Comparative analysis of CNN-based pre-trained architectures for grape leaf disease detection

1th Bramha Nimbalkar *Student, Btech*
Artificial intelligence and data science,
Vishwakarma University, Pune, India
202100381@vupune.ac.in

Abstract-Addressing plant diseases is a critical concern in agriculture, posing a significant threat to food quality. Currently, the identification of crop diseases is done by the naked eye which is time-consuming and also error-prone.[1] To take timely action against plant diseases, it is essential to monitor and detect diseases as early as possible. To reduce identification time and human error, Machine learning models, specifically deep learning models are very beneficial.[1] The models used for predictions are based on Convolutional Neural Networks (CNN) which include ResNet50, VGG16, and EfficientNetV2B3. The study utilizes an image dataset featuring grape crops sourced from Grapevine Disease Images. The three most common grape leaf diseases—black rot, ESCA, and leaf blight—have been targeted due to their adverse impact on crop yield. The findings aim to determine the most effective model for accurate and early detection of grape leaf diseases, contributing to precision grape farming. Among the three models assessed, ResNet50 is the most effective, achieving the highest accuracy of 99.11% and equal values for precision, F1 score, and recall of 0.99. These results determine that ResNet50 is significantly reducing the time and error of identification, offering the advancement of precision agriculture in the grape farming sector.

Keywords: Grape leaf disease detection, image processing, Convolutional Neural Network, Data augmentation, Black rot, ESCA, leaf blight, ResNet50, VGG16, EfficientNetV2B3.

I. INTRODUCTION

India is one of the world's leading grape producers. In India, Grapes are cultivated in an area of 111.4 thousand ha with a total production of 1,234.9 thousand tons and productivity of 11.1 tons/ha.[2] [3] Grapes are considered very important from a business perspective as they can be exported to different countries or used for table purposes.[4] It has a good amount of nutritional minerals like vitamins C, K, and B.[4] The Indian grape industry has undergone significant growth in recent years, with the adoption of modern cultivation techniques, better infrastructure, and increased exports to foreign markets. However, severe diseases adversely affect the growth and quality of grapes.[5]So, it is necessary to prevent the diseases at the early stages.

The traditional disease detecting method through the naked eye is not efficient. Some farmers use insecticides and pesticides, but it can be harmful for humans. As there is also a risk of human error, image processing techniques based on CNN can be used for achieving more efficiency.

A Convolutional Neural Network (CNN) is a type of Deep Learning neural network architecture which is used for Computer Vision application.[6][7][8] CNN is the revised version of Artificial Neural Networks (ANN) which is majorly used to obtain important features from the grid-like matrix dataset.[6] To extract important features form the input images filter are applied in the Convolution layer, spatial dimension of the image is decreased in the Pooling layer to reduce the computation, and extracted features are then integrated in the fully connected layer.[6] The network learns to optimize filter weights through backpropagation and gradient descent.[6]

This paper proposes the implementation of CNN-based pretrained models for early disease detection. The major contributions of the article include:

- Integrating pre-trained ResNet50 architecture to detect grapevine leaf diseases using a dataset of grapevine leaf images, expanded through various data augmentation techniques to establish sufficient amount of data and robustness of the proposed model.
- A comparative study of three different pretrained models, examining the trainable parameters and their influence on training and performance metrics such as accuracy, precision, recall, and F1 score.
- An in-depth analysis of the ResNet50 architecture, which outperforms the other models in accuracy, with a detailed reasoning of the factors contributing to its enhanced performance.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 06 | June - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

II. Related Work

In the context of our investigation into grape leaf disease detection, a thorough review of the literature was conducted, wherein 10 to 15 research papers were scrutinized for their relevance and contributions to the viticulture field. Out of this extensive review, five studies were identified as particularly influential, offering significant advancements in detection methodologies. These studies are highlighted in Table 1, showcasing their unique approaches and findings.

Table 1. A tabular representation of literature survey related to the grape leaf disease detection

Reference No.	ML model	Performance metrics	Inference
[3]	DR-IACNN	The detection performance of DR-IACNN model achieved an 81.1% mean average precision (mAp) with a processing speed of 15.01 fps.	The detection method based on deep learning was executed using a caffe framework on GPU platform which resulted in improvement of DR-IACNN. Here, using pre-trained models may increase the efficiency.
[9]	CNN	When tested on a PlantVillage dataset the model surpassed the performance of pretrained models thus giving an accuracy of 99%.	Algorithms in this experiment were tested using diverse evaluation metrics including accuracy, precision, recall, storage space and AUC-ROC.
[10]	Random forest	The model has obtained 91.66% accuracy in 80% - 20% training – testing dataset.	This model works differently on different fractions of the dataset, giving best results on 80% - 20%,training-testing dataset.
[11]	CNN, RNN	In this research paper, CNN model is trained on 25 epochs which results in 99.31% accuracy, whereas RNN model is trained on 10 epochs resulting in 33.03% accuracy.	The performance of the CNN model clearly stands out for image classification problems over RNN models. Also the pre- trained CNN model can outperform the RNN models.

III. Proposed Work

Our study involves the use of pre-trained CNN architecture, ResNet50 for enhancing early detection and management of grape leaf diseases such as Black rot, ESCA and blight. Using pre-trained models for a specific task leverages the knowledge and experience of that model, saving time and resources and improving the model's performance. Pre-trained models are often used as a starting point in transfer learning, where the weights of the pre-trained model are transferred to another model to leverage existing knowledge to improve performance on related problems. Moreover, as pre-trained models have been trained on large and diverse datasets, it has a comprehensive understanding of a wide range of features and patterns, enhancing their adaptability and generalization capabilities. This results in improved model performance as they provide a strong foundation for building new models based on some specific problem statement [12][13]. [14][15]. The diagram in fig. 1, involves different stages providing a clear understanding of the steps taken for our proposed work to detect the grapevine leaf diseases.

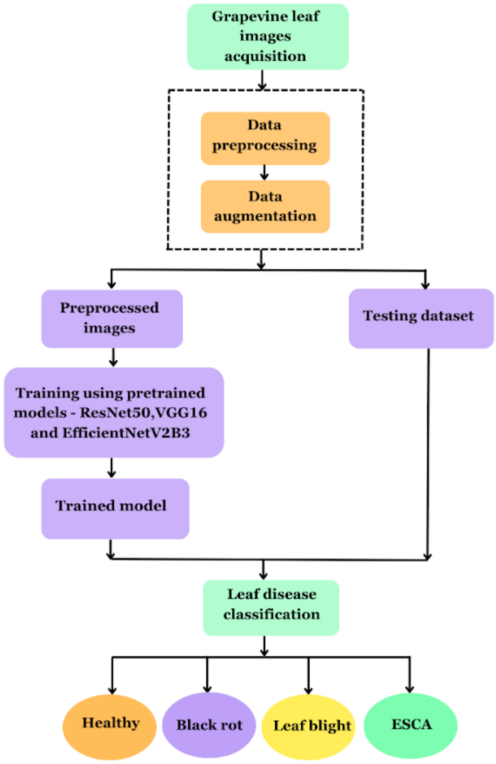

Fig 3 Flow Chart of the Grape Leaf Disease Detection.

IV. Data Acquisition

The grape plant suffers from diseases in different seasons, temperature, and humidity. For instance, it suffers from black rot in hot and humid weather and from leaf blight in low temperatures, when rain is frequent.[3] So, considering all the situations of climate, the dataset should be diverse. The dataset of grape leaves was taken from the Kaggle which includes a total of 4062 images belonging to all classes as shown in fig.1.

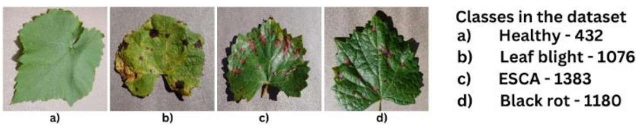

Fig. 1. Four common types of grapes leaf diseases

V. IMAGE PREPROSESSING

The processes of experiments conducted for image preprocessing are described in below sections, 4.2.1, 4.2.2 and 4.2.3.

A. *Image Augmentation*

Image augmentation is a process of creating new training examples from existing ones. To make a new sample, you slightly change the original image.[16] For instance, you could make a new image by adjusting brightness, rotating the images, cropping the original image, horizontal and vertical flipping, mirroring the images, shearing the images, etc.[16] Before augmentation, there were 4062 images and after augmentation, the count of images increased significantly which led to a total of 9027 images. The augmented images were created using ImageDataGenerator API in Keras which generates batches of images using real-time data augmentation.[17]

Table 2. Image augmentation techniques used

Sr. no.	Augmentation technique	Parameters
1	Rotation	Small-angle rotations (90 degrees) introduce variability in object orientations
2	Horizontal Flip	Enhances dataset symmetry by randomly flipping images horizontally
3	Vertical Flip	Enhances dataset symmetry by randomly flipping images vertically

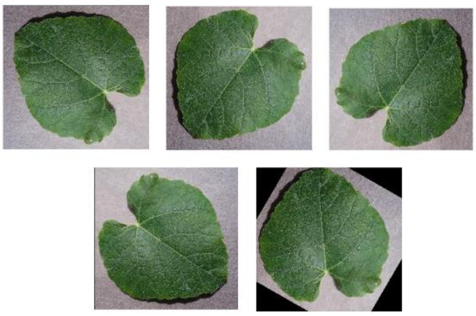

Fig 2 Data augmentation results of grape leaf.

B. *Image Renaming and Normalization*

Renaming images in the dataset is crucial as descriptive and consistent filenames provide valuable information about the content. The pattern adopted for renaming images is:

<class_name>#number

This pattern enhances the clarity of image identification and facilitates subsequent analysis and model training tasks. For renaming, the tool used was IrfanView 64 as shown in figure 3.

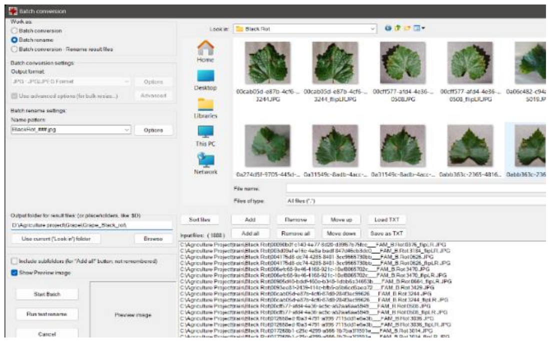

Fig 3 Renaming Process

Table 3. Image count before and after applying augmentation

Classes	No. of Images Before Augmentation	No. of Images After Augmentation
Black Rot	1180	2360
ESCA (Black Measles)	1383	2400
Healthy	423	2115
Leaf Blight (Isariopsis)	1076	2152
Total	4062	9027

Image normalization is a crucial step in image preprocessing, particularly when preparing data for deep learning models. By dividing the pixel values of input images by 0.1/255, the pixel intensities are scaled effectively to a range between 0 and 1. Normalization can help avoid numerical instability during model training by reducing the range of pixel values.

VI. FEATURE ENGINEERING AND CLASSIFICATION

We implement a pretrained network architecture, ResNet-50 for grape leaf disease detection. It contains 50 layers which are organized into 5 blocks in which each block contains a series of residual blocks. These residual blocks are very important in holding the information from the previous layers, thus helping the network to optimize its performance [18][19].

5.1.1 Convolutional Layers

Initially the network uses the convolutional layer so that the input image is processed through convolution. Then the output of the convolutional layer is then processed by the max-pooling layer in order to reduce its dimensionality. The resultant output from max-pooling layer then undergoes a sequence of residual blocks.[18]

5.1.2 Residual Blocks

There are two convolutional layers present in each residual block, followed by a subsequent batch normalisation layer and Rectified Linear Unit (ReLU) activation function. Then the input to the next residual block is the combination of the result of the second convolutional layer which is followed by another application of ReLU activation function, therefore the output from the residual block is then passed on to the next block.[18]

5.1.3 Fully Connected Layer

Finally, in the network architecture there exists a fully connected layer responsible for transforming the output of the last residual block to required output classes. The number of neurons within the fully connected layer equals the number of output classes.[18]

5.1.4 Skip connection

The use of skip connection within the ResNet-50 architecture tackles the problem of vanishing gradients which allow the gradients to pass from the later layers to the preceding layers. Due to this the deeper layers are trained in a better way thus enhancing the network's capacity to learn complex and abstract features.[18]

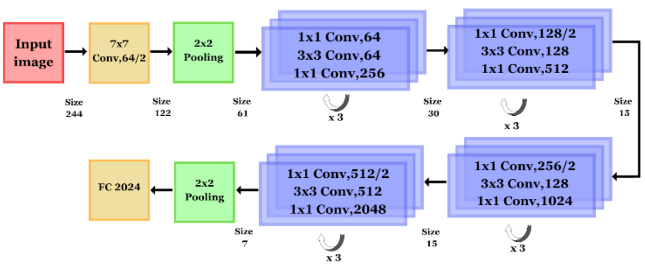

Fig. 4. Architecture of our proposed work to detect grape leaf diseases

ResNet-50, due to its use of the skip connections and residual blocks has remarkably improved its performance in terms of computer vision by performing diverse tasks thus turning out to be widely used architecture in the realms of deep learning.[20]

Table no.4: Proposed model summary for grape leaf disease detection

Layer Number	Layer Type	Kernel Size	Number of Kernels	Stride
1	7×7 Convolution	64	2	112×112×64
2	Max Pooling	3×3	-	56×56×64
3-5	Bottleneck Residual Block	1×1, 64	3×3, 64	1×1, 256
6-9	Bottleneck Residual Block	1×1, 128	3×3, 128	1×1, 512
10-13	Bottleneck Residual Block	1×1, 256	3×3, 256	1×1, 1024
14-16	Bottleneck Residual Block	1×1, 512	3×3, 512	1×1, 2048

Fig 5 Skip Connection Function

<u>Procedure for grape leaf disease detection using ResNet</u>

Step 1

The initial convolutional and max pooling layers involve a 7x7 convolutional layer featuring 64 filters and using a stride of 2, which reduces the image dimensions to 119x119. A max pooling layer with a stride of 2 is employed, resulting in a further reduction of the image size to 59x59.[21] The convolution operation can be shown through the equation (1):

$$Y = X * W + b \qquad \text{Equ(1)[21]}$$

where * is the convolution operation, X being input, W is the weight matrix, b is the bias, and Y is the output.

Step 2

All residual blocks consist of multiple convolutional layers along with a skip connection. Equation (2) shows how the output Y is computed from a residual block:

$$Y = F(X, Wi) + WsX \qquad \text{Equ.(2)[2]}$$

The function F(X, {W_i}) represents the stacked non-linear layers, such as the convolutional layer in the block , and W_sX is the skip connection. When the dimensions of X and F differ, then to align dimensions a linear projection W_s is used.[21]

Step 3

A Rectified Linear unit (ReLU) activation function is applied for each convolution operation in the residual block. The equation (3) used is

$$f(x) = \max(0, x) \qquad \text{Equ. (3)}$$

Step 4

Finally there is an average pooling layer and a fully connected layer at the end of the network that gives the results of final classification.[21] The average pooling operation can be shown using equation (4)

$$Y_{ij} = \frac{1}{K^2}\sum_{m=0}^{K-1} \sum_{n=0}^{K-1} X_{i+m,j+n} \qquad \text{Equ. (4)[22]}$$

where X is the input, Y is the output, and K is the kernel size.

VII. Feature Engineering and Classification

The dataset is trained and then evaluated on different models which include ResNet50, EfficientNetV2B3, and VGG16. Different accuracies are obtained by different models based on their architecture, as shown table 5. The evaluation metrics used for the models are Accuracy, Precision, Recall, and F1

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 06 | June - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

score. The total ResNet model parameters are 23,595,908, but the only trainable parameters in the dense layer are 8,196 . This approach leverages transfer learning to handle tasks with limited labelled data.[18]

Our experimental setup was equipped with a robust 16GB RAM to ensure efficient data processing and multitasking capabilities. Complementing this, the system featured an NVIDIA RTX 3060 graphics card, which boasts 6GB of VRAM.

Table 5. Classification Report table of all Models

Serial no	**Model**	**Accuracy**	**Precision**	**Recall**	**F1_score**
1	ResNet50	99.11	98.9	98.9	99.1
2	VGG16	95.73	95.7	96.1	95.5
3	EfficientNetV2B3	98.61	98.6	98.5	98.4

Accuracy and loss curves are the major aspects to understand the performance of the model as it provides insights about training and how the model improves its performance over time. Accuracy curve indicates how the model makes accurate predictions on training set during the training phase. On the other hand, Loss curve indicates the error measurement between the predicted output and actual output.

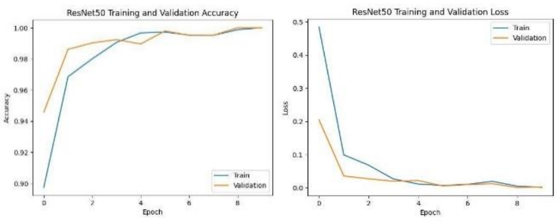

Fig 6 Accuracy and Loss Curves of ResNet50.

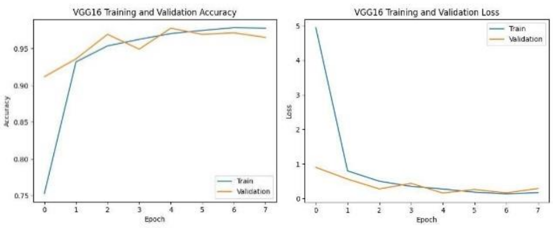

Fig 7 Accuracy and Loss Curves of VGG16.

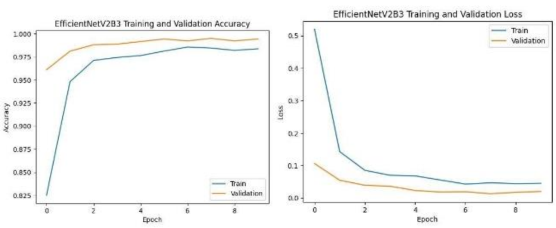

Fig 8 Accuracy and Loss Curves of EfficientNetV2B3..

Confusion matrix provides a comprehensive summary of how well a model performs on a set of test data by comparing predicted values against actual values. As shown below in Figure 12, 13, 14 these are the confusion matrix for our proposed work using the models ResNet50, VGG16, EfficientNetV2B3 respectively.

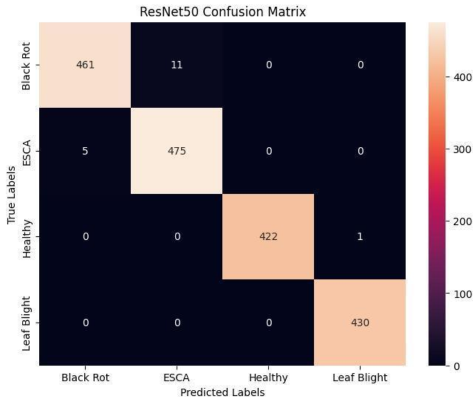

Fig 12 Confusion Matrix of ResNet50

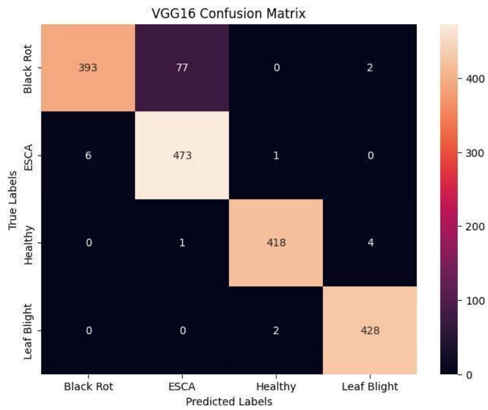

Fig 13 Confusion Matrix of VGG16.

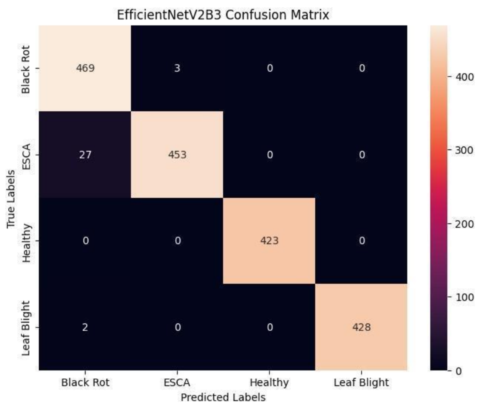

Fig 14 Confusion Matrix of EfficientNetV2B3.

VIII. DISCUSSION

It can be observed that ResNet50 is more efficient than VGG16 and EfficientNetV2B3, as shown in Fig. 15. In our study, we observed that the ResNet50 model achieved the highest accuracy of 99.11% in detecting grape leaf diseases. This superior performance can be attributed to several factors inherent to the ResNet50 architecture and our specific dataset. Firstly, the residual connections in ResNet50 help

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 06 | June - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

mitigate the vanishing gradient problem, which is crucial for learning from a deep network with 50 layer.[29][30] The vanishing gradient problem is a challenge encountered when training deep neural networks, particularly recurrent neural networks, using gradient-based learning methods and backpropagation. During the training process, each weight of the neural network is updated proportionally to the partial derivative of the error function with respect to the current weight. However, as the network depth increases, the gradients can become very small, exponentially decreasing as the backpropagation algorithm progresses from the output layer back to the input layer. This results in the earlier layers of the network learning very slowly or not at all, which can prevent the network from further training effectively.[31]

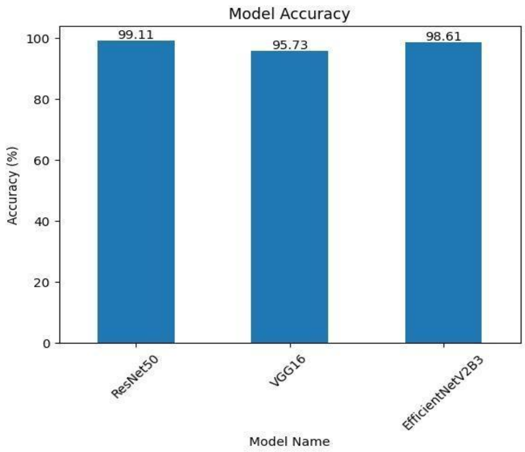

Fig 15 Accuracy Bar Graph of all Models

Secondly, the bottleneck design in ResNet50's building blocks reduces the complexity of the model, allowing for faster training without a loss in performance.[29][30] Additionally, the application of sparsification techniques such as pruning and quantization has shown to enhance the efficiency of ResNet50 models, making them more suitable for real-time applications.[29][30]

Resnet50 and EfficientNetV2B3 both skip connection to solve the vanishing gradient problem but the main difference between them is the number of layers in the ResNet50 are less than the number of layers in EfficientNetV2B3 so there is less complexity that's why it outperforms the other models.

Furthermore, the success of transfer learning with ResNet50 suggests that the model is capable of adapting pre-learned patterns to the specific task of grape leaf disease detection.[29][30] The nature of our dataset, which contains distinct features of grape leaf diseases, may also have been better captured by the ResNet50 architecture, leading to higher accuracy. Lastly, the optimization of training parameters and the use of data augmentation could have contributed to the enhanced performance of ResNet50 in our experiments.

While our results are promising, it is important to consider the limitations of our study, including the potential for overfitting and the need for further validation on larger, more diverse datasets. Future research should aim to explore these aspects to confirm the generalizability of our findings.

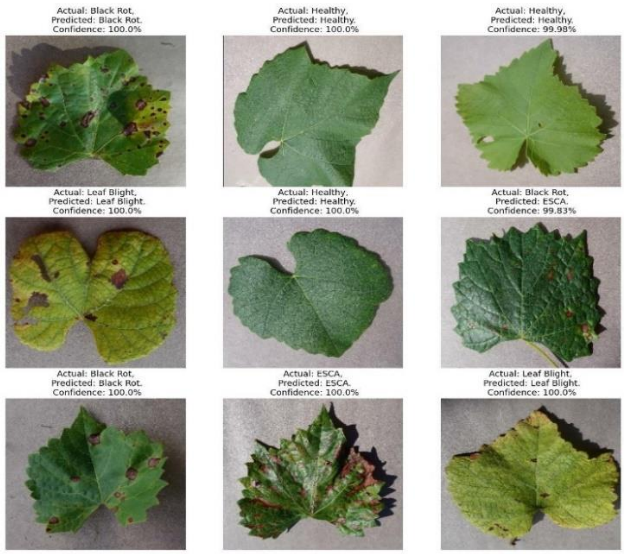

Fig 16 Prediction and Confidence

IX. Conclusion

Based on our study, it can be concluded that ResNet50 is the best model for detecting grape leaf diseases, with an accuracy of 99.11%. The other two models, VGG16 and EfficientNetV2B3, had accuracies of 95.73% and 98.61%, respectively. The architecture of ResNet50 was found to be superior to the other two models. The dataset that is used in this work contain few numbers of classes and also the size of the dataset was not big enough for our future work. We are looking forward to collecting more amount of data and also increase number of classes and check if the same model i.e. ResNet50 perform well with this dataset.

References

[1] Math, R. M., & Dharwadkar, N. V. (2022). Early detection and identification of grape diseases using convolutional neural networks. *Journal of Plant Diseases and Protection*, *129*(3), 521-532. https://link.springer.com/article/10.1007/s41348-022-00589-5

[2] Indian Grapes -Significant Growth & Global Trusted Supply by Karthikeyan D, Director at Navadhane Agro Foods Pvt. Ltd https://www.linkedin.com/pulse/indian-grapes-significant-growth-global-trusted-dhanushkodi

[3] Xie, X., Ma, Y., Liu, B., He, J., Li, S., & Wang, H. (2020). A deep-learning-based real-time detector for grape leaf diseases using improved convolutional neural networks. *Frontiers in plant science*, *11*, 751. https://www.frontiersin.org/articles/10.3389/fpls.2020.00751/full

[4] A Review on Grape Disease Detection and Classification using Image Processing by Niharika S1, N V Sahana2, Prof. Girija J 3 https://www.irjet.net/archives/V8/i9/IRJET-V8I966.pdf

[5] Grape Production in India by KhetiGaadi https://khetigaadi.com/blog/grape-production-in-india/

[6] Introduction to Convolution Neural Network https://www.geeksforgeeks.org/introduction-convolution-neural-network/
[7] Es-saady, Y., El Massi, I., El Yassa, M., Mammass, D., and Benazoun, A. (2016). "Automatic recognition of plant leaves diseases based on serial combination of two SVM classifiers," in *Proceedings of the International Conference on Electrical and Information Technologies*, Tangiers, 561–566.
[8] Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis. Comput. Electron. Agric. 145, 311–318. doi: 10.1016/j.compag.2018.01.009.
[9] M. Agarwal, S. K. Gupta and K. K. Biswas, "Grape Disease Identification Using Convolution Neural Network," *2019 23rd International Computer Science and Engineering Conference (ICSEC)*, Phuket, Thailand, 2019, pp. 224-229, doi: 10.1109/ICSEC47112.2019.8974752. https://ieeexplore.ieee.org/document/8974752
[10] Jain, B., & Periyasamy, S. (2022). Grapes disease detection using transfer learning. *arXiv preprint arXiv:2208.07647*. https://arxiv.org/pdf/2208.07647
[11] Patel, H., Palande, V., Bhat, A., Behara, A., & Gothi, R. (2022). Grape Leaf Diesease Detection Using CNN and RNN. *Available at SSRN 4108967*. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4108967
[12] Orhan G. Yalçın (2020). 4 Pre-Trained CNN Models to Use for Computer Vision with Transfer Learning.https://towardsdatascience.com/4-pre-trained-cnn-models-to-use-for-computer-vision-with-transfer-learning-885cb1b2dfc.
[13] Albawi, S., Mohammed, T.A., Al-Zawi, S., 2017. August). Understanding of a convolutional neural network. 2017 international conference on engineering and technology (ICET). Ieee, pp. 1–6.
[14] Mingxing Tan, Quoc V. Le (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. https://doi.org/10.48550/arXiv.1905.11946.
[15] Sukhpreet Kaur & Nilima Kulkarni (2023) FERFM: An Enhanced Facial Emotion Recognition System Using Fine-tuned MobileNetV2 Architecture, IETE Journal of Research, DOI: 10.1080/03772063.2023.2202158
[16] What is image augmentation and how it can improve the performance of deep neural networks https://albumentations.ai/docs/introduction/image_augmentation/
[17] Lau, S., & Lau, S. (2017). Image Augmentation for Deep Learning. *Towards Data Science*. https://towardsdatascience.com/image-augmentation-for-deep-learning-histogram-equalization-a71387f609b2
[18] Understanding ResNet-50 in Depth: Architecture, Skip Connections, and Advantages Over Other Networks by wisdomml https://wisdomml.in/understanding-resnet-50-in-depth-architecture-skip-connections-and-advantages-over-other-networks/
[19] Jingpeng Zhai, Weiran Shen, Ishwar Singh, Tom Wanyama, Zhen Gao, A Review of the Evolution of Deep Learning Architectures and Comparison of their Performances for Histopathologic Cancer Detection, Procedia Manufacturing, Volume 46, 2020, Pages 683-689,
[20] Resnet50 by Meghana https://blog.aiensured.com/resnet50/
[21] ResNet-50: The Basics and a Quick Tutorial https://datagen.tech/guides/computer-vision/resnet-50/
[22] Cross Entropy vs. Sparse Cross Entropy: When to use one over the other https://stats.stackexchange.com/questions/326065/cross-entropy-vs-sparse-cross-entropy-when-to-use-one-over-the-other
[23] What is the Mathematical formula for sparse categorical cross entropy loss? https://stackoverflow.com/questions/73436424/what-is-the-mathematical-formula-for-sparse-categorical-cross-entropy-loss
[24] Computes sparse categorical cross-entropy loss. https://keras.posit.co/reference/op_sparse_categorical_crossentropy.html
[25] Intuition of Adam Optimizer https://www.geeksforgeeks.org/intuition-of-adam-optimizer/
[26] What is Adam Optimizers https://www.geeksforgeeks.org/intuition-of-adam-optimizer/
[27] Adam Optimizer Simplified for Beginners in ML https://www.projectpro.io/article/adam-optimizer/986
[28] What is the Adam Optimizer and How is It Used in Machine Learning https://www.aiplusinfo.com/blog/what-is-the-adam-optimizer-and-how-is-it-used-in-machine-learning/
[29] ResNet-50: The Basics and a Quick Tutorial https://datagen.tech/guides/computer-vision/resnet-50/
[30] ResNet-50 on CPUs: Sparsifying for Better Performance on CPUs by neuralmagic https://neuralmagic.com/blog/benchmark-resnet50-with-deepsparse/
[31] Vanishing gradient problem https://en.wikipedia.org/wiki/Vanishing_gradient_problem