

Review

Plant Disease Detection and Classification by Deep Learning

Muhammad Hammad Saleem ¹, Johan Potgieter ² and Khalid Mahmood Arif ^{1,*} 

¹ Department of Mechanical and Electrical Engineering, School of Food and Advanced Technology, Massey University, Auckland 0632, New Zealand; h.saleem@massey.ac.nz

² Massey Agritech Partnership Research Centre, School of Food and Advanced Technology, Massey University, Palmerston North 4442, New Zealand; j.potgieter@massey.ac.nz

* Correspondence: k.arif@massey.ac.nz

Received: 25 September 2019; Accepted: 29 October 2019; Published: 31 October 2019



Abstract: Plant diseases affect the growth of their respective species, therefore their early identification is very important. Many Machine Learning (ML) models have been employed for the detection and classification of plant diseases but, after the advancements in a subset of ML, that is, Deep Learning (DL), this area of research appears to have great potential in terms of increased accuracy. Many developed/modified DL architectures are implemented along with several visualization techniques to detect and classify the symptoms of plant diseases. Moreover, several performance metrics are used for the evaluation of these architectures/techniques. This review provides a comprehensive explanation of DL models used to visualize various plant diseases. In addition, some research gaps are identified from which to obtain greater transparency for detecting diseases in plants, even before their symptoms appear clearly.

Keywords: plant disease; deep learning; convolutional neural networks (CNN)

1. Introduction

The Deep Learning (DL) approach is a subcategory of Machine Learning (ML), introduced in 1943 [1] when threshold logic was introduced to build a computer model closely resembling the biological pathways of humans. This field of research is still evolving; its evolution can be divided into two time periods—from 1943–2006 and from 2012–until now. During the first phase, several developments like backpropagation [2,3], chain rule [4], Neocognitron [5], hand written text recognition (LeNET architecture) [6], and resolving the training problem [7,8] were observed (as shown in Figure 1). However, in the second phase, state-of-the-art algorithms/architectures were developed for many applications including self-driving cars [9–11], healthcare sector [12–14], text recognition [6,15–17], earthquake predictions [18–20], marketing [21], finance [22,23], and image recognition [24–29]. Among those architectures, AlexNet [30] is considered to be a breakthrough in the field of DL as it won the ImageNet challenge for object recognition known as ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in the year 2012. Soon after, several architectures were introduced to overcome the loopholes observed previously. For the evaluation of these algorithms/architectures, various performance metrics were used. Among these metrics, top-1%/top-5% error [24,26,30,31], precision and recall [25,32–34], F1 score [32,35], training/validation accuracy and loss [34,36], classification accuracy (CA) [37–41] are the most popular. For the implementation of DL models, several steps are required, from the collection of datasets to visualization mappings are explained in Figure 2.

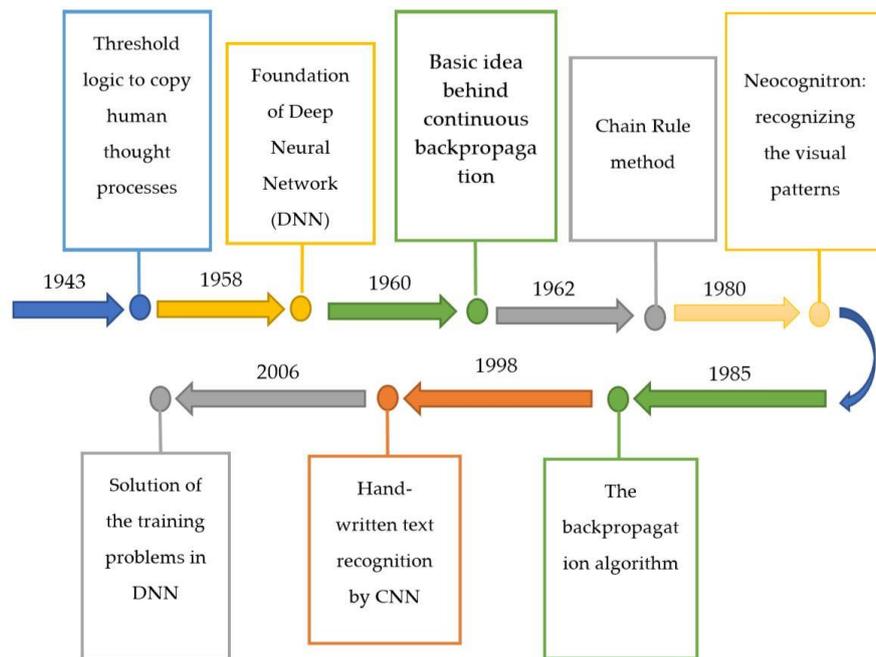


Figure 1. Summary of the evolution of deep learning from 1943–2006.

When DL architectures started to evolve with the passage of time, researchers applied them to image recognition and classification. These architectures have also been implemented for different agricultural applications. For example, in [42], classification of leaves was performed by using author-modified CNN and Random Forest (RF) classifier among 32 species in which the performance was evaluated through CA at 97.3%. On the other hand, it was not as efficient at detecting occluded objects [43]. Leaf and fruit counting were also performed by deep CNN in [44,45] and [46] respectively. For classification of crop type, [47] used author-modified CNN, [36] applied VGG 16, [34] implemented three unit LSTM, and [33] used CNN and RGB histogram technique. [47] used CA, [36] used CA and Intersection over Union (IoU), [34] used CA and F1, and [33] used F1-score as a performance metric. Among them, [33,47] did not provide training/validation accuracy and loss. Moreover, recognition of different plants has been done by the DL approach in [48–50]. [48,50] employed user-modified CNN while [49] used AlexNet architecture. All were evaluated on the basis of CA. [49] outperformed the other two in terms of CA. Similarly, crop/weed discrimination was performed in [51,52], in which the author proposed CNN be used, and two datasets were utilized for the evaluation of the model. [51] evaluated precision and recall; however, [52] obtained CA for the validation of the proposed models respectively. The identification of plants by the DL approach was studied and achieved a success rate of 91.78% [53]. On top of that, DL approaches are also used for critical tasks like plant disease detection and classification, which is the main focus of this review. There are some research papers previously presented to summarize the research based on agriculture (including plant disease recognition) by DL [43,54], but they lacked some of the recent developments in terms of visualization techniques implemented along with the DL and modified/cascaded version of famous DL models, which were used for plant disease identification. Moreover, this review also provides the research gaps in order to get a clearer/more transparent vision of symptoms observed due to diseases in the plants.

The remaining part of the paper is comprised of Section 2, describing the famous and new/modified DL architectures along with visualization mapping/techniques used for plant disease detection; Section 3, elaborating upon the Hyperspectral Imaging with DL models; and finally, Section 4, concluding the review and providing future recommendations for achieving more advancements in the visualization, detection, and classification of plants' diseases.

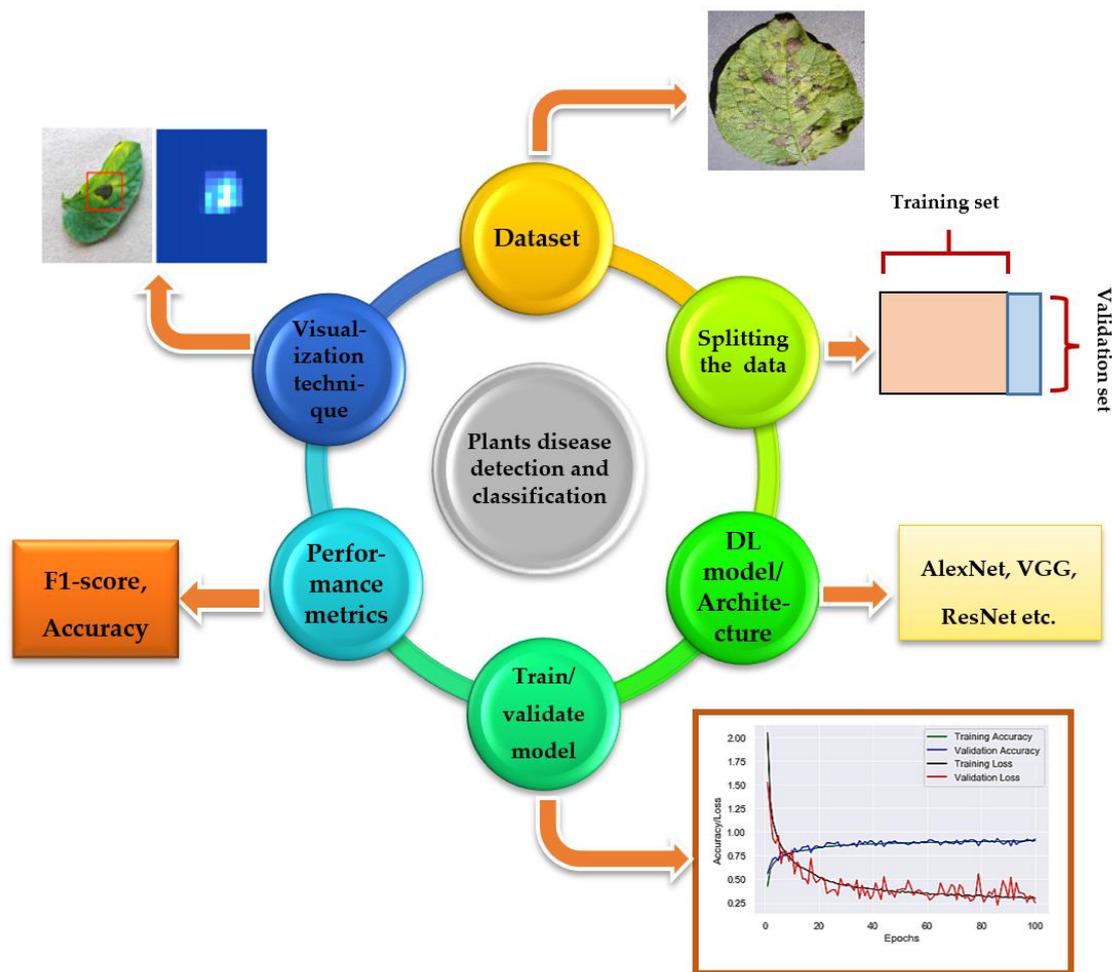


Figure 2. Flow diagram of DL implementation: First, the dataset is collected [25] then split into two parts, normally into 80% of training and 20% of validation set. After that, DL models are trained from scratch or by using transfer learning technique, and their training/validation plots are obtained to indicate the significance of the models. Then, performance metrics are used for the classification of images (type of particular plant disease), and finally, visualization techniques/mappings [55] are used to detect/localize/classify the images.

2. Plant Disease Detection by Well-Known DL Architectures

Many state-of-the-art DL models/architectures evolved after the introduction of AlexNet [30] (as shown in Figure 3 and Table 1) for image detection, segmentation, and classification. This section presents the researches done by using famous DL architectures for the identification and classification of plants' diseases. Moreover, there are some related works in which new visualization techniques and modified/improved versions of DL architectures were introduced to achieve better results. Among all of them, the PlantVillage dataset has been used widely as it contains 54,306 images of 14 different crops having 26 plant diseases [25]. Moreover, they used several performance metrics to evaluate the selected DL models, which are described as below.

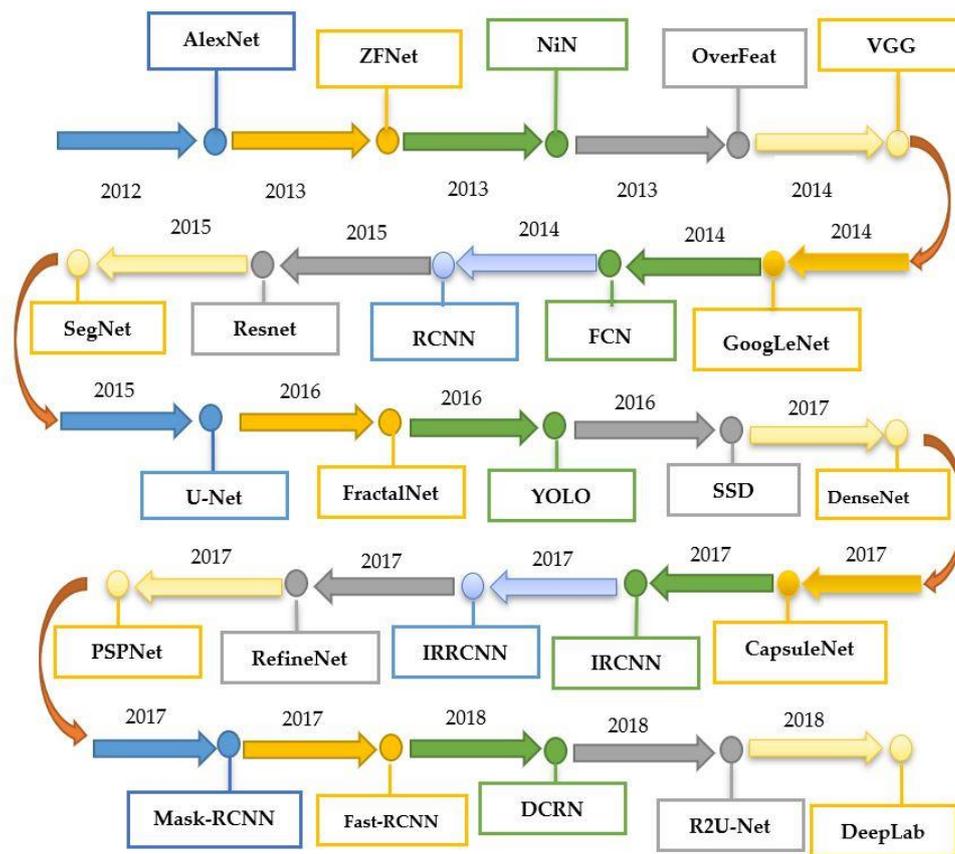


Figure 3. Summary of the evolution of various deep learning models from 2012 until now.

2.1. Implementation of DL Models

2.1.1. Without Visualization Technique

In [56], CNN was used for the classification of diseases in maize plants and histogram techniques to show the significance of the model. In [57], basic CNN architectures like AlexNet, GoogLeNet and ResNet were implemented for identifying the tomato leaf diseases. Training/validation accuracy were plotted to show the performance of the model; ResNet was considered as the best among all the CNN architectures. In order to detect the diseases in banana leaf, LeNet architecture was implemented and CA, F1-score were used for the evaluation of the model in Color and Gray Scale modes [32]. Five CNN architectures were used in [58], namely, AlexNet, AlexNetOWTbn, GoogLeNet, Overfeat, and VGG architectures in which VGG outclassed all the other models. In [35], eight different plant diseases were recognized by three classifiers, Support Vector Machines (SVM), Extreme Learning Machine (ELM), and K-Nearest Neighbor (KNN), used with the state-of-the-art DL models like GoogLeNet, ResNet-50, ResNet-101, Inception-v3, InceptionResNetv2, and SqueezeNet. A comparison was made between those models, and ResNet-50 with SVM classifier got the best results in terms of performance metrics like sensitivity, specificity, and F1-score. According to [59], a new DL model—Inception-v3—was used for the detection of cassava disease. In [60], plant diseases in cucumber were classified by the two basic versions of CNN and got the highest accuracy, equal to 0.823. The traditional plant disease recognition and classification method was replaced by Super-Resolution Convolutional Neural Network (SRCNN) in [61]. For the classification of tomato plant disease, AlexNet and SqueezeNet v1.1 models were used in which AlexNet was found to be the better DL model in terms of accuracy [62]. A comparative analysis was presented in [63] to select the best DL architecture for detection of plant diseases. Moreover in [64], six tomato plant diseases were classified by using AlexNet and VGG-16 DL architectures, and a detailed comparison was provided with the help of classification accuracy. In

the above approaches, no visualization technique was applied to spot the symptoms of diseases in the plants.

Table 1. Comparison of state-of-the-art deep learning models.

Deep Learning Models	Parameters	Key Features and Pros/Cons
LeNet	60k	First CNN model. Few parameters as compared to other CNN models. Limited capability of computation
AlexNet	60M	Known as the first modern CNN. Best image recognition performance at its time. Used ReLU to achieve better performance. Dropout technique was used to avoid overfitting
OverFeat	145M	First model used for detection, localization, and classification of objects through a single CNN. Large number of parameters as compared to AlexNet
ZFNet	42.6M	Reduced weights (as compared to AlexNet) by considering 7×7 kernels and improved accuracy
VGG	133M–144M	3×3 receptive fields were considered to include more number of non-linearity functions which made decision function discriminative. Computationally expensive model due to large number of parameters
GoogLeNet	7M	Fewer number of parameters as compared to AlexNet model. Better accuracy at its time
ResNet	25.5M	Vanishing gradient problem was addressed. Better accuracy than VGG and GoogLeNet models
DenseNet	7.1M	Dense connections between the layers. Reduced number of parameters with better accuracy
SqueezeNet	1.25M	Similar accuracy as AlexNet with 50 times lesser parameters. Considered 1×1 filters instead of 3×3 filters. Input channels were decreased. Large activation maps of convolution layers
Xception	22.8M	A depth-wise separable convolution approach. Performed better than VGG, ResNet, and Inception-v3 models
MobileNet	4.2M	Considered the depth-wise separable convolution concept. Reduced parameters significantly. Achieved accuracy near to VGG and GoogLeNet
Modified/Reduced MobileNet	0.5/0.54M	Lesser number of parameters as compared to MobileNet. Similar accuracy as compared to MobileNet
VGG-Inception	132M	A cascaded version of VGG and inception module. The number of parameters were reduced by substituting 5×5 convolution layers with two 3×3 layers. Testing accuracy was increased as compared to many well-known DL models like AlexNet, GoogLeNet, Inception-v3, ResNet, and VGG-16.

2.1.2. With Visualization Techniques

The following approaches employed DL models/architectures and also visualization techniques which were introduced for a clearer understanding of plants' diseases. For example, [55] introduced the saliency map for visualizing the symptoms of plant disease; [27] identified 13 different types of plant disease with the help of CaffeNet CNN architecture, and achieved CA equal to 96.30%, which was better than the previous approach like SVM. Moreover, several filters were used to indicate the disease spots. Similarly, [25] used AlexNet and GoogLeNet CNN architectures by using the publicly available PlantVillage dataset. The performance was evaluated by means of precision (P), recall (R), F1 score, and overall accuracy. The uniqueness of this paper was the implication of three scenarios (color, grayscale, and segmented) for evaluating the performance metrics and comparison of the two

famous CNN architectures. It was concluded that GoogLeNet outperformed AlexNet. Moreover, visualization activation in the first layers clearly showed the spots of diseases. In [65], a modified LeNet model was used to detect olive plant diseases. The segmentation and edges maps were used to spot the diseases in the plants. Detection of four cucumber diseases was done in [66] and accuracy was compared with Random Forest, Support Vector Machines, and AlexNet models. Moreover, the image segmentation method was used to view the symptoms of diseases in the plants. A new DL model was introduced in [67] named teacher/student network and proposed a novel visualization method to identify the spots of plant diseases. DL models with some detectors were implemented in [68], in which the diseases in plants were marked along with their prediction percentage. Three detectors, named Faster-RCNN, RFCN and SSD, were used with the famous architectures like AlexNet, GoogLeNet, VGG, ZFNet, ResNet-50, ResNet-101 and ResNetXt-101 for a comparative study which outlined the best among all the selected architectures. It was concluded that ResNet-50 with the detector R-FCN gave the best results. Furthermore, a kind of bounding box was drawn to identify the particular type of disease in the plants. In [69], a banana leaf disease and pest detection was performed by using three CNN models (ResNet-50, Inception-V2 and MobileNet-V1) with Faster-RCNN and SSD detectors. According to [70], different combinations of CNN were used and presented heat maps as input to the diseased plants' images and provided the probability related to the occurrence of a particular type of disease. Moreover, ROC curve evaluates the performance of the model. Furthermore, feature maps for rice disease were also included in the paper. LeNet model was used in [71] to detect and classify diseases in the soybean plant. In [72], a comparison between AlexNet and GoogLeNet architectures for tomato plant diseases was done, in which GoogLeNet performed better than the AlexNet; also, it proposed occlusion techniques to recognize the regions of diseases. The VGG-FCN and VGG-CNN models were implemented in [73], for the detection of wheat plant diseases and visualization of features in each block. In [74], VGG-CNN model was used for the detection of Fusarium wilt in radish and K-means clustering method was used to show the marks of diseases. A semantic segmentation approach by CNN was proposed in [75] to detect the disease in cucumber. In [76], an approach based on the individual symptoms/spots of diseases in the plants was introduced by using a DL model for detecting plant diseases. A Deep CNN framework was developed for identification, classification, and quantification of eight soybean stresses in [77]. In [78], rice plant diseases were identified by CNN, and feature maps were obtained to identify the patches of diseases. A deep residual neural network was extended in [79] for the development of a mobile application in which a clear identification of diseases in plants was done by the hot spot. An algorithm based on the hot spot technique was also used in [80], in which those spots were extracted by modification in the segmented image to attain color constancy. Furthermore, each obtained hot-spot was described by two descriptors, one was used to evaluate the color information of the disease and other was used to identify the texture of the hot-spots. The cucumber plant diseases were identified in [81] by using the dilation convolutional neural network. A state-of-the-art visualization technique was proposed in [82] by correlation coefficient and DL models like AlexNet and VGG-16 architectures. In [83], color space and various vegetation indices combined with CNN model (LeNet) to detect the diseases in grapes. To summarize, Table 2 outlines some of the visualization mapping/techniques.

Table 2. Visualization mapping/techniques used in several approaches.

Visualization Techniques/Mappings	References
Visualization of features having filter from first to final layer	[27]
Visualize activations in first convolutional layer	[25]
Saliency map visualization	[55]
Classification and localization of diseases by bounding boxes	[68]
Heat maps were used to identify the spots of the disease	[70]
Feature map for the diseased rice plant	[78]
Symptoms visualization method	[72]
Feature and spatial core maps	[73]
Color space into HSV and K-means clustering	[74]
Feature map for spotting the diseases	[77]
Image segmentation method	[66]
Reconstruction of images on discriminant regions, segmentation of images by binary threshold theorem, and heat map construction	[67]
Saliency map visualization	[84]
Saliency map, 2D and 3D contour, mesh graph image	[82]
Activation visualization	[85]
Segmentation map and edge map	[65]

For the practical experimentation of detection of plants' diseases, an actual/real background/environment should be considered in order to evaluate the performance of the DL model more accurately. In most of the above approaches, the selected datasets considered plain backgrounds which are not realistic scenarios for identification and classification of the diseases [25,27,32,56–58,60,61,65,72,77,78], except for a few of them that have considered the original backgrounds [35,59,68,70,73,74]. The output of the visualization techniques used in several researches are shown in Figures 4–11.

In Figure 4, feature maps from the first to the fifth hidden layer are shown as the neuron in a feature map having identical features at different positions of an image. Starting from the first layer (a), the features in feature maps represent separate pixels to normal lines, whereas the fifth layer shows some particular parts of the image (h).

Two types of visualization maps are shown in Figure 5, namely, heat map and saliency map techniques. The heat maps identify the diseases shown as red boxes in the input image, but it should be noted that one disease marked in (d) has not been detected. This problem was resolved in the saliency map technique after the application of the guided back-propagation [55]; all the spots of plant disease were successfully identified thanks to a method which is superior to the heat map.

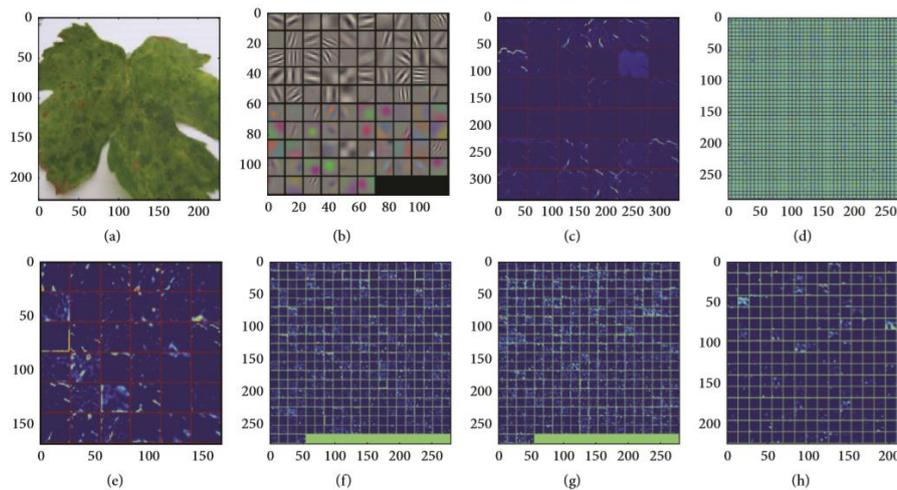


Figure 4. Feature maps after the application of convolution to an image: (a) real image, (b) first convolutional layer filter, (c) rectified output from first layer, (d) second convolutional layer filter, (e) output from second layer, (f) output of third layer, (g) output of fourth layer, (h) output of fifth layer [27].

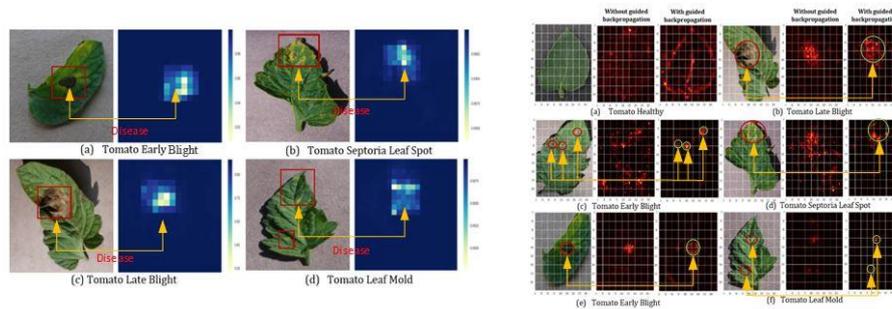


Figure 5. Tomato plant disease detection by heat map: on left hand side (a) tomato early blight, (b) tomato septoria leaf spot, (c) tomato late blight and (d) tomato leaf mold) and saliency map; on right hand side (a) tomato healthy, (b) tomato late blight, (c) tomato early blight, (d) tomato septoria leaf spot, (e) tomato early blight, (f) tomato leaf mold) [55].

Figure 6 represents the heat map to detect the disease in maize plants. First, the image was represented in the form of the probability of each portion containing disease. Then, the probabilities were placed into the form of a matrix in order to denote the outcome of all the areas of the input image.

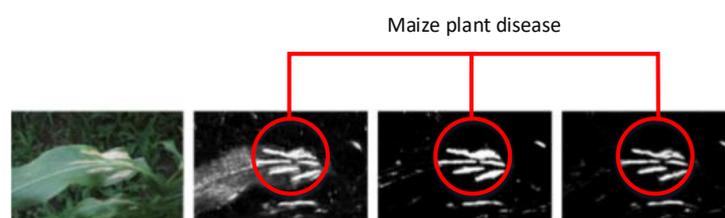


Figure 6. Detection of maize disease (indicated by red circles) by heat map [70].

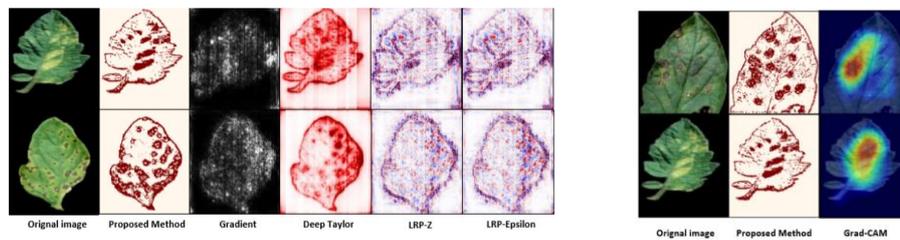


Figure 9. Comparison of Teacher/student approach visualization map with the previous approaches [67].

In order to find the significance of CNN architectures to differentiate between various diseases of plants, the feature maps were obtained as shown in Figure 10. The result proves a good performance of the proposed CNN model as it clearly identifies the disease in plants [85].

In Figure 11 the segmentation and edged maps were obtained to identify the diseases in plants. It is noted that the yellow colored area is marked as white surface in the segmentation map to show the affected part of the leaf.

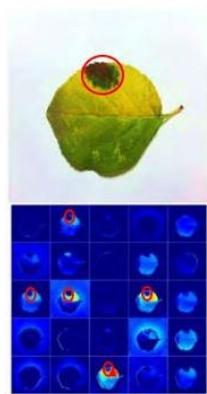


Figure 10. Activation visualization for detection of apple plant disease to show the significance of a VGG-Inception model (the plant disease is indicated by the red circle) [85].



Figure 11. Segmentation and edge map for olive leaf disease detection [65].

2.2. New/Modified DL Architectures for Plant-Disease Detection

According to some of the research papers, new/modified DL architectures have been introduced to obtain better/transparent detection of plant disease, such as [86] presented improved GoogLeNet and Cifar-10 models and their performance compared with AlexNet and VGG. It was found that improved versions of these state-of-the-art models produced a remarkable accuracy of 98.9%. In [87], a new DL model was introduced to obtain more accurate detection of plant diseases as compared to SVM, AlexNet, GoogLeNet, ResNet-20, and VGG-16 models. This model achieved 97.62% accuracy for classifying apple plant diseases. Moreover, the dataset extended in 13 different ways (rotation of 90°, 180°, 270° and mirror symmetry (horizontal symmetry), change in contrast, sharpness and brightness). Moreover, the whole dataset was transformed into Gaussian noise and PCA jittering as well. Furthermore, the selection of dataset was explained by the help of plots to prove the significance

of extending the dataset. A new CNN model named LeafNet was introduced in [88] to classify the tea leaf diseases and achieved higher accuracy than Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP). In [89], two DL models named modified MobileNet and reduced MobileNet were introduced, and their accuracy was near to the VGG model; the reduced MobileNet actually got 98.34% classification accuracy and had a fewer number of parameters as compared to VGG which saves time in training the model. A state-of-the-art DL model was proposed in [90] named PlantdiseaseNet which was remarkably suitable for the complex environment of an agricultural field. In [85], five types of apple plant diseases were classified and detected by the state-of-the-art CNN model named VGG-inception architecture. It outclassed the performance of many DL architectures like AlexNet, GoogLeNet, several versions of ResNet, and VGG. It also presented inter object/class detection and activation visualization; it was also mentioned for its clear vision of diseases in the plants.

A bar chart presented in Figure 12 indicates, from the most to the least frequently used, DL models for plant disease detection and classification. It can be clearly seen that the AlexNet model has been used in most of the researches. GoogLeNet, VGG-16, and ResNet-50 are the next most commonly used DL models. Similarly, there are some improved/cascaded versions (Improved Cifar-10, VGG-Inception, Cascaded AlexNet with GoogLeNet, reduced/modified MobileNet, modified LeNet, and modified GoogLeNet), which have been used for plant disease identification.

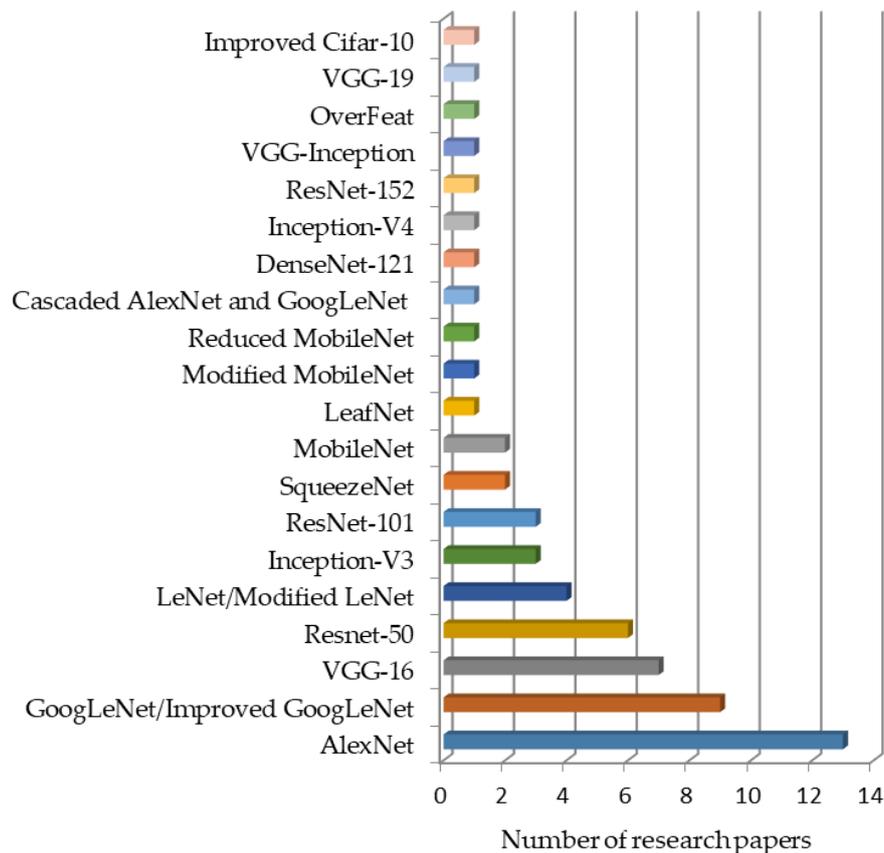


Figure 12. Deep learning models used in the particular number of research papers.

Summing up Section 2, all the DL approaches along with the selected plant species and performance metrics are shown in Table 3.

3. Hyper-Spectral Imaging with DL Models

For early detection of plant diseases, several imaging techniques like multispectral imaging [91], thermal imaging, fluorescence and hyperspectral imaging are used [92]. Among them, hyperspectral

imaging (HSI) is the focus of recent research. For example, [93] used hyperspectral imaging (HSI) to detect tomato plant diseases by identifying the region of interest, and a feature ranking-KNN (FR-KNN) model produced a satisfactory result for the detection of diseased and healthy plants. In the recent approach, HSI was used for the detection of an apple disease. Moreover, the redundancy issue was resolved by an unsupervised feature selection procedure known as Orthogonal Subspace Projection [94]. In [95], leaf diseases on peanuts were detected by HSI by identifying sensitive bands and hyperspectral vegetation index. The tomato disease detection was done by SVM classifiers based on HSI, and their performance was evaluated by F1-score, accuracy, specificity, and sensitivity [96].

Recently, HSI has been used with machine learning (ML) for the detection of plant diseases. For example, [97] described ML techniques for hyperspectral imaging for many agricultural applications. Moreover, ML with HSI have been used for three ML models, implemented by using hyperspectral measurement technique for the detection of leaf rust disease [98]. For wheat disease detection, [99] used Random Forest (RF) classifier with multispectral imaging technique and achieved accuracy of 89.3%. Plants' diseases were also detected by SVM based on hyperspectral data and achieved accuracy of more than 86% [100]. There are some other ML approaches based on HSI [101], but this review is focused on DL approaches based on HSI, presented below.

The DL has been used to classify the hyperspectral images for many applications. For medical purposes, this technology is very useful as it is used for the classification of head/neck cancer in [102]. In [103], a DL approach based on HSI was proposed through contextual information as it provides spectral and spatial features. A new 3D-CNN architecture allowed for a fast, accurate, and efficient approach to classify the hyperspectral images in [104]. This architecture not only used the spectral information (as used in previous CNN techniques [105]) but also ensured that the spatial information was also taken into account. In [106], the feature extraction procedure was used with CNN for hyperspectral image classification and used dropout and L2 regularization methods in order to prevent overfitting. Just like CNN models used for hyperspectral imaging classification, RNN models are also used with HSI as described in [107,108]. In the domain of plant disease detection, some researches utilized Hyperspectral Imaging (HSI) along with DL models to observe clearer vision for symptoms of plant diseases. A hybrid method to classify the hyperspectral images was proposed in [109] consisting of DCNN, LR, and PCA and got better results compared to the previous methods for classification tasks. In [110], a detailed review of DL with HSI technique was provided. In order to avoid the overfitting and improve accuracy, a detailed comparison provided between several DL models like 1D/2D-CNN (2D-CNN better result), LSTM/GRU (both faced overfitting), 2D-CNN-LSTM/GRU (still overfitting) was observed. Therefore, a new hybrid approach from Convolutional and Bidirectional Gated Recurrent Network named 2D-CNN-BidLSTM/GRU was proposed for the hyperspectral images, which resolved the problem of overfitting and achieved 0.75 F1-score and 0.73 accuracy for wheat diseases detection [111]. According to [112], a hyperspectral proximal-sensing procedure based on the newest DL technique named Generative Adversarial Nets (GAN) was proposed in order to detect tomato plant disease before its clear symptoms appeared (as shown in Figure 13). In [84], a 3D-CNN approach was proposed for hyperspectral images to identify the Charcoal rot disease in soybeans and the CNN model was evaluated by accuracy (95.76%) and F1-score (0.87). The saliency map visualization was used, and the most delicate wavelength resulted as 733 nm, which approximately lies in the region of the wavelength of NIR. For the detection of potato virus, [113] described it by DL on the hyperspectral images and achieved acceptable values of precision (0.78) and recall (0.88). In [114], a DL model named multiple Inception-Resnet model was developed by using both spatial and spectral data on hyperspectral UAV images to detect the yellow rust in wheat (as shown in Figure 14). This model achieved an 85% accuracy, which is quite a lot higher than the RF-classifier (77%).

Table 3. Comparison of several DL approaches in terms of various performance metrics.

DL Architectures/Algorithms	Datasets	Selected Plant/s	Performance Metrics (and Their Results)	Refs
CNN	PlantVillage	Maize	CA (92.85%)	[56]
AlexNet, GoogLeNet, ResNet	PlantVillage	Tomato	CA by ResNet which gave the best value (97.28%)	[57]
LeNet	PlantVillage	Banana	CA (98.61%), F1 (98.64%)	[32]
AlexNet, ALexNetOWTBn, GoogLeNet, Overfeat, VGG	PlantVillage and in-field images	Apple, blueberry, banana, cabbage, cassava, cantaloupe, celery, cherry, cucumber, corn, eggplant, gourd, grape, orange, onion	Success rate of VGG (99.53%) which is the best among all	[58]
AlexNet, VGG16, VGG 19, SqueezeNet, GoogLeNet, Inceptionv3, InceptionResNetv2, ResNet50, Resnet101 Inceptionv3	Real field dataset	Apricot, Walnut, Peach, Cherry	F1(97.14), Accuracy (97.86 ± 1.56) of ResNet	[35]
CNN	Experimental field dataset	Cassava	CA (93%)	[59]
Super-Resolution Convolutional Neural Network (SCRNN)	Images taken from the research center	Cucumber	CA (82.3%)	[60]
CaffeNet	PlantVillage	Tomato	Accuracy (~90%)	[61]
AlexNet and GoogLeNet	Downloaded from the internet	Pear, cherry, peach, apple, grapevine	Precision (96.3%)	[27]
AlexNet, GoogLeNet, VGG- 16, ResNet-50,101, ResNetXt-101, Faster RCNN, SSD, R-FCN, ZFNet	PlantVillage	Apple, blueberry, bell pepper, cherry, corn, peach, grape, raspberry, potato, squash, soybean, strawberry, tomato	CA (99.35%) of GoogLeNet	[25]
CNN DCNN	Image taken in real fields	Tomato	Precision (85.98%) of ResNet-50 with Region based Fully Convolutional Network(R-FCN)	[68]
CNN	Bisque platform of Cy Verse	Maize	Accuracy (96.7%)	[70]
DCNN	Images were taken in real field	Rice	Accuracy (95.48%)	[78]
AlexNet, GoogLeNet	PlantVillage	Tomato	Accuracy (0.9918 ± 0.169) of GoogLeNet	[72]
VGG-FCN-VD16 and VGG-FCN-S	Wheat Disease Database 2017	Wheat	Accuracy (97.95%) of VGG-FCN-VD16	[73]
VGG-A, CNN	Images were taken in real field	Radish	Accuracy (93.3%)	[74]
AlexNet	Images were taken in real field	Soybean	CA (94.13%)	[77]
AlexNet and SqueezeNet v1.1	PlantVillage	Tomato	CA (95.65%) of AlexNet	[62]
DCNN, Random forest, Support Vector Machine and AlexNet	PlantVillage dataset, Forestry Image dataset and agricultural field in China	Cucumber	CA (93.4%) of DCNN	[66]

Table 3. Cont.

DL Architectures/Algorithms	Datasets	Selected Plant/s	Performance Metrics (and Their Results)	Refs
Teacher/student architecture	PlantVillage	Apple, bell pepper, blueberry, cherry, corn, orange, grape, potato, raspberry, peach, soybean, strawberry, tomato, squash	Training accuracy and loss (~99%, ~0–0.5%), validation accuracy and loss (~95%, ~10%)	[67]
Improved GoogLeNet, Cifar-10	PlantVillage and various websites	Maize	Top-1 accuracy (98.9%) of improved GoogLeNet	[86]
MobileNet, Modified MobileNet, Reduced MobileNet	PlantVillage dataset	24 types of plant	CA (98.34%) of reduced MobileNet	[89]
VGG-16, ResNet-50,101,152, Inception-V4 and DenseNets-121	PlantVillage	Apple, bell pepper, blueberry, cherry, corn, orange, grape, potato, raspberry, peach, soybean, strawberry, tomato, squash	Testing accuracy (99.75%) of DenseNets	[63]
User defined CNN, SVM, AlexNet, GoogLeNet, ResNet-20 and VGG-16	Images were taken in real field	Apple	CA (97.62%) of proposed CNN	[87]
AlexNet and VGG-16	PlantVillage	Tomato	CA (AlexNet)	[64]
LeafNet, SVM, MLP	Images were taken in real field	Tea leaf	CA (90.16%) of LeafNet	[88]
2D-CNN-BidGRU	Real wheat field	wheat	F1 (0.75) and accuracy (0.743)	[111]
OR-AC-GAN	Real environment	Tomato	Accuracy (96.25%)	[112]
3D CNN	Real environment	Soybean	CA (95.73%), F1-score (0.87)	[84]
DCNN	Real environment	Wheat	Accuracy (85%)	[114]
ResNet-50	Real environment	Wheat	Balanced Accuracy (87%)	[79]
GPDCNN	Real environment	Cucumber	CA (94.65%)	[81]
VGG-16, AlexNet	PlantVillage, CASC-IFW	Apple, banana	CA (98.6%)	[82]
LeNet	Real environment	Grapes	CA (95.8%)	[83]
PlantDiseaseNet	Real environment	Apple, bell-pepper, cherry, grapes, onion, peach, potato, plum, strawberry, sugar-beets, tomato, wheat	CA (93.67%)	[90]
LeNet	PlantVillage	Soybean	CA (99.32%)	[71]
VGG-Inception	Real environment	Apple	Mean average accuracy (78.8%)	[85]
Resnet-50, Inception-V2, MobileNet-V1	Real environment	Banana	Mean average accuracy (99%) of ResNet-50	[69]
Modified LeNet	PlantVillage	Olives	True positive rate (98.6 ± 1.47%)	[65]

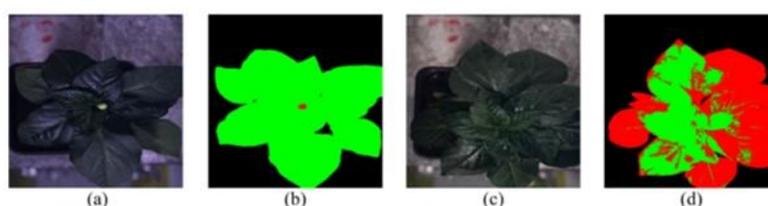


Figure 13. Sample images of OR-AC-GAN (a hyperspectral imaging model) [112].

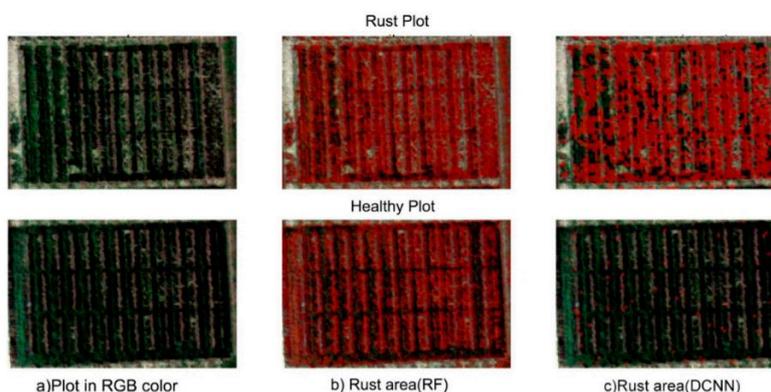


Figure 14. Hyperspectral images by UAV: (a) RGB color plots, (b) Random-Forest classifier, and (c) proposed multiple Inception-ResNet model [114].

From this section, we can conclude that, although there are some DL models/architectures developed for hyperspectral image classification in the application of plant disease detection, this is still a fertile area of research and should lead to improvements for better detection of plants' diseases [115] in different situations, like various conditions of illumination, considering real background, etc.

In Figure 13, the resultant images are taken from the proposed method described in [112]. The green-colored portion indicates the healthy part of the plant; the red portion denotes the infected portion. Note that (a) and (b) are the healthy plant images as there is no red color indication, whereas (c) has infected disease which can be seen in its corresponding figure (d).

A comparison of proposed DCNN with RF classifier and RGB colored hyperspectral images are shown in Figure 14. The red color label indicates the portion infected by rust. It should be observed that the rust plots were identified in an almost similar manner (see (b) and (c) of first row), but in the healthy plot, there was a large portion covered by the red label in (b) as compared to (c), which shows a wrong classification by RF model [114].

4. Conclusions and Future Directions

This review explained DL approaches for the detection of plant diseases. Moreover, many visualization techniques/mappings were summarized to recognize the symptoms of diseases. Although much significant progress was observed during the last three to four years, there are still some research gaps which are described below:

- In most of the researches (as described in the previous sections), the PlantVillage dataset was used to evaluate the accuracy and performance of the respective DL models/architectures. Although this dataset has a lot of images of several plant species with their diseases, it has a simple/plain background. However, for a practical scenario, the real environment should be considered.
- Hyperspectral/multispectral imaging is an emerging technology and has been used in many areas of research (as described in Section 3). Therefore, it should be used with the efficient DL architectures to detect the plants' diseases even before their symptoms are clearly apparent.
- A more efficient way of visualizing the spots of disease in plants should be introduced as it will save costs by avoiding the unnecessary application of fungicide/pesticide/herbicide.

- The severity of plant diseases changes with the passage of time, therefore, DL models should be improved/modified to enable them to detect and classify diseases during their complete cycle of occurrence.
- DL model/architecture should be efficient for many illumination conditions, so the datasets should not only indicate the real environment but also contain images taken in different field scenarios.
- A comprehensive study is required to understand the factors affecting the detection of plant diseases, like the classes and size of datasets, learning rate, illumination, and the like.

Author Contributions: Conceptualization, M.H.S. and K.M.A.; methodology, M.H.S. and K.M.A.; writing—original draft preparation, M.H.S. and K.M.A.; writing—review and editing, M.H.S., J.P., and K.M.A.; visualization, M.H.S., J.P., and K.M.A.; supervision, J.P., and K.M.A.; project administration, J.P., and K.M.A.

Funding: This research was funded by the Ministry of Business, Innovation and Employment (MBIE), New Zealand, Science for Technological Innovation (SfTI) National Science Challenge.

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

Abbreviations

The abbreviations used in this manuscript are given as under:

ML	Machine Learning
DL	Deep Learning
CNN	Convolutional Neural network
DCNN	Deep Convolutional Neural Network
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
RF	Random Forest
CA	Classification Accuracy
LSTM	Long Short-Term Memory
IoU	Intersection of Union
NiN	Network in Network
RCN	Region based Convolutional Neural Network
FCN	Fully Convolutional Neural Network
YOLO	You Only Look Once
SSD	Single Shot Detector
PSPNet	Pyramid Scene Parsing Network
IRRCNN	Inception Recurrent Residual Convolutional Neural Network
IRCNN	Inception Recurrent Convolutional Neural Network
DCRN	Densely Connected Recurrent Convolutional Network
INAR-SSD	Single Shot Detector with Inception module and Rainbow concatenation
R2U-Net	Recurrent Residual Convolutional Neural Network based on U-Net model
SVM	Support Vector Machines
ELM	Extreme Learning Machine
KNN	K-Nearest Neighbor
SRCNN	Super-Resolution Convolutional Neural Network
R-FCN	Region-based Fully Convolutional Networks
ROC	Receiver Operating Characteristic
PCA	Principal Component Analysis
MLP	Multi-Layer Perceptron
LRP	Layer-wise Relevance Propagation
HSI	Hyperspectral Imaging
FRKNN	Feature Ranking K-Nearest Neighbor

RNN	Recurrent Neural Network
ToF	Time-of-Flight
LR	Logistic Regression
GRU	Gated Recurrent Unit
AN	Generative Adversarial Nets
GPDCNN	Global Pooling Dilated Convolutional Neural Network
2D-CNN-BidGRU	2D-Convolutional-Bidirectional Gated Recurrent Unit Neural Network
OR-AC-GAN	Outlier Removal-Auxiliary Classifier-Generative Adversarial Nets

References

- McCulloch, W.S.; Pitts, W. A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.* **1943**, *5*, 115–133. [[CrossRef](#)]
- Ackley, D.H.; Hinton, G.E.; Sejnowski, T.J. A learning algorithm for Boltzmann machines. *Cogn. Sci.* **1985**, *9*, 147–169. [[CrossRef](#)]
- Kelley, H.J. Gradient theory of optimal flight paths. *Ars J.* **1960**, *30*, 947–954. [[CrossRef](#)]
- Dreyfus, S. The numerical solution of variational problems. *J. Math. Anal. Appl.* **1962**, *5*, 30–45. [[CrossRef](#)]
- Fukushima, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biol. Cybern.* **1980**, *36*, 193–202. [[CrossRef](#)]
- LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [[CrossRef](#)]
- Hinton, G.E.; Osindero, S.; Teh, Y.-W. A fast learning algorithm for deep belief nets. *Neural Comput.* **2006**, *18*, 1527–1554. [[CrossRef](#)]
- Hinton, G.E.; Salakhutdinov, R.R. Reducing the dimensionality of data with neural networks. *Science* **2006**, *313*, 504–507. [[CrossRef](#)]
- Fayjie, A.R.; Hossain, S.; Oualid, D.; Lee, D.-J. Driverless Car: Autonomous Driving Using Deep Reinforcement Learning in Urban Environment. In Proceedings of the 2018 15th International Conference on Ubiquitous Robots (UR), Hawaii Convention Center, Honolulu, HI, USA, 26–30 June 2018; pp. 896–901.
- Hossain, S.; Lee, D.-J. Autonomous-Driving Vehicle Learning Environments using Unity Real-time Engine and End-to-End CNN Approach. *J. Korea Robot. Soc.* **2019**, *14*, 122–130. [[CrossRef](#)]
- Kocić, J.; Jovičić, N.; Drndarević, V. An End-to-End Deep Neural Network for Autonomous Driving Designed for Embedded Automotive Platforms. *Sensors* **2019**, *19*, 2064. [[CrossRef](#)]
- Esteva, A.; Robicquet, A.; Ramsundar, B.; Kuleshov, V.; DePristo, M.; Chou, K.; Cui, C.; Corrado, G.; Thrun, S.; Dean, J. A guide to deep learning in healthcare. *Nat. Med.* **2019**, *25*, 24. [[CrossRef](#)] [[PubMed](#)]
- Miotto, R.; Wang, F.; Wang, S.; Jiang, X.; Dudley, J.T. Deep learning for healthcare: Review, opportunities and challenges. *Brief. Bioinform.* **2017**, *19*, 1236–1246. [[CrossRef](#)] [[PubMed](#)]
- Ravi, D.; Wong, C.; Deligianni, F.; Berthelot, M.; Andreu-Perez, J.; Lo, B.; Yang, G.-Z. Deep learning for health informatics. *IEEE J. Biomed. Health Inform.* **2016**, *21*, 4–21. [[CrossRef](#)] [[PubMed](#)]
- Goodfellow, I.J.; Bulatov, Y.; Ibarz, J.; Arnoud, S.; Shet, V. Multi-digit number recognition from street view imagery using deep convolutional neural networks. *arXiv* **2013**, arXiv:1312.6082.
- Jaderberg, M.; Simonyan, K.; Vedaldi, A.; Zisserman, A. Deep structured output learning for unconstrained text recognition. *arXiv* **2014**, arXiv:1412.5903.
- Yousfi, S.; Berrani, S.-A.; Garcia, C. Deep learning and recurrent connectionist-based approaches for Arabic text recognition in videos. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), Tunis, Tunisia, 23–26 August 2015; pp. 1026–1030.
- DeVries, P.M.; Viégas, F.; Wattenberg, M.; Meade, B.J. Deep learning of aftershock patterns following large earthquakes. *Nature* **2018**, *560*, 632. [[CrossRef](#)]
- Mousavi, S.M.; Zhu, W.; Sheng, Y.; Beroza, G.C. CRED: A deep residual network of convolutional and recurrent units for earthquake signal detection. *Sci. Rep.* **2019**, *9*, 10267. [[CrossRef](#)]
- Perol, T.; Gharbi, M.; Denolle, M. Convolutional neural network for earthquake detection and location. *Sci. Adv.* **2018**, *4*, e1700578. [[CrossRef](#)]

21. Siau, K.; Yang, Y. Impact of artificial intelligence, robotics, and machine learning on sales and marketing. In Proceedings of the Twelve Annual Midwest Association for Information Systems Conference (MWAIS 2017), Springfield, IL, USA, 18–19 May 2017; pp. 18–19.
22. Heaton, J.; Polson, N.; Witte, J.H. Deep learning for finance: Deep portfolios. *Appl. Stoch. Models Bus. Ind.* **2017**, *33*, 3–12. [[CrossRef](#)]
23. Heaton, J.; Polson, N.G.; Witte, J.H. Deep learning in finance. *arXiv* **2016**, arXiv:1602.06561.
24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
25. Mohanty, S.P.; Hughes, D.P.; Salathé, M. Using deep learning for image-based plant disease detection. *Front. Plant Sci.* **2016**, *7*, 1419. [[CrossRef](#)]
26. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
27. Sladojevic, S.; Arsenovic, M.; Anderla, A.; Culibrk, D.; Stefanovic, D. Deep neural networks based recognition of plant diseases by leaf image classification. *Comput. Intell. Neurosci.* **2016**, *2016*. [[CrossRef](#)]
28. Wan, J.; Wang, D.; Hoi, S.C.H.; Wu, P.; Zhu, J.; Zhang, Y.; Li, J. Deep learning for content-based image retrieval: A comprehensive study. In Proceedings of the 22nd ACM International Conference on Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 157–166.
29. Wu, R.; Yan, S.; Shan, Y.; Dang, Q.; Sun, G. Deep image: Scaling up image recognition. *arXiv* **2015**, arXiv:1501.02876.
30. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–8 December 2012; pp. 1097–1105.
31. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
32. Amara, J.; Bouaziz, B.; Algergawy, A. A Deep Learning-based Approach for Banana Leaf Diseases Classification. In Proceedings of the BTW (Workshops), Stuttgart, Germany, 6–10 March 2017; pp. 79–88.
33. Rebetz, J.; Satizábal, H.F.; Mota, M.; Noll, D.; Büchi, L.; Wendling, M.; Cannelle, B.; Pérez-Urbe, A.; Burgos, S. Augmenting a convolutional neural network with local histograms—A case study in crop classification from high-resolution UAV imagery. In Proceedings of the ESANN, Bruges, Belgium, 27–29 April 2016.
34. Rufswurm, M.; Körner, M. Multi-temporal land cover classification with long short-term memory neural networks. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *42*, 551. [[CrossRef](#)]
35. TÜRKÖĞLU, M.; Hanbay, D. Plant disease and pest detection using deep learning-based features. *Turk. J. Electr. Eng. Comput. Sci.* **2019**, *27*, 1636–1651. [[CrossRef](#)]
36. Mortensen, A.K.; Dyrmann, M.; Karstoft, H.; Jørgensen, R.N.; Gislum, R. Semantic segmentation of mixed crops using deep convolutional neural network. In Proceedings of the CIGR-AgEng Conference, Aarhus, Denmark, 26–29 June 2016; Abstracts and Full Papers. pp. 1–6.
37. Dyrmann, M.; Karstoft, H.; Midtby, H.S. Plant species classification using deep convolutional neural network. *Biosyst. Eng.* **2016**, *151*, 72–80. [[CrossRef](#)]
38. McCool, C.; Perez, T.; Upcroft, B. Mixtures of lightweight deep convolutional neural networks: Applied to agricultural robotics. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1344–1351. [[CrossRef](#)]
39. Santoni, M.M.; Sensuse, D.I.; Arymurthy, A.M.; Fanany, M.I. Cattle race classification using gray level co-occurrence matrix convolutional neural networks. *Procedia Comput. Sci.* **2015**, *59*, 493–502. [[CrossRef](#)]
40. Sørensen, R.A.; Rasmussen, J.; Nielsen, J.; Jørgensen, R.N. Thistle detection using convolutional neural networks. In Proceedings of the 2017 EFITA WCCA CONGRESS, Montpellier, France, 2–6 July 2017; p. 161.
41. Xinshao, W.; Cheng, C. Weed seeds classification based on PCANet deep learning baseline. In Proceedings of the 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), Hong Kong, China, 16–19 December 2015; pp. 408–415.
42. Hall, D.; McCool, C.; Dayoub, F.; Sunderhauf, N.; Upcroft, B. Evaluation of features for leaf classification in challenging conditions. In Proceedings of the 2015 IEEE Winter Conference on Applications of Computer Vision, Waikoloa Beach, HI, USA, 6–8 January 2015; pp. 797–804.

43. Kamilaris, A.; Prenafeta-Boldú, F.X. Deep learning in agriculture: A survey. *Comput. Electron. Agric.* **2018**, *147*, 70–90. [[CrossRef](#)]
44. Itzhaky, Y.; Farjon, G.; Khoroshevsky, F.; Shpigler, A.; Bar-Hillel, A. Leaf counting: Multiple scale regression and detection using deep CNNs. In Proceedings of the BMVC, North East, UK, 3–6 September 2018; p. 328.
45. Ubbens, J.; Cieslak, M.; Prusinkiewicz, P.; Stavness, I. The use of plant models in deep learning: An application to leaf counting in rosette plants. *Plant Methods* **2018**, *14*, 6. [[CrossRef](#)]
46. Rahnemoonfar, M.; Sheppard, C. Deep count: Fruit counting based on deep simulated learning. *Sensors* **2017**, *17*, 905. [[CrossRef](#)]
47. 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](#)]
48. Grinblat, G.L.; Uzal, L.C.; Larese, M.G.; Granitto, P.M. Deep learning for plant identification using vein morphological patterns. *Comput. Electron. Agric.* **2016**, *127*, 418–424. [[CrossRef](#)]
49. Lee, S.H.; Chan, C.S.; Wilkin, P.; Remagnino, P. Deep-plant: Plant identification with convolutional neural networks. In Proceedings of the 2015 IEEE International Conference on Image Processing (ICIP), Québec City, QC, Canada, 27–30 September 2015; pp. 452–456.
50. Pound, M.P.; Atkinson, J.A.; Townsend, A.J.; Wilson, M.H.; Griffiths, M.; Jackson, A.S.; Bulat, A.; Tzimiropoulos, G.; Wells, D.M.; Murchie, E.H. Deep machine learning provides state-of-the-art performance in image-based plant phenotyping. *Gigascience* **2017**, *6*, gix083. [[CrossRef](#)] [[PubMed](#)]
51. Milioto, A.; Lottes, P.; Stachniss, C. Real-time blob-wise sugar beets vs weeds classification for monitoring fields using convolutional neural networks. *Isprs Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *4*, 41. [[CrossRef](#)]
52. Potena, C.; Nardi, D.; Pretto, A. Fast and accurate crop and weed identification with summarized train sets for precision agriculture. In Proceedings of the International Conference on Intelligent Autonomous Systems, Shanghai, China, 3–7 July 2016; pp. 105–121.
53. Sun, Y.; Liu, Y.; Wang, G.; Zhang, H. Deep learning for plant identification in natural environment. *Comput. Intell. Neurosci.* **2017**, *2017*, 7361042. [[CrossRef](#)] [[PubMed](#)]
54. Singh, A.K.; Ganapathysubramanian, B.; Sarkar, S.; Singh, A. Deep learning for plant stress phenotyping: Trends and future perspectives. *Trends Plant Sci.* **2018**, *23*, 883–898. [[CrossRef](#)] [[PubMed](#)]
55. Brahim, M.; Arsenovic, M.; Laraba, S.; Sladojevic, S.; Boukhalifa, K.; Moussaoui, A. Deep learning for plant diseases: Detection and saliency map visualisation. In *Human and Machine Learning*; Springer: Berlin, Germany, 2018; pp. 93–117.
56. Sibiyi, M.; Sumbwanyambe, M. A Computational Procedure for the Recognition and Classification of Maize Leaf Diseases Out of Healthy Leaves Using Convolutional Neural Networks. *AgriEngineering* **2019**, *1*, 119–131. [[CrossRef](#)]
57. Zhang, K.; Wu, Q.; Liu, A.; Meng, X. Can Deep Learning Identify Tomato Leaf Disease? *Adv. Multimed.* **2018**, *2018*, 10. [[CrossRef](#)]
58. Ferentinos, K.P. Deep learning models for plant disease detection and diagnosis. *Comput. Electron. Agric.* **2018**, *145*, 311–318. [[CrossRef](#)]
59. Ramcharan, A.; Baranowski, K.; McCloskey, P.; Ahmed, B.; Legg, J.; Hughes, D.P. Deep learning for image-based cassava disease detection. *Front. Plant Sci.* **2017**, *8*, 1852. [[CrossRef](#)]
60. 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; pp. 989–992.
61. Yamamoto, K.; Togami, T.; Yamaguchi, N. Super-resolution of plant disease images for the acceleration of image-based phenotyping and vigor diagnosis in agriculture. *Sensors* **2017**, *17*, 2557. [[CrossRef](#)]
62. Durmuş, H.; Güneş, E.O.; Kırıcı, M. Disease detection on the leaves of the tomato plants by using deep learning. In Proceedings of the 2017 6th International Conference on Agro-Geoinformatics, Fairfax, VA, USA, 7–10 August 2017; pp. 1–5.
63. 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.* **2019**, *161*, 272–279. [[CrossRef](#)]
64. Rangarajan, A.K.; Purushothaman, R.; Ramesh, A. Tomato crop disease classification using pre-trained deep learning algorithm. *Procedia Comput. Sci.* **2018**, *133*, 1040–1047. [[CrossRef](#)]

65. Cruz, A.C.; Luvisi, A.; De Bellis, L.; Ampatzidis, Y. Vision-based plant disease detection system using transfer and deep learning. In Proceedings of the 2017 ASABE Annual International Meeting, Spokane, WA, USA, 16–19 July 2017; p. 1.
66. Ma, J.; Du, K.; Zheng, F.; Zhang, L.; Gong, Z.; Sun, Z. A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. *Comput. Electron. Agric.* **2018**, *154*, 18–24. [[CrossRef](#)]
67. Brahimi, M.; Mahmoudi, S.; Boukhalifa, K.; Moussaoui, A. Deep interpretable architecture for plant diseases classification. *arXiv* **2019**, arXiv:1905.13523.
68. Fuentes, A.; Yoon, S.; Kim, S.; Park, D. A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors* **2017**, *17*, 2022. [[CrossRef](#)] [[PubMed](#)]
69. Selvaraj, M.G.; Vergara, A.; Ruiz, H.; Safari, N.; Elayabalan, S.; Ocimati, W.; Blomme, G. AI-powered banana diseases and pest detection. *Plant Methods* **2019**, *15*, 92. [[CrossRef](#)]
70. DeChant, C.; Wiesner-Hanks, T.; Chen, S.; Stewart, E.L.; Yosinski, J.; Gore, M.A.; Nelson, R.J.; Lipson, H. Automated identification of northern leaf blight-infected maize plants from field imagery using deep learning. *Phytopathology* **2017**, *107*, 1426–1432. [[CrossRef](#)]
71. Wallelign, S.; Polceanu, M.; Buche, C. Soybean Plant Disease Identification Using Convolutional Neural Network. In Proceedings of the Thirty-First International Flairs Conference, Melbourne, FL, USA, 21–23 May 2018.
72. Brahimi, M.; Boukhalifa, K.; Moussaoui, A. Deep learning for tomato diseases: Classification and symptoms visualization. *Appl. Artif. Intell.* **2017**, *31*, 299–315. [[CrossRef](#)]
73. Lu, J.; Hu, J.; Zhao, G.; Mei, F.; Zhang, C. An in-field automatic wheat disease diagnosis system. *Comput. Electron. Agric.* **2017**, *142*, 369–379. [[CrossRef](#)]
74. Ha, J.G.; Moon, H.; Kwak, J.T.; Hassan, S.I.; Dang, M.; Lee, O.N.; Park, H.Y. Deep convolutional neural network for classifying Fusarium wilt of radish from unmanned aerial vehicles. *J. Appl. Remote Sens.* **2017**, *11*, 042621. [[CrossRef](#)]
75. Lin, K.; Gong, L.; Huang, Y.; Liu, C.; Pan, J. Deep learning-based segmentation and quantification of cucumber Powdery Mildew using convolutional neural network. *Front. Plant Sci.* **2019**, *10*, 155. [[CrossRef](#)]
76. Barbedo, J.G.A. Plant disease identification from individual lesions and spots using deep learning. *Biosyst. Eng.* **2019**, *180*, 96–107. [[CrossRef](#)]
77. Ghosal, S.; Blystone, D.; Singh, A.K.; Ganapathysubramanian, B.; Singh, A.; Sarkar, S. An explainable deep machine vision framework for plant stress phenotyping. *Proc. Natl. Acad. Sci.* **2018**, *115*, 4613–4618. [[CrossRef](#)] [[PubMed](#)]
78. Lu, Y.; Yi, S.; Zeng, N.; Liu, Y.; Zhang, Y. Identification of rice diseases using deep convolutional neural networks. *Neurocomputing* **2017**, *267*, 378–384. [[CrossRef](#)]
79. Picon, A.; Alvarez-Gila, A.; Seitz, M.; Ortiz-Barredo, A.; Echazarra, J.; Johannes, A. Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. *Comput. Electron. Agric.* **2019**, *161*, 280–290. [[CrossRef](#)]
80. Johannes, A.; Picon, A.; Alvarez-Gila, A.; Echazarra, J.; Rodriguez-Vaamonde, S.; Navajas, A.D.; Ortiz-Barredo, A. Automatic plant disease diagnosis using mobile capture devices, applied on a wheat use case. *Comput. Electron. Agric.* **2017**, *138*, 200–209. [[CrossRef](#)]
81. 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](#)]
82. Khan, M.A.; Akram, T.; Sharif, M.; Awais, M.; Javed, K.; Ali, H.; Saba, T. CCDF: Automatic system for segmentation and recognition of fruit crops diseases based on correlation coefficient and deep CNN features. *Comput. Electron. Agric.* **2018**, *155*, 220–236. [[CrossRef](#)]
83. Kerkech, M.; Hafiane, A.; Canals, R. Deep leaning approach with colorimetric spaces and vegetation indices for vine diseases detection in UAV images. *Comput. Electron. Agric.* **2018**, *155*, 237–243. [[CrossRef](#)]
84. Nagasubramanian, K.; Jones, S.; Singh, A.K.; Singh, A.; Ganapathysubramanian, B.; Sarkar, S. Explaining hyperspectral imaging based plant disease identification: 3D CNN and saliency maps. *arXiv* **2018**, arXiv:1804.08831.
85. Jiang, P.; Chen, Y.; Liu, B.; He, D.; Liang, C. Real-Time Detection of Apple Leaf Diseases Using Deep Learning Approach Based on Improved Convolutional Neural Networks. *IEEE Access* **2019**, *7*, 59069–59080. [[CrossRef](#)]

86. Zhang, X.; Qiao, Y.; Meng, F.; Fan, C.; Zhang, M. Identification of maize leaf diseases using improved deep convolutional neural networks. *IEEE Access* **2018**, *6*, 30370–30377. [[CrossRef](#)]
87. Liu, B.; Zhang, Y.; He, D.; Li, Y. Identification of apple leaf diseases based on deep convolutional neural networks. *Symmetry* **2017**, *10*, 11. [[CrossRef](#)]
88. Chen, J.; Liu, Q.; Gao, L. Visual Tea Leaf Disease Recognition Using a Convolutional Neural Network Model. *Symmetry* **2019**, *11*, 343. [[CrossRef](#)]
89. Kamal, K.; Yin, Z.; Wu, M.; Wu, Z. Depthwise separable convolution architectures for plant disease classification. *Comput. Electron. Agric.* **2019**, *165*, 104948.
90. 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](#)]
91. Veys, C.; Chatziavgerinos, F.; AlSuwaidi, A.; Hibbert, J.; Hansen, M.; Bernotas, G.; Smith, M.; Yin, H.; Rolfe, S.; Grieve, B. Multispectral imaging for presymptomatic analysis of light leaf spot in oilseed rape. *Plant Methods* **2019**, *15*, 4. [[CrossRef](#)] [[PubMed](#)]
92. Mahlein, A.-K.; Alisaac, E.; Al Masri, A.; Behmann, J.; Dehne, H.-W.; Oerke, E.-C. Comparison and Combination of Thermal, Fluorescence, and Hyperspectral Imaging for Monitoring Fusarium Head Blight of Wheat on Spikelet Scale. *Sensors* **2019**, *19*, 2281. [[CrossRef](#)] [[PubMed](#)]
93. Xie, C.; Yang, C.; He, Y. Hyperspectral imaging for classification of healthy and gray mold diseased tomato leaves with different infection severities. *Comput. Electron. Agric.* **2017**, *135*, 154–162. [[CrossRef](#)]
94. Shuaibu, M.; Lee, W.S.; Schueller, J.; Gader, P.; Hong, Y.K.; Kim, S. Unsupervised hyperspectral band selection for apple Marssonina blotch detection. *Comput. Electron. Agric.* **2018**, *148*, 45–53. [[CrossRef](#)]
95. Chen, T.; Zhang, J.; Chen, Y.; Wan, S.; Zhang, L. Detection of peanut leaf spots disease using canopy hyperspectral reflectance. *Comput. Electron. Agric.* **2019**, *156*, 677–683. [[CrossRef](#)]
96. Moghadam, P.; Ward, D.; Goan, E.; Jayawardena, S.; Sikka, P.; Hernandez, E. Plant disease detection using hyperspectral imaging. In Proceedings of the 2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA), Sydney, Australia, 29 November–1 December 2017; pp. 1–8.
97. Hruška, J.; Adão, T.; Pádua, L.; Marques, P.; Cunha, A.; Peres, E.; Sousa, A.; Morais, R.; Sousa, J.J. Machine learning classification methods in hyperspectral data processing for agricultural applications. In Proceedings of the International Conference on Geoinformatics and Data Analysis, Prague, Czech Republic, 20–22 April 2018; pp. 137–141.
98. Ashourloo, D.; Aghighi, H.; Matkan, A.A.; Mobasheri, M.R.; Rad, A.M. An investigation into machine learning regression techniques for the leaf rust disease detection using hyperspectral measurement. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2016**, *9*, 4344–4351. [[CrossRef](#)]
99. Su, J.; Liu, C.; Coombes, M.; Hu, X.; Wang, C.; Xu, X.; Li, Q.; Guo, L.; Chen, W.-H. Wheat yellow rust monitoring by learning from multispectral UAV aerial imagery. *Comput. Electron. Agric.* **2018**, *155*, 157–166. [[CrossRef](#)]
100. 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](#)]
101. Zhu, H.; Chu, B.; Zhang, C.; Liu, F.; Jiang, L.; He, Y. Hyperspectral imaging for presymptomatic detection of tobacco disease with successive projections algorithm and machine-learning classifiers. *Sci. Rep.* **2017**, *7*, 4125. [[CrossRef](#)] [[PubMed](#)]
102. Halicek, M.; Lu, G.; Little, J.V.; Wang, X.; Patel, M.; Griffith, C.C.; El-Deiry, M.W.; Chen, A.Y.; Fei, B. Deep convolutional neural networks for classifying head and neck cancer using hyperspectral imaging. *J. Biomed. Opt.* **2017**, *22*, 060503. [[CrossRef](#)] [[PubMed](#)]
103. Ma, X.; Geng, J.; Wang, H. Hyperspectral image classification via contextual deep learning. *Eurasip J. Image Video Process.* **2015**, *2015*, 20. [[CrossRef](#)]
104. Paoletti, M.; Haut, J.; Plaza, J.; Plaza, A. A new deep convolutional neural network for fast hyperspectral image classification. *Isprs J. Photogramm. Remote Sens.* **2018**, *145*, 120–147. [[CrossRef](#)]
105. Hu, W.; Huang, Y.; Wei, L.; Zhang, F.; Li, H. Deep convolutional neural networks for hyperspectral image classification. *J. Sens.* **2015**, *2015*. [[CrossRef](#)]
106. Chen, Y.; Jiang, H.; Li, C.; Jia, X.; Ghamisi, P. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 6232–6251. [[CrossRef](#)]

107. Mou, L.; Ghamisi, P.; Zhu, X.X. Deep recurrent neural networks for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3639–3655. [[CrossRef](#)]
108. Wu, H.; Prasad, S. Convolutional recurrent neural networks for hyperspectral data classification. *Remote Sens.* **2017**, *9*, 298. [[CrossRef](#)]
109. Yue, J.; Zhao, W.; Mao, S.; Liu, H. Spectral–spatial classification of hyperspectral images using deep convolutional neural networks. *Remote Sens. Lett.* **2015**, *6*, 468–477. [[CrossRef](#)]
110. Signoroni, A.; Savardi, M.; Baronio, A.; Benini, S. Deep Learning Meets Hyperspectral Image Analysis: A Multidisciplinary Review. *J. Imaging* **2019**, *5*, 52. [[CrossRef](#)]
111. Jin, X.; Jie, L.; Wang, S.; Qi, H.; Li, S. Classifying wheat hyperspectral pixels of healthy heads and Fusarium head blight disease using a deep neural network in the wild field. *Remote Sens.* **2018**, *10*, 395. [[CrossRef](#)]
112. Wang, D.; Vinson, R.; Holmes, M.; Seibel, G.; Bechar, A.; Nof, S.; Tao, Y. Early Detection of Tomato Spotted Wilt Virus by Hyperspectral Imaging and Outlier Removal Auxiliary Classifier Generative Adversarial Nets (OR-AC-GAN). *Sci. Rep.* **2019**, *9*, 4377. [[CrossRef](#)] [[PubMed](#)]
113. Polder, G.; Blok, P.M.; de Villiers, H.A.C.; van der Wolf, J.M.; Kamp, J. Potato Virus Y Detection in Seed Potatoes Using Deep Learning on Hyperspectral Images. *Front. Plant Sci.* **2019**, *10*. [[CrossRef](#)] [[PubMed](#)]
114. Zhang, X.; Han, L.; Dong, Y.; Shi, Y.; Huang, W.; Han, L.; González-Moreno, P.; Ma, H.; Ye, H.; Sobeih, T. A Deep Learning-Based Approach for Automated Yellow Rust Disease Detection from High-Resolution Hyperspectral UAV Images. *Remote Sens.* **2019**, *11*, 1554. [[CrossRef](#)]
115. Golhani, K.; Balasundram, S.K.; Vadamalai, G.; Pradhan, B. A review of neural networks in plant disease detection using hyperspectral data. *Inf. Process. Agric.* **2018**, *5*, 354–371. [[CrossRef](#)]



© 2019 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 (<http://creativecommons.org/licenses/by/4.0/>).