An AI Approach for Disease Detection in Plants

Himanshu Gajera[1], Parth Trada[2]

[1]*Student, Babaria Institute of Technology (BITS edu campus), Vadodara, India*
[2]*Student, Dharmsinh Desai University (DDIT), Nadiad, India*

Abstract- *Modern technologies have given human society the ability to produce enough food to meet the demand of more than 7.8 billion people. Grocery stores may seem like a cornucopia, offering every imaginable fruit, veggie, cereal and meat. But a report suggests that our food comes from just a few crops and types of livestock. And that could spell trouble if some crops develop disease over time that can lead to a disastrous food shortage. We evaluate the applicability of deep convolutional neural networks for the image based detection of disease sometimes even before the symptoms start to appear.*

Key Words **:** Machine Learning, Plant disease detection, DenseNet-121[1], Image Processing, Deep Learning

1. INTRODUCTION

Various efforts have been developed to prevent crop loss due to diseases. Historically, disease identification has been supported by agricultural extension organizations or other institutions, such as local plant clinics. In more recent times, such efforts have additionally been supported by providing information for disease diagnosis online, leveraging the increasing Internet penetration worldwide. Even more recently, tools based on intelligence systems have evolved and contributed to the process. Computer vision, and object recognition in particular, has made tremendous advances in the past few years. Deep neural networks have recently been successfully applied in many diverse domains as examples of end to end learning. Neural networks provide a mapping between an input(such as an image of a diseased plant) to an output(such as a crop-disease pair). In order to develop accurate image classifiers for the purposes of plant disease diagnosis, we needed a large, verified dataset of images of diseased plants. To address this problem, the Plant Village project has begun collecting tens of thousands of images of healthy and diseased crop plants and has made them openly and freely available. Here, we report on the classification of diseases in 15 crop species using 20,639 images with a convolutional neural network approach. An accuracy over 90% were reached in the detection on various plants. Comparative results of DenseNet model are presented.

1.1 Background

The problems arise with CNNs when they go deeper. This is because the path for information from the input layer until the output layer (and for the gradient in the opposite direction)becomes so big,that they can get vanished before reaching the other side.

DenseNet was developed specifically to improve the declined accuracy caused by the vanishing gradient in high-level neural networks. In simpler terms, due to the longer path between the input layer and the output layer, the information vanishes before reaching its destination.

Densely Connected Convolutional Networks, DenseNets, are the next step on the way to keep increasing the depth of deep convolutional networks. DenseNet is one of the new discoveries in neural networks for visual object recognition. DenseNet is quite similar to ResNet with some fundamental differences. ResNet uses an additive method (+) that merges the previous layer (identity) with the future layer, whereas DenseNet concatenates (.) the output of the previous layer with the future layer.

1.2 DenseNet Structure

DenseNet falls in the category of classic networks. This image shows a 5-layer dense block with a growth rate of k = 4 .

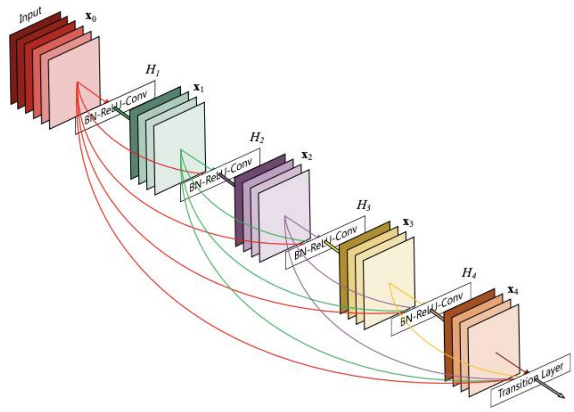


Fig: 1 DenseNet architecture

An output of the previous layer acts as an input of the second layer by using composite function operation. This composite operation consists of the convolution layer, pooling layer, batch normalization, and non-linear activation layer. These connections mean that the network has L(L+1)/2 direct connections. L is the number of layers in the architecture.

The DenseNet has different versions, like DenseNet-121, DenseNet-160, DenseNet-201, etc. The numbers denote the number of layers in the neural network. The number 121 is computed as follows:

DenseNet-121
5+(6+12+24+16) * 2 = 121
5 - Convolution and Pooling Layer
3 - Transition layers (6, 12, 24)
1 - Classification Layer (16)
2 - DenseBlock (1x1 and 3x3 Convolution)

1.3 DenseBlocks and Layers

Be it adding or concatenating, the grouping of layers by the above equation is only possible if feature map dimensions are the same. What if dimensions are different? The DenseNet is divided into DenseBlocks where a number of filters are different, but dimensions within the block are the same. Transition Layer applies batch normalization using downsampling; it's an essential step in CNN.Let's see what's inside the DenseBlock and transition layer:

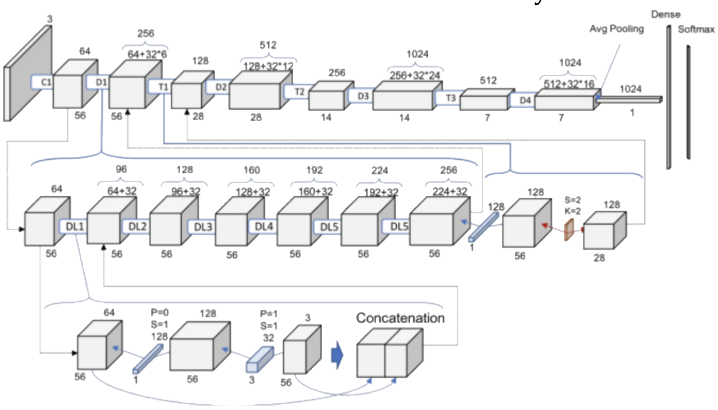


Fig: 2 DenseBlocks and Layers of DenseNet

The number of filters changes between the DenseBlocks, increases the dimensions of the channel.
The $k^{[l]} = (k^{[0]} + k(l-1))$ growth rate (k) helps in generalizing the l-th layer. It controls the amount of information to be added to each layer;

1.4 Advantages of DenseNet

- **Strong Gradient Flow:** The error signal can be easily propagated to earlier layers more directly. This is a kind of implicit deep supervision as earlier layers can get direct supervision from the final classification layer.
- **Parameter & Computational Efficiency :** For each layer, number of parameters in ResNet is directly proportional to C×C while Number of parameters in DenseNet is directly proportional to l×k×k.Since k<<C, DenseNet has much smaller size than ResNet.
- **More Diversified Features:** Since each layer in DenseNet receive all preceding layers as input, more diversified features and tends to have richer patterns.
- **Maintains Low Complexity Features:** In DenseNet, classifier uses features of all complexity levels. It tends to give more smooth decision boundaries. It also explains why DenseNet performs well when training data is insufficient.

2. DEEP LEARNING APPROACH

2.1 Dataset

A huge data set consisting of plant images has been taken with various imaging conditions from Kaggle. There are 15 different categories in which tomato, potato and other plant's diseases are described as features of our dataset. Fig.3. Labeling of images is done with an id describing diseases types.

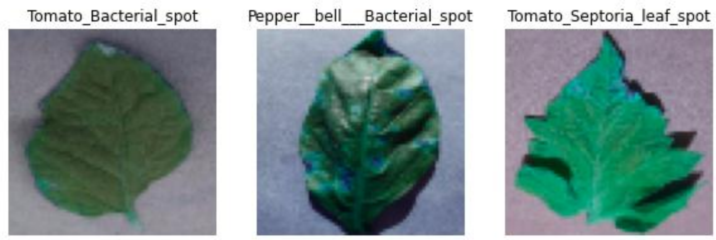


Fig: 3 Training Samples

2.2 Preprocessing

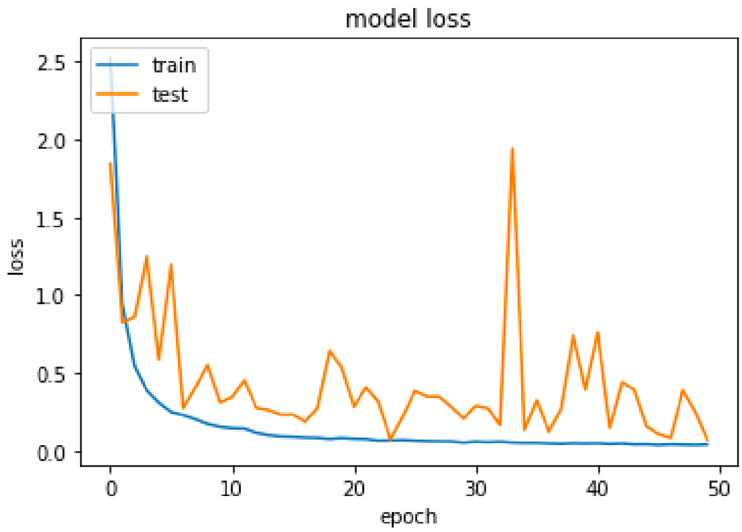
Here, the data set was taken from an online platform named Kaggle. The size of the data set is 20639 images. Before feeding the input directly into the model, the data which is the set of images must undergo some preprocessing which include resizing of images size from 256 *256 to 64 * 64 dimension. That is Training dataset, the dataset that is used to train or exercise the model. This data is labeled data set.Training data helps to minimize the loss function. Updating weights happens accordingly when the training data set is exercised in the model but validation data set does not involve any updating process. Training dataset is labeled while testing dataset is unlabelled. Also, one hot encoding is performed on the training features as diseases types.

2.3 Model Architecture

The architecture consists of mainly three parts: Input layer, DenseNet-121 structure and Output Layer. Input Layer consists of 64 * 64 neurons which is equal to the count of pixels of each individual image being passed. Here, the pixel values of the training images are being passed to the input layer. Convolutional layer 1 consists of 3 neurons. The neurons are densely connected to the neurons in the previous layer. Convolution is performed on the input pixels, which is a process of performing dot product on the pixel values with arbitrary numbers called as filters. So, the layer's output is further passed to the DenseNet-121. With the filters provided convolution operation in DenseNet-121 is performed on the received input with total 7037504 parameters and 0.5 dropout. The max pooling layer performs the max pooling operation on the received input. Then the output of the max pooling layer is flattened. Flattening is a process of converting any matrix into one dimensional array. Flatten function is applied on the convolution layer to create a single long feature vector. The total amount of neurons existing in the output layer is equal to the number of diseases. The neuron consisting of the maximum value ranging between 0-1 will be the classified disease for the given input plant sample. This result will be compared with the actual values and the error is determined. Based on the error the model tunes its underlined parameters such that the error is as minimum as possible. This operation is performed on each and every training image.

2.4 Accuracy and Loss for training and testing

We trained DenseNet-121 model with over 7.1 million parameters on 80% of our main dataset and performed testing on 20% of our dataset. We achieved almost 98% accuracy on testing phase which is quite larger than normal convolutional neural network. Also, DenseNet has less parameters compare to ResNet (25 million parameters) and due to this it clearly reduces the complexity of deep learning model as compare to any residual neural network. Performing 50 epochs of training on the above mentioned set of data, accuracy and loss are compared as :



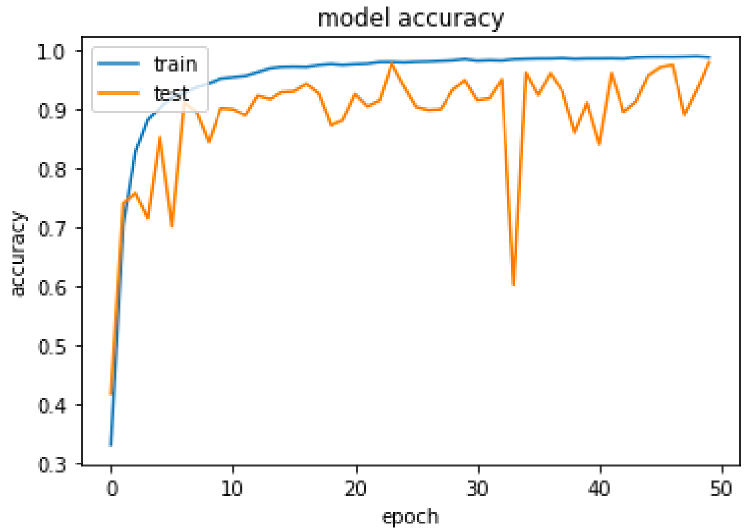


Fig: 4 Loss and Accuracy graph on training and testing dataset

3. RESULT AND ANALYSIS

After performing testing, we verify the deep learning model on several images which is not the part of dataset. Figure 5 is one of the samples from these analysis:

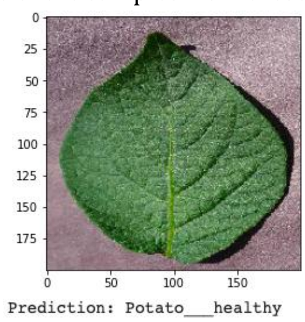


Fig: 5 Prediction Sample

DL Architectures/Models	Dataset	Accuracy
CNN	PlantVillage	92.85% [11]
AlexNet, GoogLeNet, ResNet	PlantVillage	97.28% [12]
Super-Resolution Convolutional Neural Network (SCRNN)	PlantVillage	~90% [13]
AlexNet and SqueezeNet v1.1	PlantVillage	95.65% [14]
DCNN, Random forest, Support Vector Machine	PlantVillage	93.4% [15]

Fig: 6 Comparison with other Deep Learning models

4. CONCLUTION

The use of automation and digital systems are gaining increasing demand with the technological advancement. In agricultural field loss mainly occurs due to widespread of disease. Mostly the disease are being noticed or identified when it advances to severe stage. Therefore, it causes the loss of assets like crop, time and money. This study summarizes the major image processing used for identification of leaf diseases. This approach can significantly support an accurate detection of leaf disease. The proposed system is capable of detecting the disease at the earlier stage as soon as it occurs on the leaf. By identifying the disease in the early stage, a certain amount of pesticides and fertilizers can be used as rectifying steps. Using this concept the disease identification is done for all kinds of leaves and also the user can know the affected area of leaf in some quantity. Hence saving the loss and reducing the dependency on the agricultural expert to a certain extent is possible.

5. FUTURE WORK

Farming is the necessity from ancient times till now. Humans only get to eat due to that and disease in crops is one such obstacle to the process. So the described system can be deployed with more diversified architecture on any digital platform in order to make things easier for local farmers by reducing their compulsion to take everything to the laboratories. This approach can help save their time and money and more importantly the crops.

REFERENCE

[1] G. Huang, Z. Liu and L. van der Maaten, "Densely Connected Convolutional Networks," 2018.

[2] [2017 CVPR] [DenseNet] Densely Connected Convolutional Networks

[3] Burdon J.J. 1987. Diseases and Plant Population Biology. Cambridge University Press, Cambridge.

[4] Hartl, D. L. and A. G. Clark, 1997. Principles of Population Genetics. Third Edition. Sinauer Associates, Sunderland, Massachusetts.

[5] Wolfe, M.S., and Caten C.E., eds. 1987. Populations of Plant Pathogens: Their Dynamics and Genetics. Blackwell Scientific Publications. Oxford, UK.

[6] Futuyuma, D.J. and Slatkin, M. eds. 1983. Coevolution. Sinauer Associates, Sunderland. MA.

[7] Hartl, D.L. 2000. A Primer of Population Genetics. Third Edition. Sinauer Associates, Sunderland, Massachusetts.

[8] Gleason, M. L., Daughtrey, M. L., Chase, A. R., Moorman, G. W., and Mueller, D. S. 2009. Diseases of herbaceous perennials. APS Press, St. Paul, MN. 281 pp (over 700 color photographs)

[9] Diseases of Floral Crops. 1985. D. L. Strider, editor. Praeger Publishers, 521 Fifth Ave., New York, NY 10175. Volumes I, 638 pages and II, 579 pages.

[10] Burdon J.J., Thrall P.H., Brown A.H.D. 1999. Resistance and virulence structure in two Linum marginale-Melampsora lini host-pathogen metapopulations with different mating systems. Evolution 53:704-716.

[11] Sibiya M., Sumbwanyambe M. A Computational Procedure for the Recognition and Classification of Maize Leaf Diseases Out of Healthy Leaves Using Convolutional Neural Networks. AgriEngineering. 2019;1:119–131. doi: 10.3390/agriengineering1010009. [CrossRef] [Google Scholar]

[12] Zhang K.,Wu Q.,Liu A., Meng X. Can Deep Learning Identify Tomato Leaf Disease?Adv. Multimed.2018;2018:10.doi: 10.1155/2018/6710865. [CrossRef] [Google Scholar]

[13] mara J., Bouaziz B., Algergawy A. A Deep Learning-based Approach for Banana Leaf Diseases Classification; Proceedings of the BTW (Workshops); Stuttgart, Germany. 6–10 March 2017; pp. 79–88. [Google Scholar]

[14] Durmuş H., Güneş E.O., Kırcı M. Disease detection on the leaves of the tomato plants by using deep learning; Proceedings of the 2017 6th International Conference on Agro-Geoinformatics; Fairfax, VA, USA. 7–10 August 2017; pp. 1–5. [Google Scholar]

[15] Ma J., Du K., Zheng F., Zhang L., Gong Z., Sun Z. A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. Comput. Electron. Agric. 2018;154:18–24. doi: 10.1016/j.compag.2018.08.048. [CrossRef] [Google Scholar]