

Original Research Paper

# A Weighted Voting Deep Learning Approach for Plant Disease Classification

<sup>1</sup>Assia Ennoui, <sup>1</sup>Noura Ouled Sihamman, <sup>1</sup>My Abdelouahed Sabri and <sup>2</sup>Abdellah Aarab

<sup>1</sup>Department of Computer Science, Faculty of Sciences Dhar el Mahraz, University Sidi Mohamed Ben Abdellah, Fez, Morocco

<sup>2</sup>Department of Physique, Faculty of Sciences Dhar el Mahraz, University Sidi Mohamed Ben Abdellah, Fez, Morocco

## Article history

Received: 14-08-2021

Revised: 05-11-2021

Accepted: 09-10-2021

## Corresponding Author:

Assia Ennoui

Department of Computer

Science, Faculty of Sciences

Dhar el Mahraz, University Sidi

Mohamed Ben Abdellah, Fez,

Morocco

Email: ennoui.assia@gmail.com

**Abstract:** Plant destruction is usually caused by plant diseases. Without early and reliable detection, it can affect the plants and may eventually cause permanent losses, especially for inexperienced farmers. Therefore, intelligence in agriculture is becoming more and more required. Thereafter early diagnosis and classification are crucial and a very challenging research field in the agriculture sector for its treatment. In this context, many solutions have been proposed. Deep learning has been highly successful and hardly applicable in this problem. However, through pass survey analysis, we notice that there are a few studies in DL for disease classification problems but the precision and outcomes of different traditional DL methods may vary and give a less score for classification. The proposed approach is based on a weighted combination of five deep learning architectures. The weight of each DL architecture is calculated based on its performance using genetic algorithms. The results of the proposed approach are evaluated on the publicly available Plant Village (PV) dataset. It is found that using the Deep Learning weighted voting method gives higher classification accuracy compared to the results obtained using each DL architecture separately and also compared to recent approaches in literature, which allowed us to correctly identify the leaves and to improve the classification accuracy rate to 99.21%.

**Keywords:** Smart Agriculture, Deep Learning, Ensemble Learning, Weighted Voting, Plant Disease Classification

## Introduction

It is well-known that plant diseases cause appreciable crop losses all over the world and they have a vandal effect on agricultural products. Plant diseases are the first wasting of agricultural progression and have an immediate impact on the quality and quantity of plant foods. If they are not controlled, detected and treated in time, there will be a rise in food insecurity (Fina *et al.*, 2013). Morocco such like any African countries depends solely on agriculture. Recently, FAO declares a considerable increase due to many causes such as climate change.... but plant disease detection remains the most important factor in this problem. To address the considerable problem of plant disease diagnosis and help farmers in their decisions, artificial intelligent approaches for agricultural image analysis can be utilized to develop a performant classification system that can detect plant diseases using only images of leaves (Ennoui *et al.*, 2021a). Deep learning approaches have been used for medical image classification (Filali *et al.*, 2020) In this

trend, deep learning applications are proposed and implemented to agricultural imaging tasks are thoroughly examined (Litjens *et al.*, 2017; Biswas *et al.*, 2019; Abdelhafiz *et al.*, 2019; Lundervold *et al.*, 2019).

In this study, a novel method for detecting and classifying plant diseases is proposed based on the Ensemble Learning (EL) technique. Our proposed approach aggregates five deep learning architectures naming VGG16, AlexNet, CNN, Inceptionv3 and mobileNet in an ensemble learning scheme using a weighted voting mechanism. The optimal weights are assigned by the genetic algorithm to achieve the optimal ones and improve the classification accuracy. The performance of the suggested approach is assessed on the publicly available Plant Village dataset (PV).

## Related Work

Plant diseases are a worldwide agricultural problem in precision agriculture. It is considered the one kind of natural disaster that attacks the normal growth process of plants and this problem causes plant death in most cases

during the whole growth process of plants from seed development to seedling and seedling growth. The various diseases that can affect plants can be categorized into three types: Fungal, bacterial, or viral (El-Sayed *et al.*, 2020; Vijai *et al.*, 2020).

### Viral Diseases

This type is identified by these classic symptoms mosaic patterns, yellowing, stripes and leaf rolling as you have shown in Fig. 1. Viral diseases are diffused to plants principally by insects or worms.

### Fungal Diseases

The symptoms of these diseases are leaf rust especially for corn, stem rust, white mold (*Sclerotinia*). They considered damaging to plants but not with great risk. Fungi, mold, mildew and others cause these diseases. Figure 2 shows an instance of a plant affected by *Sclerotinia* fungus.

### Bacterial Diseases

They can be detected by the attendance of rot, scab, scorch, wilt and leaf spots. Plant infections of bacterial have almost similar symptoms to fungal diseases. These types of diseases, if not treated in time, can cause serious and disastrous diseases (Park *et al.*, 2017). An example of bacterial disease is shown in Fig. 3.

In general, plant disease symptoms are hard to discover in time as result early plant diagnosis by experts is required but expensive (Mathews, 2010). There are many directed studies on machine learning-based plant diagnosis in recent years (Barbedo, 2013; Huang, 2007; Phadikar and Sil, 2008) to improve diagnosis results. In (Barbedo, 2013) authors used a multi-layer perceptron-based recognition method to detect bacterial soft rot, bacterial brown spot and phytophthora black rot appearing on orchids, this study reported an average classification accuracy of 89.6%. Others (Huang, 2007) suggest a method based on self-organized; it achieved over 70% accuracy in distinguishing between rice blast and brown spot appearing on rice leaves. In (Phadikar and Sil, 2008) researchers applied SVM for distinguishing brown spots, downy mildew and angular leaf spot on cucumbers and reached an accuracy of 83.3%. In (Zhang *et al.*, 2018) analyzed nutrient deficiency in tomatoes utilizing k-nearest

neighbor clustering based on the leaf, the authors extract two descriptors color and texture. However, it is quite difficult to handle these techniques for detecting different plant diseases (feature representation). That means all these proposed techniques are confronted with several difficulties, concerning the detection of regions of interest for succeeding processing (Xu *et al.*, 2011). The effects of deep learning architectures are noticeably greater than those of the traditional machine learning methods in the most application process. Deep learning models are capable to learn feature representations from raw data and are not dependent on feature engineering that many works have underlined the importance of deep learning in agricultural tasks comparing with the traditional ML techniques for improving the quality and safety of nutrition.

In recent years, we notice an immense interest in deep learning in various fields (LeCun *et al.*, 2015). Deep Learning (DL) architectures, including Convolutional Neural Networks (CNN), has arisen as the most encouraging approaches given their ability to learn reliable and discriminative visual characteristics. Many studies have emphasized and proved the importance of deep learning in agriculture as compared to ML methods, for instance, the famous DL model named Google Net outperformed the ML techniques including Support Vector Machine and Random Forest algorithms for the classification of disease in tomato leaves (Ennoui *et al.*, 2021a). In this section, we present recent researches that reveal the entire penetration of deep learning.

Authors in these researchers present many state-of-the-art architectures containing MobileNet (Howard *et al.*, 2020), Visual Geometry Group (VGG) (Brahimi *et al.*, 2017; Ferentinos *et al.*, 2008; Mohanty *et al.*, 2016; Zhang *et al.*, 2018; Oppenheim *et al.*, 2019), AlexNet (Howard *et al.*, 2020; Brahimi *et al.*, 2017; Ferentinos *et al.*, 2008; Fuentes *et al.*, 2017; Mohanty *et al.*, 2016; Türkoğlu *et al.*, 2019; Zhang *et al.*, 2018), ResNet (Fuentes *et al.*, 2017; Türkoğlu *et al.*, 2019; Zhang *et al.*, 2018; Too *et al.*, 2019), Inception-v4 and DenseNet (Too *et al.*, 2019), got promising results for the classification of plant leaf disease. This table displays the popular and successful architectures including classic and modern architectures. As you see in this table we identify for all architecture; the number of parameters, the authors, the year and the depth.

**Table 1:** The most successful CNN architectures

Architectures	References	Year	Nb of params	Depth
LeNet-5	LeCun <i>et al.</i>	1998	60 ,000	5
AlexNet	Krizhevsky <i>et al.</i>	2012	60 million	8
VGG	Simonyan and Zisserman	2014	138 million	19
GoogleNet	Szegedy <i>et al.</i>	2015	4 million	22
Inception V3	Szegedy <i>et al.</i>	2015	23 million	159
Inception V4	Szegedy <i>et al.</i>	2016	35 million	70
ResNet	He <i>et al.</i>	2016	25 million	152
MobileNetV1	Sandler <i>et al.</i>	2018	4.2 million	28
MobileNetV2	Mark <i>et al.</i>	2018	3.47 million	53



**Fig. 1:** Rose mosaic virus leaves



**Fig. 2:** Sclerotinia Infected by Soybeans



**Fig. 3:** Bacterial blight of peas leaves

As you see before, the number of architectures and algorithms that are used in deep learning is varied and wide. In this section, we explore the famous deep learning architectures and we take a look in more detail at each of these to give you a deeper understanding of applying these approaches in practice.

#### *LeNet-5*

Starting with LeNet-5 (LeCun *et al.*, 1989), LeNet-5 CNN is one of the earliest models developed by LeCun *et al.* (1998); in the research work, Gradient-based Learning Applied to Document Recognition. It is used to identify handwritten and machine-printed characters. This architecture is simple in that It is popular for image classification. It has 5 layers with learnable parameters and 3 convolution layers, two average pooling layers and 2 fully connected layers with a softmax classifier. The presented Fig. 4 depicts the architecture of LeNet5 in the original paper.

#### *ALexNet*

Krizhevsky *et al.* (2012) won the Image Net ILSVRC 2012 by developing a very similar architecture (Krizhevsky *et al.*, 2012) as Le Net by LeCun *et al.* (1989). Their architecture reducing the top-5 error from 26% to 15.3% and considered as a leading architecture with a total of eight layers with learnable parameters. The fundamental highlights of this proposed architecture are these: First, used ReLU instead of tanh to add non-linearity. Second, prevented their model from overfitting by using dropout instead of regularization. Finally and to reduce the size of the network they used overlap polling. It has five layers with a combination of max pooling followed by 3 fully connected layers. They use ReLU activation in each of these layers except the output layer. AlexNet architecture is shown below in Fig. 5.

#### *VGG*

While the AlexNet model focused on smaller windows sizes and strides in the first convolutional layer. VGG come out in 2014 based on depth (Simonyan and Zisserman, 2014). Simonyan introduced the popular convolutional neural network architecture called VGG in their work intituled ‘Very Deep Convolutional Networks for Large-Scale Image Recognition’ VGG concnered on another very important aspect of CNNs which is depth. Its work was focused on an analysis of how to increase the depth of the network, using smaller filters (3\*3) with more depth instead of having large filters. The best results can be achieved by pushing the depth to 16-19 weights layers. The proposed model has achieved 92.7% top-5 test accuracy in Image Net. Noted that the VGG model supports up to 19 layers. The network architecture is provided in the following Fig. 6.

In the same context, authors Szegedy *et al.* (2015) proposed a new type of deep convolutional neural network named Inception which is called Google Le Net or Inception in their work entitled "Going deeper with convolutions" (Szegedy *et al.*, 2015) for ILSVRC14. The focal point of this architecture is the improved utilization of the computing resources inside the network. They increased the depth and width of the network while keeping the computational budget constant. It is essentially a Convolutional Neural Network (CNN) that is 27 layers deep. The basic implementation of this model was called Google Net. Later publications were then named Inception VX where X refers to the number of the version. The presented Fig. 7 depicts the architecture of inception.

### ResNet

Now, we will discuss the Residual Network Known as ResNet by Kaiming He *et al.* (2016) which is one of the most robust deep neural networks which has obtained excellent performance results in the ILSVRC 2015 classification challenge. Every layer of a ResNet (Zhang *et al.*, 2018) is composed of several blocks. This is because when ResNets go deeper, they normally do it by increasing the number of operations within a block, but the number of total layers remains the same. The principal idea of ResNet is (Sibiya and Sumbwanyambe, 2019) presenting a so-named "identity shortcut connection" that skips one or more layers to deal vanishing gradient problem. Instead of hoping every few stacked layers directly fit a desired underlying mapping, residual nets let these layers fit a residual mapping. Figure 8 illustrates this architecture.

### MobileNet

Mobile Net architecture [31] introduced by Howard *et al.* (2017) in Mobile Nets: Efficient Convolutional Neural Networks for Mobile Vision Applications in 2017. It is a lightweight deep neural network that provides a robust model for embedded and mobile applications (Sahni *et al.*, 2021). It utilizes depth-wise separable convolutions instead of standard convolutions to decrease model size and computation. So we can see, Mobile Net is built on depth wise separable convolutions (Szegedy *et al.*, 2015), except for the first layer. The first layer is a full convolutional layer. The Fig. 9 below shows the Mobile Net architecture based on depth wise separable filters.

### Plant Disease Classification Based Ensemble Learning Mechanisms

Ensemble learning is a machine learning paradigm where multiple learners are trained to solve the same problem. It has been demonstrated that combining multiple classifiers can be more effective than any single one (Onan *et al.*, 2016). From a technical standpoint, ensemble learning is primarily implemented in two steps: Training weak classifiers and selectively combining the

member classifiers into a stronger classifier. Based on this source (Xiao *et al.*, 2018), it is conceivable to combine the performance of a set of classification architectures to obtain a single robust classification model each architecture has strengths and weaknesses. Usually, the members of an ensemble are built in two ways. One is to apply a single learning algorithm and the other is to use different learning algorithms over a dataset (Chen *et al.*, 2014). Then, the base classifiers are combined to form a decision classifier. Generally, to get a good ensemble, the base learners should be as more accurate as possible and as more diverse as possible. So how to choose an ensemble of some accurate and diverse base learners is a focus of concern of many researchers. In recent years, more and more researchers are concerned with ensemble learning (Zhou *et al.*, 2009). The major principles of ensemble learning for analysis could be classified into these methods, bagging, boosting, stacking and voting described as follows.

Bagging is the most basic and simplest method; it involves two steps; bootstrapping and aggregation, in which a single training algorithm is applied to different subsets of training data, with subset sampling performed with replacement (bootstrap). Once the algorithm has been trained across all subsets, bagging predicts by aggregating all of the predictions made by the algorithm across all subset. Because the sampling is done by bootstrapping, the bagging method works so well because of the diversity in the training data. It reduces variance while not increasing bias (Breiman, 1996; Petropoulos *et al.*, 2018).

Boosting: This is a machine learning algorithm in which the weights of the classifiers are iteratively adjusted. The principle underlying the boosting approach was based on the importance of more complex models. The classification piqued people's interest in the amount of weight set in each model of the training set (Wang and Ma, 2011). By decreasing the distribution and combination of weak classifiers, the AdaBoost algorithm developed the weight of each case by training a set of the same and different weak classifiers. Finally, the AdaBoost algorithm was used to construct the final decision using weighted majority voting.

Stacking is another well-known ensemble learning technique for achieving the best possible outcome which frequently considers heterogeneous weak learners, learns them in parallel and combines them by training a meta-model to output a prediction based on the predictions of the different weak models. The stacking architecture set (Wolpert, 2011) was a two-level structure made up of level-0 (base-level) classifiers and level-1 (meta-level) classifiers. The training dataset was used to train the base-level classifiers, which were then used to construct the prediction. Metadata was later used to train a metaclassifier to recognize the output of the base-level classifier as the class label. Stacking was generally used to combine the developed model with the various classifiers. These classifiers were combined in various predictions to form the final resolution.

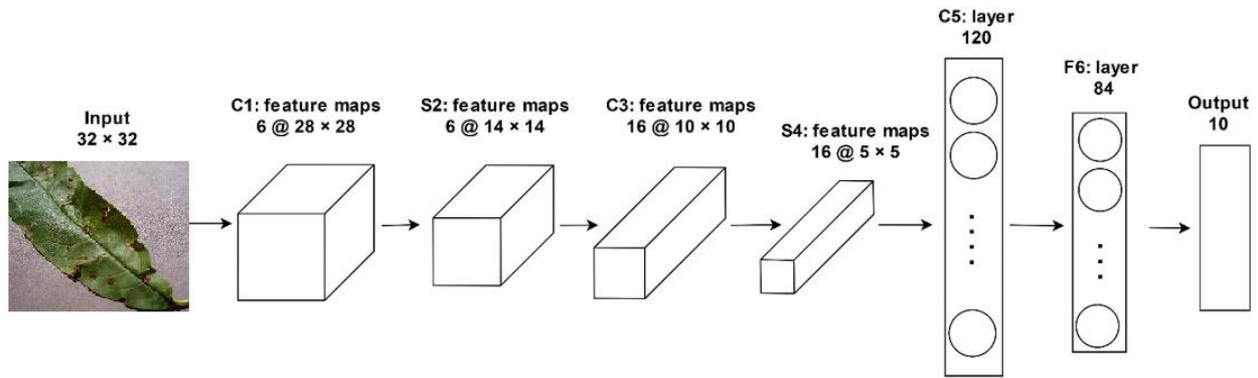


Fig. 4: LeNet-5 architecture

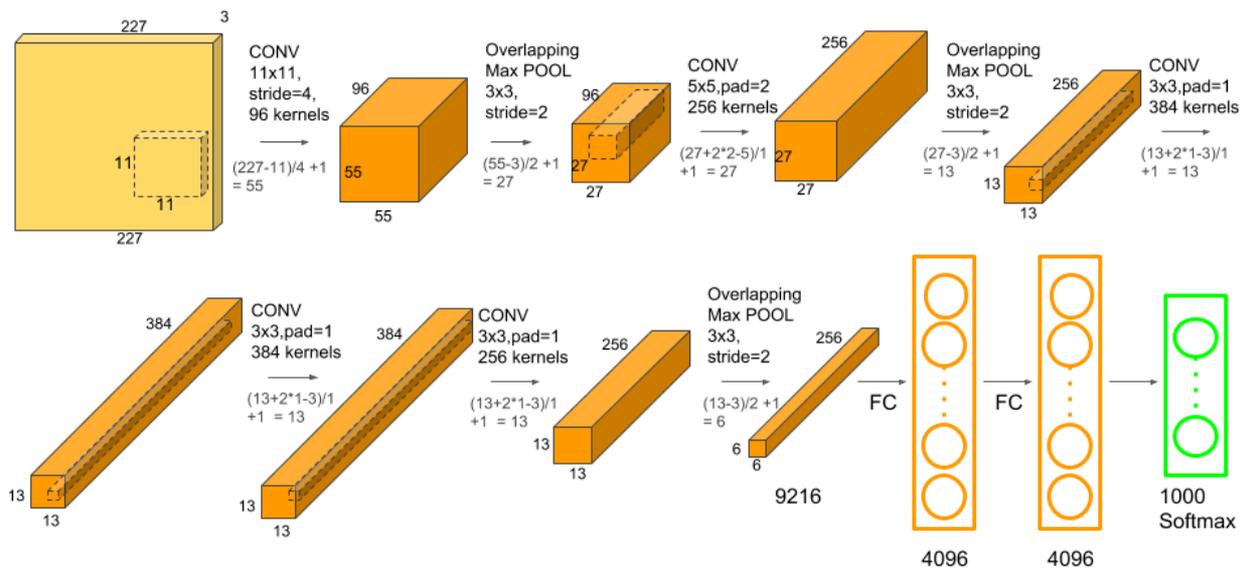


Fig. 5: Alex Net Architecture

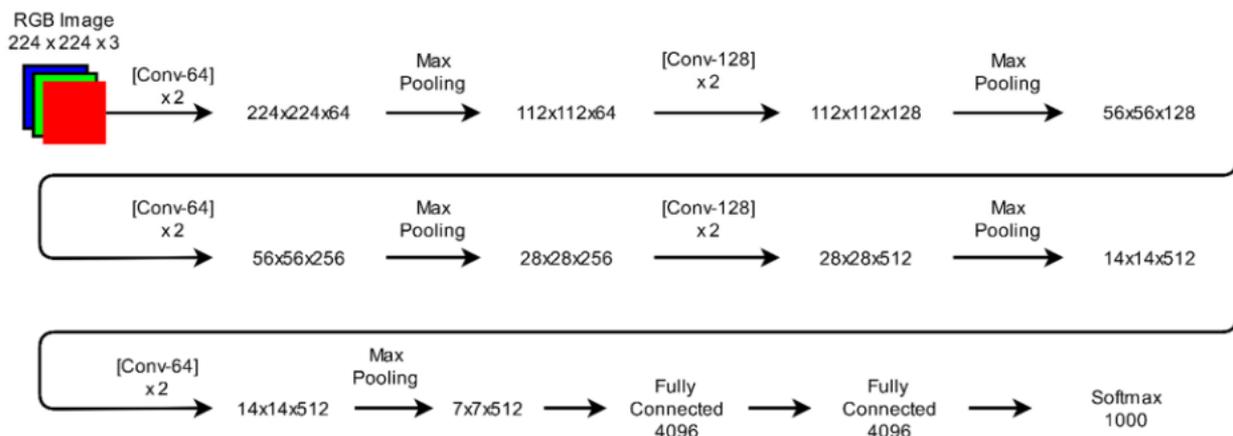
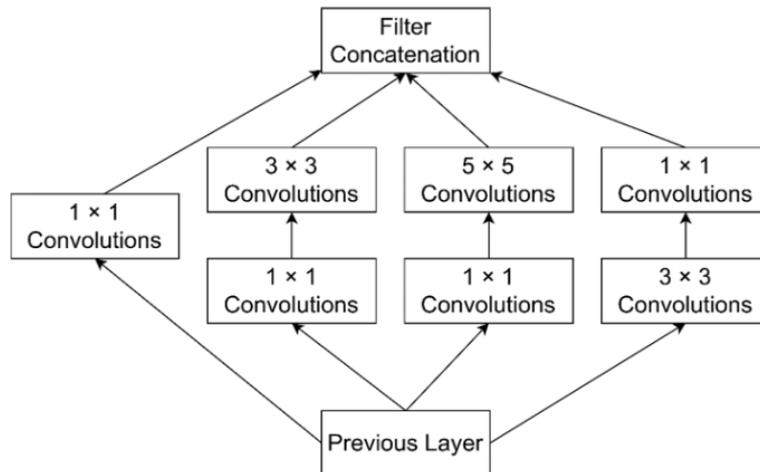


Fig. 6: VGG16 Architecture



**Fig. 7:** Inception module with dimensionality reduction

Voting method used to make decisions by selecting only one of several alternatives. Voting depended on the class predicted with the majority voting. Moreover, voting was the most utilized in the ensemble methods. Generally, the voting method consisted of unweighted voting and weighted voting. Unweighted voting consisted of simple voting and majority voting, whereas weighted voting included simple weighted voting (Kuncheva, 2014). It supports these types of voting.

Hard voting or Majority-Based Voting Mechanism includes summing the predictions for each class label and predicting the class label with the most votes that means the final class prediction is made by the majority vote which the estimator selects the class that occurs most frequently among the base models (Delgado and Ishii, 1999; Kang *et al.*, 2020).

$$\tilde{y} = \text{Mode}\{C_1(x), C_2(x), \dots, C_n(x)\}$$

Weighted voting was simple voting that was considered the suitable method for all classifiers with equivalent performance. However, the base classifier was practically carried out with different weights for defining the weight. Thus, weight voting was designed to define the weight for constructing a strong classifier (Tyagi *et al.*, 2021). We can compute a weighted majority vote by associating a weight  $w_j$  to the classifier  $C_j$ :

$$\tilde{y} = \arg \max \sum_{j=0}^m W_j C_j(x)$$

Soft voting has the same principle as earlier but is expressed in terms of probabilities. That means the final output is predicted based on the predicted probabilities  $p$  for the classifiers (Du and Swamy, 2019). For each class label and predicting the class label with the largest probability. In each case below, the probability of class labels assigned by the classifier  $C$  to input  $x$  is defined as (Cao *et al.*, 2015):

Average of Probabilities Voting:

$$\tilde{y} = \text{Average}\{C_1(x), C_2(x), \dots, C_n(x)\}$$

Product of probabilities voting:

$$\tilde{y} = \text{PROD}\{C_1(x), C_2(x), \dots, C_n(x)\}$$

Minimum of probabilities voting:

$$\tilde{y} = \text{MIN}\{C_1(x), C_2(x), \dots, C_n(x)\}$$

Maximum of probabilities voting:

$$\tilde{y} = \text{MAX}\{C_1(x), C_2(x), \dots, C_n(x)\}$$

## Proposed Methods

In this proposed approach, we present a new and powerful approach for plant disease classification based on weighted voting approach using Deep learning models as ensemble of base classifiers. The proposed approach is based on a weighted combination of five deep learning architectures. The weight of each DL architecture is calculated based on its performance using genetic algorithms. The optimal weights of each of the 5 architectures is optimized by the Genetic Algorithm (GA). The overall workflow of the proposed method has been depicted in this Fig. 10.

As we can see clearly, after applying the five above-mentioned architectures, using the different voting mechanisms to improve the classification results. Each of these architectures' classification results is calculated for each test instance and the final results are predicted based on the weighted majority voting results. Details of the voting mechanism used are given in the above sub-sections.

We summarize the mainly steps in this the following descriptions:

- Step 1:** First, before applying the ensemble learning on the Dataset, we perform a fine-tuning of each of the 5 Deep learning algorithms to be used in order to define the hyper parameters to be used.
- Step 2:** Applying the Global Average Pooling (GAP) layer reduces each feature map to a single number by simply taking the average of all values and minimizes overfitting by reducing the total number of parameters in all models.
- Step 3:** We proceed to combine the 5 DL algorithms. The weights of each of the 5 algorithms are optimized using the genetic algorithms.
- Step 4:** The final prediction is obtained by weighted voting using the optimal weights obtained by step 3

Now we select our weights parameter. Here, we will use Genetic algorithm approach to find the optimal weights for each classifier to increase the prediction accuracy.

#### *Genetic Algorithm for Tuning the Weight*

Now we select our weights parameters. Here, we will use Genetic algorithm approach to find the optimal weights for each classifier to increase the prediction accuracy.

The principal steps of the used genetic algorithm (Yuan *et al.*, 2012) are: (1) Initialization by producing a collection of many individuals and each individual denotes the weight of each classifier. (2) Fitness: The GA calculates the fitness of individuals based on some objective evaluation function, through learning the survival probability of individuals in the following evolution. (3) Selection: Using random or specific population rules to participate in cross and mutation. We select a certain number of excellent individuals with “more fitness” (4) frequently great individuals are usually of high fitness, that is, better.

## Experimental Results

### *Dataset*

In our work, each single deep learning model and the ensemble learning models were trained on a publicly available dataset called plant village (Hughes and Salathé, 2015). This dataset is one of the most used for the evaluation of plant disease classification which contains 54,306 images in total classified into 38 subsets. Some of them are displayed in the Fig. 11. It covers healthy and diseased leaves to their categories. For making it appropriate for the initial values of the trained models. The size of images was modified to  $224 \times 224 \times 3$  and normalization was measured by dividing the values of pixel by 255. To avoid the overfitting problem, we divide our dataset into 3 categories training 70, 20% for validation and 10% for testing. The dataset is provided also by the Ground of Truth (GT) that will help us to evaluate

the approach. Table 2 listed a detailed description of the used dataset and also the abbreviation of each class used in the next subsections.

### *B. Performance Measures*

In this study, we evaluate the performance of our proposed approach by calculating common classification indicators. we will consider the statistics of correct classification (also known as true positives) which means the number of classes that the classifier accurately classifies, misclassification (also known as false negatives) which is the number of classes that are mistakenly identified, False Positive signify the number of cases which are inaccurately marked as such. TRUE Negative (TN) is the number of examples that are not in such disease. These measures are accuracy, sensitivity and specificity. For precision is the ratio of correctly predicted positive observations to the total predicted positive observations call is the ratio of correctly predicted positive observations to all the observations in the actual class. F1-Score is the weighted average of Precision and Recall as defined in Equations.

$$precision = \frac{\sum_{i=1}^n precision_i}{n}$$

$$where\ precision_i = \frac{TP_i}{TP_i + FP_i}$$

$$recall = \frac{\sum_{i=1}^n recall_i}{n}$$

$$where\ recall_i = \frac{TP_i}{TP_i + FN_i}$$

$$accuracy = \frac{\sum_{i=1}^n accuracy_i}{n}$$

$$where\ accuracy_i = \frac{TP_i + FP_i}{TP_i + TN_i + FP_i + FN_i}$$

Where:

- $n$  = The number of classes,  $i$  the current class
- $TP$  = The True Positives
- $FP$  = The False Positives
- $FP$  = The False Positives
- $FN$  = The False Negatives

### *Simulations Results*

We used python language for programming ensemble learning architectures based on the three different voting mechanisms to the simplicity and availability of very useful DL frameworks and libraries. For instance, we used Keras and Tensor Flow to build all these architectures. All

our experiments were carried out on a Graphical Processing Unit (NVIDIA Quadro K2200) having the specifications: 32GB memory, 640 CUDA cores, 1045 MHz core clock and 80 GB/sec memory bandwidth.

For the experiments, we use 10-fold cross-validations for different architecture. For learning rate, we use 0.00001 and Adam method [41] for adaptive learning and cross-entropy as loss function. For Plant village, we trained all five architectures for all plants images and set the epochs into 100.

Table 3 summarizes the studied approaches that use DL models for plant diseases classification, where the results are based on accuracy, as well as the metrics that they used and the final results are presented as follows.

After examining these results, we notice all of the architectures showed a significant performance, starting with accuracy, Inception and VGG16 had the lower values with 91 and 92% respectively. Followed by the common result founded with AlexNet, CNN and Mobile Net with 95%. The best recall representing by Mobil Net provides almost excellent Classification results which achieved 97%. On the other hand, in the Precision indicator, Alex Net obtained the lower result with 86% followed by VGG16, InceptionV3 and Mobile Net with 89% to both and 94%, respectively and the highest value was obtained by CNN with 95%. In addition, we consider another metric which is processing time, we remark that Alex Net gets the best performance by taking the shorter time.

To improve the accuracy results for these models we implemented three voting mechanism-based ensemble learning classifications; Hard voting, weighted voting and soft voting. The last one includes diverse voting schemes that were tested, minimum probabilities, maximum probabilities, the product of probabilities and the average of probabilities. All architectures are combined and trained using all mechanism voting. So the accuracy scores related to these comparative analyses have been given in Table 4. Notice that ELA abbreviation means Ensemble Learning Architectures.

As seen in Table 4 and Fig. 12, the best performance was achieved using the weighted voting mechanism comparing with the other mechanisms voting for the plant village dataset. On the other hand, hard voting obtained the lowest performance compared to others. That means our proposed approach considers more information by combining the five architectures in the final prediction. So our goal was to find the more suitable voting mechanism by combining five architectures in ensemble learning to enhance the accuracy of proposed models and classify the leaves diseases in their classes. However, the proposed Ensemble Learning approaches have significantly improved the classification results and especially using the weighted voting technique with a value equal to 99.21%. We can say that the use of the vote of the 5 architectures allowed us to have the best classification rate as we see in Fig. 13.

Furthermore, Table 5 and 6 show the confusion matrix with the best output based on the performance measures, which are the weighted voting mechanism-based combined five models. Depending on the results, it is simple to visually evaluate the performance of the classifier and to determine which classes are highlighted by our approach. The rows are related to the output class, while they are related to the true class. The diagonal cells are associated with the observations that are correctly classified and the off-diagonal cells correspond to the incorrectly classified observations.

**Table 1:** The most successful CNN architectures

Architectures	References	Year	Nb of params	depth
LeNet-5	LeCun <i>et al.</i>	1998	60 ,000	5
AlexNet	Krizhevsky <i>et al.</i>	2012	60 million	8
VGG	Simonyan <i>et al.</i>	2014	138 million	19
GoogleNet	Szegedy <i>et al.</i>	2015	4 million	22
Inception V3	Szegedy <i>et al.</i>	2015	23 million	159
Inception V4	Szegedy <i>et al.</i>	2016	35 million	70
ResNet	He <i>et al.</i>	2016	25 million	152
MobileNetV1	Sandler <i>et al.</i>	2017	4.2 million	28
MobileNetV2	Mark <i>et al.</i>	2018	3.47 million	53

**Table 2:** Description of plant village dataset

Class name	Abbreviation	Number of images
Apple scab	AS	498
Apple black rot	ABR	484
Apple cedar apple rust	ACAR	220
Apple healthy	AH	1336
Blueberry healthy	BH	1231
Cherry powdery mildew	CPM	948
Cherry healthy	CHH	703
Corn healthy	CH	934
Corn Northern leaf blight	CNLB	798
Corn common rust	CCR	954
Corn cercospora leaf gray leaf spot	CCLGLS	409
Grape blackrot	GB	884
Grape esca	GE	1099
Grape leaf blight	GLB	828
Grape healthy	GH	341
Orange citrus greening	OCG	4361
Peach bacterial spot	PBS	1819
Peach healthy	PEH	280
Pepper bell healthy	PBH	1267
Pepper bell Bacterial spot	PBBS	781
Potato healthy	PH	116
Potato Late blight	PLB	768
Potato Early blight	PEB	824
Raspberry healthy	RH	208
Soybean healthy	SH	4202
Squash powdery mildew	SPM	1503
Strawberry healthy	STH	388
Strawberry leaf scorch	SLS	931
Tomato Target Spot	TTS	1136
Tomato Tomato mosaic virus	TTMV	307
Tomato Tomato Yellow Leaf Curl Virus	TTYLCV	4312
Tomato Bacterial spot	TBS	1739
Tomato Early blight	TEB	839
Tomato healthy	TH	1266
Tomato Late blight	TLB	1560
Tomato Leaf Mold	TLM	768
Tomato Septoria leaf spot	TSLP	1456
Tomato Spider mites Two spotted spider mite	TSMTSSm	1312

**Table 3:** Performance measures for every model

Models	Precision	Recall	F Score	Time(min)	Accuracy
VGG16	0.89	0.96	0.96	94.95	92%
Alex Net	0.86	0.96	0.95	94.17	95%
InceptioV3	0.89	0.95	0.95	133.12	91%
CNN	0.95	0.95	0.95	99.20	95%
Mobile Net	0.94	0.97	0.94	147.73	95%

**Table 4:** Performance measures for combining DL models and voting approaches

Models based voting mechnism	Precision	Recall	FScore	Accuracy
Hard ELA	0.9940	0.9940	0.9940	91.1%
SOFT ELA based Average Probabilities	0.989	0.988	0.9881	98.82%
SOFT ELA based Product Probabilities	0.9850	0.9850	0.9850	98.11%
SOFT ELA based MIN Probabilities	0.981	0.9835	0.9830	98.47%
SOFT ELA based MAX Probabilities	0.9850	0.9850	0.9850	99.14%
Proposed method				
Weighted voting ELA	0.9840	0.9841	0.9882	99.21%

**Table 5:** The confusion matrix using the weighted voting mechanism-based combined models for 38 classes

	PBBS	PH	PLB	PEB	RH	SH	SPM	STH	SLS	TTS	TTMV	TTYLCV	TBS	TEB	TH	TLB	TLM	TSLP	TSMTSSm
PBBS	720	0	36	0	0	0	0	0	0	0	25	0	0	0	0	0	0	0	0
PH	0	116	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PLB	0	0	637	0	0	0	0	0	0	37	94	0	0	0	0	0	0	0	0
PEB	0	0	0	824	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RH	0	0	0	0	208	0	0	0	0	0	0	0	0	0	0	0	0	0	0
SH	0	1	0	0	0	4102	0	0	0	0	100	0	0	0	0	0	0	0	0
SPM	0	0	0	0	0	0	1000	0	0	0	490	0	13	0	0	0	0	0	0
STH	0	0	0	0	0	0	0	287	0	0	53	0	0	48	0	0	0	0	0
SLS	0	0	0	0	0	0	0	0	931	0	0	0	0	0	0	0	0	0	0
TTS	0	0	0	0	0	0	0	0	0	1121	15	0	0	0	0	0	0	0	0
TTMV	0	0	0	0	0	0	0	0	0	0	307	0	0	0	0	0	0	0	0
TTYLCV	0	0	0	0	0	0	0	0	0	0	12	221	0	0	0	0	0	0	0
TBS	0	0	0	0	0	0	0	0	0	0	10	0	494	0	0	0	0	0	0
TEB	0	0	0	0	0	0	0	0	0	3	0	0	0	343	0	0	0	0	0
TH	0	0	0	0	0	0	0	0	0	4	0	0	0	0	299	0	0	0	0
TLB	0	0	0	0	0	18	0	0	0	12	30	0	0	0	0	1500	0	0	0
TLM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	768	0	0
TSLP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1456	0
TSMTSSm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1312

**Table 6:** The confusion matrix using the weighted voting mechanism-based combined models for 38 classes (continue)

	AS	ABR	ACAR	AH	BH	CPM	CHH	CH	CNLB	CCR	CCLGLS	GB	GE	GLB	GH	OCG	PBS	PEH	PBH
AS	498	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ABR	0	482	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0
ACAR	0	0	202	0	0	0	0	0	18	0	0	0	0	0	0	0	0	0	0
AH	0	0	0	1306	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0
BH	0	0	0	0	1231	0	0	0	0	9	0	0	0	0	0	0	0	0	0
CPM	0	1	0	0	0	908	0	40	0	15	0	0	0	0	0	0	0	0	0
CHH	0	0	0	0	0	0	703	0	0	0	0	0	0	0	0	0	0	0	0
CH	0	0	0	0	0	0	0	934	0	0	0	0	0	0	0	0	0	0	0
CNLB	0	0	0	0	0	0	0	0	748	0	0	50	0	0	0	0	0	0	0
CCR	0	0	0	0	0	0	0	0	0	954	0	0	0	0	0	0	0	0	0
CCLGLS	0	0	0	0	0	0	0	0	0	0	409	0	0	0	0	0	0	0	0
GB	0	0	0	0	0	0	0	0	0	80	704	0	0	0	0	0	0	0	0
GE	0	0	0	0	0	0	0	0	0	0	0	1099	0	0	0	0	0	0	0
GLB	0	0	0	0	0	0	0	0	0	0	0	0	828	0	0	0	0	0	0
GH	0	0	0	0	0	0	13	0	0	0	0	0	0	328	0	0	0	0	0
OCG	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4361	0	0	0	0
PBS	0	0	0	0	0	0	0	0	0	19	0	0	0	0	0	0	1800	0	0
PEH	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	280	0
PBH	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1203

**D. Comparison with Existing Works**

To verify the performance of the proposed technique, 5 methods are selected for comparison in this study. The Fig. 14. displays a comparison of prediction accuracy between the proposed Ensembles deep learning-based the

weighted voting mechanism with existing work on plant leaves diseases using the Plant village dataset.

In (Sladojevic *et al.*, 2016) used deep CNN to detect plant diseases from leaf images. This method can distinguish between healthy leaves and some dissimilar diseases. The CNN's whole accuracy is 94.60%. In

addition, the authors (Wang *et al.*, 2017) used also the CNN method with 50 hidden layers using Neuroph Studio framework as an IDE to build a more facilitated deep Convolutional Neural Network (CNN) whereby the convolution and pooling feature extractions were embedded in the Neuroph library. In this study, the authors found the proposed CNN gave an overall accuracy of 92.85%. On the other hand, in (Ennoui *et al.*, 2021b) proposed a DL approach to estimate disease severity. The best model was trained with transfer learning, which yields an overall accuracy of 90.4%. Comparing with other work, authors in (Ennoui *et al.*, 2021c) used machine learning approach based PDE; this study reported an average classification accuracy of 95.9%.

As you see our proposed method which EL-based weighted voting architectures provided an overall accuracy of 99.21% and proved its feasibility. As far as we can ascertain, our proposed is performs better than any in the literature to date.

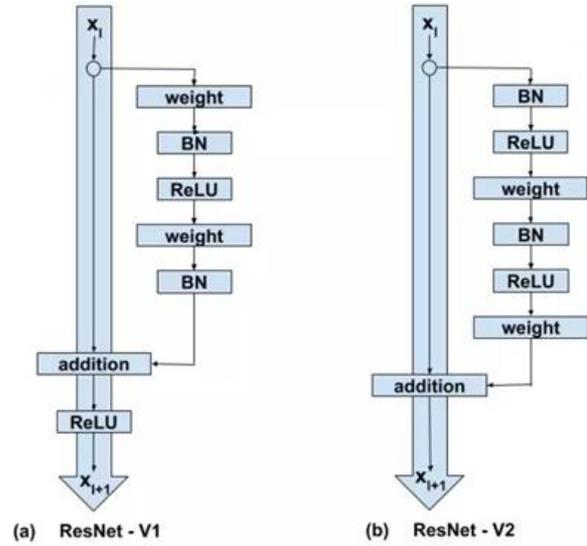


Fig. 8: Res Net Architecture and its versions

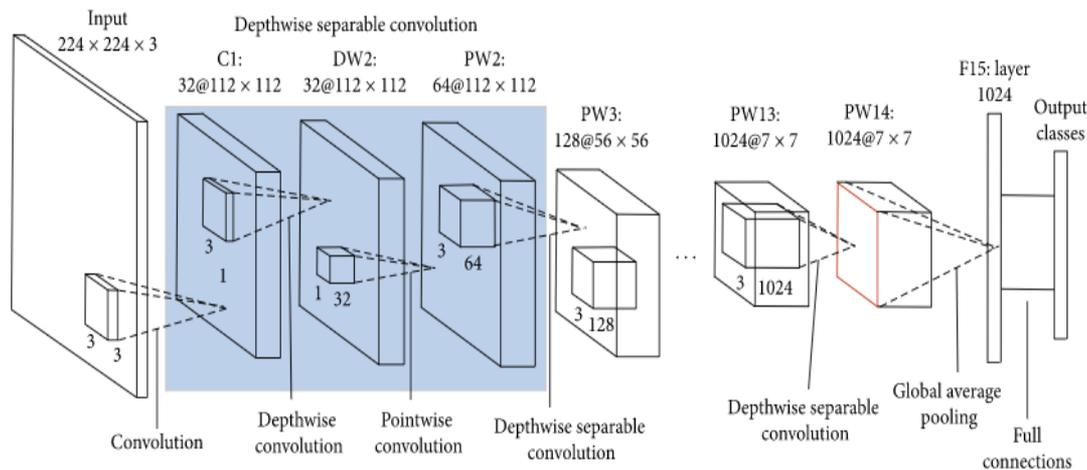


Fig. 9: Mobile Net Architecture

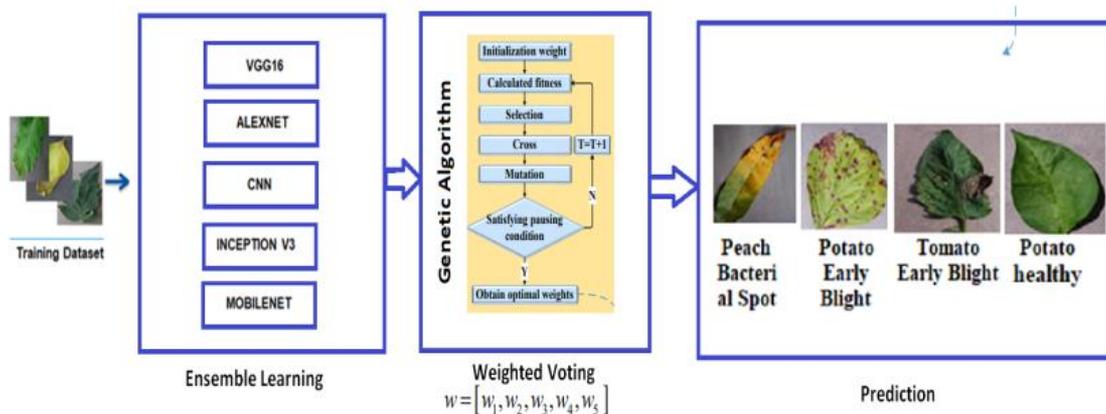
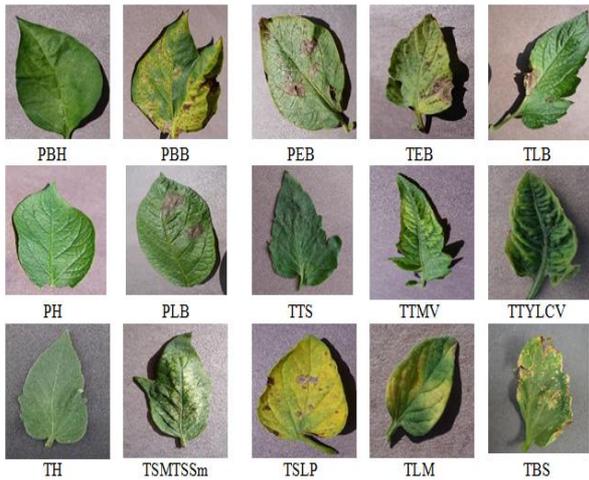
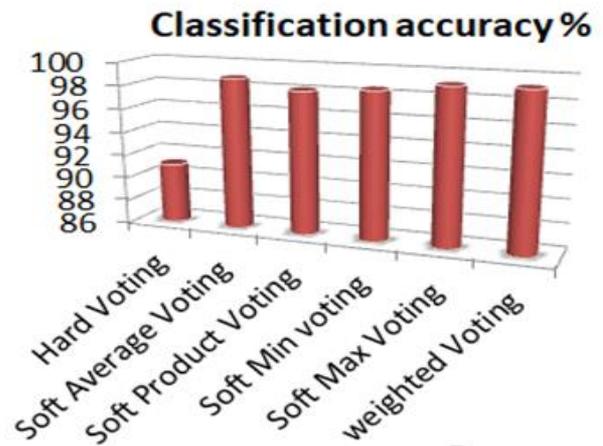


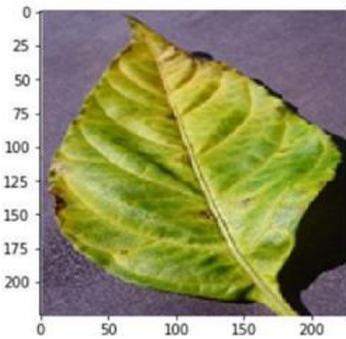
Fig. 10: Overall workflow of proposed weighted voting ensemble model for plant leaf disease classification



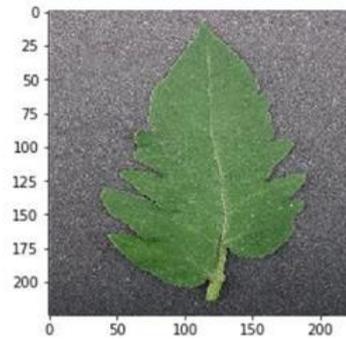
**Fig. 11:** Some healthy and diseased plants images from the Plant Village dataset



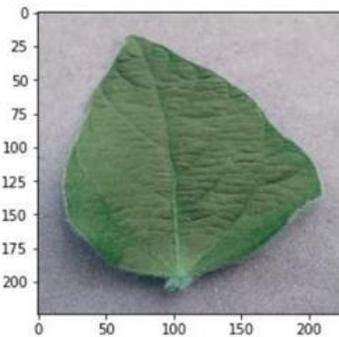
**Fig. 12:** Classification Accuracy for plant village images dataset



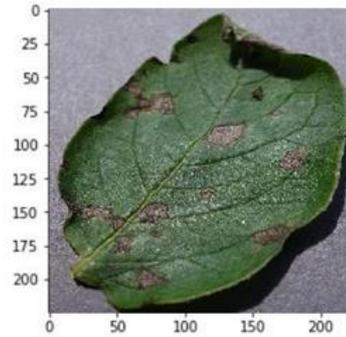
SOURCE : class : Pepper\_bell\_Bacterial\_spot  
 PREDICTED : class : Pepper\_bell\_Bacterial\_spot  
 Exactitude : 0.9903



SOURCE : class : Tomato\_healthy  
 PREDICTED : class : Tomato\_healthy  
 Exactitude : 0.9978



SOURCE : class : Soybean\_healthy  
 PREDICTED : class : Soybean\_healthy  
 Exactitude : 0.9942



SOURCE : class : Potato\_Early\_blight  
 PREDICTED : class : Potato\_Early\_blight  
 Exactitude : 0.9946

**Fig. 13:** Classification accuracy for plant village images



**Fig. 14:** A comparison of classification accuracy of the proposed method with recent approaches

## Conclusion

This study proposes and implements a weighted voting technique for plant disease detection and classification based on an Ensemble Learning approach. The main contribution is to improve classification accuracy by using a weighted voting mechanism based on a Genetic algorithm to propose the optimal weights. The obtained results prove the effectiveness of our proposed method by giving an average accuracy of 99.21% on the plant village dataset. We can conclude that this approach can be used to monitor the quality and quantity of plant production. In future work, we intend to deploy this technique on real-world applications to monitor, detect and classify the immense range of plant diseases automatically.

## Acknowledgment

We would like to express our gratitude and thank you to the LISAC laboratory, the Faculty of Sciences, the University Sidi Mohamed Ben Abdellah, Fez, Morocco for the funding and those who are involved directly or indirectly in this research.

## Author's Contributions

**Assia Ennoui:** Developpement, experimentations testing, validation and writing the manuscript.

**Noura Ouled Sihamman:** Developpement, experimentations testing.

**My Abdelouahed Sabri:** Concept developpement, experimentations testing, validation and proof reading.

**Abdellah Aarab:** Concept developpement, experimentations validation and proof reading.

All authors read and approved the final manuscript.

## Ethics

All information provided in this study is confidential and unique. This paper has neither been published nor is

under review elsewhere. There are no ethical issues associated with this research.

## References

- Abdelhafiz, D., Yang, C., Ammar, R., & Nabavi, S. (2019). Deep convolutional neural networks for mammography: Advances, challenges and applications. *BMC bioinformatics*, 20(11), 1-20. doi.org/10.1186/s12859-019-2823-4
- Barbedo, J. G. A. (2013). Digital image processing techniques for detecting, quantifying and classifying plant diseases. *SpringerPlus*, 2(1), 1-12. https://springerplus.springeropen.com/articles/10.1186/2193-1801-2-660
- Biswas, M., Kuppili, V., Saba, L., Edla, D. R., Suri, H. S., Cuadrado-Godia, E., ... & Suri, J. S. (2019). State-of-the-art review on deep learning in medical imaging. *Front Biosci (Landmark Ed)*, 24, 392-426. doi.org/10.2741/4725
- Brahimi, M., Boukhalfa, K., & Moussaoui, A. (2017). Deep learning for tomato diseases: Classification and symptoms visualization. *Applied Artificial Intelligence*, 31(4), 299-315. doi.org/10.1080/08839514.2017.1315516
- Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2), 123-140. doi.org/10.1007/BF00058655
- Cao, J., Kwong, S., Wang, R., Li, X., Li, K., & Kong, X. (2015). Class-specific soft voting based multiple extreme learning machines ensemble. *Neurocomputing*, 149, 275-284. doi.org/10.1016/j.neucom.2014.02.072
- Chen, Y., Wong, M. L., & Li, H. (2014). Applying Ant Colony Optimization to configuring stacking ensembles for data mining. *Expert systems with applications*, 41(6), 2688-2702. doi.org/10.1016/j.eswa.2013.10.063
- Delgado, J., & Ishii, N. (1999, August). Memory-based weighted majority prediction. In *SIGIR Workshop Recomm. Syst.* Citeseer (p. 85).
- Du, K. L., & Swamy, M. N. S. (2019). Combining Multiple Learners: Data Fusion and Ensemble Learning. In *Neural Networks and Statistical Learning* (pp. 737-767). Springer, London. doi.org/10.1007/978-1-4471-7452-3\_25
- El-Sayed, A. M. A., Rida, S. Z., & Gaber, Y. A. (2020). Dynamical of curative and preventive treatments in a two-stage plant disease model of fractional order. *Chaos, Solitons & Fractals*, 137, 109879. doi.org/10.1016/j.chaos.2020.109879
- Ennoui, A., Sabri, M. A., & Aarab, A. (2021a). Plant Diseases Detection and Classification Based on Image Processing and Machine Learning. In *Intelligent Systems in Big Data, Semantic Web and Machine Learning* (pp. 283-290). Springer, Cham. doi.org/10.1007/978-3-030-72588-4\_20

- Ennoui, A., Sihamman, N. O., Sabri, M. A. & Aarab, A. (2021a). Early Detection and Classification Approach for Plant Diseases based on MultiScale Image Decomposition. *Journal of Computer Science*, 17(3), 284-295. doi.org/10.3844/jcssp.2021.284.295.
- Ennoui, A., Sihamman, N. O., Sabri, M. A., & Aarab, A. (2021c, January). Analysis and Classification of Plant Diseases Based on Deep Learning. In *International Conference on Digital Technologies and Applications* (pp. 123-133). Springer, Cham. doi.org/10.1007/978-3-030-73882-2\_12
- Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. *Computers and Electronics in Agriculture*, 145, 311-318. doi.org/10.1016/j.compag.2018.01.009
- Filali, Y., Khoukhi, H. E., Sabri, M. A., & Aarab, A. (2020). Efficient fusion of handcrafted and pre-trained CNNs features to classify melanoma skin cancer. *Multimedia Tools and Applications*, 79(41), 31219-31238. doi.org/10.1007/s11042-020-09637-4
- Fina, F., Birch, P., Young, R., Obu, J., Faithpraise, B., & Chatwin, C. (2013). Automatic plant pest detection and recognition using k-means clustering algorithm and correspondence filters. *International Journal of Advanced Biotechnology and Research*, 4(2), 189-199. <https://core.ac.uk/download/pdf/30608241.pdf>
- Fuentes, A., Yoon, S., Kim, S. C., & Park, D. S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors*, 17(9), 2022. doi.org/10.3390/s17092022
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778). doi.org/10.1007/s11042-017-4440-4
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2818-2826).
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.
- Huang, K. Y. (2007). Application of artificial neural network for detecting Phalaenopsis seedling diseases using color and texture features. *Comput. Electron. Agric.* 57, 3-11. doi.org/10.1016/j.compag.2007.01.015
- Hughes, D., & Salathé, M. (2015). An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint arXiv:1511.08060*. <https://arxiv.org/abs/1511.08060>
- Kang, X. B., Lin, G. F., Chen, Y. J., Zhao, F., Zhang, E. H., & Jing, C. N. (2020). Robust and secure zero-watermarking algorithm for color images based on majority voting pattern and hyper-chaotic encryption. *Multimedia Tools and Applications*, 79(1), 1169-1202. doi.org/10.1007/s11042-019-08191-y
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 1097-1105. doi.org/10.1109/CVPR.2015.7298594
- Kuncheva, L. I., & Rodríguez, J. J. (2014). A weighted voting framework for classifiers ensembles. *Knowledge and Information Systems*, 38(2), 259-275. doi.org/10.1007/s10115-012-0586-6
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444. doi.org/10.1038/nature14539
- LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4), 541-551. doi.org/10.1162/neco.1989.1.4.541
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324. doi.org/10.1109/5.726791
- Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... & Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical image analysis*, 42, 60-88. doi.org/10.1016/j.media.2017.07.005
- Lundervold, A. S., & Lundervold, A. (2019). An overview of deep learning in medical imaging focusing on MRI. *Zeitschrift für Medizinische Physik*, 29(2), 102-127. doi.org/10.1016/j.zemedi.2018.11.002
- Mathews, D. M. (2010, October). Optimizing detection and management of virus diseases of plants. In *Proceedings of the Landscape Disease Symposium* (pp. 10-20). <https://ucanr.edu/sites/ucnfa/files/62197.pdf>
- Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Frontiers in plant science*, 7, 1419. doi.org/10.3389/fpls.2016.01419
- Onan, A., Korukoğlu, S., & Bulut, H. (2016). A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification. *Expert Systems with Applications*, 62, 1-16.3 doi.org/10.1016/j.eswa.2016.06.005
- Oppenheim, D., Shani, G., Erlich, O., & Tsror, L. (2019). Using deep learning for image-based potato tuber disease detection. *Phytopathology*, 109(6), 1083-1087. doi.org/10.1094/PHYTO-08-18-0288-R

- Park, H., Eun, J. S. & Kim, S. H. (2017). Image-based disease diagnosing and predicting of the crops through the deep learning mechanism, In Information and Communication Technology Convergence (ICTC), IEEE 2017 International Conference on, pp. 129-131. doi.org/10.1109/ICTC.2017.8190957
- Petropoulos, F., Hyndman, R. J., & Bergmeir, C. (2018). Exploring the sources of uncertainty: Why does bagging for time series forecasting work?. *European Journal of Operational Research*, 268(2), 545-554. doi.org/10.1016/j.ejor.2018.01.045
- Phadikar, S., & Sil, J. (2008, December). Rice disease identification using pattern recognition techniques. In 2008 11th International Conference on Computer and Information Technology (pp. 420-423). IEEE. doi.org/10.1109/ICCITECHN.2008.4803079
- Sahni, V., Srivastava, S., & Khan, R. (2021). Modelling Techniques to Improve the Quality of Food Using Artificial Intelligence. *Journal of Food Quality*, 2021. doi.org/10.1155/2021/2140010
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4510-4520). doi.org/10.1109/CVPR.2018.00474
- Sibiya, M., & Sumbwanyambe, M. (2019). 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. doi.org/10.3390/agriengineering1010009
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*. <https://arxiv.org/abs/1409.1556>
- Sladojevic, S., Arsenovic, M. anderla, A., Culibrk, D., & Stefanovic, D. (2016). Deep neural networks based recognition of plant diseases by leaf image classification. *Computational intelligence and neuroscience*, 2016. doi.org/10.1155/2016/3289801
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9). doi.org/10.1109/CVPR.2015.7298594
- Too, E. C., Yujian, L., Njuki, S., & Yingchun, L. (2019). A comparative study of fine-tuning deep learning models for plant disease identification. *Computers and Electronics in Agriculture*, 161, 272-279. doi.org/10.1016/j.compag.2018.03.032
- Türkoğlu, M., & Hanbay, D. (2019). Plant disease and pest detection using deep learning-based features. *Turkish Journal of Electrical Engineering & Computer Sciences*, 27(3), 1636-1651. doi.org/10.3906/elk-1809-181
- Tyagi, N., Khan, R., Chauhan, N., Singhal, A., & Ojha, J. (2021). E-Rickshaws Management for Small Scale Farmers using Big Data-Apache Spark. In *IOP Conference Series: Materials Science and Engineering* (Vol. 1022, No. 1, p. 012023). IOP Publishing. doi.org/10.1088/1757-899X/1022/1/012023
- Vijai, S., Namita, S., & Shikha, S. (2020). A review of imaging techniques for plant disease detection, *Artificial Intelligence in Agriculture*, Volume 4, Pages 229-242. doi.org/10.1016/j.aiaa.2020.10.002
- Wang, G., & Ma, J. (2011). Study of corporate credit risk prediction based on integrating boosting and random subspace. *Expert Systems with Applications*, 38(11), 13871-13878. doi.org/10.1016/j.eswa.2011.04.191
- Wang, G., Sun, Y., & Wang, J. (2017). Automatic image-based plant disease severity estimation using deep learning. *Computational intelligence and neuroscience*, 2017. doi.org/10.1155/2017/2917536
- Wolpert, D. H. (2011). Stacked generalization, *Neural Networks*, vol. 5, no. 2, pp. 241-259. doi.org/10.1016/S0893-6080(05)80023-1
- Xiao, Y., Wu, J., Lin, Z., & Zhao, X. (2018). A deep learning-based multi-model ensemble method for cancer prediction. *Computer methods and programs in biomedicine*, 153, 1-9. doi.org/10.1016/j.cmpb.2017.09.005
- Xu, G., Zhang, F., Shah, S. G., Ye, Y., & Mao, H. (2011). Use of leaf color images to identify nitrogen and potassium deficient tomatoes. *Pattern Recognition Letters*, 32(11), 1584-1590. doi.org/10.1016/j.patrec.2011.04.020
- Zhang, K., Wu, Q., Liu, A., & Meng, X. (2018). Can deep learning identify tomato leaf disease?. *Advances in Multimedia*, 2018. doi.org/10.1155/2018/6710865
- Zhou, Z. H. (2009). Ensemble learning, *Encyclopedia of Biometrics*. doi.org/10.1007/978-0-387-73003-5\_293