduction of the images and the whole will be connected to 128 fully connected layers. Figure 2 shows the architecture of the proposed CNN

Architecture of Our Method

The architecture of our method is presented in three main steps:

- Step 1: This is the pre-processing part and the division of our dataset. For the data preprocessing step, we used the histogram equalizer on the images to expand the grey-level distribution range from 0 to 255. The images will also be normalized in the range of $[0,1]$
- Step 2: It is the training of the algorithms mentioned above on our dataset
- Step 3: It is about making the classification and the recognition of diseases

Figure 3 will show the general architecture of our study.

Experimental Setup

The performance of the eight machine learning models and six deep learning models for coffee leaf disease detection and recognition were evaluated with the following metrics: Precision, the Mean square error, Recall, the F1 score, and Matthews Correlation Coefficient. We will use the Precision graph and the Loss graph for the deep learning models and for the machine learning models we use the K-fold Cross-validation and the ROC curve. The experiments were conducted using the Python programming language on a DELL desktop computer equipped with a 2.90 GHz Intel (R) Core i7-10700 CPU, 32 GB RAM, and an NVIDIA Quadro P400 GPU.

- a) Parameters Setting

Machine Learning

K-Fold cross-validation is a technique used to evaluate the performance of machine learning or deep learning models in a robust way. It divides the data set into k parts of approximately equal size. Each part is selected as it goes along for testing and the remaining parts are for training. This process is repeated k times, and then the performance is measured as the average of all test sets (Wong and Yeh, 2019). In our study k = 10. Figure 4 illustrates the K-Fold.

The Receiver Operating Characteristic (ROC) curve was used to evaluate the performance of classification algorithms. It provides a graphical representation of a classifier's performance, rather than a single value like most other metrics (Delbarre *et al.*, 2021).

Deep Learning

The models were trained using Stochastic Gradient Descent (SGD) as an optimizer with a momentum of 0.9, and the learning rate is 0.0001. The learning rate defines the learning progress of the proposed model and updates the weight parameters to reduce the network loss function. The maximum number of epochs was set to 20 and batches of 20 were used in this experiment. Table 2 summarizes the hyperparameters used in our study (Yee-Rendon *et al.*, 2021).

Evaluation Metrics

To evaluate the performance of the models in our study, we will use different evaluation metrics, namely the accuracy, Precision, Mean square error, Recall, the F1 score, and Matthews Correlation Coefficient. They are calculated from the following formulas:

- Accuracy is a performance measure that demonstrates how accurately the system placed the data in the appropriate class

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (4)$$

- The proportion of accurately categorized positive photos to all true positive images is known as precision

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

- The recall is a classifier's capacity to identify true favorable results

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

- The weighted average of recall and precision is the F1 score

$$F1\ score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (8)$$

- In machine learning, the Matthews Correlation Coefficient (MCC) is used to assess the accuracy of classifications

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (9)$$

- The average of the squared mistakes, or the mean square difference between the estimated values and the true value, is measured by an estimator's mean square error

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{y}_i)^2 \quad (10)$$

The different variables used above are:

- True Positives (TP): Images with a true label and that were classified correctly
- False Positives (FP): Images with a false label that were classified as positive
- True Negatives (TN): Images with a false label and that were classified as negative
- False Negatives (FN): Images with a true label that was classified as negative
- Y_i : The actual data of the dataset
- \hat{y}_i : The predicted data of the dataset

Results

In this section, we will present the classification and recognition results according to two scenarios: The scenario using the eight machine learning algorithms mentioned above and the second scenario will present the five deep learning algorithms with our customized CNN.

Machine Learning Scenario Results

In this section, we present the different results of model learning. Table 3 presents the test results of each model using the test data as presented in the general architecture of our study above and we obtain results ranging from 100 to 81.03%, with SVM and RF both recording the best performance and GNB recording the worst performance.

Table 3: Test accuracy of each model

Models	SVM	KNN	RF	LR	MPL Classifier	DT Classifier	GNB
Accuracy	100.0	99.92	100.0	93.65	96.54	87.70	81.03

Table 4: ROC curve and Confusion Matrix of each model

Models	ROC curve	Confusion Matrix																									
SVM	<p>Some extension of Receiver operating characteristic to multi-class</p>	<p>Confusion Matrix</p> <table border="1"> <tr><th></th><th>Cercospora</th><th>Healthy</th><th>Leaf rust</th><th>Phoma</th></tr> <tr><th>Cercospora</th><td>1564</td><td>0</td><td>0</td><td>0</td></tr> <tr><th>Healthy</th><td>0</td><td>3824</td><td>0</td><td>0</td></tr> <tr><th>Leaf rust</th><td>0</td><td>0</td><td>3619</td><td>0</td></tr> <tr><th>Phoma</th><td>0</td><td>0</td><td>0</td><td>1296</td></tr> </table>		Cercospora	Healthy	Leaf rust	Phoma	Cercospora	1564	0	0	0	Healthy	0	3824	0	0	Leaf rust	0	0	3619	0	Phoma	0	0	0	1296
	Cercospora	Healthy	Leaf rust	Phoma																							
Cercospora	1564	0	0	0																							
Healthy	0	3824	0	0																							
Leaf rust	0	0	3619	0																							
Phoma	0	0	0	1296																							
KNN	<p>Some extension of Receiver operating characteristic to multi-class</p>	<p>Confusion Matrix</p> <table border="1"> <tr><th></th><th>Cercospora</th><th>Healthy</th><th>Leaf rust</th><th>Phoma</th></tr> <tr><th>Cercospora</th><td>1564</td><td>0</td><td>0</td><td>0</td></tr> <tr><th>Healthy</th><td>0</td><td>3824</td><td>0</td><td>0</td></tr> <tr><th>Leaf rust</th><td>0</td><td>0</td><td>3619</td><td>0</td></tr> <tr><th>Phoma</th><td>0</td><td>0</td><td>0</td><td>1296</td></tr> </table>		Cercospora	Healthy	Leaf rust	Phoma	Cercospora	1564	0	0	0	Healthy	0	3824	0	0	Leaf rust	0	0	3619	0	Phoma	0	0	0	1296
	Cercospora	Healthy	Leaf rust	Phoma																							
Cercospora	1564	0	0	0																							
Healthy	0	3824	0	0																							
Leaf rust	0	0	3619	0																							
Phoma	0	0	0	1296																							
RF	<p>Some extension of Receiver operating characteristic to multi-class</p>	<p>Confusion Matrix</p> <table border="1"> <tr><th></th><th>Cercospora</th><th>Healthy</th><th>Leaf rust</th><th>Phoma</th></tr> <tr><th>Cercospora</th><td>1564</td><td>0</td><td>0</td><td>0</td></tr> <tr><th>Healthy</th><td>0</td><td>3824</td><td>0</td><td>0</td></tr> <tr><th>Leaf rust</th><td>0</td><td>0</td><td>3619</td><td>0</td></tr> <tr><th>Phoma</th><td>0</td><td>0</td><td>0</td><td>1296</td></tr> </table>		Cercospora	Healthy	Leaf rust	Phoma	Cercospora	1564	0	0	0	Healthy	0	3824	0	0	Leaf rust	0	0	3619	0	Phoma	0	0	0	1296
	Cercospora	Healthy	Leaf rust	Phoma																							
Cercospora	1564	0	0	0																							
Healthy	0	3824	0	0																							
Leaf rust	0	0	3619	0																							
Phoma	0	0	0	1296																							
LR	<p>Some extension of Receiver operating characteristic to multi-class</p>	<p>Confusion Matrix</p> <table border="1"> <tr><th></th><th>Cercospora</th><th>Healthy</th><th>Leaf rust</th><th>Phoma</th></tr> <tr><th>Cercospora</th><td>1609</td><td>0</td><td>60</td><td>35</td></tr> <tr><th>Healthy</th><td>0</td><td>3824</td><td>0</td><td>0</td></tr> <tr><th>Leaf rust</th><td>3</td><td>0</td><td>2063</td><td>59</td></tr> <tr><th>Phoma</th><td>83</td><td>17</td><td>76</td><td>3970</td></tr> </table>		Cercospora	Healthy	Leaf rust	Phoma	Cercospora	1609	0	60	35	Healthy	0	3824	0	0	Leaf rust	3	0	2063	59	Phoma	83	17	76	3970
	Cercospora	Healthy	Leaf rust	Phoma																							
Cercospora	1609	0	60	35																							
Healthy	0	3824	0	0																							
Leaf rust	3	0	2063	59																							
Phoma	83	17	76	3970																							
MPLClassifier	<p>Some extension of Receiver operating characteristic to multi-class</p>	<p>Confusion Matrix</p> <table border="1"> <tr><th></th><th>Cercospora</th><th>Healthy</th><th>Leaf rust</th><th>Phoma</th></tr> <tr><th>Cercospora</th><td>1264</td><td>0</td><td>0</td><td>0</td></tr> <tr><th>Healthy</th><td>0</td><td>3824</td><td>0</td><td>0</td></tr> <tr><th>Leaf rust</th><td>7</td><td>0</td><td>3606</td><td>6</td></tr> <tr><th>Phoma</th><td>69</td><td>7</td><td>125</td><td>2918</td></tr> </table>		Cercospora	Healthy	Leaf rust	Phoma	Cercospora	1264	0	0	0	Healthy	0	3824	0	0	Leaf rust	7	0	3606	6	Phoma	69	7	125	2918
	Cercospora	Healthy	Leaf rust	Phoma																							
Cercospora	1264	0	0	0																							
Healthy	0	3824	0	0																							
Leaf rust	7	0	3606	6																							
Phoma	69	7	125	2918																							
DTClassifier	<p>Some extension of Receiver operating characteristic to multi-class</p>	<p>Confusion Matrix</p> <table border="1"> <tr><th></th><th>Cercospora</th><th>Healthy</th><th>Leaf rust</th><th>Phoma</th></tr> <tr><th>Cercospora</th><td>1242</td><td>0</td><td>35</td><td>287</td></tr> <tr><th>Healthy</th><td>0</td><td>3824</td><td>0</td><td>0</td></tr> <tr><th>Leaf rust</th><td>32</td><td>0</td><td>3475</td><td>112</td></tr> <tr><th>Phoma</th><td>216</td><td>1</td><td>128</td><td>693</td></tr> </table>		Cercospora	Healthy	Leaf rust	Phoma	Cercospora	1242	0	35	287	Healthy	0	3824	0	0	Leaf rust	32	0	3475	112	Phoma	216	1	128	693
	Cercospora	Healthy	Leaf rust	Phoma																							
Cercospora	1242	0	35	287																							
Healthy	0	3824	0	0																							
Leaf rust	32	0	3475	112																							
Phoma	216	1	128	693																							
GNB	<p>Some extension of Receiver operating characteristic to multi-class</p>	<p>Confusion Matrix</p> <table border="1"> <tr><th></th><th>Cercospora</th><th>Healthy</th><th>Leaf rust</th><th>Phoma</th></tr> <tr><th>Cercospora</th><td>1097</td><td>0</td><td>397</td><td>270</td></tr> <tr><th>Healthy</th><td>0</td><td>3824</td><td>0</td><td>0</td></tr> <tr><th>Leaf rust</th><td>85</td><td>0</td><td>3483</td><td>51</td></tr> <tr><th>Phoma</th><td>100</td><td>31</td><td>81</td><td>684</td></tr> </table>		Cercospora	Healthy	Leaf rust	Phoma	Cercospora	1097	0	397	270	Healthy	0	3824	0	0	Leaf rust	85	0	3483	51	Phoma	100	31	81	684
	Cercospora	Healthy	Leaf rust	Phoma																							
Cercospora	1097	0	397	270																							
Healthy	0	3824	0	0																							
Leaf rust	85	0	3483	51																							
Phoma	100	31	81	684																							

Table 5: Performance of accuracy and loss of each deep learning model for validation and test

Models	Validation		Test	
	Loss	Accuracy	Loss	Accuracy
MobileNet	0.033	100.00	0.039	100.00
ResNet-50	00.160	99.97	00.180	99.94
VGG-19	09.830	97.33	09.680	97.37
DenseNet-201	0.140	99.97	00.140	99.97
InceptionV3	0.170	99.98	00.320	99.95
Custom CNN	0.014	100.00	0.013	100.00

Table 6: Accuracy, loss curve and confusion matrix of each deep learning model

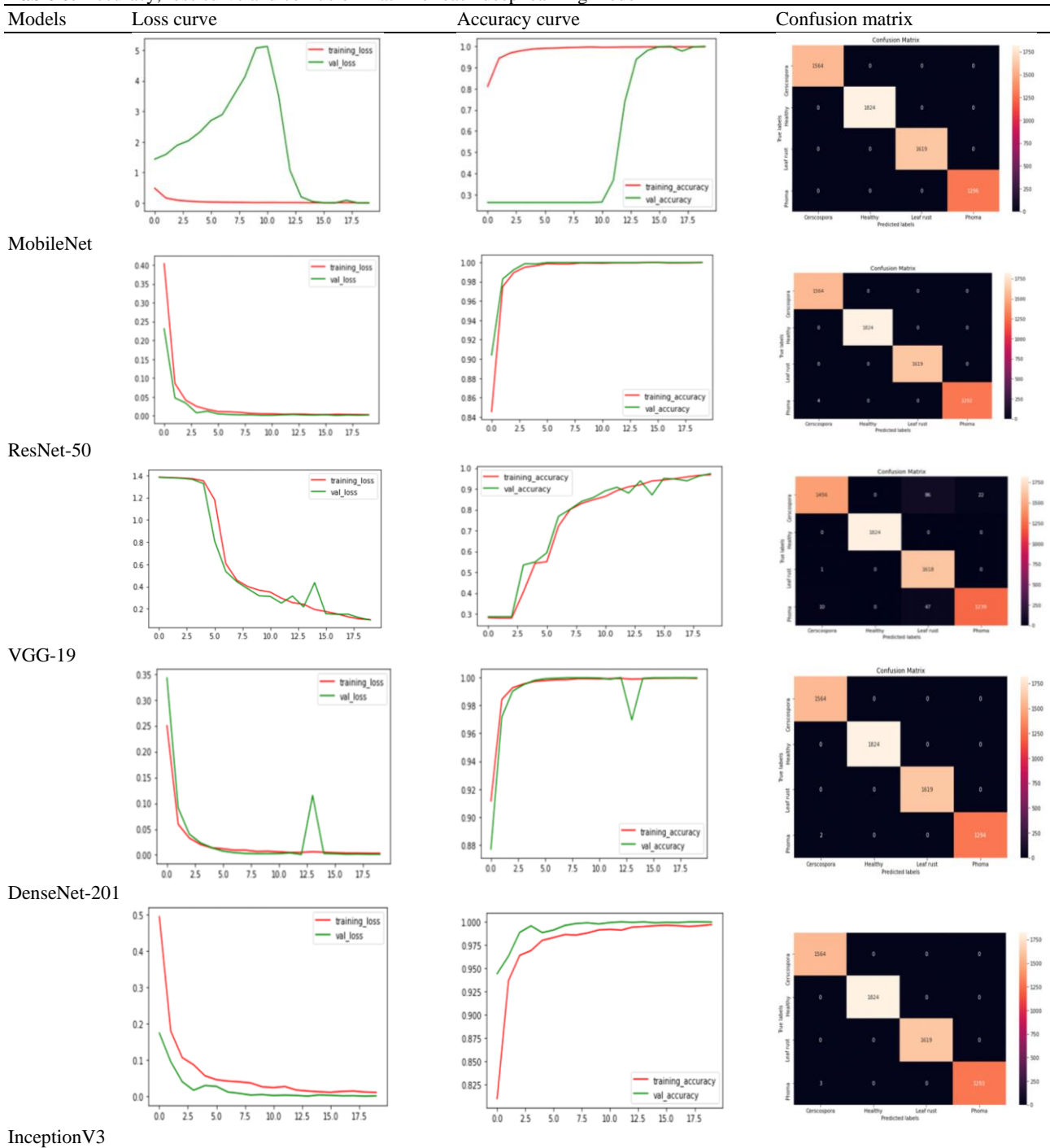


Table 7: Performance metrics for machine learning models

Models	Precision (%)	MSE (%)	F-score (%)	Recall (%)	MCC (%)
SVM	100.00	0.00	100.00	100.00	100.00
KNN	99.87	0.38	99.87	99.87	99.83
RF	100.00	0.00	100.00	100.00	100.00
LR	94.01	31.11	93.93	94.01	92.01
MPLClassifier	96.86	11.96	96.62	96.69	95.65
DTClassifier	87.22	79.94	87.15	87.13	82.78
GNB	80.26	131.90	80.23	80.72	74.24

Table 8: Performance metrics for deep learning models

Models	Precision (%)	MSE (%)	F-score (%)	Recall (%)	MCC (%)
MobileNet	100.00	0.000600	100.00	100.00	100.00
ResNet-50	99.93	00.010000	99.93	99.93	99.91
VGG-19	97.50	01.140000	97.36	97.36	96.51
DenseNet-201	99.96	0.010000	99.96	99.96	99.95
InceptionV3	99.95	0.020000	99.95	99.95	99.93
Custom CNN	100.00	0.000002	100.00	100.00	100.00

Finally, we present in table 4 the ROC curve and the confusion matrix of each model.

Deep Learning Scenario Results

We now present the test results of the deep learning models used.

Table 5 presents the performance of accuracy and loss of the validation and testing stages of the deep learning models used. We obtain the validation loss values that vary between 0.014 to 9.83% with our custom CNN recording the best performance and for the validation accuracy, we have values that vary between 97.33% and 100% with our custom CNN and the MobileNet that both record the best performance. Regarding the loss of the test, we have values that vary between 0.013 and 9.68% with our Custom which has the best performance, and for the accuracy of the test, we get scores that vary between 97.37 and 100% with our Custom CNN and the MobileNet that both record the best performance. We present the loss and accuracy curves and the confusion matrix in Table 6 below. The proposed CNN model is less complex and is faster in terms of learning time per epoch with an average of 559s while MobileNet is 667s, Inception V3 is 824s, ResNet50 is 2129s, VGG19 is 2601s, DenseNet 201 is 2981s. It minimizes the mean square error better than all the methods used in our study.

Performance Metrics

Tables 7 and 8 present the respective metrics of the machine learning and deep learning models that consolidate the performances obtained in the two scenarios above.

Discussion

Coffee plant diseases are a major threat to agriculture in general and its productivity in particular, creating many economic losses.

Several studies on the detection of plant diseases have been conducted using machine learning. The most used machine learning technique is deep learning and more precisely convolutional neural networks, which are very efficient for image-based plant disease classification. Also, we have highlighted the performance of classical machine learning models. This study showed that the method used is very effective for the classification and disease recognition of coffee plants from images.

For classical machine learning models, we get better results with 100% results with SVM and RF. Concerning the deep learning algorithms used in our study, we find that our model presents better results for the validation data and the test data. Thus, we obtain the lowest values for the loss functions which are 0.014% for the validation and 0.013% for the test. For accuracy, we obtain with the MobileNet model the best performances both for the validation and for the test with a value of 100%.

Conclusion

In this study, we used several approaches for the classification and recognition of coffee leaf diseases. Both classical machine learning and deep learning methods were used to implement our method.

The evaluation of these approaches showed the effectiveness of the SVM and Random Forest methods with 100% accuracy for the classical machine learning models while MobileNet and our custom CNN showed their effectiveness with 100% accuracy for the deep learning models. Regarding our custom CNN, it recorded less loss than MobileNet using the loss function and this shows that our model is the most robust.

SVM and Random Forest have performed well as deep learning models. In the future, we can implement a robust method in an uncontrolled environment by coupling our method with segmentation to better focus on the diseased part of the plant.

Acknowledgment

We would like to express our thanks and gratitude to all those who helped us throughout this study. Also thank you to the person in charge of the virtual university of Côte d'Ivoire for allowing me to carry out these research stays at the ImViA laboratory.

Funding Information

The authors have not received any financial support or funding to report.

Author's Contributions

Kacoutchy Jean Ayikpa: Conceptualization, methodology, software, validation, formal analysis, investigation, resource, data curation, writing-original draft, preparation, visualization.

Diarra Mamadou: Conceptualization, methodology software, investigation, resources, writing-original draft preparation, visualization.

Pierre Gouton : Supervision project administration Validation writing-review and editing

Kablan Jérôme Adou: Supervision, projet administration, validation.

Ethics

I undersigned that this article has not been published elsewhere. The authors declare no conflict of interest.

References

- Aburass, S., Huneiti, A., & Al-Zoubi, M. B. (2022). Classification of Transformed and Geometrically Distorted Images using Convolutional Neural Network. <https://doi.org/10.3844/jcssp.2022.757.769>
- Afifi, A., Alhumam, A., & Abdelwahab, A. (2020). Convolutional neural network for automatic identification of plant diseases with limited data. *Plants*, 10(1), 28. <https://doi.org/10.3390/plants10010028>
- Ahmed, A. A., & Reddy, G. H. (2021). A mobile-based system for detecting plant leaf diseases using deep learning. *Agri Engineering*, 3(3), 478-493. <https://doi.org/10.3390/agriengineering3030032>
- Akbulut, Y., Sengur, A., Guo, Y., & Smarandache, F. (2017). NS-k-NN: Neutrosophic set-based k-nearest neighbors classifier. *Symmetry*, 9(9), 179. <https://doi.org/10.3390/sym9090179>
- Alam, M. S., & Vuong, S. T. (2013, August). Random forest classification for detecting android malware. In 2013 IEEE international conference on green computing and communications and IEEE Internet of Things and IEEE cyber, physical and social computing (pp. 663-669). IEEE. <https://doi.org/10.1109/GreenCom-iThings-CPSCoM.2013.122>
- Attallah, O. (2021). MB-AI-His: histopathological diagnosis of pediatric medulloblastoma and its subtypes via AI. *Diagnostics*, 11(2), 359. <https://doi.org/10.3390/diagnostics11020359>
- Crop Diseases. (2021). Department of primary industries and regional development. <https://www.agric.wa.gov.au/pests-weeds-diseases/diseases/crop-diseases>.
- Davuluri, R., & Rengaswamy, R. (2022). Identification of Alzheimer's Disease Using Various Deep Learning Techniques-A Review. *Intelligent Manufacturing and Energy Sustainability*, 485-98. <https://doi.org/10.3844/jcssp.2022.415.425>
- Delbarre, M. A., Girardon, F., Blanc-Durand, P., Hubaut, M. A., Roquette, L., Hachulla, E., ... & Damy, T. (2021). Détection automatique des amyloses cardiaques en scintigraphie osseuse à l'aide du machine learning. *La Revue de Médecine Interne*, 42, A343. <https://doi.org/10.1016/j.revmed.2021.10.314>
- Dewi, W., & Utomo, W. H. (2021). Plant classification based on leaf edges and leaf morphological veins using wavelet convolutional neural network. *Applied Computer Science*, 17(1). <https://doi.org/10.23743/acs-2021-08>
- Dong, N., Zhao, L., Wu, C. H., & Chang, J. F. (2020). Inception v3 based cervical cell classification combined with artificially extracted features. *Applied Soft Computing*, 93, 106311. <https://doi.org/10.1016/j.asoc.2020.106311>
- Giraddi, S., Desai, S., & Deshpande, A. (2020). Deep learning for agricultural plant disease detection. In ICDSMLA 2019 (pp. 864-871). Springer, Singapore. https://doi.org/10.1007/978-981-15-1420-3_93
- Jain, C. K., Gupta, N., Gupta, B. & Passi, K. (2022). Applications of Artificial Intelligence Based Technologies in Weed and Pest Detection. *Journal of Computer Science*, 18(6), 520-529. <https://doi.org/10.3844/jcssp.2022.520.529>
- Jahromi, A. H., & Taheri, M. (2017, October). A non-parametric mixture of Gaussian naive Bayes classifiers based on local independent features. In 2017 Artificial intelligence and signal processing conference (AISP) (pp. 209-212). IEEE. <https://doi.org/10.1109/AISP.2017.8324083>
- Jepkoech, J., Mugo, D. M., Kenduiywo, B. K., & Too, E. C. (2021). Arabica coffee leaf images dataset for coffee leaf disease detection and classification. *Data in Brief*, 36, 107142. <https://doi.org/10.1016/j.dib.2021.107142>
- Kurt, I., Ture, M., & Kurum, A. T. (2008). Comparing performances of logistic regression, classification and regression tree, and neural networks for predicting coronary artery disease. *Expert Systems with Applications*, 34(1), 366-374. <https://doi.org/10.1016/j.eswa.2006.09.004>

- Lu, J., Tan, L., & Jiang, H. (2021). Review on convolutional neural network (CNN) applied to plant leaf disease classification. *Agriculture*, 11(8), 707.
<https://doi.org/10.3390/agriculture11080707>
- Planetoscope-Statistiques. (2021). Production mondiale de café.
<https://www.planetoscope.com/boisson/791-.html>
- Quels sont les meilleurs cafés d'Afrique. (2020). URL
<https://graine-de-cafe.com/meilleurs-cafes-afrique/>.
- Rahman, T., Chowdhury, M. E., Khandakar, A., Islam, K. R., Islam, K. F., Mahbub, Z. B., ... & Kashem, S. (2020). Transfer learning with deep convolutional neural network (CNN) for pneumonia detection using chest X-ray. *Applied Sciences*, 10(9), 3233.
<https://doi.org/10.3390/app10093233>
- Ramcharan, A., Baranowski, K., McCloskey, P., Ahmed, B., Legg, J., & Hughes, D. P. (2017). Deep learning for image-based cassava disease detection. *Frontiers in Plant Science*, 8, 1852.
<https://doi.org/10.3389/fpls.2017.01852>
- Saleem, M. H., Khanchi, S., Potgieter, J., & Arif, K. M. (2020). Image-based plant disease identification by deep learning meta-architectures. *Plants*, 9(11), 1451.
<https://doi.org/10.3390/plants9111451>
- Santos, L., Santos, F. N., Oliveira, P. M., & Shinde, P. (2019, November). Deep learning applications in agriculture: A short review. In Iberian Robotics conference (pp. 139-151). Springer, Cham.
https://doi.org/10.1007/978-3-030-35990-4_12
- Shao, H., Tang, R., Lei, Y., Mu, J., Guan, Y., & Xiang, Y. (2021). Rice ear counting based on image segmentation and establishment of a dataset. *Plants*, 10(8), 1625.
<https://doi.org/10.3390/plants10081625>
- Sharma, S. (2022). Analyzing Effect on Residual Learning by Gradual Narrowing Fully-Connected Layer Width and Implementing Inception Block in Convolution Layer. *Journal of Computer Science*, 18(5), 339-349
<https://doi.org/10.3844/jcssp.2022.339.349>
- Shobha, G., & Rangaswamy, S. (2018). Handbook of statistics. <https://doi.org/10.1016/bs.host.2018.07.004>
- Support Vector Machines. (2021). Scikit-Learn.
<https://scikit-learn/stable/modules/svm.html>
- Velásquez, D., Sánchez, A., Sarmiento, S., Toro, M., Maiza, M., & Sierra, B. (2020). A method for detecting coffee leaf rust through wireless sensor networks, remote sensing, and deep learning: case study of the caturra variety in Colombia. *Applied Sciences*, 10(2), 697.
<https://doi.org/10.3390/app10020697>
- Wan, S., Liang, Y., Zhang, Y., & Guizani, M. (2018). Deep multi-layer perceptron classifier for behavior analysis to estimate Parkinson's disease severity using smartphones. *IEEE Access*, 6, 36825-6833.
<https://doi.org/10.1109/ACCESS.2018.2851382>
- Wong, T. T., & Yeh, P. Y. (2019). Reliable accuracy estimates from k-fold cross validation. *IEEE Transactions on Knowledge and Data Engineering*, 32(8), 1586-1594.
<https://doi.org/10.1109/TKDE.2019.2912815>
- Xiao, J., Wang, J., Cao, S., & Li, B. (2020, April). Application of a novel and improved VGG-19 network in the detection of workers wearing masks. In *Journal of Physics: Conference Series* (Vol. 1518, No. 1, p. 012041). IOP Publishing.
<https://doi.org/10.1088/1742-6596/1518/1/012041>
- Yang, G., Chen, G., Li, C., Fu, J., Guo, Y., & Liang, H. (2021). Convolutional rebalancing network for the classification of large imbalanced rice pest and disease datasets in the field. *Frontiers in Plant Science*, 1150.
<https://doi.org/10.3389/fpls.2021.671134>
- Yee-Rendon, A., Torres-Pacheco, I., Trujillo-Lopez, A. S., Romero-Bringas, K. P., & Millan-Almaraz, J. R. (2021). Analysis of New RGB Vegetation Indices for PHYVV and TMV Identification in Jalapeño Pepper (*Capsicum annuum*) Leaves Using CNNs-Based Model. *Plants*, 10(10), 1977.
<https://doi.org/10.3390/plants10101977>
- Yu, H. J., & Son, C. H. (2019). Apple leaf disease identification through region-of-interest-aware deep convolutional neural network. arXiv preprint arXiv:1903.10356.
<https://doi.org/10.2352/J.ImagingSci.Technol.2020.64.2.020507>
- Yu, H. J., & Son, C. H. (2020). Leaf spot attention network for apple leaf disease identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops* (pp. 52-53).
<https://doi.org/10.1109/CVPRW50498.2020.00034>
- Zheng, Y., Yang, C., & Merkulov, A. (2018, May). Breast cancer screening using convolutional neural network and follow-up digital mammography. In *Computational Imaging III* (Vol. 10669, p. 1066905). SPIE. <https://doi.org/10.1117/12.2304564>