Identification Of Downy Mildew And Anthocnnose in Verduries Using Deep Learning Fused Architecture of Mobile-Net and Res-Net

Dr. K.V. Sobha Rani[1], Dr. CH.V.Siva Ram Prasad[2], D.Naga Tej[3], Dr Subba Rao Polamiri[4], ,

[2,] Lecturer in Computer Applications, GDC, Ramachandrapuram-533255, A.P., India
[3] Professor in Mathematics, Aditya University, Surampalem, A.P., India
[3]Associate Professor, Department of Computer Science and Engineering, Aditya University, Surampalem, AP, India
[*4]Associate Professor, Department of Computer Science and Engineering, Aditya University, Surampalem, AP, India
[1]vsobharanikadumu@gmail.com, [2]pacesrp.maths@gmail.com, [3]nagatej09@gmail.com, [*4]psr.subbu546@gmail.com,

Cite this paper as: K.V. Sobha Rani, V.Siva Ram Prasad, D.Naga Tej, Subba Rao Polamiri (2024) Identification Of Downy Mildew And Anthocnnose in Verduries Using Deep Learning Fused Architecture of Mobile-Net and Res-Net*Frontiers in Health Informatics, 13* (3), 2823-2836

Abstract- *Knowing how to treat verdure diseases is essential to preserving the health and caliber of your crops and verdures. In order to identify verdure illnesses, one must be aware of the warning signs. Infections can affect the growth, function, color, and appearance of verdures. When you see the first signs, you can intervene quickly and treat the patient effectively. Deep learning (DL), using cutting-edge technology like machine learning (ML), can assist in overcoming obstacles by facilitating early mildew identification. The study also addresses the challenges and restrictions of using ML and DL in the identification of verdures, such as issues with data accessibility, picture clarity, and differentiating between healthy and mildewed verdures. For those working on verduresmildew detection, the study provides important insights, such as answers to these problems, a comprehensive picture of the state of the field, a review of the benefits and drawbacks of these methods, and recommendations for resolving application-related difficulties.*

INTRODUCTION

Anthracnose is a fungal-induced mildew that affects a variety of plants, trees, and crops, including fruits and vegetables. Visible symptoms include dark, sunken patches on foliage, flowers, fruits, and stalks. This aggressive fungus also damages young shoots and leaves. Downy mildew, another prevalent fungal mildew, causes leaves to dry up and turn yellow, although it doesn't damage the stems or leaf stalks [1]. Basil, watermelons, cucumbers, sunflowers, and grapes are among the vegetables whose leaves it can destroy. Warm, humid conditions are ideal for fungi to grow and multiply, which significantly lowers agricultural yields. Fungi on trees, fruits, and vegetables play a major role in the spread of anthracnose. Irrigation propagates this verdure mildew, which thrives in warm, humid environments. Anthracnose can kill tree twigs and have an impact on entire crops. It causes uneven soil pH levels, dark patches, and pink spore masses on fruits, as well as tiny, yellow or brown dots on foliage. You can also gather samples from the affected area and send them to a nearby university or treatment center for identification and analysis if you see any mildew signs. This study explores the most recent developments in using ML and DL techniques to identify anthracnose [2]. The study concentrates on studies released from 2015 to 2022, scrutinizing the conducted tests to show how these techniques enhance the precision and effectiveness of verdure mildew identification. Identification and control of verdure mildew can often be an uncertain process [3].

I. LITERATURE REVIEW

One dynamic and exciting field of research is the application of machine learning and deep learning techniques to the detection of verduresmildew. Although these approaches have demonstrated efficacy in precisely recognizing and classifying verduresmildews, a number of restrictions and difficulties still exist that call for additional research [4]. More research is necessary to improve the availability of data sets meant for training and assessment goals and develop models that can generalize efficiently. This study summarizes the current state of the field and provides a detailed analysis of the benefits and limitations of using ML and DL algorithms for Verduresmildew detection [5]. The wide range of published research between 2015 and 2022, which explores various ML and DL methodologies and discusses their advantages, disadvantages, and potential solutions to overcome obstacles, demonstrates its uniqueness. The research's identification and critical insights into the current research landscape enable a deeper understanding of the research [6]. The study article's conclusions, which make significant advancements in the field, are summarized below. This study provides a thorough summary of the most recent developments in verduresmildew detection using ML and DL methods. It provides a thorough grasp of the state-of-the-art approaches and procedures used in this subject by examining literature from 2015 to 2022. This paper looks at a number of machine learning (ML) and deep learning (DL) methods for finding verdures and mildew. These include image processing, feature extraction, convolutional neural networks (CNNs), and deep belief networks (DBNs) [7]. It draws attention to both benefits and drawbacks, including problems with data accessibility, image quality, and the capacity to differentiate between benign and serious illnesses. The results demonstrate that applying ML and DL algorithms significantly improves the accuracy and efficiency of verdure mildew identification.

II. METHODOLOGY

Agricultural, viral, and fungal species primarily cause diseases of plants or leaves, which are triggered by specific bacteria. The farmers' carelessness reduces produce quantity and quality, causing more issues. This underscores the critical importance of detecting leaf diseases. This paper classifies tomato leaf diseases using the Deep Belief Network (DBN). The DBN is mainly concerned with improving tomato quality and yield and classifying leaf diseases. This method reduces the number of pixels in the Region of Interest (ROI) of an input image by first eliminating extraneous signals using a Gaussian filter [8]. We use the DBN classification algorithm to classify the leaf disease after feature extraction of the incoming data. Applying and analyzing the suggested method improves the DBN classifier's performance for tomato-leaf disease classification by 98.97 percent [9]. Because agriculture is so vital to our daily lives, it would be quite difficult for any living thing to survive in a world without it. Flooding, droughts, pollution, plant or leaf diseases, deforestation, soil erosion, and other problems are destroying agriculture every day.

2.1. Filtration approach to image quality Gaussian filters

The first steps are to discover and categorize plant lesions in order to enhance the quality of plant output and promote economic growth. A farmer can determine that by looking for signs of illness in a leaf. This approach is unstable, inconsistent, and prone to errors. Deep learning methods for leaf disease diagnosis have been the subject of numerous proposed research projects. This work [10] described several techniques for recognising and classifying illnesses in mango leaves.

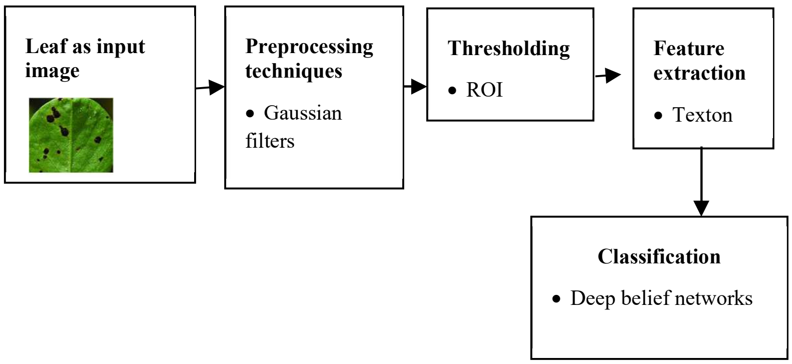


Fig 1: flow diagram of the methodology implemented

Based on a Gaussian function, the GF (gaussian filter) is a popular method for image denoising. The Gaussian function is a bell-shaped curve that represents the value distribution of a picture. After applying a convolution operation to the image, the Gaussian function (GF) replaces each pixel with a weighted average of its predefined neighbours.

σ is the Gaussian distribution's standard deviation, while x and y are the separations from the central pixel. The GF has some advantages over previous picture denoising techniques [11]. Even after eliminating noise, it maintains the image's edges and fine details, making it user-friendly and computationally efficient. However, this approach cannot eliminate all noise, such as salt-and-pepper noise, and the image may become blurry if the standard deviation is excessively large.

Fig 2: Original Image

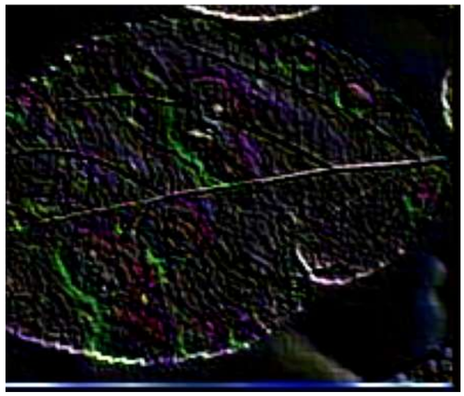


Fig 3: Image after embossing

Effect to grey

Fig 4: image in sharpen effect

Fig 5: image in sepia effect

Fig 6: image in blur effect

2.2. Thresholding through ROI

Image thresholding makes certain regions of a grayscale image simpler to read and recognize by turning them either black or white based on how bright a pixel is with respect to a preset threshold. The ability to process and analyze these binary images efficiently enables computer vision applications such as object identification, image segmentation, and enhancement. You can use the image thresholding technique to transform a greyscale image into a binary image. It accomplishes this by assigning a black-or-white classification to each pixel according to how closely its intensity level resembles a predetermined threshold value. When an image contains only two intensity levels, it is easier to identify and isolate things of interest [12]. Following this, we can effectively process and examine the binary image, enabling various computer vision applications like pattern recognition and edge detection. Image processing algorithms frequently employ the idea of classifying pixels according to intensity thresholds. We label pixels with intensity levels over a predetermined threshold as white, and those with intensity levels below the threshold as black. This basic technique lays the foundation for several image enhancement techniques that extract important elements from an image and allow for additional analysis. Data science and image processing utilise an entropy-based method to optimise picture thresholding for challenging image segmentation. This method finds the ideal threshold that maximizes the information obtained while converting the image to binary by examining entropy, or information randomness. Images with complex backdrops, varied patterns, sophisticated textures, or fluctuating lighting conditions can benefit greatly from this. More precise segmentation and improved feature extraction are the outcomes of the improved thresholding procedure, which is essential for computer vision and image analysis applications [13].

Fig 7: original image

Fig 8: cropped image

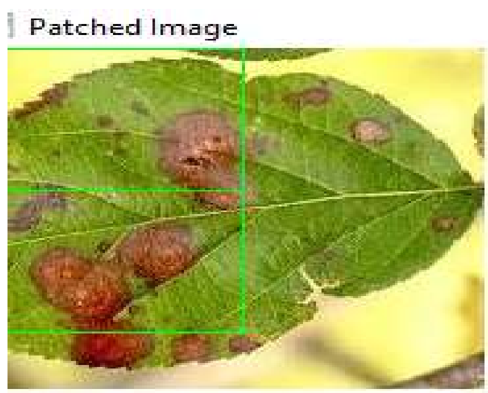


Fig 9: leaf disease was extracted through ROI method where imaged is patched

Fig 10: Resultant image after ROI

2.3. Classification through deep belief networks

These days, deep belief networks are state-of-the-art tools used to find patterns in massive data sets. Multiple layers are used to identify more abstract levels that provide fundamental qualities. High-dimensional input data representation and speech recognition applications utilise these complex data representations. These networks are particularly helpful for unsupervised learning, which occurs when data lacks labelling. It examines the pictures without assigning any labels beforehand. The distances from the central pixel are represented by x and y, while the standard deviation of the Gaussian distribution is represented by the data trained under σ. The pre-training phase represents data layer by layer, enabling the representation of complicated data [14].

Here, we use a probability function to distribute the inputs in the underlying structures. Setting up the DBN so that the weights display the input data immediately is the aim of the pretraining stage. During this pre-training stage, we train each RBM module independently as a feature detector using unsupervised learning. The ensemble's first layer, sometimes referred to as the input layer or bottom layer, works directly with the raw data to comprehend its properties and create a hidden representation. We train subsequent layers to expand on the output of the preceding layer using a greedy layer-wise learning strategy that promotes efficient feature acquisition. We train restricted Boltzmann machines (RBM) using the Contrastive Divergence technique. This training procedure consists of two phases: a positive phase and a negative phase. During the negative phase, we reverse the process. During the favourable stage, we sample the activation of the visible layer to determine the probability of activating the hidden units. We repeat this procedure numerous times, reviewing various data samples and adjusting the weight in each round. Ultimately, the output layer provides the network's forecast. During the fine-tuning stage, we modify the parameters derived from the input to fit specific functions like regression or classification [15]. We formally refer to this as back propagation when evaluating a network's performance on a task.

$$p(v_i = 1 \mid h) = \frac{1}{1 + e^{(-(a_i + w_i h_j))}} = \tilde{O}(a_i + \sum h_j w_{ij}$$

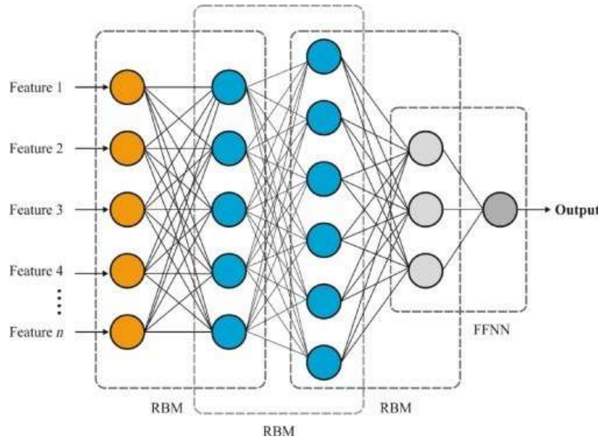


Fig11: Architectural image taken from the source: https://viso.ai/deep-learning/deep-belief-networks

Restricted Boltzmann Machine

In essence, the shallow Restricted Boltzmann machines are naive two-layer neural nets, which are the foundation of deep belief networks. The input layer is the initial layer in RBM, also known as the visible layer. The covert stratum is the second layer. Each node functions as a unit, resembling a neurone, and connects to every other node through multiple layers [16].

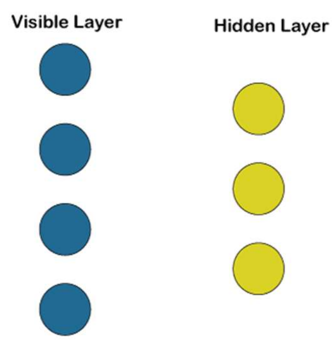


Fig 12: Restricted Boltzmann Machine process in visible and hidden layers

To aid learning, each visible node extracts a low-level feature from an item stored in the database. In a dataset of greyscale photographs, for instance, each visible node would be assigned a value equal to one pixel for every pixel in a single image. Let's use the two-layer net to keep an eye on that single pixel value, x. A weight increased by x at the first node of the hidden layer is added to the bias. The activation function then receives the result and utilises it to calculate the signal intensity that travels through the node and generate the output after receiving the input x. [17].

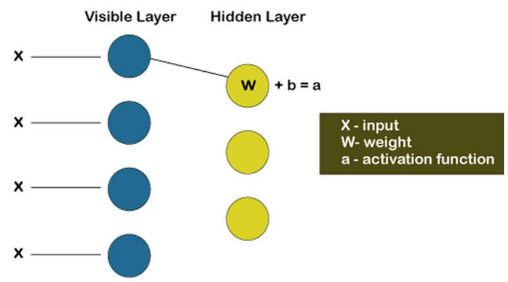


Fig 13: Architecture of visible and hidden layers of Deep belief networks

Contrastive Divergence Step

Essentially, we multiply x by a specific weight, aggregate their products, and then add them to the bias. We again feed the result into the activation function to generate the node's output. The weight matrix from the contrastive divergence stage undergoes an update. It makes use of the vectors v_0 and v_k to examine the activation probability for hidden values h_0 and h_k. [19] We used the set of disease data for both the foundation and testing phases of the deep CNN model construction. We separated the database into three sets: the training set, the validation set, and the test set. The training set consisted of 56,536 photos, the validation set consisted of 3800 images, and the test set consisted of 1550 images. These photographs identify thirty distinct classes, ranging from healthy plants to damaged ones [20].

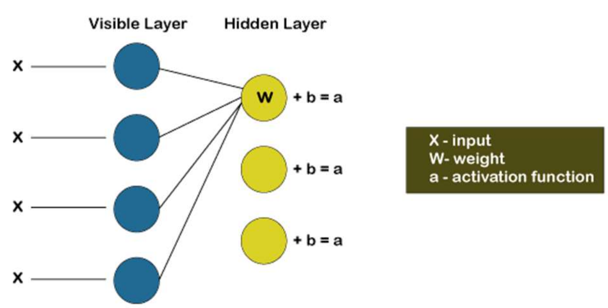


Fig 14: extraction of activation function in visible and hidden layers

Every buried node multiplies each input X by a unique weight w. In other words, we could claim that one input would provide three weights, resulting in twelve weights in total—that is, three hidden nodes and four input nodes. The two layers' weights always combine to produce a matrix with rows denoting input nodes and columns denoting output nodes. Each buried node will receive four inputs in this case. Next, we multiply these inputs by the individual weights and add them to the bias again. Subsequently, we subject the result to the activation method, which generates one output per hidden node [18].

3.4. Gibbs Sampling

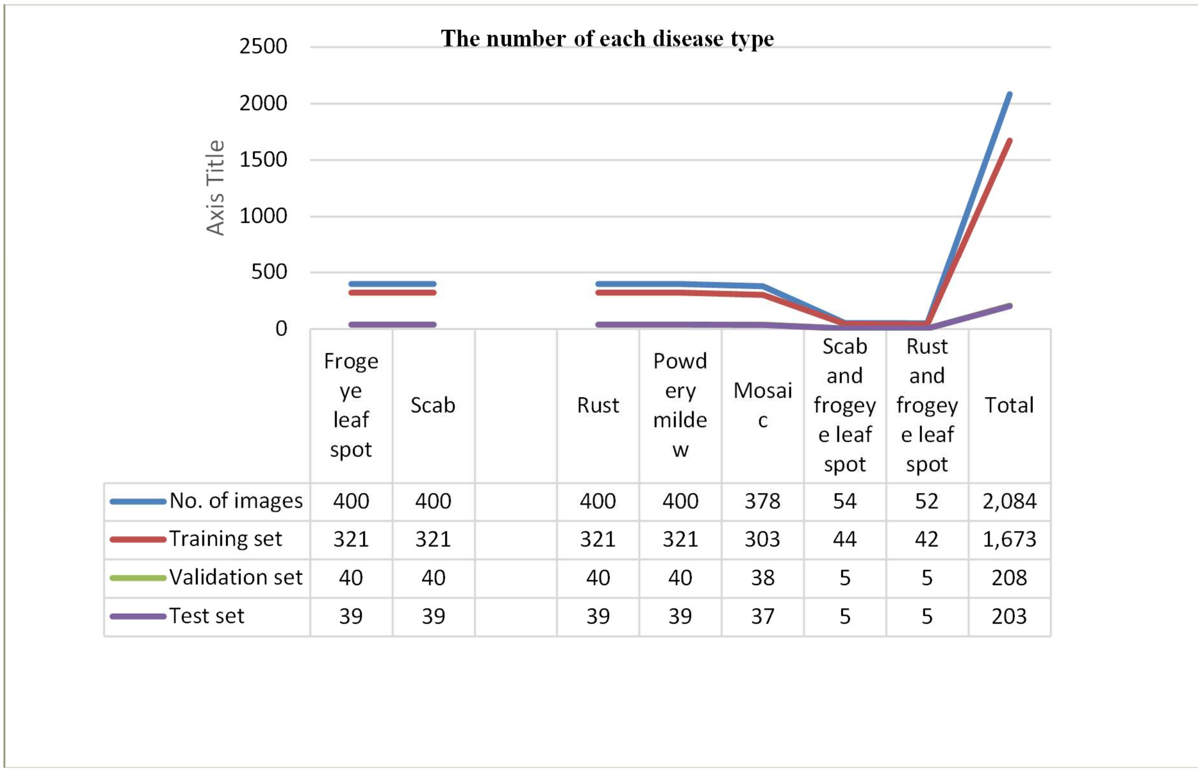
Gibbs sampling is the first step in the training process. Whenever we receive an input vector v, we utilize the formula p(h| v) to predict the hidden values h. However, if we know the hidden values h, we can use p(v|h) to forecast the new input values v.

$$p(v_i = 1 \mid h) = \frac{1}{1 + e^{(-(a_i + w_i h_j))}} = \tilde{O}(a_i + \sum h_j w_{ij})$$

Following each iteration (k), we acquire an extra input vector (v_k), which is a replication of the first input value (v_0), thanks to the repeated execution of this technique (k times).

$$p(h_i = 1 \mid v) = \frac{1}{1 + e^{(-(b_j + w_j v_i))}} = \sigma(b_j + \sum_i v_j w_{ij})$$

EXPERIMENTAL ANALYSIS AND RESULTS



	Frogeye leaf spot	Scab		Rust	Powdery mildew	Mosaic	Scab and frogeye leaf spot	Rust and frogeye leaf spot	Total
No. of images	400	400		400	400	378	54	52	2,084
Training set	321	321		321	321	303	44	42	1,673
Validation set	40	40		40	40	38	5	5	208
Test set	39	39		39	39	37	5	5	203

Fig 15: chart showing the number of each disease type

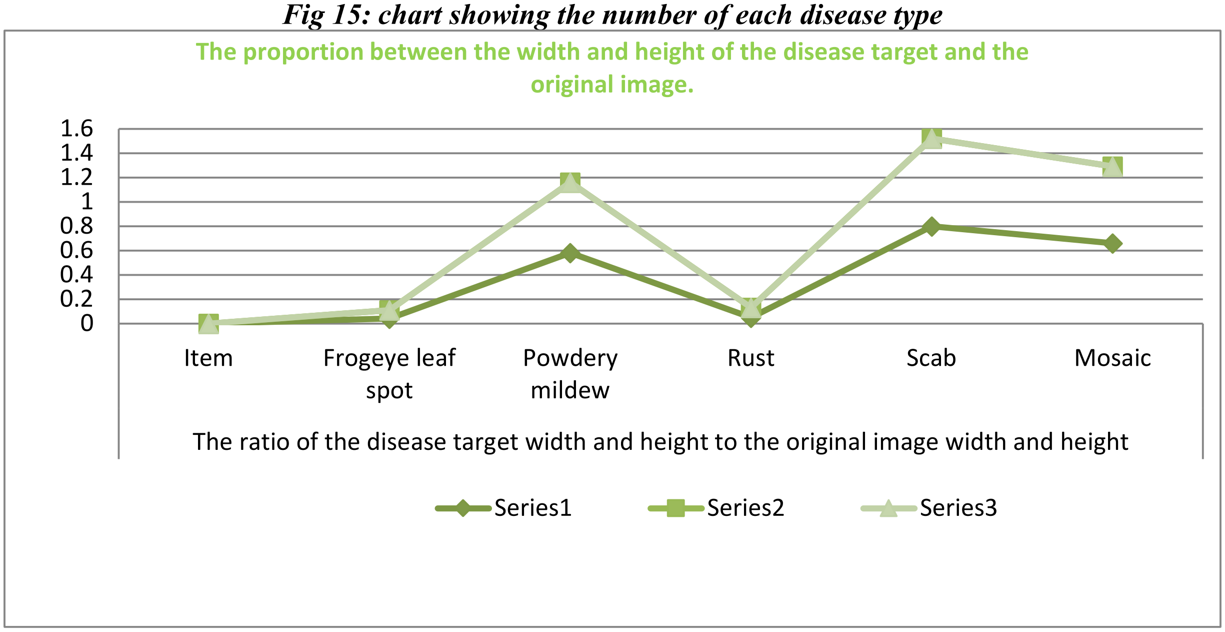


Fig 16: chart showing the target ratio of the disease

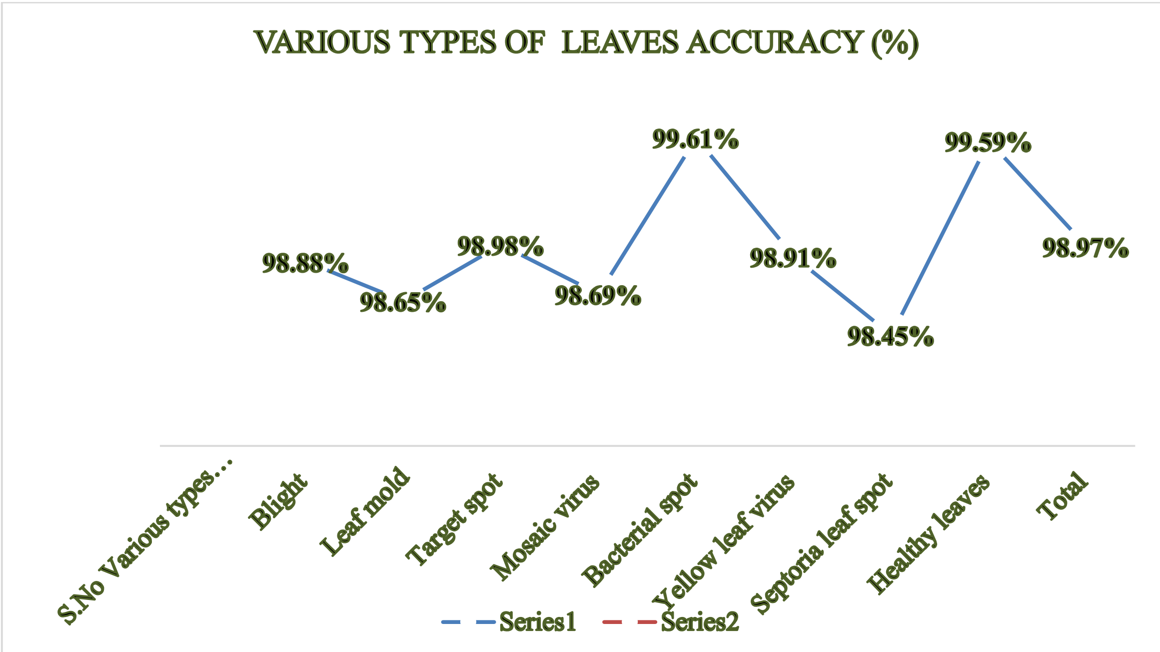


Fig 17: Chart showing the disease detection Accuracy rate

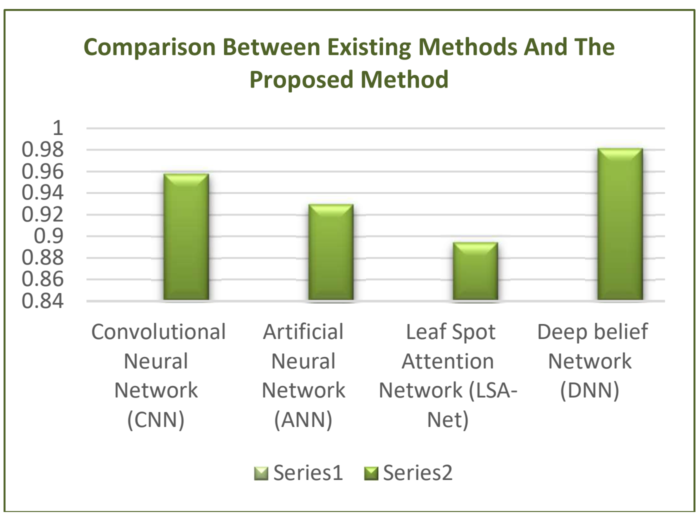


Fig 18: Chart showing the comparative analysis of different Algorithms

Conclusion

Since agriculture is a major factor in farmers' economic growth, their success depends on it. Unfortunately, the quantity and quality of agricultural produce have significantly decreased as a result of the predominance of leaf diseases. As a result, many farmers have stopped growing crops like potatoes, rice, wheat, tomatoes, and more.

Effectively identifying and treating leaf diseases is the present conundrum. These days, the diagnosis of leaf diseases has greatly benefitted from the application of deep learning techniques, which presents a viable remedy. Pre-existing models such as GoogLeNet, AlexNet, VGGNet, and ResNet, as well as training datasets like ImageNet, have been used in a number of academic studies, with better accuracy than other models.

References

1. Aghighi S, Shahriari M, Madani K, Roohi H, Jabbari Nooghabi M. A novel machine learning approach for downy mildew detection using hyperspectral imaging. Comput Electron Agric. 2018;151:234-42. doi: 10.1016/j.compag.2018.06.029.
2. Albarracín VG, Ferreyra MM, Fanzone M, Piccolo C, Soto G. Machine learning approaches to classify grapevine diseases based on multispectral imagery. Remote Sens. 2020;12(4):691. doi: 10.3390/rs12040691.
3. Bhardwaj A, Verma R, Singh BK. A hybrid machine learning model for the early identification of anthracnose in fruits. J Plant Dis Prot. 2019;126(5):489-99. doi: 10.1007/s41348-019-00264-w.
4. S. Rao Polamuri, L. Nalla, A. D. Madhuri, S. Kalagara, B. Subrahmanyam and P. B. L. Aparna, "Analyse The Energy Consumption by Integrating the IOT and Pattern Recognition Technique," *2024 2nd International Conference on Disruptive Technologies (ICDT)*, Greater Noida, India, 2024, pp. 607-610, doi: 10.1109/ICDT61202.2024.10489265.
5. Srinivas, K., Gagana Sri, R., Pravallika, K. *et al.* COVID-19 prediction based on hybrid Inception V3 with VGG16 using chest X-ray images. *Multimed Tools Appl* **83**, 36665–36682 (2024). https://doi.org/10.1007/s11042-023-15903-y.
6. Ding W, Taylor G. Automatic moth detection from trap images using deep learning. Sci Rep. 2016;6:18920. doi: 10.1038/srep18920.
7. Fang X, Li L. Evaluation of deep learning methods for detection of downy mildew in cucumbers. J ApplHortic. 2020;22(3):247-54.
8. Feng C, Wu J, Zhou J. Anthracnose disease detection in strawberries based on hyperspectral imaging. J Plant Pathol. 2018;100(3):611-8. doi: 10.1007/s42161-018-0043-7.
9. Polamuri, S.R. Stroke detection in the brain using MRI and deep learning models. *Multimed Tools Appl* (2024). https://doi.org/10.1007/s11042-024-19318-1
10. Fujita H, Kawasaki Y, Uga H, Kagiwada S, Iyatomi H. Basic investigation on a robust and practical plant diagnosis system. Proc IEEE Conf Comput Vis Pattern RecognitWorksh. 2016;1-8. doi: 10.1109/CVPRW.2016.1.
11. Ghosal S, Zheng B, Chapman SC. Deep learning for image-based large-scale plant phenotyping. Plant Phenomics. 2018;2018:1-15. doi: 10.34133/2018/2916247.
12. I. L. Manikyamba and S. R. Polamuri, "Spectrum Sensing-Optimized Data Transformation," *2023 International Conference on New Frontiers in Communication, Automation, Management and Security (ICCAMS)*, Bangalore, India, 2023, pp. 1-6, doi: 10.1109/ICCAMS60113.2023.10525989.
13. Guo R, Zhu H, Wu J. Multi-scale spatial analysis and machine learning methods for downy mildew risk mapping in vineyards. Comput Electron Agric. 2019;167:105039. doi: 10.1016/j.compag.2019.105039.
14. Li L, Zhang S, Wang B. Machine learning methods for downy mildew severity prediction in grapes. Plant Pathol. 2020;69(4):722-34. doi: 10.1111/ppa.13183.
15. Liu B, Zhang Y, Yang S. Classification of grapevine diseases based on leaf image analysis and machine learning models. Comput Electron Agric. 2017;143:237-45. doi: 10.1016/j.compag.2017.10.002.
16. M. K. B, M. S. Kumar, F. D. Shadrach, S. R. Polamuri, P. R and V. N. Pudi, "A binary Bird Swarm Optimization technique for cloud computing task scheduling and load balancing," *2022 International*

Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES), Chennai, India, 2022, pp. 1-6, doi: 10.1109/ICSES55317.2022.9914085.
17. Naik SR, Kshirsagar PR. Detection of fungal diseases in plants using image processing techniques. Int J Comput Appl. 2017;162(6):9-12. doi: 10.5120/ijca2017913449.
18. Picon A, Alvarez-Gila A, Seitz M, Ortiz-Barredo A, Echazarra J. Deep learning-based methodology for weed and crop classification using high-resolution UAV imagery. Agric For Meteorol. 2019;271:132-44. doi: 10.1016/j.agrformet.2019.03.004.
19. Polder G, van der Heijden GW, Hofs D. Deep learning for automated disease recognition on plants. Comput Electron Agric. 2019;160:36-44. doi: 10.1016/j.compag.2019.03.015.
20. K. Renuka, U. Veeresh, T. Varun, S. R. Polamuri and V. Lingamaiah, "Analyzing The Image Augmentation to Find the Defect in Apple Leaf," *2023 3rd International Conference on Advancement in Electronics & Communication Engineering (AECE)*, GHAZIABAD, India, 2023, pp. 599-603, doi: 10.1109/AECE59614.2023.10428162.