Plant Disease Detection Using DeepNets and Ensemble Technique

1st Saanidhya Vats
Department of Computer Science and Engineering
IIIT-Bhubaneswar
Bhubaneswar, India
saanidhyavats@gmail.com

2nd VNAD Chivukula
Department of Computer Engineering
IIIT-Bhubaneswar
Bhubaneswar, India
adityadatta321@gmail.com

***Abstract*—Growing food security is a significant concern in the modern world. With the world's population expected to increase by two billion in the next three decades, there is a necessity to increase food production to support the growing population. In recent years, the increase in global food production has slowed, too slow to keep up with population growth. The factors directly affecting global food production are drought and plant diseases. Detection of these diseases through manual inspection is time taking and involves a factor of human error. In this paper, we focus on the problem of detecting plant diseases accurately at an early stage to increase food production. Machine learning and deep learning-based models have the potential to solve this issue by detecting plant diseases quickly and accurately. In this work, we first analyze the performance of pre-trained deep learning models on an expanded version of the standard PlantVillage dataset and then propose an ensemble of deep learning models. The proposed ensemble model outperforms all the existing deep learning models and achieves a maximum accuracy of 99.61%.**

***Keywords*—Plant disease detection, Deep learning, Convolution neural network, Ensemble technique**

I. Introduction

Food security is one of the most pressing issues confronting humanity today. With more deaths occurring each year, it is quickly becoming a significant issue that requires attention. Food security is affected by various reasons but a prominent one is decreased crop yield [1], which is brought by pests attacking the crops. The effect of plant diseases on the global death toll is explained in [2]. The global hunger index [3] explains food security and trends. Along with hunger-related mortality, plant diseases also contribute to increased food prices. Further, new plant disease strains are emerging each year and pose a severe threat to human existence on a global scale. For developing countries, agriculture is the backbone of their economies. Plant disease-related crop failure can have a negative impact on these nations' economies.

Deep learning methods [4] are becoming very popular recently, especially in the context of classification problems [5]. Convolution neural networks [6] (CNN) have attracted interest because of their ability to extract features automatically from input data and achieve better accuracy. CNN's are time-consuming and demand more computational power while training. However, the trained models can classify images very quickly. With the above considerations, this paper aims to solve the problem of detecting disease in plants with CNNs.

To achieve this, pre-trained nets MobileNet-V2 [7], Inception-V3 [8], ResNet-50 [9], Xception [10], EfficientNet-B3 [11] and Inception-ResNet-V2 [12] have been chosen, which are individually trained on a modified version of the PlantVillage dataset [13]. The test accuracy of the models have been compared and the top 3 models with the highest test accuracy have been selected for the deep learning based ensemble technique.

II. Literature Review

Plant disease detection has been the subject of extensive investigation. The PlantVillage dataset [14] is often used for testing state-of-the-art deep learning architectures for plant disease detection. Some works in the literature have considered only a few categories from the dataset while others have used the entire dataset when constructing their architectures. Works such as disease detection in sunflowers [15] and disease detection in tomato leaves [16] focused on specific disease detection using deepnets. In [15], a hybrid model was proposed which uses ensemble stacking and a combination of VGG-16 and MobileNet, resulting in an accuracy of 89.2%. The performances of AlexNet and SqueezeNet were analyzed in [16], resulting in accuracies of 95.65% and 95.3% respectively. Similarly, model presented in [17] focuses on the identification of plant diseases in potato, pepper and tomato plants, achieving an accuracy of 98.3%, 98.5% and 95% respectively. Further, there has been a hybrid model [18] implementation using a convolutional neural network and convolutional auto-encoder with a training accuracy of 99.35% and testing accuracy of 98.38% on identifying disease in peach plants.

In [19], a DCNN-19 model is presented to recognize and categorize damaged leaves in 14 different crops with 97.22% accuracy rate. The works presented in [20] and [21] used

978-1-6654-8701-6/22/$31.00 ©2022 IEEE

NasNet and ResNet-152V2 respectively for classifying plants as diseased/healthy, achieving an accuracy of 93.82% and 94.32% respectively. Further, [22] created their own dataset and observed an increase in accuracy and a decrease in loss when data augmentation was applied to the dataset. In this work, ResNet model achieved the highest accuracy of 98.96%.

Some works focused on classifying every category in the PlantVillage dataset like [23] which used SVM classifier on the features extracted with pre-trained models like ResNet-50, GoogleNet and VGG-16, producing a top accuracy of 97.82%. Recently, there has been an effort to propose a lightweight model [24], a hybrid of VGG blocks and Inception modules, that achieves a testing accuracy of 99.16%. Further, [25] used AlexNet and GoogleNet architectures where GoogleNet achieves a highest accuracy of 99.35%. Furthermore, a novel deep learning model presented in [26] based on the Inception layer and Residual connection achieves an accuracy of 99.39% on the dataset.

III. Dataset

In this paper, a modified version of the PlantVillage dataset [13] has been used. The dataset is created using offline augmentation of the original dataset [14]. It consists of 87000 RGB images of healthy and diseased crop leaves categorized into 38 classes. This dataset contains 14 crops and 26 diseases readily available on the Kaggle platform. The training-to-test ratio is 80:20 and considers an image size of 256x256x3 for all the models. To make the model robust and to avoid overfitting, different variations are introduced by varying the factors like rotation, zoom, shear, width and height. Fig 1 shows images from each class of the dataset.

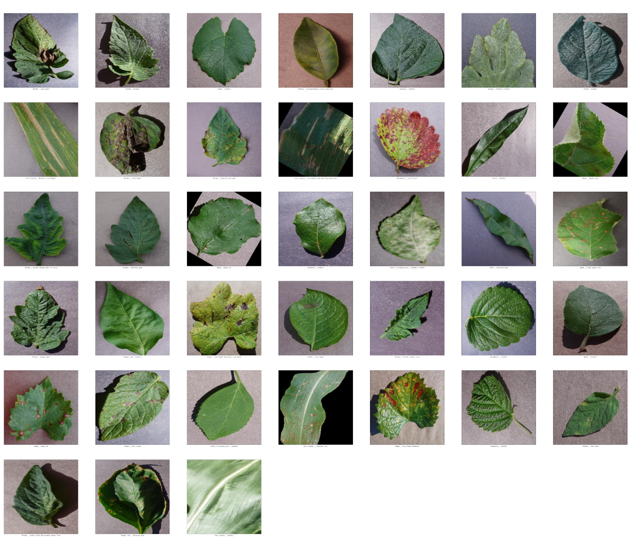

Fig. 1. Image from each class of the dataset

IV. Method Used

All the models discussed in the paper are developed using Tensorflow Keras library. They are trained and tested in Kaggle environment that offers 13GB of memory and a NVIDIA Tesla P100 GPU. To prevent the problem of overfitting, layers like Global Average Pooling and Dropout have been added to the pre-trained nets. Each network is trained using Adam Optimizer for 50 epochs. Initially, outcomes produced by the six most prominent architectures are analyzed and the top three models with the highest test accuracy are selected for the ensemble technique.

A. MobileNet-V2

MobileNet [27] is a deep learning model that uses Depthwise and Pointwise convolutions. The Depthwise convolutions add more information through layers while the Pointwise convolutions in between help to reduce the number of parameters and make the model computationally efficient. This model is developed for computer vision applications that are not complex and contains fewer features. MobileNet-V2 [7], an improved version of this design, adds two new features to the previous version's "depth-wise separable" block. The expansion layer, which is essentially a 1x1 convolution, is added first, followed by batch normalisation and ReLu activation whose goal is to project the influx of higher-dimensional data into data with fewer dimensions. This block is referred to as a "bottleneck" because the amount of data passing through is constrained. The second development compensates for the loss of this information by adding the block's input to its output through a residual connection. Compared to MobileNet-V1, this architecture is faster and more effective. Although MobileNet-V3 [28] architecture is available, this paper uses MobileNet-V2 to examine its performance on the enhanced PlantVillage dataset. The architecture produces an accuracy of 97.360% on training data and 97.826% on test data. Fig 2 represents the training graph.

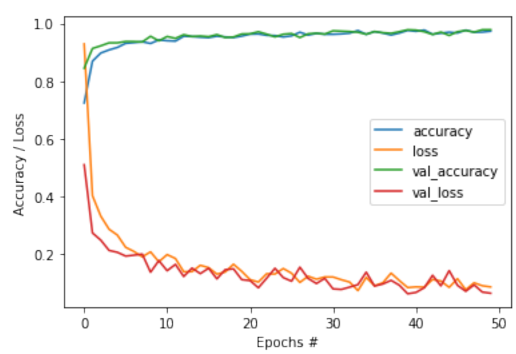

Fig. 2. Training history of MobileNet network

B. Inception-V3

The Inception module has a parallel execution of 1x1, 3x3 and 5x5 convolutions with a 3x3 max pooling. In the initial version [29], there is a single 1x1, 1x1 followed by 3x3, 1x1 followed by 5x5 convolutions and a 3x3 max pooling followed by 1x1 convolution. The purpose of 1x1 convolution is to reduce the number of parameters and make computation efficient. Inception-V2 introduces batch normalization in the architecture for better accuracy and optimization. Inception-V3 [8] introduces a noticeable reformulation to the Inception module with major changes being to split higher number convolutions into lower number convolutions. For example, a 5x5 convolution was replaced with a series of two 3x3 convolutions which reduces the computational complexity. To further reduce the computational complexity, each nxn filter is divided into a series of 1xn and nx1 filters. It is also noticed that efficient grid-size reduction, label smoothing and reducing the stride or removing pooling layer during initial Inception modules for low-resolution images yields better efficiency, reduces overfitting and increases accuracy respectively. Inception-V3 resulted in a training accuracy of 97.330% and test accuracy of 97.775%. The training graph is as shown in fig 3.

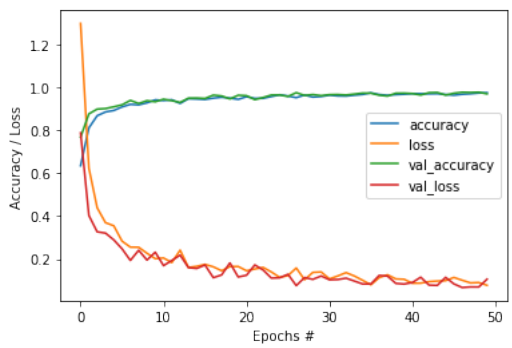

Fig. 3. Training history of Inception network

C. ResNet-50

Residual nets introduced in [9] operate on the concept of skip connections. By creating an alternate shortcut for the gradient to pass through, skip connections are used to solve the problem of vanishing gradients. Additionally, they give the model the ability to train an identity function that guarantees that the upper layers perform equally well with the lower levels. Initially, there were 34 layers and later on ResNet-50 [9] was introduced which has a greater number of layers and improved efficiency. The performance of ResNet-50 on the PlantVillage dataset is shown in fig 4 where it successfully

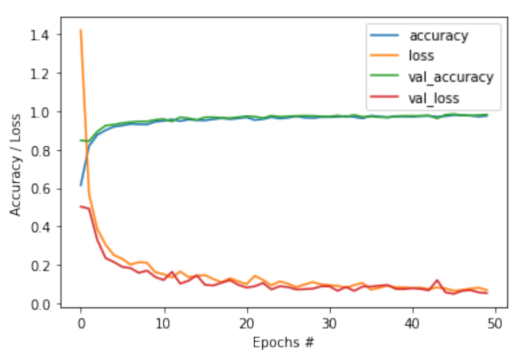

Fig. 4. Training history of ResNet-50

obtained a training accuracy of 97.750% and test accuracy of 98.395%.

D. Xception

Xception [10] is similar to the Inception network and is also known as an extreme version of Inception. It utilizes Depthwise Separable Convolutions which can be divided into Depthwise convolutions and Pointwise convolutions. The architecture is divided into 14 modules, each of which consists of a number of Pooling and Depthwise Separable Convolution layers. The 14 modules are divided into three groups, the entry flow, the middle flow and the exit flow. Each group has four, eight and two modules respectively. All the Depthwise Separable Convolution layers in the architecture use filter size of 3x3, stride 1 and "same" padding. This network has almost same number of parameters as Inception-V3 but outperforms it by a small margin on the ImageNet dataset. Its better performance can be attributed to its architecture. Xception has achieved an accuracy of 97.990% on training data and 98.515% on test data. Fig 5 represents the performance of Xception with the number of epochs.

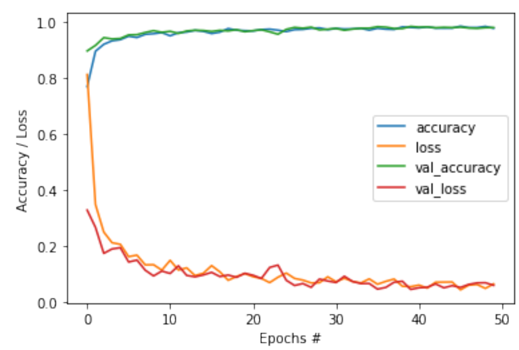

Fig. 5. Training history of Xception network

E. EfficientNet-B3

The basis of EfficientNet [11] is integrated scaling of all dimensions. The network breadth, depth and resolution are uniformly scaled using a compound coefficient. The idea behind building the architecture was to have a baseline network and then scale it along different dimensions. The baseline network is obtained by neural architecture search that utilizes mobile inverted bottleneck convolution similar to MobileNet-V2. This baseline model is scaled up to obtain a family of models called EfficientNets. EfficientNets have achieved higher accuracy than the other convolution networks on the ImageNet dataset. This paper uses the EfficientNet-B3 model. The model has obtained an accuracy of 97.730% on training data and 98.560% on test data. The performance is shown in fig 6.

F. Inception-ResNet-V2

This is a model inspired by both Inception and ResNet modules. Multiple convolutional filters of different sizes are combined by residual connections in the Inception-ResNet

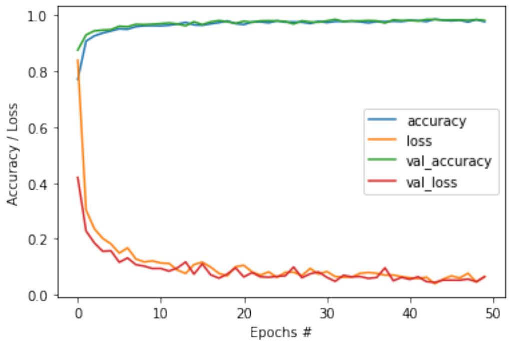

Fig. 6. Training history of Efficient network

block. A clear idea of each module can be studied in the paper [12] which discusses about Inception-V4 and how Inception-ResNet is developed from it. This paper uses Inception-ResNet-V2 which obtained an accuracy of 98.640% on training data and 98.646% on test data. The performance is shown in fig 7. Comparison of all the pre-trained deepnets is shown in table I.

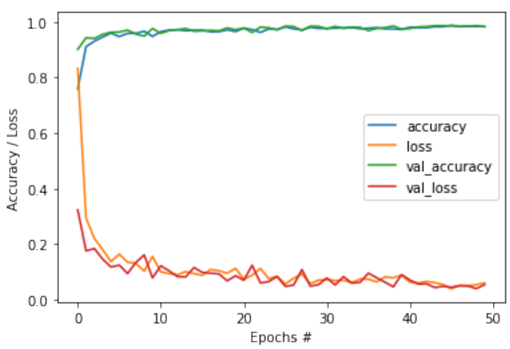

Fig. 7. Training history of Inception-ResNet

TABLE I
COMPARISON OF THE ACCURACY OF DIFFERENT MODELS

DeepNet	Training Accuracy	Test Accuracy
MobileNet-V2	97.360%	97.826%
Inception-V3	97.330%	97.775%
ResNet-50	97.750%	98.395%
Xception	97.990%	**98.515%**
EfficientNet-B3	97.730%	**98.560%**
Inception-ResNet-V2	98.640%	**98.646%**

G. Proposed Model

The model involves the use of ensemble technique. This technique is known for producing more robust results with greater accuracy due to the combined prediction powers of its ‘ensemble members‘. The variance, bias and noise issues that single models frequently experience are mitigated using the ensemble method. Most neural networks have significant variance or a propensity for overfitting. The incorporation of bias through ensemble modelling of neural networks will improve the bias-variance trade-off of the overall model. Pre-trained networks selected for this technique are Inception-ResNet-V2, EfficientNet and Xception as they have the highest test accuracy. Fig 8,9,10 represents their respective block diagrams.

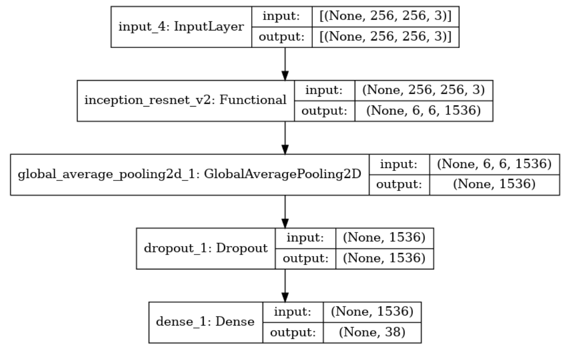

Fig. 8. Block diagram of Inception-ResNet with additional layers

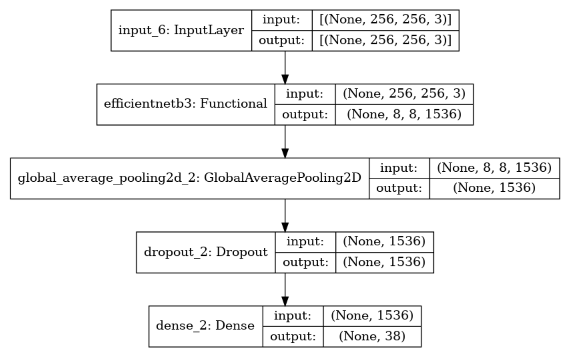

Fig. 9. Block diagram of EfficientNet with additional layers

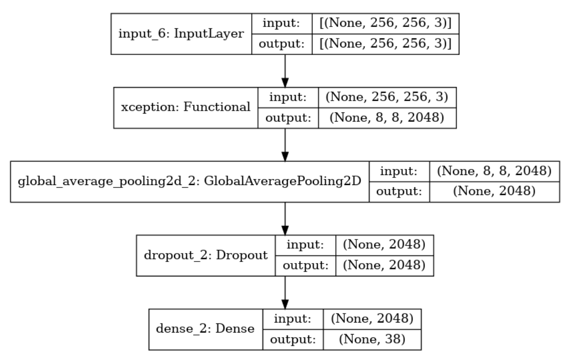

Fig. 10. Block diagram of Xception with additional layers

In the ensemble technique, the average of the results produced by these three pre-trained networks is calculated and considered as the output. The proposed model is tested 15 times resulting in a mean accuracy of 99.50% and a maximum accuracy of 99.61%. It has a total of 86.17 million parameters and takes 190 seconds to test 17572 images. Fig 11 represents the block diagram of the proposed ensemble model.

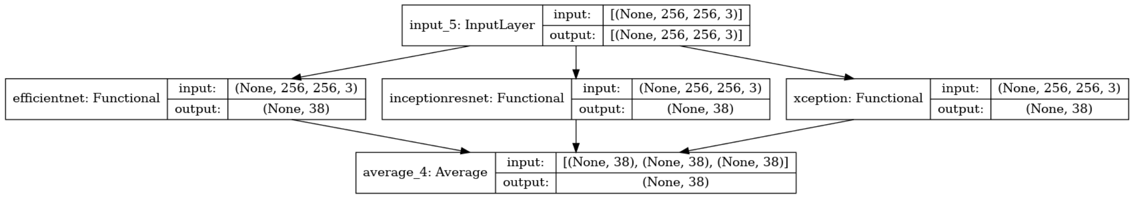

Fig. 11. Block diagram of the proposed model

CONCLUSION

Computer Vision is an exciting area to explore. With the advent of cutting-edge deep learning algorithms, the applications such as image generation, self driving cars and many more are becoming a reality. With such powerful techniques, one can always find an implementation that can benefit humanity. Early detection of plant's diseases can help us save crop produce which in turn will reduce food security issues. In this paper, we have trained 6 standard pre-trained networks on the modified PlantVillage dataset. The top 3 performing models Inception-ResNet-V2, EfficientNet-B3 and Xception with 98.646%, 98.560% and 98.515% respective test accuracy are chosen for creating an ensemble model. The deep learning based ensemble model has produced a maximum accuracy of 99.61%, outperforming the other state-of-the-art algorithms. With faster GPUs and CPUs being affordable day by day, the proposed model shows a promising future for its implementation on various species of crops.

REFERENCES

1. Cerda, R. *et al.* Primary and secondary yield losses caused by pests and diseases: Assessment and modeling in coffee. *PloS one* **12,** e0169133 (2017).
2. He, S. & Krainer, K. M. C. Pandemics of people and plants: which is the greater threat to food security? *Molecular plant* **13,** 933–934 (2020).
3. Von Grebmer, K. *et al.* Global Hunger Index: Hunger and Food Systems in Conflict Settings 2021 (2021).
4. Goodfellow, I., Bengio, Y. & Courville, A. *Deep learning* (MIT press, 2016).
5. Krizhevsky, A., Sutskever, I. & Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Communications of the ACM* **60,** 84–90 (2017).
6. LeCun, Y., Bengio, Y., *et al.* Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks* **3361,** 1995 (1995).
7. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A. & Chen, L.-C. *Mobilenetv2: Inverted residuals and linear bottlenecks* in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2018), 4510–4520.
8. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. & Wojna, Z. *Rethinking the inception architecture for computer vision* in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), 2818–2826.
9. He, K., Zhang, X., Ren, S. & Sun, J. *Deep residual learning for image recognition* in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), 770–778.
10. Chollet, F. *Xception: Deep learning with depthwise separable convolutions* in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2017), 1251–1258.
11. Tan, M. & Le, Q. *Efficientnet: Rethinking model scaling for convolutional neural networks* in *International conference on machine learning* (2019), 6105–6114.
12. Szegedy, C., Ioffe, S., Vanhoucke, V. & Alemi, A. A. *Inception-v4, inception-resnet and the impact of residual connections on learning* in *Thirty-first AAAI conference on artificial intelligence* (2017).
13. Bhattarai, S. *New Plant Diseases Dataset, Version 2* https://www.kaggle.com/datasets/vipoooool/new-plant-diseases-dataset(accessed Semptember 3 , 2022). (2018).
14. Hughes, D., Salathé, M., *et al.* An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint arXiv:1511.08060* (2015).
15. Sirohi, A. & Malik, A. *A Hybrid Model for the Classification of Sunflower Diseases Using Deep Learning* in *2021 2nd International Conference on Intelligent Engineering and Management (ICIEM)* (2021), 58–62.
16. Durmuş, H., Güneş, E. O. & Kırcı, M. *Disease detection on the leaves of the tomato plants by using deep learning* in *2017 6th International conference on agro-geoinformatics* (2017), 1–5.
17. Lakshmanarao, A., Babu, M. R. & Kiran, T. S. R. *Plant Disease Prediction and classification using Deep Learning ConvNets* in *2021 International Conference on Artificial Intelligence and Machine Vision (AIMV)* (2021), 1–6.
18. Bedi, P. & Gole, P. Plant disease detection using hybrid model based on convolutional autoencoder and convolutional neural network. *Artificial Intelligence in Agriculture* **5,** 90–101 (2021).
19. Nagaraju, M. & Chawla, P. *Plant Disease Classification using DCNN-19 Convolutional Neural Networks* in *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)* (2021), 1–6.
20. Adedoja, A., Owolawi, P. A. & Mapayi, T. *Deep learning based on nasnet for plant disease recognition using leave*

images in *2019 international conference on advances in big data, computing and data communication systems (icABCD)* (2019), 1–5.

21. Kothawade, P., Varma, S. & Gogate, U. *Plant disease detection and management using deep learning approach* in *7th International Conference on Computing in Engineering Technology (ICCET 2022)* **2022** (2022), 30–35.
22. Marzougui, F., Elleuch, M. & Kherallah, M. *A Deep CNN Approach for Plant Disease Detection* in *2020 21st International Arab Conference on Information Technology (ACIT)* (2020), 1–6.
23. Mohameth, F., Bingcai, C. & Sada, K. A. Plant disease detection with deep learning and feature extraction using plant village. *Journal of Computer and Communications* **8,** 10–22 (2020).
24. Thakur, P. S., Sheorey, T. & Ojha, A. VGG-ICNN: A Lightweight CNN model for crop disease identification. *Multimedia Tools and Applications,* 1–24 (2022).
25. Mohanty, S. P., Hughes, D. P. & Salathé, M. Using deep learning for image-based plant disease detection. *Frontiers in plant science* **7,** 1419 (2016).
26. Hassan, S. M. & Maji, A. K. Plant Disease Identification Using a Novel Convolutional Neural Network. *IEEE Access* **10,** 5390–5401 (2022).
27. Howard, A. G. *et al.* Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861* (2017).
28. Howard, A. *et al. Searching for mobilenetv3* in *Proceedings of the IEEE/CVF international conference on computer vision* (2019), 1314–1324.
29. Szegedy, C. *et al. Going deeper with convolutions* in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), 1–9.
30. Dong, S. *et al.* Deep ensemble CNN method based on sample expansion for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing* **60,** 1–15 (2022).
31. Xiong, Y., Liang, L., Wang, L., She, J. & Wu, M. Identification of cash crop diseases using automatic image segmentation algorithm and deep learning with expanded dataset. *Computers and Electronics in Agriculture* **177,** 105712 (2020).
32. Kumar, A., Razi, R., Singh, A. & Das, H. *Res-vgg: a novel model for plant disease detection by fusing vgg16 and resnet models* in *International Conference on Machine Learning, Image Processing, Network Security and Data Sciences* (2020), 383–400.
33. Wallelign, S., Polceanu, M. & Buche, C. *Soybean plant disease identification using convolutional neural network.* in *FLAIRS conference* (2018), 146–151.
34. Yilma, G., Belay, S., Qin, Z., Gedamu, K. & Ayalew, M. *Plant disease classification using two pathway encoder GAN data generation* in *2020 17th international computer conference on wavelet active media technology and information processing (ICCWAMTIP)* (2020), 67–72.
35. Kulkarni, P. *et al. Plant Disease Detection Using Image Processing and Machine Learning* June 2021.
36. Simonyan, K. & Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).