



# Plant Leaf Disease Detection Using Resnet-50 Based on Deep Learning

S. Jayashree\* <sup>1</sup>, Dr. V. Sumalatha <sup>2</sup>

<sup>1</sup> Research Scholar, Assistant Professor, VISTAS, Chennai

<sup>2</sup> Research Supervisor, Associate Professor, VISTAS, Chennai.

shreeologu23@gmail.com

**Abstract.** India's agriculture permits the world food chain by producing various crops and boosting the country's economy. Diseases pose a significant challenge to agricultural production. It causes crop disruption, lowers output, and makes it extremely hard for farmers to compensate for planting damage. Early disease detection and rapid action are essential to preventing productivity loss. Currently, several methods for analyzing illness characteristics and figuring out the stage of progression use Machine Learning (ML) for image processing. However, because disease features vary, it is challenging to determine the regional segments. Unbalanced traits can complicate the detection of diseases. To resolve this problem, initially, we collected the plant image dataset from Kaggle. We applied pre-processing steps, including Gaussian and Wiener filters, to normalize plant leaves. Furthermore, plant leaf features can be selected using the Canny Region Extraction (CRE) technique for non-edge and smoothing. Moreover, the Multi-level Threshold Segmentation (MRDS) method can identify pixel groups and classify the optimal values. Finally, the proposed ResNet50 Optimal Convolutional Neural Network (ROCNN) method can categories the results to obtain binary plant classification. As a result, accuracy for plant leaf diseases can be obtained using high false rates, imprecise recognition, high precision, F-measure and low recall efficiency.

**Keywords:** Plant leaf, Machine Learning, CRE, Gaussian and Wiener filters, MRDS, ROCNN, classification, accuracy, Recall, and F-measure.

## 1 Introduction

In today's times, even if the population develops exponentially, it is possible that agriculture can feed all humanity. Therefore, advanced technology can meet society's needs by allowing them to produce enough food. Plant diseases in agriculture can be predicted in advance based on the entire country's food supply through a predictive system. In addition, food is a significant source of employment and income, and agriculture represents a significant contribution to the global economy. Early detection of plant diseases can lead to better agricultural production management decisions. Furthermore,

© The Author(s) 2024

N. V. Suresh and P. S. Buvaneswari (eds.), *Proceedings of the International Conference on Digital Transformation in Business: Navigating the New Frontiers Beyond Boundaries (DTBNF 2024)*, Advances in Economics, Business and Management Research 283,

[https://doi.org/10.2991/978-94-6463-433-4\\_12](https://doi.org/10.2991/978-94-6463-433-4_12)

early monitoring of plant diseases can determine better agricultural production management decisions. Furthermore, isolated spots on stems, fruits, leaves or flowers can be observed on infected plants to detect abnormalities of each infectious disease or pest period, leaving unique arrangements. [1-2].

Plant pest and disease diagnosis is mentioned as one of the essential research topics based on machine vision. The technology permits images collected using machine vision to determine the presence of pests or diseases in collected images of plants. Moreover, plant disease and insect detection tools were first used in agriculture based on machine vision. Advanced, primitive optical recognition is converted to a certain magnitude [3].

Plant diseases are susceptible to various attacks due to many environmental pathogens, such as plant diseases or infections caused by environmental factors. Infection by plant pathogens is also a major cause of crop yield decline worldwide. In addition, plants are more likely to be affected by different groups of pathogens individually or by multiple pathogens, resulting in more severe disease outbreaks. However, plant diseases threaten food safety by damaging crops, reducing food provisions, and increasing food costs [4].

Nevertheless, plant pathology is the scientific learning of plant diseases, highlighting how to treat and prevent the conditions that cause plant disease. In addition, parasites cause diseases like bacteria, fungi, viruses, roundworms and other plant pathogens. Accurate and precise timing is necessary for plant disease administration and restraint, and traditional methodology are costly and consuming the time. In addition, plant diseases severely affect the agricultural industry, resulting in reduced crop yields and economic losses [5].

In this paper contribution, we initially collected a dataset of plant leaf disease from Kaggle. Secondly, we apply a preprocessing step to normalize the noise in the plant image using Gaussian and Wiener filters. Moreover, the CRE method can select plant leaf features based on non-edge and smoothing. Furthermore, the MRTS method detects and predicts groups of pixels. Based on this, a consistent classifier threshold separation between classes can be achieved. Finally, the proposed ResNet50 model is classified and provides a practical output for plant foliar diseases based on binary classification.

## 2 Literature Survey

The author suggested that current plant leaf disease diagnosis trends can be identified using progressed imaging protocols and DL [6]. The author proposed an emerging fungal plant leaf disease and defined their external morphology. However, more attention is paid to factors that control local variation to species ecology [7]. The author proposed

that data augmentation techniques can be used for neural network training based on low-quality test images to improve the accuracy of DL models [8]. The author suggested that in situ images captured with diverse resolution camera equipment can be used to detect pests and diseases in tomato plants by implementing a DL-based method [9]. The author proposed implementing a GAN-based leaf Generation Adversarial Network (GAN) generating four flag leaf disease images to train a respect classical [10]. The author proposed that artificial images of tomato plant leaves can be generated using Conditional GAN (C-GAN) to monitor a DL-based tomato disease diagnosis system [11]. A novel approach using Long Short-Term Memory (LSTM) networks can be demonstrated to outperform currently presented plant disease management methods [12]. The novel reported that various valuable applications of imaging techniques and computer vision methods could be implemented and reviewed for plant disease diagnosis and classification [13]. The author proposed that ML and image processing techniques can be implemented to evaluate recent crop pest and disease diagnosis investigations widely. However, it is expensive, time-consuming and sometimes impractical [14]. The novel suggested a pipeline for consuming GANs in unsupervised image interpretation to track data sharing in plant disease datasets. However, the classification decision margin for better performance minimizes the biases introduced due to severe class imbalance [15].

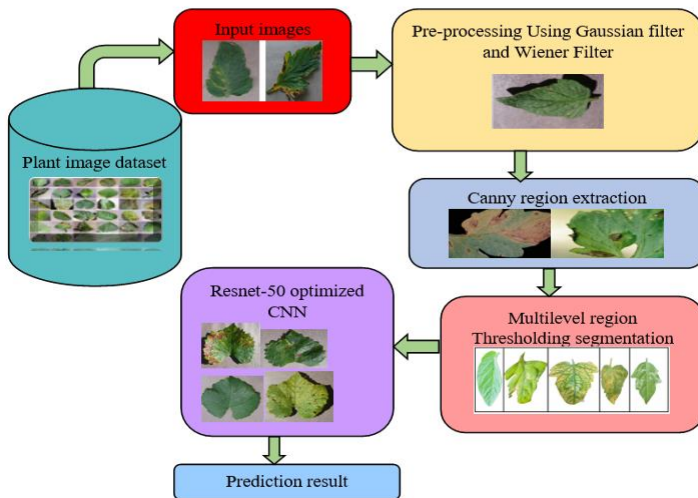
The novel reported that a Deep Convolutional Neural Network (DCNN) transfer learning method can be implemented on a large scale, using pre-trained models learned from datasets for plant leaf disease detection [16]. The author proposed that experts can develop different systems to prophet rust, root/leaf blight, fungal diseases, powdery mildew. However, plant diseases induce various losses in agricultural production and economy [17]. The author mentioned that a Leaf Wetness Sensor (LWS) can be developed locally to capture the LW Duration (LWD). Commercially obtainable temperature and moisture sensors can be used by various methods to record ambient temperature and humidity [18]. The novel suggested that various problems can be addressed through traditional plant pest and disease diagnosis methods [19]. The author proposed that a modern automatic image recognition system based on DL can advance the processes for early plant diseases detection. However, monitoring plant diseases remains a significant challenge globally in ensuring sustainable food agriculture [20].

### **3 Proposed Methodology**

In this proposed section, advances in image processing technology can identify and predict diseases by identifying and analyzing material in plant leaves cut into different areas. To make matters more challenging, the organizational construction of illness impact profiles cannot be standardized because most existing methodologies need to rec-

ognize the significance of feature dimensions. From Kaggle, the dataset is initially gathered to identify the plant leaf diseases. Normalizing the noise in the plant photos through a preprocessing phase that uses Gaussian and Wiener filters to increase the analysis's accuracy is necessitated to optimize the results.

The CRE approach makes it possible to choose leaf features that lack edges and are smooth. The MRTS approach also predicts and recognizes pixel groups, enabling uniform classifier threshold separation across all categories. To discover probable plant leaf illnesses through binary classification, the ROCNN model for variety is effectuated. As a result, if the categorization is accurate, the precision rate, recall rate, and F1 score will be high, making it easier to detect diseases early and comprehend their effects. The deployed diagram of workflow architecture was presented in the Fig 1 as illustrated below.



**Figure 1. Architecture Diagram for Proposed Resnet50-CNN Dataset**

This section contains a plant dataset, including healthy and unhealthy images. Since plant is a significant commercial crop, we obtained this dataset to investigate plant diseases that affect agriculture growth. It would be helpful to capture more images and collect additional plant image. Plant diseases, especially fungal outbreaks, can be explored using this dataset.

### 3.1 Preprocessing

This section applies pre-processing techniques using Gaussian and Wiener filters. The Gaussian distribution, also known as the normal distribution, generates normalized images of plant leaves. The expression below is frequently employed in this process (Equation 1). Where  $G$  –Gaussian distribution.

$$G(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-a)^2}{2\sigma^2}} \quad (1)$$

In this case, we can assume that the spread average or statistical expectation parameter is responsible for shifting the distribution towards zero on the x-axis:  $a=0$ , so it works in a simple form (Equation 2).

$$G(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x)^2}{2\sigma^2}} \quad (2)$$

The function in question is a negative exponential function, with one of the parameters being the square. The parameter  $\sigma$  functions as a scaling factor and is also known as the standard deviation, with the variance being the square of  $\sigma^2$ . It is important to note that the function  $x \in (-\infty, \infty)$  is bounded along the real axis. In simpler terms, the process extends to infinity on both the left and right sides.

Use the same Gaussian filter to apply it to the image horizontally and vertically. Since the function is commutative, any strategy will succeed. This Wiener filter takes into account image noise, as well as first and second-order statistics, to create a reconstructed filter transfer function. The minimizing MSE  $J(g) = E[E[e^2(n)], g(k)]$ 's-find the optimum. The MSE is given by Equation 3.

$$J(g(n)) = E[e^2(n)] \quad (3)$$

It also improves the linear estimation of images by minimizing Mean Square Error (MSE). However, it's important to note that this assumes that any wide sensitivity in the leaf image field has been corrected.

### 3.2 Canny Region Extraction (CRE)

There are two methods of seed selection: smoothing and non edging. Canny is an edge detection tool in image processing that uses gradient histograms to implement efficient operator thresholding. Grayscale gives better results, so selecting grayscale images is a good choice for edge detection and image segmentation.

Equations 4, C0, and C are selected and calculated as the two grey scale images.

$$\mu_0(T) = G \frac{1}{w_0} \sum_{o < k < 255} k \cdot p(k) \quad \& \quad \mu_1(T) = \frac{1}{w_1} \sum_{o < k < 255} k \cdot p(k) \quad (4)$$

Here,  $\mu_0, \mu_1$  is mean by  $C_0$  and  $C_1$ . Hence, in equation 5, they calculate the mean values.

$$\mu(T) = w_0 \mu_0(T) + w_1 \mu_1(T) \quad (5)$$

Here define the  $w_0$  and  $w_1$  are probability of  $C_0$  and  $C_1$  and  $\mu$  is the image, and the variances are calculated by equation 6.

$$G(T) = w_0(\mu - \mu_0)^2 + w_1(\mu - \mu_1)^2 \quad (6)$$

Here, the RGB and XC values are the midpoint, and the extended distance is Equation 7.

$$\mu = \underset{i=1 \dots 8}{\text{Max}} \|x - x_c\| \quad \& \quad \mu < T1 \quad (7)$$

Here, T1 has a predefined threshold,  $X^X y$  is a smooth value, 26 levels of total RGB color are 17576, and the maximum RGB color resolution is 17000.

### 3.3 Multi-level Region Thresholding Segmentation (MRTS)

In this section, we optimize multi-level threshold segmentation in grayscale. Some pixels can be discovered in groups that best describe an objective function. Standard classifiers based on objective functions like lowest error, inter-class variance, and maximum entropy can be used for threshold segmentation. The MRTS method entropy approaches optimize the accurate process in the overall system design. The lowest cross-entropy approach is used to optimize the multi-level threshold method. MRTS method was also more accurate regarding variance and entropy. Based on the objective functions of the two methodologies, the suggested algorithm can carry out multi-level thresholding.

The calculation of the pixel grey probability is shown in Equation 8. Let's assume, q-pixel value, H- pixel image,  $n_z \times n_d$ -pixel number of row and column,  $M \in \{0, 1, 2, \dots, Z-1\}$ - grey level set.

$$q_M = \mu \frac{H_M}{n_z \times n_d}, \quad M \in \{0, 1, 2, \dots, Z-1\} \quad (8)$$

Calculate the mean of the pixel probabilities and grey-level probabilities for the class, as shown in Equation 9 and 10. Where,  $K_M$  -probability of pixel class,  $\mu_M$  - Probability Mean,  $q_M$  -Probability of pixel class.

$$K_M = \sum_{L=T_M}^{T_M+1} q_M \quad (9)$$

$$\mu_M = \sum_{L=T_M}^{T_M-1} L \cdot \frac{q_M}{K_M} \quad (10)$$

The mean grey level of the image is calculated as shown in Equation 11. Where  $\mu_M$  - Probability of Mean.

$$\mu = \sum_{M=0}^{Z-1} M \cdot q_M \quad (11)$$

Calculating the difference between region classes is shown in Equation 12. Where,  $\sigma^2$  - maximum threshold, W-grey level Threshold.

$$\sigma^2 = \sum_{M=0}^W K_M \cdot (\mu_M - \mu)^2 \quad (12)$$

Calculate a prediction of the region entropy, shown in Equation 13. Let's assume, T-threshold,  $\psi$  - maximum entropy.

$$\psi = - \sum_{M=0}^{T_1-1} \frac{q_M}{K_0} 1N \frac{q_M}{K_0} - \sum_{M=T_1}^{T_2-1} \frac{q_M}{K_1} 1N \frac{q_M}{K_1} \dots \sum_{M=T_W}^{Z-1} \frac{q_M}{K_W} 1N \frac{q_M}{K_W} \quad (13)$$

In this segment, maximum entropy multilevel thresholding is considered an over-threshold detection procedure. Furthermore, Otsu performs maximization between multilevel inputs of class variance.

### 3.4 Resnet-50 optimized CNN

The ResNet50 model, one of the transfer learning techniques described previously, is used as a starting point rather than building a neural network from scratch to identify the disease region. To utilize the region segmentation to utilize a build from a pre-trained model rather than a model with no picture information. The size of the input image at this layer increases the amount of training and testing data that must be stored in selection memory. As a result, 224\*224\*1 is picked as the input layer for all parameters. The incoming image should have the effect of bringing the negative values to zero. The activation function Relu was performed to predict plant disease. Because Relu layer requires less computing work than the other functions, it is recommended. In Eq. (14), Relu's activation function is as follows:

$$F(x) = \psi \{0, x < 0, x > 0 \quad (14)$$

The translation layers are one of the main blocks of the CNN algorithm. Input image has been filtering support for convolution layers. They chose to filter all the data and map image data's. All have an NxN size. Eq. (15) provides the curve for this layer, which is made up of several linear filters.

$$(h)_{ij} = (W_k + x)_{ij} + b_{ij} \quad (15)$$

Here  $i, j$  is pixel of the index and  $W, b$  is weighted parameters,  $x$  is involvement parameter of feature. The network performs better after being normalized. Other layers' data dimensions could be different. It's beneficial to normalize the data dimensions of other levels. Equations (16) and (17) normalizing function reads as follows:

$$X^k = \frac{x^k - E(x^k)}{\sqrt{\text{var}(x^k + \epsilon)}} \quad (16)$$

Here  $X^k$  is the input parameter, and  $E(x^k)$  is the dimension  $\sqrt{\text{var}(x^k + \epsilon)}$  is standard deviation.  $\gamma$  and  $\beta$  is the learning variable

$$y^k = \gamma^k x^k + \beta^k \quad (17)$$

Training networks with deep learning relies on a vast amount of data, which can lead to memory events while in use. It is advised not to take the internet lightly, as turning off certain nodes may be necessary to avoid memory events. By turning off a specific node, memory events are blocked, and network performance can be improved. As its name suggests, the fully connected layer relies on every field in the layer above it and converts the previous layer's data into a one-dimensional array structure. The number of fully coupled layers differs between systems, with two general strategies being utilized: average pooling and max pooling. Filters are chosen at the next level at the pool layer. The resulting image size is computed using Eq. (18 to 19) as a result of the binning:

$$S = W2 * h2 \quad (18)$$

$$W2 = \frac{(W1-f)}{A+1} \& h2 = \frac{h1-f}{A+1} \quad (19)$$

Here  $w1$  = size of width for image,  $h1$  = size of height for image,  $f$  = dimensional filtering,  $A$  = No of steps,  $S$  = input size of data.

Here  $x$  is no of class,  $W$  is weight,  $b$  is vector, the Cross-entropy is given by the equation (20). Let's assume  $C_e$  –cross entropy.

$$C_e = -\sum_x p^1(x) \log P(X) \quad (20)$$

$p'$  is expected output,  $p$  is an actual output for classification. The binary classification is done by the classification to categorize the result.

## 4 Result and Discussion

Jupyter Notebook was used as the Python programming language's Integrated Development Environment (IDE) for the experiments carried out in this research study. The task was completed using an i5 processor and 8GB of RAM on a Windows 10 computer. Plant leaf photos were taken to verify the suggested research.

**Table 1: Values and simulation parameters for plant leaf detection**



Parameter	Values
Name of the Dataset	New Plant Diseases Dataset
No. of images	6000
Language used	Python
Used Tool	Anaconda
Testing	80%
Training	20%

Table 1 shows the simulation parameter settings for detecting foliar diseases in plants. This was done using the ResNet50-optimized CNN and the New Plant Disease Dataset (NPDD), an open dataset available on Kaggle official website. The dataset includes 6000 JPG images of leaves, both diseased and healthy.

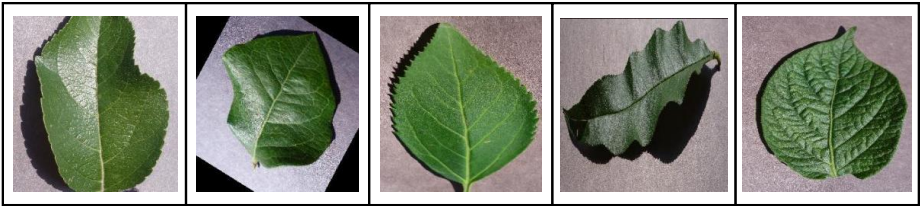


Figure 2: Sample images of healthy leaf images from dataset



Figure 3: Sample images of apple scab leaves



**Figure 4: Sample images in the dataset for grape black rot**

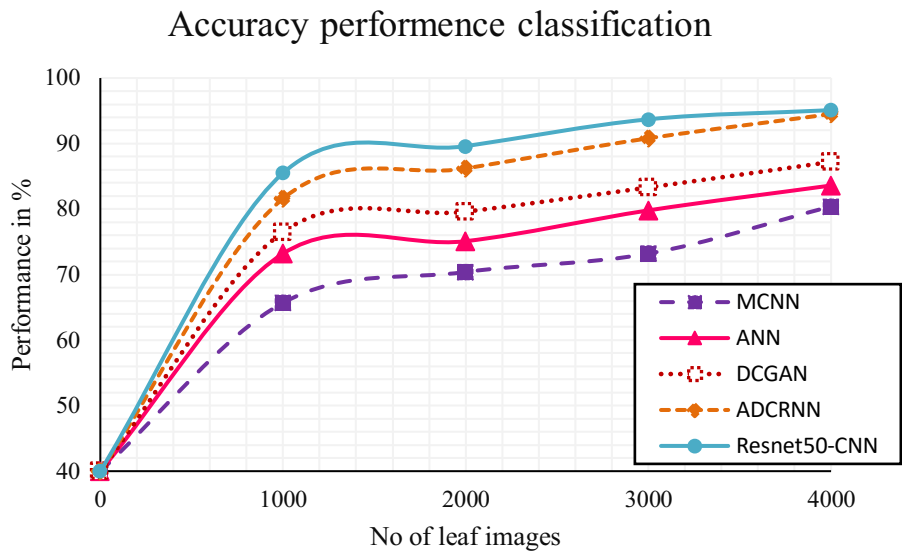


**Figure 5: Dataset sample images of potato early blight**

A newly collected plant disease dataset consists of 3,500 images of healthy leaves and 2,500 images of diseased leaves. Figure 2-5 shows an example of an image classified from a dataset. Leaf images were divided into training set and test set in the ratio of 80:20. This section analyzes the recommendations and results of existing plant disease detection methods and compares algorithms such as MCNN, ANN, and DCGAN.

**Table 2: Classification of accuracy performance of plant disease classification accuracy**

Accuracy performance of proposed method in %					
No of leaf images	MCNN	ANN	DCGAN	ADCRNN	Resnet50-CNN
1000	65.7	73.2	76.5	81.6	85.5
2000	70.4	75.1	79.6	86.2	89.6
3000	73.2	79.8	83.3	90.8	93.7
4000	80.4	83.6	87.2	94.5	95.1

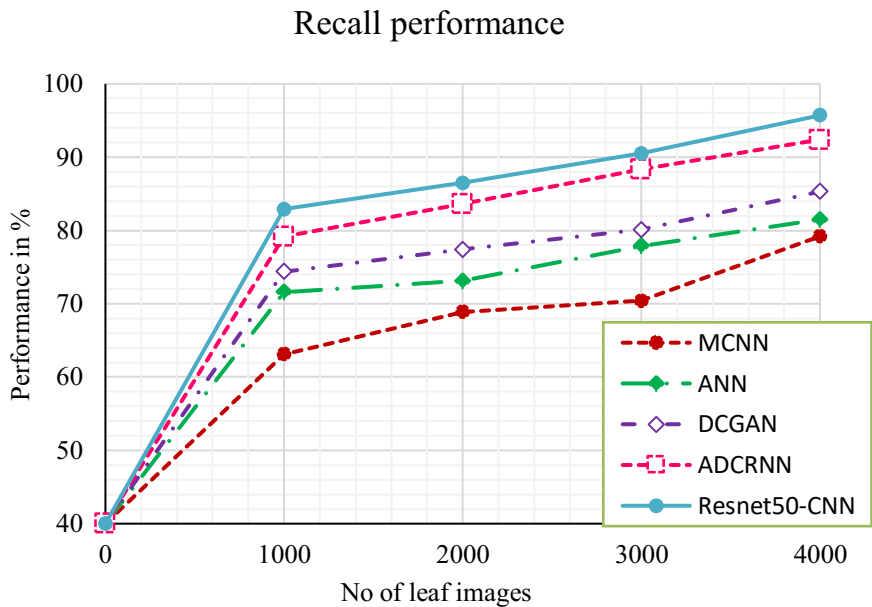


**Figure 6: Result of plant disease classification accuracy performance**

Based on the analysis presented in Figure 6 and Table 2, different approaches were used to identify plant foliar diseases. The proposed algorithm segmented the leaf shape and identified the features that effectively classified the disorders. The outcome visible that the deployed Res-net50 method achieved a classification performance of 95.1% in 4000 images, outperforming the MCNN, ANN, and DCGAN methods.

**Table 3: Outcome of the recall performance used method**

Recall performance in %					
No of leaf im- ages	MCNN	ANN	DCGAN	ADCRNN	Resnet50- CNN
1000	63.1	71.6	74.4	79.2	82.9
2000	68.9	73.1	77.4	83.6	86.5
3000	70.4	77.9	80.1	88.3	90.5
4000	79.2	81.5	85.3	92.4	95.7



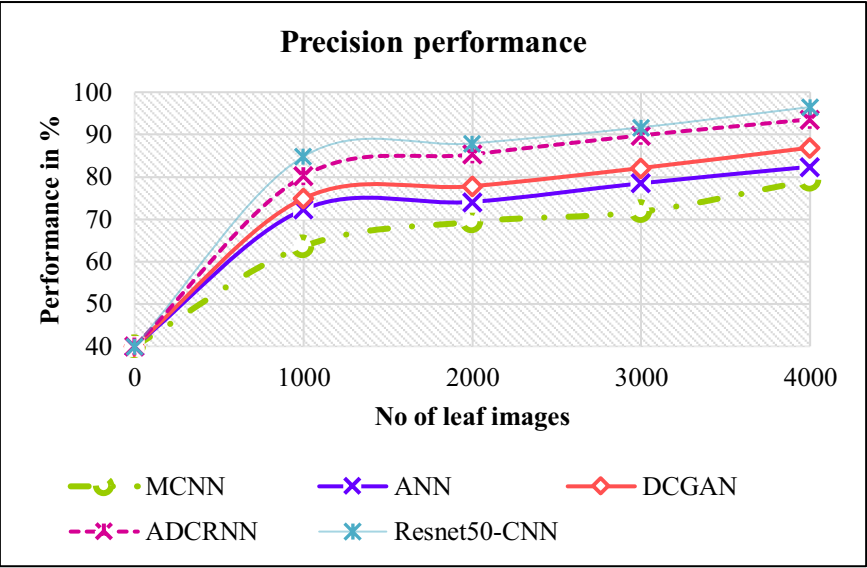
**Figure 7: Outcome of recall performance for the plant disease prediction**

For 4000 leaf images, the deployed methodology had an mean value recall of 95.7%. Figure 7 indicates that it performs better than other techniques. Table 3 displays the comparative outcome of the recall value for MCNN, ANN, and DCGAN methods.

**Table 4: Precision performance outcome of plant disease**

Precision performance in %					
No of leaf im- ages	MCNN	ANN	DCGAN	ADCRNN	Resnet50- CNN
1000	63.6	72.3	74.9	80.2	84.8
2000	69.4	74.1	77.8	85.3	87.9
3000	71.8	78.5	82.1	89.8	91.7

4000	79.2	82.4	86.9	93.6	96.5
------	------	------	------	------	------



**Figure 8: Precision performance outcome of plant disease**

After testing on 4000 leaf images, the deployed method had a mean value of 96.5%, which is high, as illustrated in Figure 8. Table 4 presents that compared to MCNN, ANN and DCGAN methodologies, this method offers better accuracy performance.

**Table 5: Outcome of F-measure performance in plant disease**

F-measure performance in %					
No of leaf images	MCNN	ANN	DCGAN	ADCRNN	Resnet50-CNN
1000	64.8	73.1	75.4	80.3	85.9
2000	69.5	74.2	78.9	85.4	89.5
3000	72.4	78.9	82.8	89.6	95.8

4000	79.5	82.4	86.9	94.1	96.8
------	------	------	------	------	------

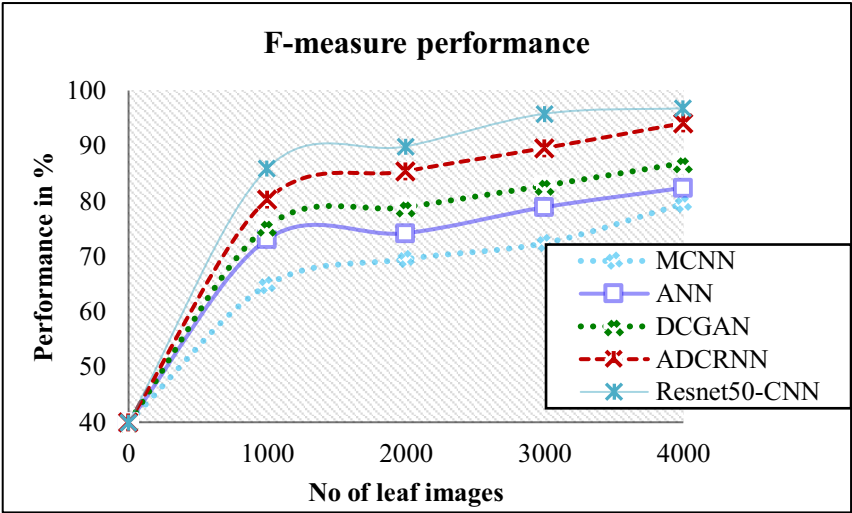
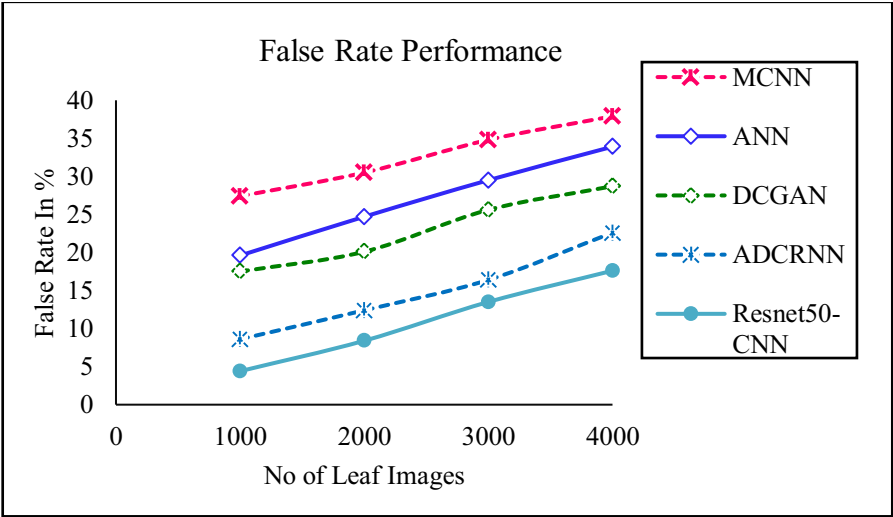


Figure 9: Outcome of F-measure performance in plant disease

Based on the data shown in Figure 9, the proposed method offers an average F-measure efficiency of 96.8% for 4,000 leaf images, outperforming the other techniques. Table 5 shows the comparison results of the F-measure performance of MCNN, ANN, and DCGAN methods.



### Figure 10: Result of plant disease false rate performance

The proposed method achieved an average false positive rate of 17.6% for 4000 leaf images, which is lower than other methods, as demonstrated in Figure 10.

## 5 Conclusion

In this paper, we used a ResNet50 methodology is classified and provides actionable outputs for plant leaf diseases based on binary classification. We collected a dataset of plant leaf diseases from the Kaggle. In addition, we use a preprocessing step to normalize the noise in the vegetation images using Gaussian and Wiener filters. In addition, the CRE method allows for selecting leaf features based on non-edge and smoothing. In addition, the MRTS method detects and predicts groups of pixels. Based on this, we can achieve stable classifier threshold separation across classes. Subsequently, techniques of plant disease classification were proposed. Therefore, the proposed classification performance is at 95.1% error rate efficiency. The simulation results show that the method can correctly classify plant diseases. Therefore, the proposed method is computationally more efficient than MCNN, ANN, and DCGAN methods in diagnosis and classification.

### Reference

1. Sunil S. Harakannanavar, Jayashri M. Rudagi, Veena I Puranikmath, Ayesha Siddiqua, R Pramodhini, Plant leaf disease detection using computer vision and machine learning algorithms, Global Transitions Proceedings, Volume 3, Issue 1, 2022, Pages 305-310, ISSN 2666-285X, <https://doi.org/10.1016/j.gltp.2022.03.016>.
2. J., A.; Eunice, J.; Popescu, D.E.; Chowdary, M.K.; Hemanth, J. Deep Learning-Based Leaf Disease Detection in Crops Using Images for Agricultural Applications. *Agronomy* 2022, 12, 2395. <https://doi.org/10.3390/agronomy12102395>
3. Liu, J., Wang, X. Plant diseases and pests detection based on deep learning: a review. *Plant Methods* 17, 22 (2021). <https://doi.org/10.1186/s13007-021-00722-9>
4. K. P. Asha Rani and S. Gowrishankar, "Pathogen-Based Classification of Plant Diseases: A Deep Transfer Learning Approach for Intelligent Support Systems," in *IEEE Access*, vol. 11, pp. 64476-64493, 2023, doi: 10.1109/ACCESS.2023.3284680.
5. A. Tabbakh and S. S. Barpanda, "A Deep Features Extraction Model Based on the Transfer Learning Model and Vision Transformer "TLMViT" for Plant Disease Classification," in *IEEE Access*, vol. 11, pp. 45377-45392, 2023, doi: 10.1109/ACCESS.2023.3273317.
6. Catherine, S., Rani, M. N., & Suresh, N. V. (2024). The Metaverse Economy: Transforming Money With Digital Currency. In *Creator's Economy in Metaverse Platforms: Empowering Stakeholders Through Omnichannel Approach* (pp. 202-209). IGI Global.

7. L. Li, S. Zhang and B. Wang, "Plant disease detection and classification by deep learning—A review", *IEEE Access*, vol. 9, pp. 56683-56698, 2021, [online] Available: <https://ieeexplore.ieee.org/abstract/document/9399342>.
8. A. Jain, S. Sarsaiya, Q. Wu, Y. Lu and J. Shi, "A review of plant leaf fungal diseases and its environment speciation", *Bioengineered*, vol. 10, no. 1, pp. 409-424, Jan. 2019.
9. O. O. Abayomi-Alli, R. Damaševičius, S. Misra and R. Maskeliūnas, "Cassava disease recognition from low-quality images using enhanced data augmentation model and deep learning", *Expert Syst.*, vol. 38, no. 7, pp. e12746, 2021, [online] Available: <https://onlinelibrary.wiley.com/doi/abs/10.1111/exsy.12746>.
10. F. Alvaro, Y. Sook, K. Sang and P. Dong, "A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition", *Sensors*, vol. 17, no. 9, pp. 2022, 2017, [online] Available: <https://www.mdpi.com/1424-8220/17/9/2022>.
11. B. Liu, C. Tan, S. Li, J. He and H. Wang, "A Data Augmentation Method Based on Generative Adversarial Networks for Grape Leaf Disease Identification," in *IEEE Access*, vol. 8, pp. 102188-102198, 2020, doi: 10.1109/ACCESS.2020.2998839.
12. A. Abbas, S. Jain, M. Gour and S. Vankudothu, "Tomato plant disease detection using transfer learning with C-GAN synthetic images", *Comput. Electron. Agricult.*, vol. 187, Aug. 2021, [online] Available: <https://www.sciencedirect.com/science/article/abs/pii/S0168169921002969>.
13. K. S. Patle, R. Saini, A. Kumar and V. S. Palaparth, "Field evaluation of smart sensor system for plant disease prediction using LSTM network", *IEEE Sensors J.*, vol. 22, no. 4, pp. 3715-3725, Feb. 2022.
14. V. Singh, N. Sharma and S. Singh, "A review of imaging techniques for plant disease detection", *Artif. Intell. Agricult.*, vol. 4, pp. 229-242, Oct. 2020.
15. Suresh, N. V., & Remy, V. A. M. (2024, February). An Empirical Study on Empowering Women through Self Help Groups. In 3rd International Conference on Reinventing Business Practices, Start-ups and Sustainability (ICRBSS 2023) (pp. 957-964). Atlantis Press.
16. Suganya, V., & Suresh, N. V. (2024). Potential Mental and Physical Health Impacts of Spending Extended Periods in the Metaverse: An Analysis. In *Creator's Economy in Metaverse Platforms: Empowering Stakeholders Through Omnichannel Approach* (pp. 225-232). IGI Global.
17. L. C. Ngugi, M. Abelwahab and M. Abo-Zahhad, "Recent advances in image processing techniques for automated leaf pest and disease recognition—A review", *Inf. Process. Agricult.*, vol. 180, pp. 26-50, Apr. 2020.
18. H. Nazki, S. Yoon, A. Fuentes and D. S. Park, "Unsupervised image translation using adversarial networks for improved plant disease recognition", *Comput. Electron. Agricult.*, vol. 168, Jan. 2020.
19. J. Chen, J. Chen, D. Zhang, Y. Sun and Y. A. Nanekaran, "Using deep transfer learning for image-based plant disease identification", *Comput. Electron. Agricult.*, vol. 173, Jun. 2020.
20. M. Kumar, A. Kumar and V. S. Palaparth, "Soil Sensors-Based Prediction System for Plant Diseases Using Exploratory Data Analysis and Machine Learning," in *IEEE Sensors Journal*, vol. 21, no. 16, pp. 17455-17468, 15 Aug.15, 2021, doi: 10.1109/JSEN.2020.3046295.
21. R. Saini, K. S. Patle, A. Kumar, S. G. Surya and V. S. Palaparth, "Attention-Based Multi-Input Multi-Output Neural Network for Plant Disease Prediction Using Multisensor System," in *IEEE Sensors Journal*, vol. 22, no. 24, pp. 24242-24252, 15 Dec.15, 2022, doi: 10.1109/JSEN.2022.3219601.
22. M. Türkoglu and D. Hanbay, "Plant disease and pest detection using deep learning-based features", *TURKISH J. Electr. Eng. Comput. Sci.*, vol. 27, no. 3, pp. 1636-1651, May 2019.



23. S. H. Lee, H. Goëau, P. Bonnet and A. Joly, "New perspectives on plant disease characterization based on deep learning", *Comput. Electron. Agricult.*, vol. 170, Mar. 2020.

**Open Access** This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

