A Smart Agriculture Framework to Automatically Track the Spread of Plant Diseases using Mask Region-based Convolutional Neural Network

Alakananda Mitra[1] [0000−0002−8796−4819],
Saraju P. Mohanty[1] [0000−0003−2959−6541], and Elias Kougianos[3] [0000−0002−1616−7628]

[1] Department of Computer Science and Engineering, University of North Texas, USA.
alakanandamitra@my.unt.edu, saraju.mohanty@unt.edu
[2] Department of Electrical Engineering, University of North Texas, USA.
elias.kougianos@unt.edu

Abstract. Plant diseases reduce agricultural production. They negatively affect fruit and crop quality and reduce yield, causing food shortages. A drop in production harms the global agricultural economy. However, early detection and disease severity estimation are key to disease management, containment, and prevention. Damage localization is the first step in estimating the severity of diseases, which is crucial for the optimum application of pesticides. The current approach needs expert advice for disease detection. For a large farm, it is an expensive and slow process. Automatic plant disease detection eliminates the tiresome task of monitoring big farms and detects the disease early enough to avoid plant degradation. In this article, we propose a fully automated method based on deep neural networks for detecting and localizing leaf diseases. The proposed method is based on Mask R-CNN network. Image augmentation has been performed to achieve higher precision from a small dataset. Transfer learning has been used to save time and achieve better performance. Our proposed method of disease detection is faster, as it automatically localizes the disease along with the disease identification from the leaf images. The images can be taken using a smart phone camera or a low altitude unmanned aerial vehicle (UAV) camera. As a case study, we have applied the method to apple leaves.

Keywords: Smart Agriculture · Smart Villages · Internet of Agro Things (IoAT) · Plant Health · Plant Disease · Apple Leaves · Mask Region-based Convolutional Neural Network (R-CNN) · ML from Small Dataset.

1 Introduction

The presence of plant diseases has a detrimental effect on the amount of food that can be produced through agriculture – it affects the crop quality and reduces the final yield. Each year crop losses total billions of dollars [26]. If plant diseases are not identified and addressed in a timely manner, it results in increased food insecurity [22]. Hence, early detection and disease severity estimation are the two major steps for disease management, containment, and prevention.

Trees are prone to various fungal pathogens that cause diseases. These diseases can affect plants at any stage of growth and manifest in a variety of plant components, from stems to fruits. Symptoms may include discoloration, form change, wilting, galls, and cankers. However, as disease symptoms are predominantly manifested on leaves, most of the research on identifying plant diseases is focused on leaves or fruits [22].

In developing countries, manual observation is still the most common method of detecting plant diseases. It is an arduous and inefficient process that consumes a lot of time. It also requires expert services, but farmers are not always able to afford such expensive services [33]. Wrong identification and improper use of pesticides cause secondary damage to the plants and contaminate the soil as well as the environment. To solve these problems, different techniques based on computer vision and deep learning have been proposed for the automatic and accurate detection of plant diseases.

Plant monitoring is crucial for disease management. The early detection of plant disease and its prevention by 2030 are two key goals of agricultural research [29]. Here, a fully automatic plant disease detection method is proposed and illustrated in Fig. 1. As a case study, apple leaves have been selected. The proposed method will track the spread of the disease by identifying the disease and detecting the damaged areas. The novelties of the work are as followed:

- The method is fully automatic. No expert service is needed for disease detection.
- Very little effort is needed from the users' side. Users only have to take pictures of the damaged leaves.
- Early detection of the disease is possible.
- This process is the first step of disease severity estimation. Estimation of disease severity plays a pivotal role in calculating the optimal quantity of pesticides.

Continuous monitoring results in the early detection of disease. With further experiments, our proposed method can be used to estimate the damage severity. We aim to save time, money, resources, organisms vital for soil and biodiversity, and to store carbon in the soil to combat climate change [4].

The rest of the paper is organized in the following way: Section 2 discusses recent work on plant disease detection. Section 3 presents an overview of the method and network architecture. Experimental details are discussed in Section 4. Section 5 presents the results and evaluates the performance of the method, along with a comparative study. Finally, the paper concludes with future work direction in Section 6.

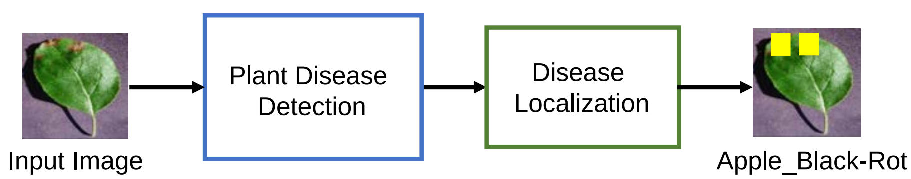


Fig. 1. Scope of the Paper

2 Prior Research Work

This section presents state-of-the-art methods based on deep neural networks for plant disease detection. Approaches based on convolutional neural networks (CNNs) are automatic and more efficient. Literature survey reveals that there are mainly two types of studies addressing plant diseases: either classification-based or regions-of-interest (ROI)-based.

2.1 Classification Based Approaches

The majority of classification-based approaches use convolutional neural networks (CNNs). These approaches are automatic and more efficient than the previously used machine learning methods, which rely largely on image processing [34]. Apple leaf disease was correctly identified in [35]. An attention network has been added with EfficientNet-B4 to incorporate both channel and spatial features. Image augmentation has been used over the collected data. High accuracy has been obtained.

Apple leaf disease has also been detected in [11] using a combined structure of DenseNet and XceptionNet as a feature extractor, and finally, classification has been done through a Support Vector Machine (SVM). Early detection of the disease has been the focus of this paper. A high accuracy of 98.82% has been achieved. Five kinds of apple leaf diseases have been detected using SSD with Inception and Rainbow concatenation [19] structure. The proposed model has achieved 78.8% mean average precision (mAP). In [10], an ensemble of pre-trained DenseNet121, EfficientNetB7, and NoisyStudent networks has been used to identify apple leaf diseases with an accuracy of 90%. The model has been deployed as a web application.

Various grape leaf diseases have been detected using a modified Inception structure [24]. A dense connectivity technique has been proposed with the Inception structure for better features. Data enhancement techniques have also been used to increase the dataset size. In the agricultural domain, no large publicly available datasets are found which demands data augmentation in most of the works. A slightly lower accuracy compared to the other papers has been achieved here.

Tomato leaf diseases have been detected using four different CNNs in [9]. InceptionV3 performed much better among the lot with laboratory data compared to field data. A custom shallow CNN has been used for nine different tomato leaf diseases in [8]. A good accuarcy of 91.2% has been achieved after training 1000 epochs. Similarly, a seven layer CNN structure has been used as feature extractor in [18] to detect four of rice leaf diseases. The features have been classified with an SVM classifier with high accuracy. Cross validation ensures the robustness of the model.

The majority of these CNN based methods identify the diseases but no localization of the disease has been performed. However, the severity of the disease can only be known when the damage area of the leaf is calculated. Localization is the first step of damage estimation.

2.2 Region-of-Interests (ROI) Based Approaches

Regions-of-Interest (ROI) deep CNNs are being used in recent plant disease detection works as these structures have the potential to segment the location of the disease. When deep neural networks are used in classification based approaches, no localization of the disease is performed. However, ROI based approaches detect the diseases along with localization. Nine types of tomato plant diseases and pests have been recognized in [14] using different regions-of-interest (ROI) based structures. Data augmentation has improved mAP maximum 30% in some cases.

A new structure DF-Tiny-YOLO has been presented in [13] for detecting apple leaf diseases. Here, use of smaller CNN kernels reduces feature dimensions and increases network depth without increasing the complexity. Another ROI-based structure, Faster R-CNN, has been used for recognizing rice plant diseases and pests in [20]. Relatively blurry videos have been used as input. A custom CNN is used as the backbone network. Higher accuracy has been achieved compared to existing structures.

There are certain articles where more than one ROI-based structures have been used. In [36], Faster R-CNN and Mask R-CNN both have used for disease identification and detection of the diseased segments. In [30], Mask R-CNN has been used to detect disease infected part of apple leaves whereas the disease has been classified with ensemble subspace discriminant analysis classifier. A hybrid contrast stretching method has been applied. Mask R-CNN has also been used in [7] for detecting strawberry diseases. A systematic approach to data augmentation has been followed, increasing the mAP to 82.43%. A strawberry disease dataset has also been presented here.

These papers indicate that various deep learning networks have achieved good success rates in identifying different plant diseases. However, more information e.g., severity of the disease is needed to control and provide solutions to prevent plant diseases.

3 Proposed Method

3.1 Overview: Proposed Agriculture Cyber Physical System

Plant disease is a serious concern for sustainable farming. Plant diseases are a farmer's worst fear since diseases can wipe out an entire crop and result in significant financial loss. The critical step to preventing plant disease is early detection. In this section, a smart agriculture [28] framework for automatic tracking of plant diseases is presented. The A-CPS is described in the context of apple leaf disease detection. Fig. 2 shows the proposed agriculture cyber physical system (A-CPS) [27] with *things*, *stakeholders*, and *networking*.

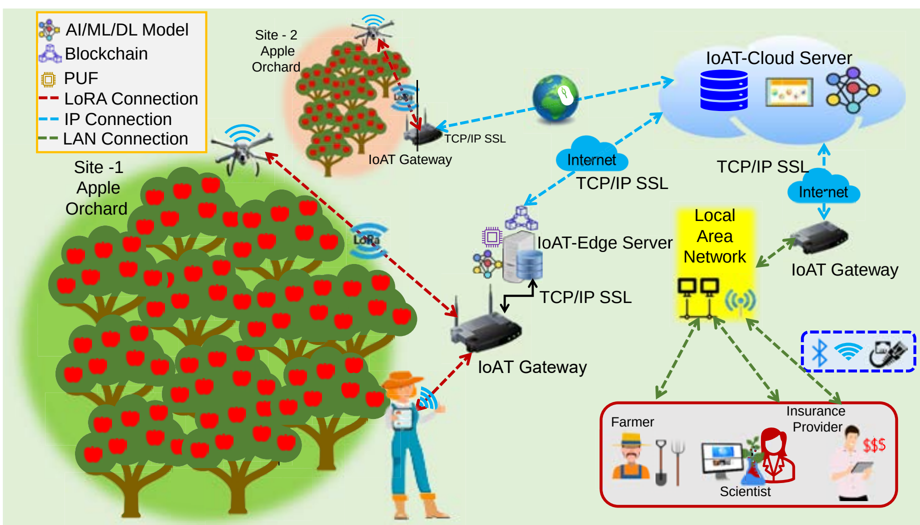


Fig. 2. System Overview: A-CPS for Plant Leaves Disease Detection. In this A-CPS, UAVs and smart phone cameras are the *things* and farmers, scientists, and insurance providers are the *stakeholders*. Various *networking* and communication options are used at different stages.

The proposed A-CPS is deployed in an edge-cloud setting in an apple orchard. In this A-CPS, farmers, insurance providers, and scientists are the main *stakeholders*. UAVs and smart phone cameras are the *things*. The application has the potential to perform at the edge along with the cloud platform.

Apple leaf images are collected from apple orchards. Low altitude Unmanned Aerial Vehicles (UAVs) or smart phone cameras are used to take the images. They are connected to the Internet-of-Agro-Things (IoAT) gateways through long-range and low-powered LoRA connections. When capturing with an UAV or a phone camera, many leaves will appear in the frame. As a result, before identifying damage to a single leaf, each leaf must be detected using object detection.

IoAT gateways are connected to the *edge server* and *cloud server* through TCP/IP SSL. The images are sent to the cloud server or edge server, if available. Then, the images are processed, diseases are detected, and damage areas are localized using the methodology mentioned in Section 3.2. Finally, the result is sent back to the user.

3.2 Methodology for Disease Detection and Localization

In this subsection, the detection method for plant disease is presented in detail. The process workflow is shown in Fig. 3. First, photos of the leaves are taken. Then they are resized to 256×256 to be detected using the trained model. A Mask Region-based Convolutional Neural Network (R-CNN) [15] has been used to detect the disease along with disease localization. Hence, the problem is considered as an object detection problem. Object detection is a task in computer vision that involves identifying the presence of one or more items in a given image as well as their location and the category of object that they belong to.

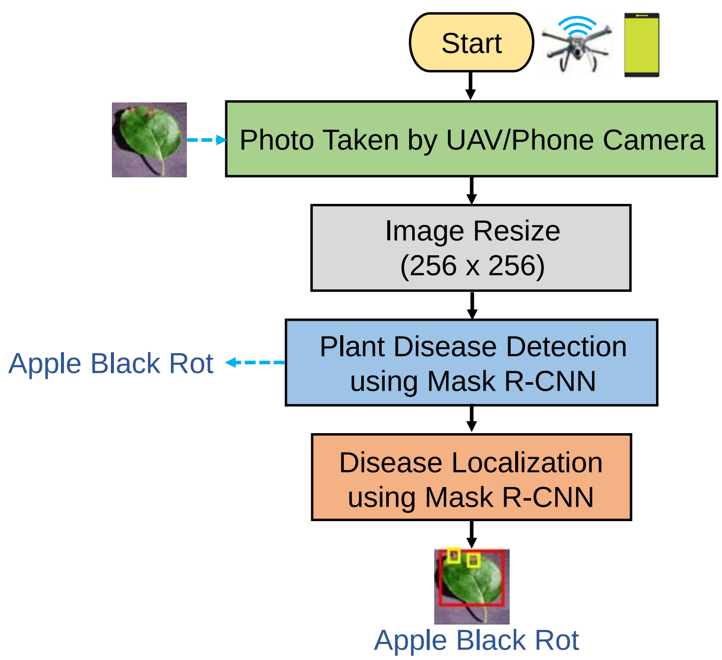


Fig. 3. Process Workflow for Disease Detection and Localization System

Recently, various deep learning networks have achieved state-of-the-art performance for object detection. Region-based Convolutional Neural Networks (R-CNN) showed promising results. However, an R-CNN is computationally expensive. It takes a long time to train it. A fast R-CNN, on the other hand, is much faster than a slow R-CNN and takes much less time to train. Faster R-CNN is faster than its predecessor since it uses a Region Proposal network rather than ROI pooling. The latest in this series is Mask R-CNN, as shown in Fig. 4.

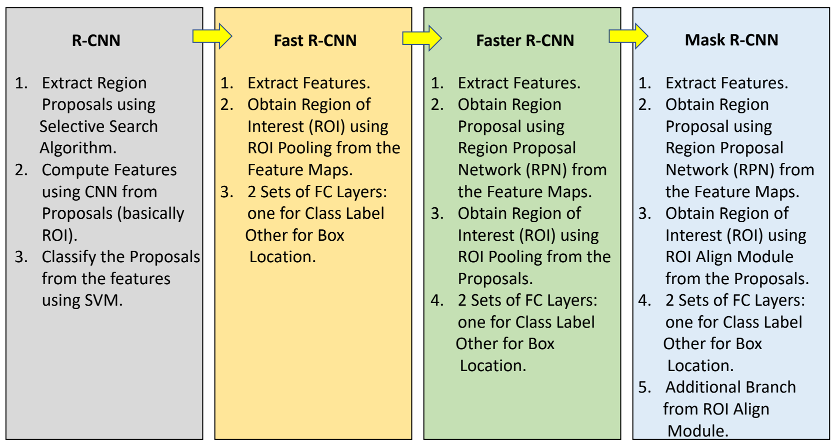


Fig. 4. Evolution of Mask R-CNN [32]

In this work, Matterport's implementation [6] of the Mask R-CNN, based on Keras and TensorFlow, has been followed. We have modified the scripts and hyper parameters as per the need of the application. The developmental process workflow is shown in Fig. 5. First, data has been selected and annotated for object detection purposes. Then, the dataset has been split into two parts: *train* and *validate*. Training of the network has been performed with augmented data. Transfer learning has been used. Finally, it was evaluated with the *test* dataset. The procedure is described in Section 3.3 and 4.

3.3 Network Architecture for Disease Detection and Localization

Mask R-CNN [15] is a general network for object instance segmentation. Image segmentation is a pixel-based division of objects in an image. It gives information about the shapes and sizes of the detected objects. In this work, Mask R-CNN has been used to localize the damage of the leaves caused by apple plant diseases. It is built over Faster R-CNN [31], as shown in Fig. 6. Here, along with the class label and bounding box as in Faster R-CNN, a mask is generated for the detected object to localize the damage area. However, for our work, accurate masks for the damage have not been generated.

– Backbone Network: A backbone network is used for feature extraction of the input image. Initially, two different backbone networks, ResNet101 + Feature Pyramid Network (FPN) and ResNet50 + FPN, have been tested, and finally, ResNet-101 [16] and FPN [23] have been selected as the backbone network. The ResNet network, pre-trained on the ImageNet [12] dataset has been used. Extracted features in this stage are used as the input for the next layer.

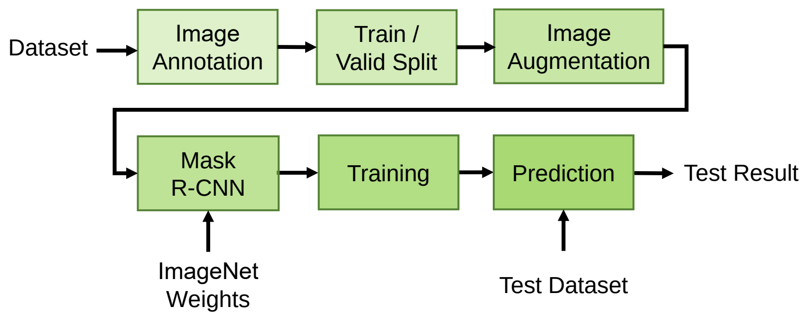


Fig. 5. Workflow for Developing the Disease Detection and Localization System

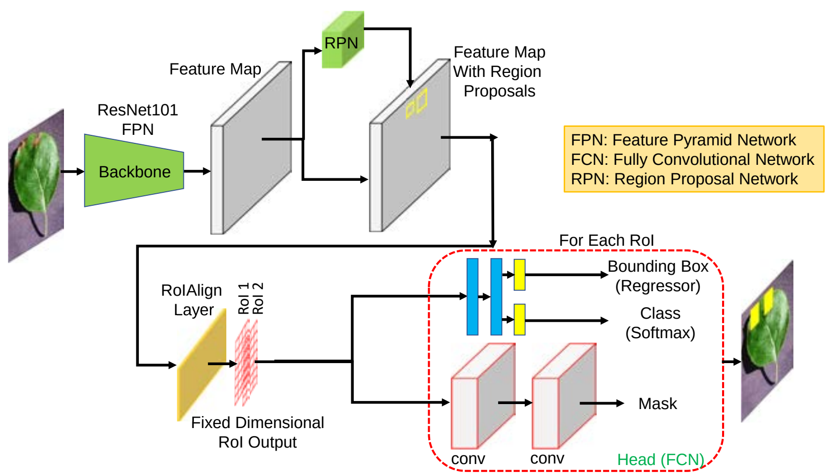


Fig. 6. Mask R-CNN Network Structure

– Region Proposal Network (RPN): Multiple regions of interest are generated using a lightweight binary classifier in a Region Proposal Network(RPN). A small network slides over the feature maps obtained from the preceding stage. It employs anchor boxes to detect numerous, overlapping, and different-sized objects. These predetermined height-and-width bounding boxes capture object classes' scale and aspect ratio. Final object detection removes background anchor boxes and filters the rest by confidence score from multiple predictions. Detection minimum score is set to 0.9 to include all the damages. Non-Max suppression selects the most confident anchor boxes. Hence, RPN simply tells us whether or not there is something in that area. The model predicts which regions or feature maps contain objects. Anchors are centered at the sliding window and have five scales, one for each feature pyramid level.
– Region of Interest (RoI): Each region proposal has different sizes and shapes. The RoIAlign layer changes the shapes and sizes of all proposals to the same shape and size. It aligns the features with the input. The number of RoI is the same as the number of detected objects. RoI is noted when the Intersection over Union (IoU) of ground truth boxes for the predicted regions are greater than or equal to 0.5.

– Head: The *Head* part takes care of the classification and segmentation. Sets of *fully connected* layers are used for bounding box classification. Classification results are predicted by the first branch of *fully connected layers* and *softmax* activation function, and the *regression* output at the second branch of *fully connected layer* is used to determine the location of the proposed regions in terms of the coordinates of the proposals. For each RoI, the parallel branch built with a *fully convolutional network* (FCN) generates binary masks of size 14×14. During prediction, these masks are scaled up to 28×28.

4 Experimental Validation

In this section, we present the experimental validation of the Mask R-CNN based plant leaf disease detection system with a case study on apple leaves.

4.1 Dataset

The publicly available *PlantVillage* dataset [17] has been used for training and evaluating the method. There are a total of 3171 apple leaf images in the dataset. The leaves are either healthy or infected with diseases. There are three types of apple leaf diseases in the dataset-*Black Rot*, *Apple Scab*, and *Cedar Apple Rust*. They show different symptoms on the leaves.

Small, orange-red dots occur on the leaves' fronts in the early stages of *Apple Rust*. These spots grow to become an orange-yellow patch with red edges. A single leaf can have dozens of disease spots if the infection is severe. 12 weeks after the commencement of sickness, the spot's surface is covered with little bright yellow dots [21].

Apple Scab begins with yellow-green radial or circular patches that become brown to black, with clearly defined edges. Smaller and thicker with curled or twisted leaves are signs of more serious illness. Infected spots will blend into one another, causing large patches to appear on leaves, giving them a burnt look [13, 21].

For *Black Rot*, small, purple-black lesions appear on the skin at the beginning of the disease. These develop into spherical spots with a yellow-brown center and brown-purple rims that resemble frog's eyes [13].

1025 infected apple leaf images among 3171 images have been randomly selected to make a balanced dataset of apple leaves. Each type has approximately $300-350$ images. The reason behind not choosing all apple leaf images is to limit the time and effort for annotation.

First, 175 random images are kept aside for evaluating the system. The remaining 850 images are divided into training and validation sets with $70:30$ ratio. Hence, there are 595 images for training and 255 images for validation. Table 1 describes the details of *training* and *validation* datasets.

Table 1. Dataset Details

Types	**Number of Images**		
	Total	**Train**	**Validation**
Apple Black Rot	300	210	90
Cedar Apple Rust	250	175	75
Apple Scab	300	210	90
	850	**595**	**255**

4.2 Image Annotation

Annotation of images with ground truth is a critical step of object detector training. Bounding boxes are drawn across the objects in the training datasets. An open source image annotation tool, MakeSense.AI [1] has been used to annotate the data. *Rect* tool has been used for annotating images. Annotation files are stored in .xml format with the two diagonally placed corners' coordinates of the bounding box. During labeling, different colors are used for different classes. Fig. 7 shows one sample annotated image of an apple leaf infected with *Apple Black Rot* and Fig. 8 shows some annotated leaf samples.

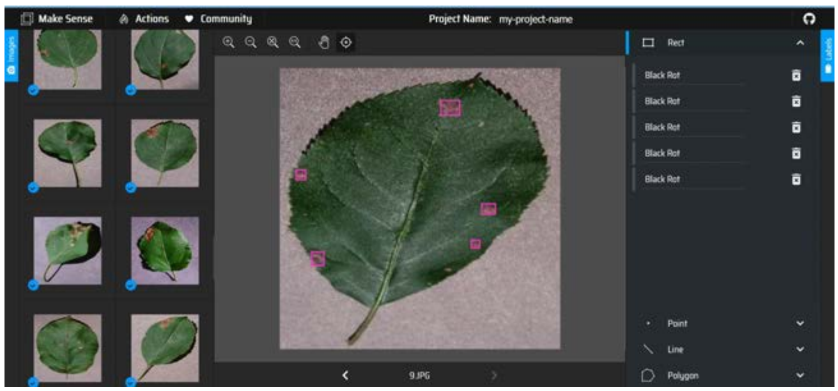


Fig. 7. Image Annotation using Image Annotation Tool MakeSense

4.3 Image Augmentation

For a deep learning model, a large dataset always increases the accuracy of the model. Here, there were only 850 images in our dataset. Data has been augmented on the go to achieve better accuracy. Horizontal flip, vertical flip, affine rotation, affine scaling, and edge detection have been applied to the 850 images. Rotation is set to any random value from $-45°$ to $45°$ and scale value from 0.5 to 1.5.

4.4 Training

The network has been trained on a system with an NVIDIA Tesla P100 GPU and 25 GB of memory. Keras, the deep learning API in Python with TensorFlow at the back-end, scikit-image, pandas, numpy, and imagaug libraries have been used. Transfer learning allowed us to improve the accuracy of the model while simultaneously reducing the amount of time spent on training [25].

While training, the structure of Mask R-CNN is kept unchanged and pre-trained *imagenet* weights of the ResNet networks are loaded. When trained on $20{,}000$ categories of more than 14M labeled images of the *ImageNet* dataset, the network already knows how to identify the most common aspects of images. Basic features like lines and edges are extracted at the lower layers, whilst more sophisticated and abstract elements, such as those that define classification, are extracted at the intermediate and higher layers.

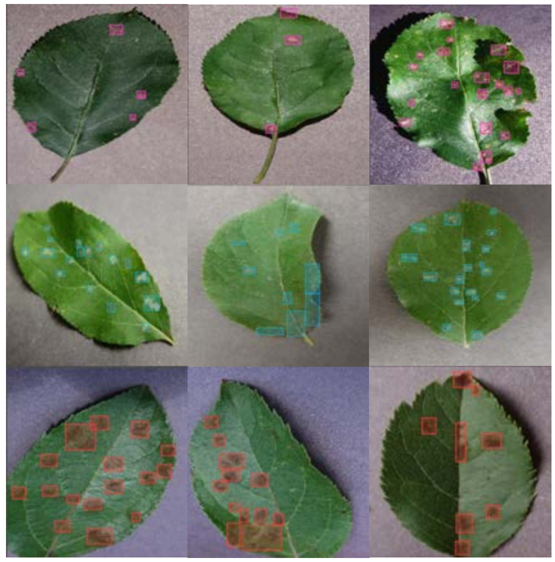


Fig. 8. Annotated Apple Leaf Images from PlantVillage Dataset [17]. Leaves are infected with three different diseases - Black Rot (top row), Cedar Apple Rust (middle row), and Apple Scab (bottom row).

For training RPN, the number of anchors per image has been kept 256 as in [6] and five square anchor boxes of side $[8,16,32,64,128]$ have been selected with a width-to-height ratio $[0.5,1,2]$. The number of steps per epoch is kept equal to the number of training images, and the number of validation steps has been chosen as the number of validation images.

The *images per GPU* is set at 1 to fit the memory. The number of available GPUs in our case is 1 which in turn sets the *batch size* at $\textit{images per GPU} \times \textit{number of GPU} = 1$. Stochastic Gradient Descent (SGD) has been chosen as the optimizer, as in [15]. Initial *learning rate* is set at 0.001. Two different backbone networks: ResNet50 + FPN and ResNet101 + FPN have been compared and finally the latter one has been chosen as the feature extractor. Table 2 shows the learning rate schedule for training the network with ResNet101 + FPN as the backbone. Several hyper parameters have been fine tuned for better detection and are stated in Table 3. The rest of the hyper parameters are set at the default values of [6].

Table 2. Learning Rate Schedule for Training with ResNet101 + FPN Backbone

Learning Rate (LR)	**Trained On**	**Epochs**
0.001	All Layers	1-40
0.0001	All Layers	41-80
0.00003	All Layers	81-120
0.00001	Head	121-160

Table 3. Fine Tuned Hyper Parameters

Hyper Parameters	Values
IMAGES_PER_GPU	1
NUM_CLASSES	1+3
STEPS_PER_EPOCH	595
VALIDATION_STEPS	255
LEARNING_RATE	0.001
TRAIN_ROIS_PER_IMAGE	128
RPN_TRAIN_ANCHORS_PER_IMAGE	64
MAX_GT_INSTANCES	200
DETECTION_MAX_INSTANCES	100
IMAGE_MIN_DIM	256
IMAGE_MAX_DIM	256
RPN_ANCHOR_SCALES	[8, 16, 32, 64, 128]

5 Performance Evaluation

5.1 Performance Metrics

Mean average precision (mAP) has been used to evaluate the performance of the model. *mAP* takes into account both false positives (FP) and false negatives (FN) and considers the trade-off between *precision* and *recall*. Because of this, *mAP* is a good metric for detection tasks. *Precision* and *recall* are defined in Eq. 1 and 2.

$$Precision = \frac{TP}{TP+FP} \tag{1}$$

$$Recall = \frac{TP}{TP+FN} \tag{2}$$

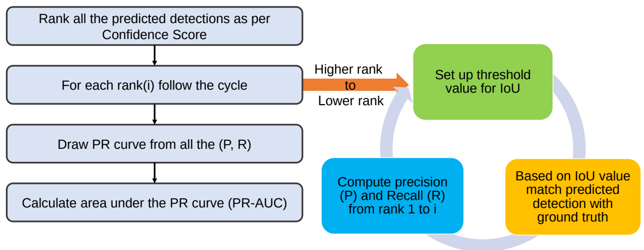
$$IoU = \frac{\textit{Area of Intersection}}{\textit{Area of Union}} = \frac{TP}{TP+FP+FN} \tag{3}$$

where, *TP* is the *true positive*, *FP* is the *false positive*, and *FN* is the *false negative*. Eq. 3 describes how to calculate the *Intersection over Union (IoU)*. It is a measure of how much the predicted boundary overlaps the ground truth boundary. Depending on the *IoU* threshold value, average precision changes. Usually, *IoU* is varied in the range $0.5 \leq IoU \leq 0.95$.

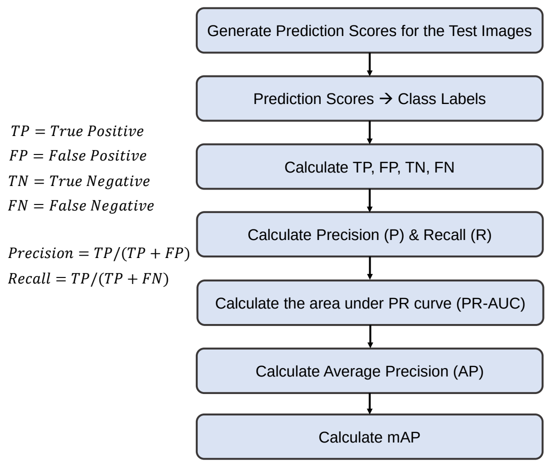
The *average precision* is defined as the area under the precision and recall curve for the object detector. First, *average precision* AP_i is calculated for each class and then using Eq. 4, *mAP* is calculated.

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{4}$$

where, N is the total number of classes. Fig. 9(a) presents the method for plotting the *Precision-Recall* curve [3] and Fig. 9(b) describes the process of calculating *mAP* [2]. The validation dataset has been used to evaluate the model. 175 unseen images of infected apple leaves from the PlantVillage dataset [17] have also been tested.



(a) Average Precision (AP)



(b) Mean Average Precision (mAP)

Fig. 9. Workflows for calculating mAP and AP

5.2 Performance Analysis

Fig. 10 shows sample predicted apple leaves from the test dataset. The first row shows the images from the PlantVillage dataset [17] and the second row shows the predicted results. It is clear from the predicted results that most of the damage parts have been detected. However, when the number of damage areas is higher, there are still room for improvement. Our model missed some of the damage areas. It is mainly due to the small size of training data. To achieve higher precision, more data is needed for training. More hyper parameter tuning in Table 3 in Section 4.4 will contribute to better *mAP*.

Table 4 shows the *mAP* for two different scenarios. When ResNet101 + FPN is used as the backbone, a higher *mAP* of 83.8% has been obtained due to the higher number of layers in ResNet101 than in ResNet50. Initially, we started with an 80 : 20 split for training and validation, but ResNet101 started to overfit. To avoid overfitting, image augmentation along with more validation images (train and validation ratio of 70 : 30) was chosen.

A few of the predictions were not completely correct. Two examples are shown in Fig. 11. Most of the damaged areas in those two leaves have been correctly predicted, but due to the excessive amount of specular reflection and on the leaf shadow, the top left image has one Apple Scab prediction though it is infected with Black Rot disease. In the top right image, which is

Fig. 10. Predicted Results by ResNet101 + FPN on Test Dataset

Table 4. Mean Average Precision (mAP) for Disease Detection Network

Backbone Network	Image Augmentation	Mean Average Precision [mAP(%)] IoU=0.5
ResNet50 + FPN	Yes	81.9
ResNet101 + FPN	Yes	**83.8**

infected with Apple Scab, several damage areas are missed. One damage area has been predicted as Black Rot. This is also due to heavy specular reflection.

Table 5 compares our work with two of the existing works. Mask R-CNN was used in both works. Additional techniques have been used to achieve higher *mAP*. However, we have concentrated on mostly fine tuning of hyper parameters without along with image augmentation. With more investigation into the effects of different hyper parameters along with image enhancement techniques, the achieved *mAP* can be increased.

6 Conclusion and Future Work

Plants, like all other forms of life, are susceptible to contracting many diseases. A plant's ability to develop to its full potential might be hampered by disease [5, 26]. Plant diseases are one of the most significant factors that contribute to crop loss. Plants and trees need to be disease free. Therefore, plant disease diagnosis and damage localization are critical to preventing crop loss. Disease severity is another important factor that needs to be known along with disease detection and damage localization. It decides the amount of pesticide needed. In this paper, we present a Mask R-CNN based method for automatic disease detection and damage localization. These are the prior steps before disease severity estimation. This preliminary research shows promise towards damage severity estimation.

Several improvements can be made in future work. Damage localization can be extended to disease severity estimation. Though apple leaf diseases have only been detected here, the process

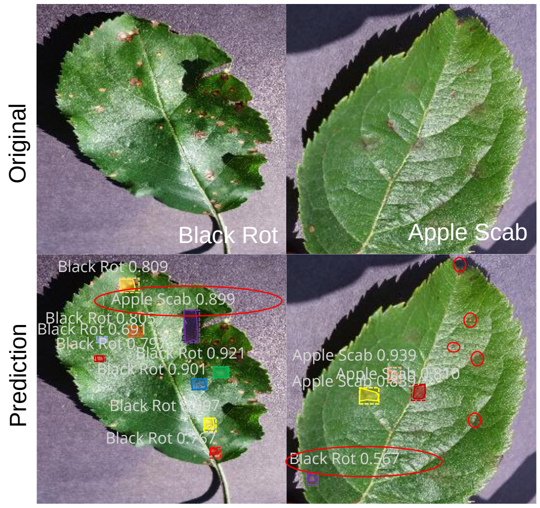


Fig. 11. Falsely Detected Results by the Detection Network. Falsely detected damages are shown with red ovals and missed damage areas are shown with red circles.

Table 5. Comparative Analysis with the Existing Works

Work	Pre-Training Dataset	Method	Metric	Remarks
[7]	MS-COCO	Mask R-CNN + Systematic approach to Image Augmentation	$mAP=$ 82.43%	For Strawberry diseases.
[30]	ImageNet	Image Enhancement + Mask R-CNN + Ensemble Subspace Discriminant	mAP=81.8%, 86.1%	For Apple leaf diseases. More complex method
Current Paper	ImageNet	Mask R-CNN + Image Augmentation	$\mathbf{mAP=83.8\%}$	Automatic tracking of the spread of disease. Case study apple leaf disease.

is valid for any plant/crop whenever the annotated dataset is available. More annotated data will increase the *mAP* of the method. As training data, single leaf images with very distinct backgrounds have been used. But, that is not the case with real-world data where there are different lighting conditions, shadows, specular reflection, and the presence of insects on the leaves. Inclusion of field data into the training images will improve the system. Research on different lighting conditions, shadows, specular reflection, and the presence of insects on the leaves is needed. Some preliminary work on shadow removal has been done in [26]. When taking pictures with UAV or phone cameras, multiple leaves will be present in the frame. Hence, before detecting the damage to a single leaf, each leaf needs to be detected through object detection. Once the damage severity is determined, work might be extended to calculate the optimum pesticide amount.

Acknowledgment

The authors want to thank Madison Huang for helping them label the images of apple leaves from [17].

References

1. Make Sense, https://www.makesense.ai/, Accessd on 13 June 2022
2. Mean Average Precision (mAP) Explained: Everything You Need to Know. https://www.v7labs.com/blog/mean-average-precision, Acessed on 8 July 2022
3. Mean average precision map for object detection. https://leimao.github.io/blog/Object-Detection-Mean-Average-Precision-mAP/, Acessed on 8 July 2022
4. New Study: Agricultural Pesticides Cause Widespread Harm to Soil Health, Threaten Biodiversity, https://biologicaldiversity.org/w/news/press-releases/new-study-agricultural-pesticides-cause-widespread-harm-to-soil-health-threaten-biodiversity-2021-05-04/, accessd on 10 June 2022
5. Plant Disease: Pathogens and Cycles. https://cropwatch.unl.edu/soybean-management/plant-disease, Accessed June 02, 2022
6. Abdulla, W.: Mask r-cnn for object detection and instance segmentation on keras and tensorflow. https://github.com/matterport/Mask_RCNN (2017)
7. Afzaal, U., Bhattarai, B., Pandeya, Y.R., Lee, J.: An Instance Segmentation Model for Strawberry Diseases Based on Mask R-CNN. Sensors **21**(19) (2021). https://doi.org/10.3390/s21196565, https://www.mdpi.com/1424-8220/21/19/6565
8. Agarwal, M., Singh, A., Arjaria, S., Sinha, A., Gupta, S.: Toled: Tomato leaf disease detection using convolution neural network. Procedia Computer Science **167**, 293–301 (2020)
9. Ahmad, I., Hamid, M., Yousaf, S., Shah, S.T., Ahmad, M.O.: Optimizing pretrained convolutional neural networks for tomato leaf disease detection. Complexity **2020** (2020)
10. Bansal, P., Kumar, R., Kumar, S.: Disease Detection in Apple Leaves Using Deep Convolutional Neural Network. Agriculture **11**(7) (2021). https://doi.org/10.3390/agriculture11070617, https://www.mdpi.com/2077-0472/11/7/617
11. Chao, X., Sun, G., Zhao, H., Li, M., He, D.: Identification of Apple Tree Leaf Diseases Based on Deep Learning Models. Symmetry **12**(7) (2020). https://doi.org/10.3390/sym12071065, https://www.mdpi.com/2073-8994/12/7/1065
12. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Proc. of IEEE conference on computer vision and pattern recognition. pp. 248–255. IEEE (2009)
13. Di, J., Li, Q.: A method of detecting apple leaf diseases based on improved convolutional neural network. Plos one **17**(2), e0262629 (2022)
14. Fuentes, A., Yoon, S., Kim, S.C., Park, D.S.: A Robust Deep-Learning-Based Detector for Real-Time Tomato Plant Diseases and Pests Recognition. Sensors (2017). https://doi.org/10.3390/s17092022, https://www.mdpi.com/1424-8220/17/9/2022
15. He, K., Gkioxari, G., Dollár, P., Girshick, R.B.: Mask R-CNN. CoRR **abs/1703.06870** (2017), http://arxiv.org/abs/1703.06870
16. 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. pp. 770–778 (2016)
17. Hughes, D.P., Salathé, M.: An open access repository of images on plant health to enable the development of mobile disease diagnostics through machine learning and crowdsourcing. CoRR **abs/1511.08060** (2015), http://arxiv.org/abs/1511.08060
18. Jiang, F., Lu, Y., Chen, Y., Cai, D., Li, G.: Image recognition of four rice leaf diseases based on deep learning and support vector machine. Computers and Electronics in Agriculture **179**, 105824 (2020)

19. 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 **7**, 59069–59080 (2019). https://doi.org/10.1109/ACCESS.2019.2914929
20. Li, D., Wang, R., Xie, C., Liu, L., Zhang, J., Li, R., Wang, F., Zhou, M., Liu, W.: A Recognition Method for Rice Plant Diseases and Pests Video Detection Based on Deep Convolutional Neural Network. Sensors **20**(3) (2020). https://doi.org/10.3390/s20030578, https://www.mdpi.com/1424-8220/20/3/578
21. Li, L., Zhang, S., Wang, B.: Apple leaf disease identification with a small and imbalanced dataset based on lightweight convolutional networks. Sensors **22**(1), 173 (2021)
22. Li, L., Zhang, S., Wang, B.: Plant disease detection and classification by deep learning—a review. IEEE Access **9**, 56683–56698 (2021)
23. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
24. Liu, B., Ding, Z., Tian, L., He, D., Li, S., Wang, H.: Grape Leaf Disease Identification Using Improved Deep Convolutional Neural Networks. Frontiers in Plant Science **11** (2020). https://doi.org/10.3389/fpls.2020.01082, https://www.frontiersin.org/article/10.3389/fpls.2020.01082
25. Mitra, A., Mohanty, S.P., Corcoran, P., Kougianos, E.: A Machine Learning Based Approach for Deepfake Detection in Social Media Through Key Video Frame Extraction. SN Comput. Sci. **2**(2), 98 (2021), doi: 10.1007/s42979-021-00495-x.
26. Mitra, A., Mohanty, S.P., Kougianos, E.: aGROdet: A Novel Framework for Plant Disease Detection and Leaf Damage Estimation. In: Proc. of 5th FIP International Internet of Things (IoT) Conference (IFIP-IoT) (2022, (Accepted))
27. Mitra, A., Singhal, A., Mohanty, S.P., Kougianos, E., Ray, C.: eCrop: A Novel Framework for Automatic Crop Damage Estimation in Smart Agriculture. SN Comput. Sci. **3**(4), 16–pages (2022), doi: 10.1007/s42979-022-01216-8.
28. Mitra, A., Vangipuram, S.L.T., Bapatla, A.K., Bathalapalli, V.K.V.V., Mohanty, S.P., Kougianos, E., Ray, C.: Everything You wanted to Know about Smart Agriculture. CoRR **abs/2201.04754** (2022), https://arxiv.org/abs/2201.04754
29. National Academies of Sciences, Engineering and Medicine and others: Science breakthroughs to advance food and agricultural research by 2030. National Academies Press (2019)
30. Rehman, Z.u., Khan, M.A., Ahmed, F., Damaševičius, R., Naqvi, S.R., Nisar, W., Javed, K.: Recognizing apple leaf diseases using a novel parallel real-time processing framework based on MASK RCNN and transfer learning: An application for smart agriculture. IET Image Processing **15**(10), 2157–2168 (2021)
31. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28** (2015)
32. Rosebrock, A.: Mask R-CNN with OpenCV, https://pyimagesearch.com/2018/11/19/mask-r-cnn-with-opencv/, Accessd on 10 June 2022
33. Singh, V., Misra, A.: Detection of plant leaf diseases using image segmentation and soft computing techniques. Information Processing in Agriculture **4**(1), 41–49 (2017). https://doi.org/https://doi.org/10.1016/j.inpa.2016.10.005, https://www.sciencedirect.com/science/article/pii/S2214317316300154
34. Wang, G., Sun, Y., Wang, J.: Automatic image-based plant disease severity estimation using deep learning. Computational intelligence and neuroscience **2017** (2017)
35. Wang, P., Niu, T., Mao, Y., Zhang, Z., Liu, B., He, D.: Identification of Apple Leaf Diseases by Improved Deep Convolutional Neural Networks With an Attention Mechanism. Frontiers in Plant Science **12** (2021). https://doi.org/10.3389/fpls.2021.723294, https://www.frontiersin.org/article/10.3389/fpls.2021.723294
36. Wang, Q., Qi, F., Sun, M., Qu, J., Xue, J.: Identification of tomato disease types and detection of infected areas based on deep convolutional neural networks and object detection techniques. Computational intelligence and neuroscience **2019** (2019)