

TOPICAL REVIEW

Machine Learning and Deep Learning for Plant Disease Classification and Detection

VASILEIOS BALAFAS^{ID}, EMMANOUIL KARANTOUMANIS^{ID},
MALAMATI LOUTA^{ID}, (Senior Member, IEEE),
AND NIKOLAOS PLOSKAS^{ID}

Department of Electrical and Computer Engineering, University of Western Macedonia, 50100 Kozani, Greece

Corresponding author: Nikolaos Ploskas (nploskas@uowm.gr)

This work was supported by the European Regional Development Fund of the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation under Project MIS 5047196. The publication of the article in OA mode was financially supported by HEAL-Link.

ABSTRACT Precision agriculture is a rapidly developing field aimed at addressing current concerns about agricultural sustainability. Machine learning is the cutting edge technology underpinning precision agriculture, enabling the development of advanced disease detection and classification methods. This paper presents a review of the application of machine learning and deep learning techniques in precision agriculture, specifically for detecting and classifying plant diseases. We propose a novel classification scheme that categorizes all relevant works in the associated classes. We separate the studies into two main categories depending on the methodology that they use (i.e., classification or object detection). In addition, we present the available datasets for plant disease detection and classification. Finally, we perform an extensive computational study on five state-of-the-art object detection algorithms on PlantDoc dataset to detect diseases present on the leaves, and eighteen state-of-the-art classification algorithms on PlantDoc dataset to predict whether or not there is a disease in a leaf. Computational results show that object detection accuracy is high with YOLOv5. For the image classification task, the networks ResNet50 and MobileNetv2 have the most optimal trade-off on accuracy and training time.

INDEX TERMS Classification, deep learning, disease detection, machine learning, object detection, precision agriculture.

I. INTRODUCTION

Any nation's economic development depends significantly on agriculture. Meeting the population's current food needs has become a difficult problem because of the growing population, frequent changes in weather, and limited resources. On top of the aforementioned challenges, crop diseases have been increasing in severity and scale. Crop diseases cause production losses that can be mitigated with continuous monitoring. Researchers from the Food and Agriculture Organization of the United Nations predicted that plant diseases alone cost the global economy about US\$220 billion annually [1]. Developing new methods to detect diseases on plants or leaves at an early stage can significantly increase yield potential.

The associate editor coordinating the review of this manuscript and approving it for publication was Pasquale De Meo^{ID}.

Precision agriculture is a rapidly developing field aimed at addressing current concerns about agricultural sustainability. Machine learning (ML) is the cutting edge technology underpinning precision agriculture, allowing the machine to learn without having to be programmed directly, and in conjunction with Internet of Things (IoT) enabled farm equipment, is the future of agriculture. There are several studies that have employed or proposed ML methods to detect or classify plant diseases. The majority of these works take as input a plant/leaf image and detect whether or not there is a disease. These works treat the problem as a classification one, either binary classification (healthy or diseased plant/leaf) or multi-class classification (where various diseases are targeted). Classical ML methods, like Random Forest (RF) and Deep Learning (DL) ones, have been utilized for this purpose. On the other hand, there are fewer works that aim to detect both the type of the disease and the diseased regions

of the input image, i.e., treating the problem as an object detection one. The latter problem is more important than the classification problem in cases where either there exist multiple plant diseases in the input image or we want to know the exact region of the diseased plants in an image that covers a large part of the crop (such images are taken by Unmanned Aerial Vehicles (UAVs)). Moreover, the object detection problem is more difficult than the classification problem and the DL methods used to detect objects do not have good results in uncontrolled places, i.e., in images where the objects are placed in a noisy environment.

In this paper, we aim to review ML and DL methods that have been applied to either classify or detect plant diseases. Various papers reviewed ML and DL methods in precision agriculture. Below, we present those papers that include works focused on plant disease detection. For example, Mekonnen et al. [2] provided a study of how different ML techniques are used in sensor data analytics in the agricultural sector. The authors highlighted the benefits of Artificial Intelligence (AI) in agriculture and emphasized that IoT and AI are increasingly becoming essential components for enhancing agricultural productivity and efficiency. They concluded that although clustering is the most common learning model for such problems, there is no general prescription for selecting algorithms, and most of the time, it is a trial and error process in doing so. Finally, they examined some of the ML methods used in agriculture such as regression algorithms, Decision Trees (DT), ensemble learning, Bayesian models, Support Vector Machines (SVM), and Artificial Neural Networks (ANNs).

Benos et al. [3] presented a review of various works aimed at exploring ML in agriculture. Initially, they described the four basic categories in which ML algorithms are involved. Crop management is subdivided into five categories (yield prediction, disease detection, weed detection, crop recognition, and crop quality), water management, soil management, and livestock management. They performed an extensive search on various search engines and selected articles for the period from 2018 to 2020. Finally, they concluded that: (i) a large percentage of research articles were about crop management, (ii) ANNs were the most effective ML models, (iii) the most investigated crops in the papers were maize, wheat, rice, and soybean, and finally, (iv) the most utilized input data was mainly RGB images and then weather, soil, water, and crop quality.

Li et al. [4] presented a review of the research progress of DL methods for crop leaf disease identification. Initially, they reviewed the relevant metrics, the datasets used in the computational experiments, and the data enhancement methods. Then, they presented works based on DL methods in order to predict diseases in crop leaves. Finally, they concluded that: (i) most DL frameworks are not robust, achieving good results only on specific datasets, (ii) most works use synthetic images generated in the laboratory, and (iii) studies focused on early detection of disease are limited.

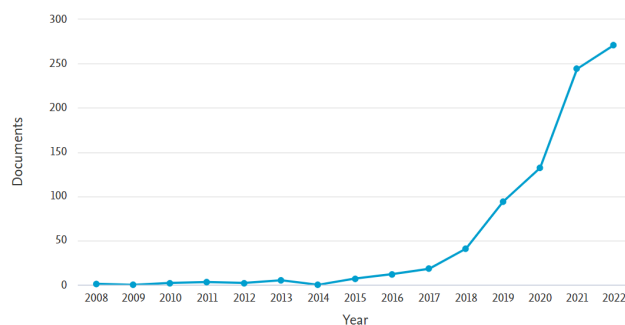


FIGURE 1. Publications per year for papers including the words “plant”, “disease”, “detection”, and “machine learning” either on their title or abstract or keywords (source: Scopus).

Liu and Wang [5] reviewed works focusing on methods for detecting plant diseases and pests published during the period from 2014 to 2020. They presented various methods proposed in the literature and discussed their advantages and disadvantages. They also presented the publicly available datasets for plant diseases and pest detection methods. Finally, they concluded that: (i) the majority of research findings are limited to the laboratory setting and are solely relevant to plant disease and pest images obtained during that period, (ii) early diagnosis is very difficult, and thus, only a few works deal with this, and (iii) less time- and memory-consuming DL models are in need for real-time prediction of diseases and pests.

Yuan et al. [6] examined the progress in technologies utilized for agricultural disease image recognition such as DL and transfer learning. They also analyzed key challenges that require attention to progress research in this field, including creating image datasets, choosing big data auxiliary domains, and improving transfer learning techniques. It is essential to create image datasets captured under actual production conditions to develop a feasible image recognition system for agricultural diseases.

Although there are numerous review papers discussing ML and DL techniques used in precision agriculture, there are only a few review papers that focus on plant disease detection methods, and none of them discusses the two main techniques used in the literature, classification and object detection methods. Typically, the review papers devote a section for plant disease detection methods. In addition, all reviews present works published until 2020. As shown in Figure 1, the publications that deal with ML and DL methods for plant disease detection have doubled since 2020. Therefore, there is a need for a comprehensive review on ML and DL methods for plant disease detection and classification.

In addition, computational comparisons of algorithms used for plant disease detection and classification are missing in the literature. Most papers in this area compare their proposed algorithm with a single state-of-the-art method. In this work, we present a comparison of five state-of-the-art object detection algorithms for identifying diseases on

plants and eighteen state-of-the-art classification algorithms for plant disease classification on a widely used dataset.

Our contributions can be summarized as follows:

- We present a detailed review of ML and DL algorithms for plant disease detection and classification. Most studies review works until 2020 and we also include recent works (published until June 2022). We distinguish these works based on the goal of the methods used, i.e., whether classification or object detection is targeted. We present a thorough review on 65 works for classification and 14 works for object detection.
- We propose a novel classification scheme that categorizes all relevant works in the associated classes. By classifying relevant papers according to these classes, we can easily provide an overview of different techniques used in the field.
- We provide a summary of the existing datasets that are being used for plant disease detection and classification tasks. We include details about the types of data they contain, as well as the classes and labels available within these datasets.
- We perform extensive computational analysis on five state-of-the-art object detection algorithms for plant disease detection and eighteen state-of-the-art classification algorithms for plant disease classification on a widely used dataset. This is the first time that such a systematic study has been carried out.
- We discuss gaps in the existing literature and future directions for this field.

The structure of the paper is as follows. Section II reviews research works that proposed/used ML and DL algorithms for predicting/recognizing plant diseases. In this section, we separate the papers according to whether they used classification or object detection techniques. In Section III, we propose a classification scheme, categorize all relevant works in the associated classes and present statistics of the most common methods/algorithms/datasets/metrics used by the reviewed papers. Section IV provides an overview of the datasets that are commonly used in the literature for plant disease detection and classification using both object detection and classification techniques. Section V includes the computational comparison of five state-of-the-art object detection algorithms for plant disease detection and eighteen state-of-the-art classification algorithms for plant disease classification. In Section VI, we highlight the practical implication of our findings. Section VII includes the open challenges that need to be addressed, while Section VIII includes the future directions for addressing these open challenges. Finally, Section IX summarizes the work of this paper.

II. LITERATURE REVIEW

This section provides a detailed literature review of disease detection techniques. All papers have been published prior to October 2022. To identify relevant studies on ML and DL methods for plant disease detection and classification,

we conducted a search using various search engines, including Scopus, ScienceDirect, Scholar, and Web of Science. Keywords like “machine learning”, “deep learning”, “classification”, “disease detection”, “healthy plant”, and “diseased plant”, were used. Once a relevant work was identified, its references were scanned in order to find relevant works that were not found in the initial search procedure. The abstract of each paper was reviewed in order for all co-authors to decide its appropriateness and inclusion in the paper. All papers that did not have a focus on the ML algorithms were excluded. For example, there exist papers that refer to plant disease detection using ML methods but their main focus is the data gathering part, e.g., through UAVs or Unmanned Ground Vehicles (UGVs). In addition, non-English texts, technical reports, review papers, and Master’s and Doctoral theses were excluded. Overall, 79 papers (65 for classification and 14 for object detection) were discovered.

A. CLASSIFICATION

Mohanty et al. [7] trained AlexNet [8] and GoogLeNet [9] for detecting 26 diseases on 14 crop species using the PlantVillage dataset [10] with 54,306 images. In the preprocessing step, they downsized images to 256×256 . They also experimented with the original PlantVillage dataset, with a gray-scaled version of this dataset, and a version in which the leaves were segmented. The models were trained from scratch as well as by utilizing pre-trained models on the ImageNet dataset through transfer learning. The authors reported an accuracy of 99.35% and 85.53% using the GoogLeNet and AlexNet models, respectively.

A plant disease detection approach based on Convolutional Neural Network (CNN) using the Caffe DL framework [11] was proposed by Sladojevic et al. [12]. They collected images from various sources and they used data augmentation techniques (affine transformation, perspective transformation, and rotation) to generate more images. In the preprocessing step, they downsized images to 256×256 and removed the duplicates. In addition, they did not consider images with smaller resolution than 500px. They modified the CaffeNet architecture by altering the last layer and the output of the softmax layer. They achieved an accuracy of 96.3% on average.

In their work, Amara et al. [13] applied the LeNet architecture [14] to classify diseases in banana leaves, belonging to three different categories. In the preprocessing step, they downsized images to 60×60 and converted to grayscale. They tested the proposed architecture on the PlantVillage dataset, achieving an accuracy between 92% and 99%.

Brahimi et al. [15] used two CNN models, AlexNet and GoogleNet, on the PlantVillage dataset for predicting diseases on tomato leaves. They pre-trained the models on ImageNet and then fine-tuned them by replacing the output layer with a new one having nine classes, as the number of the diseases that they consider in this study. In the preprocessing step, they downsized images to 256×256 . GoogleNet

achieved an accuracy from 97.71% to 99.19%, while AlexNet achieved an accuracy from 94.35% to 98.66%.

Cruz et al. [16] implemented a CNN network based on the LeNet architecture for detecting olive quick decline symptoms. They initially trained the network on the PlantVillage dataset, then retrained on a custom dataset. They improved the performance of the network by providing information about the edge patterns and the shape information. In the preprocessing step, they downsized images to 256×256 . The authors reported an accuracy of 99%.

DeChant et al. [17] proposed a three-stage approach using CNN models for identifying NLB-infected maize plants. A three-stage approach was utilized in this study by initially training various CNNs to detect lesions in small patches of the images. Then, the CNNs generated heat maps indicating regions in the images that may be affected. Finally, they trained the CNN models using as input the heat maps of the original images. They generated a custom dataset of 1,796 images. Their approach resulted in a classification accuracy of 96.7%.

In their study, Liu et al. [18] employed the AlexNet architecture to detect apple leaf diseases, utilizing a dataset of 13,689 images that contained four diseases: mosaic, brown spot, rust, and *Alternaria* leaf spot. Prior to classification, they performed image preprocessing by rotating and sharpening the images. Their approach yielded an overall accuracy of 97.62

Lu et al. [19] introduced a multi-stage CNN architecture that was based on AlexNet, aimed at identifying diseases in rice plants. The authors collected images for their study from an agricultural pest and insect pests database, as well as from a book that contained images of diseased plants. They preprocessed the images by resizing them to 512×512 pixels and then applied the ZCA-Whitening technique to remove the correlation between data. The accuracy of the proposed model was 95.48%.

Oppenheim and Shani [20] used a CNN based on VGG [21] to classify potatoes into five classes, four diseased ones and a healthy potato class. They acquired 400 images of contaminated potatoes using three simple digital cameras and augmented data by flipping and cropping the images. In the preprocessing step, they downsized images to 224×224 and converted to grayscale. They experimented with different ratios of train-test sets and concluded that the CNN can achieve an accuracy between 83% for the model trained with only 10% of the data to 96% for the model trained with 90% of the data.

Wang et al. [22] trained and fine-tuned four CNN models, i.e., VGG16 [21], VGG19 [21], InceptionV3 [23], and ResNet50 [24], for detecting apple leaf black rot. The authors utilized the PlantVillage dataset, which contained two classes of images: healthy apple leaves and apple leaf black rot. The images were further categorized by botanists into four stages: healthy, early stage, middle stage, and end-stage. In the preprocessing step, they downsized images to various

ratios according to the CNN that they used. They evaluated several models, and the VGG16 model produced the best results, achieving an accuracy of 90.4%.

Abdulridha et al. [25] demonstrated a non-destructive remote sensing approach for detecting the Laurel Wilt disease in infected avocado trees. They used a portable spectral data collection system to classify the data into healthy or not based on causes that elicit comparable symptoms, such as iron and nitrogen deficiency. Two sets of images with spectral resolutions of 10nm and 40nm were gathered in order to feed the DT and MultiLayer Perceptron (MLP) neural network model. They concluded that MLP has a near-perfect classification accuracy (100% at the early stage and 91% at the late stage) compared to DT (82% at the early stage and 82% at the late stage).

Barbedo [26] investigated the major factors influencing the design and performance of CNNs for plant disease recognition. To anticipate corn diseases, they used the PDDb dataset [27] that includes 50,000 images and 171 diseases. They also studied nine factors that influence disease detection in maize fields. They trained the model with four different datasets and the best accuracy was 87% with a subdivided dataset.

Ferentinos [28] employed simple leaf images to perform plant disease detection and analysis using CNN models. They used the PlantVillage dataset with 87,848 images. This dataset has 25 plants and 58 distinct classes. Furthermore, five different CNN algorithms were used: AlexNet, AlexNetOwTbn [29], GoogLeNet, Overfeat [30], and VGG. In the preprocessing step, they downsized images to 256×256 . The best results were obtained by the VGG algorithm with 99.48% success rate at the original image dataset and 98.87% success rate at the preprocessed image dataset.

Lu et al. [31] studied the possibility of detecting multiple diseases on tomato leaves at different stages using a portable high-resolution spectral sensor. They utilized the K-Nearest Neighbor (KNN) algorithm to classify the sensor data into four categories: healthy, asymptomatic, early stage, and late stage. Principal Component Analysis (PCA) was used to analyze 57 spectral vegetation indices. They used six principal components and two spectral vegetation indices compositions. The healthy leaves had an accuracy between 85.7% and 100%, the asymptomatic leaves between 86.4% and 100%, the early stage leaves between 73.5% and 93.0%, and the late stage leaves between 77.1% and 100%, depending on the principal component.

In their study, Fuentes et al. [32] proposed a diagnosing system based on a refinement filter bank for detecting diseases and pests on tomato plants. They aimed to overcome the issues of class imbalance and false positives by constructing a refinement filter bank framework. The primary diagnosis unit was the bounding box generator, which was used to create bounding boxes containing the diseased area's position. The secondary diagnosis unit was the CNN filter bank, which was used to validate the results. The proposed method achieved

a recognition rate of around 96% on a set of 5,000 images collected from various tomato farms in South Korea.

Kerkech et al. [33] presented a method that uses a CNN, namely LeNet, and histograms to forecast diseased areas of grapevines and identify symptoms in vineyards. In order to overcome memory constraints, the streaming technique was utilized in the learning process for loading a small amount of data in each iteration. They collected the data through visible-domain images taken by UAVs. They conducted experiments by utilizing various color spaces, vegetation indices, and their combination, to detect plant diseases. The approach achieved an accuracy of 95.8

Pineda et al. [34] predicted the disease caused by the bacterium *Dickeya dadantii* in melon leaves. They used three ML algorithms to train models and they used thermography and fluorescence imaging techniques to produce more accurate results. The ML algorithms that they used were Logistic Regression Analysis (LRA), SVM, and ANNs. They found that ANNs had better accuracy than the other models (99.1%) when classifying images as entire leaves.

Sharif et al. [35] proposed a two-phase hybrid technique for detecting and categorizing diseases in plants, specifically citrus fruits and leaves. The first phase involved identifying lesion areas, while the second phase focused on classifying the type of disease present. To select the most relevant features, the authors used a hybrid feature selection method that included PCA score, entropy, and a skewness-based covariance vector. This information is provided as input into the MultiClass SVM (M-SVM) for disease classification. They used images from open databases (PlantVillage and Citrus Diseases Database [36]) and a custom-collected image database. The proposed method achieved an accuracy of 97%.

Zhang et al. [37] enhanced maize leaf disease recognition accuracy and reduced the number of network parameters. They gathered 500 images from several sources, including the PlantVillage dataset, and divided the data into eight categories for diseased maize leaves and one category for healthy leaves. To increase the sample of images, they rotated the images by 90°, 180°, and 270°. They used Cifar10 [38] and GoogLeNet, changing the base learning rate to impact the network's identification accuracy. They additionally tweaked hyperparameter settings and enhanced the models. In the preprocessing step, they downsized images to 224 × 224. They achieved an accuracy of 98.9% for the GoogLeNet model and 98.8% for the Cifar10 model.

An early disease detection technique for avocado trees to identify Laurel Wilt was proposed by Abdulridha et al. [39]. They classified the data into five main categories: Laurel Wilt disease, healthy trees, trees infected by phytophthora root rot, and trees with iron and nitrogen deficiencies. To produce better results, they used image acquisition and preprocessing techniques, and feature extraction techniques. The classification was performed with two classification methods, MLP and KNN. They used a custom dataset acquired through two sensing systems. The best results to

predict Laurel Wilt was achieved from the MLP method with an accuracy of 99%.

Al-Saddik et al. [40] studied the optimal spectral bands for the development of a multispectral camera used on a UAV for recognizing diseased grapevine fields with the disease *Flavescence dorée*. The specific disease is a contagious one that is incurable and can lead to severe production loss. For selecting the optimum spectral bands capable of distinguishing infected from healthy leaves, two spectral analysis methodologies were presented. The first one performs a feature selection strategy, utilizing the successive project algorithm with several spectral preprocessing approaches. The second method looks at a variety of classic vegetation metrics. The two classifiers employed in this research are the SVM and discriminant analysis. The most accurate models were calculated as a function of the grapevine variety in question. The successive projection algorithm approach was better when it came to common vegetation metrics, with an accuracy in classification greater than 96%.

A new dataset consisting of a large number of labeled images of leaves captured in real environments was introduced by Arsenovic et al. [41]. They also proposed an augmentation technique that utilizes Generative Adversarial Networks (GANs). A two-stage CNN algorithm was then proposed for plant disease detection, and the proposed model achieved an accuracy of 93.67%.

The focus of Barbedo [42] was on a distinct issue related to disease detection. They pointed out that current databases lacked the necessary images to accurately represent the various conditions and characteristic symptoms, which could not be reproduced using existing data augmentation techniques. As a result, they looked at using particular lesions and areas for the task, instead of looking at the entire leaf. They managed to expand the existing database because each region has its own characteristics. They created a dataset with images taken from typical cameras, smartphones, or DSLR cameras. In the preprocessing step, they downsized images to 224 × 224 × 3. They achieved 12% higher accuracy than in the original image dataset. Four categories were predicted: healthy leaves with an accuracy of 89%, leaves with mild disease with an accuracy of 31%, leaves with moderate disease with an accuracy of 87%, and leaves with severe disease with an accuracy of 94%.

Coulibaly et al. [43] developed a feature extraction-based methodology that was predicated on the CNN model VGG16, which has been pre-trained on ImageNet. They used 124 images of pearl millet from the ImageNet dataset, with mildew diseases and healthy plants. Additionally, a data augmentation technique was utilized to augment the dataset by increasing the number of images. In the preprocessing step, they downsized images to 150 × 150. Transfer learning was employed with feature extraction, resulting in an accuracy of 95%.

Dhingra et al. [44] presented methods for recognizing the diseases on the leaves of basil using digital image

processing techniques. They used nine different classifiers to classify the diseases. The researchers gathered images from an herb garden and standardized the surface condition by removing any non-uniform distribution of dust across the leaves, ensuring consistency across all leaf categories. The nine classifiers used were DT, SVM, linear models, Naives Bayes, KNN, AdaBoost, discriminant analysis, RF, and ANNs. The classification separated the images into two categories, diseased and healthy. The dataset includes 200 healthy and 200 diseased leaves. Out of the nine classifiers used, RF achieved the highest accuracy of 98.4%.

Hu et al. [45] presented a low-shot learning method to predict diseases in tea leaves. They created their image dataset by using UAVs. The tea disease spots in the images were separated using the SVM method. Conditional Deep Convolutional Generative Adversarial Networks (CDCGAN) [46] were used to supplement disease spot samples, while VGG16 DL networks are used to identify disease spots. They compared five different methods: SVM, DT, RF, and two different CNN methods, CDCGAN and VGG16, and they concluded that CNN achieved better results. The results obtained for the prediction of tea red scab, tea red leaf spot, and tea leaf blight, reached an average accuracy of 90% for the CNN methods.

The goal of Huang et al. [47] was to develop a decision support system for spraying machines that could identify *Helminthosporium* leaf spots in wheat fields using remote sensing data from a UAV. The dataset was divided into four categories based on disease severity: normal, light, medium, and heavy. In the preprocessing step, they extracted 100×100 samples from the images. A CNN was used for classification, and the proposed method achieved an accuracy of 91.43%.

In their study, Abdulridha et al. [48] aimed to detect two diseases in tomato crops using hyperspectral imaging captured by a UAV in the range of 380–1020nm. The tomato leaves were classified into four categories: healthy, asymptomatic, early, and late disease stages. They generated 35 spectral vegetation indices to identify the best indicators for disease detection and diagnosis and used MLP and Stepwise Discriminant Analysis (STDA) classification methods. They found that the blue band (408–420nm), red band (630–650 nm), and red edge (730–750 nm) were the most effective wavebands for disease detection. The spectral signatures of the two diseases were significantly different at all stages of disease progression, despite having similar symptoms in the early stages. Both laboratory and field tests showed satisfactory results for detecting both diseases in asymptomatic, early, and late stages. The MLP algorithm outperformed STDA, achieving an accuracy of 97% to 99% for all stages.

Abdulridha et al. [49] proposed nondestructive methods to detect diseases that affect tomato crops in two different datasets, a dataset created at the laboratory (benchtop scanning) and a dataset with images captured in the field

(using a UAV). These images consisted of images from healthy and diseased plants. The diseases that they targeted were: bacterial spot, target spot, and tomato yellow leaf curl. The diseased plants were identified and classified using several Vegetative Indices and the M statistic technique. They utilized two classification methods, STDA and RBF. For all diseases, both in the symptomatic and asymptomatic stages (i.e., before symptoms become obvious to direct visual observations), the classification results showed extremely accurate discrimination between healthy and diseased plants. In the field, the best classification results were achieved for tomato yellow leaf curl, bacterial spot, and target spot with accuracies of 100%, 98%, and 96%, respectively.

In their study, Abdulridha et al. [50] aimed to detect Powdery mildew in squash at various stages of disease development, including asymptomatic, early, intermediate, and late stages. They utilized hyperspectral imaging (380–1020nm) and machine learning (ML) algorithms to perform the detection, collecting data both in the lab and in the field using a UAV. The radial basis function (RBF) was used as the classification method, and the most significant bands for distinguishing between healthy and diseased stages were identified (388nm, 591nm, 646nm, 975nm, and 1012nm). The RBF method was able to diagnose Powdery mildew disease even in the asymptomatic phases, achieving a classification accuracy of 82% and 89% for the asymptomatic and early stages, respectively, in both laboratory and field settings. The best classification results were obtained in the very late stages of disease progression, with 96% and 99% accuracy in laboratory and field settings, respectively.

Agarwal et al. [51] introduced a CNN model with eight hidden layers that is lightweight and designed for classifying nine types of diseases in tomato crops. The model outperformed traditional ML techniques and pre-trained models with an accuracy of 98.4% on the publicly available PlantVillage dataset, which had 200–1400 images in different classes. The study used various assessment metrics such as accuracy, precision, recall, and F1-score to evaluate the performance of the proposed algorithm. The authors also employed image preprocessing and augmentation techniques to enhance the performance of the CNN. In the preprocessing step, they downsized images to 128×128 . Overall, the proposed model achieved a high accuracy of 98.4% on the PlantVillage dataset.

Anagnostis et al. [52] proposed a CNN model that can categorize images of walnut tree leaves based on whether or not they are infected with anthracnose. For this purpose, a set of grayscale and RGB images were used and Fourier transformation was applied to extract features. A CNN architecture was chosen on the basis of its performance. Feature extraction is significantly impacted by the Fourier transformation in grayscale images, as it emphasizes the abrupt changes and edges of the leaves. Infected leaves are more edgy in comparison to healthy ones. In the preprocessing step, they downsized images to

256×256 . The proposed CNN architecture has better or similar performance to widely-used CNN architectures such as ResNet50, VGG16, InceptionV3, and DenseNet121 [53], which are often used as benchmarks in image classification challenges. The accuracy achieved by the proposed CNN architecture was 99%.

Chen et al. [54] compared Lasso, RF, gradient boosting, and generalized linear models in predicting the possibility of a high rate of occurrence and severity of grape downy mildew using a dataset of nine years of observations. The algorithms evaluated in this study use the date of initial infection, as well as average monthly temperatures and precipitation as input parameters. Among these algorithms, Lasso, RF, and gradient boosting perform better than generalized linear models. Precipitation is found to have a larger impact on accuracy than temperature among weather inputs, while the date of disease onset has a greater impact on accuracy than weather inputs. The best-performing algorithm is selected based on results for high incidence and severity on leaves and bunches and is used to assess the influence of various climatic scenarios on grape downy mildew risk levels. The study found that reduced rainfall and higher temperatures in April-May decrease the probability of grape downy mildew during bunch closure. The highest area under the ROC curve achieved in this study was 86%.

An image processing technique for identifying plant diseases was proposed by Cristin et al. [55] that is efficient. The input image undergoes preprocessing to eliminate noise and artifacts. The preprocessed image is then segmented using a piecewise fuzzy C-means clustering method to obtain the segments. Texture features such as information gain, histogram of oriented gradients, and entropy are extracted from each segment during the feature extraction stage. A Deep Belief Network (DBN) is employed in the classification step, which was trained with the Rider-CSA algorithm, combining the Rider Optimization Algorithm (ROA) with the Cuckoo search method. The Rider-CSA-based DBN method proposed in the study outperformed previously used methods, with a maximum accuracy of 87%, sensitivity of 86%, and specificity of 87% in the experiments.

Lee et al. [56] used four CNN models, i.e., VGG16, InceptionV3, GoogLeNet, and GoogLeNetBN [57], for plant disease identification. They utilized three datasets, PlantVillage, IPM [7], and Bing [7]. They compared the performance trained from scratch and using transfer learning from existing pre-trained models. In the preprocessing step, they downsized and cropped images to different ratios according to the type of CNN. The VGG16 network performed better than the other CNN architectures, achieving an accuracy of 99% on seen crops and 65% on unseen crops.

Girardi et al. [58] presented a DL system with image processing techniques to detect fungal diseases in maize leaves. They used image classification techniques along with AlexNet, a deep CNN. Their model can detect three different classes of fungal disease, Northern blight, Southern Blight,

and Rust. They experimented with two different problems. The first problem is with three classes: healthy, common rust, and Northern blight images, and the second problem is with four classes: healthy, common rust, Northern blight, and multiple diseases in one image. For the first problem, three distinct disease detection models were generated, trained, and evaluated. In the preprocessing step, they downsized images to make them the same size. They used 300 images in the training set, 100 for each class. They also used 150 images in the testing set, 50 for each class. The training set for the second problem consists of 400 images, 100 for each class, while the testing set includes 200 images, 50 for each class. They obtained an accuracy of 98% on the three-class classification problem and an accuracy of 70.5% on the four-class classification problem.

A CNN-based approach for plant disease diagnosis and recognition is proposed by Guo et al. [59]. The model is designed to improve accuracy, generality, and training efficiency. The approach first uses the Region Proposal Network (RPN) [60] to detect and locate the leaves in a challenging environment. Then, the Chan-Vese (CV) method [61] is used to segment the images based on the results of the RPN algorithm, which captures the features of the symptoms. Finally, the segmented leaves are fed into the transfer learning model, which is trained using the PBPC [62] of diseased leaves in a simple environment. The approach is tested for black rot, bacterial plaque, and rust disease, achieving an accuracy of 83.75%, which is higher than conventional methods, and can help minimize the impact of diseases on agricultural production.

Habib et al. [63] introduced an agromedical expert system for papaya disease identification that relies on machine vision and analyzes images taken using portable devices. The proposed system focuses on two main objectives: disease detection and disease classification. The effectiveness of the system was tested through several experiments. Initially, they defined a set of features that can help differentiate between the different diseases. Then, they used the K-means clustering algorithm to segment the disease-affected area in the collected images and extracted the relevant features to classify the diseases using the support vector machine (SVM) algorithm. In the preprocessing step, they downsized images to 300×300 . The system achieved an accuracy of 90.15%.

Karadağ et al. [64] proposed a method for identifying fusarium disease on peppers utilizing a spectroradiometer from the reflections obtained from pepper leaves in a laboratory. Four sets of pepper leaves were grown in a confined setting to obtain reflections. They proposed a two-step disease detection framework. Subsequently, the extracted feature vector was utilized in the classification process, employing ANNs, Naive Bayes (NB), and KNN. The obtained accuracy for KNN was 100%, 97.5% for ANN, and 90% for NB.

Karlekar and Seal [65] proposed a computer vision method to identify and categorize leaf diseases present in soybean

crops. Initially, the proposed approach separates the leaf section by removing the complex background from the entire image. Following this, a deep learning Convolutional Neural Network (CNN) known as SoyNet, trained on a 16-class dataset called PDDDB [27], categorizes the segmented leaf images. In the preprocessing step, they downsized images to 100×100 . The proposed model attains an identification accuracy of 98.14%, with high precision, recall, and F1-score.

Karthik et al. [66] proposed two alternative deep neural network (DNN) models to identify the type of infection present in tomato leaves. The first model applies residual learning on top of a feed-forward CNN to learn crucial classification features, while the second model utilizes the strengths of the attention mechanism and residual learning on CNNs. To assess the performance of the proposed models, the PlantVillage dataset was used, which includes three diseases: early blight, late blight, and leaf mold. They reported an accuracy of 95% and 98% for the two architectures, respectively.

The detection of mildew disease in vineyards using a CNN on UAV images was proposed by Kerkech et al. [67]. Additionally, they introduced a new object recognition technique that aligns visible and infrared images, allowing data from the two sensors to be combined. A fully CNN technique is then used to classify each pixel into different categories, such as ground, shadow, symptom, and healthy. The proposed method was able to detect over 92% of infections at the grapevine level and 87% at the leaf level, indicating that automatic disease detection in vineyards has a promising future.

Khalili et al. [68] experimented with six different ML algorithms, namely MLP, Gradient Tree Boosting (GBT), L1 regularization (LR-L1), L2 regularization (LR-L2), SVM, and RF in order to forecast charcoal rot disease in soybean on a custom dataset [69] of 2,000 healthy and diseased plants. The importance of the features considered by various ML methods was also analyzed. They applied a 10-fold cross-validation to the training set and all ML models achieved an accuracy of more than 96.79%.

The use of an optimized deep neural network for identifying and classifying paddy leaf infections in rice harvests was proposed by Ramesh et al. [70]. The Jaya algorithm was utilized for this study. Images of rice plant leaves from farm fields with normal, sheath rot, blast diseases, brown spots, and bacterial blight were collected. In preprocessing, RGB images were converted to HSV images to eliminate the background, and then binary images based on the hue and saturation sections were retrieved to determine whether or not a disease occurred in a region. Several metrics, such as accuracy, precision, and F1-score, were used to evaluate the results, and the outcomes were compared to those of Denoising Auto-Encoder (DAE), DNN, and ANN classifiers. In the preprocessing step, they downsized images to 300×450 . The proposed method achieved high accuracy rates of 98.9% for blast-affected images, 95.7% for bacterial

blight, 94% for brown spot, 92% for sheath rot, and 90.57% for normal leaf images, outperforming the other classifiers.

Verma et al. [71] trained and compared the CNN models SqueezeNet [72], InceptionV3, and AlexNet, for determining the severity of late blight infection in tomatoes. For this purpose, they utilized the PlantVillage dataset for transfer learning and feature extraction. AlexNet outperformed the other two networks with an accuracy of 93.4%.

Velásquez et al. [73] proposed a technique to detect coffee leaf rust by utilizing data from various sources, including sensor data in JSON format, as well as RGB, RGN, and RE images from UAV multispectral cameras. The proposed approach involved the integration of remote sensing and DL algorithms, specifically MLP and CNN. The sub-models were trained separately, and the resulting F1-scores for each sub-model and the composite model were calculated. The composite model achieved an F1-score of 77.5%.

Yan et al. [74] presented an enhanced version of the VGG16 model to detect diseases in apple leaves. They utilized a dataset that consisted of 10 plant types with 27 different disease categories obtained from the “2008 ‘AI Challenger’ Global Challenge” [75], as well as additional images. The authors employed transfer learning to shorten the learning time. The results demonstrate that the proposed model achieved an overall accuracy of 99.01%.

Zhang et al. [76] presented a structured technique for wheat lodging detection in experimental plots that included aerial imagery collected from UAVs. For classifying wheat lodging, ML algorithms, and CNNs were utilized. For the ML algorithms, 320 extracted features were fed into RF, NN, and SVM. For the CNNs, they used VGGNet and GoogLeNet. GoogLeNet had significantly higher accuracy than VGGNet, about 93%, while SVM had an accuracy of 92%.

Abbas et al. [77] proposed a DL approach for detecting diseases on tomato plant leaves. They utilized the Conditional Generative Adversarial Network (CGAN) [78] in order to generate synthetic images. The classification of ten types of diseases on tomato leaves was carried out using the DenseNet121 model trained on both synthetic and real images through transfer learning. In the preprocessing step, they downsized images to 224×224 . The study employed the PlantVillage dataset, and the obtained accuracy was 97.11%.

An efficient automated diagnosis system for corn plants was proposed by Akanksha et al. [79]. The proposed method consists of four stages: (i) preprocessing, (ii) feature extraction, (iii) classification, and (vi) segmentation. The preprocessing techniques convert images to RGB and remove noise. They utilized an Optimized Probabilistic Neural Network (OPNN) that was improved by using the Artificial Jelly Optimization (AJO) algorithm for classification. The system achieved an accuracy of 95.5%.

The automatic detection of bacterial spot diseases in peach plants was proposed by Bedi and Gole [80] using a hybrid model that combines a Convolutional Autoencoder (CAE) network with a CNN. They used the CAE for dimensionality

reduction to reduce the number of training parameters, and the CNN was used for classification. The model was trained and evaluated on the PlantVillage dataset, achieving an accuracy of 99.35% on the training set and 98.38% on the test set with 9,914 training parameters.

According to the study conducted by Chowdhury et al. [81], tomato leaf images were classified for tomato diseases using the EfficientNet CNN architecture [82]. The PlantVillage dataset, containing 18,162 tomato images, was used to fine-tune and train the model to detect healthy and diseased tomato leaf images. In the preprocessing step, they downsized images to 224×224 . The results indicate that the proposed model outperformed various contemporary DL algorithms. Two segmentation models, the original U-net [83] and modified U-net [84], were also trained, validated, and tested for the segmentation of tomato leaf images. The Modified U-net was found to be superior for segmenting leaf images from the background, and EfficientNetB7 was superior at extracting discriminative features from images. The performance of the models generally improved with the number of parameters used. The trained models have potential applications in automatic detection of plant diseases at an early stage, and the study achieved an accuracy of 98.6% for the U-net model, 99.9% for the EfficientNetB7, and 99.8% for the EfficientNetB4.

Dwivedi et al. [85] presented a disease detection network for grapes. This network was based on Faster RCNN [60]. During the evaluation stage, the proposed disease detection mechanism was experimented on the PlantVillage dataset, and it was found that the network was more efficient in detecting infected/diseased regions than existing methods. The results showed an overall accuracy of 99.93% for identifying esca, black rot, and isariopsis diseases.

Mishra et al. [86] developed an automated plant disease detection model using median filter preprocessing techniques and a pixel-level feature extraction technique. In this technique, the statistical features are extracted from an image. Afterward, the researchers utilized the Sine Cosine Algorithm-based Rider Neural Network (SCA-RideNN) as the classifier for detecting plant diseases. They conducted experiments with the PlantVillage dataset and reported an accuracy of around 91.56%.

For guava disease detection, Mostafa et al. [87] utilized five neural network models, namely AlexNet, SqueezeNet, GoogLeNet, ResNet50, and ResNet101 [24]. They started by preprocessing the images using the color histogram and unsharp masking method, followed by augmentation using the affine transformation method. In the preprocessing step, they downsized images to be of the same size. The results indicated that ResNet101 achieved the highest classification accuracy of 97.74%.

The aim of Patil and Patil [88] was to develop a DL method to detect a diseased cotton plant from its leaf images. They used an IoT-based platform to collect various sensor data to detect climatic changes. Their deep CNN

architecture comprised convolutional layers, ReLU layers, fully connected layers, pooling layers, and activation layers. They used a custom dataset and applied image preprocessing, augmentation, and fine-tuning techniques. In the preprocessing step, they downsized images to 256×256 . The proposed method yielded an accuracy of about 98%.

The aim of Sambasivam and Opiyo [89] was to detect cassava infections using a CNN model trained on a dataset containing 10,000 annotated images from [90]. The dataset included five fine-grained cassava leaf disease categories; however, there was a significant class imbalance issue with the data. The authors employed several techniques such as class weight, focus loss, and SMOTE to counter the imbalance and improve the model's performance. After tuning several parameters, including the learning rate, number of epochs, batch size, input shape, optimizer, and number of neurons in the hidden layer, they found that a vector input shape of (448, 448, 3) yielded the best accuracy of 93%.

Sujatha et al. [91] assessed and compared the performance of various ML and DL algorithms for the detection of diseases in citrus leaf images as described in [92]. The algorithms evaluated included SVM, RF, stochastic gradient descent, InceptionV3, VGG16, and VGG19. The results showed that the VGG16 model achieved the highest classification accuracy of 89.5%, while the RF model had the lowest accuracy of 76.8%.

A CNN-based approach proposed by Tiwari et al. [93] was used for the detection and classification of plant diseases from leaf images captured in different resolutions. The study included six crops (apple, potato, tomato, bean, citrus, and rice) and 27 different disease categories under laboratory and on-field conditions. The authors also utilized five-fold cross-validation and tested the model on unseen data to enhance the accuracy of the results. Images were gathered from several databases, such as PlantVillage, iBean leaf image dataset [94], citrus leaf images, and rice leaf images [95]. In the preprocessing step, they downsized images to 224×224 . The cross-validation accuracy was found to be 99.58%, and the average test accuracy was 99.12% for images with complex environmental situations.

The use of a double GAN for balancing datasets by generating images of diseased plant leaves was proposed by Zhao et al. [96]. Initially, they used healthy leaf images from the PlantVillage dataset as inputs for the Wasserstein GAN to obtain a pre-trained model. Subsequently, diseased leaves were used to generate 64×64 pixel images of diseased leaves using the pre-trained model. To expand the unbalanced dataset, a superresolution GAN was used to obtain 256×256 pixel images. The authors observed that the generated images from the double GAN were clearer than those from DCGAN. They achieved an accuracy of 99.80% and 99.53% for the double GAN and DCGAN, respectively.

Kathole and Munot [97] evaluated and compared the performance of various CNN models, including VGG architecture from scratch, VGG architecture with pre-trained

weights, GoogLeNet, and AlexNet, for detecting plant diseases in tomato crops. The experiments were conducted on six different disease classes and a healthy class, and the data were obtained from PlantVillage. The GoogLeNet model achieved the highest accuracy of 97.94%.

Pan et al. [98] employed multiple CNN models including AlexNet, GoogleNet, VGG16, and VGG19 to detect Northern corn leaf blight in maize crops. The researchers utilized a dataset consisting of 985 images of healthy and infected maize leaves and applied various data augmentation techniques to augment their dataset. Their results showed that the GoogleNet model achieved the best performance, with an accuracy of 99.94% in detecting Northern corn leaf blight.

The ResTS (Residual Teacher/Student) architecture, developed by Shah et al. [99], is a CNN structure that incorporates two classifiers (ResTeacher and ResStudent) and a decoder to aid in the diagnosis of plant diseases through visualization and classification. They utilized residual connections and performed batch normalization on all subsystems. The PlantVillage dataset was used to train their model, and they achieved an F1-score of 99.1%.

Two deep feature extraction-based classification models were proposed by Turkoglu et al. [100] using transfer learning on six pre-trained DL networks (AlexNet, DenseNet201, GoogleNet, ResNet18 [24], ResNet50, and ResNet101) for classifying plant disease images. They also proposed a hybrid CNN-SVM model that combined the early fusion of features extracted from deep networks. The images used for training and testing were from the Turkey-PlantDataset [101], which consists of unconstrained images of 15 types of diseases and pests observed in Turkey. The highest accuracy achieved by all the models was 97.56

The automatic plant disease detection technique proposed by Vallabhajosyula et al. [102] utilizes Deep Ensemble Neural Networks (DENN) and employs data augmentation techniques, including image enhancement, rotation, scaling, and translation. The researchers used the PlantVillage dataset and concluded that DENN's performance surpasses that of pre-trained state-of-the-art models, such as ResNet50, ResNet101, InceptionV3, DenseNet 121 and 201, MobileNetV3 [103], and NasNet [104]. The DENN model achieved an accuracy of 99.99

B. OBJECT DETECTION

Fuentes et al. [105] combined three object detection algorithms (Faster RCNN, R-FCN [106], and SSD [107]) with five feature extractors (VGG16, ResNet50, ResNet101, ResNet152 [24], ResNeXt-50) for recognizing tomato plant diseases and pests. They applied the CNN algorithms on a custom dataset of about 5,000 images using various methods to augment the dataset size. The results show that R-FCN with ResNet50 achieved the best mAP of 86%.

Jiang et al. [108] used a DL approach based on the GoogLeNet Inception structure and the rainbow concatenation for apple leaf disease detection. They generated a custom

dataset of 2,029 images of diseased apple leaves and trained their algorithm to detect five common apple leaf diseases: Alternaria leaf spot, brown spot, mosaic, grey spot, and rust. Various augmentation techniques have been applied in order to generate a more representative dataset. They achieved a detection performance of 78.80% mAP.

Kim et al. [109] proposed a system for field monitoring with a PTZ camera, a motor system, a wireless transceiver, and an image-logging module. This system periodically captures images from the field and sends them to a CNN model designed to detect onion-downy mildew diseases. The model was trained on a dataset of six classes of images captured by the field monitoring system. In the preprocessing step, they extracted 224×224 samples from images. The proposed model achieved an accuracy ranging from 74.1% to 87.2%.

Li et al. [110] presented a technique for disease detection on rice crops using a CNN. They gathered images and videos from mobile phones. They converted videos to a still frame, then transmitted it to a still-image detector for recognition, and then combined the frames into a video. They utilized Faster RCNN as the framework for the still-image detector. The results revealed that the proposed method with the custom backbone was better than ResNet50, ResNet101, YOLOv3 [111], and VGG16 for recognizing diseases and pests for rice crops.

Saleem et al. [112] combined three object detection algorithms (Faster RCNN, R-FCN, and SSD) with four feature extractors (ResNet50, ResNet101, InceptionV2, and Inception-ResNetV2 [113]) and three optimizers (stochastic gradient descent with momentum, Adam, and RMSProp) for plant disease identification. They used the training weights of the Common Objects in Context (COCO) dataset [114]. They compared the algorithms on the PlantVillage dataset. The SSD model with the feature extractor InceptionV2 trained with the Adam optimizer was the best performer with a mAP of 73.07%.

The detection of Northern maize leaf blight disease on maize crops under complex field environments was proposed by Sun et al. [115] using multi-scale feature fusion and the SSD algorithm. For this purpose, a custom dataset consisting of 18,000 images captured by a camera mounted either on a five-meter boom or on a UAV was used. The proposed technique consists of three steps: preprocessing the dataset, fine-tuning the model, and detecting the disease. The NLB dataset was selected because it is calibrated by human plant pathologists and has a high level of accuracy. Data preparation was done mainly to reduce the effect of high-intensity light on image identification and enhance detection accuracy. The proposed model achieved a total accuracy of 91.83%.

Xie et al. [116] presented a DL-based detector for identifying leaf diseases in grape crops. In order to improve the generalization of the model, they developed the grape leaf disease dataset, which included 4,449 original images captured both in the lab and in actual vineyards, and

expanded the dataset with an additional 62,286 diseased leaf images. The authors trained the model using data augmentation techniques and utilized the InceptionV1 module [117], Inception-ResNetV2 module, and SE-blocks [118] to enhance the detection performance of multi-scale and smaller diseased spots. The model achieved an overall accuracy of 81.1%.

The detection of tomato diseases and healthy leaves was the objective of the study by Zhang et al. [119]. For this purpose, they proposed an improved version of Faster RCNN, a popular object detection algorithm, which utilized a depth residual network instead of VGG16 for image feature extraction. They also employed the K-means clustering algorithm to cluster the bounding boxes obtained from the network. Three feature extraction networks were tested in the experiments, namely VGG16, Mobile, and Res101. The proposed model achieved an mAP of 98.54% when trained with the Res101 network and k-means clustering.

Roy and Bhaduri [120] proposed a multi-class plant disease DL model using techniques from the field of object detection. The model was designed to achieve a balance between detection speed and accuracy, and it has been used to identify many classes of apple plant diseases in a real-world setting. The detection rate of 56.9 frames per second was achieved, while improving the mAP and F1-score up to 91.2% and 95.9%, respectively.

Selvaraj et al. [121] proposed a pixel-based classification method coupled with ML models for identifying banana crops in complex African environments using multi-level satellite images and UAVs. They used RF for pixel-based classification, combining vegetation indices and PCA, and developed a mixed-model approach using RetinaNet and a customized classifier for simultaneous banana localization and disease classification with UAV-RGB aerial images. The proposed method achieved high accuracy of 99.4%, 92.8%, 93.3%, and 90.8% for banana bunchy top disease, Xanthomonas wilt of banana, healthy banana cluster, and individual banana plants, respectively, according to their mixed-model results.

The PlantVillage dataset was used by Wang et al. [122] to propose three novel plant disease detection techniques, namely Squeeze-and-Excitation SSD (SE_SSD), Deep Block SSD (DB_SSD), and Deep Block Attention (DBA_SSD). The authors compared the performance of their proposed algorithms with YOLOv3, YOLOv4 [123], YOLOv4 tiny, and Faster RCNN. The mAP obtained for the proposed techniques were 90.77% for SE_SSD, 89.93% for DB_SSD, and 92.20% for DBA_SSD.

An enhanced model for plant disease recognition based on the YOLOv5 network model was proposed by Chen et al. [124]. They made changes to the neck module, the SE module, and the loss function. They collected 2,375 images of rubber tree diseases. Their dataset consists of 1,203 powdery mildew images and 1,172 anthracnose images. In the preprocessing step, they downsized images

to 640×640 . The improved YOLOv5 network model achieved an accuracy of 70%.

The YOLOv4 algorithm was enhanced for detecting diseases in tomato plants by Roy et al. [125]. The researchers utilized DenseNet as the backbone of the modified algorithm and added two new residual blocks to the backbone. They evaluated the proposed model using 1,200 images from the PlantVillage dataset and compared its performance with other object detection methods. The results showed that the proposed model outperformed other state-of-the-art methods with a mAP of 96.29%.

The proposed method by Wang et al. [126] utilizes a lightweight version of the YOLOv5 model and includes an improved attention sub-module to enhance both accuracy and efficiency. Additionally, they employed the Ghostnet structure to reduce computations and replaced the original FPN/PANet structure with the BiFPN structure. The algorithm was evaluated on a custom dataset for plant disease detection, and its performance was compared to other models. The results indicate that the proposed model achieved an accuracy of 92.57%.

III. CLASSIFICATION SCHEME

In this section, we propose a classification scheme and categorize all relevant works in the associated classes. In addition, we present statistics of the most common crops/data types/methods/algorithms/datasets/metrics used by the reviewed papers. The classes that we use in our classification scheme are the following:

- *Crop*: the type of the crop that each study uses as a case study
- *Input data*: the type of the input data used in the ML algorithms
- *Dataset*: the name of the dataset that is used if this information is available in the paper
- *Models/Algorithms*: the ML algorithm that was used
- *Method*: whether classification (C) or object detection is targeted (C)
- *Metrics*: the evaluation metrics employed to measure the performance of the ML algorithms
- *Results*: a brief description of the results obtained by the ML algorithms

Table 7 in Appendix presents all the related work using the above classification scheme. Figures 2–7 show the percentage of papers in the different classes. Most papers have focused on disease detection in tomato plants. There are also various works that consider multiple crops. Corn, apple, grape, and rice are crops that have also attracted much attention in the literature. Regarding the input data type, it is obvious that most researchers use RGB images as input to their models/algorithms since most datasets include only such images. Regarding the datasets used in the studies, the majority of studies use a custom dataset, showing that there is a need to have publicly available datasets that can be utilized for benchmarking purposes. The PlantVillage dataset is also widely used.

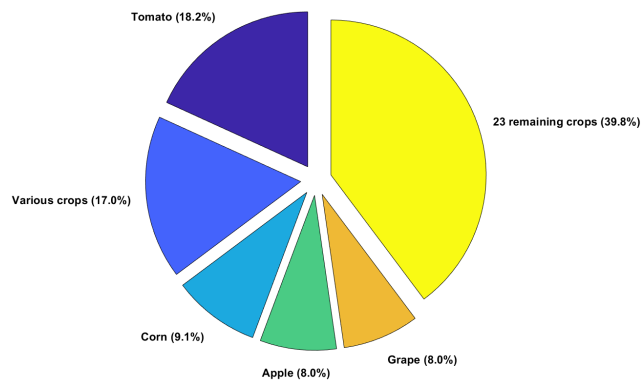


FIGURE 2. Statistics of crops.

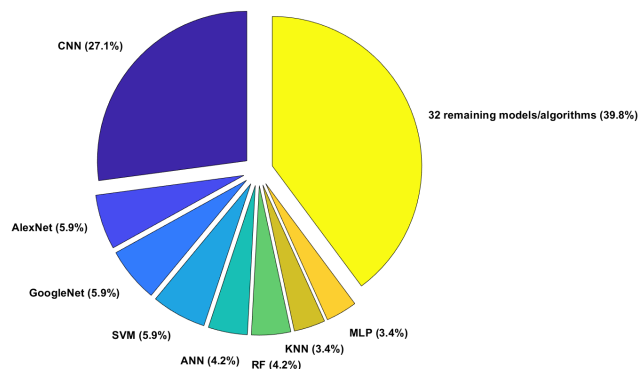


FIGURE 5. Statistics of models/algorithms.

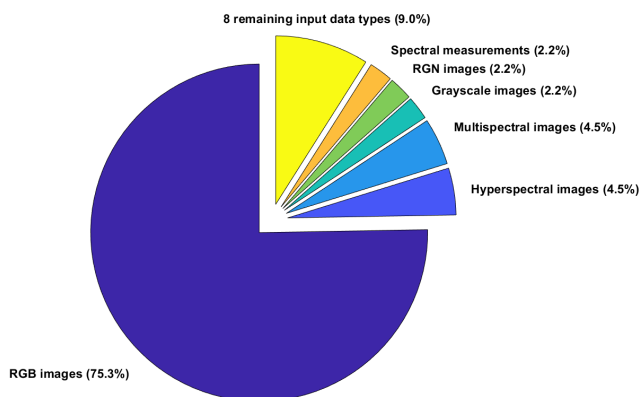


FIGURE 3. Statistics of input data types.

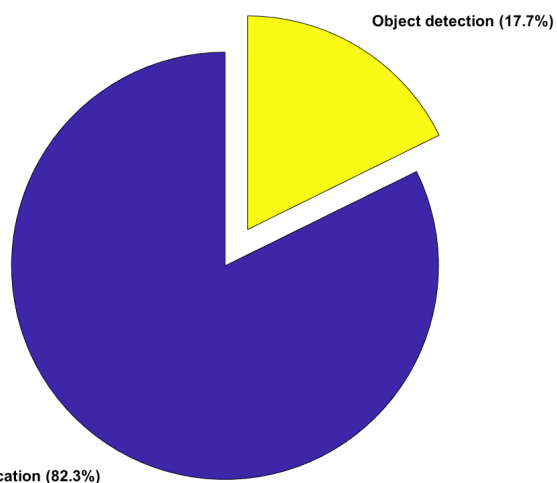


FIGURE 6. Statistics of methods.

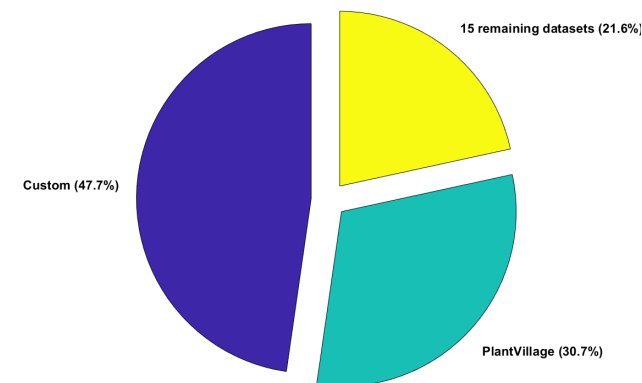


FIGURE 4. Statistics of datasets.

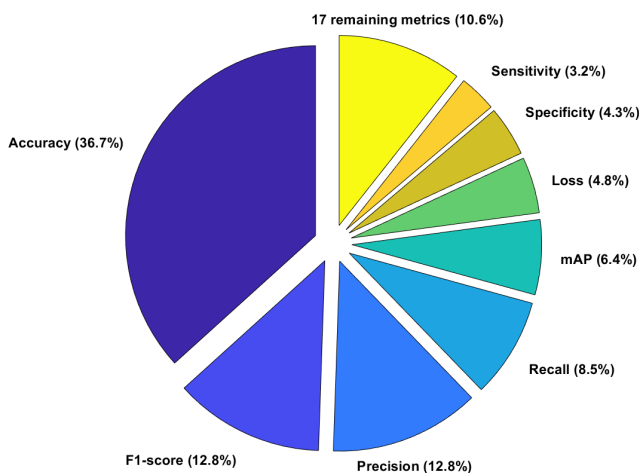


FIGURE 7. Statistics of metrics.

CNN is the most common neural network architecture used throughout the studies. There are also some works dealing with classifying diseases that use SVM, ANN, RF, KNN, and MLP algorithms. Regarding the method that each study targets, classification is the clear winner compared to object detection. Different metrics have been employed in the literature, with accuracy, F1-score, precision, and recall being among the most commonly used ones.

IV. DATASETS

This section provides an overview of publicly available datasets. These datasets serve different purposes; some of

them are used for classification to determine if a plant image is healthy or infected with a disease (discussed in Subsection IV-A), while others are used for object detection to identify diseases on plants (discussed in Subsection IV-B).

A. CLASSIFICATION

The PlantVillage dataset [10] comprises 54,303 leaf images, both healthy and diseased, categorized into 38 classes based on species and diseases. The dataset includes images of 14 crop species, such as apple, blueberry, cherry, corn, grape, orange, peach, bell pepper, potato, raspberry, soybean, squash, strawberry, and tomato. It covers 17 fungal-related diseases, four bacterial diseases, two mold (oomycete) diseases, two viral diseases, and one mite-related disease. Additionally, it provides images of disease-free healthy leaves from 12 crop species.

The iBean leaf image dataset [94] comprises images of bean leaves captured from field conditions. The National Crops Resources Research Institute (NaCRRI), the national organization in charge of research in agriculture in Uganda, and the Makerere AI lab collaborated to capture these images in several areas of Uganda. Images were captured from the field or garden using a simple smartphone, which were then analyzed by NaCRRI experts who determined which illness was present in each image. The dataset consists of 1,296 images and three classes. 428 images are for the healthy class, 432 images are for the angular leaf spot, and the remaining 436 for the bean rust.

The PlantLeaves dataset [127] is comprised of 4,503 images of plant leaves, both healthy and diseased, categorized into 22 categories based on the species and the state of health. The dataset includes 2,278 healthy leaf images and 2,225 diseased ones. The images were captured using a basic digital camera.

The PlantaeK dataset [128] is a leaf database of indigenous plants found in Jammu and Kashmir. It comprises 2,153 images of healthy and diseased plant leaves, categorized into 16 groups by species and health status. The images feature various crop species, including apple, apricot, cherry, cranberry, grapes, peach, pear, and walnut. The dataset comprises 1,223 healthy leaf images and 934 diseased leaf images.

The Plant Pathology 2020 challenge dataset [129] is a classification dataset for the foliar disease of apples. The creators of the dataset manually captured 3,651 real-world symptoms of several apple foliar diseases with varied lighting, angles, surfaces, and noise. The dataset includes 865 healthy leaves, 187 cases of complex diseases, 1,200 cases of apple scab, and 1,399 cases of cedar apple rot.

The citrus leaf images dataset [92] contains images of healthy and infected citrus plants with diseases such as black spot, canker, scab, greening, and melanosis. The dataset includes 609 images from citrus leaves, of which 58 are healthy images, and 150 images from citrus fruits, of which 22 are healthy images.

The Kaggle dataset [90] contains 9,436 annotated images and 12,595 unlabeled images of cassava leaves. The dataset contains five classes, one is the class for healthy plants and the other four are for diseases (cmd, cgm, cbsd, and cbb). NaCRRI in collaboration with the AI lab at Makerere University captured and annotated these images.

The dataset of Rice leaf images [95] includes 120 images collected from a village in India. The dataset contains 40 images of each disease, for a total of 120 images. The NLB dataset [130] is comprised of 234 images of leaf spot disease in maize crops.

B. OBJECT DETECTION

The PlantDoc dataset [131] includes 2,345 images. These images contain 13 plant species and 18 classes of diseases. This dataset is publicly available for download and it can also be used as an open dataset for benchmarks. The classes of the PlantDoc dataset are the following: Cherry leaf, Peach leaf, Cherry leaf, Peach leaf, Corn leaf blight, Apple rust leaf, Potato leaf late blight, Strawberry leaf, Corn rust leaf, Tomato leaf late blight, Tomato mold leaf, Potato leaf early blight, Apple leaf, Tomato leaf yellow virus, Blueberry leaf, Tomato leaf mosaic virus, Raspberry leaf, Tomato leaf bacterial spot, Squash Powdery mildew leaf, Grape leaf, Corn Gray leaf spot, Tomato Early blight leaf, Apple Scab Leaf, Tomato Septoria leaf spot, Tomato leaf, Soybean leaf, Bell pepper leaf spot, Bell pepper leaf, grape leaf black rot, Potato leaf, and Tomato two-spotted spider mites leaf. PlantDoc contains 8,851 annotations with an average of 3.4 annotations per image. The average image size is 0.53 mp and the distribution of the image sizes starts from 0.01 mp to 24.00 mp. The median image ratio is 800×675 . The balance of the classes of PlantDoc is presented in Figure 8. The classes Blueberry leaf, Tomato leaf yellow virus, and Peach leaf are overrepresented with more than 600 images for each class and the classes Tomato leaf late blight, Tomato Early blight leaf, Apple rust leaf, Apple Scab Leaf, grape leaf black rot, Corn rust leaf, Corn Gray leaf spot, Soybean leaf, Potato leaf, and Tomato two-spotted spider mites leaf are under-represented with less than 220 images for each class. A sample of four images with their annotations are shown in Figure 9.

The CropDeep dataset [132] is composed of 31,147 images containing over 49,000 annotated instances from 31 different classes. The images were captured in greenhouses under various conditions using different cameras. Additionally, the IP102 dataset [133] is a comprehensive benchmark dataset for recognizing insect pests. It comprises over 10,000 images divided into 102 categories, with insect pests that mainly target one agricultural product grouped together in the same top-level category. The IP102 dataset has a hierarchical taxonomy.

Table 1 presents the statistics for each dataset including: (i) the name of the dataset, (ii) the type of the crop, (iii) the number of images, (iv) the number of classes, and (v) whether classification (C) or object detection is targeted (C).

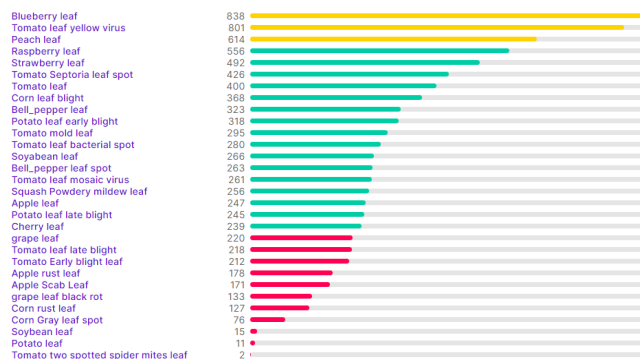


FIGURE 8. Class balance of the PlantDoc dataset.

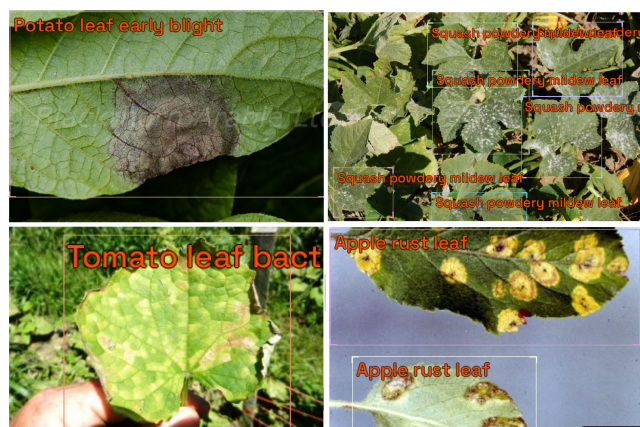


FIGURE 9. Visualization of the PlantDoc dataset with image annotations.

TABLE 1. Datasets statistics.

Dataset	Crop	Images	Classes	Method
PlantVillage	Various crops	54,303	38	C
iBean	Bean	1,296	5	C
PlantLeaves	Various crops	4,502	22	C
PlantaeK	Various crops	2,153	16	C
Plant Pathology	Apples	3,651	4	C
Citrus leaf images	Citrus fruits	759	6	C
Kaggle	Cassava	22,031	5	C
Rice leaf images	Rice	120	3	C
NLB	Maze	234	1	C
PlantDoc dataset	Various crops	2,345	18	O
CropDeep	Various crops	31,147	31	O
IP102	Corn	10,000	102	O

V. COMPUTATIONAL STUDY

In this computational study, we train five state-of-the-art object detection algorithms and eighteen state-of-the-art classification algorithms on the PlantDoc dataset. Before starting the training process, we applied a preprocessing step. At the PlantDoc dataset, there was a total of 28 annotations that needed to be fixed. The bounding boxes were out of frame in some images, thus they were trimmed to match the image’s edges. Some images had bounding zero pixels

and were afterwards dropped. The training set contained 25 of these, whereas the test set contained three. After that, we applied image resizing, auto-orientation, and image denoising.

All computational experiments were performed on an Intel Xeon Silver 4214 CPU processor with 128 GB of main memory and 48 cores, a clock of 2.2 GHz, running under Ubuntu 20.10 64-bit, and on a NVIDIA Tesla V100 GPU with 32 GB GDDR5 and a core clock of 1683 MHz.

For the training process, we utilized Fastai’s `learn.lr_find()` [134] method to find the optimal learning rate for each algorithm. The idea of this method comes from the Train Learner over a few iterations [135]. This method starts with a very low `start_lr` and changes it at each mini-batch until it reaches a very high `end_lr`. We record the loss at each iteration and check how it relates to the learning rate to determine the optimal value before the loss diverges.

For the object detection problem, we trained and evaluated the performance and the accuracy of the following algorithms:

- 1) *EfficientDet* [136] includes a collection of eight highly efficient and accurate algorithms developed by Google. EfficientDet detectors are single-shot detectors like SSD and RetinaNet. The EfficientNet, trained in the ImageNet [137] dataset, serves as the network’s backbone. The model utilizes a BiFPN as the feature network, while a separate class and box network is responsible for generating the class and bounding box predictors. On the COCO dataset, the overall mAP of EfficientDet-D7 is 52.2%.
- 2) *Faster RCNN* [60] belongs to the two-stage detector family because its object detection work occurs in two major stages. Faster RCNN is slower than single-stage detectors, but it is more accurate. On the COCO dataset, the overall mAP@.5 is 42.7%.
- 3) *RetinaNet* [138], a single-stage object detection model, is considered one of the top performers in detecting both dense and small objects. It has two main improvements over other single-stage models: FPN [139] and Focal Loss [138]. On the COCO dataset, RetinaNet’s mAP@.5 is 59.1%.
- 4) *SSD* [107] is based on the VGG16 architecture but lacks fully connected layers. For object detection, it employs a single deep neural network. The VGG architecture was employed as the foundation, and several layers were utilized for prediction at various scales. On the COCO dataset, the mAP@.5 is 46.5%.
- 5) *YOLOv5* [140] is a notable upgrade from YOLO [141], an object detection algorithm that partitions images into a grid system, with each grid cell responsible for detecting objects within it. YOLOv5 is a significant improvement over its predecessor. YOLOv5 uses the PANet structure that includes various layers. On the COCO dataset, the average accuracy is 43.5% AP.

Table 2 provides a summary of the hyperparameters used for the trained object detection algorithms in our study.

The hyperparameters include the learning rate (LR), batch size (BS), optimizer, and backbone architecture. The choice of hyperparameters for our trained object detection algorithms played a pivotal role in configuring and effectively training our models for leaf disease detection. Rather than relying on default values, we meticulously considered each hyperparameter based on its impact on model performance. For the EfficientDet algorithm, a learning rate (LR) of $5e-2$ was chosen with a batch size (BS) of 8. The Adam optimizer was utilized, and the EfficientNet served as the backbone architecture. This configuration likely leverages the efficient architecture and large learning rate to achieve rapid convergence. In the case of Faster RCNN, a lower learning rate of $2e-4$ was employed along with a batch size of 16. Stochastic Gradient Descent (SGD) [142] was selected as the optimizer, and ResNet-50 served as the backbone. The lower learning rate and larger batch size indicate a more stable and gradual training process. For RetinaNet, a smaller learning rate of $7e-5$ was used with a batch size of 8. Similar to EfficientDet, the Adam [143] optimizer was chosen, and ResNet-50 [24] was employed as the backbone architecture. This configuration may favor a balance between rapid convergence and fine-tuning. The SSD algorithm used a learning rate of $7e-5$ with a larger batch size of 32. The Adam optimizer was also selected, but the VGG16 [21] architecture was chosen as the backbone. This combination of parameters aims at accommodating a larger batch size while leveraging the SSD architecture. Lastly, YOLOv5 used a relatively high learning rate of $3e-3$ with a batch size of 32. Similar to the other models, Adam was used as the optimizer, and the new CSP-Darknet53 [144] architecture served as the backbone, a modification of the Darknet architecture used in previous versions [123]. This configuration may achieve rapid convergence and adapt to the YOLO architecture.

TABLE 2. Hyperparameters of the algorithms.

Algorithm	LR	BS	Optimizer	Backbone
EfficientDet	$5e-2$	8	Adam	EfficientNet
Faster RCNN	$2e-4$	16	SGD	ResNet-50
RetinaNet	$7e-5$	8	Adam	ResNet-50
SSD	$7e-5$	32	Adam	VGG16
YOLOv5	$3e-3$	32	Adam	CSPDarknet53

Table 3 presents the mean average precision (mAP) results of the leaf disease detection models after training.

To evaluate the overall accuracy of the object detector models, we utilized several COCO challenge metrics, namely (i) AP, (ii) $AP^{IoU=.50}$, (iii) $AP^{IoU=.75}$, (iv) AP small, (v) AP medium, and (vi) AP large. The assessment results for the COCO metrics are presented in Table 4.

Based on the results, it is evident that YOLOv5 exhibits superior accuracy compared to the other algorithms, as it outperforms them in terms of detecting objects of all sizes in the images. This superiority is reflected not only in the strict AP metric of the COCO challenge but also in other evaluation

TABLE 3. mAP of the trained leaf disease detection models.

Algorithm	mAP
EfficientDet	0.256552
Faster RCNN	0.276112
RetinaNet	0.285083
SSD	0.273251
YOLOv5	0.393168

TABLE 4. Average precision on COCO metrics.

Algorithm	AP	$AP^{IoU=.50}$	$AP^{IoU=.75}$	AP small	AP medium	AP large
EfficientDet	0.257	0.415	0.275	0.050	0.159	0.346
Faster RCNN	0.259	0.424	0.278	0.026	0.158	0.370
RetinaNet	0.285	0.431	0.310	0.050	0.180	0.372
SSD	0.273	0.467	0.288	0.031	0.152	0.355
YOLOv5	0.393	0.560	0.430	0.000	0.323	0.459

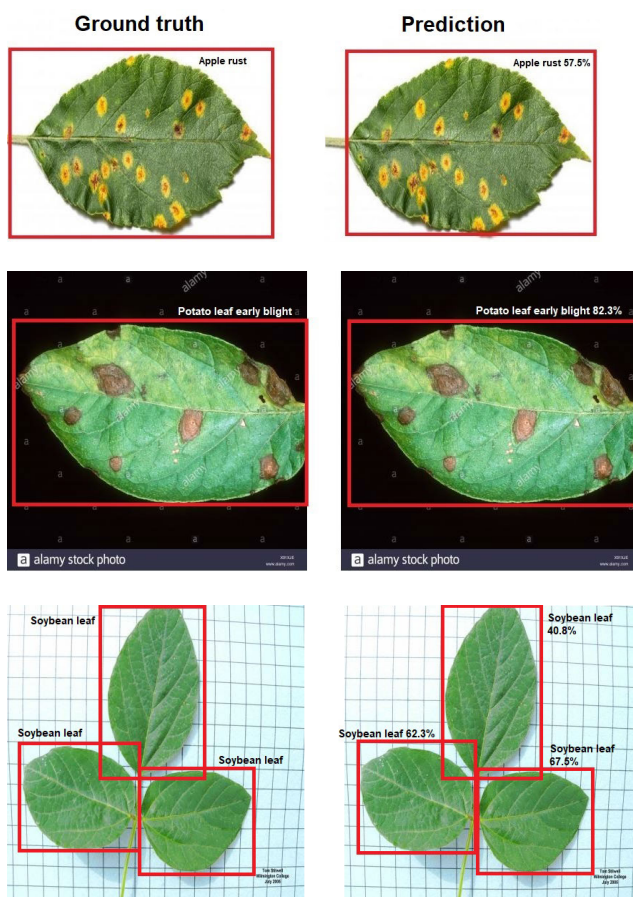


FIGURE 10. YOLOv5 ground truth vs prediction visualization.

metrics such as small, medium, and large object detection. EfficientDet, Faster RCNN, RetinaNet, and SSD have almost the same accuracy for this dataset. From the visualization of the results, it is shown that all algorithms were able to identify and localize most of the leaf species and diseases. However, there are images that show that the detection results are suboptimal, as shown in Figure 10.

TABLE 5. Hyperparameters for classification models.

Model	LR	BS	Optimizer
AlexNet	0.001	64	Adam
DenseNet121	0.001	32	SGD
DenseNet161	0.001	32	SGD
DenseNet169	0.001	32	SGD
DenseNet201	0.001	32	SGD
MobileNetV2	0.001	64	Adam
ResNet50	0.001	32	SGD
ResNet101	0.001	32	SGD
ResNet152	0.001	32	SGD
ResNeXt50_32x4d	0.001	32	SGD
ResNeXt101_32x8d	0.001	32	SGD
ShuffleNet_v2_x2_0	0.001	64	Adam
VGG16	0.001	32	SGD
VGG16_bn	0.001	32	SGD
VGG19	0.001	32	SGD
VGG19_bn	0.001	32	SGD
WideResNet50_2	0.001	32	SGD
WideResNet101_2	0.001	32	SGD

We trained and evaluated eighteen CNN model architectures that are currently considered state-of-the-art for the image classification problem. AlexNet, DenseNet (121, 161, 169, 201), MobileNetV2, ResNet (50, 101, 151), ResNeXt (50, 101) [145], ShuffleNet [146], VGG, and WideResNet (50, 101) [147]. Eighteen state-of-the-art CNN model architectures pre-trained on the ImageNet dataset were trained and evaluated for the image classification problem. The models were trained using the same hyperparameters for 200 epochs and were assessed based on their accuracy and training time. The Stochastic Gradient Descent optimizer and the softmax cross entropy loss function were utilized by all models.

The hyperparameters presented in Table 5 for the various classification models were selected after careful experimentation and tuning to achieve optimal model performance. These hyperparameters, including the learning rate (LR), batch size (BS), and optimizer, play a crucial role in determining the effectiveness of the training process and ultimately impact the model's accuracy and training times. The choice of a learning rate of 0.001 is a common starting point for many deep-learning models. It is often used as a default value and then fine-tuned during experimentation. The batch size of 32 or 64 is also a result of experimentation. Smaller batch sizes may provide more stable convergence, but larger batch sizes can lead to faster training times. As for the optimizer, models like AlexNet and MobileNetV2 benefit from the Adam optimizer, which adapts learning rates individually for each parameter. On the other hand, models like DenseNet and ResNet variants perform well with the Stochastic Gradient Descent (SGD) optimizer. The choice of optimizer depends on the model architecture and its convergence characteristics.

TABLE 6. Comparison of classification models in terms of accuracy and training time.

Model	Accuracy	Training time
AlexNet	0.444915	52m 25s
DenseNet121	0.610169	77m 9s
DenseNet161	0.601695	126m 2s
DenseNet169	0.576271	83m 29s
DenseNet201	0.559322	92m 52s
MobileNetV2	0.572034	67m 5s
ResNet50	0.559322	73m 26s
ResNet101	0.580508	89m 22s
ResNet152	0.601695	105m 50s
ResNeXt50_32x4d	0.576271	84m 40s
ResNeXt101_32x8d	0.593220	147m 20s
ShuffleNet_v2_x2_0	0.203390	64m 27s
VGG16	0.580508	96m 46s
VGG16_bn	0.567797	106m 54s
VGG19	0.525424	112m 1s
VGG19_bn	0.572034	123m 11s
WideResNet50_2	0.601695	148m 47s
WideResNet101_2	0.567797	122m 45s

Table 6 presents the results. The primary evaluation metric in this computational study is accuracy, which is the ratio of the number of correct predictions to the total number of predictions.

According to the study's results, the DenseNet121 model outperformed all other models by achieving the highest accuracy of 61.01%. DenseNet161, ResNet152, and WideResNet50_2 also performed well, achieving an accuracy of 60.16%. On the other hand, ShuffleNet_v2_x2_0 had the worst accuracy of 20.33% on the PlantDoc dataset.

The validation accuracy curve of all the models from epoch 0 to epoch 200 is depicted in Figure 11, whereas Figure 12 displays the training accuracy curve of all the models. All of the networks achieved a training accuracy of more than 97%, except for AlexNet which was not able to reach more than 85% of training accuracy. Finally, we compared all of the training times in Figure 13. MobileNetv2, ResNet50, and DenseNet121 had the shortest training times of 67m, 73m, and 77m, respectively, for an accuracy of more than 55%. The ideal balance between accuracy and training time was achieved from DenseNet121 by achieving the highest accuracy and one of the fastest training times. The reason for DenseNet's superior performance is that it requires fewer parameters than traditional CNNs, eliminating the need to learn redundant feature maps. Moreover, some variations of ResNet models have revealed that several layers do not contribute much and can be discarded. Finally, in the surveyed papers presented in the literature review, there are no algorithms that have been tested in the PlantDoc dataset. However, many studies use the PlantVillage dataset. The algorithms used in the PlantVillage dataset for the classification category, specifically for RGB images, achieve

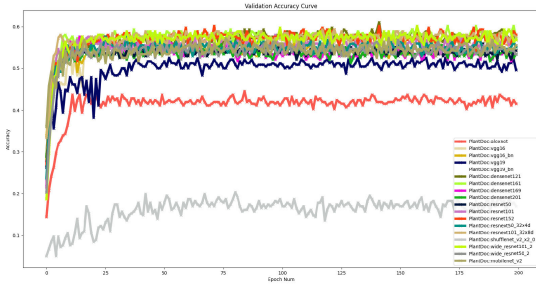


FIGURE 11. Comparison of the validation accuracy of the classification models.

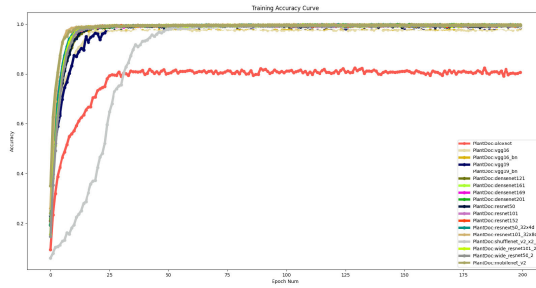


FIGURE 12. Comparison of the training accuracy of the classification models.

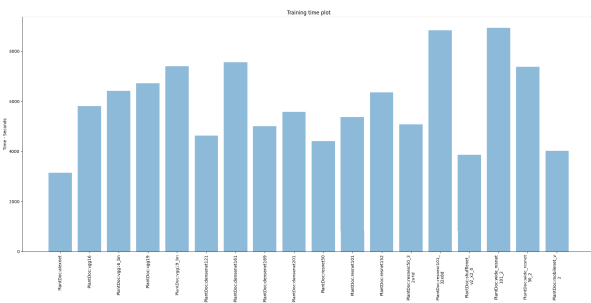


FIGURE 13. Total training time of each classification model.

a performance accuracy of up to 99% [7], [13], [15]. In contrast, the algorithms utilized in the PlantDoc dataset reach a maximum accuracy of 60% for the same category. Moving on to the object detection category, the algorithms employed in the PlantVillage dataset demonstrate an accuracy of up to 96% [112], [122], [125]. However, the algorithms utilized in the PlantDoc dataset achieve a lower accuracy of up to 56% for object detection. The size and diversity of the dataset can significantly impact algorithm performance. The PlantVillage dataset is larger and contains a more diverse set of images, thus it provides the algorithms with a broader range of examples to learn from. A larger dataset can help the algorithms generalize better to new, unseen data. Both in the classification method and in the object detection method we use the algorithms that are state-of-the-art and which are also used in other studies for different datasets.

VI. DISCUSSION

In the field of computer vision, the decision-making process for selecting appropriate methodologies often takes into

account the trade-offs between classification and object detection methods. The classification method serves as a foundational concept in machine learning, providing simplicity and interpretability. It allows for the straightforward categorization of inputs into predefined classes. Additionally, its computational efficiency makes it suitable for applications with limited resources. However, the classification method does have its limitations. It lacks fine-grained information, such as spatial details or the ability to handle multiple objects simultaneously. This makes it less suitable for tasks that require precise localization or the identification of multiple objects at once. On the other hand, object detection methods not only provide class labels but also offer precise spatial information, which addresses the challenge of localization. These methods excel in scenarios where multiple objects need to be handled and are versatile for tasks like instance segmentation and keypoint detection. Despite their advantages, object detection methods are more complex to implement. They require bounding box regression and labor-intensive data annotation. Moreover, they often demand significant computational resources, which restricts their deployment in real-time and resource-constrained environments. The choice between classification and object detection depends on the available data and the specific requirements of the task.

High-performing disease detection algorithms are pivotal in precision agriculture, aiding early disease identification, reducing crop losses, and efficient resource allocation. They precisely detect diseases like fungal infections and viral outbreaks, minimizing the need for extensive treatments. Their implementation may require specific hardware, e.g., GPUs, and robust computational resources, with infrastructure and cost considerations being crucial. Moreover, quality labeled datasets, capturing diverse disease manifestations and environmental conditions, are vital for accurate model training. Finally, the integration with existing precision agriculture systems, through standardized interfaces and APIs, and user-friendliness, with intuitive interfaces and interpretable outputs, empowers farmers and practitioners, fostering trust. In practical terms, several projects exemplify the application of these advanced algorithms in precision agriculture:

- PlantVillage [10] utilizes machine learning and computer vision for disease detection in crops. Its user-friendly mobile application enables farmers to take pictures of their crops and receive automated disease diagnoses. This project showcases the practical application of classification algorithms for real-time disease identification, benefiting small-scale farmers.
- John Deere’s See & Spray technology [148] exemplifies object detection in precision agriculture. It leverages computer vision and AI to identify and precisely target individual weeds, reducing the need for indiscriminate herbicide application. This project illustrates how object detection algorithms optimize resource usage.
- Blue River Technology [149] employs computer vision and machine learning for their See & Spray system,

autonomously identifying and selectively spraying herbicides only where needed. This project demonstrates the practical implications of object detection algorithms in enhancing the sustainability and cost-effectiveness of crop management.

- IBM's Smart Agriculture solution [150] leverages AI and machine learning to offer insights into crop health and disease detection. By integrating data from various sources, including satellite imagery and IoT sensors, it provides actionable recommendations to farmers. This project showcases the integration of advanced algorithms with a data-driven approach in precision agriculture.

VII. CHALLENGES IN PLANT DISEASE DETECTION

After a detailed review of ML and DL algorithms for plant disease detection and classification and the detailed computational study on five state-of-the-art object detection algorithms for plant disease detection and eighteen state-of-the-art classification algorithms for plant disease classification on a widely-used dataset, we have identified several challenges in practical applications of plant disease detection:

- 1) There is a lack of models that handle non-image data. Most existing classification and object detection algorithms focus solely on image data, neglecting other relevant information such as temperature and humidity. Developing techniques to incorporate non-image data is essential for more accurate predictions.
- 2) There are only a few completely annotated open datasets. Many studies rely on the PlantVillage dataset, which was obtained in a controlled laboratory setting. Generating larger datasets under real-world conditions is crucial. Collaborative efforts are needed to create representative datasets.
- 3) Most works treat the disease detection problem as a classification problem, either binary classification or multi-class classification. While many works treat disease detection as a classification problem, more emphasis should be placed on object detection to identify both the disease type and affected regions in the image.
- 4) Most papers use a single dataset used to train and test the model. Models trained on a single dataset often perform poorly on different datasets. It is essential to consider diverse datasets to improve model robustness.
- 5) Overreliance on CNN architectures: While CNNs yield good results, exploring other neural network architectures like recurrent neural networks can enhance disease detection methods.
- 6) Small leaf and early-stage disease recognition: Current datasets mainly consist of images with large leaves. Annotating datasets for early-stage disease detection and small leaf recognition is necessary.
- 7) Challenges with illumination and occlusion: Existing algorithms struggle with images under varying lighting

conditions and occlusion. More robust methods are needed to address these issues.

- 8) Computational efficiency: Many models are computationally intensive, hindering real-time applications. Researchers should focus on improving the computational efficiency of their models.

VIII. FUTURE DIRECTIONS IN PLANT DISEASE DETECTION

In addition to the challenges mentioned above, there are several promising directions for future research in plant disease detection:

- 1) Integration of non-image data: Develop models that can effectively integrate non-image data, such as environmental factors, into disease detection algorithms to improve prediction accuracy.
- 2) Creation of diverse and real-world datasets: Collaborate with experts to generate large, representative datasets under real-world agricultural conditions to enhance the generalizability of models.
- 3) Emphasis on object detection: Explore the potential of object detection methods for predicting plant diseases, which can provide more detailed information about disease localization.
- 4) Robustness across datasets: Develop models that perform consistently well across various datasets to ensure their practical utility.
- 5) Exploration of alternative neural network architectures: Experiment with different neural network architectures beyond CNNs, such as recurrent neural networks, to uncover their potential in disease detection.
- 6) Early-stage and small leaf recognition: Annotate datasets specifically for early-stage disease recognition and the identification of diseases on plants or leaves with small sizes.
- 7) Addressing illumination and occlusion challenges: Implement techniques to enhance the robustness of algorithms in the presence of variable lighting conditions and occluded images.
- 8) Improved computational efficiency: Focus on optimizing model architectures and algorithms to make them suitable for real-time applications.

IX. CONCLUSION

The aim of this study is to examine existing research that utilizes ML and DL techniques in precision agriculture, with a particular focus on plant disease detection and classification methods. Additionally, a new classification scheme is introduced, which categorizes all relevant works into their respective classes. We separate the studies into two main categories depending on the methodology that they use (i.e. classification and object detection). Furthermore, we present the available datasets for plant disease detection and classification, and provide details about their classes and data, and whether the specific dataset is suitable for classification or object detection.

TABLE 7. Classification of the related works.

Ref	Crop	Input Data	Dataset	Models/Algorithms	Method	Metrics	Results
[7]	Various crops	RGB images	PlantVillage	AlexNet, GoogleNet	C	Accuracy	Accuracy = 99%
[12]	Various crops	Various formats	Custom	CaffeNet	C	Accuracy	Accuracy = 96.3%
[13]	Banana	RGB images	PlantVillage	LeNet	C	Accuracy	Accuracy = 99%
[15]	Tomato	RGB images	PlantVillage	AlexNet, GoogleNet	C	Accuracy	Accuracy = 99%
[16]	Olive	RGB images	PlantVillage and Custom	LeNet	C	Accuracy, precision, F1-score, recall, Matthew's correlation coefficient	Accuracy = 99%
[17]	Corn	RGB images	Custom	CNN	C	Accuracy	Accuracy = 96.7%
[105]	Tomato	RGB images	Custom	CNN	O	Accuracy	Accuracy = 86%
[18]	Apple	RGB images	Custom	AlexNet	C	Accuracy	Accuracy = 97.62%
[19]	Rice	RGN images	Various sources	AlexNet	C	Accuracy, missing report rate, false report rate	Accuracy = 95.48%
[20]	Potato	RGB images	Custom	ResNet50	C	Accuracy	Accuracy = 96%
[22]	Apple	RGB images	PlantVillage	VGG16, VGG19, InceptionV3, ResNet50	C	Accuracy	Accuracy = 90.4%
[25]	Avocado	Hyperspectral images	Custom	MLP, DT	C	Accuracy	Accuracy = 82–100%
[26]	Corn	RGB images	PDDDB	CNN	C	Accuracy	Accuracy = 87%
[28]	Various crops	RGB images	PlantVillage	CNN	C	Accuracy, average error	Accuracy = 99.53%
[31]	Tomato	Spectral measurements	Custom	KNN	C	Accuracy	Accuracy = 73.5–100%
[32]	Tomato	RGB images	Custom	CNN	C	Accuracy, loss	Accuracy = 96%
[33]	Grape	RGB images	Custom	CNN	C	Accuracy	Accuracy = 95.8%
[34]	Melon	Thermography, Fluorescence	Custom	LRA, AVM, ANN	C	Accuracy	Accuracy = 99.1%
[35]	Citrus	RGB images	PlantVillage, Citrus Diseases database, Custom	SVM	C	Accuracy	Accuracy = 97%
[37]	Corn	RGB images	PlantVillage and other websites	GoogLeNet, Cifar10	C	Accuracy, loss	Accuracy = 98.9%
[39]	Avocado	RGB and multispectral images	Custom	MLP, KNN	C	Accuracy	Accuracy = 99%
[40]	Grape	Multispectral images	Custom	SVM, LDA	C	Accuracy	Accuracy = 96 %
[41]	Various crops	RGB images	Custom	CNN	C	Accuracy	Accuracy = 93.67%
[42]	Various crops	RGB images	Custom	GoogLeNet	C	Accuracy	Accuracy = 94%
[43]	Millet	RGB images	ImageNet	CNN	C	Accuracy, precision, recall, F1-score	Accuracy = 95%
[44]	Basil	RGB images	Custom	DT, RF, SVM, Ad-aBoost, GLM, ANN, NB, KNN, LDA	C	Accuracy, recall, precision, error rate	Accuracy = 98.4%
[45]	Tea	RGB images	Custom	GAN	C	Accuracy	Accuracy = 90%
[47]	Wheat	RGB images	Custom	CNN	C	Accuracy, Standard error	Accuracy = 91.43%

TABLE 7. (Continued.) Classification of the related works.

[108]	Apple	RGB images	Custom	CNN	O	Accuracy, mAP	mAP = 78.80%
[48]	Tomato	Hyperspectral images	Custom	MLP, STDA	C	Accuracy	Accuracy = 99%
[49]	Tomato	Hyperspectral images	Custom	STDA, RBF	C	Accuracy	Accuracy ≥ 96%
[50]	Squash	Hyperspectral images	Custom	RBF	C	Accuracy	Accuracy = 82%
[51]	Tomato	RGB images	PlantVillage	CNN	C	Accuracy, precision, recall, F1-score	Accuracy = 98.4%
[52]	Walnut	Grayscale and RGB images	Custom	CNN	C	Accuracy, precision, recall, F1-score	Accuracy = 99%
[54]	Grape	Time-series data	Custom	Lasso, RF, gradient boosting, generalized linear models	C	AUC	AUC = 86%
[55]	Various crops	RGB, grayscaled, and segmented images	PlantVillage	DBN	C	Accuracy, sensitivity, specificity	Accuracy = 87%
[110]	Rice	RGB images, video	Custom	Faster RCNN	O	Precision, recall, F1-score	Precision = 90.9%
[58]	Corn	RGB images	Custom	AlexNet	C	Accuracy	Accuracy = 98%
[59]	Various crops	RGB images	PPBC	CNN	C	Accuracy	Accuracy = 83.75%
[63]	Papaya	RGB images	Custom	K-means, SVM	C	Accuracy, sensitivity, specificity	Accuracy = 90.15%
[64]	Pepper	Spectral measurements	Custom	ANN, NB, KNN	C	Accuracy	Accuracy = 100%
[65]	Soybean	RGB images	PDDDB	CNN	C	Accuracy, precision, recall, F1-score	Accuracy = 98.14%
[66]	Tomato	RGB images	PlantVillage	CNN	C	Accuracy, loss	Accuracy = 98%
[67]	Grape	Multispectral and RGB images	Custom	CNN	C	Accuracy, precision, recall, F1-score, Dice, RMSE	Accuracy = 92%
[68]	Soybean	Multispectral images	Custom	MLP, GBT, LR-L1, LR-L2, SVM, and RF	C	Accuracy, sensitivity, specificity, precision, negative predictive value, F-measure, Matthews correlation coefficient, AUC	Accuracy = 96.79%
[109]	Onion	RGB images	Custom	CNN	O	mAP, loss	mAP = 87.2%
[56]	Various crops	RGB images	PlantVillage, IPM, Bing	VGG16, InceptionV3, GoogLeNet, GoogLeNetBN	C	Accuracy	Accuracy = 99%
[70]	Rice	RGB images	Custom	ANN with Jaya algorithm	C	Accuracy, Precision, F1-score	Accuracy ≥ 90%

TABLE 7. (Continued.) Classification of the related works.

[112]	Various crops	RGB images	PlantVillage	CNN	O	mAP	mAP = 73.07%
[121]	Banana	RGB images	Custom	RF	O	Accuracy, Precision, F1-score	Accuracy ≥ 90%
[71]	Tomato	RGB images	PlantVillage	CNN	C	Accuracy	Accuracy = 93.4%
[115]	Corn	RGB images	Corn northern leaf blight dataset	SSD	O	mAP	mAP = 91.83%
[73]	Coffee	Sensor data, RGB, RGN, and RE images	Custom	ANN	C	F1-score	F1-score = 77.5%
[116]	Grape	RGB images	Custom	CNN	O	Accuracy, mAP	Accuracy = 81.1%
[119]	Tomato	RGB images	AIChallenge 2018	Faster RCNN	O	mAP	mAP = 98.54%
[120]	Apple	RGB images	Kaggle PlantPathology Apple	CNN	O	mAP	mAP = 91.2%
[122]	Various crops	RGB images	PlantVillage	SSD	O	mAP	mAP = 92.20%
[74]	Apple	RGB images	2008 'AI Challenger' Global Challenge	CNN	C	Accuracy, precision, recall, F1-score	Accuracy = 99.01%
[76]	Wheat	RGB images	Custom	RF, NN, SVM, CNN	C	Accuracy, precision, recall, F1-score	Accuracy = 93%
[77]	Tomato	RGB images	PlantVillage	GAN	C	Accuracy, precision, recall, F1-score	Accuracy = 97.11%
[79]	Corn	RGB images	PlantVillage	Hybrid OPNN-AJO	C	Accuracy, sensitivity, specificity	Accuracy = 95.5%
[80]	Peach	RGB images	PlantVillage	Hybrid CAE-CNN	C	Accuracy, loss	Accuracy = 99.35%
[81]	Tomato	RGB images	PlantVillage	CNN	C	Accuracy, precision, sensitivity, F1-score, specificity, loss, IoU, dice	Accuracy = 99.9%
[85]	Grape	RGB images	PlantVillage	GLDNN	C	Accuracy, precision, recall, mAP	Accuracy = 99.93%
[86]	Various crops	RGB images	PlantVillage	SCA-RideNN	C	Accuracy, sensitivity, specificity	Accuracy = 91.56%
[87]	Guava	RGB images	Custom	AlexNet, SqueezeNet, GoogLeNet, ResNet50, ResNet101	C	Accuracy, specificity, F1-score, precision, Kappa-Cohen index	Accuracy = 97.74%
[88]	Cotton	RGB images	Custom	CNN	C	Accuracy, F1-score	Accuracy = 98%
[89]	Cassava	RGB images	Kaggle	CNN	C	Accuracy, precision, recall, F1-score, support, MSE	Accuracy = 93%
[91]	Citrus	RGB images	Citrus leaf images	CNN	C	Accuracy, precision, F1-score, AuC	Accuracy = 89.5%
[93]	Apple, potato, tomato, bean, citrus, rice	RGB images	PlantVillage, iBean, citrus leaf images, rice leaf images	CNN	C	Accuracy, specificity, recall, precision, F1-score	Accuracy = 99.58%
	Apple, corn,						

TABLE 7. (Continued.) Classification of the related works.

[96]	grape, potato, tomato	RGB images	PlantVillage	GAN	C	Accuracy	Accuracy = 99.80%
[124]	Rubber tree	RGB images	Custom	CNN	O	Accuracy, mAP	Accuracy = 70%
[97]	Tomato	RGB images	PlantVillage	CNN	C	Accuracy, loss	Accuracy = 97.94%
[98]	Corn	RGB images	Custom	AlexNet, GoogleNet, VGG16, VGG19	C	Accuracy, loss	Accuracy = 99.94%
[125]	Tomato	RGB images	PlantVillage	YOLOv4	O	Accuracy, precision, recall, F1-score, mAP	mAP = 96.29%
[99]	Various crops	RGB images	PlantVillage	CNN	C	F1-score, Training loss, Validation loss	F1-score = 91.1
[100]	Various crops	RGB images	Turkey-PlantDataset	Hybrid CNN-SVM	C	Accuracy, F1-score	Accuracy = 97.56%
[102]	Various crops	RGB images	PlantVillage	DENN	C	Accuracy, Precision, Recall, F1-score	Accuracy = 99.99%
[126]	Various crops	RGB images	Custom	YOLOv5	O	Accuracy, precision, F1-score, recall, mAP, loss	Accuracy = 92.57%

In addition, we perform an extensive computational study on five state-of-the-art object detection algorithms on the PlantDoc dataset to detect the diseases on the leaves, and eighteen state-of-the-art classification algorithms on the PlantDoc dataset, to predict whether or not there is a disease in a leaf and which one it is. For the object detection problem, YOLOv5 is able to achieve a high accuracy not only at the strict AP metric of the COCO challenge but also in the detection of small, medium, and large objects in the images. For the classification problem, the networks ResNet50 and MobileNetv2 had the most optimal trade-off on accuracy and training time. ResNet50 achieved a total accuracy of 61.01% and was trained on about 18 minutes, while MobileNetv2 achieved a total accuracy of 59.74% and was trained for about 16 minutes.

In future work, we plan to study more algorithms for classification and object detection on more datasets to see whether the results are consistent across different datasets. We also intend to study some image preprocessing and data augmentation techniques to see whether the accuracy of the algorithms can be improved with these techniques.

APPENDIX CLASSIFICATION OF THE RELATED WORKS

See Table 7.

REFERENCES

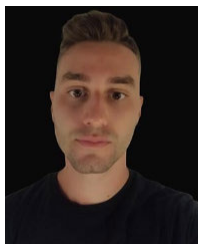
- [1] Food and Agriculture Organization. (2019). *New Standards to Curb the Global Spread of Plant Pests and Diseases*. Accessed: Nov. 8, 2022. [Online]. Available: <https://www.fao.org/news/story/en/item/1187738/icode/>
- [2] Y. Mekonnen, S. Namuduri, L. Burton, A. Sarwat, and S. Bhansali, "Machine learning techniques in wireless sensor network based precision agriculture," *J. Electrochem. Soc.*, vol. 167, no. 3, Jan. 2020, Art. no. 037522.
- [3] L. Benos, A. C. Tagarakis, G. Dolias, R. Berruto, D. Kateris, and D. Bochtis, "Machine learning in agriculture: A comprehensive updated review," *Sensors*, vol. 21, no. 11, p. 3758, May 2021.
- [4] L. Li, S. Zhang, and B. Wang, "Plant disease detection and classification by deep learning—A review," *IEEE Access*, vol. 9, pp. 56683–56698, 2021.
- [5] J. Liu and X. Wang, "Plant diseases and pests detection based on deep learning: A review," *Plant Methods*, vol. 17, no. 1, pp. 1–18, Dec. 2021.
- [6] Y. Yuan, L. Chen, H. Wu, and L. Li, "Advanced agricultural disease image recognition technologies: A review," *Inf. Process. Agricult.*, vol. 9, no. 1, pp. 48–59, Mar. 2022.
- [7] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers Plant Sci.*, vol. 7, p. 1419, Sep. 2016.
- [8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1097–1105.
- [9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
- [10] Kaggle. (2018). *Plantvillage Dataset*. Accessed: Nov. 8, 2022. [Online]. Available: <https://www.kaggle.com/datasets/emmarex/plantdisease>
- [11] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, Nov. 2014, pp. 675–678.
- [12] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," *Comput. Intell. Neurosci.*, vol. 2016, pp. 1–11, May 2016.
- [13] J. Amara, B. Bouaziz, and A. Algergawy, "A deep learning-based approach for banana leaf diseases classification," in *Proc. Datenbanksys. Für Bus., Technol. Web (BTW), Workshopband*, CA, USA, Jul. 2017, pp. 1–24.

- [14] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, Dec. 1989.
- [15] M. Brahim, K. Boukhalifa, and A. Moussaoui, "Deep learning for tomato diseases: Classification and symptoms visualization," *Appl. Artif. Intell.*, vol. 31, no. 4, pp. 299–315, Apr. 2017.
- [16] A. C. Cruz, A. Luvisi, L. De Bellis, and Y. Ampatzidis, "X-FIDO: An effective application for detecting olive quick decline syndrome with deep learning and data fusion," *Frontiers Plant Sci.*, vol. 8, p. 1741, Oct. 2017.
- [17] C. DeChant, T. Wiesner-Hanks, S. Chen, E. L. Stewart, J. Yosinski, M. A. Gore, R. J. Nelson, and H. Lipson, "Automated identification of northern leaf blight-infected maize plants from field imagery using deep learning," *Phytopathology*, vol. 107, no. 11, pp. 1426–1432, Nov. 2017.
- [18] B. Liu, Y. Zhang, D. He, and Y. Li, "Identification of apple leaf diseases based on deep convolutional neural networks," *Symmetry*, vol. 10, no. 1, p. 11, Dec. 2017.
- [19] Y. Lu, S. Yi, N. Zeng, Y. Liu, and Y. Zhang, "Identification of rice diseases using deep convolutional neural networks," *Neurocomputing*, vol. 267, pp. 378–384, Dec. 2017.
- [20] D. Oppenheim and G. Shani, "Potato disease classification using convolution neural networks," *Adv. Animal Biosci.*, vol. 8, no. 2, pp. 244–249, 2017.
- [21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
- [22] G. Wang, Y. Sun, and J. Wang, "Automatic image-based plant disease severity estimation using deep learning," *Comput. Intell. Neurosci.*, vol. 2017, pp. 1–8, Jul. 2017.
- [23] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.
- [24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
- [25] J. Abdulridha, Y. Ampatzidis, R. Ehsani, and A. I. de Castro, "Evaluating the performance of spectral features and multivariate analysis tools to detect laurel wilt disease and nutritional deficiency in avocado," *Comput. Electron. Agricult.*, vol. 155, pp. 203–211, Dec. 2018.
- [26] J. G. A. Barbedo, "Factors influencing the use of deep learning for plant disease recognition," *Biosyst. Eng.*, vol. 172, pp. 84–91, Aug. 2018.
- [27] J. G. A. Barbedo, L. V. Koenigkan, B. A. Halfeld-Vieira, R. V. Costa, K. L. Nechet, C. V. Godoy, M. L. Junior, F. R. A. Patricio, V. Talamini, L. G. Chitarra, S. A. S. Oliveira, A. K. N. Ishida, J. M. C. Fernandes, T. T. Santos, F. R. Cavalcanti, D. Terao, and F. Angelotti, "Annotated plant pathology databases for image-based detection and recognition of diseases," *IEEE Latin Amer. Trans.*, vol. 16, no. 6, pp. 1749–1757, Jun. 2018.
- [28] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Comput. Electron. Agricult.*, vol. 145, pp. 311–318, Feb. 2018.
- [29] A. Krizhevsky, "One weird trick for parallelizing convolutional neural networks," 2014, *arXiv:1404.5997*.
- [30] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated recognition, localization and detection using convolutional networks," 2013, *arXiv:1312.6229*.
- [31] J. Lu, R. Ehsani, Y. Shi, A. I. de Castro, and S. Wang, "Detection of multi-tomato leaf diseases (late blight, target and bacterial spots) in different stages by using a spectral-based sensor," *Sci. Rep.*, vol. 8, no. 1, pp. 1–11, Feb. 2018.
- [32] A. F. Fuentes, S. Yoon, J. Lee, and D. S. Park, "High-performance deep neural network-based tomato plant diseases and pests diagnosis system with refinement filter bank," *Frontiers Plant Sci.*, vol. 9, p. 1162, Aug. 2018.
- [33] M. Kerkech, A. Hafiane, and R. Canals, "Deep leaning approach with colorimetric spaces and vegetation indices for vine diseases detection in UAV images," *Comput. Electron. Agricult.*, vol. 155, pp. 237–243, Dec. 2018.
- [34] M. Pineda, M. L. Pérez-Bueno, and M. Barón, "Detection of bacterial infection in melon plants by classification methods based on imaging data," *Frontiers Plant Sci.*, vol. 9, p. 164, Feb. 2018.
- [35] M. Sharif, M. A. Khan, Z. Iqbal, M. F. Azam, M. I. U. Lali, and M. Y. Javed, "Detection and classification of citrus diseases in agriculture based on optimized weighted segmentation and feature selection," *Comput. Electron. Agricult.*, vol. 150, pp. 220–234, Jul. 2018.
- [36] USDA/APHIS/PPQ Center for Plant Health Science and Technology. (2010). *Citrus Diseases Database*. Accessed: Nov. 8, 2022. [Online]. Available: <http://idtools.org/id/citrus/diseases/gallery.php>
- [37] X. Zhang, Y. Qiao, F. Meng, C. Fan, and M. Zhang, "Identification of maize leaf diseases using improved deep convolutional neural networks," *IEEE Access*, vol. 6, pp. 30370–30377, 2018.
- [38] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," M.S. thesis, Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada, 2009.
- [39] J. Abdulridha, R. Ehsani, A. Abd-Elrahman, and Y. Ampatzidis, "A remote sensing technique for detecting laurel wilt disease in avocado in presence of other biotic and abiotic stresses," *Comput. Electron. Agricult.*, vol. 156, pp. 549–557, Jan. 2019.
- [40] H. Al-Saddik, J. C. Simon, and F. Cointault, "Assessment of the optimal spectral bands for designing a sensor for vineyard disease detection: The case of 'Flavescence dorée,'" *Precis. Agricult.*, vol. 20, no. 2, pp. 398–422, Apr. 2019.
- [41] M. Arsenovic, M. Karanovic, S. Sladojevic, A. Anderla, and D. Stefanovic, "Solving current limitations of deep learning based approaches for plant disease detection," *Symmetry*, vol. 11, no. 7, p. 939, Jul. 2019.
- [42] J. G. A. Barbedo, "Plant disease identification from individual lesions and spots using deep learning," *Biosyst. Eng.*, vol. 180, pp. 96–107, Apr. 2019.
- [43] S. Coulibaly, B. Kamsu-Foguem, D. Kamissoko, and D. Traore, "Deep neural networks with transfer learning in millet crop images," *Comput. Ind.*, vol. 108, pp. 115–120, Jun. 2019.
- [44] G. Dhingra, V. Kumar, and H. D. Joshi, "A novel computer vision based neurotrophic approach for leaf disease identification and classification," *Measurement*, vol. 135, pp. 782–794, Mar. 2019.
- [45] G. Hu, H. Wu, Y. Zhang, and M. Wan, "A low shot learning method for tea leaf's disease identification," *Comput. Electron. Agricult.*, vol. 163, Aug. 2019, Art. no. 104852.
- [46] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2015, *arXiv:1511.06434*.
- [47] H. Huang, J. Deng, Y. Lan, A. Yang, L. Zhang, S. Wen, H. Zhang, Y. Zhang, and Y. Deng, "Detection of helminthosporium leaf blotch disease based on UAV imagery," *Appl. Sci.*, vol. 9, no. 3, p. 558, Feb. 2019.
- [48] J. Abdulridha, Y. Ampatzidis, S. C. Kakarla, and P. Roberts, "Detection of target spot and bacterial spot diseases in tomato using UAV-based and benchtop-based hyperspectral imaging techniques," *Precis. Agricult.*, vol. 21, no. 5, pp. 955–978, Oct. 2020.
- [49] J. Abdulridha, Y. Ampatzidis, J. Qureshi, and P. Roberts, "Laboratory and UAV-based identification and classification of tomato yellow leaf curl, bacterial spot, and target spot diseases in tomato utilizing hyperspectral imaging and machine learning," *Remote Sens.*, vol. 12, no. 17, p. 2732, Aug. 2020.
- [50] J. Abdulridha, Y. Ampatzidis, P. Roberts, and S. C. Kakarla, "Detecting powdery mildew disease in squash at different stages using UAV-based hyperspectral imaging and artificial intelligence," *Biosyst. Eng.*, vol. 197, pp. 135–148, Sep. 2020.
- [51] M. Agarwal, S. K. Gupta, and K. K. Biswas, "Development of efficient CNN model for tomato crop disease identification," *Sustain. Comput., Informat. Syst.*, vol. 28, Dec. 2020, Art. no. 100407.
- [52] A. Anagnostis, G. Asimari, E. Papageorgiou, and D. Bochtis, "A convolutional neural networks based method for anthracnose infected walnut tree leaves identification," *Appl. Sci.*, vol. 10, no. 2, p. 469, Jan. 2020.
- [53] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2261–2269.
- [54] M. Chen, F. Brun, M. Raynal, and D. Makowski, "Forecasting severe grape downy mildew attacks using machine learning," *PLoS ONE*, vol. 15, no. 3, Mar. 2020, Art. no. e0230254.
- [55] R. Cristin, B. S. Kumar, C. Priya, and K. Karthick, "Deep neural network based rider-cuckoo search algorithm for plant disease detection," *Artif. Intell. Rev.*, vol. 53, no. 7, pp. 4993–5018, Oct. 2020.

- [56] S. H. Lee, H. Goëau, P. Bonnet, and A. Joly, "New perspectives on plant disease characterization based on deep learning," *Comput. Electron. Agricult.*, vol. 170, Mar. 2020, Art. no. 105220.
- [57] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
- [58] S. Giraddi, S. Desai, and A. Deshpande, "Deep learning for agricultural plant disease detection," in *Proc. ICDSMLA*. Singapore: Springer, 2020, pp. 864–871.
- [59] Y. Guo, J. Zhang, C. Yin, X. Hu, Y. Zou, Z. Xue, and W. Wang, "Plant disease identification based on deep learning algorithm in smart farming," *Discrete Dyn. Nature Soc.*, vol. 2020, pp. 1–11, Aug. 2020.
- [60] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015.
- [61] P. Getreuer, "Chan-veise segmentation," *Image Process. Line*, vol. 2, pp. 214–224, Aug. 2012.
- [62] Institute of Botany of the Chinese Academy of Sciences. (2022). *Plant Photo Bank of China*. Accessed: Nov. 8, 2022. [Online]. Available: <https://ppbc.iplant.cn/>
- [63] M. T. Habib, A. Majumder, A. Z. M. Jakaria, M. Akter, M. S. Uddin, and F. Ahmed, "Machine vision based papaya disease recognition," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 32, no. 3, pp. 300–309, Mar. 2020.
- [64] K. Karadağ, M. E. Tenekeci, R. Taştan, and A. Bilgili, "Detection of pepper fusarium disease using machine learning algorithms based on spectral reflectance," *Sustain. Computing: Informat. Syst.*, vol. 28, p. 100299, 2020.
- [65] A. Karlekar and A. Seal, "SoyNet: Soybean leaf diseases classification," *Comput. Electron. Agricult.*, vol. 172, May 2020, Art. no. 105342.
- [66] R. Karthik, M. Hariharan, S. Anand, P. Mathikshara, A. Johnson, and R. Menaka, "Attention embedded residual CNN for disease detection in tomato leaves," *Appl. Soft Comput.*, vol. 86, Jan. 2020, Art. no. 105933.
- [67] M. Kerkech, A. Hafiane, and R. Canals, "Vine disease detection in UAV multispectral images using optimized image registration and deep learning segmentation approach," *Comput. Electron. Agricult.*, vol. 174, Jul. 2020, Art. no. 105446.
- [68] E. Khalili, S. Kouchaki, S. Ramazi, and F. Ghanati, "Machine learning techniques for soybean charcoal rot disease prediction," *Frontiers Plant Sci.*, vol. 11, Dec. 2020, Art. no. 590529.
- [69] E. Khalili. (2020). *Soybean Charcoal Rot Disease Prediction Dataset*. Accessed: Nov. 8, 2022. [Online]. Available: <https://github.com/Elham-khalili/Soybean-Charcoal-Rot-Disease-Prediction-Dataset-code>
- [70] S. Ramesh and D. Vydeki, "Recognition and classification of paddy leaf diseases using optimized deep neural network with Jaya algorithm," *Inf. Process. Agricult.*, vol. 7, no. 2, pp. 249–260, Jun. 2020.
- [71] S. Verma, A. Chug, and A. P. Singh, "Application of convolutional neural networks for evaluation of disease severity in tomato plant," *J. Discrete Math. Sci. Cryptogr.*, vol. 23, no. 1, pp. 273–282, Jan. 2020.
- [72] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: Alexnet-level accuracy with 50x fewer parameters and <0.5 MB model size," 2016, *arXiv:1602.07360*.
- [73] D. Velásquez, A. Sánchez, S. Sarmiento, M. Toro, M. Maiza, and B. Sierra, "A method for detecting coffee leaf rust through wireless sensor networks, remote sensing, and deep learning: Case study of the caturra variety in Colombia," *Appl. Sci.*, vol. 10, no. 2, p. 697, Jan. 2020.
- [74] Q. Yan, B. Yang, W. Wang, B. Wang, P. Chen, and J. Zhang, "Apple leaf diseases recognition based on an improved convolutional neural network," *Sensors*, vol. 20, no. 12, p. 3535, Jun. 2020.
- [75] E&M AI Lab. (2022). *Global AI Challenge for Building E&M Facilities*. Accessed: Nov. 8, 2022. [Online]. Available: <https://www.globalaichallenge.com/en/home>
- [76] Z. Zhang, P. Flores, C. Ighathinathane, D. L. Naik, R. Kiran, and J. K. Ransom, "Wheat lodging detection from UAS imagery using machine learning algorithms," *Remote Sens.*, vol. 12, no. 11, p. 1838, Jun. 2020.
- [77] A. Abbas, S. Jain, M. Gour, and S. Vankudothu, "Tomato plant disease detection using transfer learning with C-GAN synthetic images," *Comput. Electron. Agricult.*, vol. 187, Aug. 2021, Art. no. 106279.
- [78] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.
- [79] E. Akanksha, N. Sharma, and K. Gulati, "OPNN: Optimized probabilistic neural network based automatic detection of maize plant disease detection," in *Proc. 6th Int. Conf. Inventive Comput. Technol. (ICICT)*, Jan. 2021, pp. 1322–1328.
- [80] P. Bedi and P. Gole, "Plant disease detection using hybrid model based on convolutional autoencoder and convolutional neural network," *Artif. Intell. Agricult.*, vol. 5, pp. 90–101, Jan. 2021.
- [81] M. E. H. Chowdhury, T. Rahman, A. Khandakar, M. A. Ayari, A. U. Khan, M. S. Khan, N. Al-Emadi, M. B. I. Reaz, M. T. Islam, and S. H. M. Ali, "Automatic and reliable leaf disease detection using deep learning techniques," *AgriEngineering*, vol. 3, no. 2, pp. 294–312, May 2021.
- [82] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.
- [83] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.
- [84] V. Petsiuk. (2019). *Lung-Segmentation-2D: Ng Fields Segmentation on CXR Images Using Convolutional Neural Networks*. Accessed: Nov. 8, 2022. [Online]. Available: <https://github.com/imlab-uip/lung-segmentation-2d>
- [85] R. Dwivedi, S. Dey, C. Chakraborty, and S. Tiwari, "Grape disease detection network based on multi-task learning and attention features," *IEEE Sensors J.*, vol. 21, no. 16, pp. 17573–17580, Aug. 2021.
- [86] M. Mishra, P. Choudhury, and B. Pati, "Modified ride-NN optimizer for the IoT based plant disease detection," *J. Ambient Intell. Humanized Comput.*, vol. 12, no. 1, pp. 691–703, Jan. 2021.
- [87] A. M. Mostafa, S. A. Kumar, T. Meraj, H. T. Rauf, A. A. Alnuaim, and M. A. Alkhalayal, "Guava disease detection using deep convolutional neural networks: A case study of guava plants," *Appl. Sci.*, vol. 12, no. 1, p. 239, Dec. 2021.
- [88] B. V. Patil and P. S. Patil, "Computational method for cotton plant disease detection of crop management using deep learning and Internet of Things platforms," in *Evolutionary Computing and Mobile Sustainable Networks*. Singapore: Springer, 2021, pp. 875–885.
- [89] G. Sambasivam and G. D. Opiyo, "A predictive machine learning application in agriculture: Cassava disease detection and classification with imbalanced dataset using convolutional neural networks," *Egyptian Informat. J.*, vol. 22, no. 1, pp. 27–34, Mar. 2021.
- [90] Kaggle. (2019). *Cassava Disease Classification*. Accessed: Nov. 8, 2022. [Online]. Available: <https://www.kaggle.com/c/cassava-disease/overview>
- [91] R. Sujatha, J. M. Chatterjee, N. Jhanjhi, and S. N. Brohi, "Performance of deep learning vs machine learning in plant leaf disease detection," *Microprocessors Microsyst.*, vol. 80, Feb. 2021, Art. no. 103615.
- [92] H. T. Rauf, B. A. Saleem, M. I. U. Lali, M. A. Khan, M. Sharif, and S. A. C. Bukhari, "A citrus fruits and leaves dataset for detection and classification of citrus diseases through machine learning," *Data Brief*, vol. 26, Oct. 2019, Art. no. 104340.
- [93] V. Tiwari, R. C. Joshi, and M. K. Dutta, "Dense convolutional neural networks based multiclass plant disease detection and classification using leaf images," *Ecol. Informat.*, vol. 63, Jul. 2021, Art. no. 101289.
- [94] AIR Lab Makerere University. (2020). *iBean Leaf Image Dataset*. Accessed: Nov. 8, 2022. [Online]. Available: <https://github.com/AI-Lab-Makerere/ibean> and <https://www.fao.org/news/story/en/item/1187738/icode/>
- [95] H. B. Prajapati, J. P. Shah, and V. K. Dabhi, "Detection and classification of rice plant diseases," *Intell. Decis. Technol.*, vol. 11, no. 3, pp. 357–373, Aug. 2017.
- [96] Y. Zhao, Z. Chen, X. Gao, W. Song, Q. Xiong, J. Hu, and Z. Zhang, "Plant disease detection using generated leaves based on DoubleGAN," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 19, no. 3, pp. 1817–1826, May/Jun. 2021.
- [97] V. Kathole and M. Munot, "Deep learning models for tomato plant disease detection," in *Advanced Machine Intelligence and Signal Processing*. Singapore: Springer, 2022, pp. 679–686.
- [98] S.-Q. Pan, J.-F. Qiao, R. Wang, H.-L. Yu, C. Wang, K. Taylor, and H.-Y. Pan, "Intelligent diagnosis of northern corn leaf blight with deep learning model," *J. Integrative Agricult.*, vol. 21, no. 4, pp. 1094–1105, Apr. 2022.

- [99] D. Shah, V. Trivedi, V. Sheth, A. Shah, and U. Chauhan, "ResTS: Residual deep interpretable architecture for plant disease detection," *Inf. Process. Agricult.*, vol. 9, no. 2, pp. 212–223, Jun. 2022.
- [100] M. Turkoglu, B. Yanikoğlu, and D. Hanbay, "Plantdiseasenet: Convolutional neural network ensemble for plant disease and pest detection," *Signal, Image Video Process.*, vol. 16, no. 2, pp. 301–309, 2022.
- [101] M. Turkoglu. (2021). *Turkey-Plantdataset*. Accessed: Nov. 8, 2022. [Online]. Available: <https://github.com/mturkoglu23/PlantDiseaseNet>
- [102] S. Vallabhajosyula, V. Sistla, and V. K. K. Kolli, "Transfer learning-based deep ensemble neural network for plant leaf disease detection," *J. Plant Diseases Protection*, vol. 129, no. 3, pp. 545–558, Jun. 2022.
- [103] A. Howard, M. Sandler, B. Chen, W. Wang, L.-C. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le, "Searching for MobileNetV3," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1314–1324.
- [104] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8697–8710.
- [105] A. Fuentes, S. Yoon, S. Kim, and D. Park, "A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition," *Sensors*, vol. 17, no. 9, p. 2022, Sep. 2017.
- [106] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 379–387.
- [107] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis. Cham, Switzerland*: Springer, 2016, pp. 21–37.
- [108] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang, "Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks," *IEEE Access*, vol. 7, pp. 59069–59080, 2019.
- [109] W.-S. Kim, D.-H. Lee, and Y.-J. Kim, "Machine vision-based automatic disease symptom detection of onion downy mildew," *Comput. Electron. Agricult.*, vol. 168, Jan. 2020, Art. no. 105099.
- [110] D. Li, R. Wang, C. Xie, L. Liu, J. Zhang, R. Li, F. Wang, M. Zhou, and W. Liu, "A recognition method for rice plant diseases and pests video detection based on deep convolutional neural network," *Sensors*, vol. 20, no. 3, p. 578, Jan. 2020.
- [111] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.
- [112] M. H. Saleem, S. Khanchi, J. Potgieter, and K. M. Arif, "Image-based plant disease identification by deep learning meta-architectures," *Plants*, vol. 9, no. 11, p. 1451, Oct. 2020.
- [113] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 4278–4284.
- [114] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis. Springer*, 2014, pp. 740–755.
- [115] J. Sun, Y. Yang, X. He, and X. Wu, "Northern maize leaf blight detection under complex field environment based on deep learning," *IEEE Access*, vol. 8, pp. 33679–33688, 2020.
- [116] X. Xie, Y. Ma, B. Liu, J. He, S. Li, and H. Wang, "A deep-learning-based real-time detector for grape leaf diseases using improved convolutional neural networks," *Frontiers Plant Sci.*, vol. 11, p. 751, Jun. 2020.
- [117] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
- [118] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
- [119] Y. Zhang, C. Song, and D. Zhang, "Deep learning-based object detection improvement for tomato disease," *IEEE Access*, vol. 8, pp. 56607–56614, 2020.
- [120] A. M. Roy and J. Bhaduri, "A deep learning enabled multi-class plant disease detection model based on computer vision," *AI*, vol. 2, no. 3, pp. 413–428, Aug. 2021.
- [121] M. G. Selvaraj, A. Vergara, F. Montenegro, H. A. Ruiz, N. Safari, D. Raymaekers, W. Ocimati, J. Ntamwira, L. Tits, A. B. Omondi, and G. Blomme, "Detection of banana plants and their major diseases through aerial images and machine learning methods: A case study in DR Congo and republic of Benin," *ISPRS J. Photogramm. Remote Sens.*, vol. 169, pp. 110–124, Nov. 2020.
- [122] J. Wang, L. Yu, J. Yang, and H. Dong, "DBA_SSD: A novel end-to-end object detection algorithm applied to plant disease detection," *Information*, vol. 12, no. 11, p. 474, Nov. 2021.
- [123] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.
- [124] Z. Chen, R. Wu, Y. Lin, C. Li, S. Chen, Z. Yuan, S. Chen, and X. Zou, "Plant disease recognition model based on improved YOLOv5," *Agronomy*, vol. 12, no. 2, p. 365, Jan. 2022.
- [125] A. M. Roy, R. Bose, and J. Bhaduri, "A fast accurate fine-grain object detection model based on YOLOv4 deep neural network," *Neural Comput. Appl.*, vol. 34, no. 5, pp. 3895–3921, Mar. 2022.
- [126] H. Wang, S. Shang, D. Wang, X. He, K. Feng, and H. Zhu, "Plant disease detection and classification method based on the optimized lightweight YOLOv5 model," *Agriculture*, vol. 12, no. 7, p. 931, Jun. 2022.
- [127] S. Siddharth, U. Singh, A. Kaul, and S. Jain, "A database of leaf images: Practice towards plant conservation with plant pathology," Mendeley Data, 2019. [Online]. Available: <https://data.mendeley.com/datasets/hb74ynkjc/1>
- [128] V. P. Kour and S. Arora, "Plantaek: A leaf database of native plants of jammu and kashmir," in *Recent Innovations in Computing*. Singapore: Springer, 2022, pp. 359–368.
- [129] R. Thapa, K. Zhang, N. Snavely, S. Belongie, and A. Khan, "The plant pathology challenge 2020 data set to classify foliar disease of apples," *Appl. Plant Sci.*, vol. 8, no. 9, p. e11390, Sep. 2020.
- [130] Science Data Bank. (2019). *Corn Northern Leaf Blight Dataset*. Accessed: Nov. 8, 2022. [Online]. Available: <https://www.scidb.cn/en/detail?dataSetId=557575357876666368&dataSetType=project&code=p00001&tID=journalOne>
- [131] D. Singh, N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, "PlantDoc: A dataset for visual plant disease detection," in *Proc. 7th ACM IKDD CoDS 25th COMAD*, 2020, pp. 249–253.
- [132] Y.-Y. Zheng, J.-L. Kong, X.-B. Jin, X.-Y. Wang, and M. Zuo, "CropDeep: The crop vision dataset for deep-learning-based classification and detection in precision agriculture," *Sensors*, vol. 19, no. 5, p. 1058, Mar. 2019.
- [133] H. Wu, T. Wiesner-Hanks, E. L. Stewart, C. DeChant, N. Kaczmar, M. A. Gore, R. J. Nelson, and H. Lipson, "Autonomous detection of plant disease symptoms directly from aerial imagery," *Plant Phenome J.*, vol. 2, no. 1, pp. 1–9, Jan. 2019.
- [134] J. Howard and S. Gugger, "Fastai: A layered API for deep learning," *Information*, vol. 11, no. 2, p. 108, Feb. 2020.
- [135] L. N. Smith, "A disciplined approach to neural network hyperparameters: Part 1—Learning rate, batch size, momentum, and weight decay," 2018, *arXiv:1803.09820*.
- [136] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10778–10787.
- [137] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
- [138] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2999–3007.
- [139] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.
- [140] G. Jocher. (2022). *Ultralytics/YOLOv5: V3.1—Bug Fixes and Performance Improvements*. Accessed: Nov. 8, 2022. [Online]. Available: <https://github.com/ultralytics/yolov5>
- [141] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.
- [142] S. Ruder, "An overview of gradient descent optimization algorithms," 2016, *arXiv:1609.04747*.
- [143] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
- [144] Ultralytics. *Architecture Summary*. Accessed: Jul. 1, 2023. [Online]. Available: https://docs.ultralytics.com/yolov5/tutorials/architecture_description/
- [145] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," 2016, *arXiv:1611.05431*.
- [146] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6848–6856.

- [147] S. Zagoruyko and N. Komodakis, "Wide residual networks," 2016, *arXiv:1605.07146*.
- [148] J. Deere. *See & Spray Ultimate*. Accessed: Sep. 14, 2023. [Online]. Available: <https://www.deere.com/en/sprayers/see-spray-ultimate/>
- [149] A. Yeshe, P. Gourkhede, and P. Vaidya, "Blue river technology: Futuristic approach of precision farming," *Just Agricult.*, vol. 2, no. 7, pp. 1–14, 2022.
- [150] IBM. *Agriculture—Environmental Intelligence Suite*. Accessed: Sep. 14, 2023. [Online]. Available: <https://www.ibm.com/products/environmental-intelligence-suite/agriculture>



VASILEIOS BALAFAS received the B.Sc. (integrated master's) degree from the Department of Electrical and Computer Engineering, University of Western Macedonia, Greece, in 2021, where he is currently pursuing the Ph.D. degree, working on developing optimization and constraint programming algorithms for cloud computing optimization. His Diploma thesis focused on computationally comparing object detection methods. He has published one journal article and two conference papers.



EMMANOUIL KARANTOUMANIS received the B.Sc. (integrated master's) degree from the Department of Informatics and Telecommunications Engineering, University of Western Macedonia, Greece, in 2019. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Western Macedonia, working on developing hybrid optimization and machine learning algorithms in order to optimize black-box problems.

His Diploma thesis focused on applying machine learning algorithms in order to predict power consumption in data centers. He has published five conference papers.



MALAMATI LOUTA (Senior Member, IEEE) received the M.Eng. and Ph.D. degrees in electrical and computer engineering, in 1997 and 2000, respectively, and the M.B.A. degree from the National Technical University of Athens, in 2004. She is currently the Director of the Telecommunication Networks and Advanced Services Laboratory and an Associate Professor with the Electrical and Computer Engineering Department, School of Engineering, University of Western Macedonia, Greece. Her research interests include telecommunication networks and advanced services engineering. She is a member of the ACM and the Technical Chamber of Greece. She serves as an associate editor, the general chair, the technical program committee chair, a member, a session organizer, and a reviewer for a number of international conferences and journals.



NIKOLAOS PLOSKAS received the B.Sc., M.Sc., and Ph.D. degrees in applied informatics from the University of Macedonia, Thessaloniki, Greece, in 2007, 2009, and 2014, respectively. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, University of Western Macedonia. His research interests include mathematical programming, optimization, operations research, analysis and design of algorithms, cloud computing, and machine learning. He is also serving on the editorial board for four scientific journals. He has served on the organizing/program committee for 20 international conferences.

...