using Deep Learning Techniques

Chavda Rohit P.
Department of Computer Engineering
Atmiye University
Rajkot,Gujarat,India

Tosal Bhalodia
Department of Computer Engineering
Atmiya University
Rajkot,Gujarat,india

***Abstract*— Indian economy is majorly dependent on the agricultural products to enhance its economy. Majority of populations in India are indulge into the agriculture field directly or indirectly. Since last few years, research community has spent lot of time for the automation of agriculture field to increase the outcome of agriculture produces through Machine learning and Deep learning Approach. In precision farming, early disease detection is playing key role to increase the agricultural output that can also enhance the quality of food production and minimize. Deep Learning methods have proved successful in the field of image recognition in terms of both classification and detection. In this review paper, we will review the literature of existing work done in automated diseases detection in the variety of plants. Analyzing the detection accuracy will be the outcome of optimization in processing the raw images, extracting low level features, datasets, tuning of hyper parameters, etc. Our work, mainly focused on analyzing the detection accuracy through various state-of-the-art algorithms in Deep Learning like DenseNet-121, ResNet-50, VGG-16, Inception V4, etc.**

Keywords— Plant diseases detection and classification, Transfer Learning, Deep Learning, DenseNet-121, ResNet-50, VGG-16, Inception V4.

Introduction

Agriculture plays a major role in the economy of any country. In India, economic growth of 18% of the populations are depended on the agriculture produces. For the past 3 years, the gross value added (GVA) by agriculture to the country's total economy has increased from 17.6% to 20.2%. Hence, the impact of plant disease and infections from pests on agriculture may affect the world's economy by reducing the production quality of food. So, early diagnosis of the crop diseases may become very essential to improve the production. Identifying the types of diseases in the plant at the early stage is crucial research issue which will pave the way of better decision making for the protection of plant at early stage. Generally, a unique mark can be spotted into an infected plants on the stems, fruits, leaves, etc. Manually identifying such abnormality in such infected plants required expertise and huge manpower cost, which in turn increase the cost of production. Furthermore, it might be a time consuming as well as possibilities of misclassification due to limited knowledge of infections on the plants. Few decades ago, a lot of research on image processing has been done to identify diseases information on the plants. However, due to emergence of computer vision technologies especially, deep learning, the task of finding abnormality in the plant diseases has given an impetus.

In this review paper, we will try to analyze the work of diseases detection on the plants through Convolution Neural Network (CNN) based deep learning techniques like DenseNet-121, ResNet-50, VGG-16, and Inception V4 which are proved the state-of-the-art for the image classification and object detection. Our main focus is on review of existing work done by researchers to detect the plant diseases on the various types of plants through above mentioned algorithms. CNN is widely explored by the research community due to its efficiency of the extracting the low level features from the images.

In this review paper, we have focused on following research questions on disease detection in the various plants leaves.

- Preprocessing of raw images
- Low level feature identification
- Methods of improving the classification and recognition accuracy
- Analysis of Public datasets of plants
- Analysis of CNN based methods used for classification of plant disease.

Rest of the paper is organized as follows;

In section I, we have introduced the concept of deep learning for the image classification and detection. Section II describes the state-of-the-art algorithms in the field of convolution neural network. In section III, review of existing work has been done. Section IV analyze the existing work and potential research issues in the field of diseases detection from the plant images. Section V concludes the paper.

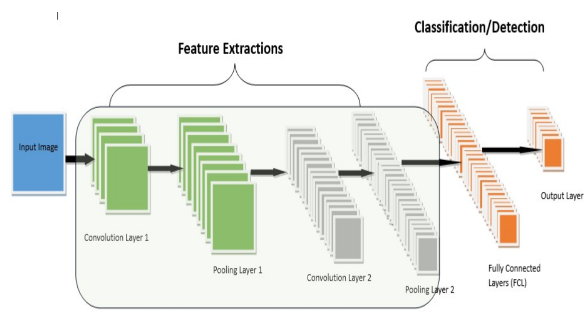

Fig. 1. Deep learning Architecutre

I. INTRODUCTION TO DEEP LEARNING FOR IMAGE CLASSICIATION AND DETECTION

Before invention of Deep learning, object detection methods are built based on handcrafted features and shallow trainable architectures of neural networks. For complex image recognition tasks, their performance got stagnant by constructing complex ensembles which combine multiple low-level image features with high-level context from object detectors and scene classifiers. With the rapid increase in the development of deep learning architectures, low level feature extractions are introduced to address the problems existing in traditional architectures. These models behave differently in network architecture as shown in fig. 1.

A. Feature Extractions

Feature extraction is the process of converting unprocessed data into numerical features so that the information contained in the original data set may be processed. Compared to using machine learning on raw data directly, it produces better outcomes.

To extract the features from the input training data, convolution layers are employed. A collection of filters is included in each convolution layer to aid in the extraction of features. Generally speaking, the complexity of the features that convolution layers learn grows with CNN model depth.

By considering the convolution of a piece of the sample data, features are retrieved. The stride length and padding value determine how much of the data section the filter traverses each time. Zero padding may or may not be applied to data samples prior to convolution.

After that, the convolution output is run via a Rectified Linear Unit (ReLU) activation unit. The data is transformed into its non-linear form by this unit. Only in the event that the convolution output is negative is the ReLU output trimmed to zero.

After that, a pooling layer processes the ReLU output. Convolution captures duplicate features, which are removed by the pooling layer. Thus, the size of the data sample is decreased by this layer. Pooling works on the assumption that adjacent image pixel values are almost the same. Pooling is done using the average, minimum, and maximum of four neighboring pixel values. A 2*2 filter can typically reduce an input image's size in half. Zero padding may or may not be applied to the input data prior to pooling.

The CNN model repeats this process of sequentially transferring data via the convolution and pooling layers. This method is performed two to four times for learning purposes. A multi-layer neural network is then used to process the output from the subsequent convolution and pooling layers. In this case, a neuron unit serves as a feature map that contains data about a unit.

B. Fully Connected Neural Network

In a fully connected layer of a neural network, each neuron uses a weights matrix to apply a linear transformation to the input vector. Because of this, every potential layer-to-layer connection is present, meaning that each input affects every output in turn in the output vector.

A collection of dependent non-linear functions make up neural networks. A neuron is responsible for each distinct function. The neuron in fully connected layers uses a weights matrix to apply a linear transformation to the input vector. Next, a non-linear activation function f applies a non-linear transformation on the product.

The dot product between the layer's input and weights matrix is wrapped by the activation function "f." Keep in mind that as the model is trained, the columns in the weights matrix will all have various quantities and will be tuned.

II. DEEP LEARNING ARCHITURES FOR IMAGE CLASSIFICATION AND DETECTION

A. DensNet-121

Every convolutional layer of a conventional feed-forward convolutional neural network (CNN), with the exception of the first, which receives input, gets the output of the preceding convolutional layer and generates an output feature map, which is then forwarded to the following convolutional layer. As a result, there are 'L' direct connections for 'L' layers, one for each layer and the layer after that.

Nevertheless, the "vanishing gradient" issue appears as the CNN's layer count rises, or as the layers go deeper. As a result, the network's capacity to train efficiently is diminished since more information may "vanish" or be lost along the journey from the input to the output layers.

By altering the conventional CNN design and streamlining the layer-to-layer connection structure, DenseNets alleviate this issue. Densely linked Convolutional Network is the term given to an architecture in which every layer is directly linked to every other layer: DenseNet. There are L(L+1)/2 direct connections for 'L' layers.

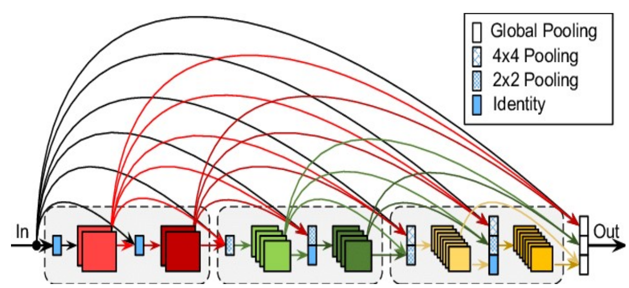

Fig. 2. DensNet121 Architecture

DenseNet-121 comprises the following layers:

Layer 1: 1 7x7 Convolution

Layer 2: 58 3x3 Convolution

Layuer 3: 61 1x1 Convolution

Layer 4: 4 AvgPool

Layer 5: 1 Fully Connected Layer

Layer 6: DenseNet's components consist of:

Connectivity:

The feature maps from every previous layer are concatenated and utilized as inputs in each subsequent layer rather than being summed. Because duplicate feature maps are removed, DenseNets require fewer parameters than an equivalent standard CNN, enabling feature reuse. Thus, the feature-maps of all previous layers, x0,...,xl-1, are sent into the lth layer as input:

$X_l = H_l([x_0, x_1, \ldots, x_{l-1}])$

where [x0,x1,...,xl-1] represents the feature-map concatenation, or the output generated in each of the layers that came before l (0,...,l-1). To make implementation easier, Hl's many inputs are concatenated into a single tensor.

DenseBlocks

When feature map sizes vary, it is not practical to utilize the concatenation method. Nonetheless, downsampling layers—which does this by reducing the dimensionality of feature maps—is a crucial component of CNNs since it allows for faster computing.

DenseNets are split up into DenseBlocks in order to facilitate this; inside a block, the size of the feature maps stays the same, but the number of filters that separate them varies. The number of channels is cut in half by the so-called Transition Layers, which are the layers that lie in between the blocks.

Growth Rate

Consider the characteristics as the network's overall condition. Upon passing through each thick layer, the feature map becomes larger as each layer adds 'K' features to the global state (pre-existing features). This parameter 'K' is known as the network's growth rate; it controls how much data is added to each layer of the network. The lth layer has k feature maps if each function H l generates them.

$k_l = k_0 + k * (l-1)$

Bottleneck Layers

Even while each layer only generates k feature-maps as output, there may be a large number of inputs—especially for layers that come after. To increase computing speed and efficiency, a 1x1 convolution layer might be added as a bottleneck layer before each 3x3 convolution.

B. ResNet 50

State-of-the-art results can be achieved by training a robust image classification model, such as ResNet50, on huge datasets. A significant advancement in it is the utilization of residual connections, which enable the network to pick up a set of residual functions that translate the input into the intended output. Without having to deal with the issue of vanishing gradients, the network is now able to learn even deeper structures thanks to these residual connections.

Multiple convolutional layers are followed by batch normalization and ReLU activation in the convolutional layers of ResNet50. These layers are in charge of taking characteristics like edges, textures, and forms out of the input picture. Max pooling layers, which decrease the spatial dimensions of the feature maps while maintaining the most significant features, come after the convolutional layers. The two main ResNet50 building blocks are the identification block and the convolutional block. The identity block is a straightforward block that adds the input back to the output after passing it through several convolutional layers. As a result, the network is able to learn residual functions, which convert input into desired output. With the inclusion of a 1x1 convolutional layer to lower the number of filters before the 3x3 convolutional layer, the convolutional block resembles the identity block. The building component of the 50-layer ResNet is designed as a bottleneck. A bottleneck residual block minimizes the number of parameters and matrix multiplications by using 1×1 convolutions, also referred to as a "bottleneck." This makes training each layer considerably faster. Instead of using two layers, it employs a stack of three layers.

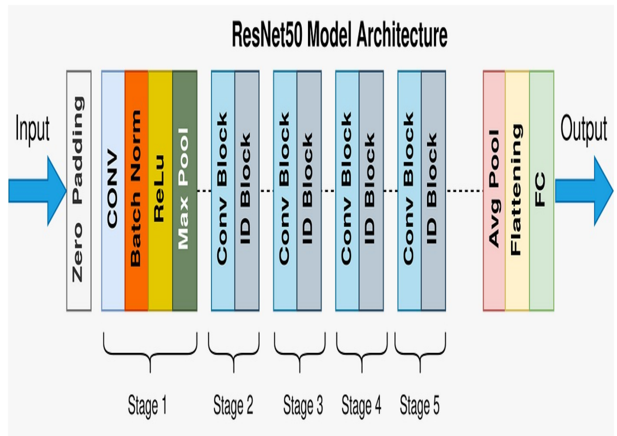

Fig. 3. ResNet50 Architecture

The components of the 50-layer ResNet architecture are as follows:

- 7×7 kernel convolution alongside 64 other kernels with a 2-sized stride.
- max pooling layer with a 2-sized stride.
- 9 more layers—3×3,64 kernel convolution, another with 1×1,64 kernels, and a third with 1×1,256 kernels. These 3 layers are repeated 3 times.
- 12 more layers with 1×1,128 kernels, 3×3,128 kernels, and 1×1,512 kernels, iterated 4 times.
- 18 more layers with 1×1,256 cores, and 2 cores 3×3,256 and 1×1,1024, iterated 6 times.
- 9 more layers with 1×1,512 cores, 3×3,512 cores, and 1×1,2048 cores iterated 3 times.

Average pooling, followed by a fully connected layer with 1000 nodes, using the softmax activation function.

C. VGG 16

ConvNets, or convolutional neural networks, are a type of artificial neural network. There are many hidden layers, an output layer, and an input layer in a convolutional neural network. One of the greatest computer vision models available today is the CNN (Convolutional Neural Network) variant known as VGG16. Using an architecture with relatively tiny (3 × 3) convolution filters, the model's developers analyzed the networks and improved the depth, demonstrating a considerable improvement over the prior-art setups. They increased the depth to around 16–19 weight layers, or 138 trainable parameters.

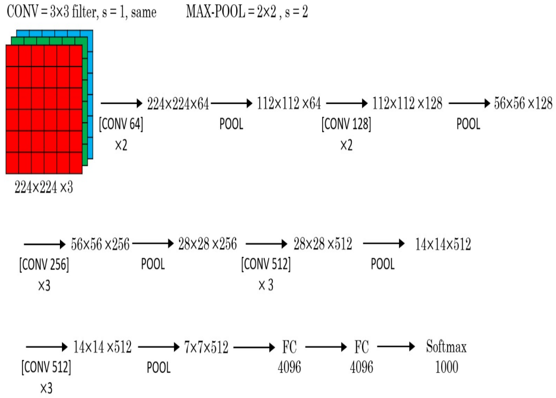

Fig. 4. VGG16 Architecture

With 92.7% accuracy, the object identification and classification algorithm VGG16 can categorize 1000 photos into 1000 distinct categories. It is a well-liked image classification system that is simple to apply with transfer learning.

The sixteen in VGG16 stands for sixteen weighted layers. Although VGG16 contains twenty-one layers total—sixteen convolutional layers, five Max Pooling layers, and three Dense layers—it only includes sixteen weight layers, or learnable parameters layers.

D. AlexNet

Alex Krizhevsky, a student of Hinton, who won the 2012 ImageNet competition, built AlexNet. Additionally, deeper neural networks—like the superb vgg, GoogleLeNet—were presented after that year. The accuracy rate of its official data model is 57.1%, with the top 1-5 reaching 80.2%. Even for conventional machine learning classification techniques, this is already quite impressive.

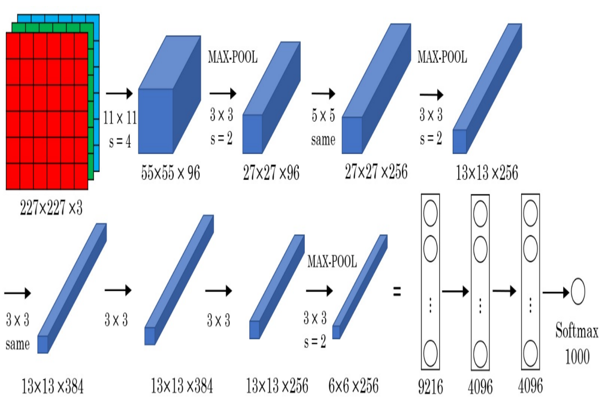

Fig. 5. AlexNet Architecture

AlexNet architecture used GPU to boost the training performance. Five convolution layers, three max-pooling layers, two normalized layers, two fully linked layers, and one softmax layer make up AlexNet. A convolution filter and a non-linear activation function known as "ReLU" make up each convolution layer. Because there are completely linked layers in the pooling layers, the max-pooling function is carried out, and the input size is fixed. Although the input size is stated as 224x224x3 in most places, it really comes out to be 227x227x3 because of some padding. Above all, there are more than 60 million parameters in AlexNet.

Important Features:

"ReLU" is utilized as an activation function instead of "tanh."

AlexNet shows that deep CNNs can be trained substantially faster with saturating activation functions like Tanh or Sigmoid. With the use of ReLUs, AlexNet may reach a training error rate of 25%, as seen in the graphic below (solid curve). This is six times quicker than a network that uses tanh (dotted curve). This was assessed using the CIFAR-10 dataset.

Another frequently mentioned idea is dropout, which can successfully stop neural networks from over fitting. To keep the model from over fitting, a regular approach is employed as opposed to the generic linear model. Dropout is implemented in the neural network by changing the architecture of the neural network. Update the parameters in accordance with the neural network's learning strategy after randomly deleting a particular number of neurons with a certain probability while maintaining the same number of neurons in the input and output layers. The following time around, re-random Delete a few neurons till the training is finished.

Data augmentation techniques include cropping, flipping, jittering, and color normalization. Data augmentation is also use translation and noise to artificially enhance the training set's size and produce a batch of "new" data from the current data

III. Literature Review

This section provides a detailed literature review of detection and classification using different machine learning and deep learning techniques. All papers have been published prior to 2022. For the purpose of locating pertinent research on ML and DL techniques for plant disease classification and detection. We searched utilizing a number of search engines, including Web of Science, ScienceDirect, Scholar, and Scopus. Terms like "deep learning," "machine learning," "classification," "disease detection," "healthy plant," and "diseased plant" were employed. All co-authors evaluated each paper's abstract to determine whether or not it belonged there and could be included in the final version.

Review of A CNN-based deep learning models have been reviewed. Authors have used the pre-trained model using transfer learning method to train plant diseases using 4 CNN based models namely V4, VGG-16, ResNet-50, and DenseNet-121. They proposed that using transfer learning method, DenseNet-121 which has 121 deep layers outperformed all other networks with 99.87% training accuracy and 99.81% testing accuracy. Data set of Plant Village containing 54,305 images of healthy and infected plants have been used to compare the accuracy of all the methods. Where 80% of datasets have been used for training and 20% of datasets used for the validation. For preprocessing, they have adopted on-time image augmentation technique which creates temporary augmented images from the given raw image during training time to address the issue of overfitting and space. Robustness of the training set has been achieved through image augmentation techniques [1].

Authors have used Cascading Autoencoder with Attention Residual U-Net (CAAR-UNet), leverages deep learning to achieve precise segmentation and classification of plant leaf diseases. The model architecture provides a promising method for image segmentation problems, with enhanced efficiency, less overfitting, and greater generalization capabilities. The resulting model takes an image of a plant leaf as input and produces a segmented image with diseased areas highlighted and classified [2].

The research explores the application of pre-trained deep convolutional neural networks (CNNs) in this classification task, utilizing an open dataset comprising 52 categories of various diseases and healthy plant leaves. This study evaluated the performance of pre-trained deep CNN models, including Xception, InceptionResNetV2, InceptionV3, and ResNet50, paired with EfficientNetB3-adaptive augmented deep learning (AADL) for precise disease identification. Performance assessment was conducted using parameters such as batch size, dropout, and epoch counts, determining their accuracy, precision, recall, and F1 score. The EfficientNetB3-AADL model outperformed the other models and conventional feature based methods, achieving a remarkable accuracy of 98.71% [3].

Researcher also presented review on application of Machine learning and deep learning in the field of diseases detection w.r.t. various data sets. They proposed the extensive study of 5 state of the art algorithm for plant diseases detection. Computational results shown that object detection accuracy is high with YOLOv5 [4].

A Deep Transfer Learning Approach for Pathogen-Based Classification of Plant Diseases. An automated plant disease detection and its classification are done using EfficientNetV2B2 and EfficientNetV2B3 model along with identifying the pathogen responsible for it using transfer learning technique. Author have used new data set called Agri-ImageNet as well as images of leaves, bulb, and flowers of sunflower and cauliflower captured in a natural realistic environment to overcome the issues in Plant Village Dataset like homogeneous backgrounds and controlled settings [5].

MaizeNet Deep Learning Approach have been proposed for Effective Recognition of Maize Plant Leaf Diseases. MaizeNet was used for the correct localization and classification of various maize crop leaf disorders. They have further improved the performance using Faster-RCNN approach that utilizes the ResNet-50 model with spatial-channel attention as its base network for the computation of deep keypoints which are then localized and categorized into various classes. The work was tested on a standard database named Corn Disease and Severity that contains samples from three different classes of maize plant diseases which are captured under diverse conditions such as complex background, brightness, and color and size variations. The MaizeNet model attains an average accuracy score of 97.89% along with mAP value of 0.94 [6].

authors have used the Tomato and cotton plant crops to detect the 12 commonly available diseases in those plants. A deep learning-based lightweight 2D CNN architecture have been proposed to classify 12 diseases and 2 class labels has healthy. The proposed architecture outperformed all the pre-trained models like VGG16, VGG19 and InceptionV3 despite having fewer parameters. The app works very impressively and classified the correct disease in a shorter period of time of about 4.84 ms due to the lightweight nature of the model [7].

The author proposed transformer embedded ResNet model. Using this model author try to remove the noise from the cassava leaves so the model will extract the features from the cassava leaves. And also proposed a FAMP-Softmax method to learn strict classification boundaries from the cassava leaf dataset. Author will compare the result of Xception, VGG16 Inception-v3, ResNet-50 and DenseNet121 and achieve performance improvements accuracy of 3.05%, 2.62%, 3.13%,2.12% and 2.62% [8].

Author used Plant Village dataset which includes 50,000 images of 14 different crops. For the classification from the different corps images used different Pre-Trained models. And for the detection CNN approach is applied which having 3 convolution and 3 max pooling layers having 2 fully connected layers for detect 9 diseases of tomato leaves. Classification accuracy is from the 76% to 100% on different classes and average accuracy of proposed model is 91.2% which having 10 classes [9].

The author used deep learning techniques for the detection. Authors have used ResNet50 architecture for the dieses identification of potato, tomato, corn and performance of this architecture is 98.7% for the plant disease [10].

The researcher explores transfer learning technique for recognition and classification of sunflower dieses. Author used Pre-Trained VGG16 and MobileNet architectures for classification and stacking the ensemble result of both and developed hybrid model from the both models [11].

Author used fully connected CNN method for the classification of rice crops. Model will apply Otsu's global thresholding technique for the image binarization to remove the background noise from the image [12].

Author apply Residual Network (ResNet) and transfer learning technique Inception ResNet architecture on Palm leaf dataset which having 2631 colored images with varied sizes used to train and test models which achieve 99.62% accuracy and also author apply augmentation technique on palm leaves dataset [13].

IV. ANALYSIS OF EXISTING WORK

Extensive research work on the existing system has been done in the previous sections. Where, majority of researchers have used fusion of state-of-the art architectures to improve the accuracy of dieses detection system. Below table 1 represents analysis of the some research work done for the dieses detection system.

As shown in table 1 we have taken the papers into account who focused on variety of plant diseases among several plants. We have also mentioned the dataset used by respective papers as well as the type of disease address the most common dataset used is plantvillage which consist of more than 65000 samples of different plants, we have also taken in account the architectures which used by the researchers as well as we mention the performance of all the architectures.

	Focus	Addressed				Performance
[1]	14	38	Plant Village	38	DenseNet121 ResNet50 VGG16 InceptionV4	99.81 98.73 82.75 97.59
[2]	Several	Several	Kaggle	58	Inception V3 InceptionReNetV2 ResNet50 Xception EfficientNetB3AADL	97.08 97.74 95.94 98.50 98.71
[3]	39	40	Plant Village Coffee Leaf Custom dataset	40	Cascading Autoencoder With attention Residual U-Net (CAAR-UNet)	95.26
[4]	Several	Several	PlantDoc	Various	Yolov5	High Accuracy
[5]	Sun Flower Cauli Flower	Several	Sunflower Cauliflower AgriImageNet	Varous	EfficientNetV2B2 EfficientNetV2B3	96.82 97.35
[6]	Maize	3	Corn Disease and Severity	3	MaizeNet	97.89
[7]	Tomato	14	Plant Village	14	2D CNN Architecture	97.36
[8]	Cassava	5	Kaggle	5	T-RNet(Transformer embedded Res-Net)	90.63
[9]	Tomato	9	Plant Village	9	Mobilenet VGG-16 InceptionV3 Proposed Model	63.75 77.2 63.4 91.2
[10]	Potato Tomato Corn	16	Plant Village	16	ResNet50	98.7
[11]	Sunflower	4	Google Image	4	VGG-16 MobileNet	81 86
[12]	Rice	3	Kaggle	3	CNN	99.7
[13]	Palm	3	Kaggle	3	ResNet	99.62

Table 1: Analysis of Existing work

V. CONCLUSION

Dieses detection using deep learning architectures are proved to be the promising way to optimize the detection accuracy. Various authors have proposed the independent or fusion techniques for classification and detection of plant dieses detections. As per the analysis of existing system, we could achieve the higher performance by using architecture during training time. However, lot many issues are tends to require to optimizing the deep neural networks during validation and testing phase.

REFERENCES

[1] Eunice, J., Popescu, D. E., Chowdary, M. K., & Hemanth, J. (2022). Deep learning-based leaf disease detection in crops using images for agricultural applications. *Agronomy*, *12*(10), 2395.

[2] Adnan, F., Awan, M. J., Mahmoud, A., Nobanee, H., Yasin, A., & Zain, A. M. (2023). EfficientNetB3-Adaptive Augmented Deep Learning (AADL) for Multi-Class Plant Disease Classification. IEEE Access.

[3] Abinaya, S., Kumar, K. U., & Alphonse, A. S. (2023). Cascading Autoencoder with Attention Residual U-Net for Multi-Class Plant Leaf Disease Segmentation and Classification. IEEE Access.

[4] Balafas, V., Karantoumanis, E., Louta, M., & Ploskas, N. (2023). Machine learning and deep learning for plant disease classification and detection. IEEE Access.

[5] Rani, K. A., & Gowrishankar, S. (2023). Pathogen-based Classification of Plant Diseases: A Deep Transfer Learning Approach for Intelligent Support Systems. IEEE Access.

[6] Masood, M., Nawaz, M., Nazir, T., Javed, A., Alkanhel, R., Elmannai, H., ... & Bourouis, S. (2023). MaizeNet: A deep learning approach for effective recognition of maize plant leaf diseases. IEEE Access.

[7] Peyal, H. I., Nahiduzzaman, M., Pramanik, M. A. H., Syfullah, M. K., Shahriar, S. M., Sultana, A., ... & Chowdhury, M. E. (2023). Plant Disease Classifier: Detection of Dual-Crop Diseases using Lightweight 2D CNN Architecture. IEEE Access.

[8] Zhong, Y., Huang, B., & Tang, C. (2022). Classification of cassava leaf disease based on a non-balanced dataset using transformer-embedded ResNet. Agriculture, 12(9), 1360.

[9] Agarwal, M., Singh, A., Arjaria, S., Sinha, A., & Gupta, S. (2020). ToLeD: Tomato leaf disease detection using convolution neural network. Procedia Computer Science, 167, 293-301.

[10] Arshad, M. S., Rehman, U. A., & Fraz, M. M. (2021, May). Plant disease identification using transfer learning. In 2021 International Conference on Digital Futures and Transformative Technologies (ICoDT2) (pp. 1-5). IEEE.

[11] Malik, A., Vaidya, G., Jagota, V., Eswaran, S., Sirohi, A., Batra, I., ... & Asenso, E. (2022). Design and evaluation of a hybrid technique for

detecting sunflower leaf disease using deep learning approach. Journal of Food Quality, 2022, 1-12.

[12] Upadhyay, S. K., & Kumar, A. (2022). A novel approach for rice plant diseases classification with deep convolutional neural network. International Journal of Information Technology, 1-15.

[13] Ahmed, M., & Ahmed, A. (2023). Palm tree disease detection and classification using residual network and transfer learning of inception ResNet. Plos one, 18(3), e0282250.