



Review

# Review on Convolutional Neural Network (CNN) Applied to Plant Leaf Disease Classification

Jinzhu Lu <sup>1,2,\*</sup>, Lijuan Tan <sup>1,2</sup>  and Huanyu Jiang <sup>3</sup> 

<sup>1</sup> Modern Agricultural Equipment Research Institute, Xihua University, Chengdu 610039, China; tanlijuan@stu.xhu.edu.cn

<sup>2</sup> School of Mechanical Engineering, Xihua University, Chengdu 610039, China

<sup>3</sup> College of Biosystems Engineering and Food Science, Zhejiang University, Hangzhou 310058, China; hyjiang@zju.edu.cn

\* Correspondence: lujingzhu1103@163.com

**Abstract:** Crop production can be greatly reduced due to various diseases, which seriously endangers food security. Thus, detecting plant diseases accurately is necessary and urgent. Traditional classification methods, such as naked-eye observation and laboratory tests, have many limitations, such as being time consuming and subjective. Currently, deep learning (DL) methods, especially those based on convolutional neural network (CNN), have gained widespread application in plant disease classification. They have solved or partially solved the problems of traditional classification methods and represent state-of-the-art technology in this field. In this work, we reviewed the latest CNN networks pertinent to plant leaf disease classification. We summarized DL principles involved in plant disease classification. Additionally, we summarized the main problems and corresponding solutions of CNN used for plant disease classification. Furthermore, we discussed the future development direction in plant disease classification.

**Keywords:** plant disease classification; deep learning; machine learning; convolutional neural network



**Citation:** Lu, J.; Tan, L.; Jiang, H. Review on Convolutional Neural Network (CNN) Applied to Plant Leaf Disease Classification.

*Agriculture* **2021**, *11*, 707. <https://doi.org/10.3390/agriculture11080707>

Academic Editors: Grazia Licciardello and Giuliana Loconsole

Received: 27 May 2021

Accepted: 23 July 2021

Published: 27 July 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

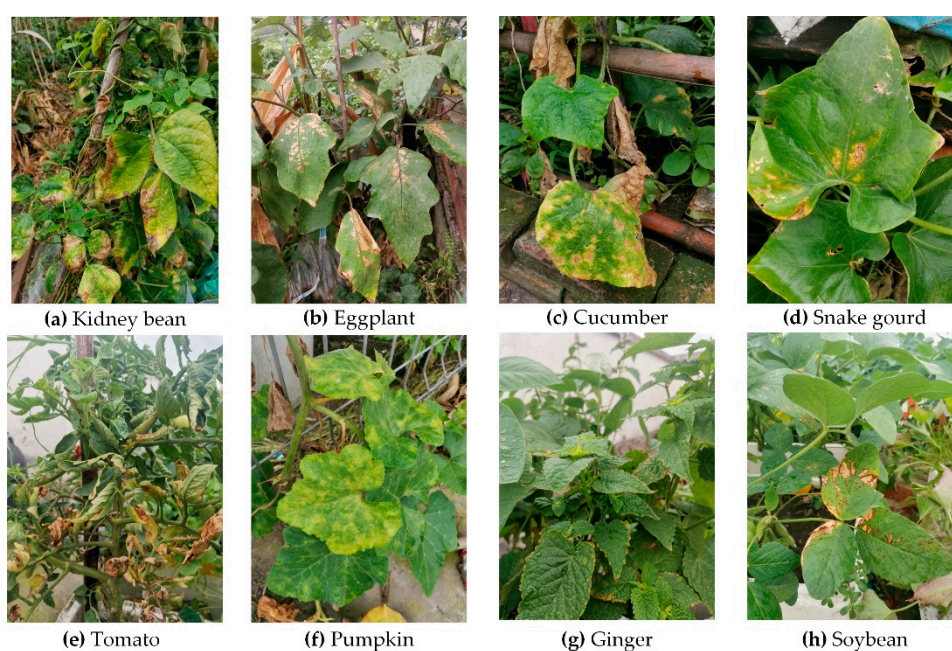
## 1. Introduction

The Food and Agriculture Organization of the United Nations (<http://www.fao.org/publications/sofi/2020/en/>, accessed on 5 December 2020) reported that the number of hungry people in the world has been increasing slowly since 2014. Current estimates show that nearly 690 million people are hungry, and they account for 8.9% of the world's total population; this figure represents an increase of 10 million in 1 year and nearly 60 million in 5 years. Meanwhile, more than 90% of people in the world rely on agriculture. Farmers produce 80% of the world's food [1]; however, more than 50% of crop production is lost due to plant diseases and pests [2]. Thus, recognizing and detecting plant disease accurately is necessary and urgent.

The diverse plant diseases have an enormous effect on growing food crops. An iconic example is the Irish potato famine of 1845–1849, which resulted in 1.2 million deaths [3]. The diseases of several common plants are shown in Table 1. Plant diseases can be systematically divided into fungal, oomycete, hyphomycete, bacterial, and viral types. We have shown some pictures of plant disease in Figure 1. The pictures in Figure 1 were taken in the greenhouse of Chengdu Academy of Agriculture and Forestry Sciences. Researchers and farmers have never stopped exploring how to develop an intelligent and effective method for plant disease classification. Laboratory test approaches to plant samples, such as polymerase chain reaction, enzyme-linked immunosorbent assay, and loop-mediated isothermal amplification, are highly specific and sensitive in identifying diseases.

**Table 1.** Common diseases of several common plants.

Plant	Major Types of Disease			Reference
	Fungal	Bacterial	Viral	
Cucumber	Downy mildew, powdery mildew, gray mold, black spot, anthracnose	Angular spot, brown spot, target spot	Mosaic virus, yellow spot virus	Kianat et al. (2021) [4], Zhang et al. (2019) [5], Agarwal et al. (2021) [6]
Rice	Rice stripe blight, false smut, rice blast	Bacterial leaf blight, bacterial leaf streak	Rice leaf smut, rice black-streaked dwarf virus	Chen et al. (2021) [7], Shrivastava et al. (2019) [8]
Maize	Leaf spot disease, rust disease, gray leaf spot	Bacterial stalk rot, bacterial leaf streak	Rough dwarf disease, crimson leaf disease	Sun et al. (2021) [9], Yu et al. (2014) [10]
Tomato	Early blight, late blight, leaf mold	Bacterial wilt, soft rot, canker	Tomato yellow leaf curl virus	Ferentinos (2018) [11], Abbas et al. (2021) [12]

**Figure 1.** Leaf spot in eight common plants. We took these pictures in the greenhouse of Chengdu Academy of Agriculture and Forestry Sciences.

However, conventional field scouting for diseases in crops still relies primarily on visual inspection of the leaf color patterns and crown structures. People observe the symptoms of diseases on plant leaves with the naked eye and diagnose plant diseases based on experience, which is time and labor consuming and requires specialized skills [13]. At the same time, the disease characteristics among different crops are also different due to the variety of plants; this condition brings a high degree of complexity in the classification of plant diseases. Meanwhile, many studies have focused on the classification of plant diseases based on machine learning. Using machine learning methods to detect plant diseases is mainly divided into the following three steps: first, using preprocessing techniques to remove the background or segment the infected part; second, extracting the distinguishing features for further analysis; finally, using supervised classification or unsupervised clustering algorithms to classify the features [14–17]. Most machine learning studies have focused on the classification of plant diseases by using features, such as the texture [18], type [19], and color [20] of plant leaf images. The main classification methods include support vector machines [19], K-nearest neighbor [20], and random forest [21]. The major disadvantages of these methods are summarized as follows:

Low performance [22]: The performance they obtained was not ideal and could not be used for real-time classification.

Professional database [23]: The datasets they applied contained plant images that were difficult to obtain in actual life. In the case of PlantVillage, the dataset was taken in an ideal laboratory environment, such that a single image contains only one plant leaf and the shot is not influenced by the external environment (e.g., light, rain).

Rarely used [24,25]: They often need to manually design and extract features, which require research staff to possess professional capabilities.

Requiring the use of segmented operation [26]: The plants must be separated from their roots to gain research datasets. Obviously, this operation is not good for real-time applications.

Most of the traditional machine learning algorithms were based on laboratory conditions, and the robustness of the algorithms is insufficient to meet the needs of practical agricultural applications. Nowadays, deep learning (DL) methods, especially those based on convolutional neural networks (CNNs), are gaining widespread application in the agricultural field for detection and classification tasks, such as weed detection [27], crop pest classification, and plant disease identification [28]. DL is a research direction of machine learning. It has solved or partially solved the problems of low performance [22], lack of actual images [23], and segmented operation [26] of traditional machine learning methods. The important advantage of DL models are that they can extract features without applying segmented operation while obtaining satisfactory performance. Features of an object are automatically extracted from the original data. Kunihiko Fukushima introduced the Neocognitron in 1980, which inspired CNNs [29]. The emergence of CNNs has made the technology of plant disease classification increasingly efficient and automatic.

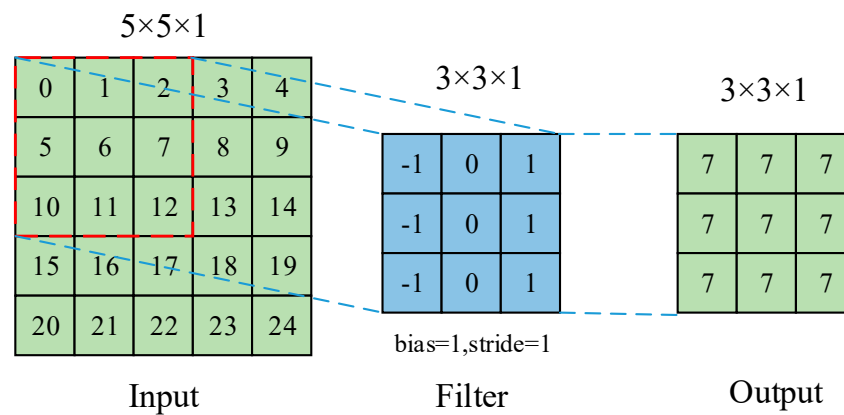
The main works of this study are given as follows: (1) we reviewed the latest CNN networks pertinent to plant leaf disease classification; (2) we summarized DL principles involved in plant disease classification; (3) we summarized the main problems and corresponding solutions of CNN used for plant disease classification, and (4) we discussed the direction of future developments in plant disease classification.

## 2. Deep Learning

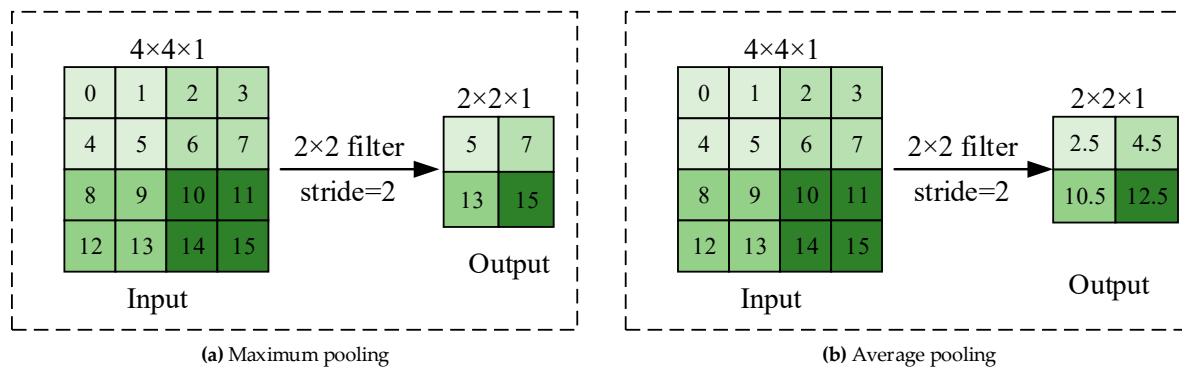
DL is a branch of machine learning [30] and is mainly used for image classification, object detection [31–34], and natural language processing [35–37].

DL is an algorithm based on a neural network for automatic feature selection of data. It does not need a lot of artificial feature engineering. It combines low-level features to form abstract high-level features for discovering distributed features and attributes of sample data. Its accuracy and generalization ability are improved compared to those of traditional methods in image recognition and target detection. Currently, the main types of networks are multilayer perceptron, CNN, and recurrent neural network (RNN). CNN is the most widely used for plant leaf disease classification. As for other DL networks, such as fully convolutional networks (FCNs) and deconvolutional networks, they are usually used for image segmentation [38–41] or medical diagnosis [42,43] but are not used for plant leaf disease classification. CNN usually consists of convolutional, pooling, and fully connected layers. The convolutional layer uses the local correlation of the information in the image to extract features. The process of convolution operation is shown in Figure 2. A kernel is placed in the top-left corner of the image. The pixel values covered by the kernel are multiplied with the corresponding kernel values, and then the products are summated, and the bias is added at the end. The kernel is moved over by one pixel, and the process is repeated until all possible locations in the image are filtered, which is shown in Figure 2. The pooling layer selects features from the upper layer feature map by sampling and simultaneously makes the model invariant to translation, rotation, and scaling. The commonly used one is maximum or average pooling. The process of the pooling operation is shown in Figure 3. Maximum pooling is to divide the input image into several rectangular regions based on the size of the filter and output the maximum value for each region. As for average pooling, the output is the average of each region. Convolutional and pooling layers often appear alternately in applications. Each neuron in the fully connected layer is connected to the upper neuron, and

the multidimensional features are integrated and converted into one-dimensional features in the classifier for classification or detection tasks [44].



**Figure 2.** The process of convolution operation.



**Figure 3.** The process of pooling operation.

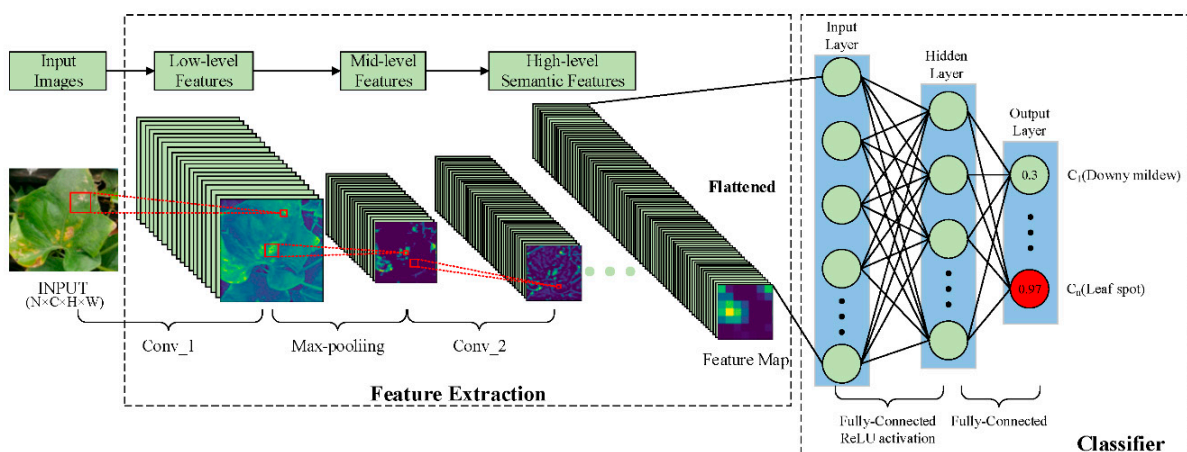
For classification tasks, various CNN-based classification models have been developed in DL-related research, including AlexNet, VGGNet, GoogLeNet, ResNet, MobileNet, and EfficientNet. AlexNet [45] was proposed in 2012 and was the champion network in the ILSVRC-2012 competition. This network contains five convolutional layers and three fully connected layers. AlexNet has the following four highlights: (a) it is the first model to use a GPU device for network acceleration training; (b) rectified linear units (ReLUs) were used as the activation function; (c) local response normalization was used; (d) in the first two layers of the fully connected layer, the dropout operation was used to reduce overfitting. Then, the deeper networks appeared, such as VGG16, VGG19, GoogLeNet. These networks use smaller stacked kernels but have lower memory during inference [46]. Later, researchers found that when the number of layers of a deep CNN reached a certain depth, blindly increasing the number of layers would not improve the classification performance but would cause the network to converge more slowly [47,48]. Until 2015, Microsoft lab proposed the ResNet network and won the first place in the classification task of the ImageNet competition. The network creatively proposed residual blocks and shortcut connections [49], which solves the problem of gradient elimination or gradient explosion, making it possible to build a deeper network model. ResNet influenced the development direction of DL in academia and industry in 2016. MobileNet was proposed by the Google teams in 2017 and was designed for mobile and embedded vision applications [50]. In 2019, the Google teams proposed another outstanding network: EfficientNet [51]. This network uses a simple yet highly efficient compound coefficient to uniformly scale all dimensions of depth/width/resolution, which will not arbitrarily scale the dimensions of the network as in traditional methods. As for plant disease classification tasks, it is not necessary to use

deep networks, because simple models, such as AlexNet and VGG16, can meet the actual accuracy requirements.

The DL model can be realized using programming languages, such as Python, C/C++. The open-source DL framework provides a series of application programming interfaces, supports model design, assists in network deployment, and avoids code duplication [52]. At present, DL frameworks, such as PyTorch (<https://pytorch.org/>, accessed on 5 March 2021), Tensorflow (<https://www.tensorflow.org/>, accessed on 7 March 2021), Caffe (<https://caffe.berkeleyvision.org/>, accessed on 8 March 2021), and Keras (<https://keras.io/>, accessed on 10 March 2021) are widely used.

The rapid increase of DL is inseparable from the widespread development of GPU. The implementation of deep CNN requires GPUs to provide computing power support, otherwise it will cause the training process to be quite slow or make it impossible to train CNN models at all. At present, the most used is CUDA. When NVIDIA launched CUDA (Computing Unified Device Architecture) and AMD launched Stream, GPU computing started [46], and now, CUDA is widely used in DL.

Image classification is a basic task in computer vision. It is also the basis of object detection, image segmentation, image retrieval, and other technologies. The basic process of DL is shown in Figure 4, taking the task of classification of diseases on the surface of snake gourd leaves as an example. In Figure 4, we use a CNN-based architecture to extract features, which mainly include convolutional, max-pooling, and full connection layers. The convolutional layer is mainly used to extract features of snake gourd plant leaf images. The shallow convolutional layer is used to extract some edge and texture information, the middle layer is used to extract complex texture and part of semantic information, and the deep layer is used to extract high-level semantic features. The convolutional layer is followed by a max-pooling layer, which is used to retain the important information in the image. At the end of the architecture is a classifier, which consists of full connection layers. This classifier is used to classify the high-level semantic features extracted by the feature extractor.



**Figure 4.** Convolutional neural networks for snake gourd leaf disease classification.

In Figure 4, we input a batch of images into the feature extraction network to extract the features and then flatten the feature map into the classifier for disease classification. This process can be roughly divided into the following three steps.

1. Step 1. Preparing the Data and Preprocessing
2. Step 2. Building, Training, and Evaluating the Model
3. Step 3. Inference and Deployment

### 2.1. Data Preparation and Preprocessing

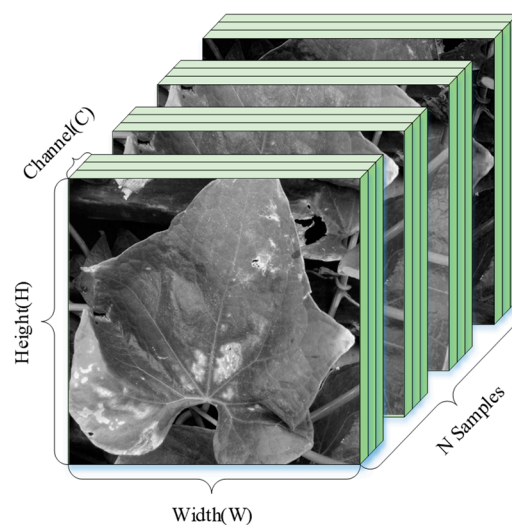
Data are important for DL models. The results are bound to be inaccurate no matter how complex and perfect our model is as long as the quality of the input data is poor. The typical percentages of the original dataset intended for training, validation, and test are 70:20:10, 80:10:10, and 60:20:20.

A DL dataset is usually composed of a training set, a validation set, and a test set. The training set is used to make the model learn, and the validation set is usually used to adjust hyperparameters during training. The test set is the sample of data that the model has not seen before, and it is used to evaluate the performance of the DL model. We collected some public plant datasets from the two websites Kaggle (<https://www.kaggle.com/datasets>, accessed on 12 February 2021) and BIFROST (<https://datasets.bifrost.ai/>, accessed on 15 February 2021), which can be used for detection or classification tasks, as shown in Table 2. In the literature of DL techniques applied to plant disease classification, the most used public datasets are PlantVillage [53–55] and Kaggle [56]; notably, many authors also collect their own datasets [57–60].

**Table 2.** Some public plant datasets from Kaggle and BIFROST.

Name	Number of Images	Classes	Task	Type of View	Source
New Plant Diseases Dataset	87,000	38	Image classification	Field data	Kaggle
PlantVillage Dataset	162,916	38	Image classification	Uniform background	Kaggle
Flowers Recognition	4242	4	Image classification	Field data	Kaggle
Plant Seedlings Dataset	5539	12	Target detection	Field data	BIFROST
Weed Detection in Soybean Crops	15,336	4	Target detection	Uniform background	Kaggle

For snake gourd leaf disease classification, we need a large number of leaf images of different disease categories. Meanwhile, the disease image data of each category were roughly balanced. If one disease with a particularly large number of image data is considered, then the neural network will be biased toward this disease. Apart from sufficient data on category balance, it also needs data to preprocess including image resize, random crop, and normalization. The shape of the data varies according to the framework used. Figure 5 shows the tensor shape of the input for the neural network, where H and W represent the height and width of the preprocessed image, C represents the number of image channels (gray or RGB), and N represents the number of images input to the neural network in a training session.



**Figure 5.** The tensor shape of the input neural network in PyTorch.

## 2.2. Building Model Architecture, Training, and Evaluating the Model

Before training, a suitable DL model architecture is needed. A good model architecture can result in more accurate classification results and more rapid classification speed. Currently, the main network types of DL are CNN, RNN, and generative adversarial networks (GAN). Among various works, CNN is the most widely used feature extraction network for the task of plant disease detection and classification [55,61–65].

After the model architecture is established, different hyperparameters are set for training and evaluation. We can set some parameter combinations and use the grid search method to iterate through them to find the best one. When training the neural network, training data are placed into the first layer of the network, and each neuron updates the weight of the neuron through back-propagation according to whether the output is equal to the label. This process is repeated until new capability is learned from existing data. However, whether the trained model has learned new capabilities is unknown. The performance of the model was evaluated by criteria, such as accuracy, precision, recall, and F1 score. The concept of a confusion matrix must be introduced first prior to introducing these indexes specifically. The confusion matrix shows the predicted correct or incorrect results in binary classification. It consists of four elements: true positive (*TP*, correctly predicted positive values), false positive (*FP*, incorrectly predicted positive values), true negative (*TN*, correctly predicted negative values), and false negative (*FN*, incorrectly predicted negative values). Then, the accuracy can be calculated as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

Among all the positives predicted by the model, precision predicts the proportion of correct predictions.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Among all real positives, recall predicts the correct proportion of positives [66].

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

The F1 value considers precision (*P*) and recall (*R*) rates.

$$F1 = \frac{2}{\frac{1}{P} + \frac{1}{R}} = \frac{2 \times P \times R}{P + R} \quad (4)$$

In the studies on plant disease classification, accuracy is the most common evaluation index [53,60,64,67,68]. Larger values of accuracy, precision, and recall are better. Within a certain range, when the value of the F1 score is smaller, the better the generalization performance of the trained model is. When the training and evaluation are complete, the trained model has a new capability; then, this capability is applied to new data.

## 2.3. Inference and Deployment

The inference is the capability of the DL model to quickly apply the learning capability by the trained model to new data and quickly provide the correct answer based on data that it has never seen [69]. After the training process is completed, the networks are deployed into the field for inferring a result for the provided data, which they have never seen before. Only then can the trained deep learning models be applied in real agricultural environments. We can deploy the trained model to the mobile terminal, cloud, or edge devices, such as by using an application on the mobile phone to take photos of plant leaves and judge diseases [70]. In addition, in order to use the trained model better in the field, the generalization ability of the model needs to be improved, and we can continuously update the models with the new labeled datasets to improve the generalization ability [71].

### 3. Problems and Solutions

Before 2015, no notable breakthrough was obtained in plant disease classification. With the fast development of DL since 2015, DL has been widely used in plant disease detection and classification and represents state-of-the-art technology in this field. For plant leaf disease classification, CNN-based models are the most used. In this section, we introduce and summarize the problems and solutions existing in the development of CNN-based DL methods applied to plant disease detection and classification. The problems are caused by extrinsic and intrinsic factors. Sections 3.1 and 3.2 discuss extrinsic factors, and Sections 3.3 and 3.4 describe intrinsic factors.

#### 3.1. Insufficient Datasets

The most important problem of CNN-based DL's application of plant disease classification is insufficient datasets in size and diversity. All the other introduced problems are also partially due to this condition.

Mohanty et al. tested the classic network models AlexNet and GoogLeNet with a public database of 54,306 images collected under controlled conditions to identify 14 crop species and 26 diseases. They obtained a top accuracy of 99.35%, which demonstrates the feasibility of this method. However, the accuracy of the model was greatly reduced when it was tested on a set of images taken under conditions different from the images used for training because of the insufficient diversity of the training set. In addition, plant disease identification in this experiment was realized under ideal conditions, such as single leaves, facing up, in a homogeneous background; thus, the accuracy rate would be much lower in practical applications [53]. Fuentes et al. aimed to introduce a robust DL-based detector for real-time tomato disease and pest recognition. All images of plant diseases and pests were taken in-place, including background variations, different illumination conditions, and multiple sizes of objects. The precision would be lower in practical application due to the insufficient number of samples [72]. Sufficient datasets have an important influence on the practical application. However, collecting data is easily affected by environmental factors, such as season and climate, and image labeling is also a time-consuming and laborious task. These factors make producing an effective dataset extremely difficult. Currently, five ways, namely, transfer learning, data augmentation techniques, few-shot learning, citizen science, and data sharing, can be used to resolve dataset problems.

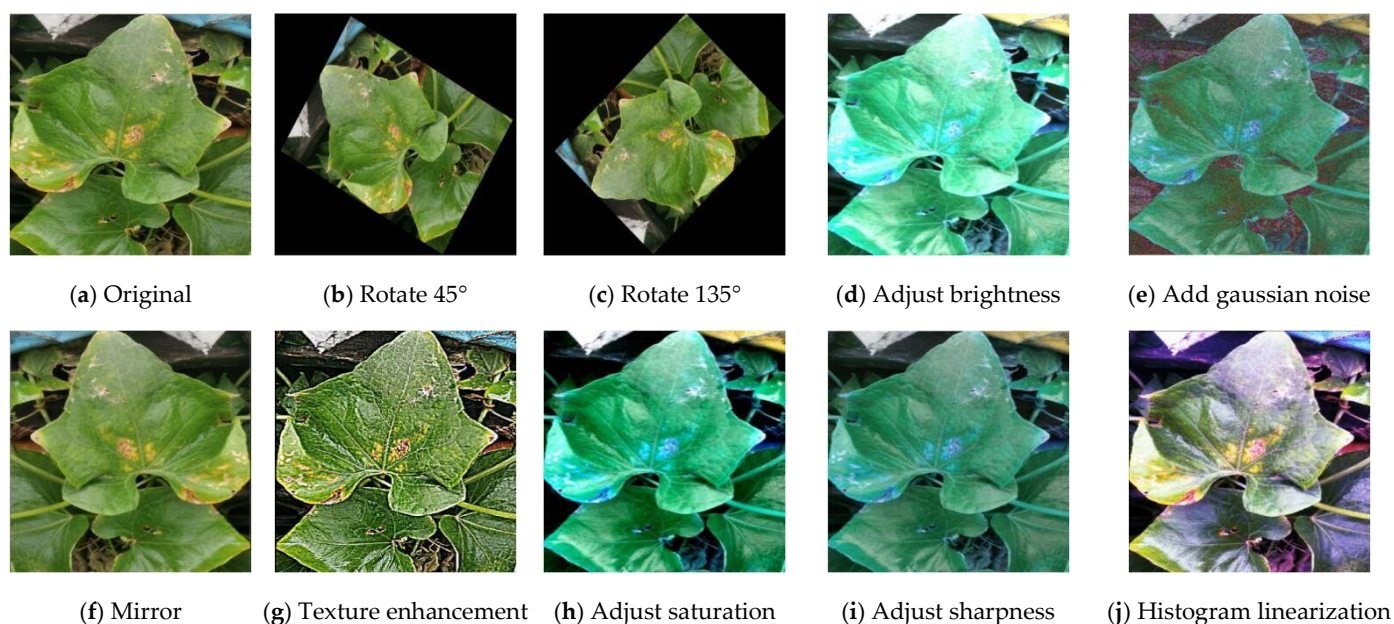
Transfer learning is a machine learning technique, where the attained capability from the previous task is transferred to later tasks [36]. Only a few layers of pretrained networks are retrained with the new databases, which is good for reducing the need for masses of datasets [73]. Mukti et al. utilized a transfer learning model based on ResNet50 to recognize plant diseases. Their dataset contains 87,867 images. A total of 80% of the dataset was used for training and 20% for validating. The highest accuracy they attained was 99.80% [1]. Coulibaly et al. proposed an approach using transfer learning to recognize mildew diseases in pearl millet. This approach was based on a classical CNN model VGG16 and pretrained on public dataset ImageNet. The experiment resulted in a satisfactory performance with an accuracy of 95% and a recall of 94.5% [74]. Abdalla et al. used three transfer learning methods for semantic segmentation of oilseed rape images; the experiment resulted in an accuracy of 96% and demonstrated that transfer learning gained high performance in this segmentation task [75]. Chen et al. proposed a DL architecture named INC-VGGN, which utilized the transfer learning by modifying the pretrained VGGNet for the identification task of plant leaf diseases. The proposed model achieved an accuracy of 91.83% on the public dataset PlantVillage and 92.00% on their own dataset [60]. Table 3 summarizes some studies that used transfer learning technology for classification or detection tasks.



**Table 3.** Studies on transfer learning technology applied to the identification task.

Pretrained Model	Dataset	Number of Class	Best Accuracy	Reference
ResNet50	PlantVillage (extended)	38	99.80%	Mukti and Biswas (2019) [1]
VGG16	Millet crop images (own)	7	95.00%	Coulibaly et al. (2019) [74]
VGG16	Plant images (own)		93.00%	Abdalla et al. (2019) [75]
VGGNet	ImageNet	9	91.83%	Chen et al. (2020) [60]
ResNet-101	NBAIR (extended)	40	95.02%	Thenmozhi et al. (2019) [76]
AlexNet	ImageNet (partial)	2	98.00%	Suh et al. (2018) [77]

The data augmentation technologies can efficiently increase the number of datasets. We show some traditional image data augmentation methods, such as rotation, mirror symmetry, and adjusting saturation in Figure 6. We have learned some newest augmentation technologies: AugMix [78], population-based augmentation [79], Fast AutoAugment [80], RandAugment [81], and CutMix [82].

**Figure 6.** Traditional image data augmentation methods.

Liu et al. used data augmentation technologies to solve the problem of insufficient apple pathological images for the identification of four apple leaf diseases. The researchers used direction disturbance (rotation transformation and mirror symmetry), light disturbance, and principal component analysis jittering to disturb natural images. With the application of these image processing technologies, the dataset expanded from 1053 images to 13,689 images, and the accuracy with the expanded database improved 10.83% over that in the nonexpanded database [83]. The researchers in [58] used three augmentation methods (noise addition, color jittering, and radial blur) to increase the number of databases. Douarre et al. used a novel data augmentation strategy, namely, plant canopy simulation, to generate new annotated data for the segmentation task of plant disease. The results showed that simulated data had increasing segmentation performance [84]. Table 4 summarizes some studies on using data augmentation technologies to expand the dataset.

Another method is few-shot learning (FSL), which needs small training sets but with a small drop in accuracy. Argüeso et al. [85] introduced FSL algorithms for plant disease classification to address the problem of requiring large annotative image datasets for DL methods. They split the 54,303 images of the PlantVillage dataset into a source and a target domain. First, they used the fine-tuning Inception V3 network in the source domain to learn general plant leaf characteristics. Then, these characteristics were transferred to

the target domain to learn new leaf types from few images. For the FSL method, the DL Siamese network with Triplet loss was utilized. The results demonstrated that dataset size could be reduced by 89.1% with only a 4% loss in accuracy, that is, this method is good for small training sets.

The concept of citizen science was proposed in 1995. In this method, nonprofessional volunteers collect and/or process data as part of a scientific inquiry. In the case of plant disease and pest classification, farmers and field workers upload the collected images to a server; then, those images would be properly labeled and processed by an expert [86]. This idea has been applied in practice. PEAT (a company in Berlin) has built an Android APP called Plantix that supports farmers with small networks.

Another method for expanding datasets is data sharing. Now, many studies focus on automatic disease classification around the world. If the various datasets are shared and properly integrated, then the database will be more representative. This condition will promote more meaningful and satisfactory research results.

**Table 4.** Studies on using data augmentation technologies to expand the dataset.

Expanded Dataset	Methods	Best Accuracy	Reference
From 1053 to 13,689 images	Direction disturbance and light disturbance and PCA (Principal components analysis) jittering	97.62%	Bin et al. (2017) [83]
From 10,820 to 32,460 images	Noise addition, color jittering, and radial blur	96.17% (improved 3.15%)	Lin et al. (2018) [58]
From 54,309 to 87,848 images	Cropping, resizing	99.53%	Ferentinos (2018) [11]
From 1567 to 46,409 images	Segmentation, resizing	94.00% (improved 12%)	Arnal Barbedo (2019) [86]
From 5000 to 43,398 images	Resizing, crop, rotation, noise...	85.98%	Fuente et al. (2017) [72]
From 4483 to 33,469 images	Affine transformation, perspective transformation, and rotation	96.30%	Srdjan et al. (2016) [87]

### 3.2. Nonideal Robustness

In classic DL problems, we often assume that the training and test sets have the same distribution. Usually, we train the model on the training set and test the model on the test set. However, the test scenario is often uncontrollable in actual application. The distribution of the test set is really different from the training set due to various factors, such as the influence of season and climate. Under the circumstances, the overfitting problem appears, that is, the trained model does not work well in practical application. This nonideal robustness problem was confirmed by Mohanty et al. [53], who trained and tested deep CNN (DCNN) models with the PlantVillage dataset; the top accuracy they obtained was 99.35%. However, when the DCNN models were tested on a set of images taken under conditions that were different from the training set, the accuracy dropped to 31% [53]. Similarly, Ferentinos used CNN models (i.e., AlexNet, GoogLeNet, and VGG) to detect and recognize plant diseases with a public dataset PlantVillage. When the model was trained and tested with PlantVillage, the best success was 99.53% with the VGG model. However, when they trained the VGG model with laboratory images and tested it with field images, the success rate was only up to 33.27% [11].

Three ways can be used to improve the robustness of CNN models. Compressed models that have a simpler set of parameters show more robustness and less overfitting. However, compressed models achieve poor performance in dealing with complex recognition. Unsupervised-based DL methods are also good at achieving more robust performances. Compared with the overall performance of supervised DL models, that of unsupervised models often drops largely. Another method is multicondition training (MCT). Yuwana et al. proposed MCT to train more robust DCNNs. They investigated two types of distortion: blurring and rotations. They evaluated the model on a tea disease dataset with 5632 images. The results showed that MCT improved the robustness of DCNN to some extent [59]. Still, another method is persistently enriching the diversity of datasets,

for example through using different geographical locations and cultivation conditions. It is not a simple task, and social work and cooperation are particularly important.

### 3.3. Symptom Variations

When detecting plant diseases, we usually assume that the symptoms of the disease will not change. The symptoms of plant diseases are the results of the interaction of diseases, plants, and the environment [88]. Changes in any one of the three may lead to changes in disease symptoms, as discussed below.

In general, plant disease has the following three variations: (1) at different development stages of the disease, the symptoms shown may be different [73,88]; (2) in the same period, multiple diseases may be observed on the same plant leaves. If multiple diseases are clustered together, then the symptoms may change drastically, which brings difficulty in identifying the types of diseases [88]; (3) similar symptoms may appear among different diseases, which increases the difficulty of disease classification. Meanwhile, the age [89], genotype [90], and healthy tissue color variation (and consequent contrast alterations) [88,91] of the plant itself may cause difficulty in recognizing plant diseases. Other factors, such as temperature, humidity, wind, soil condition, and sunlight, may also alter the symptoms of a specific disease.

The interaction of diseases, plants, and the environment may lead to all kinds of symptom variations, which bring great challenges to image capture and annotation. Two methods can be used to solve this problem:

1. collecting images of specific diseases that contain the entire range of variation [88]; and
2. gradually enriching the diversity of the database in practical applications [73].

The first method is unrealistic because collecting images of the entire range of variation is a very labor-intensive and financially demanding task, and whether researchers have collected variations completely is unclear. The other method is much more realistic, and this method is currently extensively used by researchers to effectively increase the diversity of data.

### 3.4. Image Background

The influence of the picture background on the final classification is unclear. Two situations should be considered. One is that a regularization process is used when collecting images, which generates relatively homogeneous backgrounds. In this case, the background is usually retained. It will not reduce the classification effect and may also improve the classification accuracy. Mohanty et al. used three different versions of the whole PlantVillage dataset (color, grayscaled, and segmented) to identify plant diseases and assess the influence of image background on classification results. The results showed that the performance of the DCNN model using colored images was slightly higher than that of the model using the segmented version of the images [53]. The other situation occurs when images are collected in real-time conditions with a busy background, and some features of the background are similar to the region of interest. Under these circumstances, leaf segmentation technology is needed. Otherwise, the model will also learn the features of the background during training, which will lead to erroneous classification results.

In general, there are five methods that can be used for leaf segmentation. The threshold segmentation technique, which segments the foreground by setting a specific threshold, has a serious disadvantage. Usually, the same threshold is used for all pixels, which may produce incorrect holes or even divide the object into several pieces. This disadvantage will lead to the subsequent process, such as image classification, being harmed [92]. Meanwhile, obtaining a reasonable threshold, which is usually selected by manual work, is difficult. K-means clustering is automatic and works well in most circumstances but is time consuming and unsuitable for high-speed scenes [93]. Otsu, which is an effective and adaptive thresholding method, has been widely used for image segmentation [94]. Although the Otsu method works well with regard to time consumption and is threshold adaptive, it will not produce an appropriate threshold when the gray-level histogram approximates

a unimodal distribution [95]. One more method is DL FCN. FCN is trained pixel to pixel on semantic segmentation to achieve the pixel-level classification of images. If we ignore time and memory limitations, then the FCN method can segment images of any size but has some drawbacks, such as inadequately considering the relationship between pixels [96]. The final segmentation method is watershed segmentation, which is an effective segmentation method. The main drawback of this algorithm is the over-segmentation; three optimized watershed algorithms, namely, hierarchical watershed segmentation, post-merging watershed segmentation, and marker-based watershed segmentation [89], have been proposed to solve this problem. No single segmentation method is suitable for all problems. The combined use of different methods would be a good choice. Gao and Lin proposed a fully automatic segmentation method for medicinal plant leaf images in a complex background. First, they used a vein enhancement and extraction operation to obtain an accurate foreground marker image. Then, the marker-controlled watershed method was used to realize image segmentation. The results of the test experiment showed that the proposed method was better than many other automatic image segmentation methods, such as DL FCV [96].

#### 4. Discussion

Table 5 provides and explains all the necessary information to help readers choose one or more criteria and compare different DL models at a glance. As shown in Table 5, most authors use similar network architectures and thus attain similar experiment results. Accordingly, new tests with more challenging datasets and new leaner DL architectures should be implemented; otherwise, much repetition work will appear.

**Table 5.** Studies on different CNN methods applied to plant leaf disease identification.

No.	Reference	Task	Dataset	Method	Accuracy	Pros and Cons
1	Mohanty et al. (2016) [53]	Identify 14 crop species and 26 diseases	54,306 images from PlantVillage	AlexNet, GoogLeNet	99.35%	Not good for practical application
2	Fuentes et al. (2017) [72]	Detect diseases and pests in tomato plants using images captured in-place by camera devices	5000 images taken under different conditions and scenarios	VGGNet and Residual Network (ResNet)	83% (mean)	Lacking number of samples, the precision would be lower in practical application
3	Chen et al. (2020) [60]	Identify rice and maize leaf diseases	500 images of rice and 466 images of maize	VGGNet, Inception	92%	Future works will focus on deploying the module on mobile devices and applying it to more real-world applications
4	Bin et al. (2017) [83]	Identify four common types of apple leaf diseases (mosaic, rust, brown spot, and <i>Alternaria</i> leaf spot)	13,689 images of diseased apple leaves	A novel architecture based on AlexNet, image generation technique	97.62%	The image generation technique proposed in this paper can enhance the robustness of the convolutional neural network model
5	Brahimi et al. (2017) [67]	Classify nine diseases of tomato leaves	14,828 images	AlexNet, GoogLeNet	99.18%	Lacking number of samples
6	Mishra et al. (2020) [55]	Recognize two corn leaf disease (rust, northern leaf blight)	Some of PlantVillage dataset and some real-time images	DCNN (Deep Convolutional Neural Network)	88.46%	Only two corn diseases are identified and classified, and the dataset is not enough.
7	Darwish et al. (2019) [56]	Diagnose three maize plant diseases	15,408 images from Kaggle	VGG16&19	98.2%	The diversity of dataset is not enough
8	Ferentinos (2018) [11]	Plant disease detection and diagnosis	87,848 images (PlantVillage)	AlexNetOWTBn, VGG	99.53% (best)	Obtaining significantly high success rate
9	Yuwana et al. (2019) [59]	Train more robust deep convolutional neural networks	5632 images of tea	Multicondition training (MCT), AlexNet, GoogLeNet	None-	Only two segmentation methods (blur with kernel size of 5 and rotation of 40°) were used

As for the unique challenge, insufficient datasets or tedious labeling work, besides the methods discussed in Section 3.1, unsupervised and semi-supervised model methods may be a good choice. In the unsupervised models, such as generative adversarial networks (GANs) [97] and variational autoencoders (VAEs) [98], only normal samples are used for training, which solves the problem of difficulty in obtaining disease datasets. The existing few-shot classification studies are mainly based on supervised learning schemes, ignoring the helpful information of unlabeled samples [99]. However, the semi-supervised algorithms use both a few annotated samples and many unannotated samples to train a model and can use unlabeled samples to solve the difficulty of network training in the case of a few labeled samples. Therefore, the use of unsupervised and semi-supervised model methods may be a good research direction in the future.

As for the network design, the models proposed between 2017 and 2021 are slightly different from the earlier ones. They are specially focused on reducing the number of networks parameters [94], designing the networks to be trained with a small database [88], and designing the networks to be trained with field images [100]. Undoubtedly, the trend of designing computationally efficient classification networks will continue to develop in the future [101].

Today, the quick development of intelligent devices, such as smartphones, personal computers, fixed cameras, and UAV, is making image classification projects more convenient and intelligent. He et al. proposed a scheme based on the combination of android clients and servers, which are ubiquitous in our daily lives. The scheme consists of two parts: (1) mobile phone client, through which users can upload the collected images to the server; (2) server-side program, which processes the images and returns the classification results to the user. Meanwhile, the server also needs to store the relevant results in the database to facilitate the query of users [102]. Turui (Beijing, China) Information Technology Co., LTD. (<https://www.mapsharp.com/wzsy>, accessed on 22 July 2021) developed the “Insect Prophet” pest monitoring product. Using the cloud platform, it can easily realize the functions of taking photos to identify pests and counting insects. With the quick development of intelligent devices, the application of deep learning in daily life will become more and more extensive. However, agricultural areas are sometimes far from well-connected regions. Under this circumstance, edge devices and mobile clients, which do not need to send data to the server and can be deployed offline, could be great measures.

Meanwhile, some research shows [103,104] that the electrical signal response produced within plants can be used for real-time detection of plant diseases. Plants perceive the environment by generating electrical signals that essentially represent changes in underlying physiological processes [105]. Under the influence of stress (such as disease), the metabolic activities of various cells and tissues of plants are unstable, which is bound to be reflected in physiological electrical properties. Therefore, the extraction of meaningful features from the generating electrical signals (such as the varying capacitance, conductivity, impedance) and the use of such extracted features [106] would be a good research direction for the classification of plant diseases. For example, Najdenovska et al. used plant electrophysiological signals recorded from 12 tomato plants contaminated with spider mites for an automated classification of the plant’s abnormal state caused by spider mites, and this study got an accuracy of 80% [104].

## 5. Conclusions

DL methods have gained widespread application in plant disease detection and classification. It has solved or partially solved the problems of traditional machine learning methods. DL, which is a branch of machine learning, is mainly used for image classification, target detection, and image segmentation. In this paper, we reviewed the latest CNN networks pertinent to plant leaf disease classification. We introduce the process of CNN methods applied to plant disease classification and summarize DL principles involved in plant disease classification. We also summarize some problems and corresponding solutions of DL used for plant disease classification with extrinsic and intrinsic factors as

listed below: (1) insufficient datasets: transfer learning, data augmentation techniques, citizen science, and data sharing; (2) no-ideal robustness: compressed model, unsupervised DL model, and multicondition training; (3) symptom variations: collecting an entire range of variation and gradually enriching the diversity of dataset; (4) image background: threshold segmentation technique, K-means clustering, Otsu, DL FCN, and watershed segmentation. Furthermore, we discussed the future development direction in plant disease classification, for example, plant electrophysiology and the combination of the mobile phone client and the server-side program would be good future research directions [106]. Such a combination is good for the practice and real-time application of DL methods in plant disease classification.

**Author Contributions:** Conceptualization, J.L. and H.J.; supervision, J.L.; visualization, J.L. and H.J.; writing—original draft preparation, L.T.; writing—review and editing, L.T. and J.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Sichuan Science and Technology Program under grant 2021YFN0020; the Innovation Fund of Postgraduate, Xihua University under grant YCJJ2020041; the Sichuan Science and Technology Program under grant 2019YFN0106; the Key Project of Xihua University under grant DC1900007141, and the National Natural Science Foundation of China under grant 31870347.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- Mukti, I.Z.; Biswas, D. Transfer Learning Based Plant Diseases Detection Using ResNet50. In Proceedings of the 2019 4th International Conference on Electrical Information and Communication Technology (EICT), Khulna, Bangladesh, 20–22 December 2019.
- Arunnehr, J.; Vidhyasagar, B.S.; Anwar Basha, H. Plant Leaf Diseases Recognition Using Convolutional Neural Network and Transfer Learning. In *International Conference on Communication, Computing and Electronics Systems*; Bindhu, V., Chen, J., Tavares, J.M.R.S., Eds.; Springer: Singapore, 2020; pp. 221–229.
- Hughes, D.P.; Salathe, M. An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv* **2015**, arXiv:1511.08060.
- Kianat, J.; Khan, M.A.; Sharif, M.; Akram, T.; Rehman, A.; Saba, T. A joint framework of feature reduction and robust feature selection for cucumber leaf diseases recognition. *Optik* **2021**, *240*, 166566. [[CrossRef](#)]
- Zhang, S.; Zhang, S.; Zhang, C.; Wang, X.; Shi, Y. Cucumber leaf disease identification with global pooling dilated convolutional neural network. *Comput. Electron. Agric.* **2019**, *162*, 422–430. [[CrossRef](#)]
- Agarwal, M.; Gupta, S.; Biswas, K.K. A new conv2d model with modified relu activation function for identification of disease type and severity in cucumber plant. *Sustain. Comput. Inform. Syst.* **2021**, *30*, 100473.
- Chen, J.; Zhang, D.; Zeb, A.; Nanekaran, Y.A. Identification of rice plant diseases using lightweight attention networks. *Expert Syst. Appl.* **2021**, *169*, 114514. [[CrossRef](#)]
- Shrivastava, V.K.; Pradhan, M.K.; Minz, S.; Thakur, M.P. Rice plant disease classification using transfer learning of deep convolution neural network. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2019**, *XLII-3/W6*, 631–635. [[CrossRef](#)]
- Sun, H.; Zhai, L.; Teng, F.; Li, Z.; Zhang, Z. qRgls1. 06, a major QTL conferring resistance to gray leaf spot disease in maize. *Crop. J.* **2021**, *9*, 342–350. [[CrossRef](#)]
- Yu, C.; Ai-hong, Z.; Ai-Jun, R.; Hong-qin, M. Types of Maize virus diseases and progress in virus identification techniques in China. *J. Northeast Agric. Univ.* **2014**, *21*, 75–83. [[CrossRef](#)]
- Ferentinos, K. Deep learning models for plant disease detection and diagnosis. *Comput. Electron. Agric.* **2018**, *145*, 311–318. [[CrossRef](#)]
- Abbas, A.; Jain, S.; Gour, M.; Vankudothu, S. Tomato plant disease detection using transfer learning with C-GAN synthetic images. *Comput. Electron. Agric.* **2021**, *187*, 106279. [[CrossRef](#)]
- Sankaran, S.; Mishra, A.; Ehsani, R.; Davis, C. A review of advanced techniques for detecting plant diseases. *Comput. Electron. Agric.* **2010**, *72*, 1–13. [[CrossRef](#)]
- Barbedo Arnal, J.G. An Automatic Method to Detect and Measure Leaf Disease Symptoms Using Digital Image Processing. *Plant Dis.* **2014**, *98*, 1709–1716. [[CrossRef](#)]

15. Feng, Q.; Dongxia, L.; Bingda, S.; Liu, R.; Zhanhong, M.; Haiguang, W. Identification of Alfalfa Leaf Diseases Using Image Recognition Technology. *PLoS ONE* **2016**, *11*, e168274.
16. Omrani, E.; Khoshnevisan, B.; Shamshirband, S.; Saboohi, H.; Anuar, N.B.; Nasir, M.H.N.M. Potential of radial basis function-based support vector regression for apple disease detection. *Measurement* **2014**, *55*, 512–519. [[CrossRef](#)]
17. Barbedo Arnal, J.G. A new automatic method for disease symptom segmentation in digital photographs of plant leaves. *Eur. J. Plant Pathol.* **2017**, *147*, 349–364. [[CrossRef](#)]
18. Springer. *SVM-Based Detection of Tomato Leaves Diseases*; Springer: Berlin/Heidelberg, Germany, 2015.
19. Rumpf, T.; Mahlein, A.K.; Steiner, U.; Oerke, E.C.; Dehne, H.W.; Plümer, L. Early detection and classification of plant diseases with Support Vector Machines based on hyperspectral reflectance. *Comput. Electron. Agric.* **2010**, *74*, 91–99. [[CrossRef](#)]
20. Hossain, E.; Hossain, M.F.; Rahaman, M.A. A Color and Texture Based Approach for the Detection and Classification of Plant Leaf Disease Using KNN Classifier. In Proceedings of the 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE), Cox's Bazar, Bangladesh, 7–9 February 2019.
21. Mohana, R.M.; Reddy CK, K.; Anisha, P.R.; Murthy, B.R. Random forest algorithms for the classification of tree-based ensemble. *Mater. Today Proc.* **2021**. [[CrossRef](#)]
22. Türkoğlu, M.; Hanbay, D. Plant disease and pest detection using deep learning-based features. *Turk. J. Electr. Eng. Comput. Sci.* **2019**, *27*, 1636–1651. [[CrossRef](#)]
23. Arivazhagan, S.; Shebiah, R.N.; Ananthi, S.; Varthini, S.V. Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture features. *Agric. Eng. Int. CIGR J.* **2013**, *15*, 211–217.
24. Jiang, F.; Lu, Y.; Chen, Y.; Cai, D.; Li, G. Image recognition of four rice leaf diseases based on deep learning and support vector machine. *Comput. Electron. Agric.* **2020**, *179*, 105824. [[CrossRef](#)]
25. Gao, J.; French, A.P.; Pound, M.P.; He, Y.; Pridmore, T.P.; Pieters, J.G. Deep convolutional neural networks for image-based *Convolvulus sepium* detection in sugar beet fields. *Plant Methods* **2020**, *16*, 29. [[CrossRef](#)] [[PubMed](#)]
26. Athanikar, G.; Badar, P. Potato leaf diseases detection and classification system. *Int. J. Comput. Sci. Mob. Comput.* **2016**, *5*, 76–88.
27. Yu, J.; Sharpe, S.M.; Schumann, A.W.; Boyd, N.S. Deep learning for image-based weed detection in turfgrass. *Eur. J. Agron.* **2019**, *104*, 78–84. [[CrossRef](#)]
28. Bansal, P.; Kumar, R.; Kumar, S. Disease Detection in Apple Leaves Using Deep Convolutional Neural Network. *Agriculture* **2021**, *11*, 617. [[CrossRef](#)]
29. Wang, H.; Raj, B. On the origin of deep learning. *arXiv* **2017**, arXiv:1702.07800.
30. Deng, L.; Yu, D. Deep Learning: Methods and Applications. *Found. Trends Signal Process.* **2014**, *7*, 197–387. [[CrossRef](#)]
31. Dyrmann, M.; Karstoft, H.; Midtby, H.S. Plant species classification using deep convolutional neural network. *Biosyst. Eng.* **2016**, *151*, 72–80. [[CrossRef](#)]
32. Kussul, N.; Lavreniuk, M.; Skakun, S.; Shelestov, A. Deep Learning Classification of Land Cover and Crop Types Using Remote Sensing Data. *IEEE Geosci. Remote. Sens. Lett.* **2017**, *14*, 778–782. [[CrossRef](#)]
33. Wason, R. Deep learning: Evolution and expansion. *Cogn. Syst. Res.* **2018**, *52*, 701–708. [[CrossRef](#)]
34. Geetharamani, G.; Pandian, A. Identification of plant leaf diseases using a nine-layer deep convolutional neural network. *Comput. Electr. Eng.* **2019**, *76*, 323–338.
35. Huang, G.; Liu, Z.; Laurens, V.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708. Available online: [https://openaccess.thecvf.com/content\\_cvpr\\_2017/papers/Huang\\_Densely\\_Connected\\_Convolutional\\_CVPR\\_2017\\_paper.pdf](https://openaccess.thecvf.com/content_cvpr_2017/papers/Huang_Densely_Connected_Convolutional_CVPR_2017_paper.pdf) (accessed on 22 July 2021).
36. Too, E.C.; Yujian, L.; Njuki, S.; Yingchun, L. A comparative study of fine-tuning deep learning models for plant disease identification. *Comput. Electron. Agric.* **2018**, *161*, 272–279. [[CrossRef](#)]
37. He, K.; Zhang, X.; Ren, S.; Jian, S. Identity Mappings in Deep Residual Networks. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016.
38. Guan, S.; Kamona, N.; Loew, M. Segmentation of Thermal Breast Images Using Convolutional and Deconvolutional Neural Networks. In Proceedings of the 2018 IEEE Applied Imagery Pattern Recognition Workshop (AIPR), Washington, DC, USA, 9–11 October 2018.
39. Fakhry, A.; Zeng, T.; Ji, S. Residual Deconvolutional Networks for Brain Electron Microscopy Image Segmentation. *IEEE Trans. Med. Imaging* **2017**, *36*, 447–456. [[CrossRef](#)] [[PubMed](#)]
40. Liu, J.; Wang, Y.; Li, Y.; Fu, J.; Li, J.; Lu, H. Collaborative Deconvolutional Neural Networks for Joint Depth Estimation and Semantic Segmentation. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 5655–5666. [[CrossRef](#)] [[PubMed](#)]
41. Wang, J.; Wang, Z.; Tao, D.; See, S.; Wang, G. Learning Common and Specific Features for RGB-D Semantic Segmentation with Deconvolutional Networks. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2016.
42. Gehlot, S.; Gupta, A.; Gupta, R. SDCT-AuxNet: DCT Augmented Stain Deconvolutional CNN with Auxiliary Classifier for Cancer Diagnosis. *Med. Image Anal.* **2020**, *61*, 101661. [[CrossRef](#)]
43. Duggal, R.; Gupta, A.; Gupta, R.; Mallick, P. *SD-Layer: Stain Deconvolutional Layer for CNNs in Medical Microscopic Imaging*; Springer: Cham, Switzerland, 2017.
44. Gu, J.; Wang, Z.; Kuen, J.; Ma, L.; Wang, G. Recent Advances in Convolutional Neural Networks. *Pattern Recognit.* **2015**, *77*, 354–377. [[CrossRef](#)]

45. Krizhevsky, A.; Sutskever, I.; Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [[CrossRef](#)]
46. Gao, Z.; Luo, Z.; Zhang, W.; Lv, Z.; Xu, Y. Deep Learning Application in Plant Stress Imaging: A Review. *AgriEngineering* **2020**, *2*, 430–446. [[CrossRef](#)]
47. Bengio, Y.; Simard, P.; Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Netw.* **1994**, *5*, 157–166. [[CrossRef](#)] [[PubMed](#)]
48. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. *J. Mach. Learn. Res.* **2010**, *9*, 249–256.
49. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *arXiv* **2016**, arXiv:1512.03385.
50. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.
51. Tan, M.; Le, Q.V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *arXiv* **2019**, arXiv:1704.04861.
52. Parvat, A.; Chavan, J.; Kadam, S.; Dev, S.; Pathak, V. A survey of deep-learning frameworks: 2017 International Conference on Inventive Systems and Control (ICISC). In Proceedings of the 2017 International Conference on Inventive Systems and Control (ICISC), Coimbatore, India, 19–20 January 2017.
53. Mohanty, S.P.; Hughes, D.P.; Marcel, S. Using Deep Learning for Image-Based Plant Disease Detection. *Front. Plant Sci.* **2016**, *7*, 1419. [[CrossRef](#)]
54. Brahimi, M.; Mahmoudi, S.; Boukhalifa, K.; Moussaoui, A. Deep interpretable architecture for plant diseases classification. In Proceedings of the 2019 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA), Poznan, Poland, 18–20 September 2019.
55. Mishra, S.; Sachan, R.; Rajpal, D. Deep Convolutional Neural Network based Detection System for Real-time Corn Plant Disease Recognition. *Procedia Comput. Ence* **2020**, *167*, 2003–2010. [[CrossRef](#)]
56. Darwish, A.; Ezzat, D.; Hassanien, A.E. An optimized model based on convolutional neural networks and orthogonal learning particle swarm optimization algorithm for plant diseases diagnosis. *Swarm Evol. Comput.* **2019**, *52*, 100616. [[CrossRef](#)]
57. Amanda, R.; Kelsee, B.; Peter, M.C.; Babuali, A.; James, L.; Hughes, D.P. Deep Learning for Image-Based Cassava Disease Detection. *Front. Plant Sci.* **2017**, *8*, 1852.
58. Lin, Z.; Mu, S.; Shi, A.; Pang, C.; Student, G.; Sun, X.; Student, G. A Novel Method of Maize Leaf Disease Image Identification Based on a Multichannel Convolutional Neural Network. *Trans. ASABE* **2018**, *61*, 1461–1474. [[CrossRef](#)]
59. Yuwana, R.S.; Suryawati, E.; Zilvan, V.; Ramdan, A.; Fauziah, F. Multi-Condition Training on Deep Convolutional Neural Networks for Robust Plant Diseases Detection. In Proceedings of the 2019 International Conference on Computer, Control, Informatics and its Applications (IC3INA), Tangerang, Indonesia, 23–24 October 2019.
60. Chen, J.; Chen, J.; Zhang, D.; Sun, Y.; Nanekaran, Y.A. Using deep transfer learning for image-based plant disease identification. *Comput. Electron. Agric.* **2020**, *173*, 105393. [[CrossRef](#)]
61. Fujita, E.; Kawasaki, Y.; Uga, H.; Kagiwada, S.; Iyatomi, H. Basic investigation on a robust and practical plant diagnostic system. In Proceedings of the 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), Anaheim, CA, USA, 18–20 December 2016.
62. Ghazi, M.M.; Yanikoglu, B.; Aptoula, E. Plant identification using deep neural networks via optimization of transfer learning parameters. *Neurocomputing* **2017**, *235*, 228–235. [[CrossRef](#)]
63. Hidayatuloh, A.; Nursalman, M.; Nugraha, E. Identification of Tomato Plant Diseases by Leaf Image Using Squeezenet Model. In Proceedings of the 2018 International Conference on Information Technology Systems and Innovation (ICITSI), Bandung, Indonesia, 22–26 October 2018.
64. Juncheng, M.; Keming, D.; Feixiang, Z.; Lingxian, Z.; Zhihong, G.; Zhongfu, S. A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. *Comput. Electron. Agric.* **2018**, *154*, 18–24.
65. Bollis, E.; Pedrini, H.; Avila, S. Weakly Supervised Learning Guided by Activation Mapping Applied to a Novel Citrus Pest Benchmark. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020.
66. Ge, M.; Su, F.; Zhao, Z.; Su, D. Deep Learning Analysis on Microscopic Imaging in Materials Science. *Mater. Today Nano* **2020**, *11*, 100087. [[CrossRef](#)]
67. Brahimi, M.; Boukhalifa, K.; Moussaoui, A. Deep Learning for Tomato Diseases: Classification and Symptoms Visualization. *Appl. Artif. Intell.* **2017**, *31*, 1–17. [[CrossRef](#)]
68. Arsenovic, M.; Karanovic, M.; Sladojevic, S.; Anderla, A.; Stefanovic, D. Solving Current Limitations of Deep Learning Based Approaches for Plant Disease Detection. *Symmetry* **2019**, *11*, 939. [[CrossRef](#)]
69. Peng, Y.; Wang, Y. An industrial-grade solution for agricultural image classification tasks. *Comput. Electron. Agric.* **2021**, *187*, 106253. [[CrossRef](#)]
70. Wang, Y.; Wang, J.; Zhang, W.; Zhan, Y.; Guo, S.; Zheng, Q.; Wang, X. A survey on deploying mobile deep learning applications: A systemic and technical perspective. *Digit. Commun. Netw.* **2021**. [[CrossRef](#)]
71. Gao, J.; Westergaard, J.C.; Sundmark, E.H.R.; Bagge, M.; Liljeroth, E.; Alexandersson, E. Automatic late blight lesion recognition and severity quantification based on field imagery of diverse potato genotypes by deep learning. *Knowl. Based Syst.* **2021**, *214*, 106723. [[CrossRef](#)]



72. Fuentes, A.; Yoon, S.; Kim, S.C.; Park, D.S. A Robust Deep-Learning-Based Detector for Real-Time Tomato Plant Diseases and Pests Recognition. *Sensors* **2017**, *17*, 2022. [[CrossRef](#)]
73. Barbedo, J.G.A. Factors influencing the use of deep learning for plant disease recognition. *Biosyst. Eng.* **2018**, *172*, 84–91. [[CrossRef](#)]
74. Coulibaly, S.; Kamsu-Foguem, B.; Kamissoko, D.; Traore, D. Deep neural networks with transfer learning in millet crop images. *Comput. Ind.* **2019**, *108*, 115–120. [[CrossRef](#)]
75. Abdalla, A.; Cen, H.; Wan, L.; Rashid, R.; He, Y. Fine-tuning convolutional neural network with transfer learning for semantic segmentation of ground-level oilseed rape images in a field with high weed pressure. *Comput. Electron. Agric.* **2019**, *167*, 105091. [[CrossRef](#)]
76. Thenmozhi, K.; Reddy, U.S. Crop pest classification based on deep convolutional neural network and transfer learning. *Comput. Electron. Agric.* **2019**, *164*, 104906. [[CrossRef](#)]
77. Suh, H.K.; IJsselmuiden, J.; Hofstee, J.W.; van Henten, E.J. Transfer learning for the classification of sugar beet and volunteer potato under field conditions. *Biosyst. Eng.* **2018**, *174*, 50–65. [[CrossRef](#)]
78. Hendrycks, D.; Mu, N.; Cubuk, E.D.; Zoph, B.; Gilmer, J.; Lakshminarayanan, B. AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty. *arXiv* **2019**, arXiv:1912.02781.
79. Ho, D.; Liang, E.; Stoica, I.; Abbeel, P.; Chen, X. Population Based Augmentation: Efficient Learning of Augmentation Policy Schedules. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 2731–2741.
80. Lim, S.; Kim, I.; Kim, T.; Kim, C.; Kim, S. Fast AutoAugment. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 6665–6675.
81. Cubuk, E.D.; Zoph, B.; Shlens, J.; Le, Q.V. Randaugment: Practical automated data augmentation with a reduced search space. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020.
82. Yun, S.; Han, D.; Chun, S.; Oh, S.J.; Yoo, Y.; Choe, J. CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 6023–6032. Available online: [https://openaccess.thecvf.com/content\\_ICCV\\_2019/papers/Yun\\_CutMix\\_Regularization\\_Strategy\\_to\\_Train\\_Strong\\_Classifiers\\_With\\_Localizable\\_Features\\_ICCV\\_2019\\_paper.pdf](https://openaccess.thecvf.com/content_ICCV_2019/papers/Yun_CutMix_Regularization_Strategy_to_Train_Strong_Classifiers_With_Localizable_Features_ICCV_2019_paper.pdf) (accessed on 22 July 2021).
83. Bin, L.; Yun, Z.; Dongjian, H.; Yuxiang, L. Identification of Apple Leaf Diseases Based on Deep Convolutional Neural Networks. *Symmetry* **2017**, *10*, 11.
84. Douarre, C.; Crispim-Junior, C.F.; Gelibert, A.; Tougne, L.; Rousseau, D. Novel data augmentation strategies to boost supervised segmentation of plant disease. *Comput. Electron. Agric.* **2019**, *165*, 104967. [[CrossRef](#)]
85. Argüeso, D.; Picon, A.; Irusta, U.; Medela, A.; Alvarez-Gila, A. Few-Shot Learning approach for plant disease classification using images taken in the field. *Comput. Electron. Agric.* **2020**, *175*, 105542. [[CrossRef](#)]
86. Arnal Barbedo, J.G. Plant disease identification from individual lesions and spots using deep learning. *Biosyst. Eng.* **2019**, *180*, 96–107. [[CrossRef](#)]
87. Srdjan, S.; Marko, A.; Andras, A.; Dubravko, C.; Darko, S. Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. *Comput. Intell. Neurosci.* **2016**, *2016*, 3289801.
88. Barbedo Arnal, J.G. A review on the main challenges in automatic plant disease identification based on visible range images. *Biosyst. Eng.* **2016**, *144*, 52–60. [[CrossRef](#)]
89. Zhang, H.; Tang, Z.; Xie, Y.; Gao, X.; Chen, Q. A watershed segmentation algorithm based on an optimal marker for bubble size measurement. *Measurement* **2019**, *138*, 182–193. [[CrossRef](#)]
90. Peressotti, E.; Duchene, E.; Merdinoglu, D.; Mestre, P. A semi-automatic non-destructive method to quantify grapevine downy mildew sporulation. *J. Microbiol. Methods* **2011**, *84*, 265–271. [[CrossRef](#)] [[PubMed](#)]
91. Clément, A.; Verfaille, T.; Lormel, C.; Jaloux, B. A new colour vision system to quantify automatically foliar discoloration caused by insect pests feeding on leaf cells. *Biosyst. Eng.* **2015**, *133*, 128–140. [[CrossRef](#)]
92. Ling, Q.; Yan, J.; Li, F.; Zhang, Y. A background modeling and foreground segmentation approach based on the feedback of moving objects in traffic surveillance systems. *Neurocomputing* **2014**, *133*, 32–45. [[CrossRef](#)]
93. Du, H.; Chen, X.; Xi, J. An improved background segmentation algorithm for fringe projection profilometry based on Otsu method. *Opt. Commun.* **2019**, *453*, 124206. [[CrossRef](#)]
94. Singh, B.; Toshniwal, D.; Allur, S.K. Shunt connection: An intelligent skipping of contiguous blocks for optimizing MobileNet-V2. *Neural Netw.* **2019**, *118*, 192–203. [[CrossRef](#)]
95. Xu, X.; Xu, S.; Jin, L.; Song, E. Characteristic analysis of Otsu threshold and its applications. *Pattern Recognit. Lett.* **2011**, *32*, 956–961. [[CrossRef](#)]
96. Gao, L.; Lin, X. Fully automatic segmentation method for medicinal plant leaf images in complex background. *Comput. Electron. Agric.* **2019**, *164*, 104924. [[CrossRef](#)]
97. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks. *Adv. Neural Inf. Process. Syst.* **2014**, *3*, 2672–2680.
98. Gunduz, H. An efficient dimensionality reduction method using filter-based feature selection and variational autoencoders on Parkinson’s disease classification. *Biomed. Signal Process. Control.* **2021**, *66*, 102452. [[CrossRef](#)]

99. Li, Y.; Chao, X. Semi-supervised few-shot learning approach for plant diseases recognition. *Plant Methods* **2021**, *17*, 68. [[CrossRef](#)] [[PubMed](#)]
100. Anami, B.S.; Malvade, N.N.; Palaiah, S. Deep learning approach for recognition and classification of yield affecting paddy crop stresses using field images. *Artif. Intell. Agric.* **2020**, *4*, 12–20. [[CrossRef](#)]
101. Rahman, S.; Wang, L.; Sun, C.; Zhou, L. Deep Learning Based HEp-2 Image Classification: A Comprehensive Review. *Med. Image Anal.* **2020**, *65*, 101764. [[CrossRef](#)] [[PubMed](#)]
102. He, Y.; Zhou, Z.; Tian, L.; Liu, Y.; Luo, X. Brown rice planthopper (*Nilaparvata lugens* Stal) detection based on deep learning. *Precis. Agric.* **2020**, *21*, 1385–1402. [[CrossRef](#)]
103. Shre, K.C. An Approach towards Plant Electrical Signal Based External Stimuli Monitoring System. Ph.D. Thesis, University of Southampton, Southampton, UK, 2017.
104. Najdenovska, E.; Dutoit, F.; Tran, D.; Plummer, C.; Raileanu, L.E. Classification of Plant Electrophysiology Signals for Detection of Spider Mites Infestation in Tomatoes. *Appl. Sci.* **2021**, *11*, 1414. [[CrossRef](#)]
105. Chatterjee, S.K.; Das, S.; Maharatna, K.; Masi, E.; Santopolo, L.; Mancuso, S.; Vitaletti, A. Exploring strategies for classification of external stimuli using statistical features of the plant electrical response. *J. R. Soc. Interface* **2015**, *12*, 20141225. [[CrossRef](#)]
106. Chatterjee, S.K.; Malik, O.; Gupta, S. Chemical Sensing Employing Plant Electrical Signal Response-Classification of Stimuli Using Curve Fitting Coefficients as Features. *Biosensors* **2018**, *8*, 83. [[CrossRef](#)] [[PubMed](#)]