Accurate Diagnosis of Plant Leaf Diseases Using Transfer Learning and CNN-Dense Classification Fusion

Swapna Jamal (0009-0007-7942-0508)[1*], J. E Judith (0000-0001-9185-4412)[2]
[1*]Department of computer Applications
Noorul Islam Centre for Higher Education, Thackalai
Kumaracoil, Tamil Nadu 629180
[1*]swapnashan@gmail.com
[2]Department of Computer science Engineering
Noorul Islam Centre for Higher Education, Thackalai
Kumaracoil, Tamil Nadu 629180
[2]judith@niuniv.com

Abstract: The main goal of this study are to develop a highly accurate and computationally efficient model for the early detection of plant leaf diseases, ensuring timely intervention and effective crop management. The approach focuses on enhancing image quality through bilateral filtering for noise reduction, thereby improving clarity for precise feature extraction. It further aims to leverage transfer learning using ResNet-101, EfficientNet-B0, and MobileNet-V3 Large to capture detailed disease features. Additionally, the study seeks to optimize classification performance by integrating a CNN-Dense classification system with Bayesian optimization for hyperparameter tuning, thereby boosting model robustness and precision. Ultimately, the research aims to contribute to global food security by enabling reliable and early disease detection, which is essential as the world's population continues to grow.

Keywords: Plant Leaf, Filter, Convolutional neural network (CNN), Resnet, Efficient Net, Mobile Net, Bayesian Optimization.

1. INTRODUCTION

Agriculture in India is critical to economic growth, with farmers selecting crops based on soil type, local weather, and crop economic value [1]. Regardless of the circumstances, the industry attempts to boost productivity. Precision agriculture (PA) is a modern technique that increases agricultural output and can contribute to economic development [2]. PA can identify plant pests, weeds, and illnesses, lowering pest populations and increasing crop yields [3, 4]. Pesticides are used to control pests and illnesses, however, the predominance of crop diseases in industrial agriculture creates problems such as decreased yields, financial losses, and negative effects on production [5]. Early diagnosis of plant diseases is vital for agricultural production management and decision-making. Identifying them has become a critical issue in recent years [6].

Identifying crop diseases effectively is critical for timely treatment and minimizing further harm [7]. Precision agriculture and management are increasingly relying on artificial intelligence and machine learning to identify patterns and structural analogies in photographs [8]. Plant diseases are identified and classified using general image processing techniques such as color analysis and thresholding. Several standard and novel strategies for detecting plant diseases can be used with other image pre-processing methods to improve feature extraction [9]. The agricultural sector research uses machine learning (ML) to diagnose crop diseases using a variety of techniques, including fuzzy logic, SVM, sporadic forest, and K-means system [10]. It is essential for plant pressure testing and the identification, detection, and prediction of crop diseases [11]. Accurate strategy selection in plant disease (PD) requires space data. Plant diseases are detected by ML techniques, which also classify PD into various stages or types [12]. Fine development, an infectious organism that affects harvest varieties' yield, is subject to ML counts. However, the extent of their work is restricted to determining the crop's disease type [13].

DL models have been employed to obtain varying degrees of accuracy on lab photos, with the highest accuracy reached when exact classification models are verified on data identical to the training data [14]. Despite their high success rate, some limitations preclude them from being used as a universal tool for real-time plant leaf disease identification [15]. Previous research employed DL models to diagnose illnesses on the Plant Village dataset, which has low variability [16]. Recently, the efficiency of DL models during training on individual lesions and locations was examined. To expand the dataset size, image segmentation and augmentation were employed. As a result, dealing with plant leaf disease requires a comprehensive approach.

Problem statement

Key drawbacks in preprocessing for plant leaf disease detection include high computational demand, occurring due to complex operations requiring large parameter sets, making models unsuitable for resource-constrained applications. Another drawback is the inaccurate isolation of diseased regions, occurring when segmentation techniques lack sensitivity to subtle or overlapping disease patterns, causing misclassification. Additionally, cross-crop applicability and resolution limitations arise as some preprocessing methods rely on crop-specific texture patterns, making them less adaptable to diverse conditions. Lastly, dependency on large training datasets occurs when models require extensive data to learn hierarchical features, limiting practical use with smaller datasets.

Focus of the work

- Detection of plant leaf disease by introducing an advanced method that combines transfer learning and revolutionary image processing techniques.
- The improvement of image clarity through noise reduction through the use of a bilateral filter.
- Using modern algorithms like ResNet-101, EfficientNet-B0, and MobileNet-V3 Large, transfer learning is implemented to effectively extract rich, pertinent features from cleaned images.
- Integration of a CNN-Dense classification system that classifies different plant leaf diseases accurately by using feature fusion. Utilizing Bayesian optimization to optimize hyperparameters ensures high model accuracy and efficient use of computational resources.

- Together, these developments improve plant leaf disease detection's accuracy, dependability, and efficiency and offer a strong basis for early disease prediction.

The structure of this article's remaining portions follows Section 2, which specifies the numerous previous studies conducted in this field. Section 3 presents the current work's methodology and then goes on to explain the transfer techniques that are mentioned. The results and a thorough discussion are presented in section 4 and Section 5 concludes the paper.

2. RELATED WORKS

Identifying leaf diseases through image processing has been extensively researched and continues to be a significant area of interest for researchers. Various methods exist for detecting crop diseases based on their occurrence in different plant parts, such as the leaf, node, or stem. This paper focuses on identifying crop diseases by analyzing their effects on leaves.

Ashwin Kumar et.al [17] the article presents an automated methodology that uses an OMNCNN to identify and categorize plant leaf diseases. The model consists of four stages: pre-processing, segmentation, feature extraction, and classification. Bilateral filtering and Kapur's thresholding are employed to identify afflicted areas, and the emperor penguin optimizer algorithm enhances disease identification. However, the method's precision reduces detection efficiency.

Perveen et.al [18] A dual-branch apple leaf disease diagnostic system (DBNet) has been created to deal with lesion similarities and complex surroundings. The system includes a multiscale joint branch with an attention branch with several dimensions, which improves identification accuracy while reducing lesion similarities, but it can be computationally and resource-heavy.

Rahman et.al [19] article describes an image processing method for recognizing and managing leaf illnesses in tomato crops, which employs the Grey Level Co-Occurrence Matrix (GLCM) algorithm to generate 13 statistical features from tomato leaves and classify them into distinct diseases. However, the method has disadvantages such as low image resolution, detection failures caused by lighting conditions and acquisition angles, and restricted applicability due to disease manifestation variances.

Mahum et.al [20] A deep learning algorithm has been designed to divide potato leaves into five categories: healthy, verticillium-wilt, leaf roll, late blight, and early blight. The "Plant Village" dataset serves as the model's training source, with other classes obtained manually. Diseases are classified effectively using the Efficient Dense Net model, which employs a reweighted cross-entropy loss function and dense connections to reduce overfitting. This is the first technology that has successfully identified and classified four illnesses in potato leaves.

Peng et.al [21] A novel image retrieval method for leaf disease localization and identification has been suggested, which combines classification prediction and metric learning. The system detects leaf objects using YOLOv5s and extracts features with ResNet50. Mobius enables rapid vector searches. However, combining advanced models may raise computational complexity and resource requirements, thereby impacting training and processing times.

Kotwal et.al [22] suggested a hybrid approach for plant leaf disease classification (PLDC) based on optimal automatic deep learning (DL). The proposed model first performed preprocessing by shrinking images and applying Gaussian filtering. The UNet technique was then utilized to segment the disease-infected zone to obtain the pertinent region and improve the accuracy of disease classification. The hunter-prey optimization (Hunt-PO) algorithm was employed to adjust the weight of the UNet model during segmentation. For feature extraction, essential features were extracted using the gray level co-occurrence matrix (GLCM), scale-invariant feature transform (SIFT), and a Gabor filter. Finally, artificial driving-EfficientNet (AD-ENet) was used to perform PLDC based on the derived features. However, the image was misclassified due to a small amount of noise present in the sample images.

Aldakheel et.al [23] created sophisticated agricultural disease prediction systems. The dataset was enhanced and the resilience of the model was reinforced through the use of data augmentation techniques such as horizontal flip and histogram equalization. The YOLOv4 algorithm was thoroughly evaluated, and its performance was contrasted with that of well-known target recognition techniques including Densenet, Alexanet, and neural networks. The main drawback is the high dependency on diverse and extensive datasets, which occurs due to the model's need for robust generalization across various plant species and environments, leading to challenges in data collection and processing.

The evaluation of approaches for identifying and detecting plant leaf diseases has made progress, however, there are limitations. The DBNet model increases accuracy but is computationally demanding. The OMNCNN model struggles to isolate disease areas accurately. The tomato disease technique employs GLCM and SVM but has limitations in cross-crop applicability and image resolution. The DenseNet-based technique classifies potato leaf diseases using a vast amount of training data. Transfer learning improves model accuracy and efficiency while lowering training data and increasing model flexibility.

3. PROPOSED METHOD

The research proposal suggests an upgraded method for detecting plant leaf diseases through image processing techniques. It employs a bilateral filter for pre-processing to eliminate noise from leaf images, hence boosting clarity and precision. Transfer learning models such as ResNet-101, EfficientNet-B0, and MobileNet-V3 Large are used to recover rich features from cleaned photos. As seen in the block diagram, this method is highly accurate, and efficient, and requires minimal computer resources.

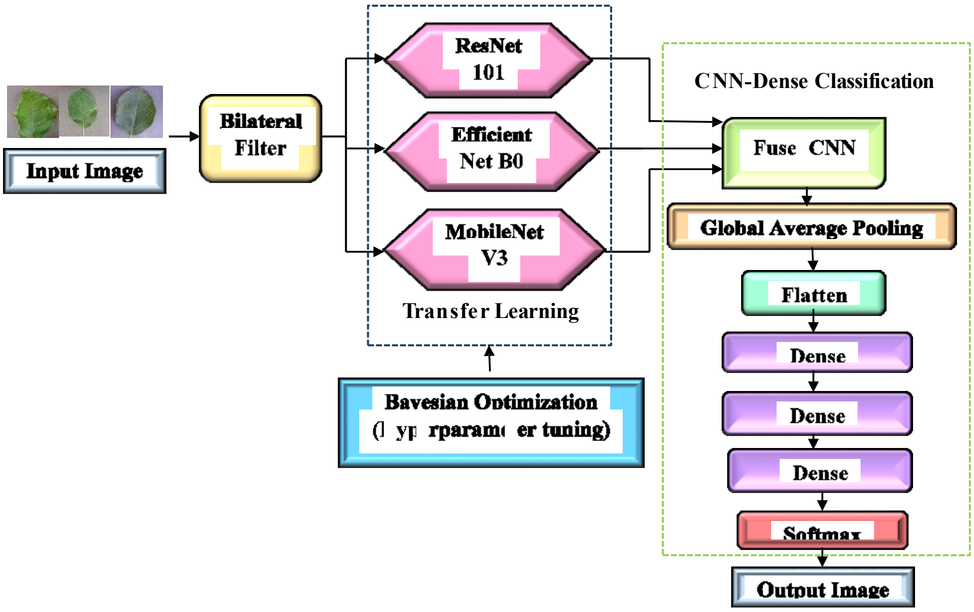


Figure 1: Block Diagram of the proposed Method

After feature extraction, the features are fused using a CNN-Dense classification system. Bayesian optimization is utilized for hyperparameter tuning, optimizing performance, and managing computational resources, while enhancing accuracy without significantly increasing computational demands. This ensures that the classification model is finely tuned for high accuracy and robustness in detecting different plant leaf diseases.

3.1 Bilateral Filter

The BF technique takes a plant leaf image as input and improves image quality by removing noise. It is a nonlinear filter that smoothest and maintains edges. The BF weight is calculated using Euclidean distance and a weighted average of ambient pixel brightness and intensity. This method is a weighted average for Gaussian filters calculated by calculating the distance between two pixels, as shown by eq. (1):

$$GF(\beta,\gamma) = \frac{\sum \delta,\alpha f(\delta,\alpha) w(\beta,\gamma,\delta,\alpha)}{\sum \delta,\alpha w(\beta,\gamma,\delta,\alpha)} \tag{1}$$

serves as basis for the weighting coefficient $w(\beta,\gamma,\delta,\alpha)$. The source of the domain kernel is given as Eq. (2):

$$Dk(\beta,\gamma,\delta,\alpha) = exp\left(-\frac{(\beta-\delta)^2+(\gamma-\alpha)^2}{2\sigma_{dk^2}}\right) \tag{2}$$

Additionally, the range kernel can be found in Eq. (3)

$$rk(\beta,\gamma,\delta,\alpha) = exp\left(-\frac{f(\beta,\gamma)-f(\delta,\alpha)^2}{2\sigma_{dk^2}}\right) \tag{3}$$

Eq. (4) is used to create the BF weight functions after the two kernels are multiplied.

$$w(\beta,\gamma,\delta,\alpha) = exp\left(-\frac{(\beta-\delta)^2+(\gamma-\alpha)^2}{2\sigma_{dk^2}} - \frac{f(\beta,\gamma)-f(\delta,\alpha)^2}{2\sigma_{dk^2}}\right) \tag{4}$$

The bilateral filter effectively reduces noise while maintaining crucial edge details, improving the quality of plant leaf images. This method helps to preserve the fine details of the texture and structure of the leaf by smoothing pixel values based on both spatial distance and intensity similarity. Clearer and more consistent images are obtained by applying the bilateral filter, which reduces noise levels and variations in leaf image quality. This enhanced image quality serves as a useful preprocessing step for a strong and dependable plant leaf disease classification, enabling more precise feature extraction and disease detection.

3.2 Transfer learning based Feature Extraction

Deep learning models must be trained from the beginning, which takes a lot of time and computing resources, such as powerful GPUs and millions of training instances. Transfer learning, which makes use of CNNs that have already been trained, can help to overcome these difficulties. Pre-trained on large datasets, models such as ResNet-101, EfficientNet-B0, and MobileNet-V3 Large, may effectively extract features from cleaned images, hence lowering training time and processing needs. Thus, transfer learning achieves high accuracy while overcoming the shortcomings of high resource needs and delayed training processes. Preprocessed data is sent into each of the networks listed below to extract features.

3.2.1 ResNet-101

Batch normalization and ReLU activation come next. Next, a 3x3 pool size max pooling layer is added. This series of actions is carried out in the conv5 step. The network can learn and extract essential features more effectively by suppressing features with less information and concentrating on features with significant information. This method enhances the model's overall feature extraction capabilities and sensitivity to important characteristics. Figures 2 and 3 show the ResNet 101 Architecture and residual Node Structure given below

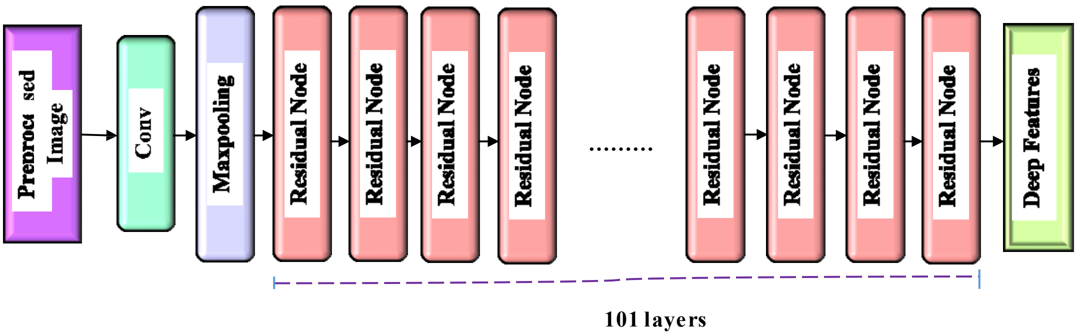


Figure 2: ResNet 101 Architecture

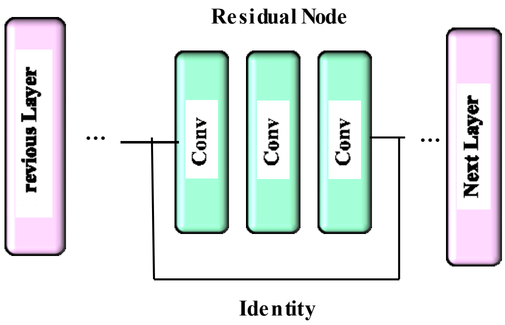


Figure 3: Residual Node Structure

3.2.2 EfficientNet-B0

A series of architectures called Efficient Net was created to determine the most effective way to scale CNNs and enhance model performance. The authors developed the CNN architecture by employing this method. The eight models in the efficient net model group are numbered B0 through B7; higher-number models have more parameters and are more accurate. By changing the network depth, CNNs can capture features that are richer and more complex. Nonetheless, network training becomes increasingly challenging because of the vanishing gradient issue. The network's width may be changed to gather more detailed details. Additionally, training is simple and the starting point of the EfficientNet-B0 model is capable of receiving input images that provide the dimensions of the image as well as its width and height. To capture features across layers, this technique employs a mobile inverted bottleneck convolutional and numerous convolutional layers with a 3 × 3 receptive field. Eq. (5) below illustrates how to scale the width, depth, and resolution. Figure 4 depicts the architecture of Efficient net B0 given below

$$w = \alpha^{\varphi};\ d = \gamma^{\varphi};\ r = \beta^{\varphi} \tag{5}$$
$$s.t.\ \gamma.\alpha^{2}.\beta^{2} \approx 2;\ \alpha \geq 1, \beta \geq 1, \gamma \geq 1$$

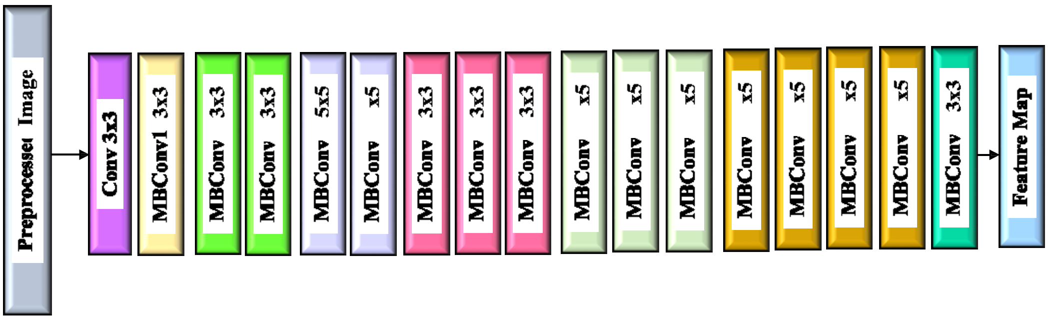


Figure 4: Efficient Net B0 Architecture

where a grid search on a smaller starting model is used to identify the constant coefficients α, γ, and β. are used to scale the network's width, depth, and resolution. Efficient Net can systematically modify these parameters due to its compound scaling technique, which guarantees a balanced increase in both computing efficiency and network capacity. Efficient Net can efficiently capture a broad variety of features in this way, from high-level abstractions to low-level details. On the other hand, because of their shallow depth, large and shallow networks frequently miss high-level information, which limits their capacity to learn complicated representations. A complete feature extraction process is made possible by Efficient Net's thoughtfully designed architecture, which captures both high-level and fine-grained features that are essential for precise predictions.

3.2.3 MobileNet-V3 Large

One of the components of MobileNet V3's stem block is the convolutional layer or conv. In this case, the spatial information from input I→data is captured by the learnable filters. A BN layer comes immediately after the convolution layer and before the ReLU activation layer. Figure 5 illustrates the architecture of mobile net v3 large and Eq. (6) is given below

$$Block_{Stem} = Conv([Block_{norm}[ReLuConv(I)]]) \quad (6)$$

Squeeze and excitation operations, separable convolutions, and a linear bottleneck are the components of the model's significant blocks, also known as inverted residual blocks, which are based on the idea of depth-wise separable convolutions given as Eq. (7)

$$Block_{Invres} = L_{Bottleneck}\ ([Depth_{sep}[X_{stem}]]) \quad (7)$$

To calculate the average of each feature map in the final layer for a latent feature vector, the acquired feature maps are pooled using the Global Average pooling function given as eq. (8)

$$Pool_{(avg,global)}(Block_{(resinv)}) \quad (8)$$

The following is the feature vector that was retrieved from MobileNet V3 is given as eq. (9)

$$FV_1 = [\mathrm{m}_1, \mathrm{m}_2, \ldots, \mathrm{m}_n] \quad (9)$$

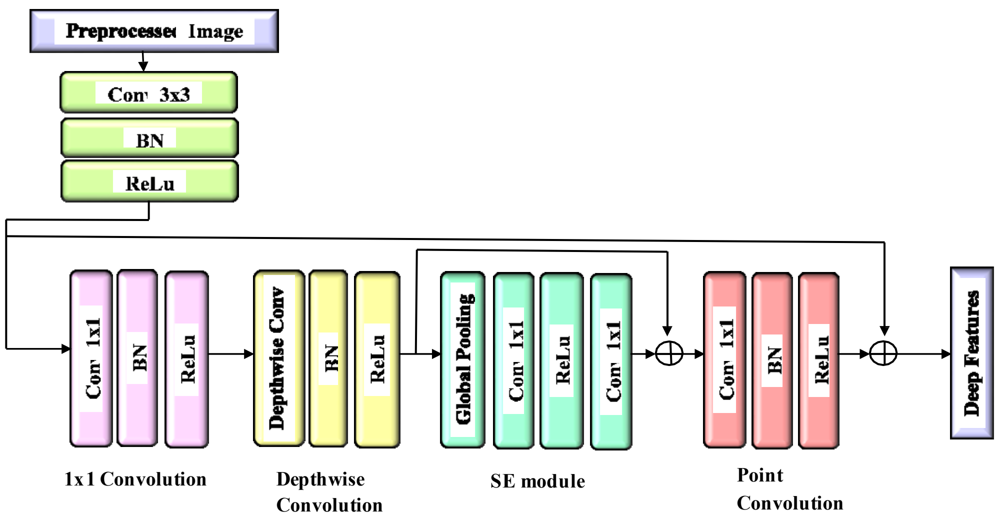


Figure 5: Mobile Net V3 Large Architecture

3.3 CNN Dense Classification Layer

The study suggests creating a module that combines down-sampling feature maps from fused convolution blocks with output maps from the averaging pooling layer. This feature fusion module minimizes feature loss while increasing feature information. The module improves target pest visibility by increasing the contrast between pest features and backdrop and decreasing similarity in feature maps.

Which is the result of the linear and nonlinear transformation in the fused-convolution block. The formula $F(A)$ can be represented as eq. (10)

$$F(A) = T_l\{T_{nl}[T_l(A)]\} \quad (10)$$

where $T_l$denotes linear transformation and T_{nl} stands for nonlinear transformation. The feature map that results from the average pooling layer's linear transformation is $X(A)$. It has a wide perceptual field and no redundant information. It can be stated concisely as eq. (11)

$$X(A) = T_l(A) \quad (11)$$

It can be explained by Equation (12).

$$Z = add[T_l\{T_{nl}[T_l(A)]\}, T_l(A)] \quad (12)$$

Finally, the classification process includes Global Average Pooling to reduce dimensionality, followed by a sequence of three dense layers for complex feature representation. The output is then flattened before passing through a Softmax layer, which accurately categorizes the different plant leaf diseases.

3.4 Bayesian Optimization for Hyperparameter Tuning

Bayesian optimization is an algorithm made up of four parts: an objective function, an optimizer method, and a hyperparameter space search. The goal function is only specified after the model has been set up, defined, allocated, trained, and tested, therefore it is considered random. The Bayes theorem is used in the optimization approach, hence the name Bayesian. This method optimizes the model's performance using a collection of hyperparameters and alternative solutions.

In this case, given the proof that data D provides, the posterior probability $P(m|D)$, defined by Eq. (13), where m is a model proportional to the probability $P(D|m)$) of overserving D given model m multiplied by the prior probability of $P(m)$.

$$P(m|D) = \frac{P(D|m)P(m)}{P(D)} \quad (13)$$

Gaussian Process (GP) was employed as the probabilistic model in this investigation. The hyperparameters are the domain in which the data generated by the GP model are located.
As a result, Bayesian optimization is more efficient than random and grid searches.

4. RESULT AND DISCUSSION

This section includes the findings and information from the suggested model's evaluation. The section goes into detail about the dataset and shows a depiction of the testing setup. Many tests to assess the efficacy of the proposed method have been reported.

4.1 System Configuration

The system configurations for applying the Transfer learning model with Python 3.9. For basic computing, an Intel Core i5 processor and 16 GB of RAM are suggested. Nvidia GPUs can accelerate computations in Python frameworks such as TensorFlow and PyTorch. Python programs, libraries, and datasets can all be kept on a 1 TB HDD, resulting in quicker read/write speeds and better system performance. The Windows 10 operating system supports Python and related libraries.

4.2 Dataset description

The Plant Village collection, generated by researchers at Pennsylvania State University, includes over 50,000 photos of healthy and damaged plant leaves from 14 different species. It helps to build computer vision algorithms for automated plant disease identification. The dataset is utilized in a variety of programs and investigations. Eighty percent of the dataset was used for testing, with the remaining 20% used for training. The suggested framework showed good optimization and learning with a batch size of 64 and 16 subdivisions.

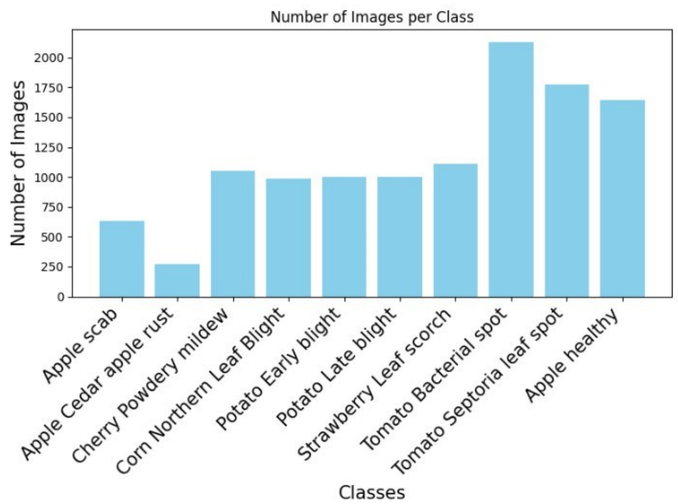


Figure 6: Dataset distribution

4.3 Result Obtained from the Proposed Method

This section provides the result of preprocessed output and Prediction output will be discussed below.

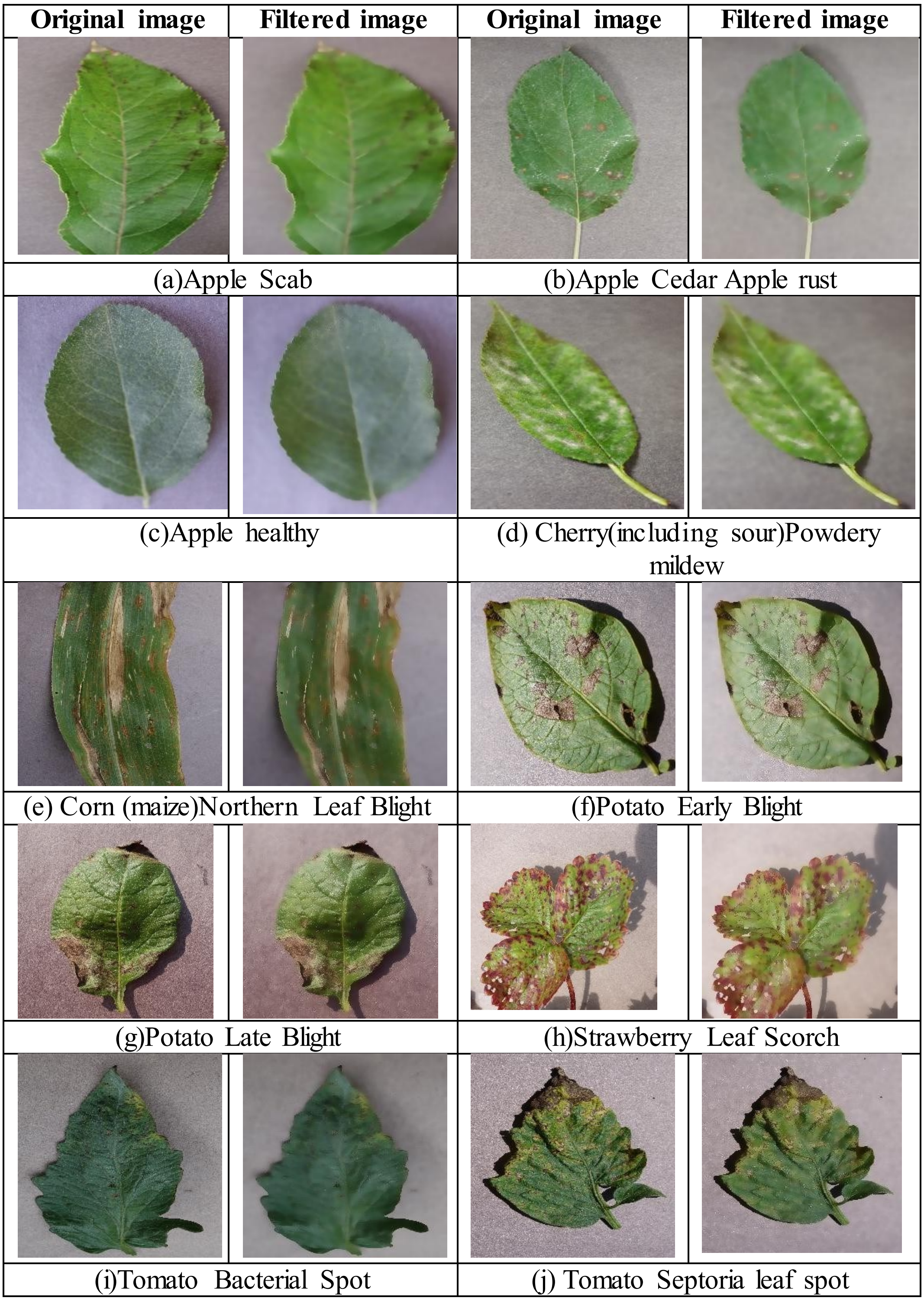


Figure 7: Preprocessed Output

Figure 7 depicts the preprocessed output using a bilateral filter. The bilateral filter in the preprocessed output effectively reduces noise while maintaining important edge details, improving the images of plant leaves. By using spatial and intensity similarities to smooth pixel values, this method preserves the leaf's primary features and textures. Consequently, the images exhibit enhanced clarity and consistency, thereby furnishing a robust basis for precise feature extraction and disease classification. Enhancing the overall quality and dependability of plant leaf disease detection requires this preprocessing step.

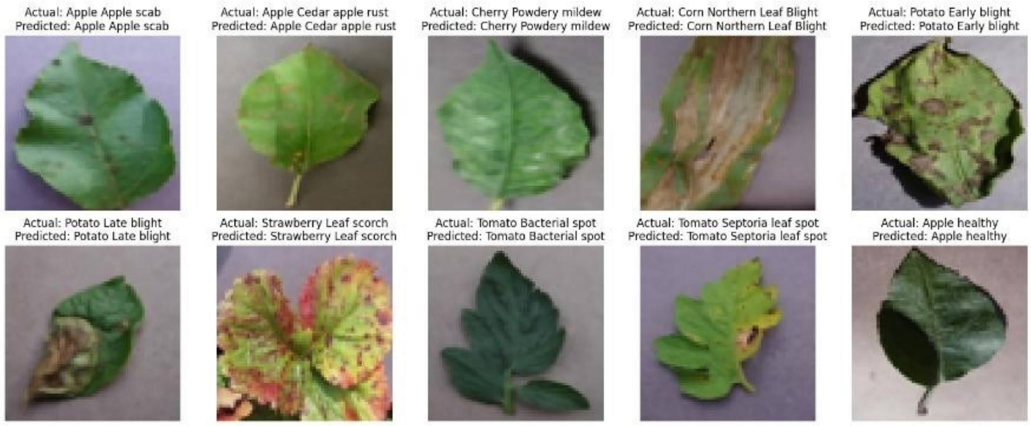
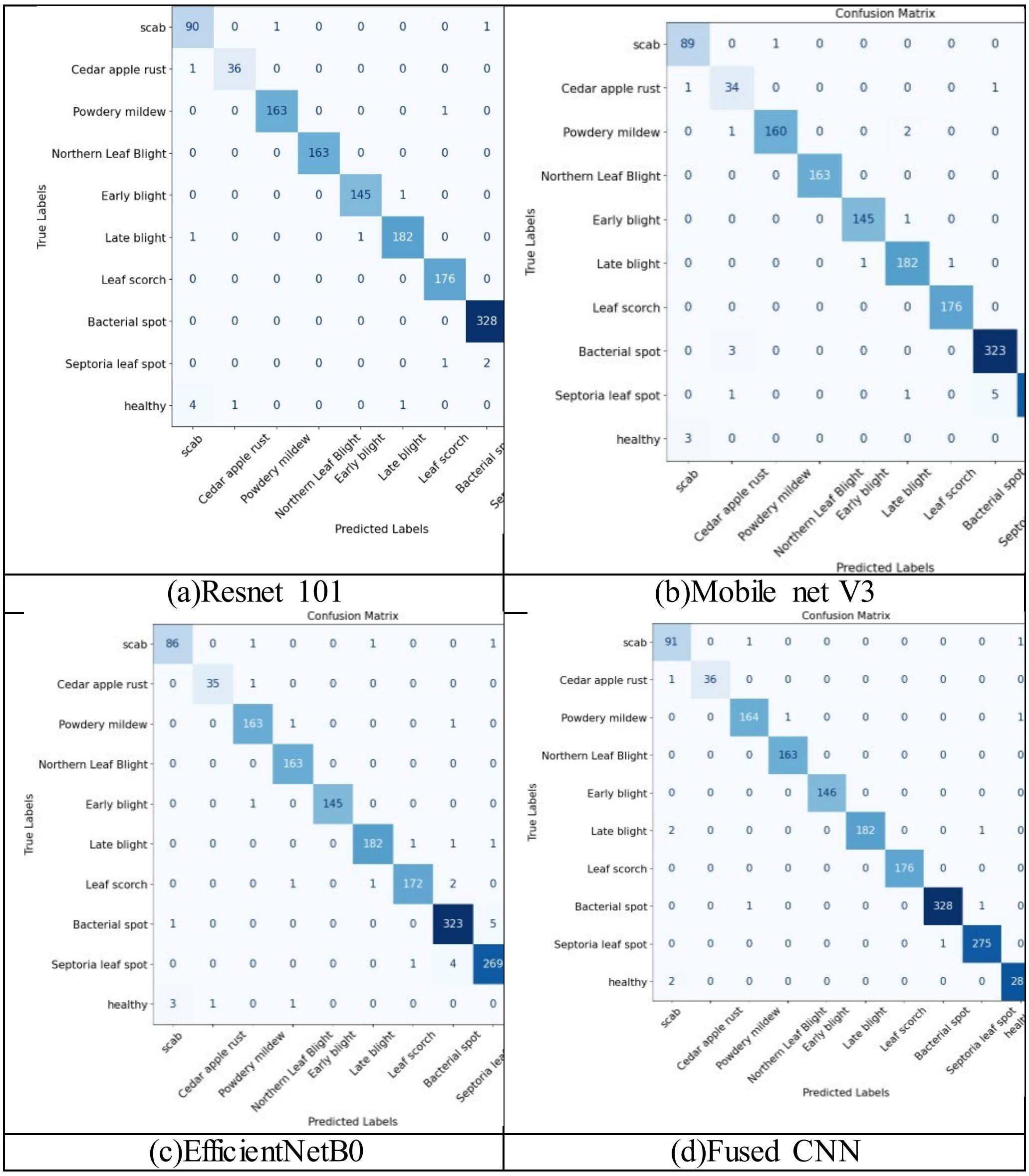


Figure 8: Prediction Output

The prediction output from a CNN-based dense classifier that used the Plant Village dataset to facilitate transfer learning is shown in Figure 8. Plant leaf images are classified with high accuracy into distinct disease categories by the model, which has been fine-tuned using pre-trained weights. The CNN architecture processes each image by first extracting features using convolutional layers and then classifying those using dense layers. Plant leaf conditions are illustrated by the output, which is represented graphically in terms of predicted disease labels and related confidence scores.

4.4 Performance Evaluation

The proposed approach's performance is assessed using a confusion matrix, which is a table that summarises the effectiveness of a classification algorithm. Figure 9 shows the confusion matrix for the proposed transfer learning classifier. It indicates the model's performance when comparing predicted and real labels, such as TP, TN, FP, and FN.

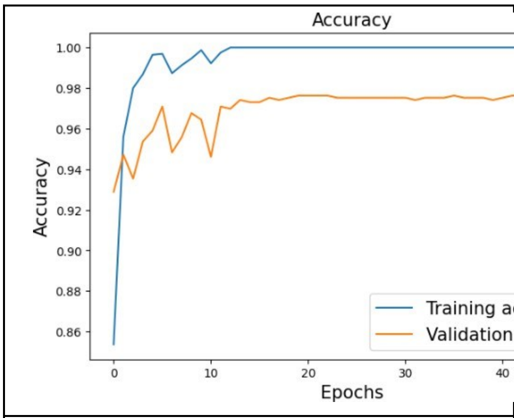


(a)Resnet 101 (b)Mobile net V3

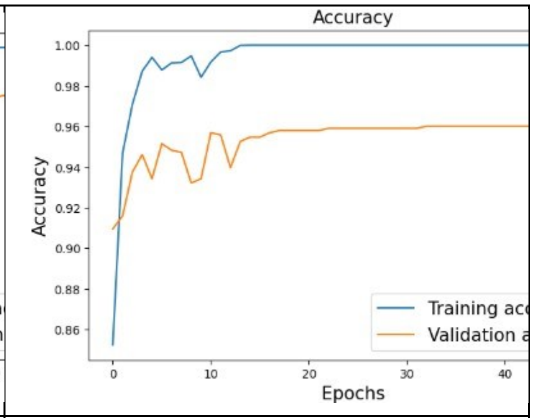
(c)EfficientNetB0 (d)Fused CNN

Figure 9: Confusion Matrix for Transfer Learning

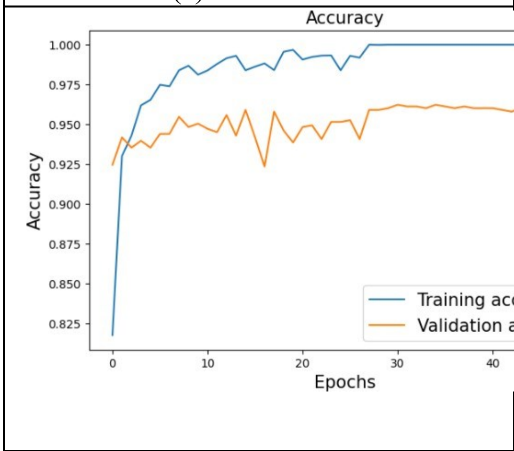
Assessing a transfer learning technique's performance and identifying potential issues like underfitting and overfitting involves interpreting training, accuracy, and loss.



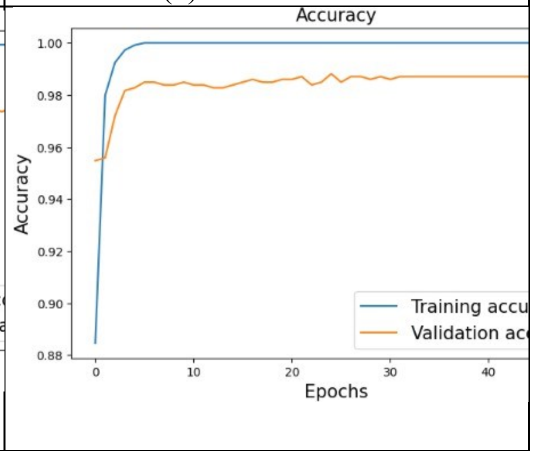
(a)Resnet 101



(b)Mobile net v3



(c)Efficient Net B0



(d)Fusion CNN

Figure 10: Training and Validation Accuracy

The suggested method represents the actual and expected classes with a matrix, where each row represents the ground truth and each column represents the predicted class. The matrix members contain information about the classifier's performance. Figure 10 depicts training and validation accuracy, with training and validation losses discernible. The model's training graph shows that it is still learning, and a constant decreasing trend in error loss is required as accuracy improves.

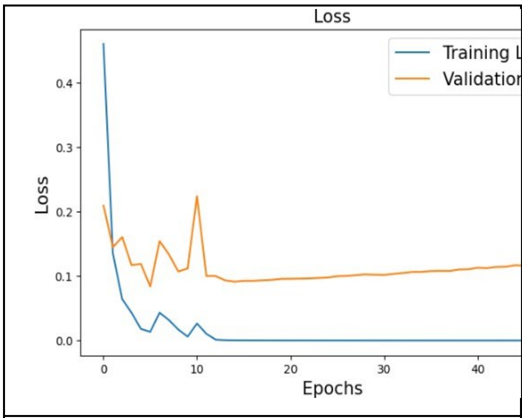
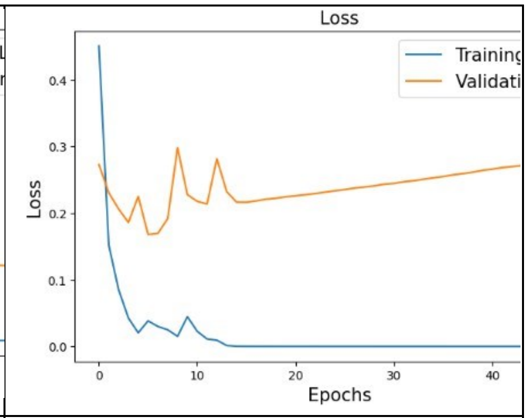
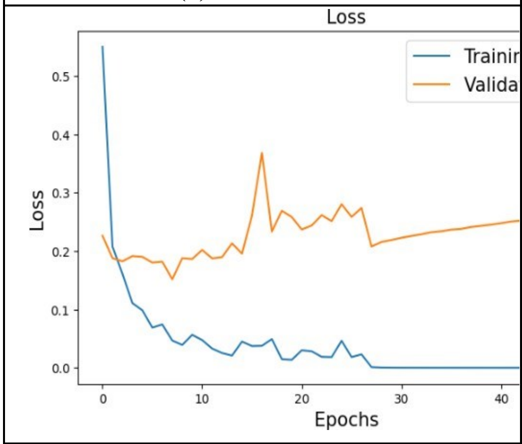
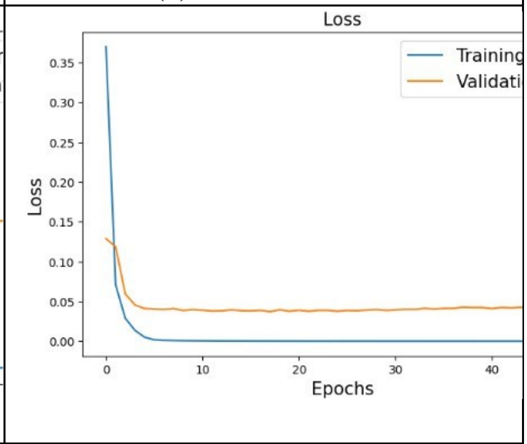
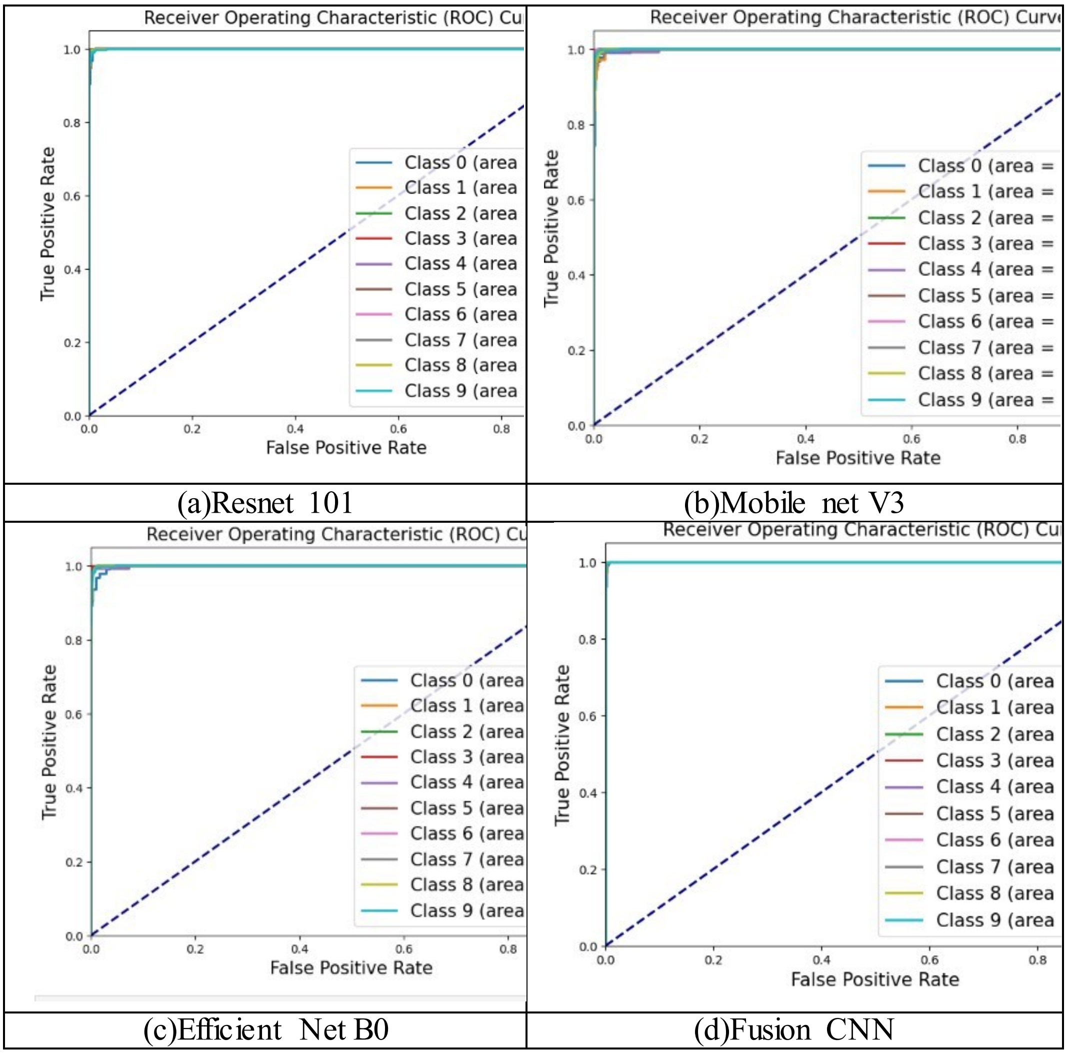
	
(a)Resnet 101	(b)Mobile net V3
	
(c)Efficient Net B0	(d)Fusion CNN

Figure 11: Training and Validation Loss

The model has learned everything it can from the training set of data when the loss plateaus. The research employs both original and upgraded datasets, yielding an enormous total dataset that can yield important results that serve as both proof of concept and insights into the viability of a method in practical settings.



(a)Resnet 101 (b)Mobile net V3 (c)Efficient Net B0 (d)Fusion CNN

Figure 12: RoC Curve

Figure 12 depicts the transfer learning model's ROC curve and performance at various threshold levels. TPR and FPR plots demonstrate their ability to discriminate plant leaf disease groups. The curve's shape and AUC reflect its ability to classify diseases, with a larger AUC suggesting better overall performance. The model obtains a high TPR while maintaining a low FPR, indicating superior prediction accuracy.

4.5 Comparative Analysis:

The transfer learning typical with existing models such as Resnet [21], Squeeze-and-Excitation deep block (SSD) [24], Yolo v5 [25], Alexander [26], SVM Classifier [27], DenseNet-121 [28] and Convolutional Neural Network (CNN) [14].

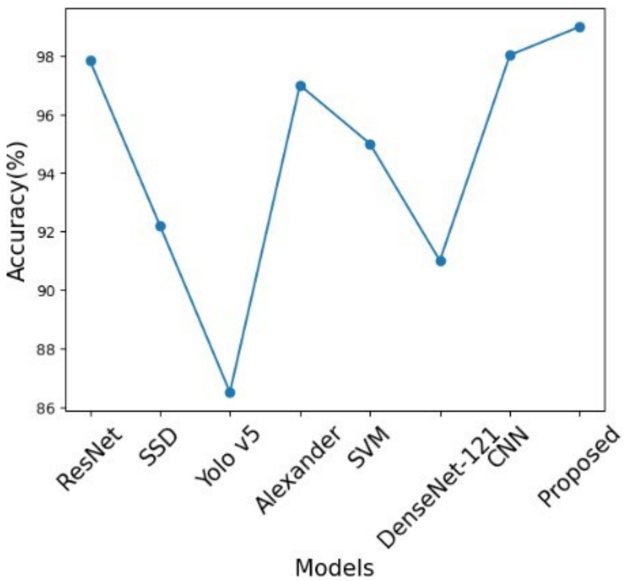


Figure 13: Accuracy Comparison

Figure 13 shows that the transfer learning model outperforms conventional techniques in classifying plant leaf diseases, with a consistently higher accuracy of 99.9%.

Table 1: Accuracy Comparison of proposed transfer learning with existing models

Models	**Accuracy (%)**
ResNet	97.84
SSD	92.20
Yolo V5	86.5
Alexander	97
SVM	95
DenseNet-121	91
HCNN	98.02
Proposed	**99.9**

The proposed transfer learning model surpasses existing models in classifying plant leaf diseases with an outstanding accuracy of 99.9%, surpassing many techniques, this highlights the efficacy of the transfer learning technique, which uses pre-trained networks to boost classification accuracy. The model's ability to change and update existing knowledge of specific plant leaf disease traits improves disease detection precision and dependability.

5. CONCLUSION

The study proposes a unique method for detecting plant leaf disease using advanced image processing and the TL Method. Bilateral filtering is used to remove noise from leaf images, resulting in greater precision and clarity. Transfer learning is combined with advanced algorithms such as ResNet-101, EfficientNet-B0, and MobileNet-V3 Large to extract detailed features, which improves detection performance. When the CNN-Dense classification system is combined with Bayesian optimization for hyperparameter adjustment, its accuracy improves even further. The proposed method demonstrates a high classification accuracy of 99.9% while maintaining computational efficiency, making it a reliable tool for plant leaf disease detection. This performance surpasses all existing approaches, with HCNN achieving 98.02%, ResNet and Alexander models reaching 97.84% and 97%, respectively, SVM attaining 95%, SSD at 92.20%, and DenseNet-121 at 91%. YOLO V5 records the lowest accuracy at 86.5%. These findings align with recent research that emphasizes the effectiveness of integrating advanced transfer learning models and optimized classification systems for plant disease detection. The proposed model's superior results further validate this trend, highlighting its robustness and adaptability compared to traditional and contemporary approaches. Future research will concentrate on enhancing feature extraction methods and incorporating more data sources into the model.

REFERENCES

[1]. Sujatha, R., Chatterjee, J. M., Jhanjhi, N. Z., & Brohi, S. N. (2021). Performance of deep learning vs machine learning in plant leaf disease detection. *Microprocessors and Microsystems*, *80*, 103615.

[2]. Zubler, A. V., & Yoon, J. Y. (2020). Proximal methods for plant stress detection using optical sensors and machine learning. *Biosensors*, *10*(12), 193.

[3]. Roberts, D. P., Short Jr, N. M., Sill, J., Lakshman, D. K., Hu, X., & Buser, M. (2021). Precision agriculture and geospatial techniques for sustainable disease control. *Indian Phytopathology*, *74*(2), 287-305.

[4]. Souto, A. L., Sylvestre, M., Tölke, E. D., Tavares, J. F., Barbosa-Filho, J. M., & Cebrián-Torrejón, G. (2021). Plant-derived pesticides as an alternative to pest management and sustainable agricultural production: Prospects, applications, and challenges. *Molecules*, *26*(16), 4835.

[5]. Li, L., Zhang, S., & Wang, B. (2021). Plant disease detection and classification by deep learning—a review. *IEEE Access*, *9*, 56683-56698.

[6]. Jackulin, C., & Murugavalli, S. (2022). A comprehensive review of the detection of plant disease using machine learning and deep learning approaches. *Measurement: Sensors*, *24*, 100441.

[7]. Gajjar, R., Gajjar, N., Thakor, V. J., Patel, N. P., & Ruparelia, S. (2022). Real-time detection and identification of plant leaf diseases using convolutional neural networks on an embedded platform. *The Visual Computer*, 1-16.

[8]. Sharma, A., Jain, A., Gupta, P., & Chowdary, V. (2020). Machine learning applications for precision agriculture: A comprehensive review. *IEEE Access*, *9*, 4843-4873.

[9]. Sinha, A., & Shekhawat, R. S. (2020). Review of image processing approaches for detecting plant diseases. *IET Image Processing*, *14*(8), 1427-1439.
[10]. Hassan, S. M., Amitab, K., Jasinski, M., Leonowicz, Z., Jasinska, E., Novak, T., & Maji, A. K. (2022). A survey on different plant diseases detection using machine learning techniques. *Electronics*, *11*(17), 2641.
[11]. Abade, A., Ferreira, P. A., & de Barros Vidal, F. (2021). Plant diseases recognition on images using convolutional neural networks: A systematic review. *Computers and Electronics in Agriculture*, *185*, 106125.
[12]. Khalid, M. M., & Karan, O. (2024). Deep learning for plant disease detection. *International Journal of Mathematics, Statistics, and Computer Science*, *2*, 75-84.
[13]. Sangeetha, T., & Mohanapriya, M. (2022). A novel exploration of plant disease and pest detection using machine learning and deep learning algorithms. *Mathematical Statistician and Engineering Applications*, *71*(4), 1399-1418.
[14]. Jasim, M. A., & Al-Tuwaijari, J. M. (2020). Plant leaf disease detection and classification using image processing and deep learning techniques. *In 2020 International Conference on Computer Science and Software Engineering (CSASE)*, 259-265.
[15]. Harakannanavar, S. S., Rudagi, J. M., Puranikmath, V. I., Siddiqua, A., & Pramodhini, R. (2022). Plant leaf disease detection using computer vision and machine learning algorithms. *Global Transitions Proceedings*, *3*(1), 305-310.
[16]. Kotwal, J., Kashyap, R., & Pathan, S. (2023). Agricultural plant diseases identification: From traditional approach to deep learning. *Materials Today: Proceedings*, *80*, 344-356.
[17]. Ashwinkumar, S., Rajagopal, S., Manimaran, V., & Jegajothi, B. (2022). Automated plant leaf disease detection and classification using optimal MobileNet based convolutional neural networks. *Materials Today: Proceedings*, *51*, 480-487.
[18]. Perveen, K., Kumar, S., Kansal, S., Soni, M., Alshaikh, N. A., Batool, S., & Osei, B. (2023). [Retracted] Multidimensional Attention-Based CNN Model for Identifying Apple Leaf Disease. *Journal of Food Quality*, *2023*(1), 9504186.
[19]. Rahman, S. U., Alam, F., Ahmad, N., & Arshad, S. (2023). Image processing-based system for the detection, identification, and treatment of tomato leaf diseases. *Multimedia Tools and Applications*, *82*(6), 9431-9445.
[20]. Mahum, R., Munir, H., Mughal, Z. U. N., Awais, M., Sher Khan, F., Saqlain, M., & Tlili, I. (2023). A novel framework for potato leaf disease detection using an efficient deep learning model. *Human and Ecological Risk Assessment: An International Journal*, *29*(2), 303-326.
[21]. Peng, Y., & Wang, Y. (2022). Leaf disease image retrieval with object detection and deep metric learning. *Frontiers in Plant Science*, *13*, 963302.
[22]. Kotwal, J. G., Kashyap, R., & Shafi, P. M. (2024). Artificial driving based EfficientNet for automatic plant leaf disease classification. *Multimedia Tools and Applications*, *83*(13), 38209-38240.

[23]. Aldakheel, E. A., Zakariah, M., & Alabdalall, A. H. (2024). Detection and identification of plant leaf diseases using YOLOv4. *Frontiers in Plant Science*, *15*, 1355941.
[24]. Wang, J., Yu, L., Yang, J., & Dong, H. (2021). DBA_SSD: A novel end-to-end object detection algorithm applied to plant disease detection. *Information*, *12*(11), 474.
[25]. Chen, Z., Wu, R., Lin, Y., Li, C., Chen, S., Yuan, Z., & Zou, X. (2022). Plant disease recognition model based on improved YOLOv5. *Agronomy*, *12*(2), 365.
[26]. Hassan, S. M., Maji, A. K., Jasiński, M., Leonowicz, Z., & Jasińska, E. (2021). Identification of plant-leaf diseases using CNN and transfer-learning approach. *Electronics*, *10*(12), 1388.
[27]. Chuanlei, Z., Shanwen, Z., Jucheng, Y., Yancui, S., & Jia, C. (2017). Apple leaf disease identification using genetic algorithm and correlation-based feature selection method. *International Journal of Agricultural and Biological Engineering*, *10*(2), 74-83.
[28]. Arathi, B., & Dulhare, U. N. (2023, March). Classification of cotton leaf diseases using transfer learning-dense net-121. In *Proceedings of the third international conference on advances in computer engineering and communication systems: ICACECS 2022,* 393-405.