Journal of Information Systems Engineering and Management
2025, 10(18s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Healthy- Disease Mustard Leaf Set: A Dataset for Mustard Disease Detection

Shilpa Juneja[1], Parneeta Dhaliwal[2], Atul Srivastava[3*]
[1]Department of Computer Science and Technology, Manav Rachna University, Faridabad Sec-43, Haryana 121001, India.
E-mail: shilpajuneja11@gmail.com
[2] Department of Computer Science and Technology, Manav Rachna University, Faridabad Sec-43, Haryana 121001, India.
E-mail: parneeta07@gmail.com
[3*] Amity School of Engineering and Technology, Amity University Uttar Pradesh, Lucknow. E-mail: atul.nd2@gmail.com
*Corresponding Author

ARTICLE INFO

Received: 18 Dec 2024

Revised: 28 Jan 2025

Accepted: 12 Feb 2025

ABSTRACT

Introduction: Agriculture is vital to the economy, particularly in developing countries in which it may serve as the primary source of employment as well as revenue. Plant diseases cause India to lose 10-35% of its yearly crop yield. The absence of lab equipment and experience makes early plant disease identification challenging. One of the biggest obstacles to enabling vision-based plant disease identification is the lack of a suitable large-scale non-lab dataset.

Objectives: The main objective of this study is to introduce Healthy-Disease Mustard Leaf Set (HDMLS), a dataset consisting of both healthy and diseased mustard leaves to promote deep- learning based identification of Alternaria Blight and White Rust disease, evaluating its effectiveness in terms of Accuracy, Precision, F1-Score, mIoU, mAP, Specificity and Sensitivity.

Methods: Approximately 350 person hours were spent clicking photos for our dataset, which has 2889 images of mustard plant leaves containing 1 healthy and 4 diseased classes. To increase the diversity of the images and model's capacity for generalization, data augmentation is applied on the dataset increasing its size to 5,663. Pre-processing is done to remove unneseccary features. With an 80-20 split, dataset is separated into training, validation and test sets. Three different state-of-art CNN models- ResNet-50, Xception and DenseNet are applied to both our self-created dataset and PlantDoc dataset which is freely available. Performances are then evaluated and compared.

Results: Confusion matrices have been developed for every model trained on the dataset in order to assess the models. Model Performance is compared using metrics such as precision, recall, accuracy, FPR, F1-Score, Specificity, Sensitivity, m AP, mIoU. 20 epochs are used to evaluate results. Xception model and ResNet-50 model achieve same highest accuracy of 0.88. ResNet-50 model also achieve high mAP (0.70) and mIoU (0.52). DenseNet model also achieves good accuracy (0.84) and mAP (0.63) but it has low mIoU (0.35).

Conclusions: ResNet-50 performs exceptionally well in terms of mAp, , mIoU, and accuracy indicating that it can recognize, classify, and segment objects with superior performance.

Keywords: Agriculture; Mustard Disease Diagnosis; Artificial Intelligence; Deep Learning; HDMLS.

INTRODUCTION

In a country like India, the agriculture sector is very important because the economy is directly or indirectly dependent on agriculture. This emphasizes the need to take care of plants from seedlings to production of the desired product [1]. The prevalence of insect and plant diseases is increasing alarmingly and is a threat to food security. These diseases have significant economic, social and environmental effects and endanger food security. The annual cost of plant diseases alone to the world economy is around US$220 billion [2]. The Indian Council of Agricultural Research reports that diseases and pests cause more than 35% of crop production to be lost annually [3]. Farmers still face problems in timely diagnosis of plant diseases. Other than speaking with other farmers or calling the Kisan Helpline, they are limited in their alternatives [4]. Recognizing diseased leaves requires knowledge of plant diseases. In addition, laboratory equipment is usually required to detect diseased leaves. This study explores the potential of computer vision to identify plant diseases in a scalable and cost-effective manner. The recent development of deep convolutional neural networks has led to significant advances in computer vision. The ability to diagnose plant diseases through image processing opens up new opportunities to apply deep learning techniques to real-world agricultural problems. This leads to improved agricultural knowledge, crop yield and disease control. Most vision-based systems in use today require simple backgrounds and high-resolution images. Guarding crops against plant diseases plays a crucial role in fulfilling the increasing need for both the quality and quantity of food.

This study examines the feasibility of diagnosing plant diseases using computer vision in a scalable and cost-effective manner. In recent years, deep convolutional neural networks have made several advances that have dramatically advanced computer vision. Training large neural networks can take a long time, but once trained, the model can classify images very quickly and is useful for consumer applications on smartphones. By enabling new ways to integrate the understanding of deep learning techniques with real- world agricultural problems, image processing for plant disease detection will help improve agricultural knowledge, crop productivity, and pest management. Most vision-based systems in use today high-resolution images with fixed colour backgrounds. We focus on images taken in natural environments where background noise cannot be ignored, unlike most Indian farmers who use cheap mobile devices with natural light and backgrounds providing the best query resolution for plants and products. With this in mind, we highlight two key contributions:

a) Creation of Dataset: a dataset which originally contains 2889 images of 1 plant species having 5 different classes (4 diseased and 1 healthy).

b) Compare the selected dataset with other dataset and show how it can help identify diseases with more accuracy.

To determine the need for the dataset in an uncontrolled environment, we evaluated the dataset using three different object recognition and classification architectures.

RELATED WORK

The main area of recent research is the use of machine learning and deep learning to diagnose plant diseases. In order to diagnose plant diseases, traditional machine learning approaches like feature extraction and classification have been applied extensively. In order to train a classifier that can differentiate between healthy and unhealthy plants, these techniques extract characteristics from colour, texture and shape of the images.

In this section, we assess the latest methods relevant to the study to create a framework for detecting plant diseases through leaf analysis, a method that has evolved over the years in diverse research endeavours. Also, datasets supporting plant disease detection research have been assessed.

1) Several Deep Learning Techniques for Plant Disease Identification

Pandith et al. [5] analysed whether or not it is possible to forecast the yield of mustard crops by employing techniques from machine learning and conducting soil research. Five distinct supervised machine learning approaches were utilised in the study in order to analyse the data that was gathered. Soil samples were collected from various districts in the Jammu region for mustard production. The findings of this study highlight the importance of soil analysis and the viability of utilising machine learning techniques in the process of crop production forecasting. With a 91% final testing accuracy, a tea leaf disease has been identified using feature extraction and Neural Network Ensembles (NNE) [6].

Using a mobile application, Sameerchand et al. [7] presented a recognition method utilising CNN to categorise 70 medicinal plants with 90.00% accuracy. By applying CNN and SVM to a rice plant disease detection approach, Vimal et al. [8] achieved 91.37% accuracy. Dharani et al. [9] emphasised that agriculture is important to a nation's economy, and timely monitoring and recommendations are crucial for managing agriculture and estimating crop yields. The study highlights the benefits of applying AI to boost crop yields and the need for hybrid and recurrent neural networks in the agricultural industry. An analysis of the results is provided in the article, showing that hybrid networks and RNN outperform other networks like CNN and ANN. Sharma et al. [10] address the importance of plant crop disease identification in agricultural productivity and offer a DL-based multiple classes' model to identify mustard crop diseases. The images were categorised using a CNN into four different levels of illness severity for binary and multi-classification. The research's accuracy in binary classification was 95.6% and its accuracy in multi-classification was 96.66%. For the agricultural sector, the proposed model has the potential to significantly aid in the creation of cost-effective and efficient disease monitoring system. Hridoy et al. [11] suggested a deep CNN that they call MPNet consisting of deep separable convolution layers as well as inception modules. The study used eleven image enhancing techniques to produce a collection of images of mustard plants with nine classes of healthy and diseased leaves. There are 47760 images of leaves, stem and pods in this dataset. MPNet obtained 97.11% accuracy in identifying 2388 test images. Its performance is compared to four cutting-edge CNNs, one of which, MobileNetV2, achieved 92.83% accuracy in the test. According to the results, MPNet has the potential to significantly improve mustard agriculture in the long run and is able to correctly identify diseases that impact the mustard plant.

2) Datasets for disease detection of plants

As far as we know, the PlantVillage dataset [12] is the sole publicly accessible dataset for identifying plant diseases. As the images in the PlantVillage dataset were taken under laboratory conditions rather than in actual agricultural fields, their practical utility is likely to be limited. Acquiring adequate large-scale non-lab datasets is still an important challenge to enabling vision-based plant disease detection. In light of this, a dataset- PlantDoc- for the visual identification of plant diseases was introduced. Although this dataset has been carefully curated, some of the images may be incorrectly identified because of a lack of in-depth field expertise. In comparison we collected real-life images of healthy and diseased mustard leaves.

OBJECTIVES

The objective of this study is to create an extensive dataset for early detection of mustard plant diseases. To support deep-learning based disease recognition, the dataset -which includes both healthy and diseased leaf images- has been carefully created, and augmented. This study aims to gather, categorize and pre-process images in a methodical manner so that deep learning models may be trained with them and dataset's effectiveness can be evaluated by training three different deep learning models. This study also contributes to the development and advancement of the scalable framework so that farmers and researchers can be assisted in improving mustard crop production.

METHODS

The PlantDoc dataset which can be openly accessed contains a smaller number of images. Because many leaves may actually be present in plant images with variable lighting conditions and background types, this dataset reduces the efficiency of disease detection. In light of this, we now go through the carefully collected dataset which we call HDMLS and the techniques applied on this dataset. Models trained on real-world images are necessary to account for real-world subtleties. This fact inspired us to collect the dataset. Mustard oil is made out of mustard seeds and very popular in India as it is used for cooking. Mustard production is one of the most profitable agricultural practices because it offers a huge market and economic value [6]. But when it comes to growing mustard plants, it leads to intensive disease category types which causes heavy financial loss. An early and accurate detection of mustard diseases, is necessary to prevent financial losses in the production system as well to reduce a need for spraying with agro-chemicals that can bring long-term ecological sustainability maintaining food security. Accurate and timely diagnosis of mustard diseases is essential to minimize financial losses and reduce pesticide use, both of which contribute to sustainable development in agriculture. In an effort to increase productivity, deep learning techniques for plant disease detection have recently emerged as a significant area of study in agriculture nowadays.

We collected 2889 images of 5 classes of mustard plant leaves which include healthy and diseased leaves. Five classes of diseased leaves are White Rust at early stage, White Rust at middle stage, White Rust at later stage, Alternaria Blight and Healthy. Each category represents a specific condition or growth stage of the mustard plant. We applied data augmentation to our self-collected mustard leaves dataset to enhance diversity of the images and improve model's ability to generalize. Total number of images after applying data augmentation techniques were 5,663. These images are the foundation for training and evaluating the deep learning models in this study. The dataset contains images of mustard leaves with different diseases. It is divided into training, validation, and test sets with an 80-20 split for training and validation. Number of images in HDMLS after augmentation used in this study is given below:

1. **Alternaria Blight: 1,235 images.**
2. **White Rust at Early Stage: 2,443 images**
3. **White Rust at Middle Stage: 706 images**
4. **White Rust at Later Stage: 274 images**
5. **Healthy: 1,005 images**

Several augmentation techniques were applied to the original images, including:

- **Rotation:** Random rotation of images by up to 40 degrees.
- **Shifting:** Horizontal and vertical shifting of images by 20%.
- **Shear:** A slight shearing transformation to simulate perspective changes.
- **Zooming:** Zoom in or out of images by up to 20%.
- **Horizontal Flip:** Random horizontal flipping of images.
- **Fill Mode:** Filling missing pixels using the nearest value during transformations.

These techniques created new variations of the original images, effectively increasing the size and variability of the dataset. These techniques provided additional training data that helps the model perform better by reducing over fitting and improving accuracy in real-world scenarios. This approach ensures the model can recognize mustard leaves from different angles, sizes, and positions, enhancing its ability to classify various stages and conditions of the plant. Figure 1 shows the statistics of HDMLS.

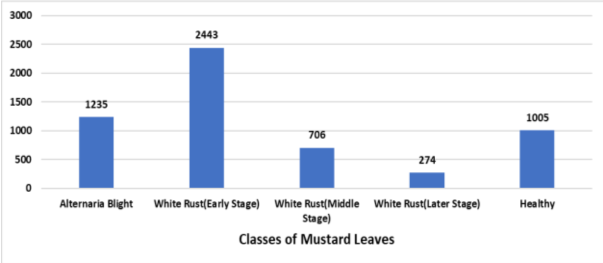


Figure 1. Statistics of classes of mustard leaves in HDMLS

1) Collection of Dataset

Seasonal changes, as well as other changes such as temperature, fog can also affect mustard leaf diseases. With this in mind, mustard plant leaves are collected from a variety of weather conditions for general use. Our main focus has been capturing images of White Rust, Alternaria Blight diseased leaves along with healthy leaves with the intention of increasing the size of our dataset. Images of White Rust dataset is comprised of three distinct disease classes to classify the images into three stages according to varying degree of intensity. Images of white rust diseased leaves are considered in early stage if disease is present in less area. They are considered in middle stage if diseased area is comparatively high. If diseased area is very high then images are considered in later stage. Real-time data collected from the mustard fields in Sanp Ki Nangli Village, Sohna, Haryana, have been analysed to gain insights into the agricultural practices in the region. The Panasonic Lumix DSLR camera, boasting 16 megapixels of effective pixels and measuring 8*89*73 mm, was employed to capture the images for the dataset. This implies that the image captured had dimensions of 8 mm in length, 89 mm in width, and 73 mm in height. A total of 2889 5184*3456 pixel images ranging in size from 6-7 megabytes (MB) have been gathered. The collection of images varies from healthy mustard crop images to diseased ones. It is simply because all the settings available from taking images within camera are default parameters. These details will be adjusted as the job moves through pre-processing to meet what is needed by this model along with achieving the best performance in detecting Alternaria Blight and White Rust diseased images. Image samples of White Rust, Alternaria Blight diseased and healthy images of mustard leaves are shown in Figure 2.

a) White Rust b) Alternaria Blight c) Healthy
Figure 2. Sample diseased and healthy mustard leaves in HDMLS

2) Dataset Pre-processing

Since data cleaning eliminates noise and unnecessary features that are not needed for experimental work, data pre-processing is crucial. This step is all about bringing the data in alignment as the collected images have different sizes and resolutions. For pre-processing, geometric transformation was used to ensure uniform size and resolution. Applied to all the images in our dataset, this twisted the result into proper alignment against rotation, translation and scale. Each image was enlarged and resized during the scaling phase. In the translating phase, pixels were translated as well. At the end, we rotated the images by changing a rotation. Following pre-processing has been done on the collected dataset for all the models which is given below:

Explicit Preprocessing:

— Resizes images to 180x180 pixels.

— Batches the images into groups of 32.

— Encodes labels into one-hot vectors for multi-class classification.

Implicit Preprocessing by TensorFlow:

— Loads and splits data into training and validation sets.

— Resizes images to a consistent size (180x180).

— Shuffles the training set.

— Batches the images for efficient processing.

Before Preprocessing:

— Images may vary in size and format, with inconsistent labels.
— No batching or efficient handling of data.
— Clutters, shadows in an image.

Figure 3, 4, 5, 6 and 7 show the images of diseased and healthy leaves before and after pre-processing.

Before After

Figure 3. White Rust leaf at early stage

Before After

Figure 4. White Rust leaf at middle stage

Before After

Figure 5. White Rust leaf at later stage

Before After

Figure 6. Healthy Leaf

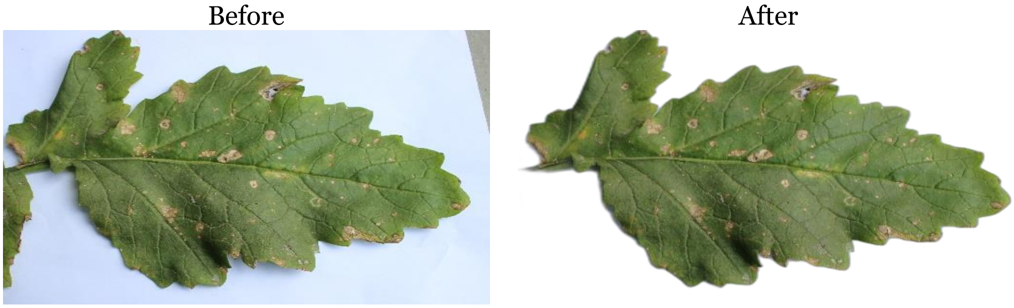


Figure 7. Alternaria Blight Leaf

After Preprocessing:

— All images are uniformly resized, and labels are one-hot encoded.

— The dataset is properly batched and split for efficient model training.

— The pre-trained model's weights provide a strong starting point for classification, with the added layers adapting to the specific problem.

3) Training of DL architectures

In this study, three deep learning models are employed to classify mustard leaf images into various categories, such as Alternaria Blight, Early-Stage White Rust, Middle Stage White Rust, Later Stage White Rust, and Healthy. These models—ResNet-50, Xception, and DenseNet—were selected due to their proven performance in image classification tasks, especially in the field of plant disease detection. Experiment has been done on the self-collected dataset. Out of total images, 80% of images are used to train the model and 20% of the images are used to test the models.

The experiments were carried out on a machine equipped with the following hardware and software specifications:

— **Processor:** 11th Gen Intel(R) Core (TM) i3-1115G4 @ 3.00GHz, offering sufficient processing power for standard deep learning tasks.
— **Memory:** 8.00 GB of installed RAM (7.70 GB usable), enabling effective handling of small to moderate datasets.
— **Architecture:** A 64-bit operating system, x64-based processor, ensuring compatibility with modern machine learning libraries.
— **Operating System:** Windows version 23H2, ensuring up-to-date system stability and security.

For this study, we employed Keras and TensorFlow as the primary deep learning frameworks. Despite the limited computational resources compared to high-end GPUs, this setup proved capable of executing the models for smaller-scale deep learning applications and testing purposes.

Below is an overview of each model's architecture and the reason of being chosen for this study.

I. ResNet-50:

ResNet-50 is a deep convolutional neural network (CNN) that uses residual learning to improve training performance. It is based on the ResNet architecture, which was introduced to tackle the problem of vanishing gradients in very deep networks. The "50" in ResNet-50 refers to the 50 layers in the network. Architecture of ResNet-50 is given below:

a) Pre-trained ResNet50 Model

The backbone of the model is the ResNet50 architecture, which is a pre-trained model from Keras, initialized with weights from ImageNet. This model consists of 50 layers and is widely used for image classification tasks due to its proven ability to perform well in a wide range of image recognition problems. In this study, we used ResNet50 with the following configuration:

Include Top: False argument means that the fully connected layers (top layers) are excluded. This allows us to add custom layers that are more suitable for our specific task (mustard leaf classification).

Input Shape: The input shape is set to (180, 180, 3), meaning the images are resized to 180x180 pixels and consist of three-color channels (RGB).

— **Weights:** The model is initialized with pre-trained weights from ImageNet to leverage knowledge from a large dataset, improving performance on our smaller mustard leaves dataset. By freezing the layers of ResNet50, we prevent the weights from being updated during training, which helps retain the useful features learned from ImageNet. Only the custom layers added on top are trained.

b) Custom Convolutional Layers:

After the ResNet50 backbone, we add several custom convolutional layers to refine the model for the mustard leaf classification task called Conv2D Layers. The specifications are as follows:-

○ *First Conv2D Layer:* 64 filters with a kernel size of (3, 3), followed by BatchNormalization, ReLU Activation, and MaxPooling with a pool size of (2, 2).

○ *Second Conv2D Layer:* 128 filters with a kernel size of (3, 3), followed by BatchNormalization, ReLU Activation, and MaxPooling.

○ *Third Conv2D Layer:* 256 filters with a kernel size of (3, 3), followed by BatchNormalization, ReLU Activation, and MaxPooling.

These layers help extract features at different levels of abstraction, from simple edges to more complex patterns, improving the model's ability to understand and classify mustard leaf stages or diseases.

c) Global Average Pooling Layer

After the convolutional layers, we use Global Average Pooling. This operation reduces the spatial dimensions of the output from the convolutional layers, producing a single value for each feature map. This allows the model to focus on global patterns rather than local ones. The output of this layer is a vector that can be passed to fully connected layers.

d) Fully Connected (Dense) Layers

After the feature extraction and pooling, the model adds a fully connected layer to make the final classification. **Dense Layer** has 512 neurons and a ReLU activation function, followed by a Dropout layer with a rate of 0.5 to prevent overfitting. **Output Layer** has a number of neurons equal to the number of classes (5 in this case: Alternaria Blight, Early-Stage White Rust, Healthy, Later Stage White Rust, and Middle Stage White Rust). The activation function used here is Softmax, which outputs the class probabilities and selects the class with the highest probability.

e) Model Compilation and Training

The model is compiled with Adam optimizer having a learning rate of 0.0001, which is known for its efficiency in training deep learning models. Categorical Crossentropy is used as loss function which is suitable for multi-class classification problems. Then the model is trained with training and validation datasets. The training process uses 20 epochs, meaning the model will go through 20 complete cycles of the training data to adjust its weights.

II. Xception

Xception (Extreme Inception) is an architecture that builds on the Inception model, but with the key innovation of replacing the standard Inception modules with depthwise separable convolutions. This makes the model more efficient and effective in capturing complex patterns in image data. Below are the key components of the architecture:

a) Pre-trained Xception Model

— **Input Shape:** (180, 180, 3) – images resized to 180x180 pixels with 3 color channels (RGB).

— **Include Top:** False – Excludes the top classification layers of Xception.

— **Weights:** Pre-trained on ImageNet, which is used to initialize the model.

b) Frozen Layers

All layers of the pre-trained Xception model are frozen to prevent weight updates during training, allowing the model to focus on learning from the additional layers added to the network.

c) Custom Convolutional Layers

— **Convolution Layer 1:** 64 filters, kernel size (3, 3), ReLU activation, followed by max pooling with a pool size of (2, 2).

— **Convolution Layer 2:** 128 filters, kernel size (3, 3), ReLU activation, followed by max pooling with a pool size of (2, 2).

— **Convolution Layer 3:** 256 filters, kernel size (3, 3), ReLU activation, followed by max pooling with a pool size of (2, 2).

d) Global Average Pooling Layer

The output from the final convolutional layer is passed through a Global Average Pooling layer to reduce the dimensionality and prevent overfitting.

e) Fully Connected Layers

— **Dense Layer:** 512 neurons with ReLU activation.
— **Dropout Layer:** To prevent overfitting, a Dropout layer with a rate of 0.5 is used.
— **Output Layer:** A softmax layer with neurons equal to the number of classes (5).

f) Compilation and Training

The model is compiled with Adam optimizer having a learning rate of 0.0001, which is known for its efficiency in training deep learning models. Categorical Crossentropy is used as loss function which is suitable for multi-class classification problems. Then the model is trained with training and validation datasets. The training process uses 20 epochs, meaning the model will go through 20 complete cycles of the training data to adjust its weights.

III. DenseNet

In DenseNet, each layer receives input from all preceding layers, allowing the network to reuse features and improve the flow of information. The DenseNet121 model consists of the following components:

a) Pre-trained DenseNet121 Backbone:
We utilize the DenseNet121 architecture, which is a deep CNN model that employs dense connections between layers. This architecture helps mitigate the vanishing gradient problem, enhances feature propagation, and reduces the number of parameters. The pre-trained model is loaded without its top layers (include_top=False) and uses ImageNet weights (weights='imagenet'). This allows the model to benefit from transfer learning, leveraging features learned from a large dataset like ImageNet.

b) Custom CNN Layers:

After the pre-trained DenseNet model, the following layers are added to further refine the model for the mustard plant classification task:

— **Convolutional Layers:** Three convolutional layers (with 64, 128, and 256 filters) to capture more high-level patterns. These layers use a kernel size of 3x3.

— **Batch Normalization:** Each convolutional layer is followed by batch normalization to improve the convergence of the model by normalizing activations.

— **Activation Function:** ReLU activation is applied after each convolutional and batch normalization layer to introduce non-linearity.

— **Max Pooling:** Max pooling is used after each convolutional layer to reduce the spatial dimensions and retain the most important features.

c) Global Average Pooling:
This layer reduces the feature maps to a single vector by averaging over spatial dimensions, helping to summarize the features effectively.

d) Fully Connected (Dense) Layers:
— A dense layer with 512 neurons and ReLU activation is used to connect the extracted features to the final output layer.

— A dropout layer is applied with a rate of 0.5 to prevent overfitting during training.

e) Output Layer:

A softmax activation function is applied to the final dense layer to produce probability scores for each class. The number of output neurons equals the number of classes (5), corresponding to the plant conditions (Alternaria Blight, Early Stage Whiterust, Middle Stage Whiterust, Later Stage Whiterust, Healthy).

f) Training Process:

The model was trained for 20 epochs using the Adam optimizer with a learning rate of 0.0001. The loss function used for training was categorical crossentropy, which is appropriate for multi-class classification problems.

RESULTS

We applied three different state-of-art CNN models- ResNet-50, Xception, DenseNet- on our self-created dataset to recognize the diseases in mustard leaves and three models- ResNet-50, Xception, DenseNet- on PlantDoc dataset which is freely available. We considered images of only Tomato, Grape, and Apple leaves for training above three models. Table I shows the comparison of performance of the trained models on both the datasets.

Table I. Comparison of performance of three models on HDMLS and PlantDoc datasets

Model	HDMLS			PlantDoc Apple Leaf Dataset			PlantDoc Tomato Leaf Dataset			PlantDoc Grape Leaf Dataset		
	Accuracy	mIoU	mAP	Accuracy	mIoU	mAP	Accuracy	mIoU	mAP	Accuracy	mIoU	mAP
ResNet50	0.88	0.52	0.79	0.82	0.47	0.61	0.86	0.19	0.27	0.70	0.46	0.84
Xception	0.88	0.42	0.68	0.86	0.29	0.43	0.82	0.10	0.12	0.55	0.27	0.29
DenseNet	0.84	0.35	0.63	0.68	0.34	0.60	0.82	0.09	0.18	0.55	0.27	0.29

For evaluating the models, confusion matrices have been created for each model trained on the dataset. Precision, Recall, Accuracy, FPR, F-Score, Specificity, Sensitivity, mIoU and mAP are used for the comparison of the performance of the models. Results are evaluated for 20 epochs.

Confusion matrices of ResNet-50, Xception and DenseNet models are shown in figure 8.

1090	1	144	0	0
20	1570	847	1	5
90	18	897	0	0
68	20	24	149	13
87	205	125	16	273

a. ResNet-50

1029	62	77	4	63
117	1727	411	7	181
92	134	741	3	35
69	62	32	62	49
42	196	79	4	385

b. Xception

1059	25	140	9	2
66	1563	790	20	4
117	161	719	7	1
29	121	74	49	1
50	337	251	28	40

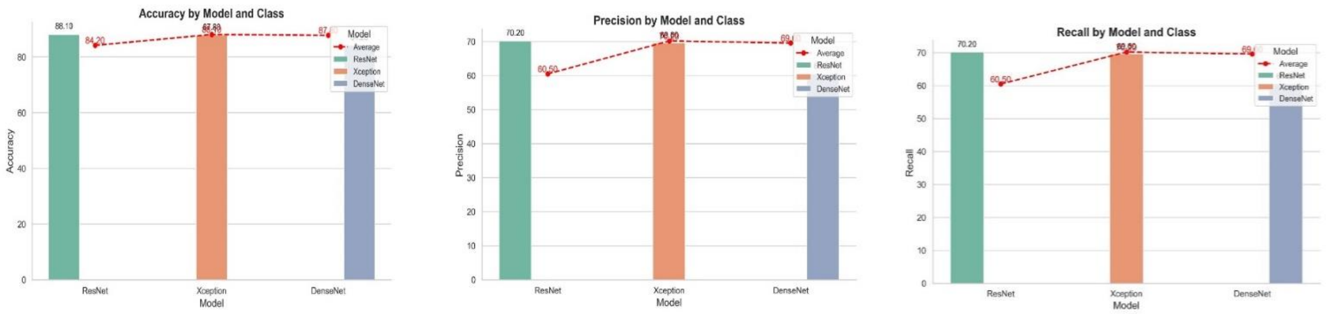
c. DenseNet

Figure 8. Confusion Matrix

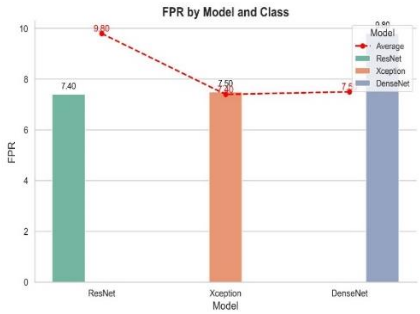
Table II. Performance of models on HDMLS

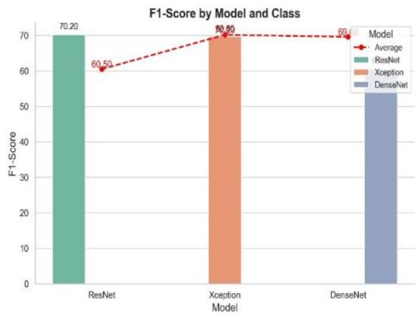
Metric	ResNet-50	Xception	DenseNet
Precision	0.70	0.70	0.60
Recall	0.70	0.70	0.60
Accuracy	0.88	0.88	0.84
FPR	0.07	0.07	0.09
F1-Score	0.70	0.70	0.60
Specificity	0.92	0.92	0.90
Sensitivity	0.70	0.70	0.60
mIoU	0.52	0.46	0.35
mAP	0.79	0.68	0.63

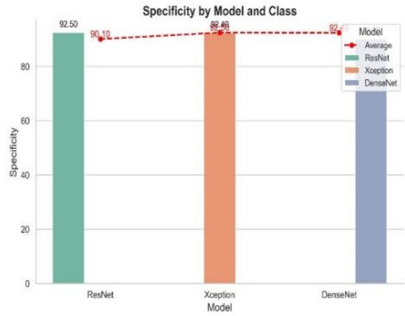
Table II. displays the evaluation metrics of the ResNet-50, Xcepetion and ResNet models to detect the mustard leaf diseases. Similar information, in comparative view, is graphical represented in the figure 9.



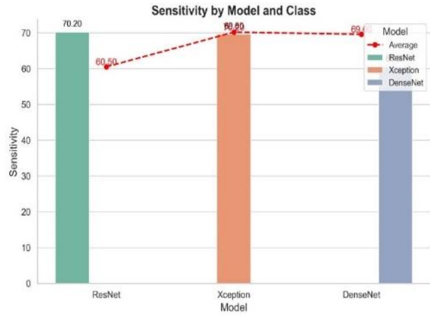
a. Accuracy **b. Precision** **c. Recall**







d. FPR **e. F-1 Score** **f. Specificity**



g. Senstivity

Figure 9. Comparitive view of performance metrics for all three models

DISCUSSION

In this research, we used cutting-edge object detection models to combat the problem of identifying healthy or diseased leaves in images. One of our primary contributions is the proposal of Healthy- Disease Mustard Leaf Set, an entirely novel dataset for the diagnosis of mustard plant diseases. Among all the models, ResNet-50, Xception models achieves the same accuracy (0.88) that is highest which indicates that they are the most effective in classifying the samples correctly. However, ResNet-50 model has high mAP (0.70) and mIoU (0.52). This indicates that ResNet-50 model is much effective in localization and segmentation tasks. DenseNet model also achieves good accuracy (0.84) and mAP (0.63) but it has low mIoU (0.35). This indicates that the model is good at detecting presence of diseases and classifying them but it is not accurately delineating their boundaries or predicting the shapes of the diseased areas at the pixel level. Xception model also achieves good mAP (0.68) but it has low mIoU (0.46). This indicates that it performs well in terms of identifying diseased area and classifying them correctly but it is also struggling like DenseNet with the precise localization of areas particularly in context of segmentation tasks. ResNet-50 excels in mAP, mIoU and accuracy which suggests the model is able to identify, categorise, and segment objects with high performance among all.

REFRENCES

[1] Strange, R.N. and Scott, P.R., 2005. Plant disease: a threat to global food security. Annu. Rev. Phytopathol., 43(1), pp.83-116. https://doi.org/10.1146/annurev.phyto.43.113004.133839.

[2] GN Agrios. 2005. Plant pathology 5th Edition: Elsevier Academic Press. Burlington, Ma. USA (2005), 79–103.

[3] T Mohapatra. 2018. ICAR News July-September 2018. Published in monthly newsletter, https://www.icar.org.in/sites/default/files/ ICARNewsJulySeptember2018.pdf.

[4] Government of India. 2019. Kisan Knowledge Management System. https://dackkms.gov.in/account/login.aspx.

[5] V. Pandith, H. Kour, S. Singh, J. Manhas, and V. Sharma, "Performance Evaluation of Machine Learning Techniques for Mustard Crop Yield Prediction from Soil Analysis," J. Sci. Res., vol. 64, no. 2, pp. 394–398, 2020. doi: 10.37398/JSR.2020.640254.

[6] Zhou, Z.H. and Chen, S.F., 2002. Neural network ensemble. CHINESE JOURNAL OF COMPUTERS-CHINESE EDITION-, 25(1), pp.1-8.

[7] Sameerchand Pudaruth, Mohamad Fawzi Mahomoodally, Noushreen Kissoon, and Fadil Chady, "MedicPlant: A mobile application for the recognition of medicinal plants from the Republic of Mauritius using deep learning in real-time," in IAES International Journal of Artificial Intelligence (IJ-AI), vol. 10, pp. 938-947, 2019. doi: 10.11591/ijai.v10.i4.pp938-947.

[8] Vimal K. Shrivastava, Monoj K. Pradhan, Sonajharia Minz, and Mahesh P. Thakur, "RICE PLANT DISEASE CLASSIFICATION USING TRANSFER LEARNING OF DEEP CONVOLUTION NEURAL NETWORK," in Int. Arch. Photogramm. Remote Sens. Spatial Inf., vol. XLII-3/W6, pp. 631–635, 2019. https://doi.org/10.5194/isprs-archives-XLII-3-W6-631-2019.
[9] M. K. Dharani, R. Thamilselvan, P. Natesan, P. C. D. Kalaivaani, and S. Santhoshkumar, “Review on crop prediction using deep learning techniques,” J. Phys. Conf. Ser., vol. 1767, no. 1, pp. 10–26, 2021. doi: 10.1088/1742-6596/1767/1/012026.
[10] R. Sharma and V. Kukreja, “Mustard Downy Mildew Disease Severity Detection using Deep Learning Model,” in International Conference on Decision Aid Sciences and Application (DASA), 2021, pp. 466–470. doi: 10.1109/DASA53625.2021.9682305.
[11] Hridoy RH, Arni AD, Hassan MA. Recognition of Mustard Plant Diseases Based on Improved Deep Convolutional Neural Networks. In2022 IEEE Region 10 Symposium (TENSYMP) 2022 Jul 1 (pp. 1-6). IEEE., doi: 10.1109/TENSYMP54529.2022.9864487.
[12] Singh D, Jain N, Jain P, Kayal P, Kumawat S, Batra N. PlantDoc: A dataset for visual plant disease detection. InProceedings of the 7th ACM IKDD CoDS and 25th COMAD 2020 Jan 5 (pp. 249-253). doi: 10.1145/3371158.3371196.