Plant Disease Detection

Nikitha Pilli, Kompalli Jwala Manjunath, Anish Tavva

VIT-AP University

nikitha.20bci7019@vitap.ac.in, manjunath.20bci7329@vitap.ac.in, anish.20bci7027@vitap.ac.in

**Abstract-**

**Agricultural production is among the fundamental industries for human life. Simultaneously, the digitization of all disciplines facilitated the completion of several challenging jobs. Integrating innovation and digitalisation is essential for the agricultural industry to serve both farmers and consumers. Owing to the use of technologies and continuous monitoring, it’s indeed possible to discover illnesses at their early phases and remove them to increase crop output. Small - scale farmers have a recurring problem in the form of crop diseases, that affects their livelihood and food stability. Crop development and production are crucial factors that affect farming and farmers financially, socially, as well as in every manner imaginable. In order to detect agricultural illnesses at the opportune moment, it is required to conduct careful surveillance at different phases of crop development. Automatic detection and categorization of diverse agricultural diseases are required for reliable diagnosis under this regard. Deep learning’s CNN network category is mostly utilised for picture categorization. The suggested approach focused primarily upon this transfer learning phenomena using the pretrained ResNet-50 model, and then calculated the efficiency of the model using several quality assessment metrics. The primary objective of the proposed study is to discover a way to address the problem of detecting 38 distinct kinds of plant illnesses by using easiest technique and lowest computational resources to obtain better results than the standard models. This system’s precision value of 99.1% demonstrates the viability of CNN technique, even under adverse situations.**

**Key Words: *Agriculture*, CNN*, Deep Learning, Image Classification, Plant Disease, ResNet.***

**Introduction**

Approximately 58 percent of India’s populace relies on farming as their major source of income. India comes in 2nd worldwide with regards to agricultural output. In 2018, it was claimed that farming employed over than 50 percent of the workforce, thereby generating 18 to 20 percent of India’s GDP. Thus, India has proved to be among the most productive countries in terms of agricultural productivity and output. Due to the fact that the most of the population relies on farming, it’s indeed vital to realise the issues facing this industry. It therefore seems reasonable to conclude that if farming were to be revolutionised, it could tremendously help the nation & amp; in addition to improving the circumstances of farmers, it could also provide a large deal of jobs and growth chances with in farming sector. Agriculture is plagued by multiple issues, such as ineffective agricultural tactics and procedures, limited use of compost, vermicompost, and fertilisers, lack of water supply, different crop diseases, etc. [1]

Diseases are very detrimental to the health of plants, and that in turn affects them development. India has made significant strides in pesticide, fungicide, and herbicide advancement and research. But each year, owing to unknown factors, plants fall to a variety of recognised illnesses, resulting in the loss of countless tons of yield. The assault of such different forms of crop diseases causes a significant reduction of crop yield both qualitatively and quantitatively. Approximately 20- 30 percent of crop insufficiency may be attributed to diseased plants. In order to prevent huge production, efficiency, and overall agricultural loss, it is essential to identify crop illnesses. As manual identification is exceedingly time-consuming &amp; susceptible to mistake, it might result in incorrect diagnosis. Current technological advancements have made the detection and diagnosis of plant diseases conceivable and achievable, hence paving the road for improved plant management in the event that a plant is infected. The suggested approach for the identification of plant leaf diseases concentrates on 14 plant species. Image Classification, Voice Recognition, and Processing of Natural Language has all shown exceptional performance over recent years due to Deep Learning.[2] Utilizing a CNN to address the issue of identifying plant diseases yields excellent results. CNN is acknowledged as the most effective Object Recognition technique.

CNN are utilised for the creation of a predictive model that is operated on the input picture and changes the input in order to identify the output labels. The suggested system has two components. One is the acquisition of illness characteristics, while the other is their categorization. In this study, we have presented a method for detecting plant illnesses using photographs of their leaves. Image segmentation is a subfield of signal processing that extracts the image’s characteristics and other valuable details. Machine learning is a kind of artificial intelligence that performs tasks automatically or follows specific instructions. The primary objective of machine learning seems to be to comprehend data for training and incorporate it into models that will be helpful to humans. Using the vast quantity of training data, it may aid in making sound judgments and anticipating the proper outcome. For categorization, the factors of leaf colour, leaf damage, leaf surface area, and leaf texture are used. In this study, we analysed many picture metrics or features in order to accurately detect various plant leaf diseases. Previously, plant disease identification was performed by professionals by leaf examination or chemical methods. This requires a huge team of specialists in addition to continual plant monitoring, which is expensive for large fields.[2] In these kinds of settings, the proposed approach is useful for observing huge agricultural fields. Detecting plant illnesses automatically by observing the signs on the plant’s leaves makes the process simpler and less expensive. The suggested solution to crop disease diagnosis is substantially less costly and needs shorter effort for predicting than existing deep learning-based systems.

**Literature Review**

*1. Automated Image Capturing System for Deep Learning-based Tomato Plant Leaf Disease Detection and Recognition*[3]  
*Publication Year:* 2019  
*Author:* Robert G. de Luna, Elmer P. Dadios, Argel A. Bandala   
*Journal Name:* International Conference on Advances in Big Data, Computing and Data Communication Systems

This research created a new method for the effective identification of diseases in tomato plants. A motorised picture capture container was constructed to record four sides of each tomato crop in order to spot and identify crop disease. The test subject had been a particular kind of tomato called Diamante Max. The technique was developed in order to diagnose the illnesses Phroma Rot, Leaf Miner, as well as Target Spot. Collecting both damaged and regular leaves for data. Next, build a neural network using deep convolution to recognise 3 disorders. CNN was used by the system to determine what of the tomato illnesses are prevalent on the observed tomato plants. The F-RCNN-trained abnormality detection model obtained a confidence score of 80%, but the Transfer Learning illness identification model had a precision of 95.75 percent. The automatic picture capture system was applied in practise and achieved 91.67 percent precision in identifying leaf diseases on tomato plants.[3]

*2. CNN based Leaf Disease Identification and Remedy Recommendation System*[4]  
*Publication Year:* 2019  
*Author:* Sunku Rohan, Triveni S Pujar, Suma VR Amog Shetty, Rishabh F Tated   
*Journal Name:* IEEE conference paper

Farming has a significant effect on our lives. Our country's economic most vital segment is farming. It is challenging for farmers to recognise plant diseases, resulting in decreased agricultural output. Nonetheless, recordings and photographs of leaves that offer agricultural experts with a better perspective may yield a superior answer to ensure that the issue associated with plant disease may be resolved.  Due to technological advancement and innovation, equipment is now able to identify and diagnose crop illnesses. Recognize illnesses sooner in order to reduce their detrimental effects on the crop. This research focuses on the identification of crop diseases utilizing image processing methods. This work accesses an accessible dataset including 5000 photos of normal and sick plant leaves, and then uses semi-supervised algorithms to identify 4 kinds of disease.[4]

*3. Real-Time Detection of Apple Leaf Diseases Using Deep Learning Approach Based on Improved Convolution Neural Networks*[5]  
*Publication Year:* 2019  
*Author:* Bin Liu, Peng Jiang, Yuehan Chen, Dongjian He, Chunquan Liang  
*Journal Name:* IEEE ACCESS

This article discusses 5 apple leaf diseases, including brown spot, mosaic, aria leaf spot, rust and grey spot that has an impact on apple Deep learning approaches were employed in this study to enhance CNNs for detecting apple leaf illnesses. In this publication, the apple leaf disease dataset (ALDD) is utilised, that consists of complex images as well as lab images, and also the remainder is built through the use of image annotation and data augmentation innovations in order to develop a new apple leaf disease detection model which utilizes deep-CNNs by employing Google Net Inception structure and Rainbow concatenation. In test data consisting of 26,377 photos of apple leaf illnesses, the suggested INAR- system is trained to identify 5 prevalent apple leaf illnesses using the test dataset. Experimentally, the INAR-SSD model achieves a detection accuracy of 78.80%, with a high detection rate of 23.13 FPS. Findings suggest that innovative INAR-SSD model offers a high-performance approach for earlier detection of apple leaf illnesses which can identify various illnesses in real-time with more precision and quicker detection speed than earlier techniques.[5]

*4. Identification of plant leaf diseases using a nine-layer deep convolution neural network*[6]  
*Publication Year:* 2019  
*Author:* Geetharamani G., Arun Pandian J.  
*Journal Name:* Computers and Electrical Engineering 76 (2019)

In this article, we identify plant leaves diseases utilizing deep learning in CNN. The CNN model has been trained utilising more than 39 distinct classes of plant leaf diseases and backdrop photos from an accessible dataset. including six kinds of data augmentation techniques and utilised for gamma correction, picture flipping, principal component analysis (PCA) colour augmentation, rotation etc.   Whole are aware that data enhancement is used. This may enhance the model's performance. Various training ranges of epochs, batch sizes, and dropouts were used to develop the model. The suggested model delivers superior outcomes in comparison with CNN and transfer learning techniques when utilising the validation information. Despite simulation, the suggested model obtains a classification performance of 96.46 percent. The precision of CNN is superior than that of transfer learning techniques.[6]

*5. A Segmentation Improved Robust PNN Model for Disease Identification in Different Leaf*[7] *Publication Year:* 2016  
*Author:* Rekha Chahar, Priyanka Soni  
*Journal Name:* IEEE International Conference.

This article offers Farming Images of vegetables, fruits, plants, and flowers, as well as those leaves diseases. The detection of the agricultural product type connected with the illness. Such illnesses are particular to the product's roots, seeds, or leaf components. This really is important for providing illness detection from a distant lab. This task is separated into 2 phases. In the initial phase, a classification model is developed based on the ring project is built to examine the characteristics of leaf pictures. Once the traits are discovered, the PNN classifier is used to determine the disease's presence. The purpose of this study is to determine the health and infectious illness depending on area identification. The research is conducted using arbitrarily obtained leaf photos from internet for various plants.[7]

**Dataset**

*Description of the dataset*

These images were taken from the “new plant disease dataset” that was just recently published on Kaggle and are related to plant diseases.[8] When recreating this dataset using offline augmentation, the original dataset, which was known as the plant village dataset, served as the point of departure. Over eighty-seven thousand RGB photographs of plant leaves, depicting both healthy and diseased conditions, are included in this dataset. These photographs have been organized into a total of 38 different classifications for your viewing pleasure. There are a total of 14 different kinds of plants that are taken into consideration within the parameters of this dataset. The following plants are taken into consideration in this dataset: orange, grape, raspberry, tomato, peach, apple, cherry (including sour), soybean, pepper bell, corn (maize), potato, blueberry, squash, and strawberry. This particular dataset takes into account a total of 26 distinct diseases that can be found in plants.

*Dataset preparation*

After importing the data, we normalized it by converting the pixel values of each picture, which ranged from 0 to 255, into 0-1 so that neural networks could work more effectively with it. This allowed neural networks to better recognize images. Following the completion of the conversion of all of the pixel values to the torch tensor, the complete array of pixel values is then divided by 255. We used a batch size of 32 when we were training the model, and that was our starting point. In addition to that, we shuffled the records in the dataset. It is helpful so that batches that were produced during different epochs do not appear to be the same as one another. If we continue to act in this manner, ultimately our model will become more robust.

*Dataset formulation*

The dataset was first prepared in accordance with the instructions provided in Section 3.1, and then it was immediately fed into the model after being partitioned into training, validation, and test sets. After this, the model was run. The entire dataset is then proportionately divided into a training set and a validation set in the ratio of 80/20, and the directory structure is preserved throughout the process. In the end, in order to make a prediction, a new directory that contains 33 test images was established. This was done for the purpose of making a prediction.

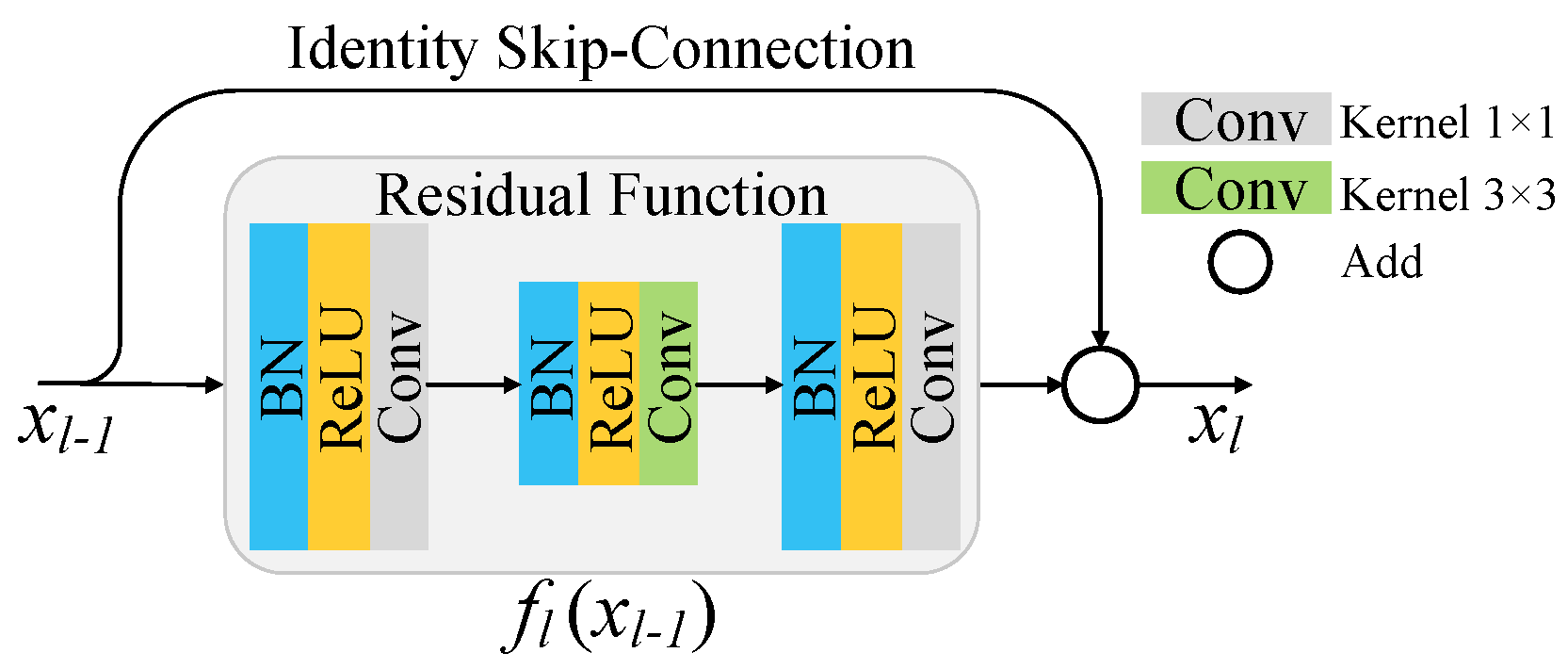
**Proposed work**

*Model used*

For the categorization of plant leaf diseases, we employed a Residual Neural Network, more often referred to as ResNet.[9] ResNet is a kind of artificial neural network. It represents a gateless / open-gated form of Highway Net, which was the initial operational extremely deep feedforward neurological network with 100s of layers. Prior neural networks were just a few layers deep, compared to the Highway Net’s 100s of layers. In Residual neural networks, in contrast to traditional networks, each and every layer feeds into the succeeding layer.[2] We employ a neural network that has residual blocks, so that each layer feeds into the next as well as directly into the layers that are about 2–3 hops away. This helps prevent over-fitting by preventing the network from learning too much information too quickly. The majority of Residual neural network models are constructed using double or triple layered skips which include nonlinearities (ReLU) with batch normalization between the layers. These are the most common types of ResNet models. There have been 2 primary reasons in order to add skip connections: either to prevent the issue of vanishing gradients, which leads to simpler to optimise networks, in which the gating mechanisms promote flow of information throughout many layers ("information highways"), or to alleviate the Deterioration (precision saturation) problem, which occurs when adding additional layers to a sufficiently deep model leads to increasing error in training. The network is substantially simplified by skipping, resulting in the use of reduced layers in the early phases of training. This expedites the learning process by lessening the influence of vanishing gradients, since there are lesser layers for the information to flow through. As it continues to understand the feature space, the neural network will then progressively recover the levels that it skipped. At the very conclusion of training, when each of the layers have been extended, it begins to remain closer towards the manifold, and as a result, it understands more quickly. A network that does not include any leftover pieces is able to explore a greater amount of feature space. Because of this, it becomes more susceptible to disturbances that lead it to depart the manifold, and it requires additional training data in order to recover from them. Within the scope of our project, we used ResNet9. [10]

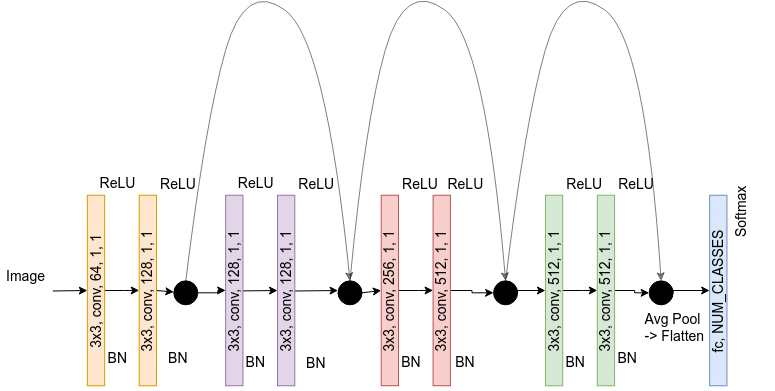
*Residual Block*

**Residual Blocks are a kind of skip-connection blocks which, as opposed to learning unreferenced functions, are trained to understand residual functions with regard towards the layer inputs. They were included into the ResNet design as a necessary component. The concept of residual blocks may be thought of as a subcategory of highway networks that do not have any gates within its skip connections. In its most basic form, residual blocks make it possible for memories or knowledge to go from the first to the final layer. In actuality, the performance of residual networks is comparable to that of any other highway network, regardless of the fact that skip links do not have any gates. This paper's methods are built on residual blocks as the underlying building block. When constructing the ResNet, the dimensional CNN layers were utilised alongside 3 distinct filters. It was decided to include a 1x1 kernel for filters f1 and f3.[11] On the contrary, the f2 filter made use of a kernel that was 3x3 in size. In every instance, the deployed stride was indeed a 1x1 configuration. Each and every residual block may be broken down into 3 primary steps. The levels are nearly identical to one another, with the exception of the first level, which includes a 2x2 max pooling layer immediately following the very first Conv2D layer. This is a straightforward example of a residual block:**



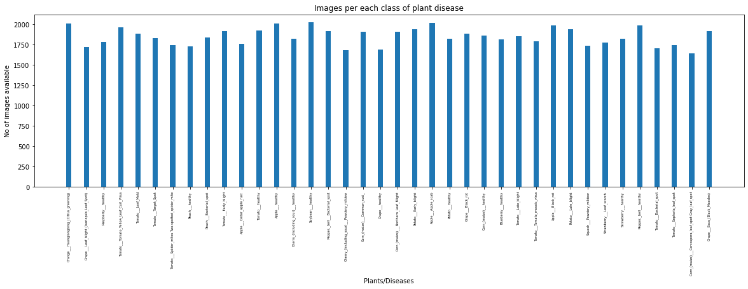
*Model architecture*

Architecture based on ResNet9 has been used. The ResNet9 network has nine levels and skip connections between them. It is made up of a total of 8 convolution layers, and there is a skip block inserted in between each pair of layers. The ReLU activation layer and the batch normalization layer were inserted into the gap that was left between the two subsequent Conv2D layers. Before going on to the next stage, the results of the previous stage are attached to the image that was started with, and then they are processed by a ReLU activation layer. Only once this is done are we able to proceed to the next stage.[9]



**Methodology**

We began by loading necessary modules such as os (for interacting with files), NumPy (for numerical calculations), pandas (for working with data frames), torch (for pytorch module), and matplotlib (for charting information). Google Colab was used throughout the whole project. The dataset was uploaded to Google Colab by first transferring the zip file to Google Drive and then mounting it to Google Colab. We then unzipped the file via Google Colab since it is considerably quicker than manually unzipping the file. Initially, we analysed the data and determined that we had images of the leaves of 14 plants, eliminating healthy leaves, and 26 kinds of images that depict a specific plant illness. Additionally, the dataset is balanced across all 38 classes. In all, 70295 pictures were utilized to train the ResNet model.



The second phase was Data preparation. The image data were imported using torchvision. datasets. ImageFolder was then used to convert the pixel values (0-255) of each picture to 0-1, since neural networks operate best with normalized input. The full pixel array is transformed to torch tensor & subsequently split by 255. The shape of the picture is (3,256,256), where 3 represents the no of channels (RGB) whereas 256 x 256 represents the image's breadth and height. In forward propagating of a CNN, the batch length is the no of pictures delivered as input all at once. Batch size essentially specifies the number of images which would be transmitted over a network. Here, a batch size of 32 was chosen. The following image shows the photos used for the first round of training.



The third phase is to model the convolutional neural network.  GPUs are recommended when working with image datasets as CPUs are best for general-purpose processing, whereas GPUs are suited for training deep learning as they can perform multiple calculations simultaneously. They feature a huge number of cores, that improves the parallel processing of various operations. In addition, deep learning calculations must manage enormous volumes of data, making GPU storage bandwidth optimal. To utilize a GPU effortlessly, if any is available, we write two helper methods (get default device & to device) and also a helper class (DeviceDataLoader) to transport our model as well as data to the GPU as needed. Then, we concluded training & validation data loaders by using DeviceDataLoader to autonomously transport batches of images to the GPU.

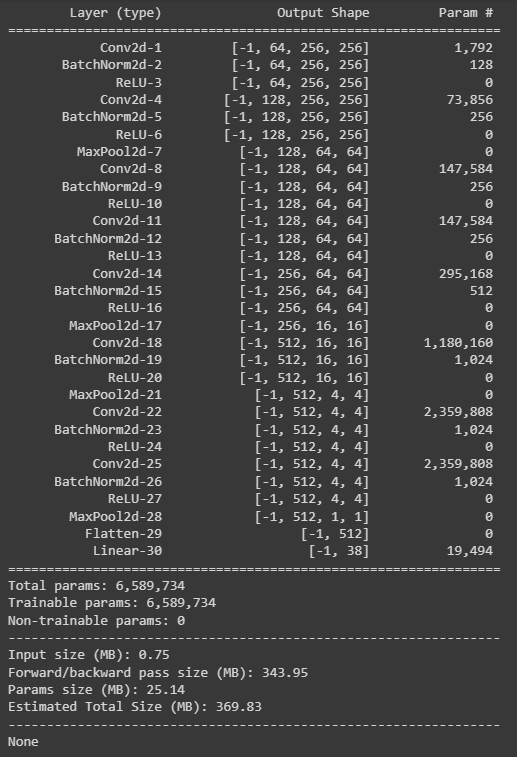
The next stage is to construct the model architecture. We used ResNet (particularly ResNet9), which has been one of the most significant advances in computer vision since its introduction in 2015. After implementing the residual block, we defined our ImageClassificationBase class with the following functions:

1. training step - Determine how "wrong" the model is proceeding following the validation or training step. This function is used in addition to an accuracy measure that is unlikely to being differentiable (this means that the gradient cannot be calculated, which is required for the model to progress over training).

A cursory examination of the Pytorch documentation reveals this cost function which is: Cross Entropy.

1. validation step - Just because an accuracy measure cannot be utilized during model training does not imply that it should not be implemented! In this instance, accuracy measure would be determined by a threshold & tallied if the difference amongst the model's predictions & the true label is less than the threshold• validation epoch end - We want to monitor the validating losses or accuracies as well as train losses following each and every epoch, and every time we do so, we must ensure that the gradient isn't getting tracked.
2. validation\_epoch\_end - Following every epoch, we intend to monitor the validating loss/accuracy and training loss, & every time we do so, we must ensure that the gradient isn't getting tracked.
3. epoch\_end - We additionally wish to output validation loss/accuracy, training loss, & learning rate following every epoch since we employ a learning rate scheduler whose function is to modify the learning rate following every training batch.

We also built an accuracy function that computes the accuracy of the employed model on complete set of outputs such that it could be used as a measure in fit one cycle.

The implemented architectural design is detailed below

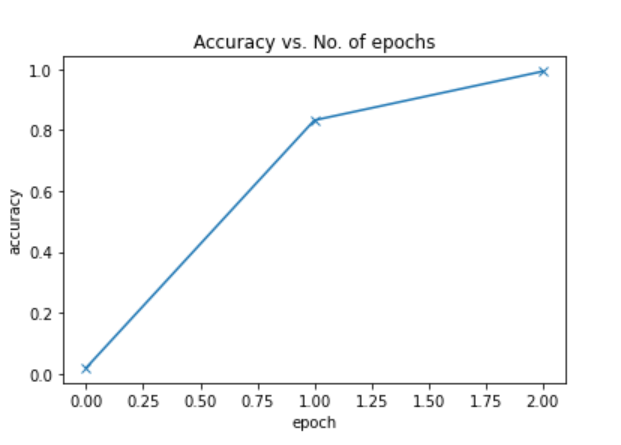
The last phase is training the model. Before training the model, we established two utility functions: evaluate, which will execute the validation step, and fit one cycle, which will carry out the complete training procedure. In fit one cycle, the following strategies have been employed:

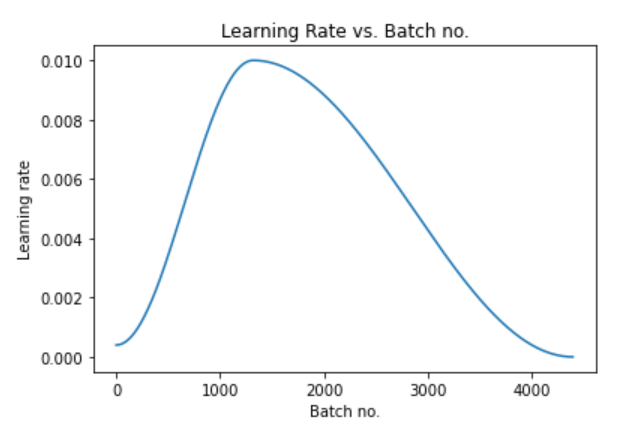
1. Learning Rate Scheduling: Rather than employing a set learning rate, we would implement a learning rate scheduler that modifies the learning rate following each training batch. There are numerous strategies for changing the learning rate during the course of training; the one we'll be using is called the "One Cycle Learning Rate Policy," which involves beginning with a small learning rate, slowly increasing it with every batch to a large learning rate for approximately 30% of epochs, & then gradually reducing it to a very small value for the rest of the epochs.
2. Weight Decay: Additionally, we utilize weight decay, a regularization strategy that stops the weights against getting too big by introducing an extra term to loss function.
3. Gradient Clipping: In addition to layer weights & outputs, it is useful to restrict gradient values to a restricted range to minimize unintended parameter changes caused by big gradient values. This simple but powerful method is known as gradient clipping.

In addition, we documented the learning rate employed by each batch. Then, we trained the model using 2 epochs.

**Results**

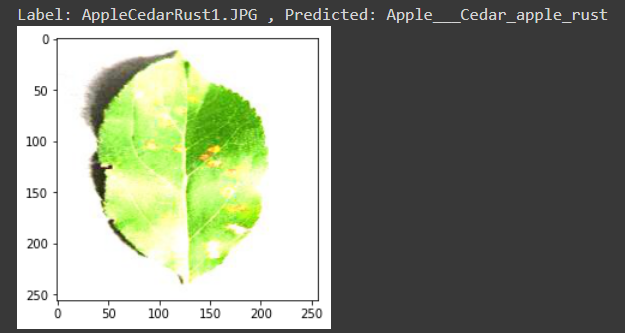
We utilized 2 epochs to train the model, with each epoch taking approximately 16 minutes, for a total runtime of 32 minutes. After training, we examined the precision of the validation data and found that it was 99.1% accurate. Let us now examine the graphs to better comprehend the results.



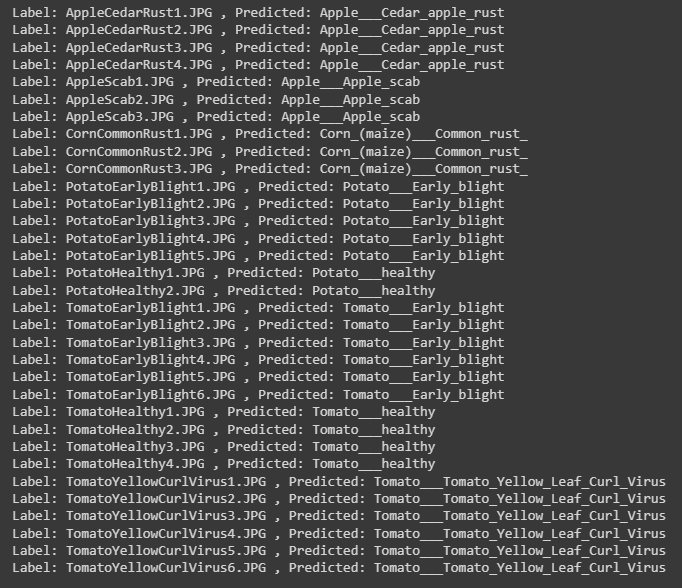


Following this, we provided the test dataset as an input for the disease prediction. The sample dataset contained 33 images. Due to the fact that we only have 33 images in our test data, we evaluated the model on all images. As you can see below, the model accurately predicted every test image.

First, we took a single leaf image from the test dataset and gave it as an input to the model for predicting. As you can see below, the model accurately predicted for that particular image as you can see below:



Due to the fact that we only have 33 images in our test data, we then evaluated the model on all images. As you can see below, the model accurately predicted every test image.



**Conclusion**

Humans have analysed and created plant-based food items for fibre, medicine, and other uses for generations. Crop diseases are among the numerous risks which should be addressed while farming crops. Therefore, it is essential that we improve food quality & maintain a steady agriculture sector, since this protects the food security of the country. There has been Widespread usage of Convolutional neural network techniques in the identification of plant diseases. Conventional object detection & categorization issues have been resolved with the help of CNN.

Small scale farmers rely on early and precise plant disease detection to avert losses. In this research, a pre-trained CNN was tweaked, and the ResNet 9 architecture was used to deploy the model. New plant disease dataset was used for classifying and training the model. ResNet9 was used as it is a relatively small architecture which reduced the computational time significantly without compromising the accuracy. The objective of the "Plant Disease Detection Using CNN" research is to develop a neural network competent of recognizing 14 crop species and 26 prevalent illnesses. When verified in a controlled setting, an overall accuracy of 99.2% is shown. This attained precision is contingent on a variety of criteria, including illness stage, disease kind, background data, and object composition. Since the model is trained with a simple backdrop and a single leaf, it is ideal to imitate these characteristics. In this instance, augmentation and transfer learning were advantageous to the model, making the CNN's generalizations more reliable. Nevertheless, it is crucial to diversify training data by including alternate background data, new plant architecture, and other disease stages.

This research demonstrates conclusively that CNNs may be used to strengthen small-scale farmers with their battle against plant disease. Future efforts should concentrate on diversifying training data. Without these innovations, the fight with plant diseases will continue. Expenses will be reduced by eliminating the needless administration of fungicide/pesticide/herbicide if a more effective method of seeing diseased spots on crops is used. The intensity of crop diseases varies over time; hence, CNN models must be enhanced/modified to identify and categorize diseases throughout their whole occurrence cycle. The CNN model/architecture must be effective across a variety of light circumstances; hence the datasets must not only represent the actual environment, but also include photographs captured in various field scenarios. A detailed investigation is necessary to comprehend the aspects that influence the identification of plant illnesses, such as the categories and quantity of datasets, the learning rate, as well as the amount of light.

**References**

[1] N. M. madhav and U. Scholar, “Plant Disease Detection Using Deep Learning,” *International Research Journal on Advanced Science Hub (IRJASH) Special*, vol. 03, 2021, Accessed: Nov. 25, 2022. [Online]. Available: www.rspsciencehub.com

[2] L. Poole, D. B.- Intelligence, B. Data, C. and Data, and undefined 2021, “Investigating popular cnn architectures for plant disease detection,” *ieeexplore.ieee.org*, doi: 10.1109/icABCD51485.2021.9519341.

[3] R. G. de Luna, E. P. Dadios, and A. A. Bandala, “Automated Image Capturing System for Deep Learning-based Tomato Plant Leaf Disease Detection and Recognition,” *IEEE Region 10 Annual International Conference, Proceedings/TENCON*, vol. 2018-October, pp. 1414–1419, Feb. 2019, doi: 10.1109/TENCON.2018.8650088.

[4] V. Suma, R. A. Shetty, R. F. Tated, S. Rohan, and T. S. Pujar, “CNN based Leaf Disease Identification and Remedy Recommendation System,” *Proceedings of the 3rd International Conference on Electronics and Communication and Aerospace Technology, ICECA 2019*, pp. 395–399, Jun. 2019, doi: 10.1109/ICECA.2019.8821872.

[5] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang, “Real-Time Detection of Apple Leaf Diseases Using Deep Learning Approach Based on Improved Convolutional Neural Networks,” *IEEE Access*, vol. 7, pp. 59069–59080, 2019, doi: 10.1109/ACCESS.2019.2914929.

[6] G. Geetharamani and A. P. J., “Identification of plant leaf diseases using a nine-layer deep convolutional neural network,” *Computers & Electrical Engineering*, vol. 76, pp. 323–338, Jun. 2019, doi: 10.1016/J.COMPELECENG.2019.04.011.

[7] P. Soni and R. Chahar, “A segmentation improved robust PNN model for disease identification in different leaf images,” *1st IEEE International Conference on Power Electronics, Intelligent Control and Energy Systems, ICPEICES 2016*, Feb. 2017, doi: 10.1109/ICPEICES.2016.7853301.

[8] “New Plant Diseases Dataset | Kaggle.” https://www.kaggle.com/datasets/vipoooool/new-plant-diseases-dataset (accessed Dec. 20, 2022).

[9] “plant disease detection resnet - Google Scholar.” https://scholar.google.com/scholar?hl=en&as\_sdt=0%2C5&q=plant+disease+detection+resnet&btnG= (accessed Dec. 20, 2022).

[10] T. Subetha, R. Khilar, M. C.-M. T. Proceedings, and undefined 2021, “A comparative analysis on plant pathology classification using deep learning architecture–Resnet and VGG19,” *Elsevier*, Accessed: Dec. 20, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S221478532039670X?casa\_token=7Jfi9TwMmIMAAAAA:RyvNmyVUnzvYsFJJoUp3ncdDNgrYx0Y0JnfDMGtsZdkvOTXX6ki3NvfjP3FPepv5UefrZO1OhqA

[11] S. Vallabhajosyula, V. Sistla, V. K.-J. of P. D. and, and undefined 2022, “Transfer learning-based deep ensemble neural network for plant leaf disease detection,” *Springer*, Accessed: Dec. 20, 2022. [Online]. Available: https://idp.springer.com/authorize/casa?redirect\_uri=https://link.springer.com/article/10.1007/s41348-021-00465-8&casa\_token=RJkSRr7UjbUAAAAA:rSlBlhEA\_klZgGgZZu9OkG17yBzNxSz4etM8vJ20F4iPN7FUmKbWMscmhwIUc2-NsIyFUlWTMpp86pXg0w