International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7390
Article Received: 29 April 2023 Revised: 24 June 2023 Accepted: 18 July 2023

A Hybrid Machine Learning Model to Recognize and Detect Plant Diseases in Early Stages

Dr. Sudeepthi Govathoti[1], Deepthi Kamidi[2], Dr Y Madhavi Reddy[3], Dr. Mahesh Kotha[4], Gangolu Yedukondalu[5], Dr. Y. Krishna Bhargavi[6], Bh. Prashanthi[7]
[1]Assistant Professor, Department of CSE, GITAM University, Hyderabad, India.
E-mail: sudeepthi.chinnam@gmail.com.
[2]Assistant Professor, Department of CSE, Vignan Institute of Technology and Science, Hyderabad.
E-Mail: deepthikamidi83@gmail.com
[3]Assistant professor, H&S (Mathematics), CVR college of Engineering, Hyderabad, India.
E-mail: madhavireddy75@gmail.com
[4]Associate professor, Department of CSE (AI&ML), CMR Technical Campus, Hyderabad.
E-Mail: kotha.mahesh528@Gmail.com.
[5]Professor, CSE (AI&ML) Department, Vignana Bharathi Institute of Technology, Hyderabad, Telangana.
E-mail: gyedukondalu@gmail.com
[6]Associate Professor, Department of CSE, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, E-mail: ykrishna915@grietcollege.com
[7]Assistant professor, Department of CSE, Gokaraju Rangaraju Institute of Engineering and Technology Hyderabad, India, E-mail: bhupathi.prashanthi@gmail.com.

Abstract: This paper presents an improved Inception module to recognise and detect plant illnesses substituting the original convolutions with architecture based on modified-Xception (m-Xception). In addition, ResNet extracts features by prioritising logarithm calculations over softmax calculations to get more consistent classification outcomes. The model's training utilised a two-stage transfer learning process to produce an effective model. The results of the experiments reveal that the suggested approach is capable of achieving the specified level of performance, with an average recognition fineness of 99.73 on the public dataset and 98.05 on the domestic dataset, respectively.

Keywords: Plant disease, ML, ResNet, Xception network, KNN, CNN, AlexNet and SqueezeNet v1.1 models.

I. INTRODUCTION

Agriculture has been a vital contributor to the economic development, employment, food supply, infrastructure, trading, and other aspects of both developing nations and nations that have already established themselves. In a number of nations still undergoing economic growth, agriculture is still responsible for a sizeable amount of the Gross Domestic Product (GDP). Because several of Africa's countries do not even allow any of their public finances to agriculture, the sector does not contribute anything to their countries' overall gross domestic product. Gulati and Juneja (2022). Most of those living in abject poverty are employed in some capacity within the agricultural industry, which is their primary source of income. Crop output is anticipated to be boosted by forty percent by 2030 Khush (2005), to fulfil the challenge set before it. If there is any chance of reaching food security on a worldwide scale, then the production of crop needs to be raised in an environmentally friendly manner. This increase in crop production could be made possible by making varieties with high yields and getting rid of agricultural pests like bugs and diseases.

Fungi, bacteria, and viruses are the three pathogens most likely to cause crop plants to die. Because these diseases keep spreading, the total amount of crop grown eventually decreases. It causes Asia to lose between 10 and 15% of all the crop it grows (Gianessi) (2014). Early detection of these diseases is very important because it increases production and production quality, reduces pesticides used, and helps the economy grow. Traditional methods, such as manual or optical observation, are hard to do and require a lot of work. However, they are the only ones that can be used. Because the diseases are so hard to figure out, even the most experienced agronomists and pathologists often fail to find the right cause. Because of this, traditional methods are often subjective, inaccurate, and require a lot of work. Because of this, it would be helpful to build an automatic system that would help agronomists, pathologists, and even farmers figure out what's wrong with crop faster and take steps to stop it right away Shrivastava et al (2021).

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7390
***Article Received**: 29 April 2023 **Revised:** 24 June 2023 **Accepted:** 18 July 2023*

With the rise of smartphones and improved computer power, ML and image recognition methods have been used more and more in recent years. This includes fields as different as medical image disposal Wells III (2016); Miki et al. (2017); biomimetics Mondal and Bours (2017); and food analysis Gokmen and Sugut (2007), Lopez et al. (2011), and applications in industry. Identification of fingerprints: Lee and Kim (2016) and Kien et al (2019). Some of them are Zhao et al (2021) Hung et al. (2019), Ashourloo et al. (2016), and Cao et al. (2016) all say that our performance in these areas is the best it can be. The way these methods work now is about the same as how other methods work.

Also, as computer-assisted disease diagnosis has improved, it has led to the development of computer vision algorithms and Convolutional Neural Networks (CNN), which are better at identifying and classifying objects. Recent studies, such as those by Sutaji and Yldz (2022), Rahman et al. (2020), Bari et al. (2021), and Asfaqur Rahman et al (2021). Deep learning CNN has recently become the most popular way to figure out what diseases are occurring in leaves. Singla et al. (2022a,b), Ramesh and Vydeki (2020), and Chen et al. (1998) are some of the most recent people to write about this subject (2022). Also, a classifier is often used to identify and group crop diseases, and this area of study has gotten some attention in recent years. Sharma et al. discussed different ways to find crop diseases using machine learning and deep learning (transfer learning) (2022). Crop can get sick from the brown spot, bacterial blight, and crop blast. Upadhyay et al. came up with a good way to diagnose and classify crop plant diseases based on lesions' size, shape, and color on photos of the plant's leaves (Upadhyay and Kumar, 2022). The proposed model uses a way to divide a picture into two parts called "global thresholding," which was invented by Otsu. This makes it possible to eliminate background noise in the original image. The author suggests using a CNN trained with 4,000 pictures of each damaged leaf and 4,000 pictures of healthy crop leaves to find these three diseases in crop. Kabir et al. (Kabir et al., 2021) mostly named the diseases that hurt the different kinds of crop. Collecting visual information is the first step in determining what's wrong with crop plants and how to treat them. Next, we put this image data into three groups: Leaf Blast, Brown Spot, and Healthy leaf. The diseases that hurt the crop plants can be correctly diagnosed and treated. Shrivastava et al. (Shrivastava et al., 2021) looked at the effectiveness of many pre-trained deep CNN models, such as AlexNet, Vgg16, ResNet152V2, InceptionV3, InceptionResNetV2, Xception, MobileNet, and InceptionResNetV2. The cutting-edge machine learning method deep learning (DL), more specifically deep convolutional neural network (CNN), has quickly become the best way to solve problems of all sizes because it works so well and breaks records. Because of this, deep learning is now the best way to handle tasks of different sizes, beating out other methods. Ramesh and Vydeki (2020), Rahman et al. (2020), Bari et al. (2021), Asfaqur Rahman et al. (2021), Sharma and Singh (2022), Singla et al (2022).

The increasing popularity of CNN techniques can be attributed to their capacity to automatically extract picture features and diagnose and classify plant diseases using a strategy that covers the full process from start to finish. Some examples of how models use images for classification: MobileNetV2, introduced by Sandler et al. (2018), is a unique mobile architecture that improves the state-of-the-art performance of mobile models over a range of model sizes, as well as across a variety of workloads and benchmarks. More than that, we have discussed some great ways to use these mobile models for object detection inside a special framework called SSD Lite. Mobile DeepLabv3, a condensed version of DeepLabv3, is introduced and used to demonstrate model construction for mobile semantic segmentation. For example, Sandler et al. (2018). With their proof of concept, Sifre and Mallat (2014) show that joint scattering and group convolutions can be used on generic object image datasets. Separable convolutions, which operate similarly to rigid-motion convolutions, have been shown to increase the performance of convolutional networks. More than that, it is demonstrated that a non-invariant version of the rigid-motion scattering can achieve results that are on par with the output of the first layers of convolutional networks. Zoph et al. (2018) developed a new transferable search space. We coined the term "NASNet search space" to describe this brand-new indexing framework. Additionally, it uses the CIFAR-10 dataset to look for the optimal convolutional layer (or "cell"). To use it on the ImageNet dataset, we stack several more of these cells, each with its own set of parameters, to create a convolutional architecture we call the NASNet architecture. This structure is a convolutional model. In addition, we introduce Scheduled Drop Path, a novel regularisation technique. One of our contributions is a strategy that greatly improves generalisation in NASNet models.

To develop a Convolution Neural Network (ConvNet), it is common practice to limit the number of resources available. The accuracy of the ConvNets can be increased by scaling them up if more resources become available in the future. Model scaling was studied methodically by Tan and Le (2019), who found that optimising the relationship between network depth, width, and resolution could boost performance. It's based on a new approach to scaling. As a result, we've devised a technique for uniformly upscaling depth, width, and resolution using a compound coefficient that's easy to implement and yields excellent results. Our results show that this method may efficiently scale up both MobileNets and ResNet. Further, we build a new baseline network using neural architecture search. We next apply this network on a larger scale to produce a set of models we call Efficient Nets. Compared to previous ConvNets, these models are vastly superior in accuracy and efficiency.

Recent studies have shown that deep learning-based models can classify and find plant diseases, even though they have many problems. In this study, we add new convolutional layers to the Xception module to help it recognise and diagnose diseases in crop, cotton and vegetable plants. So, the already-trained ResNet was combined with the modified Xception (m-Xception) module to extract high-quality visual features of crop illness. The classification results are then made more stable by adding algorithm to the softmax. In a nutshell, here are the most important results of this study.

- Current solutions rely on the Inception model, incapable of creating space behaviour and extracting feature maps from noisy areas. This is due to the significant number of false positives generated by the model, which, in the end, diminishes the model's overall effectiveness. This is taken care of by the Xception architecture, which, in the model that has been proposed, basically pulls features from all noisy level portions while also enhancing the performance of the model in comparison to the one that came before it.
- Mobile net suffers from an overfitting problem due to a lack of regularisation, which produces flash results on an unseen dataset and causes model performance scores to decline. To overcome this problem, regularisation is used in the proposed model to improve scores and show that log functions work perfectly.
- The proposed model employs the Xception architecture as the backbone, adding an LPG-Softmax layer to achieve robust classification over noisy and non-noisy images. Features are extracted from the images, and then Xception with 3D convolutional layers manages the classification. The features pass it as input into the last layer. After that, the log softmax layer computes probabilistic values for each image; the highest probabilistic value for a particular class gives the image category.

II. RELATED WORK

Numerous contemporary deep learning models and architectures have emerged subsequent to the inception of AlexNet [10], primarily for the purposes of image detection, segmentation, and classification. This section outlines the investigations conducted utilizing prominent deep learning architectures for the purpose of identifying and categorizing diseases affecting plants. Additionally, there exist several relevant studies that have introduced novel visualization techniques and enhanced versions of deep learning architectures in order to attain superior outcomes. The PlantVillage dataset has gained significant usage due to its comprehensive collection of 54,306 images depicting 26 plant diseases across 14 distinct crop types. Furthermore, the authors employed multiple performance metrics to assess the chosen deep learning models, which are delineated as follows.

The study employed Convolutional Neural Network (CNN) for the purpose of classifying diseases in maize plants. Additionally, histogram techniques were utilized to demonstrate the model's significance. The findings are documented in reference [11]. The study conducted in reference [12] utilized fundamental convolutional neural network (CNN) structures such as AlexNet, GoogLeNet, and ResNet to accurately classify tomato leaf diseases. The model's performance was evaluated by plotting the training and validation accuracy. Among all the Convolutional Neural Network architectures, ResNet was determined to be the most optimal. The implementation of LeNet architecture was utilized to detect diseases in banana leaves, with the evaluation of the model in Color and Gray Scale modes being conducted through the use of CA and F1-score, as documented in reference [22]. In the study referenced as [18], a total of five distinct CNN architectures were employed, including AlexNet, AlexNetOWTbn, GoogLeNet, Overfeat, and VGG. The results indicated that VGG architecture demonstrated superior performance compared to the other models.

The study conducted in reference [15] involved the identification of eight distinct plant diseases through the implementation of three classifiers, namely Support Vector Machines (SVM), Extreme Learning Machine (ELM), and K-Nearest Neighbor (KNN). These classifiers were utilized in conjunction with advanced deep learning models such as GoogLeNet, ResNet-50, ResNet-101, Inception-v3, InceptionResNetv2, and SqueezeNet. A comparative analysis was conducted among the models, and it was observed that ResNet-50 with SVM classifier exhibited superior performance in terms of performance metrics such as sensitivity, specificity, and F1-score. As per the reference

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7390
***Article Received**: 29 April 2023 **Revised:** 24 June 2023 **Accepted:** 18 July 2023*

[19], the identification of cassava disease was carried out using a novel deep learning model, namely Inception-v3. The classification of plant diseases in cucumber was conducted in [10] using two fundamental versions of Convolutional Neural Networks (CNN), resulting in the highest accuracy of 0.823. In reference [11], it was reported that the conventional technique for identifying and categorizing plant diseases was substituted with Super-Resolution Convolutional Neural Network (SRCNN).

The study employed AlexNet and SqueezeNet v1.1 models for the categorization of tomato plant disease. The results indicated that AlexNet outperformed SqueezeNet v1.1 in terms of accuracy, with a reported accuracy of 62%. In reference [13], a comparative analysis was conducted to determine the optimal deep learning architecture for the purpose of detecting plant diseases. In addition, a study conducted in reference [64] employed AlexNet and VGG-16 deep learning architectures to classify six diseases affecting tomato plants. The study also included a comprehensive analysis of classification accuracy, allowing for a detailed comparison between the two models. The aforementioned methodologies did not incorporate any form of visualization methodology to detect the indications of plant diseases.The present study utilized deep learning models and architectures, as well as visualization techniques, to enhance comprehension of plant diseases. In a previous study, the saliency map was proposed as a means of visualizing plant disease symptoms . Another study utilized the CaffeNet CNN architecture to identify 13 distinct types of plant disease, achieving a classification accuracy of 96.30%, surpassing previous methods such as SVM (27). Furthermore, multiple filters were employed to identify the locations of disease. In a similar vein, the authors employed the AlexNet and GoogLeNet convolutional neural network architectures, utilizing the publicly accessible PlantVillage dataset. [20] The assessment of the performance was conducted through the utilization of precision (P), recall (R), F1 score, and overall accuracy metrics.

This paper's distinctiveness lies in its utilization of three distinct scenarios (color, grayscale, and segmented) to assess performance metrics and compare two prominent CNN architectures. The study's findings indicate that GoogLeNet exhibited superior performance compared to AlexNet. Furthermore, the activation of visualization in the initial layers distinctly revealed the areas affected by illnesses. A study conducted in reference [17] employed a modified LeNet architecture for the purpose of identifying diseases in olive plants. The utilization of segmentation and edges maps facilitated the detection of plant diseases. In a study conducted by [16], the identification of four distinct diseases affecting cucumbers was performed. The accuracy of this identification process was subsequently evaluated through comparison with models utilizing Random Forest, Support Vector Machines, and AlexNet. Furthermore, the technique of image segmentation was employed to visualize the indications of ailments in the flora. In reference [7], a novel deep learning model known as the teacher/student network was presented, along with a unique visualization technique for detecting areas of plant disease. In reference [8], deep learning models incorporating detectors were utilized to annotate plant diseases and predict their corresponding probabilities.

A comparative study was conducted to evaluate the performance of three detectors, namely Faster-RCNN, RFCN, and SSD, in conjunction with well-known architectures such as AlexNet, GoogLeNet, VGG, ZFNet, ResNet-50, ResNet-101, and ResNetXt-101. The study aimed to identify the optimal architecture among the selected ones. The study's findings indicate that ResNet-50, when paired with the R-FCN detector, yielded the most optimal outcomes. Additionally, a bounding box was utilized to delineate the specific type of ailment present in the flora. The authors of reference [9] conducted a study on the detection of disease and pests in banana leaves. This was achieved through the utilization of three Convolutional Neural Network (CNN) models, namely ResNet-50, Inception-V2, and MobileNet-V1, in conjunction with Faster-RCNN and SSD detectors. As reported by reference [3], various combinations of Convolutional Neural Networks (CNN) were employed to generate heat maps from images of diseased plants. These heat maps were subsequently utilized to determine the likelihood of a specific type of disease manifestation. Additionally, the ROC curve is utilized to assess the efficacy of the model. Additionally, the paper incorporated feature maps pertaining to crop disease. The LeNet architecture was employed in reference [1] for the purpose of identifying and categorizing ailments present in the soybean crop. A study was conducted in [2] to compare the performance of AlexNet and GoogLeNet architectures in detecting tomato plant diseases. The results indicated that GoogLeNet outperformed AlexNet.

Additionally, the study proposed the use of occlusion techniques to identify the regions affected by the diseases. In reference [3], the VGG-FCN and VGG-CNN models were utilized to detect wheat plant diseases and to visualize features within each block. The study utilized the VGG-CNN model to detect Fusarium wilt in radish, as documented in reference [4]. Additionally, the K-means clustering method was employed to demonstrate the disease markers. In reference [5], a convolutional neural network

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7390
***Article Received**: 29 April 2023 **Revised:** 24 June 2023 **Accepted:** 18 July 2023*

(CNN) was suggested as a means of semantic segmentation for the purpose of identifying disease in cucumber. A DL model was employed in [6] to detect plant diseases by utilizing an approach centered on the individual symptoms/spots of the plants. In reference [7], a Deep Convolutional Neural Network (CNN) architecture was devised to perform the tasks of identification, classification, and quantification of eight distinct types of stress affecting soybean crops.

In a previous study referenced as [8], crop plant diseases were detected through the utilization of Convolutional Neural Networks (CNN). The resulting feature maps were subsequently utilized to identify the specific patches of diseased areas. In reference [7], a mobile application was developed utilizing an extended deep residual neural network to facilitate the accurate identification of plant diseases through the use of a hot spot. In reference [8], an algorithm utilizing the hot spot technique was employed. The aforementioned spots were obtained through modifications made to the segmented image, with the aim of achieving color constancy. In addition, each hot-spot that was acquired was characterized by two descriptors. One of these descriptors was utilized to assess the disease's color information, while the other was employed to discern the texture of the hot-spots. The identification of diseases in cucumber plants was accomplished through the utilization of a dilation convolutional neural network, as reported in reference [11]. In reference to [12], a cutting-edge method of visualization was introduced utilizing correlation coefficient and deep learning models such as AlexNet and VGG-16 architectures. The study described in reference [13] employed a convolutional neural network model (LeNet) in conjunction with color space and multiple vegetation indices to identify diseases present in grape plants.

III. PRELIMINARIES

In this section, we will discuss a few different approaches that serve as the foundation for the algorithm we have developed.

3.1. Inception model

Convolution layer filter training is attempted in three dimensions. There are three dimensions in this space the two physical ones and the channel width and height. As a result, we may map spatial and cross-channel correlations simultaneously using a single convolution kernel. In order to simplify and improve the effectiveness of this process, the Inception module expressly breaks it down into a collection of procedures that would independently analyse cross-channel and spatial connections. The procedure would be more efficient as a result. This notion inspired the creation of the Inception module. For clarity, a typical Inception module will begin by examining cross-channel correlations using several 1x1 convolutions. This will transform the input data into three or four spaces, each smaller than the original. The module will then use common 3x3 or 5x5 convolutions to map all correlations in these smaller 3D regions.

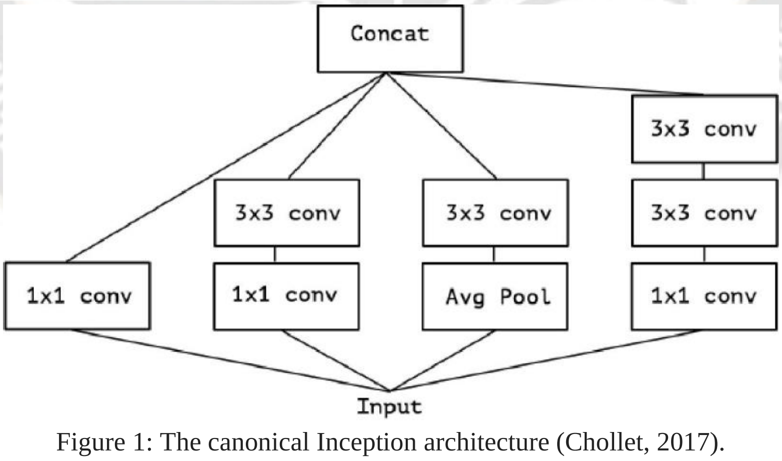

Figure 1: The canonical Inception architecture (Chollet, 2017).

Figure 1 depicts this scenario for the Inception model. The primary hypothesis underpinning Inception is that cross-channel and spatial correlations are sufficiently dissociated to the extent that it is desirable not to map them together.

3.2. Xception Network

The Xception design contains 36 convolutional layers, which are the primary source of characteristics extracted by the networks (Chollet, 2017). Because picture classification is our main focus in experimental evaluation for the time being, we will use a convolutional base layer first, followed by a logistic regression layer. Another option is to place fully-connected layers before the logistic regression layer. All modules comprising the 36 convolutional layers, except for the first and last, are surrounded by linear residual connections.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7390
***Article Received**: 29 April 2023 **Revised:** 24 June 2023 **Accepted:** 18 July 2023*

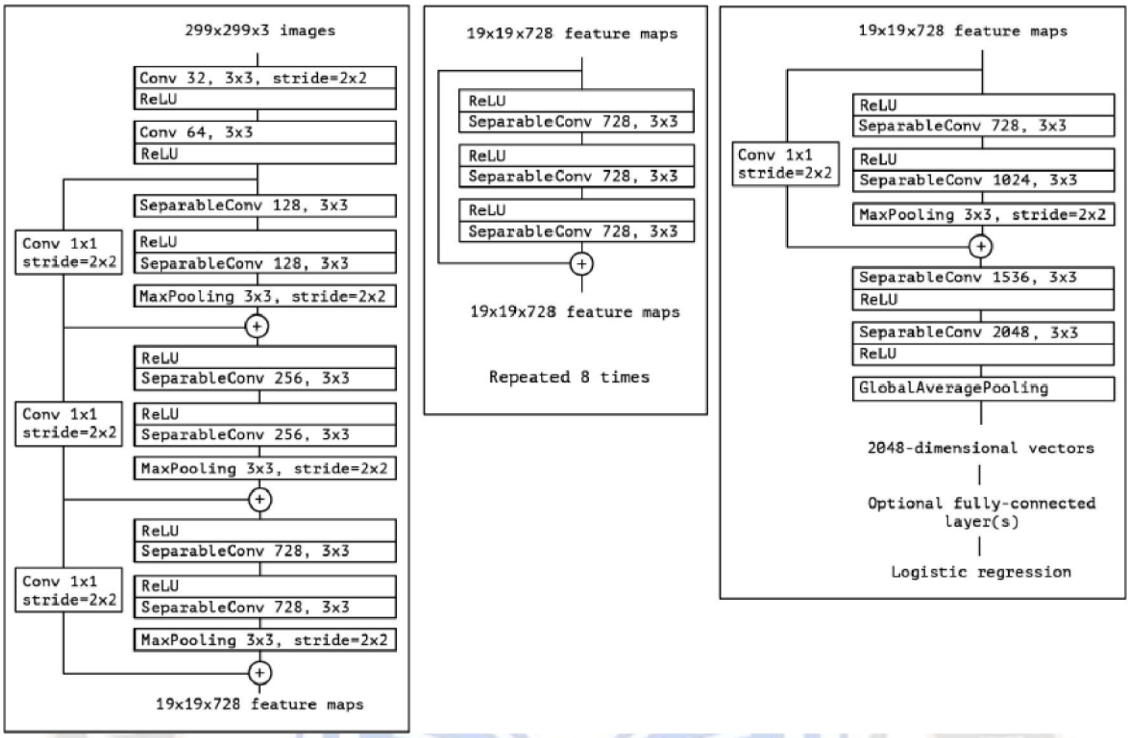

Figure 2: The Xception architecture (Chollet, 2017).

The parameters of the network are summarised in Figure 2. A simple way to think about the Xception architecture is as a linear stack of depth-separable convolution layers connected by residual connections. Just a high-level overview of the system is provided here. When compared to architectures like VGG-16 Simonyan and Zisserman (2014), which only takes 30 to 40 lines of code, using a high-level framework like Keras (2015) or TensorFlow-Slim Guadarrama and Silberman (2016) is quite different. This makes it easy to specify and make changes to the architecture. For those interested, Keras and TensorFlow power an open-source version of Xception. This code is included in the Keras Applications module 1, released under the MIT license.

3.3. ResNet-101

The name "ResNet-101" refers to a specific kind of convolutional neural network with 101 hidden layers (He et al., 2016). To use a trained network, the ImageNet database, which contains over a million images, must be loaded (Deng et al., 2009). Photos of things can be sorted into a thousand different categories by the pretrained network, including "keyboard," "mouse," "pencil," and "many animals." This has allowed the network to learn rich feature representations for a wide range of image formats. When an image is posted to the internet, its maximum possible size is 224 pixels on either side. Classify allows us to use the ResNet-101 model to quickly and easily label new images.

IV. PROPOSED MODEL

Normal Inception performs 1x1 convolution after depth-wise NXN convolution; Xception performs 1x1 convolution before any NXN convolution; however, in the model proposed here, 8x8 convolution is performed before any NXN convolution, whereas 4x4 is performed in the experiment case, demonstrating that there is no immediate ReLU for non-linearity. Although residual connections are always present in the proposed architecture, which makes results on the higher side even with non-linearity and changed order of operations, the experiment achieves improved scores from existing convolution settings even with any immediate ReLU function, which reduces complexity and returns a similar level of performance.

ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7390
***Article Received**: 29 April 2023 **Revised:** 24 June 2023 **Accepted:** 18 July 2023*

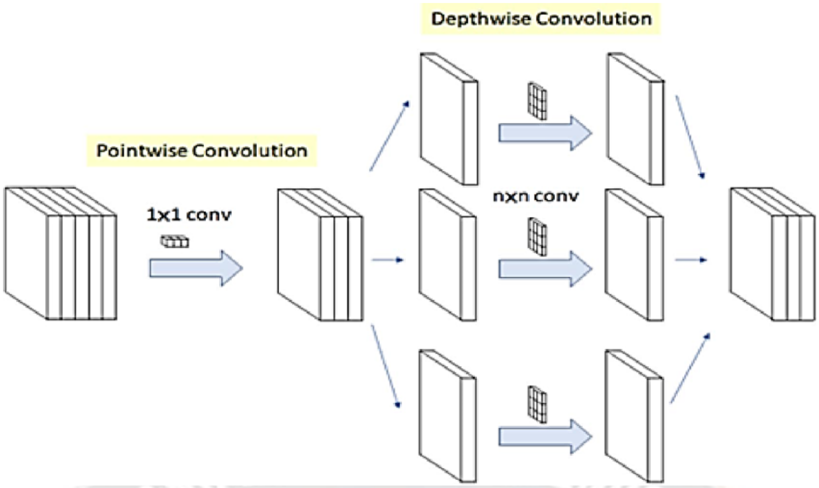

Figure 3: The convolution architecture.

Figure 3 represents the core order of convolution layers followed while implementing the proposed Xception Net, which shows how internal layers were arranged.

The log (Softmax(x)) function is applied to an n−dimensional input tensor. The LogSoftmax formulation can be simplified as:

$$\log soft\max = \frac{\log(\exp(x))}{\sum \exp(x)} = x - \log((\exp(x)\log(\sum \exp(x))) \quad (1)$$

This log(Softmax(x)) function is a loss function to optimise the proposed model performance. The Regularize function is used to regularise the layers to avoid over-fitting the model at any training instance.

$$loss = \frac{l}{n}\sum_{1}^{n} L_i + \lambda RW_i^2$$

$$where - L = l * \log(avg(r)) \quad (2)$$

Algorithm 1 shows the preprocessing steps. This algorithm starts with creating a model using the current dataset. In Xception models, annotations are unnecessary; classes are identified based on the folder's name. First, iterate through the data images, moving them into newly created directories based on the zip dataset's class names. The balance between classes must then be checked to see if the number of samples in one class differs by more than 10% from the other class. Then it performs a horizontal rotation over the class's images, which have fewer samples if they are not varying. Finally, it returns dataset directories; after creating rotated samples, it also returns dataset directories.

Algorithm 1: Data preprocessing
Data: Dataset D *Result: Preprocessed dataset*
for Image I in D do *for class Os.mkdir() do* *Move I into defined directory.* *If count of each class directory > 0.10 then* *for I in directory: do* *Cv2.horizontal rotate()* *Return D* *Return D* *D_test, D_ train = D.test_ train_ split*

Algorithm 2 shows the steps for recognising and detecting of the object. This algorithm uses a training set to extract features from I. Further, a model is created and summarised, and then training on the model begins. The model is saved in the current directory after training.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7390
Article Received: 29 April 2023 Revised: 24 June 2023 Accepted: 18 July 2023

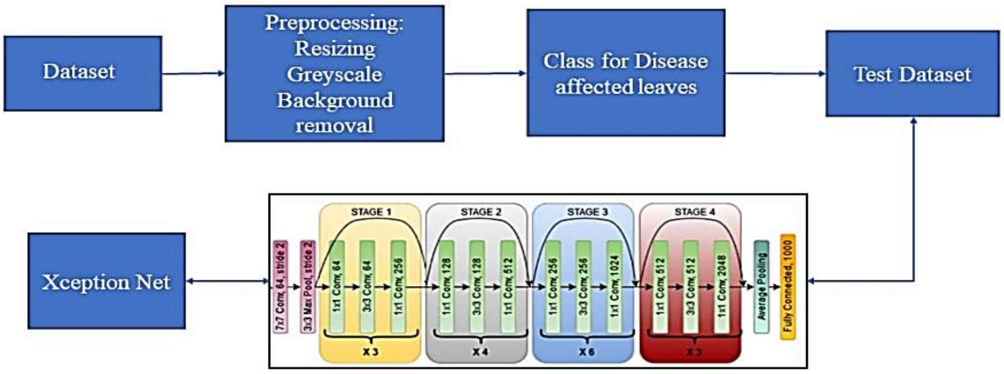

Figure 4: Flowchart of the proposed model

Figure 4 depicts how the dataset flow in the proposed model. The process begins with the preprocessing of the dataset by resizing the images, which is followed by gray scaling and background removal to extract regions, then classes for the images of leaf diseases taken as labels, which are taken from label text files, after which ResNet extracts features from the prepared images. And then, Xception makes the detection and returns a class for the image if the disease is present on the leaf.

Algorithm 2: Recognise and detection()
Data: Dataset D *Result: Trained Model File*
for Image I in dataset D_trained do *Frtlist=[]* *Frtlist.append(ResNet101(I))* *Model = Xception(1024, preordained.model, Frtlist)* *Model.compile(metrics = ['acc'])* *Model.Fit(Dtrain)* *Model.save(./)*

V. EXPERIMENTAL RESULTS

During the experiments, we evaluate how well the proposed m-Xception model performs in comparison to other models, such as MobileNet Sandler et al. (2018), MobileNet-V2 Sandler et al. (2018), Xception Sifre and Mallat (2014), NASNetMobile Zoph et al. (2018), EfficientNet-B0 Tan and Le (2019), and M-Inception Chen et al (2022).

5.1 Data Set

In our tests, we utilised both the Plant Village Hughes et al. (2015) dataset, which is accessible to the general public, as well as the plant picture dataset, which was collected locally. Both of these sets of data were amassed in the same geographic region. One of these is referred to as Plant Village, and it is tasked with the responsibility of amassing images of plant leaves from all around the world for the aim of validating disease detection algorithms. This dataset is comprised of a total of 54,305 images of plant leaves, all of which have been organised into a total of 38 distinct groups. There are a total of 14 different species, 12 healthy plants, and 26 ill plants included in this collection. Every single picture of a plant leaf was obtained through the use of photography, and the conditions for doing so were very particular. Lighting must have been consistent and there must have been a lack of obstructions in the background. It has been brought to our attention that multiple angles and distances have been used to capture the same leaf in photographs. To begin, while the model was being trained, the one-hot encoding of the category variable was completed successfully. Here we arrive at our second point. The photographs were then uniformly resized to 224 pixels on a side so that the model could be properly positioned within them.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7390
***Article Received**: 29 April 2023 **Revised:** 24 June 2023 **Accepted:** 18 July 2023*

Figure 5 (a) Dataset Samples (b) Performance Evaluation

Accuracy, Recall, Specificity, and F-score are some metrics used as evaluation indicators to determine how well various models performed. These metrics were derived from the statistics of correct and incorrect detections. To put this into a context that is easier to understand, these measurements and their enlarged computations for multi-class activities that use macro averages are written out as equations (3-7).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

$$precision = \frac{TP}{TP+FP} \quad (4)$$

$$recall = \frac{TP}{TP+FN} \quad (5)$$

$$Fscore = \frac{2*TP}{2*TP+FP+FN} \quad (6)$$

$$Specificity = \frac{TN}{FP+FN} \quad (7)$$

VI. Results and Discussion

In this section, we discuss the effectiveness of the proposed m Xception with the existing model.

Figure 6 shows the performance of the proposed model per epoch; the figure 6(a) graph shows loss reduction as the number of epochs increases, and higher values indicate that the losses are decreasing. And the 6(b) curve shows accuracy for training and validation; based on the above curves, it is evident that the model can overcome overfitting as the curves for accuracy for both training and validation move in the same rhythm.

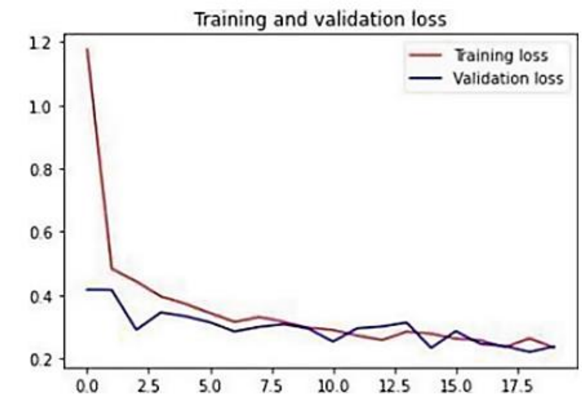

Figure 6 (a) Graphical representation of training and validation loss of the model

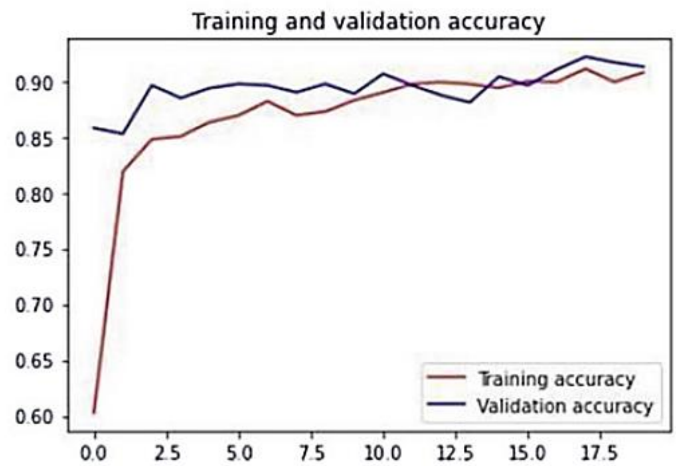

Figure 6 (b) Graphical representation of training and validation accuracy of the model

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7390
***Article Received**: 29 April 2023 **Revised:** 24 June 2023 **Accepted:** 18 July 2023*

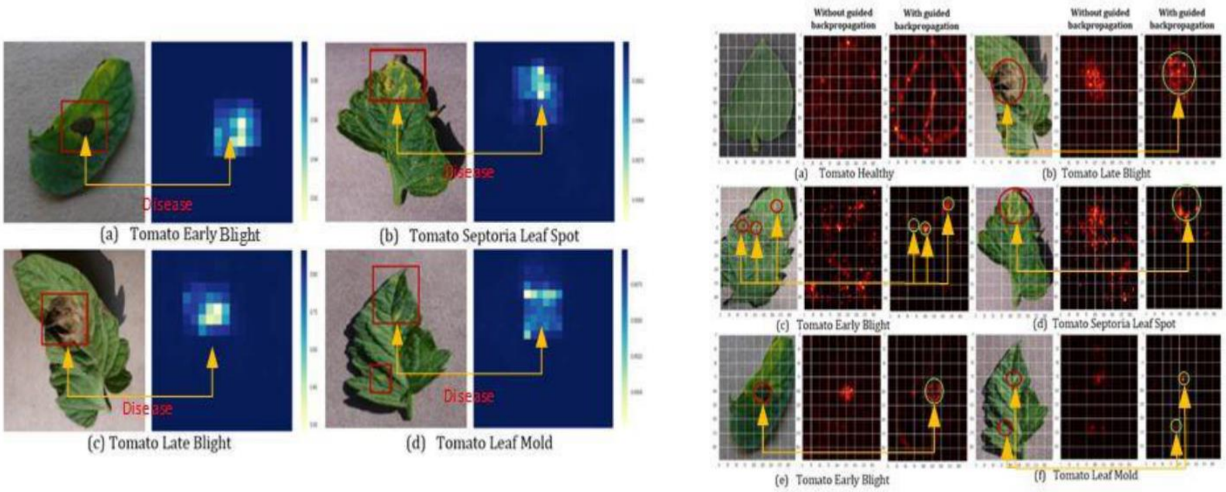

Figure 6. Disease Detection Samples

Table 1 compares the proposed model's performance on village data with the performance of known solutions depending on the number of epochs. Epochs refer to the number of iterations followed by training performed across the whole network. On both 10 and 30 epochs, the models included in this table were analysed and compared to the proposed model. Accuracy in training and validation and losses incurred during training and validation were utilised as metrics to evaluate the proposed model. It was noted from the table that the suggested model outperforms existing solutions in terms of accuracy and losses. This was found because the proposed model improved its accuracy while simultaneously reducing its losses when 10 epochs were considered. And when 30 epochs were considered, the proposed model also performed better than existing solutions, representing that the proposed model is also not inclined towards overfitting as the number of epochs increased, which resembles that model's ability to balance variance and bias. And when 50 epochs were considered, the proposed model performed better than the existing solutions

Models/methods	10 epochs				30 epochs			
	Trai ning acc. %	Valida tion acc. %	Train in g losse s	Valida ti on losses	Trai ning acc. %	Valida tion acc. %	Train in g losse s	Valid ati on losses
MobileNet [24]	75.6	75.4	88.14	88.41	73	71	83	82
MobileNet-V2 [24]	82	82	76.44	73	76	75	85	81
Xception [25]	83.24	84	73.1	71	81	79	81	75
NASNetMobile[26]	87.1	87	74	77	80	72	73	80
EfficientNet-B0[27]	88.54	88	89	82	81	78	77	75
modified Inception (M-Inception) [23]	94.11	95.32	71	73	82	81	79	72
Proposed method	**98.78**	**98**	**62.1**	**55**	**92**	**91**	**61**	**56**

Table 1: Comparative Analysis of the Proposed Method

VII. CONCLUSION

In this paper, the M-Xception model is proposed. Existing solutions use Inception as a classification module with mobile net, and to feature extract in the proposed model, they use a logarithmic-based softmax layer. These changes significantly reduce false positive responses, which ultimately increase true positives, which are reflected in the accuracy figures of the proposed model. In addition, the local experimental materials consist of photographs of various plants taken in realistic settings of wild fields, complete with various backgrounds and varying degrees of illumination intensity. Despite this, the experiment produced an acceptable conclusion, confirming the usefulness and practicality of the suggested technique for diagnosing crop diseases. Even though the circumstances of the setting were difficult, the experiment nevertheless had the potential to result in a valuable finding. Even if the model has a very high accuracy and memory efficiency, the performance of the model can still be further improved by performing additional fine-tuning operations on the backbone network. This is because the model's performance is directly related to the amount of memory the model uses. One way to accomplish this goal is to perform more activities on the backbone network. We have great expectations that in the not-too-distant future, we will be able to considerably increase the performance of the model while at the same time lowering its size of it. This goal will be attained by unfreezing more layers of bottom convolutional layers and fine-tuning the network architecture.

REFERENCES:

[1] Darwish, D. Ezzat, A. E. Hassanien, An optimized model based on convolutional neural networks and orthogonal learning particle swarm optimization algorithm for plant diseases diagnosis, Swarm and Evolutionary Computation 52 (2020) 100616.

[2] S. H. Lee, H. Go¨eau, P. Bonnet, A. Joly, New perspectives on plant disease characterization based on deep learning, Computers and Electronics in Agriculture 170 (2020) 105220.

[3] S. Radhakrishnan, An improved machine learning algorithm for predicting blast disease in paddy crop, Materials Today: Proceedings (2020).

[4] S. Hern´andez, J. L. L´opez, Uncertainty quantification for plant disease detection using bayesian deep learning, Applied Soft Computing 96 (2020) 106597.

[5] T. R. Gadekallu, D. S. Rajput, M. P. K. Reddy, K. Lakshmanna, S. Bhattacharya, S. Singh, A. Jolfaei, M. Alazab, A novel pca–whale optimization-based deep neural network model for classification of tomato plant diseases using gpu, Journal of Real-Time Image Processing (2020) 1–14.

[6] Z. Iqbal, M. A. Khan, M. Sharif, J. H. Shah, M. H. ur Rehman, K. Javed, An automated detection and classification of citrus plant diseases using image processing techniques: A review, Computers and electronics in agriculture 153 (2018) 12–32.

[7] J. G. A. Barbedo, Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification, Computers and electronics in agriculture 153 (2018) 46–53.

[8] Parjane, V. A. ., Arjariya, T. ., & Gangwar, M. . (2023). Corrosion Detection and Prediction for Underwater pipelines using IoT and Machine Learning Techniques. International Journal of Intelligent Systems and Applications in Engineering, 11(2s), 293 –. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/2626

[9] A. Abdalla, H. Cen, A. El-manawy, Y. He, Infield oilseed rape images segmentation via improved unsupervised learning models combined with supreme color features, Computers and Electronics in Agriculture 162 (2019) 1057–1068.

[10] Prof. Sharayu Waghmare. (2012). Vedic Multiplier Implementation for High Speed Factorial Computation. International Journal of New Practices in Management and Engineering, 1(04), 01 - 06. Retrieved from http://ijnpme.org/index.php/IJNPME/article/view/8

[11] J. G. Barbedo, Factors influencing the use of deep learning for plant disease recognition, Biosystems engineering 172 (2018) 84–91.

[12] D. Hughes, M. Salath´e, 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).

[13] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, NIPS'12, Curran Associates Inc., 2012, p. 1097–1105.

[14] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Computer Vision and Pattern Recognition (CVPR), 2015. URL http://arxiv.org/abs/1409.4842

[15] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., Imagenet large scale visual recognition challenge, International journal of computer vision 115 (3) (2015) 211–252.

[16] S. Zhang, S. Zhang, C. Zhang, X. Wang, Y. Shi, Cucumber leaf disease identification with global pooling dilated convolutional neural network, Computers and Electronics in Agriculture 162 (2019) 422–430.

[17] Z. Tang, J. Yang, Z. Li, F. Qi, Grape disease image classification based on lightweight convolution neural networks and channelwise attention, Computers and Electronics in Agriculture 178 (2020) 105735.

[18] Y. Li, J. Nie, X. Chao, Do we really need deep cnn for plant diseases identification?, Computers and Electronics in Agriculture 178 (2020) 105803.

[19] Z. Li, Y. Yang, Y. Li, R. Guo, J. Yang, J. Yue, A solanaceae disease recognition model based on seinception, Computers and Electronics in Agriculture 178 (2020) 105792.

ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7390
***Article Received**: 29 April 2023 **Revised:** 24 June 2023 **Accepted:** 18 July 2023*

[20] M. Agarwal, S. K. Gupta, K. Biswas, Development of efficient cnn model for tomato crop disease identification, Sustainable Computing: Informatics and Systems 28 (2020) 100407.

[21] W. Zeng, M. Li, Crop leaf disease recognition based on self-attention convolutional neural network, Computers and Electronics in Agriculture 172 (2020) 105341.

[22] J. Chen, W. Wang, D. Zhang, A. Zeb, Y. A. Nanehkaran, Attention embedded lightweight network for maize disease recognition, Plant Pathology 70 (3) (2021) 630–642.

[23] J. Chen, D. Zhang, A. Zeb, Y. A. Nanehkaran, Identification of rice plant diseases using lightweight attention networks, Expert Systems with Applications 169 (2021) 114514.

[24] C. Szegedy, Scene classification with inception-7, 2015.

[25] Asfaqur Rahman, M., Shahriar Nawal Shoumik, M., Mahbubur Rahman, M., Hasna Hena, M., 2021. Rice disease detection based on image processing technique, in: Smart Trends in Computing and Communications: Proceedings of SmartCom 2020, Springer. pp. 135–145.

[26] Singla, P., Sharma, R., Kukreja, V., Bansal, A., et al., 2022a. Deep learning based multi-classification model for rice disease detection, in: 2022 10th In ternational Conference on Reliability, Infocom Technologies and Optimisation (Trends and Future Directions)(ICRITO), IEEE. pp. 1–5

[27] Upadhyay, S.K., Kumar, A., 2022. A novel approach for rice plant diseases classification with deep convolutional neural network. International Journal of Information Technology 14, 185–199.

[28] A. Karlekar, A. Seal, Soynet: Soybean leaf diseases classification, Computers and Electronics in Agriculture 172 (2020) 105342.

[29] Sutaji, D., Yıldız, O., 2022. Lemoxinet: Lite ensemble mobilenetv2 and xception models to predict plant disease. Ecological Informatics , 101698.