Deep learning approaches for object recognition in plant diseases: a review

Zimo Zhou, Yue Zhang, Zhaohui Gu, Simon X. Yang

School of Engineering, University of Guelph, Guelph N1G 2W1, Canada.

Correspondence to: Prof. Simon X. Yang, School of Engineering, University of Guelph, 50 Stone Rd E, Guelph, N1G 2W1, Canada. E-mail: syang@uoguelph.ca

Abstract

Plant diseases pose a significant threat to the economic viability of agriculture and the normal functioning of trees in forests. Accurate detection and identification of plant diseases are crucial for smart agricultural and forestry management. Artificial intelligence has been successfully applied to agriculture in recent years. Many intelligent object recognition algorithms, specifically deep learning approaches, have been proposed to identify diseases in plant images. The goal is to reduce labor and improve detection efficiency. This article reviews the application of object detection methods for detecting common plant diseases, such as tomato, citrus, maize, and pine trees. It introduces various object detection models, ranging from basic to modern and sophisticated networks, and compares the innovative aspects and drawbacks of commonly used neural network models. Furthermore, the article discusses current challenges in plant disease detection and object detection methods and suggests promising directions for future work in learning-based plant disease detection systems.

Keywords: Plant disease detection, deep learning, object detection, plant disease management

1. INTRODUCTION

The world’s population has been rapidly increasing, growing from approximately 6 billion to 8 billion over the past 20 years. By early 2023, India had surpassed China as the world’s most populous nation, with a population of 1.42 billion. Consequently, the demand for food and freshwater resources continues to rise,
necessitating improvements in production efficiency. To address these challenges, various techniques have been developed and applied to monitor the growth conditions of cash crops and accelerate the fruit harvesting process. Sensors are used to collect data to ensure that environmental conditions remain within the required range. Machines equipped with recognition systems have been built to enhance the efficiency of the harvest process. Additionally, advancements in manufacturing techniques and the increasing globalization of human activities have facilitated the spread of invasive species and the transmission of plant diseases. For instance, the invasive vector known as Asian Citrus Psyllids from South Asia has led to the destruction of 50 million citrus trees\textsuperscript{[1]}. To prevent massive economic losses, people employ detection methods to identify plant diseases or pests and implement early tree disposal. However, the detection of these issues can be challenging due to the large number of test samples and the need for a significant labor force. Intelligent algorithms, which have been extensively studied, have played a crucial role in enhancing detection and recognition systems.

Human beings possess an innate ability to perceive colors and patterns in the world around them. Inspired by the human visual system, researchers have developed computer vision algorithms to extract features from images. This field, a significant component of artificial intelligence, has made remarkable progress, thanks to advances in deep learning. For decades, researchers have been dedicated to applying machine vision to various image-related tasks, including edge detection, image recognition, and segmentation. Recently, the domain of smart farming has witnessed a surge in interest in implementing cutting-edge computer vision techniques for a range of tasks. Vision systems enable the capture of valuable image information through image sensors, providing an efficient means of collecting data for model training. In pursuit of cost reduction and improved data quality, individuals have introduced energy-efficient, long-range cameras designed for real-world environments\textsuperscript{[2,3]}. Some researchers employ ground-based cameras and other Internet of Things (IoT) techniques to monitor animal growth and predict plant growth projections\textsuperscript{[4,5]}. Others focus on applying innovative computer vision methods to recognize aerial imagery, which holds great promise. A real-time image processing method was proposed in\textsuperscript{[6]} for the purpose of weed detection and management based on a deep neural network (DNN) object detection model called R-CNN\textsuperscript{[7]}. Moreover, some researchers work on developing an intelligent system for collecting sensor data with unmanned aerial vehicles (UAV) and using it on crop growth monitoring and classifying value crops\textsuperscript{[8,9]}. Object identification, a fundamental task in computer vision, plays a pivotal role in extracting and analyzing information from images. It contributes to numerous high-level machine learning tasks, including scene understanding, visual reasoning, instance segmentation, and image question answering\textsuperscript{[10]}. The goal of object detection is to classify one or more effective targets into predefined categories and find them on still picture or video data. The rapid technological evolution of DNNs has significantly advanced object detection, drawing the attention of scholars. Today, object detection finds extensive applications in agriculture, including plant disease and pest detection, crop differentiation, weed classification, plan recognition, and fruit counting and harvesting.

Countries such as China, India, and Brazil have agriculture as a cornerstone of their economic development. According to estimates from the Food and Agriculture Organization, plant diseases result in a staggering 220 billion loss in the global economy\textsuperscript{[11]}. To boost the yields of economic crops and preserve forest resources, people have been dedicated to plant disease prevention for decades. Traditionally, in plant disease control principles, efforts were focused on preventing pathogen occurrence or altering environments conducive to common diseases. Once pathogens and susceptible host plants interact, an effective approach is to manage their development and transmission within acceptable limits\textsuperscript{[12]}. Visual observation and biological assays have been commonly used methods, involving extensive human labor for comparing healthy and infected plants or prolonged cultivation. However, with rising labor costs and the urgent need for disease management, automated detection and control techniques have gained popularity in both industry and research sectors. These methods offer a combination of cost-effectiveness and reduced labor demands. Various approaches have been explored for plant disease detection. For instance, Wang et al.\textsuperscript{[13]} modified the initial Single-Shot Detector (SSD)\textsuperscript{[14]} by adding a block attention module, and get 92.2 percent mean Average Precision (mAP) result on
the PlantVillage dataset, which contains over 500 hundreds health and diseased tree leaves categorized to 38 species. To generate more diverse datasets with various environmental conditions, six distinct augmentation techniques were employed, including scaling, rotation, noise injection, gamma correction, image flipping, and PCA color augmentation. In another study, a modified R-CNN was implemented for the early treatment of tomato leaf infections and compared to the state-of-the-art models. Weeds pose a significant threat to agricultural crops, depriving them of essential resources like water, light, and nutrients. To reduce economic losses and decrease reliance on herbicides in weed management, some one-stage detectors have been utilized for weed control. In summary, object detection algorithms hold tremendous promise in the realm of plant disease detection and prevention.

Normally, for building an automatic plant disease detection system, the initial step is to the identification of infected plants. The identifying process may become challenging due to the subtle characteristic and variability of symptoms among color and shape in different stages of infection. The object detection data rely on information captured from image sensors, which are often affected by the shooting condition such as temperature, visibility, illumination, and humidity. These impacts may influence the data obtained from certain locations to have largely different features. Some data augmentation methods were proposed to improve the robustness of the detection method and reduce the imbalance in training data. Typically, object detection based plant disease identification systems have five following steps: image acquisition, image processing and labeling, model training and tuning, infected plants detection, and disease treatment and control. The completion of these processes has been achieved using various new technologies. The development of UAVs and satellites allows for capturing large amounts of imagery data from the air. The acceleration of graphic calculation and expansion of video random-access memory enhance the possibility of building complicated feed forward models. The most crucial improvement is the rapidly flourishing deep neural network based detection algorithms. Throughout the evolution of deep learning, the back-propagation method proposed by Rumelhart, Hinton, and Williams endows the model with the ability to optimize models by differences between predictions and target values. In 1980, a 'Neocognitron' neural network was proposed by Fukushima, who pioneered the ideas of feature extraction, pooling layer design, and convolutional layer. After that, LeCun et al. built the first convolution neural network (CNN) for a handwriting character classification problem considering where the name of CNN originated from. With the growth and development of these techniques, CNNs can extract high-dimensional feature information and show great potential in the field of plant disease detection. Some scholars have reviewed the implementation of the CNN model in agriculture by utilising this potent technology. Mohanty et al. investigate some deep learning models and analyze their performance on a public dataset. Saleem et al. evaluate various deep learning approaches and introduce some visualisation ways to provide a clear understanding of plant illnesses. Hyper-spectral imaging are also discussed in this survey. In another review work, Li et al. research the detection of plant diseases and insect pests, and the difficulties are also examined. The author also list some augmentation methods from traditional algorithms to deep learning based generative model, which would assist researchers with insufficient datasets in their studies. Expect focusing on the detecting models, imaging techniques are an important component of a detection system as well. Singh et al. provide useful information about some advanced imaging technologies of computer vision for disease detection. This paper analyze the imaging methods of different sensors and list their carriers, and application on different agriculture systems are summarized.

This paper presents a comprehensive review of state-of-the-art studies on deep learning-based object detection models and their applications in the detection of plant diseases. The contributions of this survey can be summarized as follows: (1) The advancement of applying deep learning based detecting methods on plant disease recognition is thoroughly studied and reviewed; (2) Common plant species and the diseases they are susceptible to while being cultivated are explored; (3) A comprehensive study of contemporary representative object detection algorithms is presented, including comparisons of their innovations and drawbacks; (4) The main issues and challenges in implementing object detection methods in agriculture are discussed, and potential
future development directions are analyzed.

The rest of this article is organized as follows: Section 2 introduces the plant diseases of different species, and several deep learning object detection methods are applied. Section 3 introduces some widely used object detection models. Section 4 presents the challenges in this task and some future trends. Conclusions are discussed in Section 5.

2. PLANT DISEASE DETECTION

From the definition of Dr. H. H. Whetzel\cite{whetzel1954plant}, plant disease is described as a physiological process that is harmful to plants, manifested by abnormal cellular activity, and displayed in recognizable pathological conditions, which are caused by a continuous irritation from a causative agent. The causing factor could be a biological organism or an environmental condition. The disorders can be abiotic diseases or biotic diseases. External conditions, including nutrient deficiency, pH imbalance, compaction, and salt damage, usually cause non-infectious diseases. A biologically diseased plant can have the following pathogens: bacteria, fungi, viruses, Parasitic plants, and nematodes. Bacteria are tiny single-celled organisms with cell walls that reproduce via binary fission. Common ways for bacteria to invade plants can be through the plant's open wound, spread by the infected insect, and soil-borne infection through root system\cite{schelken2007bacterial}.

2.1. Tomato disease detection

Every year, around 180 million tons of tomatoes are produced, and it is one of the primary meals on people's dinner tables. The pathology for tomato disease is complex, and the symptom can manifest on fruits or leaves. Some common disease includes bacteria spot that can affect fruit, leaf, and stem, and bacteria canker appearing on the crown and above. Other than that, early blight, late blight, leaf mold, and septoria leaf spot are caused by \textit{Alternaria} fungus, \textit{Phytophthora infestans}, \textit{Passalora fulva}, and \textit{Septoria lycopersici} respectively\cite{garcia2017bacterial}. Images in Figure 1 compare the different characteristics of 4 types of diseases obtained from the PlantVillage dataset\cite{plantvillage}. These diseases usually manifest as dark lesion spots on the stem and leaf. For virus-induced plant diseases, the yellow leaf curl virus and mosaic virus are frequent illnesses for tomato plants. In\cite{li2018characterization}, three tomato diseases are investigated, and training and validation datasets are collected and annotated by their work. The bacterial canker dominates the data, which could be one of the most serious diseases for tomatoes. The occurrence of bacteria canker caused by \textit{Clavibacter michiganensis} was found in Michigan, USA, in 1910\cite{farrar2002development}. Since it is a prevalent disease, a lot of researchers have conducted their efforts in the detection of this issue. Natarajan \textit{et al.}\cite{natarajan2020automated} use Faster R-CNN deep learning model\cite{ren2015faster} to detect infected tomato 1090 images from Andhra Pradesh. Another group compared Basic R-CNN\cite{wu2019basic}, Fast R-CNN\cite{girshick2015fast}, and Faster R-CNN method on their tomato leaf infection data\cite{zhang2020deep}. Jiaotao built an improved YOLO model with a visual attention mechanism added to focus

![Figure 1. Four common tomato diseases symptoms on leaves\cite{plantvillage}.](image-url)

2.1. Tomato disease detection

Every year, around 180 million tons of tomatoes are produced, and it is one of the primary meals on people's dinner tables. The pathology for tomato disease is complex, and the symptom can manifest on fruits or leaves. Some common disease includes bacteria spot that can affect fruit, leaf, and stem, and bacteria canker appearing on the crown and above. Other than that, early blight, late blight, leaf mold, and septoria leaf spot are caused by \textit{Alternaria} fungus, \textit{Phytophthora infestans}, \textit{Passalora fulva}, and \textit{Septoria lycopersici} respectively\cite{garcia2017bacterial}. Images in Figure 1 compare the different characteristics of 4 types of diseases obtained from the PlantVillage dataset\cite{plantvillage}. These diseases usually manifest as dark lesion spots on the stem and leaf. For virus-induced plant diseases, the yellow leaf curl virus and mosaic virus are frequent illnesses for tomato plants. In\cite{li2018characterization}, three tomato diseases are investigated, and training and validation datasets are collected and annotated by their work. The bacterial canker dominates the data, which could be one of the most serious diseases for tomatoes. The occurrence of bacteria canker caused by \textit{Clavibacter michiganensis} was found in Michigan, USA, in 1910\cite{farrar2002development}. Since it is a prevalent disease, a lot of researchers have conducted their efforts in the detection of this issue. Natarajan \textit{et al.}\cite{natarajan2020automated} use Faster R-CNN deep learning model\cite{ren2015faster} to detect infected tomato 1090 images from Andhra Pradesh. Another group compared Basic R-CNN\cite{wu2019basic}, Fast R-CNN\cite{girshick2015fast}, and Faster R-CNN method on their tomato leaf infection data\cite{zhang2020deep}. Jiaotao built an improved YOLO model with a visual attention mechanism added to focus
on the tomato virus problems in China. Xuewei et al. worked on the tomato images collected from the greenhouse and performed several YOLO-based methods on their data. A more advanced object detection method, YOLO v4, is leveraged to provide a strategy that is effective and efficient for identifying various plant diseases under challenging situations, and it was tested on a tomato diseases data set. Several works are also focusing on these diseases: a system using Fuzzy Support Vector Machine (SVM) and R-CNN based classifier, an improved You Only Look Once (YOLO) method for both tomato disease classification in both and, and a research work uses Mask R-CNN proposed by He et al., which outperforms the former mentioned method in tomato diseases detecting systems.

2.2. Citrus orchard disease detection
Orange is one of the most popular and extensively grown fruits in the world, and orange juice and cans are prominent market items. However, some primary citrus diseases like citrus canker and greening have caused substantial financial losses for citrus orchards. Figure 2 illustrates the overall features of plant diseases and their variations on the leaves and branches, which are collected from the University of California Agriculture and Natural Resources. Some intelligent detection systems are employed for early detection to help control the occurrence of disease infection and transmission. Su et al. accomplish research on citrus disease recognition in Guangxi Province, China, by using a Region-based detection method superior to other machine learning methods like Support Vector Machine (SVM). Dhiman et al. use a selective search object detection method to evaluate the disease’s Severity Level in Citrus Fruits. Both the previous works’ detecting algorithms, which are based on typical machine learning techniques, and the following discussion use deep learning. The citrus greening disease, also known as HongLongBing, is a severely destructive disease in the orange industry. Dai et al. improve Cascade R-CNN with an atrous pooling strategy for detecting the citrus psyllid, which is the insect vector of this disease. Some academics like develop end-to-end region proposal training models based on Faster R-CNN built for different citrus diseases. The following works improved the YOLO structure in their experiments and achieved excellent results. In a research from Song et al., an automatic system is designed on a dataset with two types of diseases: Canker and Greening. Researchers like build a detecting mobile app for facilitating in the timely detection and prevention of HLB transmission in citrus fields. Both of above two groups prefer YOLO architecture as their ideal detection method. Some research uses their citrus disease dataset to make comparisons between one-stage and two-stage methods. While most researchers pay attention to deep learning models with large amount parameters, da Silva et al. build some efficient mobile networks aiming at implementing real-time detection on smartphones.

2.3. Maize disease detection
Maize is a significant food crop, and numerous food products such as corn starch, corn oil, and popcorn are derived from it. It serves as a vital food source and can be processed into various food items, including corn starch, corn oil, and popcorn. Maize, however, faces common diseases, including Southern Corn Leaf Blight, Northern Corn Leaf Blight, Curvularia Leaf Spot, and Gray Leaf Spot, which result from fungal infections. Bacterial stalk rot, caused by Erwinia dissolvens, is another issue, as well as the Maize Streak Virus, which can
lead to stunted development and reduced grain yields. In a survey of details of maize diseases are discussed and a deep learning method is proposed for classifying four categories of maize leaves. In order to implement an intelligent diagnosis and reduce human labor, researchers perform some object detection approaches to maize disease detection. Kumar et al. use Faster R-CNN and residual network block to detect three common maize diseases. An improved R-CNN method called MFaster R-CNN is proposed and tested on a public dataset containing 1442 images, which performed much better than the basic R-CNN model. Other groups using Mask R-CNN as component of their agriculture systems. Some concentrate on the common rust in maize and identify the severity of the disease by counting the number of pustules. Besides images obtained from a ground-level camera, Stewart et al. utilize a UAV at a 6-meter altitude and take over 7000 pictures. The R-CNN model and its extensions have produced excellent detection results, whereas some researchers have chosen the YOLO series methods as their primary study strategies. In, the author proposed a one-stage model named MFF-CNN based on YOLO, and a coordinate attention block is added to this detector, which outperforms other baseline models. A newly proposed YOLO v4 network structure is adopted in to precisely find and identify tar spot disease lesions. In the Philippines, a study using YOLO v5, a more advanced detection tool improved based to YOLO v4, on three maize crop diseases yields a score of 97 percent average precision.

2.4. Pine wilt detection

*Bursaphelenchus xylophilus*, commonly known as the pine wood nematode (PWN), is responsible for causing pine wilt disease (PWD). Although the signs of the disease have been observed since 1905, the illness was initially recorded in Japan in 1971. PWD has devastating consequences for conifers across East Asia, North America, and Europe, significantly impacting the global forest economy. Typically, the carry vector of PWN is a type of beetle known as *Monochamus alternatus*. Images in Figure 4 show some infected trees from the ground and aerial views, captured during an aerial photography project in Yichang City, China. Given these reasons, it’s imperative to prevent the spread of the disease among pine trees and take effective measures to manage infected trees. One straightforward technique to curb disease propagation is the removal of damaged trees, including their stumps, with immediate incineration of the timber. When it comes to imagery analysis, remote sensing photography is often used to gather data, given the challenges of acquiring data within dense forests. Various methods have been employed to detect and locate pine trees exhibiting symptoms in remote sensing data, with some researchers adopting the YOLO detector as part of their detection systems. Wu et al. improved the model with an Efficient Network structure and replaced the active function with a new one called Mish. Zhu et al. employed a helicopter equipped with a high-resolution camera and collected over 2900 images from Jilin Province, China. They divided the detection process into several stages, which are able to combine the detector and GoogleNet for wilt degree classification. Another work proposed a method that enhances the backbone network with a MobileNet v2 structure and a convolutional...
block attention module\cite{78}. Moreover, a focus unit is also included in the detection approach, with the goal of extracting features from pine trees at various levels\cite{79}. The data collection is aided by the DJI drone at three different altitudes, which will increase the robustness of the detecting model. Following the researcher's work on improving R-CNN based model for pine wood detection. For deploying two-stage methods, the development is generally based on the baseline of Faster R-CNN or Mask R-CNN method. Deng et al.\cite{80} compared the primary method with several backbone networks and achieved more extraordinary results with some improvement on the regional proposal network and the pre-selected anchors. A two-stage detection process is trained to eliminate the background at the first step and to conduct the detection stage from the premier results\cite{81}. In\cite{82}, a receptive field block and a multilevel pyramid approach are selected to strengthen Mask R-CNN model. The above research projects are based on image processing and recognition of RGB images, while some groups leverage the high dimensional feature in multi-spectral images. Qin et al.\cite{83} and Park\cite{84} use drones mounted with multi-spectral cameras for training data collection and conducting deep neural network detection.

2.5. Disease detection for other plants

Deep neural networks perform the targeted detection tasks and outperform detection work on additional plant diseases like soybean, apple, and tea tree. The origins of soybean can be traced back to East Asia, China, as early as 7000 BCE\cite{85}. It now becomes the main source of plant protein in people's daily consumption, and soybean oil extracted from soybean is the most widely unitized cooking oil, which contains a relatively balanced fatty acid composition. In\cite{86}, a multi-feature Faster R-CNN model is designed for soybean bacteria spot, and data augmentation techniques are leveraged. In\cite{87}, botrytis cinerea and bacterial spots are detected by Fast
R-CNN combined with transfer learning. Tea is a valuable cash crop as well as a popular beverage. Some researchers try to propose methods for several major tea leaf diseases like brown blight, Leaf scab, and tea coal diseases\cite{88-92}. Above presented works use Fast and Faster R-CNN as their basic models, while others add various module on detection method and proposed new detecting algorithms. A super-resolution method is used to rebuild clearer pictures for lacking sufficient feature problems in tea leaf UAV images, and the Receptive field blocks\cite{93} are added to the basic detector for accurate and swift detection in\cite{94}. Illnesses like grape esca and black rot have brought substantial losses to grape farms. Many organizations perform research on grape leaf diagnosis detection. Spatial attention and channel attention are used in\cite{95}. A Squeeze and excitation network is added in\cite{96} to assist the detector in focusing on the illness region rather than the background. The works as mentioned earlier demonstrate the broad application and prospects of object detection models in agricultural disease.

3. DEEP LEARNING BASED DISEASE DETECTION METHODS

The one-stage and two-stage approaches are the most commonly applied deep convolutional neural network approaches for object detection. Researchers have focused their efforts in recent years on growing larger networks by inventing more complicated neural networks, constructing more extensive loss functions, and developing augmentation methods to mimic real-world data. Generally, the development of object detection has two periods, which are traditional detectors based on handcrafted features and deep learning model period. Before deep CNNs were widely used in detection systems, the design of the detection method relied much on the imaginative idea of handcrafted features. Figure 5 shows the following analyzed models are arranged in chronological order.

3.1. Detectors before deep learning

Viola-Jones Detector: Paul Viola and Micheal Jones proposed an object detection method in 2001 for face recognition, which is configured with sliding windows for seeking Haar-like features from the detection image essentially\cite{97}. The Haar features are squared shape kernels produced by Alfred Haar, similar to computation kernels in CNN. The features contain vertical and horizontal line features, edge features, and features for four rectangles in order to fit the line patterns of human facial contours in gray-level images. The method deploys cascade detection stages to reduce the computing cost and increase detection accuracy. Although the Viola-Jones detector looks like a simple structure, it enormously decreases the running time compared to algorithms from the same era.

HOG Detector: Initial proposed by Dalal and Triggs in 2005\cite{98}, the Histogram of Oriented Gradients (HOG) is a feature descriptor primary objective for pedestrian detection. It has a significant improvement in balancing scale-invariant feature transform and the non-linearity in distinguishing objects at its time. The detection steps include: calculating horizontal and vertical gradients, respectively, generating a histogram for each $8 \times 8$ cell on the processed image, and visualizing the HOG descriptors on images.

DPM: A variety of enhancements was proposed by Felzenszwalb et al.\cite{99} in 2008 to detect an object in a divide and conquer procedure that recognizes each part of a target. This detector comprises a root filter and several part filters where all configurations of component filters may be automatically learned. The construction of the object detection model reached a plateau after some updates on these modules\cite{100,101}. Since 2012, the object detection field has witnessed a rebirth of CNN and the explosive creation of deep neural network based detection models. The mainstream approaches can be divided into two parts: one-stage detection and two-stage detection.
3.2. Two-stage detection

The convolutional network has experienced a period of spurt growth since Krizhevsky et al. \cite{102} designed a deep neural network called Alexnet and outperform all the other traditional method in Imagenet large-scale vision recognition challenge in 2012 \cite{103}.

R-CNN: Researchers have been struggling to manually build information filters for years until Grishick et al. \cite{7} released a network generating regions of CNN features (R-CNN), which is a precursor of the DNN objects detecting model. Using selective search, the R-CNN model proposed a bunch of object recommendations. The generated proposals will be re-scaled to fix-size feature maps, followed by information features extracted by multiple pre-trained CNN kernels on ImageNet. Ultimately, SVM classifiers present the confidence and classification scores for each region. Despite outperforming previous machine learning methods impressively, this model has several flaws. First of all, the large number of proposed regions need sufficient memory units in computation hardware and waste a substantial portion of time for an integrated step of CNN forward passes, respectively. At the same time, some of the bounding boxes may have big junction areas. Furthermore, the SVM method is responsible for classifying each hypothetical object at the end after CNN extracts the characteristics. Because the individual composition models must be trained separately, the end-to-end training mode is impossible in this DNN model. Additionally, when facing large-scale data, SVM models are not able to make classifications through the inconspicuous margins between classes.

Fast R-CNN: Girshick had once again improved the method and came up with a Fast R-CNN model \cite{38} and overcame the disadvantages in R-CNN in 2015. A Region of Interest (ROI) network layer is built to obtain fix size feature maps from different proposed regions. This method is quite similar to the max pooling layer, except it uses a $2 \times 2$ grid and applies max pooling on each portion in four divided grids, regardless of their differences in size. The greatest contribution of this model is that it enables us to train a bounding box regressor and a detector in a single network architecture, which reaches 58.5 percent of mAP on the VOC 2007 dataset \cite{104}.

Faster R-CNN: In 2015, Ren et al. \cite{37} created a Region Proposal Network (RPN) and configured this network into Fast R-cnn to design the first network with the capability of performing end-to-end training on an object
Figure 6. Architecture of Faster RCNN. It has four loss outputs for the refine of RPN and the final prediction.

detection task. The mechanism of the detector can be parted into two functions. The first portion, named classifier, calculates confidence of the proposed regions, which can be treated as a likelihood whether there is an object in the region. The other part is the regression module tells where the proposal is located in the coordinate system. In this model, nine anchors are selected for each sliding window. The design of FPN reaches an extremely low cost of region proposal, which allow the entire system to perform a real-time detection process. There have been some efforts made to improve the computational complexity in Faster R-CNN [105]. Many new algorithms innovate and improve upon the foundation of Faster R-CNN. The Figure 6 shows the architecture of Faster RCNN.

Cascade, Mask and Libra R-CNN: Some models are built based on Faster R-CNN and optimized with technical inventions, which attract attention and arouse discussions among people. Cai et al. [106] design a cascade architecture based on Faster R-CNN and name it Cascade R-CNN. The generated object proposals will be sent to the main structure, where Intersection over Union (IoU) score between proposals and ground truth objects will be calculated. The concept of IoU has been widely accepted in object detection, and some loss functions are calculated based on this equation. Typically, the IoU threshold decides whether a detected sample is positive or negative. When the threshold is set too high, the number of positive suggested bounding boxes becomes extremely small, resulting in an over-fit problem. Therefore, the cascade architecture used a multi-stage detector with an increasing IoU threshold, where the output from the last stage fits as the input for the next stage, which is called an iterative bounding box in the paper. Another reason Cascade R-CNN performs so well is that various detector heads are designed at different stages. Their findings indicate that the cascaded architecture can be applied to other models and enhances Average Precision (AP) for various backbones and detectors in the COCO dataset [107]. Another work presents an improvement on the RoI pooling layer in Faster RCNN. He et al. [48] devise Mask RCNN model and introduce RoI Align method to solve the problem in RoI pooling, where the RoI pooling has two times quantization and results are rounded at pooling. RoI Align corrected the deviation of the regressed position by using bilinear interpolation. Besides that, a mask branch is added to the network to make a prediction of binary mask for each pixel, which enables the model to produce a segmentation mask for each object. Pang et al. [108] improve R-CNN by modifying the pyramid architecture. The highlight of Libra R-CNN model is the process of multi-level feature fusion. The model firstly generates four scales of feature maps, and then they are reshaped for integration and refinement. Finally, the results are added back to the original features to enhance the original features.

FPN: Most of the proposed methods process their detectors on one extracted feature from the end of CNN, which may lose the characteristic of small objects in the images. However, the feature maps from the top layer of the network contain coarse location information, which leads to inaccurate predictions of bounding
boxes. A pyramid structure solves this issue by running detectors on feature maps from multiple scales. The Feature Pyramid Network (FPN) contains a top-down architecture, where the lower level of feature maps are generated from the lateral layer after a 1 × 1 convolutional kernel and the up-sampling results from the top-level information. This technique enables FPN to create high-level semantic features at all scales. For the tremendous upgrade, FPN becomes a basic structure for some latest networks.

3.3. One-stage detection

Several two-stage detection techniques are given above, where the models only require scanning the pre-proposed interesting regions rather than the complete image. People benefit from the accuracy of two-stage methods but criticize the long processing time at training and inference. On the contrary, one-stage detector attain all bounding boxes at one-time inference, which are likely to be employed in a mobile device and a real-time system. Basic one-stage networks predict the results on a grid defined on the input image, and no intermediary recommended regions will be generated. For incipient one-stage networks, images are separated into a n × n grid, and detectors perform the process on that grid. The feature extraction part will be pre-trained on various huge datasets to make the model converge faster, and then the last few layers’ weights will be removed. At the top of the network, the models can generate low-resolution feature maps for classification tasks, and each cell on the map represents the object’s prediction located in a specific portion of the grid. The classification and regression information will be stored through the channel dimension in that cell. Multi-objects appear in the image leading to multi times matching in the network. The neural network model will generate a designed output with 5 + C channel for each detected object. The first five channels t_x, t_y, t_w, t_h, P_obj stands for four characteristic of the location and the confidence of whether it has an object, and the others illustrate the score for each category in the task.

YOLO: The origins of these onetime processing networks is the You Only Look Once (YOLO) model. YOLO is a ground-breaking object identification approach that first makes regression on the item’s position in real time. Redmon et al. designed DarkNet-19 for the YOLO network, including several 3 × 3 and 1 × 1 convolutional layer alternatives. Five Max pooling layers downscale the image by half each time from 224 × 224 to 7 × 7. In their work, they train their network on PASCAL VOC 2007 challenge data that has twenty categories, and the grid divides the images is designed as 7 × 7. Additionally, two bounding boxes are signed to a grid cell. So, the prediction outputs from YOLO’s network should be 7 × 7 × (2 × 5 + 20). All layers are added with a Leaky ReLU activation function for the final layer. However, it is conceivable for one item to have a bounding box in each grid cell. An approach known as Non-Maximum Suppression (NMS) solves this problem, allowing the maximum existing and suppressing the rest. The main goal of NMS is to eliminate bounding boxes with a lot of overlap and maintain the one with the highest confidence level. The algorithm will first choose the prediction with the highest percentage of ground truth label overlap. Then, a threshold is used to remove detection with a high IoU from the selected prediction when several bounding boxes are created. This algorithm will operate on each category independently to remove the redundant. Although Faster R-CNN outperforms YOLO regarding mean Average Precision (mAP) score, the latter has a significantly higher Frame Per Second (FPS) rate.

YOLO Series: A series of improvements are made to the basic model for a lower cost on computation and parameters and higher accuracy on data validation. In YOLO v2, the author has devised a few strategies for dealing with miss-localization issues. Batch Normalization (BN) is applied after each convolutional layer in the network. The BN layer calculates the mean and variance values across the training batch and normalizes the data into a standard distribution. Secondly, they pre-train their model on a higher resolution dataset with 224 × 224, which results in an improvement of 4% in mAP. Redmon et al. improve their YOLO model into a third version named YOLO v3. Figure 7 shows the main architecture of YOLO v3, where several convolutional blocks are illustrated. To design the best aspect ratio of anchor boxes, an unsupervised learning method, K-means clustering, is trained on the VOC and COCO datasets. The backbone network in this architecture
uses the skip connection approach from He et al.\textsuperscript{115} called DarkNet-53 but has a faster training time than ResNet-101. Instead of the max pooling layer, the author uses a two-times stride in the convolutional layer. Inspired by FPN, three distinct scales of detection are made in YOLO v3, and three distinct detecting heads generate predictions based on three scales, each with a two-times difference. Later, YOLO v4 and YOLO v5 are released in succession. YOLO v4\textsuperscript{42} and YOLO v5\textsuperscript{68} both leverage the Mosaic data augmentation method and cross stage partial (CSP) architecture\textsuperscript{116}, while YOLO v5 use self-adaptive anchor boxes and the other not. In one of the most powerful object detection networks, YOLO v7\textsuperscript{117}, in this year, the authors define a detector in the middle and adhere to a strategy of going from coarse to fine and employ gradient flow propagation paths to determine which network modules should and or not employ re-parameterization procedures. Reparameterization’s function is to speed up the network while preserving model performance. This is mostly accomplished by fusing batch normalisation (BN) layers with convolutional layers and by integrating many convolutions into a single convolution module.

**RetinaNet:** Another well-known model in a one-stage camp is RetinaNet, proposed by Tsung-Yi et al.\textsuperscript{118}. Similar to YOLO v3, ResNet\textsuperscript{115}, and FPN is considered. Nevertheless, RetineNet contains five distinct detectors at each scale, and the anchor boxes are set at three different aspect ratios (1:2, 1:1, 2:1). The main contribution in this paper is the newly designed loss function. The presented loss function attempts to address the imbalance ratio of positive and negative samples, where the scaling factor might raise the weights of items that is difficult to detect in the final loss function. The formulation is introduced in the comparison section. The RetinaNet at its time outperformed all two-stage networks in accuracy and speed.

**CornerNet and CenterNet:** While the above-mentioned methods are capable of real-time detection, it’s important to note that the quality of anchor box design plays a pivotal role in determining the upper limit of detection algorithm performance. This aspect often presents challenges in parameter tuning. A solution to this complexity is presented in CornerNet\textsuperscript{119}, which eliminates the need for intricate anchor box setups through the use of a Corner Pooling layer. This layer calculates the sum of the maximum values encountered horizontally to the right and vertically downwards from the feature points. CornerNet recognizes an item as a pair of crucial points in the enclosing box’s top-left and bottom-right corners and uses a position encoding to match two points belonging to the same object. CornerNet surpasses most of the one-stage methods and produces comparable results to two-stage detectors, where mAP on COCO equals 57.8%. In\textsuperscript{120}, Duan et al. analyze the reason that causes generating a large portion of wrong bounding boxes when using a low IoU score and assume CornerNet architecture has limitations in perceiving information within the bounding box. Therefore, CenterNet focuses on detecting three essential points of an object with a bottom-up detection method, which
Table 1. Comparison of 9 object detection models in ascending order based on precision

<table>
<thead>
<tr>
<th>Models</th>
<th>Backbone</th>
<th>Evaluation (AP50)</th>
<th>Benefits</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faster RCNN</td>
<td>VGG-16</td>
<td>42.1</td>
<td>Use RPN to generate a proposal and replace Max pooling with RoI.</td>
<td>RoI pooling makes a loss of translation invariance in the feature representations.</td>
</tr>
<tr>
<td>CornerNet</td>
<td>Hourglass-104</td>
<td>57.8</td>
<td>Eliminate the design of anchor boxes and related hyperparameters.</td>
<td>May have a mismatch in pairing two corners for one object.</td>
</tr>
<tr>
<td>YOLO v3</td>
<td>Darknet-53</td>
<td>57.9</td>
<td>Maintain high processing time with DarkNet compared to ResNet.</td>
<td>Struggling when detection data contains an uneven distribution of object classes.</td>
</tr>
<tr>
<td>RetinaNet</td>
<td>ResNet-101</td>
<td>59.1</td>
<td>Introduced focal loss function for the imbalance between positive and negative samples.</td>
<td>Localization accuracy may not be as precise as two-stage detectors.</td>
</tr>
<tr>
<td>Mask RCNN</td>
<td>ResNet-101</td>
<td>60.5</td>
<td>Introduce RoI Align for misalignment issues and pixel-level masks.</td>
<td>Overlook the spatial information differences between different receptive field sizes and hard-to-segment objects with intricate boundaries.</td>
</tr>
<tr>
<td>Cascade RCNN</td>
<td>ResNet-101</td>
<td>62.1</td>
<td>Cascaded stage with different IoU to reduce over-fitting.</td>
<td>Difficulty in finding threshold and increased computational complexity.</td>
</tr>
<tr>
<td>CenterNet</td>
<td>Hourglass-104</td>
<td>64.5</td>
<td>One-time feed-forward at inference, no non-maximum suppress operation needed.</td>
<td>Hard to detect objects with overlapped center.</td>
</tr>
<tr>
<td>DETR</td>
<td>ResNet-101</td>
<td>64.7</td>
<td>Achieve better performance on large objects with global attention, fewer parameters, and floating-point operations per second.</td>
<td>Not good at small objects and need much more time to converge.</td>
</tr>
<tr>
<td>YOLO v5</td>
<td>CSPDarknet-53</td>
<td>65.7</td>
<td>Design CSPDarknet to gain higher extraction capabilities and employ PAN structure for handling small objects.</td>
<td>The architectures bring in too large computation complexity for real-time detection.</td>
</tr>
</tbody>
</table>

DETR: After self-attention-based models revolutionized tasks like machine translation and sentence generation in natural language processing, there emerged a growing interest in applying this powerful approach to computer vision. The paper ‘Attention Is All You Need’ introduced a Transformer model,[121] fully relying on the self-attention model, which obtains global information. In 2020, the DETR model,[122] was proposed, which adopts the Transformer architecture as both an encoder and decoder. It does so after generating sets of image features through a CNN backbone network. During this process, the model flattens the 2D information maps and feeds them into a Transformer encoder equipped with positional encoding. This innovation marked a new era for computer vision, leading to further modifications and advancements based on the DETR model.[123,124]

3.4. Deployment of detection models

When a model is well-fitted and demonstrates excellent performance on a test set, the goal is to apply it for real-time plant disease detection. Most research efforts concentrate on the earlier stages of the machine learning lifecycle, including data processing, model selection, and model assessment. However, model deployment, despite its complexity, is a topic that often receives less attention. Deployment typically requires a solid background in software engineering and embedded design. In the context of object detection models, two-stage detectors are developed to achieve higher precision but trade off inference time for prediction accuracy. Consequently, most efforts aimed at deploying object detection algorithms opt for one-stage methods as the primary detection module in an engineering system. Researchers like [57] build a detecting mobile app for timely detecting HuangLongBing in citrus orchards. Huang et al.[125] propose a object detection model named CoDeNet and deploy it on field programmable gate arrays. Another group[126] use YOLO v5s as their detector and apply it to real-time detection of melon leaf diseases system is implemented. In this work, the size of deep learning model is only 1.1 MB, and inference time reach 13.8 ms. Moveover, Sange...
train an Inception v3 network on their banana diseases data, and get result of over 96% accuracy. The model is deployed to a mobile device and can perform real-time classification with low memory cost. For applications that are released and can be downloaded from app store, an mobile application is built to recognize plant diseases, which can be an alert system in agriculture cultivation. Plantix is a mobile application helping farmers to diagnose and treat crops problems. Many applications can be found in app stores, claiming to recognize and provide suitable treatment suggestions. However, according to a study by Siddiqua et al., out of the 600 mobile apps evaluated, only Plantix was successful in identifying plants from images and detecting diseases. The remaining apps fell short in this regard. It’s worth noting that deep learning object identification models have not been the subject of extensive research thus far, leaving room for further exploration and improvement.

3.5. Comparisons of detection models and evaluation methods

Comparing object detecting models as well as choosing suitable assessment techniques is a crucial component of building a diseases recognition system. Various models provide trade-off between processing speed and prediction accuracy. The choice of evaluation methodologies depends on the analysis of application’s unique characteristics. Each deep neural network is invented to reach state-of-the-art results on evaluation. Experiments have been done to validate the various loss functions claimed converge speed. The performance can be evaluated using loss function and evaluation metrics, where the loss function can be used to assess the model’s performance and the size of the discrepancy between predictions and targets in training process. Evaluation metrics can present the evaluation findings on the test dataset once the model has been properly fitted.

Evaluation Metrics: Evaluation metrics are employed to assess the performance of a trained object detection model. Typically, these metrics are not used for the optimization process during neural network training. Instead, they serve the purpose of model comparison and gauging how effectively the model performs on unseen data. The IoU score is introduced to measure the overlap area of bounding boxes. Models such as Faster RCNN, Mask RCNN, and Cascade RCNN utilize IoU loss to refine their bounding box predictions. IoU represents as

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} = \frac{A \cup B}{A \cap B},$$

where A stand for the model proposed region and B is the labeled or truth area of the object. IoU is frequently used to assess a detection box’s quality, and a threshold may be established to categorise it as a positive or negative prediction. The fitting level of predicted bounding boxes are shown by Precision and Recall, which can be described as follows respectively:

$$Precision = \frac{\#\ true\ positives}{\#\ predicted\ positives},$$

$$Recall = \frac{\#\ true\ positives}{\#\ actual\ positives}.$$ 

Precision can be calculated at different IoU thresholds. Currently, most object detection challenges and evaluation processes employ mean Average Precision (mAP) as a method to assess a model’s performance on the test set. When generating predictions, different IoU thresholds result in different numbers of detection boxes. A high threshold can enhance the accuracy of model detection but significantly reduce the recall score. For each detection category, the model computes the Area Under the Curve (AUC) using various IoU thresholds to evaluate the model’s capabilities. The Precision-Recall AUC is theoretically determined by integrating the area beneath the precision-recall curve. It quantifies the model’s performance across different IoU thresholds and can be described as

$$PR-AUC = \int_0^1 prec(rec) d(rec).$$
where $r_e$ is the recall score and $pr_e(r_e)$ is the corresponding precision value at a recall value. Typically, AP values are calculated discretely, and this calculation can be thought of as a discrete version of the Area Under the Curve (AUC). Interpolation is a common technique in this process, where precision scores are selected at 11 interpolated recall points for computation. The average of these precision values is then considered as the Average Precision (AP) score. Based on this calculation, the model derives a mean value across all categories to determine the final mAP score. In the context of the COCO challenge, recall values are chosen within the range of 0.5 to 0.95 with an interval of 0.05. While metrics like Average Recall (AR) and AP with object size thresholds are also relevant, AP remains the primary challenge metric.

**Loss functions:** In machine learning, functions such as mean absolute error, mean square error, and Huber loss are commonly employed and compared. Similarly, in object detection, the choice of loss function holds significant importance as it guides the model toward better data fitting. During training, a multitude of annotated images are fed into an object detection model, and its internal parameters are updated iteratively as it processes each batch of images. For the classification problems, Faster RCNN and YOLO both use Cross Entropy (CE) loss for each bounding box prediction, where the binary CE can be described as:

$$L_{CE} = \begin{cases} -\log(p), & \text{if } y = 1; \\ -\log(1 - p), & \text{if } y = 0 \end{cases}$$

(5)

where $p$ stands for the probability of sample for positive classification. If the number of classes is greater than 2, the loss function can be illustrated as:

$$L_{CE} = -\sum_{i=1}^{N} y_i \log(p_i)$$

(6)

where $y$ is the one hot value for each label, and $t$ is the category index. However, the CE loss struggles for optimization when data has an imbalance category amount. Focal loss addresses this problem by reducing the weight of easy examples with a large number of samples. In RetinaNet, the Focal loss is based on CE loss, which calculates the classification loss and regression loss separately. Additionally, the author introduces a dynamic scaling factor in the $C_E$ loss. The Focal loss can be described as

$$FL(p_t) = -(1 - p_t)^{\gamma} \log(p_t) = \begin{cases} -(1 - p)^{\gamma} \log(p), & \text{if } y = 1 \\ p^{\gamma} \log(1 - p), & \text{if } y = 0 \end{cases}$$

(7)

where $\gamma$ is the hyper-parameter to smoothly adjust the rate.

The other part of the loss function measures the location of bounding boxes. The total backward loss function can be represented as the combination of classification and localization loss. Based on the calculations of IoU, the most popular and fundamental loss–IoU loss, which can be defined as

$$L_{IoU} = 1 - \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|},$$

(8)

where $B$ is the prediction bounding box and $B^{gt}$ is the ground truth box. There exists some disadvantage in IoU loss, which has zero loss value when there is no intersection between prediction and ground truth. In [132], Rezatofighi et al. improve the IoU loss for the sensitivity imbalance to objects with different scales and the zero loss issues by involving distance calculation of two center points. YOLO v4, v5, and DETR also use it as a primary localizing loss function. The Generalized Intersection over Union (GIoU) is proposed as follows:

$$L_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|},$$

(9)

where $C$ represents a box exactly covering the prediction and the truth box. By measuring the area of the covered box, the algorithm can measure the distance when the two boxes do not overlap. However, when the
prediction is within the objects and covered by the truth box, the GIoU loss no longer performs well. Distance Intersection over Union (DIoU) loss \cite{133} resolve the problem by computing the Euclidean distance of two center points between the prediction and target box. YOLO v4 and v5 algorithms explore DIoU loss as well. The expression is illustrated in following formula:

$$L_{DIoU} = 1 - IoU + \rho^2 \frac{(b, b^{gt})}{c^2}, \quad (10)$$

where $\rho$ is the Euclidean equation, $b, b^{gt}$ are the center of the predicted boxes and ground truth box respectively, and $c$ is the diagonal length of two boxes. Compared to IoU loss, this method forces the model to converge quickly, and the bounding box will move rapidly toward the object.

Several object detection models have been harnessed for plant disease detection. In Table 1, we have selected widely used and representative models, comparing their evaluations on the MS-COCO dataset using the Average Precision (AP) indicator at a 0.5 Intersection over Union (IoU) score as a reference for their applicability to plant disease data. The models are arranged according to their performance from Faster R-CNN to DETR, where Faster R-CNN represents the baseline for developing improvement, and DETR can be regarded as the prior Transformer detector based detection model. In this section, we also delve into the merits and limitations of these models, considering aspects such as innovations, computational costs, time efficiency, and overall performance.

4. CHALLENGES AND FUTURE TREND

Despite encouraging results on some datasets from conventional deep learning methods, there are still some restrictions for applying CNN methods to plant disease detection. Figure 8 is add for the sake of readabilty.
4.1. The subtlety and similarity of image features

In plant disease detection, the precision requirement increases with the evaluation of the technique. The imagery features may vary at different stages of infection. In the early stages of a plant infection, like inoculation and penetration, the pathogen invades the plant tissue and grows. No visible signs or traits are found on the plants at this time. Another difficulty in disease detection in precision agriculture is the similarity of plant disease. Some disease pathogens like fungi and bacteria can cause cankers or dark spots on leaves, making recognition from the subtle distinguishes more challenging. Besides the similarity, the coexistence of several diseases at once is a common situation for infection. Similar disease species can sometimes live on the same host due to small niche differences\[134\]. These coexistences make it tougher to make a classification. Almost all target identification techniques rely on exhibiting disease traits on infected plants. Then neural network based detection approaches are used to learn these extracted features for identification and detection. Precision farming aims to make efficient treatments for certain diseases. Management strategies like chemical and biological control can affect some closely related pathogens. Otherwise, the abuse of biological and chemical reagents will lead to potential severe contamination or health and safety risks.

To address these challenges, several techniques can be employed. One approach involves the use of multi-spectral and hyper-spectral image sensors, which capture image data with higher dimensions. This enhanced data can provide more detailed information for analysis. Another strategy is to employ small object detection methods, which can break down the appearance of disease symptoms into smaller features, enabling more precise classification. The issues related to the ambiguity of neural network models’ handling capacity may be attributed to inaccurately classified data. Data-centric intelligence approaches are emerging as promising methods in the development of large language models, with a focus on improving both the quantity and quality of the data used in model training. In data-centric AI, the model remains relatively fixed, while researchers concentrate on curating high-quality data. This method has seen impressive results on natural language processing, where GPT-3\[135\] outperforms GPT-2\[136\] by employing higher quality dataset with only lightly tuning the model. Inspired by this improvement, Wang et al.\[137\] investigate the impact of using data-centric experiments for fruit detection. Although this work still shows some restrictions of this research method, great potential, and possibilities can be expected in future works. Researchers start to seek the possibility of deploying large vision model, after seeing success of large language models. In 2023, Meta release a large segment model named SAM\[138\], which has capability of using quick engineering to tackle a variety of downstream segmentation jobs on unseen data. However, the effectiveness of such models in complex environments or for specific intricate tasks still requires validation. As a result, efforts continue to enhance the detection performance of large vision models like SAM.

4.2. Environmental impact on agriculture data

Similar plant features may differ significantly from data collected under various environmental and weather conditions in images, especially in UAV imagery data. Different circumstances like humidity level, illumination temperature, and foggy weather would influence the accuracy of recognition in disease diagnosis. In dim light conditions, the requirement of sensitivity to the image sensor is increased. Otherwise, it will reduce details and introduce noise in plants’ physical form. Some noise algorithms are brought in to imitate the real-world situation\[139,140\]. Olusola et al. use image blur methods like Gaussian blurring and motion blur for data preprocessing to reduce the image quality to enhance the detection model on cassava leaf diseases. Fog is another important factor that deters the model from learning important disease characteristics. Fog develops when the moisture level reaches the ground-level condensation point, which shortens the data collection period. What makes the defogging processing tougher is the scattering of various fog particle sizes according to the imaging mechanisms\[141\].

Techniques for resolving these difficulties are evolving. Some researchers have been working on designing de-fog algorithms for de-weathering issues. A multi-scale depth fusion method is presented for defog on images
with optimization method to search solution in the depth map \[^{142}\]. A hybrid filter method combined with median and guided filtering is proposed to increase the robustness in defog methods \[^{143}\]. Zeng et al. \[^{144}\] proposed an illumination-adaptive network for person detection, which is able to eliminate the impact of illumination discrepancy. Besides using operators and filters, some generative-based approaches are applied \[^{145-148}\]. Although these works enhance the performance of CNN models on foggy images, there is still room for improvement. One possible way is to propose fog imitation learning as an augmentation in representing the characteristics. Currently, de-weathering algorithms mainly focus on optimizing specific aspects or directions, such as de-hazing or illumination-adaptive algorithms. More comprehensive models need to be designed to address more negative factors at once. A possible way is to enable the image enhancing adaptively by using an image processing module with several designed filters, which shows great results combined with YOLO v3 for object detection in problematic weather conditions \[^{149}\]. Thus, there is reason to believe that the fusion of the adaptive processing method and deep CNN has a promising future.

### 4.3. Spatial domain issues

Deep learning and object detection methods have found widespread applications in agriculture recognition. However, a common challenge arises when these models are trained on data collected from specific locations, leading to performance deterioration when applied to new regions. This challenge is exacerbated by the variability in environmental conditions and terrains across different areas. For example, different regions may employ varying planting densities due to geographical factors, and the growth patterns of plants with unique characteristics or species can further contribute to performance variations. Additionally, in many agricultural applications, the size of the testing dataset often significantly surpasses that of the training dataset. This imbalance is frequently attributed to the high labor costs associated with annotating large datasets, which are crucial for training highly accurate models. The resulting dataset imbalance can lead to reduced test accuracy when attempting to make inferences from previously unobserved data.

Several research efforts have aimed to mitigate the challenges posed by domain shift and imbalance, with domain adaptation emerging as a prominent and trending research area. In the work by Hsu et al. \[^{150}\], an intermediate data domain is built to brighten the gap between the source domain and unlabeled data and gradually complete simpler adaptation sub-tasks. Addressing the issues in a teacher-student framework, the proposed model in \[^{151}\] leverages adversarial learning and applies weak-strong augmentation and mutual learning between the student and teacher models. Faster R-CNN is selected as the base model. Some researchers have applied domain adaptation techniques in plant disease detection, trying to increase accuracy by domain shift. Alvaro et al. use open set domain adaptation on data from three farms with CNN models for tomato disease detection and obtain over 90 percent precision \[^{152}\]. A new unsupervised domain adaptation method is devised in \[^{153}\], using uncertainty regularization for cross-species disease recognition problems. While domain adaptation methods have shown promise in improving model accuracy in target domains, certain challenges persist. Some of these methods still rely on deep learning models proposed several years ago as their base, necessitating the generalization of frameworks to encompass newer detection models like the YOLO series and anchor-free detection models. Moreover, these methods often assume the availability of a substantial amount of unlabeled target data, which may not always hold true. Class imbalance issues can also arise in the target domain. Furthermore, the target domain may not perfectly align with real-world data, complicating adaptation efforts. Some widely used domain adaptation benchmarks like \[^{154,155}\] may exhibit distribution variances compared to actual application scenarios, highlighting the need for strategies to narrow the gap between test data and real-world conditions.

### 5. CONCLUSIONS

Plant disease recognition has presented a complex challenge in the field of neural network prediction. The impressive discriminative capabilities exhibited by neural networks have attracted a growing number of re-
searchers and practitioners to embrace deep learning models for agricultural engineering applications, particularly in the context of plant disease detection. Given the extensive body of research in this domain, this study aims to narrow its focus to common plant disease detection issues within agriculture. This manuscript offers a comprehensive survey that delves into the application of deep neural network object detection for disease recognition, tracing the historical development, and highlighting technological advancements in object detection models.

The timely and accurate detection and recognition of plant diseases from images constitute crucial tasks for plant disease detection systems. However, the diverse characteristics of disease symptoms can pose significant challenges for deep neural network models attempting to glean essential features from image data. In response, sophisticated object detection algorithms have emerged, designed to extract features from aerial or ground photographs. Some network architectures, such as pyramid layers, have proven valuable in capturing information from objects of varying sizes within images. The evolution of techniques that offer improved cost-effectiveness and time efficiency has greatly facilitated the implementation of real-time detection systems. Additionally, methods like data augmentation and domain adaptation are leveraged to address issues related to inference on unlabeled data. Furthermore, this review explores the interdisciplinary intersection of plant disease recognition, deep learning, and object detection, shedding light on the challenges associated with real-world disease recognition inference.

DECLARATIONS
Authors’ contributions
Made substantial contributions to the research and investigation process, reviewed and summarized the literature: Zhou Z, Zhaohui Gu, Yue Zhang, and Zimo Zhou. Collaborated on writing the article: Zhou Z, Zhang Y, Gu Z. Leadership responsibility, commentary, and critical review: Yang SX.

Availability of data and materials
Not applicable.

Financial support and sponsorship
This research was funded by the Natural Sciences and Engineering Research Council (NSERC) of Canada.

Conflicts of interest
All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Copyright
© The Author(s) 2023.

REFERENCES
15. Hughes D, Salathe M. An open access repository of images on plant health to enable the development of mobile disease diagnostics. arXiv. org; 2020. DOI


