Journal of Current Science and Technology, January-April 2023 Vol. 13 No. 1, pp. 23-35

ISSN 2630-0656 (Online)

Cite this article: Bhola, A., Verma, S., & Kumar, P. (2023, January). A comparative analysis of deep learning models for cucumber disease classification using transfer learning. *Journal of Current Science and Technology, 13*(1), 23-35. DOI:

Journal of Current Science and Technology

Journal homepage: https://jcst.rsu.ac.th

A comparative analysis of deep learning models for cucumber disease classification using transfer learning

Amit Bhola*, Sahil Verma, and Prabhat Kumar

Department of Computer Science and Engineering,
National Institute of Technology Patna, Bihar, India - 800005

*Corresponding author; E-mail: amitb.phd19.cs@nitp.ac.in

Received 2 May 2022; Revised 27 September 2022; Accepted 19 October 2022;
Published online 29 January 2023

Abstract

Increasing agricultural productivity continues to be a major challenge for society due to the rapid growth of the global human population and economic prosperity. However, improving agricultural productivity requires proper identification and minimization of diseases that degrade both the quality and quantity of the crops. The scientific community has stressed that the use of recent technologies such as deep learning, the internet of things, computer vision, etc. are vital to address various challenges in the agriculture sector. Furthermore, the use of computer vision to automatically identify diseases is growing in popularity. This paper provides a comparative analysis of six pre-trained deep learning models, namely VGG16, VGG19, ResNet50, ResNet101, InceptionV3, and Xception, for disease detection in cucumber plants. The pre-trained models are fine-tuned using transfer learning and evaluated using different metrics such as training accuracy, testing accuracy, and the number of epochs. The results obtained demonstrate that VGG16, despite being the smallest model in terms of the number of layers, outperforms the rest of the models in all of the evaluation metrics. The VGG16 models obtain testing accuracy of 98% and training accuracy of 99.91% while being trained for 8 epochs. In addition, it is observed that models with a larger number of layers, such as ResNet50 and ResNet101, exhibit fluctuations in accuracy while training due to the execution of fairly large models on a comparatively small dataset. However, InceptionV3 and Xception, despite having a greater number of layers, perform better than ResNet models due to the presence of Inception modules which are better equipped to detect different-sized targets. The findings of this study may be utilized to optimize the best-performing models for disease classification in other plants, and the fine-tuned VGG16 model can be integrated with mobile devices for real-time disease classification.

Keywords: *cucumber disease classification; deep learning; digital agriculture; plant disease; transfer learning.*

1. Introduction

Agriculture is considered one of the primary pillars that provides support to human existence. Apart from being the primary source of food supply, agriculture also contributes to the global economy and provides ample opportunity for employment. As this report suggests, 41.49% of the workforce in India was employed in agriculture in 2020 (Trading Economics, n.d.). Furthermore, agriculture shares rose by 2.1% in India's gross domestic product (GDP) for the year 2020–21 (Shagun, 2021). However, the rise in global population has escalated the demand of food supply, thus exerting pressure to increase agricultural productivity. Modern techniques such as machine learning (Kumar, Kumar, & Palaparhy, 2021; Nagasubramanian et al., 2021), deep learning (Bosilj, Aptoula, Duckett, & Cielniak, 2020; Jin et al., 2020; Naveen, & Sivakumar, 2021) and internet of things (IoT) (Sinha, Shrivastava, & Kumar,

2019) have proven to be a boon for the agriculture industry. Not only do they possess the potential to increase productivity, but also ensure the sustainability of the environment by enhancing the conventional methods used by farmers.

However, plant diseases remain one of the major challenges faced by farmers. Plant diseases degrade both the quality and quantity of the plants (Syed-Ab-Rahman, Hesamian, & Prasad, 2022). Early detection of such diseases can significantly help control damage to plants (Suwanagul, Kokaew, & Suwanagul, 2013). Conventional techniques used for the detection of these diseases, such as manually by the human eye, are time-consuming when carried out on a large scale. Convolution neural network (CNN)-based deep learning models can be used for the detection of these diseases at a high success rate in comparatively lesser time. These models can decrease manual efforts by covering large quantities of plant disease detection in a short amount of time (Moawed, 2016). The introduction of transfer learning has decreased the time required for the development of deep learning models. Transfer learning enables deep learning models to use the existing knowledge obtained by the models while being trained on a task.

In this paper, several state-of-the-art CNN models, namely VGG16 (Simonyan & Zisserman, 2015), VGG19 (Simonyan & Zisserman, 2015), ResNet50 (He, Zhang, Ren, & Sun, 2016), and ResNet101 (He et al., 2016), are fine-tuned with the use of transfer learning for disease classification in cucumber plants. The rest of the paper is organized as follows: the remaining part of Section 1 provided details about related works carried out for disease classification in different plants and provides a glimpse of the pre-trained models used in the proposed work. Section 2 highlights the objective of the research work. Section 3 describes the proposed approach, dataset, and experimental setup used in the work. Section 4 presents the results obtained and their comparison for different models used in this study. Finally, Section 5 concludes the paper.

1.1 Related works

Various research work has been carried out on plant disease classification using image data, where CNNs were the prime focus of the studies. However, in recent years, there has been a paradigm shift from developing CNN models from scratch to fine-tuning the existing state-of-the-art convolutional networks with the use of transfer learning (Becherer, Pecarina, Nykl, & Hopkinson, 2019). In Ganatra and Patel (2020), the authors fine-tuned pre-trained models like VGG16, InceptionV4, ResNet101, and ResNet50 for the classification of plant diseases found in the PlantVillage dataset. The results obtained illustrated that the ResNet101 model outperformed the rest of the pre-trained models with a test accuracy of 99.73%. Atila, Ucar, Akyol, and Ucar (2021) proposed the EfficientNet model for the classification of diseases found in PlantVillage dataset and obtained 99.97% accuracy with a precision of 99.39%.

In Saleem, Potgieter, and Arif (2020), the authors analyzed existing state-of-the-art pre-trained models by training the models with different optimizers from the PlantVillage dataset. The Xception model trained with adam optimizer obtained the highest validation accuracy of 99.81%. In Zeng, Ma, Cheng, Zhou, and Pang (2020), the authors used the InceptionV3 model to detect the susceptibility of citrus fruits to diseases with an accuracy of 74.38%. Other authors further employed the DC-GAN model (Ma, Shuai, Ran, Liu, & Ye, 2020) to augment the dataset and obtained 92.60% accuracy.

In Singh, Chouhan, Jain, and Jain (2019), the authors proposed a multi-layered CNN architecture to detect fungal diseases in mango with an accuracy of 97.13%. Zhou, Zhou, Xing, and Song (2021) proposed a restructured residual dense network to identify tomato plant diseases with an accuracy of 95%. In Bao, Huang, Hu, and Liang (2021), the authors proposed a deep neural network model based on GoogLeNet (Szegedy et al., 2015) for the classification of diseases in maize plants and obtained an accuracy of 98.9%.

In Bhatt, Sarangi, Shivhare, Singh, and Pappula (2019), the authors used an ensemble of Adaptive Boosting (Schapire, 2013) cascaded with a decision tree classifier to classify corn leaf images into four different categories. Features were extracted from the images using CNNs to identify diseases and classify the images accordingly with an accuracy of 98%. In Paymode and Malode (2022), the authors proposed a deep learning-based model for the classification of sick and healthy leaves for grapes and tomatoes. The proposed model attained an accuracy of 98.40% for grapes and 95.71% for tomatoes. In Hassan, Maji, Jasinski, Leonowicz, and Jasinska (2021), deep CNN models were implemented to identify diseases in the plant leaves of apple, corn, cherry, etc. The authors fine-tuned pre-trained models, namely InceptionV3, InceptionResNetV2, MobileNetV2, and EfficientNetB0, to obtain disease classification accuracies of 98.42%, 99.11%, 97.02%, and 99.56%, respectively. In Subramanian, Shanmugavadivel, and Nandhini (2022), the authors used a dataset of 18,888 photos of healthy and diseased leaves to classify three common maize leaf diseases using different pre-trained models, namely VGG16, ResNet50, InceptionV3, and Xception. The results obtained demonstrated that VGG16, InceptionV3, and Xception obtained accuracy rates of over 99%.

Table 1 Comparative analysis of related works

Ref.	Year	Dataset Description	Performance Comparison
Ganatra and Patel (2020)	2020	PlantVillage (Classes: 38 Images: 87,000)	Resnet101 / Resnet50 / VGG16 / InceptionV4 Train Acc. **99.87** / 99.85 / 83.43 / 99.65 Val. Acc. **99.80** / 99.76 / 82.30 / 98.30 Test Acc. **99.73** / 99.70 / 81.63 / 98.36
Atila et al. (2021)	2021	PlantVillage (Augmented Image Class: 61,486)	EN / AN / Resnet50 / VGG16 / InceptionV3 Avg Acc. **99.97** / 99.92 / 99.88 / 99.94 / 99.93 Avg Pre. **99.39** / 93.87 / 97.91 / 98.82 / 98.70
Saleem et al. (2020)	2020	PlantVillage (Classes: 38; Images: 54,306)	(Adam opt.) AN+GN / Improved GN / Xception Val. Acc. 98.57 / 99.04 / **99.81** F1-score 98.36 / 98.64 / **99.78**
Zeng et al. (2020)	2020	PlantVillage crowdAI	InceptionV3 / AN / Resnet34 Acc. **74.37** / 72.34 / 72.15 Train loss **0.5223** / 0.5814 / 0.9649
Singh et al. (2019)	2019	Self: Mango (Classes: 4; Images: 1070)	PSO / SVM / RBFNN / MCNN Acc. 88.39 / 92.75 / 94.20 / **97.13** MRR 11.61 / 7.25 / 5.80 / **2.87**
Zhou et al. (2020)	2021	AI Challenger Tomato Dataset	Deep CNN / ResNet50 / DenseNet121 / RRDN Test Acc. 93.21 / 88.49 / 91.96 / **95.00**
Bao et al. (2021)	2021	PlantVillage: Maize (Classes: 4; Images: 3852)	AN+SVM / LeNet / GN / MCNN / CNN Acc. 95.00 / 97.89 / **98.90** / 92.31 / 98.87
Bhatt et al. (2019)	2019	PlantVillage: Corn (Classes: 4; Images: 2000)	(Adaboost) Common Rust / Leaf Spot / Leaf Blight / Total Precision **98.00** / 96.00 / 97.00 / 97.00 Recall **98.00** / 95.00 / 98.00 / 98.00 F1-score **98.00** / 94.00 / 96.00 / 97.00
Paymode and Malode (2022)	2022	PlantVillage: Grapes, Tomatoes	VGG16 / DL-VGG / AN+VGG16 / DT-VGG / DCNN $Acc._{(grapes)}$ **98.40** / 97.53 / 97.50 / 91.83 / 88.46 $Acc._{(tomatoes)}$ **95.71** / 95.00 / 91.83 / 86.10 / 81.11
Hassan et al. (2021)	2021	PlantVillage (Classes: 38; Images: 54,305)	Inc.V3 / Inc.RN-V2 / MN-V2 / EfficientNetB0 Test Acc. 98.42 / 99.11 / 97.02 / **99.56**

Ref.	Year	Dataset Description	Performance Comparison				
Subramanian et al (2022)	2022	PlantVillage, Kaggle (Classes: 4; Images: 18,888)		VGG16	ResNet50	Inception	Xception
			Val. Acc.	**99.79**	93.14	99.40	99.75
			Test. Acc.	99.59	93.63	99.66	**99.93**
Fan et al. (2022)	2022	Public Apple Leaf Dataset (Images: 1821)		DenseNet	VGG16	VGG19	Inc. V3
			Val. Acc.	78.52	73.79	81.49	**91.28**

Abbreviations: AN: AlexNet, GN: GoogleNet, EN: EfficientNet, Opt.: Optimizer, Val.: Validation, MRR: Missing Report Rate, DL: Deep Learning, DT: Deep Transfer, RRDN: Restructured Residual Dense Network, Inc.: Inception, RN: ResNet, MN: MobileNet

In Fan et al. (2022), the authors proposed a feature-fusion based approach and enhanced InceptionV3 network to recognize diseased leaves in apple trees with an accuracy of 91.28%. In Dhaka et al. (2021), the authors presented an in-depth survey on the implementation of deep CNN models for disease identification in different plants. The survey focused on CNN architectures, datasets used, size of the datasets, and experimental results obtained by CNN models used on the datasets. In Kundu et al. (2021), the authors proposed an integrated framework of IoT and deep learning to automatically detect blast and rust diseases in pearl millet. The authors tested several benchmark models, namely Custom-Net, InceptionResNetV2, InceptionV3, ResNet50, VGG16, and VGG19, to identify the most suitable model. The experimental results revealed Custom-Net was the most accurate model with an accuracy of 98.78%. In Hussain et al. (2021), the authors proposed the ReviseNet model for object detection. The proposed model obtained mean absolute error (MAE) scores of 0.033, 0.029, and 0.036 on DUTS, HKU-IS, and ECSDD datasets, respectively. Table 1 highlights the comparative analysis of the proposed works carried out for disease detection in different plants.

1.2 Background

Transfer learning is a technique that exploits the knowledge obtained by state-of-the-art models from performing previous tasks and uses that knowledge as a starting point for training models for new tasks. Models like VGG16 (Simonyan & Zisserman, 2015) and ResNet50 (He et al., 2016) have been pre-trained on the ImageNet dataset repository (Deng et al., 2009), and the knowledge obtained by these models was transferred to new models which were then fine-tuned to obtain the desired result for the assigned task. For this paper, benchmark pre-trained models such as VGG16, VGG19 (Simonyan & Zisserman, 2015), ResNet50, ResNet101 (He et al., 2016), InceptionV3 (Szegedy et al., 2015), and Xception (Chollet, 2017) are used for disease classification in cucumber plants. A brief discussion about each of these pre-trained models is provided below.

1.2.1 VGG16 (Simonyan & Zisserman, 2015)

Visual Geometry Group (VGG) models were developed to increase the depth of classical CNNs for improving the model's performance. VGG16 is a CNN architecture that won the first prize in the ImageNet Challenge 2014. This model has a large network with approximately 138 million parameters and 16 layers (13 convolution layers and 3 fully connected layers). The convolution layers consist of repetitive cells with each cell containing a 3 × 3 convolution filter with stride 1 and maxpool layer with 2 × 2 filter of stride 2. In the end, two fully connected layers followed by softmax are used for classification.

1.2.2 VGG19 (Simonyan & Zisserman, 2015)

VGG19 is a variant of the VGG model that is based on VGG16 architecture. The basic difference is in the depth of the architecture. It follows the same structure pattern of convolution and maxpool blocks as present in VGG16 but supports 19 layers (16 convolution layers, 3 fully connected layer, 5 maxpool layers, and 1 softmax layer).

1.2.3 ResNet (He et al., 2016)

ResNet is based on residual networks, i.e., "identity shortcut connection" that skips one or more layers. ResNet won the first prize in the

ImageNet Challenge 2015. It introduces the idea of residual blocks to address the issue of vanishing gradients. The architecture is based on the skipped connection that skips layers to form a residual block. These residual blocks are further stacked together to form the network. Conventional ResNet architecture consists of 152 layers that are almost 8 times deeper than the VGG network. However, different versions of the ResNet model have been proposed in literature such as ResNet50 and ResNet101.

1.2.4 Inception (Szegedy et al., 2015)

The Inception model was developed to tackle the issue of variation in the size of striking parts of images. A larger kernel would tend toward a more global distribution and a smaller kernel toward a local distribution. To overcome this, filters of different sizes were introduced on the same level, thus making the model's architecture wider as compared to deeper. An Inception network consists of repeated components known as Inception modules. These modules allow using multiple layers of filters instead of a single-size filter. Different Inception architecture versions have been proposed such as InceptionV1, InceptionV2, InceptionV3, and InceptionResNet.

Table 2 Pre-Trained models used

Pre-Trained Models	Depth	Parameter
VGG16	16	138,357,533
VGG19	19	143,667,240
ResNet50	50	25,636,712
ResNet101	101	44,707,176
InceptionV3	159	23,851,784
Xception	126	22,910,480

1.2.5 Xception (Chollet, 2017)

The Xception model is based on the Inception model that consists of 126 layers. Xception is an extension of the Inception architecture which replaces the standard Inception modules with depthwise separable convolutions. The Inception modules present in the Inception model were interpreted in Xception as an intermediate step between the conventional convolutions and the depthwise separable convolutions (a depthwise convolution followed by a pointwise convolution). Depthwise convolution is the channel-wise $n \times n$ spatial convolution and pointwise convolution is the 1×1 convolution to change the dimension. Thus, the resultant architecture with depthwise separable convolutions outperformed VGG-16, ResNet, and Inception V3 in most classical classification challenges.

Table 2 depicts the pre-trained models highlighting the number of layers and parameters used. The VGG16 model has the least number of layers, while InceptionV3 has the highest number of layers. The Xception model contains the least number of trainable parameters despite being approximately seven times deeper than the VGG19 model, which contains the highest number of parameters.

2. Objectives

Plant diseases are one of the primary reasons for food insecurity all around the globe. However, the leaves of a plant contain visual characteristics that can be used to determine the plant's health (Jadhav, & Garg, 2022). The cucumber crop dominates a sizable portion of India's total summer growing crop, and it also has economic and nutritional benefits. Cucumbers are rich in electrolytes that help in preventing dehydration, and they support cardiovascular, bone, and skin health. In terms of the economic aspect, India has emerged as the largest exporter of cucumber/gherkins in the world, exporting USD 223 million worth of cucumber in 2020–21 (Press Information Bureau Government of India Ministry of Commerce & Industry, 2022). However, cucumber plants may get infected by a number of harmful diseases that can lower quality and yield. Manually monitoring plant disease is quite challenging as it takes a great deal of effort and time. Thus, the work proposed in this article focuses on disease detection in cucumber plants using transfer learning techniques. The primary objectives of the work is twofold:

a) Propose a deep learning model for disease detection in cucumber plants to reduce the manual effort required by conventional techniques.
b) Analyze and compare the performance of different pre-trained architectures to obtain the best model which can be used in the future for disease detection in other similar plants.

The next section discusses the methodology and highlights the proposed approach for the research work.

3. Methodology

This section describes the proposed approach, dataset, and experimental setup that are used in the analysis of various deep learning models for disease classification in the cucumber plant.

3.1 Proposed approach

Figure 1 depicts the workflow of the proposed system. The dataset is initially fed into the image data generator where the images are pre-processed using techniques like rescaling, zooming, shearing, and flipping. The images obtained are then fed into the proposed model for classification into diseased and healthy classes. As shown in Figure 1, the proposed model inherits knowledge from the benchmark pre-trained models with the use of transfer learning. In this work, several pre-trained models, namely VGG16, VGG19, ResNet50, ResNet101, InceptionV3, and Xception, are tested to identify the best model for the task of classifying diseases in the cucumber plant. The layers inherited from the pre-trained models are frozen to make them untrainable (Sharma, Kumar, & Deka, 2022), however, additional layers are added to fine-tune the deep learning model.

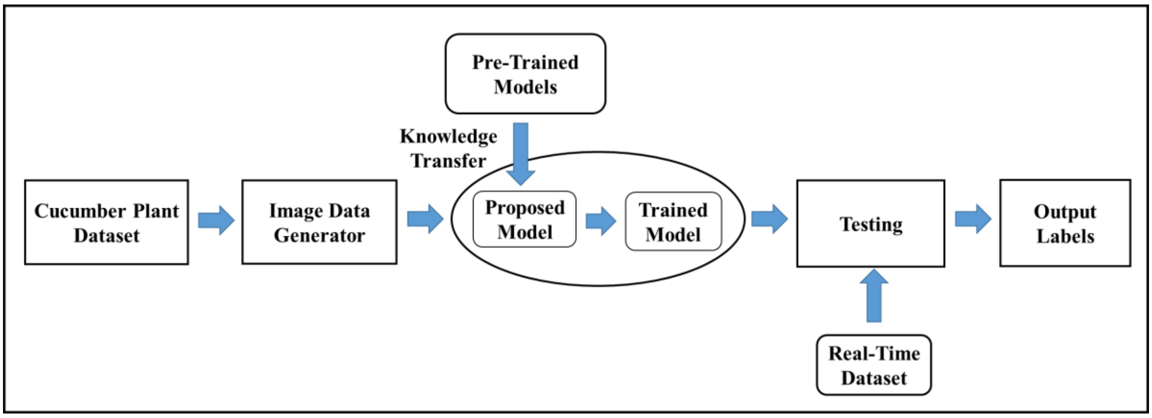

Figure 1 Flowchart of Proposed Approach

Figure 2 depicts the architecture of the proposed model, where the base architecture of the pre-trained model is conjoined by a flattened layer followed by two dense layers. In the end, the sigmoid layer is added to classify images into diseased and healthy cucumber plants. To identify the best-performing model, the proposed model has been tested by systematically increasing the number of dense layers from 1 to 10, and the addition of two dense layers obtained the best accuracy. Furthermore, the ReLU activation function is used along with the dense layers to overcome the vanishing gradient problem encountered by the neural networks, where the gradients of the loss function approach zero while backpropagating, thus making the network hard to train. ReLU overcomes this issue by multiplying several ReLU derivatives in backpropagation equations to result in either zero or one. Moreover, each of the pre-trained models is trained with an early stopping criterion of patience 10, such that the training procedure terminates if validation accuracy fails to increase for 10 consecutive epochs. Finally, the proposed model, after being fine-tuned and trained, is tested on real-time datasets to produce the output labels corresponding to the input images.

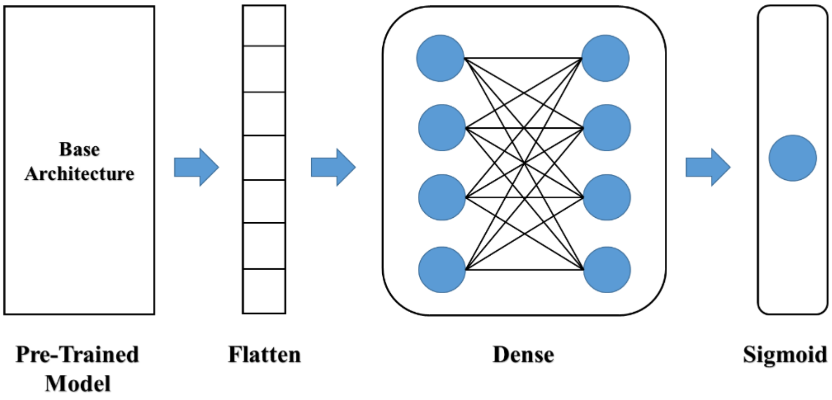

Figure 2 Architecture of proposed model

3.1.1 Dataset used

The dataset used for the experiment includes images of cucumber disease taken from the cucumber plant diseases dataset (Negm, 2020). The dataset includes a total of 691 images divided into two classes. We used 591 images to train the model and 100 images to test the model. Figures 3 and 4 represent images of healthy and infected cucumber plants, respectively.

Figure 3 Healthy cucumber plant

Figure 4 Infected cucumber plant

3.1.2 Experimental setup

In this research work, the experiment was carried out on a system with an Intel 8th gen i7-H processor, 16GB of RAM, and 4GB Nvidia Geforce GTX 1050ti graphic card that contained 768 cuda cores. Python was used as the programming language and Jupyter notebook was used to run the program. Standard software libraries were used, such as Keras, Tensorflow, Matplotlib, and Numpy (Duggal, 2022).

4. Results and discussion

In the proposed work, different state-of-the-art models are fine-tuned using transfer learning for disease detection in cucumber plants. Owing to the fact that different models vary depending on their depth or the number of parameters, this work considers different evaluation metrics such as training accuracy, testing accuracy, and number of epochs required to achieve convergence. Furthermore, all the models are trained using adam optimizer and binary cross-entropy where an early callback of patience 10 has been used during the training process so that the training process is terminated if no improvement is recorded in the performance over a period of 10 epochs.

Figures 5 and 6 depict the Loss vs Epoch and Accuracy vs Epochs graphs of variants of the VGG model, namely VGG16 and VGG19, respectively. Although VGG19 contains more convolution layers than VGG16 the result obtained represents an identical convergence pattern for both of the models. On the other hand, both variants of the ResNet models, ResNet50 and ResNet101, projected major fluctuations in the validation accuracy during the training process, as shown in Figures 7 and 8, respectively. The fluctuations present in the graphs are due to the application of a fairly large and complex model in comparison to the dataset, which results in the model searching for an optimal solution rather than settling for a solution. The ResNet models are approximately two times and four times larger than the VGG models that converge to the solution swiftly

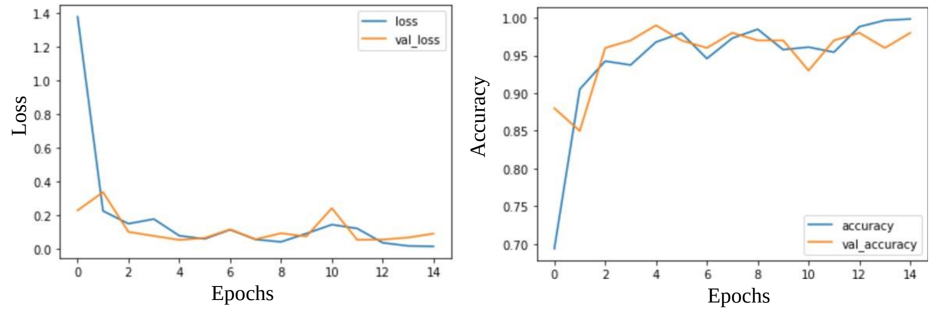

Figure 5 Loss vs Epochs and Accuracy vs Epochs of VGG16

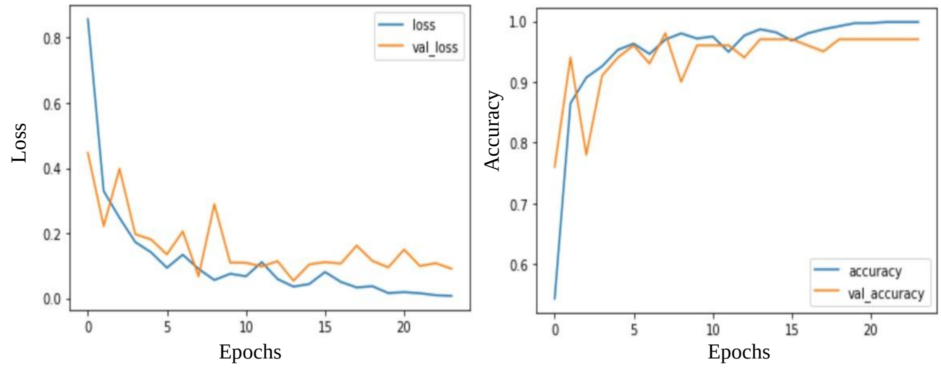

Figure 6 Loss vs Epochs and Accuracy vs Epochs of VGG19

The Loss vs Epoch and Accuracy vs Epoch graphs of InceptionV3 and Xception shown in Figures 9 and 10, respectively, demonstrate fluctuations similar to the ResNet models due to their large structure size. However, variable-sized filters used in the Inception modules of these models extract more efficient features, thus enabling them to converge to a solution in lesser time.

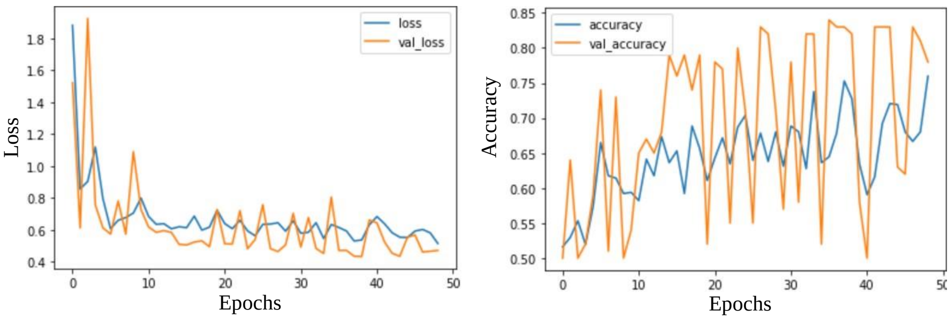

Figure 7 Loss vs Epochs and Accuracy vs Epochs of ResNet50

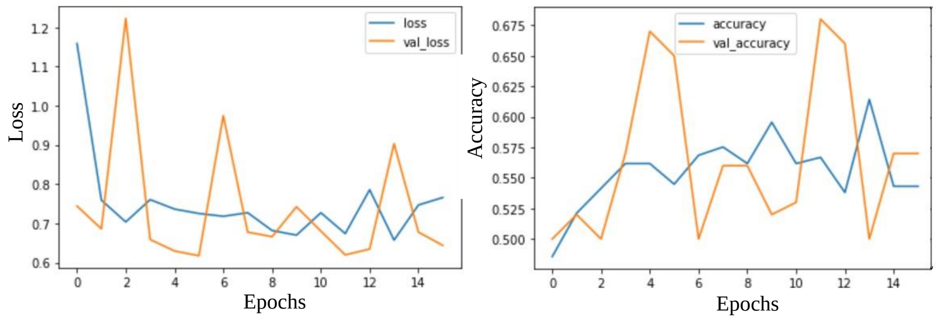

Figure 8 Loss vs Epochs and Accuracy vs Epochs of ResNet101

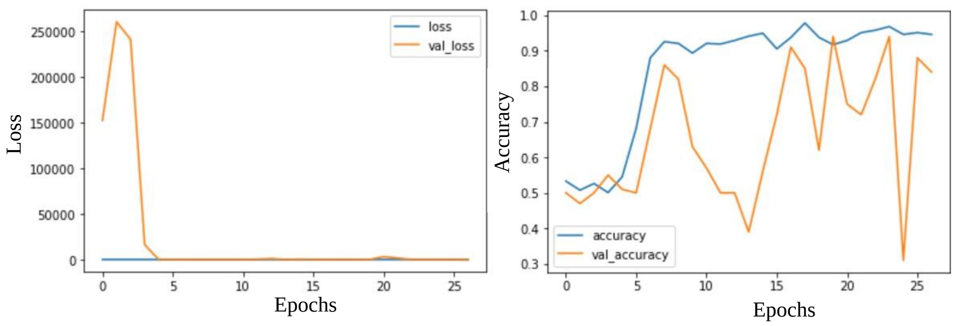

Figure 9 Loss vs Epochs and Accuracy vs Epochs of InceptionV3

Table 3 presents an in-depth comparison of the pre-trained models used in the proposed work on the basis of size, depth, time (ms) per inference step on GPU, epochs, and testing accuracy. The results obtained demonstrate that VGG16, despite being the smallest model in terms of depth, achieves the best training accuracy of 99.91% and testing accuracy of 98% in just 8 epochs, whereas ResNet101 records the worst training accuracy of 56.45%. Moreover, VGG19 obtained similar testing accuracy as compared to VGG16 but lagged behind by 3% in terms of training accuracy. This may be due to the three additional layers present in VGG19 requiring more epochs to attain convergence.

The Xception model follows VGG19 in terms of testing accuracy but requires almost twice the number of epochs as compared to VGG16 and VGG19 due to its larger size (number of layers). Although both InceptionV3 and Xception consist of comparatively greater numbers of layers than ResNet50 and ResNet101, they record higher training and testing accuracies due to the presence of Inception modules. As the diseased target areas vary in size from image to image, the Inception module is better equipped to detect these target areas because of the presence of different-sized filters. The ResNet50 model was trained for the highest number of epochs and obtained 83% testing accuracy, whereas ResNet101 records the least training and testing, and training and testing accuracy were 56.45% and 68%, respectively.

The results obtained demonstrate that VGG16 outperformed the rest of the models. Furthermore, from Table 3, it can be observed that VGG16 consists of the least number of layers as well as the minimum time (ms) required per inference step on GPU among the compared models. However, VGG16 requires extensive space as compared to the other models, which is a major drawback.

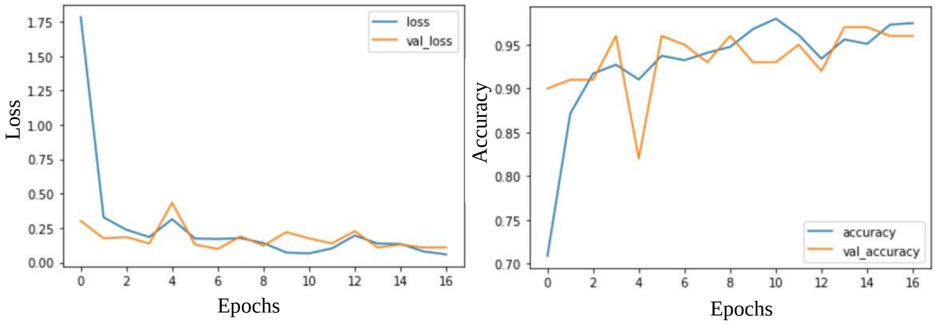

Figure 10 Loss vs Epochs and Accuracy vs Epochs of Xception

On the other hand, InceptionV3 and Xception require the least amount of space among the compared models, however, they require 2.7ms and 3.9ms more time per inference step on GPU as compared to VGG16. Moreover, both InceptionV3 and Xception lag behind VGG16 by 4% and 1%, respectively, in terms of accuracy. VGG19 performs equally well compared to VGG16 in terms of accuracy and the number of epochs but requires more space and has a higher time per inference step on GPU.

Table 3 Pre-trained models performance comparison

Pre-Trained Models	Size (MB)	Depth	Time (ms) per Inference Step (GPU)	Epochs	Training Accuracy	Testing Accuracy
VGG16	528	16	4.2	8	99.91%	98%
VGG19	549	19	4.4	8	96.02%	98%
ResNet50	98	107	4.6	45	74.35%	83%
ResNet101	171	209	5.2	12	56.45%	68%
InceptionV3	92	189	6.9	24	96.27%	94%
Xception	88	81	8.1	15	96.17%	97%

The results obtained in this paper add another dimensionality to the comparison of pre-trained models by introducing epochs along with accuracies. The results obtained also indicate that the VGG16 model is the most efficient model, in terms of both accuracy and training epochs, for disease detection in cucumber plants. Furthermore, the fine-tuned model can be implemented for the real-time detection of diseases in cucumber plants to automate the manual procedure used by farmers for the detection of diseases.

5. Conclusions and future works

In this paper, several benchmark pre-trained models, namely VGG16, VGG19, ResNet50, ResNet 101, InceptionV3, and Xception, are fine-tuned and analyzed for disease detection in cucumber plants. The results obtained identify the VGG16 model as the most efficient model among all tested with an accuracy of 98%. In addition, the VGG16 model requires the least amount of CPU and GPU in time which makes it a viable option for integration with real-time devices. In future research, the proposed approach could be implemented for disease detection in other plants. It also possesses the potential of being applicable in other sectors, such as healthcare, for classifying X-ray images or bio-metric systems for facial recognition.

6. References

Atila, Ü., Uçar, M., Akyol, K., & Uçar, E. (2021). Plant leaf disease classification using EfficientNet deep learning model. *Ecological Informatics*, *61*, 101182. https://doi.org/10.1016/j.ecoinf.2020.101182

Bao, W., Huang, X., Hu, G., & Liang, D. (2021).

Identification of maize leaf diseases using improved convolutional neural network. *Nongye Gongcheng Xuebao/Transactions of the Chinese Society of Agricultural Engineering*, *37*(6), 160-167. https://doi.org/10.11975/j.issn.1002-6819.2021.06.020

Becherer, N., Pecarina, J., Nykl, S., & Hopkinson, K. (2019). Improving optimization of convolutional neural networks through parameter fine-tuning. *Neural Computing and Applications*, *31*(8), 3469-3479. https://doi.org/10.1007/s00521-017-3285-0

Bhatt, P., Sarangi, S., Shivhare, A., Singh, D., & Pappula, S. (2019). Identification of diseases in corn leaves using convolutional neural networks and boosting. In *ICPRAM 2019 - Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods*, 894–899. https://doi.org/10.5220/0007687608940899

Bosilj, P., Aptoula, E., Duckett, T., & Cielniak, G. (2020). Transfer learning between crop types for semantic segmentation of crops versus weeds in precision agriculture. *Journal of Field Robotics*, *37*(1), 7-19. https://doi.org/10.1002/rob.21869

Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 1800-1807. https://doi.org/10.1109/CVPR.2017.195

Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition* (pp. 248-255). Ieee.

Dhaka, V. S., Meena, S. V., Rani, G., Sinwar, D., Ijaz, M. F., & Woźniak, M. (2021). A survey of deep convolutional neural networks applied for prediction of plant leaf diseases. *Sensors*, *21*(14), 4749. https://doi.org/10.3390/s21144749

Duggal, N. (2022, Dec 21). Top 10 Python Libraries for Data Science for 2023. *Simplilearn.* Retrived form https://www.simplilearn.com/difference-between-descriptive-inferential-statistics-article

Fan, X., Luo, P., Mu, Y., Zhou, R., Tjahjadi, T., & Ren, Y. (2022). Leaf image based plant disease identification using transfer learning and feature fusion. *Computers and Electronics in Agriculture*, *196*, 106892. https://doi.org/10.1016/j.compag.2022.106892

Ganatra, N., & Patel, A. (2020). Performance Analysis Of Fine-Tuned Convolutional Neural Network Models For Plant Disease Classification. *International Journal of Control and Automation*, *13*(3), 293-305.

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 (Switzerland)*, *10*(12), 1388. https://doi.org/10.3390/electronics10121388

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 770-778. https://doi.org/10.1109/CVPR.2016.90

Hussain, R., Karbhari, Y., Ijaz, M. F., Woźniak, M., Singh, P. K., & Sarkar, R. (2021). Revise-net: Exploiting reverse attention mechanism for salient object detection. *Remote Sensing*, *13*(23), 4941. https://doi.org/10.3390/rs13234941

Jadhav, S., & Garg, B. (2022). Comprehensive Review on Machine Learning for Plant Disease Identification and Classification with Image Processing. In *Proceedings of International Conference on Intelligent Cyber-Physical Systems*. https://doi.org/10.1007/978-981-16-7136-4_20

Jin, X. B., Yu, X. H., Wang, X. Y., Bai, Y. T., Su, T. L., & Kong, J. L. (2020). Deep learning predictor for sustainable precision agriculture based on internet of things system. *Sustainability*, *12*(4). https://doi.org/10.3390/su12041433

Kumar, M., Kumar, A., & Palaparthy, V. S. (2021). Soil Sensors-Based Prediction System for Plant Diseases Using Exploratory Data Analysis and Machine

Learning. *IEEE Sensors Journal*, *21*(16), 17455-17468. https://doi.org/10.1109/JSEN.2020.3046295

Kundu, N., Rani, G., Dhaka, V. S., Gupta, K., Nayak, S. C., Verma, S., ... & Woźniak, M. (2021). Iot and interpretable machine learning based framework for disease prediction in pearl millet. *Sensors*, *21*(16), 5386. https://doi.org/10.3390/s21165386

Ma, L., Shuai, R., Ran, X., Liu, W., & Ye, C. (2020). Combining DC-GAN with ResNet for blood cell image classification. *Medical and Biological Engineering and Computing*, *58*(6), 1251-1264. https://doi.org/10.1007/s11517-020-02163-3

Moawed, M. M. (2016). Evaluation of morphological and anatomical characters for discrimination and verification of some Medicago sativa (L.) Cultivars. *Indian Journal of Agricultural Research*, *50*(2), 183-192. https://doi.org/10.18805/ijare.v50i2.9589

Nagasubramanian, G., Sakthivel, R. K., Patan, R., Sankayya, M., Daneshmand, M., & Gandomi, A. H. (2021). Ensemble Classification and IoT-Based Pattern Recognition for Crop Disease Monitoring System. *IEEE Internet of Things Journal*, *8*(16), 12847-12854. https://doi.org/10.1109/JIOT.2021.3072908

Naveen, P., & Sivakumar, P. (2021). A deep convolution neural network for facial expression recognition. *Journal of Current Science and Technology*, *11*(3), 402-410. https://doi.org/10.14456/jcst.2021.40

Negm, K. (2020). *Cucumber plant diseases dataset.* Retrieved form https://www.kaggle.com/datasets/kareem3egm/cucumber-plant-diseases-dataset

Paymode, A. S., & Malode, V. B. (2022). Transfer Learning for Multi-Crop Leaf Disease Image Classification using Convolutional Neural Network VGG. *Artificial Intelligence in Agriculture*, *6*, 23-33. https://doi.org/10.1016/j.aiia.2021.12.002

Press Information Bureau Government of India Ministry of Commerce & Industry. (2022). Retrived form https://pib.gov.in/newsite/PrintRelease.aspx?relid=229987

Saleem, M. H., Potgieter, J., & Arif, K. M. (2020). Plant disease classification: A comparative evaluation of convolutional neural networks and deep learning optimizers. *Plants*, *9*(10), 1-17. https://doi.org/10.3390/plants9101319

Schapire, R. E. (2013). Explaining adaboost. In *Empirical Inference: Festschrift in Honor of Vladimir N. Vapnik* (pp. 37-52). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-41136-6_5

Shagun, K. (2021). Agri share in GDP hit 20% after 17 years: Economic Survey. *Down to Earth*. Retrieved form https://www.downtoearth.org.in/news/agriculture/agri-share-in-gdp-hit-20-after-17-years-economic-survey-75271

Sharma, M., Kumar, C. J., & Deka, A. (2022). Early diagnosis of rice plant disease using machine learning techniques. *Archives of Phytopathology and Plant Protection, 55*(3), 259-283. https://doi.org/10.1080/03235408.2021.2015866

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

Singh, U. P., Chouhan, S. S., Jain, S., & Jain, S. (2019). Multilayer Convolution Neural Network for the Classification of Mango Leaves Infected by Anthracnose Disease. *IEEE Access*, *7*, 43721-43729. https://doi.org/10.1109/ACCESS.2019.2907383

Sinha, A., Shrivastava, G., & Kumar, P. (2019). Architecting user-centric internet of things for smart agriculture. *Sustainable Computing: Informatics and Systems*, *23*, 88-102. https://doi.org/10.1016/j.suscom.2019.07.001

Subramanian, M., Shanmugavadivel, K., & Nandhini, P. S. (2022). On fine-tuning deep learning models using transfer learning and hyper-parameters optimization for disease identification in maize leaves. *Neural Computing and Applications*, *34*. https://doi.org/10.1007/s00521-022-07246-w

Suwanagul, D., Kokaew, J., & Suwanagul, A. (2013). Diversity of causal fungi in weed diseases and potential use as a biological weed control for vegetable plots in Thailand. *Rangsit Journal of Arts and Sciences*, *3*(1), 39-43.

Syed-Ab-Rahman, S. F., Hesamian, M. H., & Prasad, M. (2022). Citrus disease detection and classification using end-to-end anchor-based deep learning model. *Applied Intelligence*, *52*(1), 927-938. https://doi.org/10.1007/s10489-021-02452-w

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9). https://doi.org/10.1109/CVPR.2015.7298594

Trading Economics. (n.d.). *India - Employment in agriculture (% of total employment).* Reteived form https://tradingeconomics.com/india/employment-in-agriculture-percent-of-total-employment-wb-data.html

Zeng, Q., Ma, X., Cheng, B., Zhou, E., & Pang, W. (2020). GANS-based data augmentation for citrus disease severity detection using deep learning. *IEEE Access*, *8*, 172882-172891. https://doi.org/10.1109/ACCESS.2020.3025196

Zhou, C., Zhou, S., Xing, J., & Song, J. (2021). Tomato Leaf Disease Identification by Restructured Deep Residual Dense Network. *IEEE Access*, *9*, 28822-28831. https://doi.org/10.1109/ACCESS.2021.3058947