

# Plant Leaf Disease Detection using Convolutional Neural Network and Random Forest Classifier

Ashutosh Kumar Singh<sup>1</sup>, Dr. Bharati Chaurasia<sup>2</sup>

<sup>1</sup>Ph.D. Scholar, <sup>2</sup>HoD (EE & EEE)

<sup>1,2</sup> RKDF Institute of Science & Technology, Bhopal, India.

**Received on:** 16 July, 2021

**Revised on:** 28 August, 2021

**Published on:** 30 August, 2021

**Abstract-** Plant diseases represent adverse factors that cause a serious reduction in the quality and quantity of agricultural crops. Commonly, expert biologists or farmers observe plants with the naked eye for disease, but this method is often inaccurate and can take a great deal of time. In this study, we use artificial intelligence and computer vision techniques to achieve the objective of designing and developing an intelligent mechanism for the classification of leaf diseases. In this paper, data augmentation is performed on the Plant Village dataset images (for apple, potato and rice plants) and their deep features are extracted using Convolutional Neural Network (CNN). These features are classified by Random Forest classifier and result is attained in terms of f-score and accuracy.

**Keywords-** CNN, Deep Features, DL, ML, Random Forest Classifier.

## I-INTRODUCTION

Diseases, pests and other undesirable substances present in crops can cause a sharp decline in agricultural production [1]. The impact of these dangerous factors on crops is directly reflected in the decline in the quality and quantity of crops. To combat, control and manage the effects of biological organisms and diseases, the term “pesticides” was coined [2]. Typically, the diagnosis of plant pests and diseases is usually analyzed by visual inspection based on the appearance, morphology and

other characteristics of the leaves. It is recommended that this visual examination be performed and analyzed only by a highly trained biologist, as misdiagnosis can lead to irreparable loss of yield. It should be noted that pest and disease control research is usually expensive and requires the presence of a specialized biologist to diagnose and prevent the spread and transmission of any disease as early as possible [3]. Recently, AI has found a large number of applications in all areas of daily life, leading to the emergence of the terms machine learning (ML) and deep learning (DL), which, in terms of simplicity, allow machines to “learn” a large number of patterns and then take action. The link between DL technology and computer vision has led to the emergence of intelligent algorithms that analyze and classify patterns or images with more accurate performance than the average person [4].

In this research paper, we compare the most common imaging algorithms for analyzing and classifying objects. Our work tries to simulate which algorithm predicts the best outcome when diagnosing the disease in plant leaves. It is expected that the results will be used to determine which algorithm is most effective in creating a smart system for detecting leaf diseases.

## II-PROPOSED METHODOLOGY

This research work presents the plant leaf disease detection of apple, potato and rice leaves using deep

features based feature extraction followed by the classification using Random Forest classifier. Figure 1 shows the generalized block diagram for the proposed work.

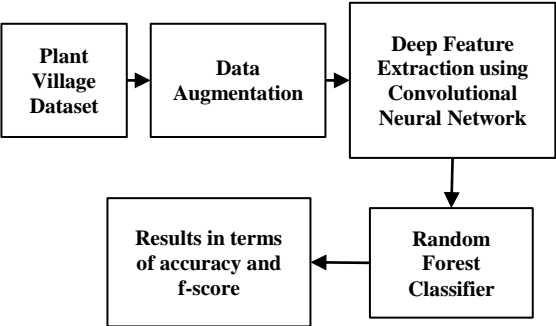


Fig 1- Block diagram for proposed approach of plant leaf disease detection

A. Image Acquisition

A lot of information is required to train intelligent visualization and classification systems. In general, machine learning and deep learning systems improve

their performance when training with large amounts of data. In this article, we have used the implementation of the Plant Village database [5] [6]. This "dataset" consists of 54,323 images of 14 plants and is divided into 38 groups of healthy leaves of plants with different types of diseases. This study used 11654 images of three types of plants, apple (7771 images), potato (3763 images) and rice (120 images).

To train and evaluate deep learning systems, the data volume should be divided into research sets and assessment sets. The dataset generated from this study is unique in that it contains images of different sizes and provides more sensor power.

Plant leaf images were obtained from the PlantVillage dataset [6] for preprocessing, feature extraction, feature selection, and classification. Tables 1, 2, and 3 show examples of various leaf diseases of apple, potato and rice plants, respectively, from the dataset.

Table 1-Apple plant leaf with disease (Source: PlantVillage Dataset [6])





















Healthy apple leaf					
Apple leaf with scab disease					
Apple leaf with black rot disease					
Apple leaf with cedar apple rust disease					

Table 2-Potato plant leaf with disease (Source: PlantVillage Dataset [6])










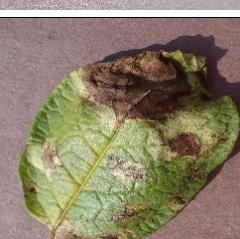

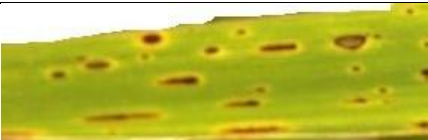

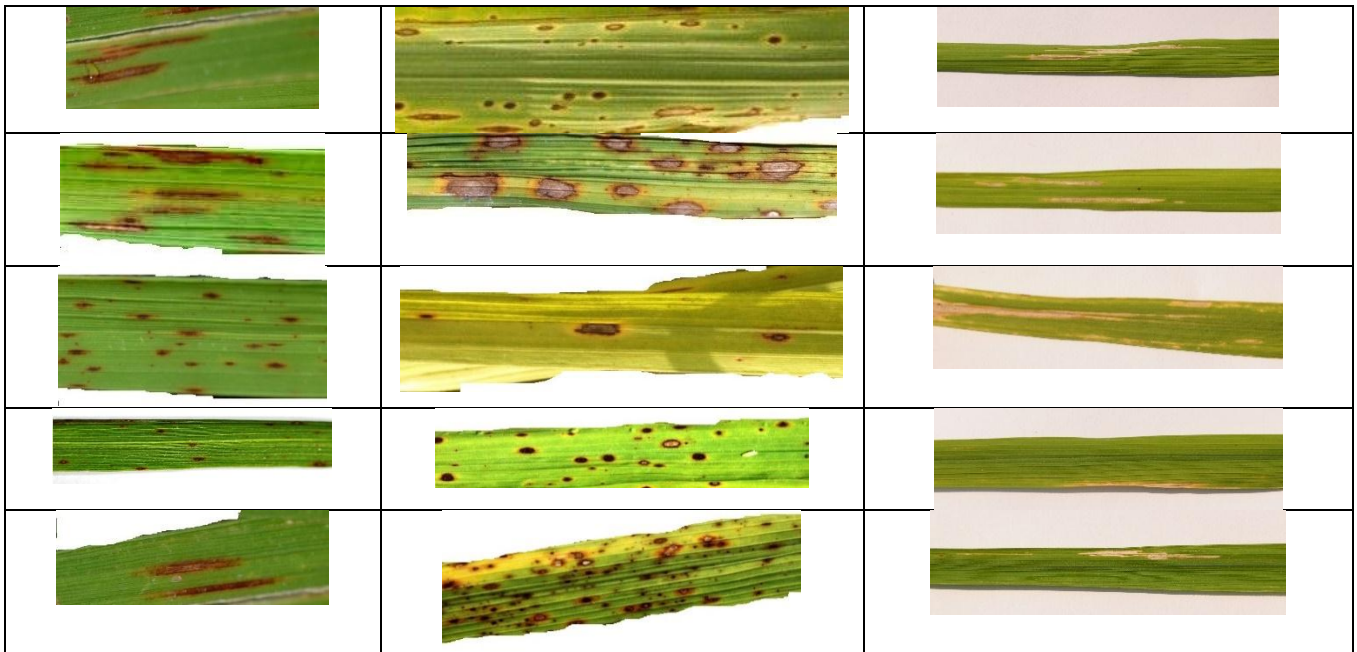
Healthy leaf		Leaf with Late blight disease	
			
			
			
			
			

Table 3-Rice plant leaf with disease (Source: PlantVillage Dataset [6])

Rice leaf with leaf smut disease	Rice leaf with brown-spot disease	Rice leaf with bacterial leaf blight disease
		





### B. Data Augmentation

This means that training the model must have enabled it to learn the main features of a data set. For this, it is necessary that:

- **Data Space:** The learning data space covers the spectrum of possibilities, i.e. contain the largest number of different examples corresponding to the context of use of the model.
- **Features Space:** The feature space of the training data also covers the spectrum of possibilities, i.e. contain the greatest possible number of representations of each features of the data.

To satisfy the first point, we must therefore collect the greatest possible variety of training images corresponding to the context of use and the objective of our model.

And to satisfy the second point, we must apply Data Augmentation techniques to the training images at our disposal, the most common of which are affine transformations (horizontal and / or vertical flip, rotation). There are also non-affine transformations such as, for example, variation in brightness and contrast, wrap (perspective), resizing, random crop (random part of an image), jitter (random noise) or cutout (squares random blacks).

### C. Deep Features Extraction using Convolutional Neural Network (CNN)

The convolutional neural networks or CNNs by its initials are a specialized type of neural networks recommended for the processing of data with a topology

in the form of mesh or grid. The type of data most used with this type of networks are the images (meshes of x and y pixels), although time series are also used (data in a dimension with an additional dimension to be The temporal dimension), data in three dimensions such as magneto resonance scanners or videos (two dimensions associated with the images plus one dimension associated with the temporal development of the video) [7].

CNNs have been applied to many tasks with great success. Recently the level of human sight has been surpassed in terms of image recognition thanks to the utilization of a deep convolutional neural network [8].

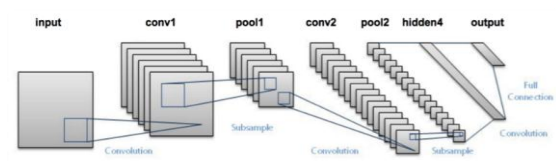


Fig 2- Typical architecture of a deep convolutional neural network [9]

#### 1. The Operation of Convolution

In its most general form, a convolution is an operation applied to two functions with real numbers as arguments. The operation of convolution is defined by the following mathematical expression:

$$s(t) = \int x(a)w(t - a)da \quad (1)$$

Commonly the convolution operation is symbolized by:

$$s(t) = (x * w)(t) \quad (2)$$

Using the terminology associated with convolutional neural networks, the first term (in this case  $(x)$ ) of the convolution operation is often referred to as input, while the second argument (in our case  $w$ ) it's called kernel. The output or result of the operation or convolution is usually called a feature map.

When working with a computer, discrete data will be available, so that what used to be an integral function of logical functions, will have to become a sum of "discrete" functions also continuous, of the following shape:

$$S(t) = (x * w)(t) = \sum_{a=-\infty}^{\infty} x(a)w(t-a) \quad (3)$$

In deep learning applications, the input is usually a vector of several dimensions (tensor), and the kernel is often a multidimensional vector of parameters that are modified by the learning algorithm. For example, if it is used as input data, an image  $I$ , the most frequent is that a two-dimensional kernel is used, which in this case we will denote as  $K$ :

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(i-m, j-n)K(m,n) \quad (4)$$

The scope of the present work, it was decided to use convolutional neural networks with fewer parameters when compared to the canonical solutions in the literature. Thus, the chosen architectures were:

**LeNet:** Initially proposed for the task of recognizing handwritten digits, this network consists of two convolutional layers followed by layers of max pooling in order to extract characteristics. Finally, a final convolutional layer is followed by two completely connected layers for classification of the outlet [10];

**AlexNet:** Aiming at the use of a convolutional neural network architecture with good performance reported in the related works, this network is composed of 5 initial convolutional layers and 3 layers completely connected at the end to produce the classification. It also has intermediate layers of dropout and max pooling [9];

**MobileNet:** For use on mobile and embedded devices, this convolutional neural network is based on deep separable convolution operations, which reduces the burden of operations to be carried out in the first layers [11];

**ShuffleNet:** It is based on two operations introduced by the authors, the so-called group convolutions, which are multiple convolutions in which each covers a portion of the input channels, and the shuffling of channels, which randomly mix the output channels of the convolutions in group. According to its proponents, this architecture has

a low computational cost while maintaining good accuracy [12];

**EffNet:** It resembles the MobileNet and ShuffleNet networks in terms of the use of in-depth separable convolution operations, but introduces a new convolutional block that reduces the computational burden while exceeding the state of the art performance for some widely known databases [13].

Considering the previously mentioned architectures, we have the number of trainable, non-trainable and total parameters as shown in Table 4. Note that the sum of the number of total parameters is lower than that of the VGG16 and VGG19 architectures [14], which suggests that their combination, according to a committee, may prove to be less costly than well-established architectures in the literature, under the terms considered. The considered architectures were trained according to the methodology previously described and evaluated individually in view of the performance in the test set.

Table 4-Parameters of the convolutional neural networks considered

Architecture	Trainable Parameters	Non-Trainable Parameters	Total Parameters
LeNet	19479481	0	24973547
AlexNet	30847124	0	33741772
MobileNet	3184795	20748	3119529
ShuffleNet	1897632	32054	1976586
EffNet	604552	2033	619832
<b>Total</b>	<b>56013584</b>	<b>54835</b>	<b>64431266</b>

#### D. Classification by Random Forest Classifier

A random forest is a classifier consisting of a set of base classifiers such as a decision tree shown:

$$\{h(x, \theta_k), k = 1, \dots, L\} \quad (5)$$

Random forests are composed of a set of binary decision trees in which randomness has been introduced.

Random forests were introduced by Breiman (2001) by the following very general definition [15]:

Let  $(\hat{h}(\theta_1), \dots, \hat{h}(\theta_q))$  a collection of tree predictors, with  $\theta_1, \dots, \theta_q$  random variables independent of  $\mathcal{L}_n$ . The predictor of random forests  $\hat{h}_{RF}$  is obtained is aggregating this collection of random trees as follows:

- $\hat{h}_{RF}(x) = \frac{1}{q} \sum_{l=1}^q \hat{h}(x, \theta_l)$  Average of individual tree predictions in regression.

- $\hat{h}_{RF}(x) = \arg \max_{1 \leq k \leq K} \sum_{l=1}^q 1_{\hat{h}(x, \theta_l)=k}$   
Majority vote among individual predictions trees in classification.

### III-SIMULATION RESULTS

After the execution of the proposed methodology, the performance of the architectures was assessed individually, according to the metrics of accuracy and F-Score. These results are summarized in Table 5, ordered in decreasing order by accuracy.

Table 5- Individual performance of convolutional neural network models. The acronyms TP, TN, FP and FN denote, respectively, true positive values, true negatives, false positives and false negatives

Model	TP	TN	FP	FN	Accuracy	F-Score
MobileNet	237	715	33	9	95.77%	91.86%
AlexNet	224	706	46	18	93.56%	87.50%
EffNet	148	700	122	24	85.31%	66.97%
LeNet	88	650	182	74	74.24%	40.74%
ShuffleNet	173	386	97	338	56.23%	44.30%

It is observed that three of the best networks obtained accuracy greater than 95%, with MobileNet, in particular, reaching the highest value among the observed networks. In the case of the LeNet and Shuffle Net networks, in particular, considering the proportions between classes and the aspects of accuracy, there is a low performance measured by the F-Score.

### IV-CONCLUSION

The Convolutional Neural Networks used in this work represents a Deep Learning architecture that has been achieving remarkable prominence in image recognition. Five convolutional neural network architectures were trained and tested for the problem in question, to mention: LeNet, ShuffleNet, AlexNet, EffNet and MobileNet, the latter having achieved better performance among them. All networks were combined in committees subject to three voting strategies, by majority, mediated by deep features based random forest classifier. As a result, this work achieved an accuracy of 95.77% in the detection of leaf diseases of apple, potato and rice plants.

### REFERENCES

- [1] Strange, R.N. and Scott, P.R., 2005. Plant disease: a threat to global food security. *Annual review of phytopathology*, 43.
- [2] Rangarajan, A.K., Purushothaman, R. and Ramesh, A., 2018. Tomato crop disease classification using pre-trained deep learning algorithm. *Procedia computer science*, 133, pp.1040-1047.
- [3] Nutter, F.W., Esker, P.D. and Netto, R.A.C., 2006. Disease assessment concepts and the advancements made in improving the accuracy and precision of plant disease data. *European Journal of Plant Pathology*, 115(1), pp.95-103.
- [4] Barbedo, J.G.A., 2019. Plant disease identification from individual lesions and spots using deep learning. *Biosystems Engineering*, 180, pp.96-107.
- [5] Hughes, D. and Salathé, M., 2015. An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint arXiv:1511.08060*.
- [6] PlantVillage Dataset for leaf disease detection, Available online at: <https://www.kaggle.com/emmarex/plantdisease>
- [7] Tm, P., Pranathi, A., SaiAshritha, K., Chittaragi, N.B. and Koolagudi, S.G., 2018, August. Tomato leaf disease detection using convolutional neural networks. In *2018 Eleventh International Conference on Contemporary Computing (IC3)* (pp. 1-5). IEEE.
- [8] Toda, Y. and Okura, F., 2019. How convolutional neural networks diagnose plant disease. *Plant Phenomics*, 2019, p.9237136.
- [9] Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2012. ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems* 25 (NIPS 2012).
- [10] LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P., 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), pp.2278-2324.
- [11] Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M. and Adam, H., 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.
- [12] Zhang, X., Zhou, X., Lin, M. and Sun, J., 2018. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6848-6856).
- [13] Freeman, I., Roese-Koerner, L. and Kummert, A., 2018, October. Effnet: An efficient structure for convolutional neural networks. In *2018 25th IEEE International Conference on Image Processing (ICIP)* (pp. 6-10). IEEE.
- [14] Simonyan, K. and Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- [15] Breiman, L., 2001. Random forests. *Machine learning*, 45(1), pp.5-32.