

# Crop Disease Auto-Localization and Classification

G Deepank<sup>1</sup>, Shivani Bisht<sup>2</sup>, Aditya Verma<sup>3</sup>, R Tharun Raj<sup>4</sup> and Jyothi R<sup>5</sup>

<sup>1-5</sup> PES University, Bangalore 560085, Karnataka, India

deepankg@pesu.pes.edu<sup>1</sup> shivani11bisht@gmail.com<sup>2</sup>  
aavermadps@gmail.com<sup>3</sup> tharunr123@gmail.com<sup>4</sup> jyothir@pes.edu<sup>5</sup>

**Abstract.** Plant disease detection plays an important role in the agriculture field, as different diseases affecting the growth of plants is inevitable. If suitable measures are not taken in time with respect to this aspect, then it harms the crop, resulting in substantial decrease in the quality of yield produced. This is where AI can be quite useful. It reduces human interference in monitoring big farms and aids in the detection of symptoms of diseases in time to find a solution. Computer vision, machine learning and deep learning algorithms are being used to process this data. In this paper, digital image processing and deep learning models (different architectures and algorithms) were used for classification (type of microorganism and disease) with localization and detection of the diseases (pre-conditions/symptoms) present in the plant crops. An attempt was made to carry out image segmentation in a novel way, using the DICOM format. The GradCAM algorithm was used to perform a detailed analysis of various deep learning algorithms to validate the accuracy with which each algorithm classifies the diseases. Furthermore, bounding box regression was performed to locate the disease symptoms on each image. Lastly, the whole process was automated by hosting it on a web application for an easier user experience.

**Keywords:** Deep Learning, Classification, Localization, Crops, Image Processing.

## 1 Introduction

The agricultural sector is more than just a food source in these times. The world and the Indian economy are highly dependent on agricultural yields. Thus, in agriculture, detection, and diagnosis of various diseases in crops play a prominent role which can end up benefiting farmers and consumers alike. To detect any plant disease at an early stage and have an early diagnosis, automated disease detection techniques can be useful. For example, brown spot disease is a harmful disease found in rice crops. This disease can spread throughout the entire leaf and it can kill the whole crop in a short amount of time. The current method for plant disease detection and diagnosis is an observation made by experts through which detection and identification of plant diseases is done. For accomplishing this task, large teams of experts, with some amount of monitoring of the plants is required, which is expensive in an economical point of view. In many countries, some farmers, gardeners, and researchers do not have the ideas or the proper facilities and resources that they can consult the experts,

due to which consulting experts costs a lot and is also very time consuming. In such circumstances, the proposed technique proves to be advantageous in monitoring a number of crops at once. The automated detection of diseases by just feeding the symptoms or just uploading a picture of the plant leaves makes it easier and cheaper. The method also supports machine vision ideologies to provide image based automatic process control, and inspection.

Plant disease detection and diagnosis done by the on-site way is a more effortful and time consuming task and it is less accurate and can be done only in limited locations and areas as resources are not spread out. If automated detection techniques are made use of, it will take less time, less effort, and return much more accurate results, which makes it more accessible to people with little or no access to such resources and expert advice. Processing images of disease affected crops can help in accurately identifying diseases, and also help to come up with possible solutions to the disease in the crop. Image processing can also be used for detecting the disease affected area.

## **2 Problem Statement**

This project will facilitate the classification and identification of different crop diseases using various deep learning models (algorithms) and techniques. The evaluation, visualization, and analysis of the different implemented models are also carried out. Finally, the best model will be used for deploying into a web application. The dataset for the project will be created alongside, which will include labeling, classification, bounding box coordinates of areas of interest and augmenting of a corpus of image data. Localization allows us to find the area of interest i.e., the affected areas (pre-conditions) of the plant crop such as discoloration, spots, holes, etc. The proposed methodology also involves subjecting images to 'medical imaging' using the DICOM format. This technique allows us to capture issues that are not fully visible to agronomists by naked-eye observation. The obtained high spectral resolution gives us much more useful content to describe the analyzed area of concern of the affected crop.

## **3 Literature Review**

Extensive research has been done in the area of detection and classification of rice crop diseases using machine learning. Some of the related works on this subject are presented below.

Singh and Misra [4] presented an algorithm for image segmentation technique which was to be used for automatic detection and classification of plant leaf diseases. In which an important aspect for disease detection in plant leaves was done by using a genetic algorithm, which gives a number of optimum solutions, not a single solution. So different image segmentation results could be obtained at the same time. In the image recognition and segmentation processes, mostly green colored pixels were masked. In this, a threshold value was computed to be used for these pixels. Then in the following way mostly green pixels were masked: if pixel intensity of the green

component was less than the pre-computed threshold value, then zero value was assigned to the red, green and blue components of the pixel. In the infected clusters the masked cells were removed to obtain the useful segments.

Nagasubramanian et al [18] deployed a modernistic three-dimensional deep convolutional neural network, which precisely assimilated hyperspectral information. Hyperspectral imaging was investigated with respect to disease detection and classification as discussed in [12]. They decided to work on charcoal rot as the disease, because apart from soybean, it affects important crops like corn, cotton etc. Moreover, there are limited chemical measures available, so resistance breeding was an important countermeasure as mentioned in [16]. The CNN model was developed such that it comprised 2 convolutional layers, associated with which there were 2 max-pooling layers and finally, 2 completely connected layers. Kernels of different sizes were considered to convolve the input, particularly 2 kernels of size  $3 \times 3 \times 16$  were employed in the first layer and 4 kernels of the same size in the second layer. The activation function chosen for the output was Rectified Linear Unit(ReLU). Furthermore, a saliency map was plotted to highlight areas of the plant image that were most indicative of the presence of the disease. Thus, they concluded hyperspectral imaging techniques in combination with proper deep learning models can be used to identify specific features that distinguishes a healthy plant from an infected one, with sufficiently high accuracy.

Yan Guo et al [20] proposed the use of a region proposal network (RPN) in order to recognize and localize leaves in complex surroundings. The RPN is trained on a dataset consisting of leaf images in different environments and the images of the symptomatic leaves are returned. Different techniques were studied for feature extraction, such as using color and texture features to extract disease spots which was mentioned in a paper by Raza and colleagues [17]. The use of transfer learning for feature extraction and building a disease identification system was also studied as proposed by Couilbaly et al. [19]. These images are then segmented by making use of the Chan-Vese algorithm that performs iterative calculations to obtain specific leaf contours based on the minimum energy function and set zero level. These segmented images are then fed into a pre-trained transfer learning model to identify diseases in a simple white background. The correct rate percentage average of the proposed model is much better than traditional deep learning models like ResNet101. Although the model works with a reasonably good accuracy of 83.57%, since the Chan-Vese algorithm makes use of repetitive iterative calculations, the model works really slow when dealing with datasets of large sizes.

Konstantinos P. [3] developed CNN models for plant disease detection using images of healthy and diseased plants and their leaves. The models were trained from a database of 87,848 images that contained 25 different plants that made up a set of 58 distinct classes of like [plant, disease] combinations, which also contained healthy plants. From this experiment, the best performance was found to have a success rate of 99.53% to identify [plant, disease] combinations from all the architectures that were trained. Based on these results, only some of the CNN architectures were assessed and formed a plant disease diagnosis system. Based on images of both diseased and healthy plants and their leaves, 5 Convolutional Neural Network

architectures were picked for this particular problem. The LuaJIT with Torch7 ML computational framework was used for the implementation of these models and their training and testing. In the database, over one third (37.3%) of the images were captured in real cultivation conditions out in the field. Random images were selected using python scripts. Cropping images to 256x256 pixels and keeping training/testing ratio as 80/20 was what made up the pre-processing. Both laboratory conditions as well as real world images were used. Results showed that all models have a better performance when they use the original leaf images. As a consequence, more training time was needed. The VGG model ended up achieving the highest percentage of successful classification (99.53%) with a top – 1 error = 0.47%.

Madhu et al. [15] had the objective to collect, preprocess and classify diseases. The data collection was done physically, using high resolution cameras. While preprocessing, they used algorithms such as noise removal, color space conversion, contrast improvement, and brightness correction. Pictorial attributes were transformed into quantifiable attributes using feature extraction. For classification, fuzzy networks, SVMs etc. were used.

Kulkarni and Patil [5] aimed to discover a methodology for early and accurate plant diseases detection, by making use of many image processing techniques and neural networks. Their method offered results with a recognition rate of up to 91%. In the paper written by Bashir and Sharma [7], they mention performing disease detection in *Malus domestica* through algorithms like K-mean clustering, texture and color analysis.

Akhila and Deepan [6] proposed a deep-learning-based approach to detect plant leaf diseases in different plants using images of its leaves. They made use of three main detector models: Faster Region-based Convolutional Neural Network (Faster R-CNN), Region-based Fully Convolutional Network (R-FCN) and Single Shot Multibox Detector (SSD). This choice of models was inspired from the work of Ren S et al [11]. In this paper, they attempted real time object detection using region proposal networks. The detector implemented by them captured images with the help of many camera devices and they were also collected from various online resources. In conclusion, their models were successful in discerning and recognizing different kinds of diseases in various plants. They also attempted to provide solutions relevant to the plant disease identified.

Majumdar et. al. [10] proposed a novel method based on integrated analysis of digital images for detection of rust disease on wheat leaves. They first transformed the image to grayscale which is dependent on the HSI color space. The feature extraction part of this method was done using Fuzzy C means clustering. Finally, a multi-layered perceptron has been used for classification and detection. The model provided 85% accuracy for identification of rust, and 97% accuracy for classification was achieved.

Weizheng et. al. [8] proposed a new method of plant disease classification which was tested out on soybean plants with gray spot disease. The infected leaves were placed on a white background, and the pictures were then segmented using the Otsu method in the gray channel to minimize distortions. The areas of interest on the leaves, where the disease symptoms are present, were segmented using the Sobel operator. Then operations of augmentation such as region filling etc. were applied on

the H component in the HSI scale. Then, the diseases' degree is quantified by calculating the quotient of areas of interest spots and healthy leaf areas.

Bauer et. al. [9] developed systems for classification of leaf diseases based on multispectral images of sugar beet leaves. The images of diseased leaves were taken in a laboratory environment, using multispectral and RGB style cameras. The leaves were either healthy or infected by some diseases which are caught by sugar beet leaves. Using these images, 3D models were made for each of the leaves, and the detection and classification was done using K-means clustering and adaptive Bayes methods. They received 86% classification accuracy for rust disease, as well as 91% accuracy for cercospora leaf spots.

Ampatzidis et. al. [13] presented and reviewed robot based solutions and applications on plant pathology and management, and emerging agricultural technologies and development for the novel types of intra-urban agriculture. With respect to agricultural productivity, Oerke [1] gave an overview on different types of crop losses as well as various methods of pest control developed over the course of the last hundred years. Barbedo [2] dove deep into the main factors that affect the effectiveness and design of deep neural networks applied to the field of plant pathology. Here he provided an in-depth analysis of the subject, in which advantages and shortcomings were highlighted.

## **4 Proposed Methodology**

### **4.1 Dataset Preparation**

A labeled dataset of rice crop diseases with bounding box coordinates was prepared initially. Various images of diseased rice crops were found and brought to one place. The images were then moved into folders with their respective disease name using python scripts. The images were then subjected to augmentation to expand the dataset size for training purposes. The images were subjected to different kinds of rotation, cropping and noise addition. For disease detection and localization of diseases, the dataset was annotated, along with the bounding box coordinates. Localized bounding boxes were manually created for all of the original as well as augmented images, using LabelImg. These coordinates were stored in XML files. Localization helps in finding areas of interest i.e., affected areas (pre-conditions) of plant crops like discoloration, spots, holes, etc. A CSV file containing the file paths of all the images, their respective disease labels and the bounding box coordinates of their areas of interest was created to be parsed and preprocessed later while creating the model.

### **4.2 Image Segmentation using DICOM format**

This technique helps in capturing issues that are not visible to agronomists. High spectral resolution provides more informational content to describe the analyzed area of the object. Using the `img2dcm` command on Linux, all the images were subject to

the DICOM segmentation and stored. The file paths of the newly created DICOM images were added alongside the file paths of the original images to the CSV file.

#### **4.3 Feature Extraction**

Feature extraction techniques are used to obtain features or characteristics that will help us in object detection and classification. OpenCV and NumPy libraries were used to perform feature extraction by resizing the dimensions of the images after having subjected them to normalization.

#### **4.4 Transfer Learning**

There are some pre-existing generally well performing convolutional models such as VGG16, Xception, Resnet50 etc., that can be leveraged for classification and identification. The models are trained on the ImageNet database which consists of images of regular objects. The weights that these models were trained on were used to perform classification for our dataset consisting of rice crop images by changing the dimensions of the final few layers.

#### **4.5 Model Analysis**

After the training, the dataset is split into train and test datasets which are used for validation. Next, the accuracy of the predicted images was validated and the results of the different models were presented. Using these results, the performance of different models were analyzed.

#### **4.6 Grad-CAM: for model performance analysis**

Grad-CAM [5] is generally used to emphasize the most important areas from the point of view of the CNN model used for the classification. This makes the model more transparent. It uses the gradient information in the final layer to get an idea of the importance of the individual neurons for decision of interest. Grad-CAM was used to verify what the models learnt by highlighting the areas the model used to classify the image. This method is used to verify whether the classification of models is happening based on the features decided initially or some other co-incidental feature in the images.

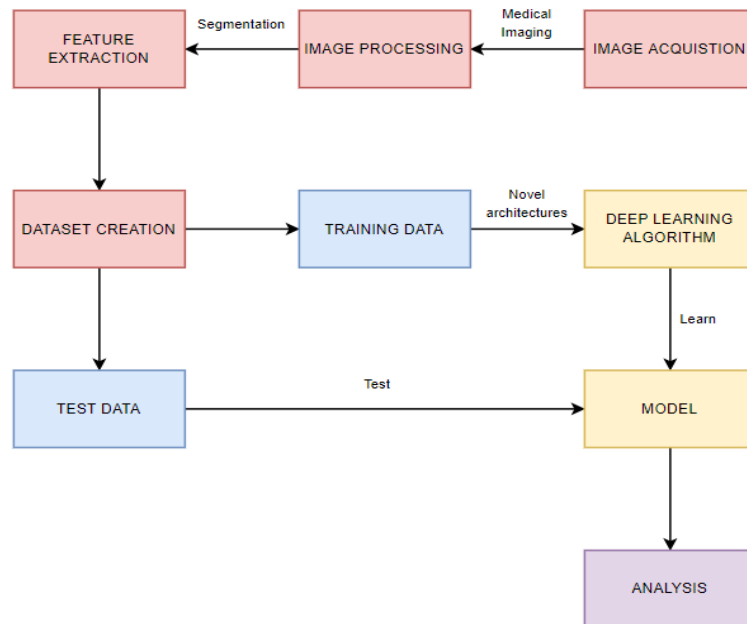
#### **4.7 Bounding Box Regression**

The bounding box coordinates are in PascalVOC format. These coordinates are normalized based on image dimensions. This method reused the classification model used earlier as it was already pre-trained to do feature extraction. As a result, the

model is aware of regions of interest. This speeds up our regression problem. The model loss and accuracy using IoU(intersection over union) metric was calculated. The model is trained for 400 epochs. And IoU of about 0.5 is obtained. To understand model performance a sample of high and low IoU images are visualized.

#### 4.8 Model Deployment

After the different models are evaluated and compared, the model with the best measure of the defined metrics will be deployed as a small web application. While saving the model, its trained weights will be saved along with its layers. In the web app, a user can give their own image of a rice crop leaf as the input for classification. The features will be extracted from this image and the saved model will give the relevant output, which is the bounding box coordinates locating the disease symptoms on the image, as well as the identified disease of the plant. The possible solutions and preventive measures for the detected disease are also displayed for the users' convenience.



**Fig. 1.** Illustration of the proposed methodology

## 5 Implementation

### 5.1 Augmentation and Pre-processing

This was done using TensorFlow and Keras modules. To perform augmentation, the 'skimage' library in Python was used. Utilities like 'sk.transform.rotate' and 'sk.util.random\_noise' were used to rotate the images by a random angle and to introduce some noise into the images, respectively. Along with the techniques mentioned, the images were also flipped, both horizontally and vertically to obtain a working dataset of 240 images. The pre-processing stage involved annotating the images and manually drawing localized bounding boxes around the infected areas using 'LabelImg'.

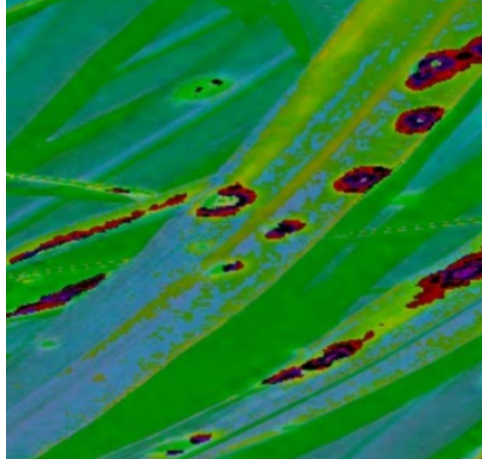
### 5.2 DICOM Segmentation

To recreate medical image segmentation the closest to a real-world equipment, the img2dcm unix command was used, which is based on the DICOM format. The img2dcm command was used to convert all the images to DICOM format. In this format, the symptoms were projected more prominently and this allowed us to test if the method helped increase the prediction and detection accuracy of the model and if it reduced the runtime and model training. The reason why the symptoms got projected clearly and saliently is still quite unknown. But the likely reason is mostly due to the visualization behaving like a lossless compression and this is most attributed to an inconsistency in between the color space indicated by the DICOM Photometric Interpretation and the actual color space indicated by the encapsulated Pixel Data JPEG stream.



**Fig. 2.** Sample rice plant disease image illustration





**Fig. 3.** Plant disease image after applying medical imaging

### 5.3 VGG16 Model

Used pre-existing models that were trained on the ImageNet database that consists of regular objects. Made use of the weights that the model was trained on, to perform classification for the dataset consisting of rice crop images by changing the dimensions of the final few layers. The layers from the input layer to the last max pooling layer (dimensions given by  $7 \times 7 \times 512$ ) is called the feature extraction part of the model, while the rest of the network is called the classification part of the model. After this, the model is loaded with the expected input image size of  $224 \times 224$ . The image object is converted to a NumPy array and the dimensions are increased and adjusted appropriately. The features can now be extracted with the predict method.

### 5.4 Resnet50 Model

A pretrained model trained on the ImageNet database and weights was loaded and used here. The last few layers of the model were modified to fit the logic of our problem statement. The layers from the input layer to the last max pooling layer (dimensions given by  $7 \times 7 \times 2048$ ) is called the feature extraction part of the model, while the rest of the network is called the classification part of the model. After this, the model is loaded with the expected input image size of  $224 \times 224$ . The image object is converted to a NumPy array and the dimensions are increased and adjusted appropriately. The features can now be extracted with the predict method. The output feature vector was then flattened into a list of  $7 \times 7 \times 2048$  which equals a total of 1,00,352 dimensions.

### 5.5 MobileNet V2 Model

The model is trained on the extensive ImageNet database and the corresponding weights. The dimensions of the last few layers which perform the classification are adjusted to enable the model to classify rice crop images instead of regular objects. The initial part of the network is left unchanged to make use of the pre-trained weights and perform feature extraction. Specifically, a sequential layer was added along with two dense layers and a dropout layer. The model was compiled using the RMSProp optimizer because it converges at a relatively faster rate. Model was tested on both the original images and the images subjected to DICOM segmentation and the results were compared.

### 5.6 Xception Model

The model was trained on the ImageNet database and weights. Similar to the other models, the initial layers aren't modified to allow the model to perform feature extraction. The last output layer which has dimensions of  $2048 \times 3 \times 3$  is flattened to get 18432. A GlobalAveragePooling2D layer was added first and then a fully-connected layer was added to allow the model to perform classification. Model was tested on both the original images and the images subjected to DICOM segmentation and the results were compared.

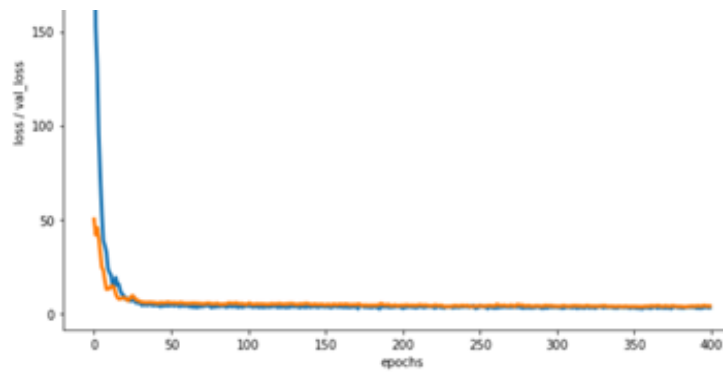
### 5.7 Grad-CAM Analysis

This helps us in validating the performance of the CNN model. It's used to check what are the areas of interest that the model is looking into while performing the classification. Using the gradient information in the final layer, it obtains an understanding of the importance or relevance of individual neurons for decision of interest. Grad-CAM allows us to identify the important neurons in the network and a method to combine these neurons with their appropriate names to help provide graphical/textual information regarding the model decisions. If the model is looking at the wrong areas while performing classification, certain parameters can be adjusted to improve the accuracy of the model.

### 5.8 Bounding Box Regression

This was done using TensorFlow and Keras modules. All rows with "healthy leaf" category are dropped. This is because there are no coordinates annotated for this category. Dataset is split into train and test in the ratio of 80:20. The bounding box coordinates are normalized with respect to image dimensions. For calculating the model loss and accuracy for optimization, an IoU metric function was defined. It is used as the custom loss function while training. The saved neural network model (used earlier for classification) is modified and reused. The last layer was replaced

with a linear activation layer. The model is compiled and trained using the Adam optimizer function for about 400 epochs. The model performance is then visualized by plotting a graph of epochs vs loss and epochs vs accuracy for both train and validation sets. To understand model performance, denormalization is done first and then both the misdetections and relatively better detections are visualized. The new regression model weights and architecture is then saved in a H5 file for later use during the deployment stage.



**Fig. 4.** Plot of loss/validation loss for localization

## 5.9 Deployment

From the analysis of different models done, the overall best performing model is chosen for model deployment. The VGG model, which gave the best performance overall, is saved along with its weights and architecture to an H5 file. Using the Flask microframework, this model is imported for use. Using the GUI, the user can then upload their own image file. Feature extraction is done on this image, and is fed to the imported VGG model. A predefined list with each index corresponding to a different disease in order of prediction is maintained. The VGG model then predicts the disease label in the form of a list of probable values, each value corresponding to the probability of a disease. The highest value from this list is chosen, and the corresponding disease of that value index is shown as output on the GUI.

## 6 Results and Discussion

### 6.1 Classification: models analysis and evaluation

EDA of the whole dataset was done to figure out the intricate aspects of the final dataset. The dataset was then pre-processed with feature engineering to add some new columns and simplify the data, and prepped for model training. After feature

engineering and extraction, the data was then fed into different neural networks and split into train and test batches of appropriate size.

The models used for this purpose were: VGG16, Resnet, Xception, and MobileNet. Table 1 below shows all the models performance in doing classification.

**Table 1.** Model accuracies of different models

Model	Train Accuracy	Test Accuracy
VGG16	98.96%	93.75%
Xception	94.79%	93.75%
MobileNet	98.44%	91.67%
ResNet50	97.92%	91.67%

From the above table it can be observed that all the selected models perform a remarkable job in classifying rice crop diseases. However, VGG16 and Xception have the best test accuracies, with VGG16 taking the edge in train accuracy. From this, the conclusion can be drawn that the VGG16 model seems to be the best performing model for the current working dataset.

To get a better overall understanding of the VGG16 model, the full confusion matrix was plotted, as shown in Table 2:

**Table 2.** Classification report of the VGG16 model

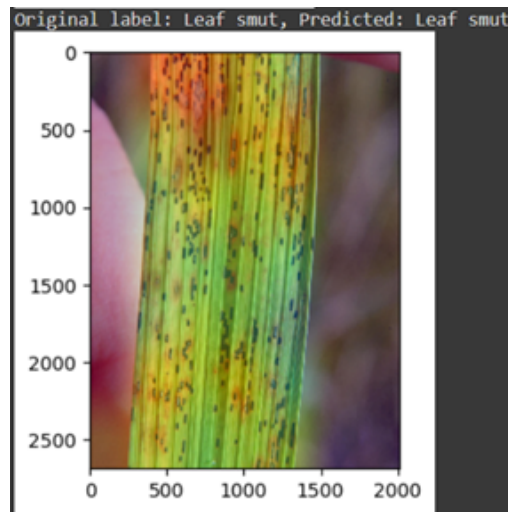
	precision	recall	f1-score	support
Bacterial leaf blight	0.94	0.94	0.94	16
Brown spot	0.82	0.90	0.86	10
Healthy leaf	1.00	1.00	1.00	11
Leaf smut	1.00	0.91	0.95	11
accuracy			0.94	48
macro avg	0.94	0.94	0.94	48
weighted avg	0.94	0.94	0.94	48

As seen in the confusion matrix in Table 2, the precision, recall and F1 scores of the whole model used, as well as these metrics for individual sub-datasets are shown. The column named “support” signifies the number of images of the particular disease label chosen for testing. Due to the predetermined 80-20 split on 240 images, it can be observed that a total of 48 images were used for testing the model.

## 6.2 Grad-CAM Visualization

After the model training, Grad-CAM visualization was done. This model was made to verify that the image classification models are looking at the right areas and symptoms while classifying the image. This was an important part of the analysis as it could be verified whether the models were using the right features and sections of images to classify them.

A heatmap of the classified image was made and superimposed on the image:

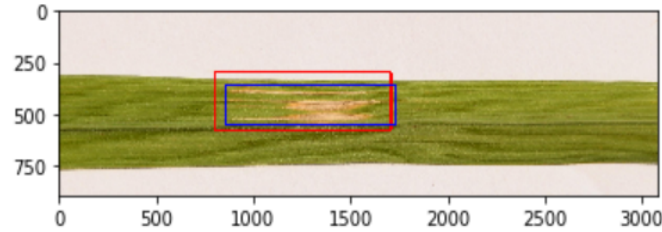


**Fig. 5.** Superimposition of model result and image

The yellow areas show where the classification model was “looking” when it made the classification. From Fig. 5, it can be seen that the yellow areas of the heatmap are over the smut-affected part of the leaf. This confirms that the model analyzed here (VGG16) is looking at the right areas of the different images for classification.

## 6.3 Bounding Box Regression

The bounding box coordinates used for localization of disease symptoms were fed into the best performing classification model (VGG16) and split into train and test batches to test out if the model can localize symptoms of the diseases in an unknown leaf.



**Fig. 6.** Bounding box visualization in a sample image

It is evident from the above image that our model is able to detect symptoms in plant crops satisfactorily. Here in the sampled bacterial leaf blight image, the ‘blue box’ represents groundtruth coordinates whereas the ‘red box’ depicts predicted coordinates. The degree of overlapping of the red and blue boxes signifies the closeness of the actual and predicted localized area of the disease. This is called IoU (Intersection over Union). After running our model for 400 epochs, an IoU score of 0.5 was achieved. While this is not entirely perfect, it appears sufficient for our application.

#### 6.4 Medical imaging as a segmentation technique

The use of DICOM for medical imaging as a segmentation process provided promising results. It achieved the same, if not, better accuracy in doing classification with localization. Using segmented images also gave a higher starting point in terms of model training. This was helpful as model training was relatively faster. Achieving higher accuracy in detecting plant diseases at smaller training times can be very effective in applications like ours where time is of utmost importance.

**Table 3.** Comparison of time taken for DICOM segmented and unsegmented rice crop images

Models used	Training time for unsegmented image (in seconds)	Training time for segmented image (in seconds)
Xception	71	44
MobileNet	20	13

## 7 Conclusion and Future Work

The following section summarizes the various steps taken in implementing the project.

A complete working dataset was developed to perform model training. Different neural network architectures were used for identification, classification with localization like Resnet50, MobileNet, VGG16, and Xception networks. These

different neural network models were then evaluated and analyzed. For novelty, medical imaging was used in the segmentation phase for crop disease detection, following which the accuracy and speed of classification was compared for unsegmented and segmented images. Then, the Grad-CAM was used to validate the performance of the best performing neural network model. Using this, it was confirmed that the classification was happening due to the relevant features in images, and not some other common external feature of images. Further, bounding Box Regression was used to find the localized area of interest of the disease in each crop image using Intersection over Union loss metric. The corresponding model was evaluated and analyzed.

Further work can involve including better coordinate features while labeling the localized areas of interest of a diseased leaf. This will help in optimization of the model. Some leaves have more than one area of interest, and instead of taking all diseased areas as one connected area, each area can be marked separately, and then use multiple bounding box regression for better accuracy.

Another future experimentation could involve adopting 'region proposals' strategy for finding areas of interest in crop disease detection like R-CNN, Fast-RCNN, Faster-RCNN etc. Also, mask R-CNN can be used which tries to detect and segment the locations and shapes of the infected areas. The improvement made by the Mask R-CNN is that the ROI Pooling layer is optimized into the ROI Align layer and a Mask branch is added while the final classification and positioning are carried out so that shapes of lesion spots can be accurately segmented on the leaf. The Mask R-CNN adds a third output through the Mask branch, i.e The Mask R-CNN produces an output for each ROI.

## References

1. E.-C. Oerke: Crop losses to pests. In: The Journal of Agricultural Science, vol. 144, no. 1, pp. 31–43 (2006).
2. J. G. A. Barbedo: Factors influencing the use of deep learning for plant disease recognition. In: Biosystems Engineering, vol. 172, pp. 84–91 (2018).
3. Konstantinos P. Ferentinos: Deep learning models for plant disease detection and diagnosis. In: Computers and Electronics in Agriculture, vol 145, pp. 311-318, Elsevier, ISSN 0168-1699 (2018).
4. Singh, Vijai., Misra, A.K.: Detection of plant leaf diseases using image segmentation and soft computing techniques. In: Information Processing in Agriculture, vol. 4, issue 1, pp. 41-49, Elsevier, ISSN 2214-3173 (2017).
5. Kulkarni, Anand H., Patil, Ashwin, RK: Applying image processing techniques to detect plant diseases. Int J Mod Eng Res; 2(5):3661–4 (2012).
6. M. Akila., P. Deepan: Detection and Classification of Plant Leaf Diseases by using Deep Learning Algorithm: International Journal OF Engineering Research & Technology (IJERT). In: ICONNECT, vol. 6, issue 07 (2018).
7. Bashir, Sabah., Sharma, Navdeep.: Remote area plant disease detection using image processing IOSR J Electron Commun Eng, 2 (6), pp. 31-34 ISSN: 2278-2834 (2012)

8. Weizheng, S., Yachun, W., Zhanliang, C., and Hongda, W.: Grading Method of Leaf Spot Disease Based on Image Processing. In: Proceedings of the 2008 international Conference on Computer Science and Software Engineering, vol. 06. CSSE. IEEE Computer Society, Washington, DC, pp. 491-494, doi: <http://dx.doi.org/10.1109/CSSE.2008.1649>. (2008)
9. Sabine, D. Bauer., Korc, Filip., Forstner, Wolfgang.: The Potential of Automatic Methods of Classification to Identify Leaf Diseases from Multispectral images. In: Springer Science+Business Media, LLC, Precision Agric, 12:361–377, doi: 10.1007/s11119-011-9217-6 (2011)
10. D, Majumdar., D. K, Kole., A, Chakraborty., D. D, Majumder.,: An Integrated Digital Image Analysis System for Detection, Recognition & Diagnosis of Disease in Wheat Leaves. In: Third International Symposium on Women in computing and Informatics, WCI'15, Kochi, India, pp.400-405 (2015).
11. Ren, S., He, K., Girshick, R., Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In: IEEE Trans. Pattern Anal. Mach. Intell. 39, 1137–1149 (2016).
12. Lowe A., Harrison N., French, AP: Hyperspectral image analysis techniques for the detection and classification of the early onset of plant disease and stress. In: Plant Methods. pp. 13:80. (2017)
13. Y. Ampatzidis., L, De, Bellis., A. Luvisi.,: iPathology: robotic applications and management of plants and plant diseases. Sustainability, vol. 9, no. 6, pp. 1010, (2017).
14. R. R. Selvaraju., M. Cogswell., A. Das., R. Vedantam., D. Parikh., D. Batra: Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In: IEEE International Conference on Computer Vision (ICCV), pp. 618-626, doi: 10.1109/ICCV.2017.74. (2017).
15. Madhu, Jadon., Agarwal, Rashi., Singh, Raghuraj., Shilpa, D Kaistha.: A Review on Digital Image Processing Techniques for Automatic Detection, Quantification and Identification of Plant Diseases. In: International Journal of Computer Applications 173(8):14-19, (2017).
16. Romero, Luna, MP., Mueller, D., Mengistu, A., Singh, AK., Hartman, GL., Wise, KA: Advancing our understanding of charcoal rot in soybeans. J Integr Pest Manag;8:8 (2017).
17. M. Raza, M. Sharif, M. Yasmin, M. A. Khan, T. Saba, and S. L. Fernandes: Appearance based pedestrians' gender recognition by employing stacked auto encoders in deep learning. In: Future Generation Computer Systems, vol. 88, pp. 28–39, (2018).
18. Nagasubramanian, K., Jones, S., Singh, A.K.: Plant disease identification using explainable 3D deep learning on hyperspectral images. In: Plant Methods 15, 98 (2019).
19. S. Coulibaly, B. Kamsu-Foguem, D. Kamissoko, and D. Traore: Deep neural networks with transfer learning in millet crop images, In: Computers in Industry, vol. 108, pp. 115–120 (2019).
20. Yan, Guo., Jin, Zhang., Chengxin Yin., Xiaonan, Hu., Yu, Zou., Zhipeng, Xue., Wei, Wang: Plant Disease Identification Based on Deep Learning Algorithms in Smart Farming (2020).