JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Deep Learning-Based PDConv Model for Plant Disease Detection, Classification and Prediction

Muhammad Harris Hashmi[1]
Department of Computer Science,
University of Alabama at Birmingham, 1402 10th Avenue S., Birmingham, AL 35294, USA

Bhanu Priya Mahant[2]
Director of Research
Voxbourne Services Pvt Ltd 300 Prospect Ave 4H Hackensack NJ 07601

Muhammad Salik Salam
Department of Computer Science,
University of Alabama at Birmingham, 1402 10th Avenue S., Birmingham, AL 35294, USA

Muhammad Ahsan
Department of Computer Science
University of Alabama at Birmingham,1402 10th Avenue S.
Birmingham, AL 35294, USA

Abstract*— Plant diseases (PD) have a detrimental effect on agricultural output, quality, economics, and quantity. To detect such diseases, the life cycle of plants should be properly monitored from the beginning. It is necessary to automate the disease-detecting equipment to expedite this process. Developing a disease detection system should take advantage of deep learning (DL) techniques. No environmental issues had hindered agriculture until recently when several plant leaf diseases began to reduce harvests. It is difficult to detect PD early, which is only one of the many problems that producers face. To solve this problem, scientists created the 'PDConv model, an automated PD categorization system built on a deep convolution neural network (Deep CNN). In this four-step technique, we preprocessed the data, added it to it, extracted features, and classified the data. In this study, we classify data and extract features using a Deep CNN, the layered architecture of which is briefly discussed. The complete system was implemented with the help of the suggested method utilizing python programming language and the Jupyter notebook environment. Using data from the FGVC7 dataset, researchers can create deep learning (DL) PD classification system that is both accurate and efficient. When compared to CNN, the PDConv model performed exceptionally well. A CNN model may be trained with only the segmented and annotated parts of images rather than with whole images. After being trained on the preexisting CNN model, the PDConv model's results on independent data sources increased from 61.91% to 95.46%. In addition, 82% of the test data set showed an increase in self-assessment confidence, indicating strong development in this area. Proof that CNNs may be used effectively for identifying PD.

***Keywords*—Plant Disease Detection, Image Segmentation, Classification, Deep Learning, Deep CNN, FGVC7 Dataset.**

I. INTRODUCTION

Diseases that affect plants can restrict the development of entire species, making the timely diagnosis of such threats crucial. A big step forward in agriculture is developing software that automatically identifies PD from plant leaves. In addition, the productivity and quality of crops benefit from early and timely detection of PD.[1]. The wide variety of crops grown means that even a trained agronomist or plant pathologist may miss a disease's telltale symptoms on a single leaf. Diseases are often diagnosed in urban centers, but visual inspection is still the norm in rural parts of underdeveloped nations. [2]. It also demands constant supervision by professionals. In isolated regions, farmers may have to travel a considerable distance to visit a specialist, which is time-consuming and costly. [3][4]. Farmers and agronomists benefit from the high throughput and pinpoint accuracy of automated computing systems for diagnosing PD.

Researchers have proposed several approaches to address the issues above. PD categorization using machine learning (ML) may make use of many different kinds of feature sets. The most often used feature sets are traditionally created or based on DL. DL models appear to provide the substantial potential for better performance after the development has been made in a subset of ML. The categorization and identification of PD symptoms have used various cutting-edge DL structures and visualization techniques. There is a fatal flaw in many DL models used for automated PD diagnosis, which has been significantly reduced after examining independent data. Utilizing the FGVC7 dataset and a model of CNN dubbed 'PDConv,' this work proposed a plausible explanation for this problem. Python and TensorFlowTM are used to create the model. Classification & prediction accuracy has been enhanced in the proposed model, while overfitting has been mitigated.

The remaining five parts cover different aspects of the topic. In Section II, we will go through some of the most cutting-edge methods for automated PD detection that have been published so far. Section III provides a detailed description of the proposed study, including a full research approach and a block diagram. Section IV presents the model's findings for identifying the PD Bacterial Spot in its leaves. This paper comes to a close in Section V. The conclusion and suggestions are presented in Section VI.

II. LITERATURE REVIEW

In this part, we present a selection of articles about the detection of PD via various sophisticated approaches.

Vijaykanth Reddy and Sashi Rekha (2021) In this research, CNN was utilized to replace this void by applying transfer learning (TL) to pre-trained deep models. It was proposed that cascading CNN with AlexNet and GoogleNet would produce a deep leaf disease prediction framework (DLDPF). The technique is known as Deep CNN based on CIDCNN-TL (Cascade Inception with TL). TensorFlow and Keras, along with the Python data science platform, are applied to create the proposed framework. DLDPF is connected to many DL models, including ResNet-20, GoogLeNet, VGGNet-16, and AlexNet. The empirical investigation uses Apple leaf datasets. Experimental results demonstrate that DLDPF is better than existing DL models for automated lead disease prediction. [5].

KP and Anitha (2021) The goal of the research is to use a DL model to put the many PD into groups. The CNN model is used because it has been so successful at classifying images. The model of DL can make more accurate and faster predictions than looking at the leaves of plants by hand. During this step, a dataset is used to train CNN and already trained models, such as DenseNet, ResNet, and VGG. The ResNet model is the most accurate of them all. [6].

Hirani *et al.* (2021) In recent years, CNNs have been used in various tasks, such as classifying images, extracting features, and dividing images into parts. One of these uses is to find PD, which is important because plant diseases are one of the most important reasons crops don't grow well. Different DL techniques have been used over time to address the issue of identifying and classifying plant diseases. But some things can't be done with these methods. Recent work with computer vision tasks that use transformer networks has shown much promise. This study looks at how these methods compare to conventional CNN methods when it comes to detecting PD. their transformer model's best accuracy for validation is 97.98%.[7].

In this study, **Mohameth, Bingcai and Sada (2020)** Assess the usefulness of transfer learning (TL) and deep feature extraction (FE) when applied to CNN architectures. Many different architectural models have been taught in the field of DL, with some attaining performance levels of over 99.53 percent. They will use SVM and KNN to categorize all the data we collect. The freely available Plant Village Dataset makes their research possible. The result demonstrates that SVM is the superior classifier for identifying leaf diseases. [8].

Kumar *et al.* (2020) use Deep CNNs to distinguish between healthy and diseased coffee plants (DCNNs). Among the most prevalent biotic stressors experienced by coffee plants are Cercospora Spots, Leaf Miner, Coffee Leaf Rust, and Phoma, each associated with specific types of leaf damage. Our most notable contribution is the suggestion of a distinct neural network (NN) for diagnosing coffee plant problems. The proposed NN has a record-breaking 97.61 % accuracy in identifying and classifying diseases affecting one coffee plant. Second, this method may help Indian farmers find coffee plant problems early on, boosting coffee output in the country.[9].

Militante, Gerardo and DIonisio, (2019) This research presents a practical method for simultaneously diagnosing a wide range of PD. Apples, corn, sugarcane, grapes, potatoes, and tomatoes are just some of the crops that the method was developed to identify. Many PD are also detectable by the method. The dataset used to train the DL models to identify and distinguish plant illnesses, and the absence of these diseases consists of 35,000 photos of healthy plant leaves and diseases. The trained model has a 96.5% accuracy rate, and the system has registered up to 100% accuracy when identifying the plant species and the disease kind.[10]

Ferentinos (2018) Employing basic leaf photos of healthy and sick plants, CNN models have been constructed for PD detection and diagnosis utilizing DL techniques. For model training, we used a publicly available library of 87,848 pictures representing 25 plant species over 58 classes of [plant, illness] pairs, encompassing healthy and diseased plants. Various model architectures were trained, with the highest performance obtaining a success rate of 99.53% in determining the appropriate [plant, illness] pair (or healthy plant). The model's impressive success makes it valuable advising or early warning tool. The technique might be further developed to provide an integrated PD diagnosis system that works in real production settings.[11].

In this study, they apply DL techniques to the problem of disease detection in plants. A comprehensive examination of the current literature failed to reveal any evidence that writers explored a groundbreaking method for detecting PD through examining photographs of affected leaves.

III. PROPOSED WORK

A Deep CNN architecture, referred to as the "PDConv" model, is provided as the answer, and its implementation is described in this section. Picture segmentation, data augmentation, classification, image preprocessing, and FE are also briefly outlined, as are the other stages of the suggested model. Finally, a summary of the suggested model and its accompanying block diagram is provided.

A. Problem Statement

PD has long been a major source of anxiety for farmers, as they lower crop value and output. In underdeveloped countries that rely on only one or two crops for their economies, PD can have devastating impacts, ranging from mild symptoms to the destruction of whole agricultural regions, which results in enormous costs and negative economic repercussions. However, in today's environment, farmers face a wide range of difficulties in their pursuit of agricultural success. As a result of widespread plant leaf diseases reducing agricultural yields, farmers no longer experience environmental problems. Concerns for the producers include how to detect PD quickly. It can be difficult to tell what kind of disease appears on a plant's leaf only by looking at it. Thus, it would be useful to have an autonomous system of specialists capable of doing so promptly. Diagnosing PD has become increasingly sophisticated as more and more disease signs in plants may be observed by simply looking at the leaves. This is because of the troublesome nature of farming and the myriad psychological issues these people face.

Current techniques of identifying plant conditions have limitations that have been studied at length. These are the difficulties that each individual cited:

- Pictures taken in real life have limited data.
- more accurate disease classification
- determining the stage of the disease

It was not addressed in the preceding work that came before CNN training to preprocess data. A high-performance level in the actual world is, therefore, quite useful. Few layers in the model led to poor training results.

B. Proposed Methodology

The analysis of native data has revealed that many DL models used for automated PD diagnosis were critically flawed. To achieve this goal, an ML system for categorizing PD was developed. Called PDConv. The PDConv model's block diagram in operation is shown in Figure 1.

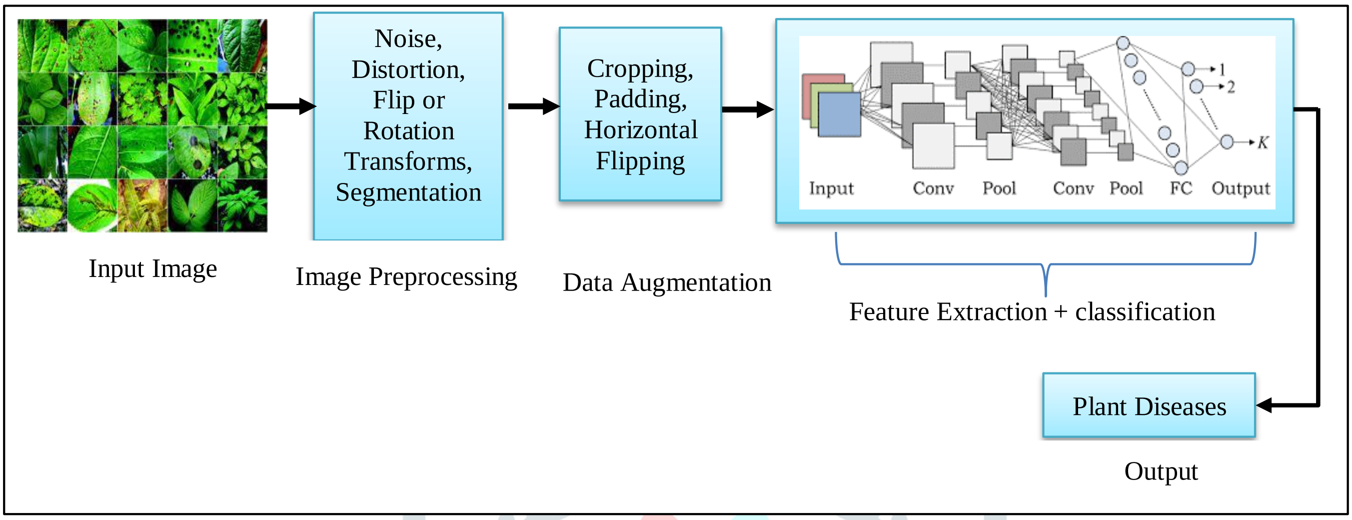

Figure 1: Flowchart of the suggested HAIL-RF model

This study found a workable solution using segmented image data for CNN models. The FGVC7 PD data collection is compiled in the repository database. Then, the pictures are put through a series of image processing preprocessing steps, including segmentation, classification, etc. It downsizes, changes the colors, and otherwise normalizes the raw images it took of leaves at this stage. When an image is read, the sound, distortion, or rotation are converted by 256x256 pixels, eliminating any random parts. Segmentation of images has also been performed in the pre-processing stage. Next, the dataset is resized to address the overfitting problem. Experts may increase the variety of data used to train models by data augmentation (DA), which does not need many new data to be collected. Researchers have turned to DA techniques such as horizontal flipping, cropping, and padding to train massive neural networks. Then, employ a CNN model to pull out and label features. The framework for the model is written in Python and uses TensorFlowTM. The presented framework enhances classification and prediction accuracy while minimizing overfitting.

a) Image Preprocessing

Preprocessing data is essential to many ML and AI-related professions, such as algorithm development and computer vision. Pre-processing methods are used to prepare a picture for further processing to increase its usefulness. In the preliminary processing phase of PD detection, the image is shrunk. Before being shown, the pictures undergo pre-processing, during which they are enhanced, their colors converted, and any unwanted noise is eliminated. Noise, distortion, flipping, and rotating the pictures are all employed after they have been collected. [12].

The pre-processing method for image zeroes in plants' problematic regions (disease-infected leaves). Image improvement, Segmentation, and color space conversion are commonplace in preprocessing images. To begin, the filter enhances the quality of the digital photo. After the picture has been filtered using the template, the RGB color space is used as the parameter for color. The picture is then further broken down into an easily analyzed important component. Unfortunately, it's not easy to get rid of the backdrop without the help of the user. Therefore, system automation is limited.

b) Image Segmentation

Segmentation is a technique of dividing a large picture into smaller ones.[13]. Thresholding is the simplest method of picture segmentation. Using thresholds, a greyscale image may be converted into a binary one. Segmentation of color pictures yields binary images. Segmentation is labeling each original picture pixel with two or more categories. Given more than two categories, several binary representations often emerge. In image processing, thresholding is commonly viewed as dividing a picture into smaller parts, or junks, using a threshold value of at least one color or greyscale to demarcate the border. Getting a binary image initially can help reduce the complexity of the data and streamline the identifying and categorizing steps.[14].

Create an image with the greyscale threshold values as the basis. Light or dark can be chosen as a target for an item. Plants.threshold.binary(gray_img, threshold, max_value, object_type="light") returns thresholded/binary picture

Parameters:

- gray_img - Grayscale image data
- threshold - Threshold value (0-255)
- max_value - Exceeding this value (255 = white) will have an effect.
- Object type - the opposite of "dark" (default: "light"). If an object is lighter than its backdrop, it has passed the typical threshold. The reverse threshold occurs if the item is darker than the background.

c) Feature Extraction

The detection of PD is complicated by problems with PD characteristic extraction. Textures, forms, colors, and attributes associated with motion all contribute to a picture's uniqueness and are thus important for feature extraction [15]. Based on the function map, the CNN extracts features. The convolution layer is employed to extract features from each picture input continuously. Each filter performs a dot product between itself and the input pixels to determine its effect on the image's raw pixel values. The filter's function map was made in two dimensions. Because of this, the network can discover the filters with expert input location (i.e., curves, edges).

d) Data Augmentation

The data augmentation (DA) technique uses simple data manipulation and advanced image transformation techniques to generate new data from existing training data. Techniques for expanding data can increase the size of an existing dataset without requiring new data collection. The most common imaging methods are inverting, cropping, rotating, colorizing, and adding noise to an image. These methods use various picture formatting techniques to produce visually appealing final products. To avoid overfitting, if the initial dataset is too small to train on, or if we need to squeeze our model more effectively, it is advised that we use DA. Some DA approaches and their potential benefits are discussed (blur, contrast, illumination, scaling, rotation, and projective conversion) (Marzougui et al., 2021).

e) Classification

To classify a picture means to assign a label to each of its pixels or vectors according to a predetermined set of criteria. The classification rule may use one or more spectral or textural properties. In this instance, CNN has been used to categorize a dataset of plant leaf images. CNN's are used for picture recognition and categorization because of their high level of accuracy. The CNN is based on a hierarchical model that creates a network (like a funnel) and then generates a fully connected layer that links all neurons and processes the output.

C. Deep Convolutional Neural Network (CNN) Model

When processing images and analyzing data, the emerging technique of DL offers accurate findings and shows tremendous potential. He has just branched out into agriculture after seeing the success of DL in other fields. Therefore, we have developed a DL-based automated method for detecting plant leaf diseases. DL requires sophisticated NNs and may capture multi-level, hierarchical pictures. For this purpose, one of the most powerful and fundamental DL techniques for modeling complex processes and pattern identification is utilized CNNs. The input, such as an image of the imperfect world, is "mapped" to the desired output by CNN. Since CNNs have shown exceptional performance in machine vision, this study applies CNNs to train specialized models for recognizing and diagnosing plant diseases using just leaf images. Significant progress in image processing has been made thanks to the evolution of DL techniques, particularly CNN.

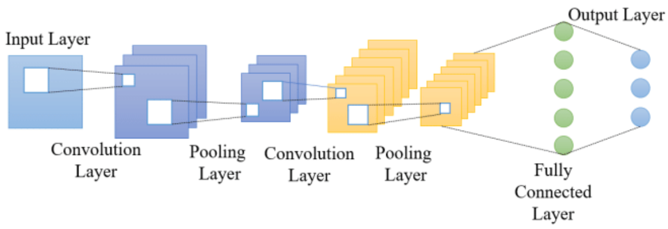

Figure 2: Architecture of Convolutional-Neural-Network Layers

A CNN has three main layers: the convolution layer, the pooling layer, and the fully linked layer. Convolution's primary use is in the automated feature elimination process (Fig 2). Multiple filters for learning are available. With raw pixel image sliding window values, each filter is utilized to compute the dot product of a filtered pixel and the input pixel. As a result, we have what is called the feature map, which is a 2-dimensional filter activation map. Thus, the network acquires knowledge of the filters if the input is located using prior information (i.e., curves, edges). CNN discovers their respective ideals throughout training. After these convolution layers come to the subsampling layers. Every subsampling layer's map size is decreased, and support for low-rotation and low-translation input variations is included. Sub-windows within each feature chart display the pooling layer's outputs, which are sized according to the greatest activation value of the input layer. A model based on SoftMax is applied to a fully connected layer that comes after the model to compute class scores. SoftMax classification takes a vector of learning features as input and predicts a class label for the output. The "PDConv" architecture is used in several forms of convolutional neural networks.[16].

IV. RESULTS ANALYSIS AND DISCUSSION

An in-depth discussion and analysis of the developed model and its findings were given here. The FGVC7 data set is useful for training DL plant disease classification techniques to quickly and accurately identify plant pathogens. Results from tests measuring a range of performance indicators have been collected. When compared to CNN, the PDConv model produced superior results.

A. Dataset Information Analysis

Many species of plants, both healthy and injured, are included in this dataset, using images drawn from various freely available sources. This compiles data from the open-source data collection available to the public about the Fine-Grained Recognition Challenge presented at FGVC7. In this competition, participants will practice semi-supervised learning on data that is only partially labeled. Fine-grained class similarity, a substantial imbalance between domain and classes, and differences between the unlabeled and labeled data are some of the challenges this data set aims to highlight for realists. The FGVC7 workshop at CVPR 2020 will feature this challenge. At the discretion of the workshop organizers, leading contributor teams may discuss and present their research during the session. Numerous plant species and diseases are represented in this data set. Appropriate data sets are required throughout the whole object recognition research process, from the early stages of concept development through the final assessments of the performance of detection techniques. The medical illustrations in the data sets came through online searches using disease and plant names in several languages, including Latin, Serbian, English, German, and Hungarian. The dataset included 15 categories for the pictures. A total of 13 groups discussed leaf-observable PD. A new class was added to the dataset to distinguish between healthy and sick leaves. Images of just flourishing leaves are included. An additional category in the dataset of background images helped achieve a more precise categorization.

B. Dataset Preparation

As the images had different proportions, they were resized to 196 by 196 pixels. This resolution was the same for retaining image characteristics and minimizing processing time. All of the images were initially saved in an RGB 3 channel, and a little script was built to append a number to the numerical value of each image. Photographs with the prefixes "bacterial" and "fungal" have been assigned the value "0,". In contrast, pictures with the prefix "healthy" have been assigned the value "1." Images are stored in the mind in the same order as they appear in the folder, they came in. Separating the dataset into testing, validation, and training subsets guarantees that neither the training nor the testing subsets will contain sufficient or any images from the other class. If the test set is drawn from the loaded images in the loaded dataset, then the classification performance cannot be accurately evaluated. Because the names of the files dictated the sequence of the pictures, a generator of random numbers was used to generate a new order before recording the pictures.

C. *Data Split*

The dataset was split in half: one half was used for training the classifier, while the other half was used to validate the training results. To detect biases from occurring unnecessarily if the dataset only contains photos from a random Python category, all divisions provide a 70/30 ratio for the leaves image class. The random number generator was used to restore visual order and safeguard data sets from being exposed to unnecessary external simplicity. Images of leaves with varying forms, angles, disease severity, the leaf's primary color, ambient brightness, light source, and so on will provide the possible diversity across all groups. The dataset's variety ensures that the model is trained and evaluated across the broadest possible range of conditions.[17].

D. *Experimental Results*

This section contains findings about training and validation of the database, which contains enhanced images. CNNs have been proved capable of learning features when trained on larger datasets. This part describes screenshots of the results obtained while utilizing the suggested model to conduct experiments. In addition, the suggested method was used throughout the classification phase, namely with the CNN and PDConv model classifications. Studies with the best outcome and a description of the models are given here. These assessments were conducted using a grayscale and RGB picture dataset.

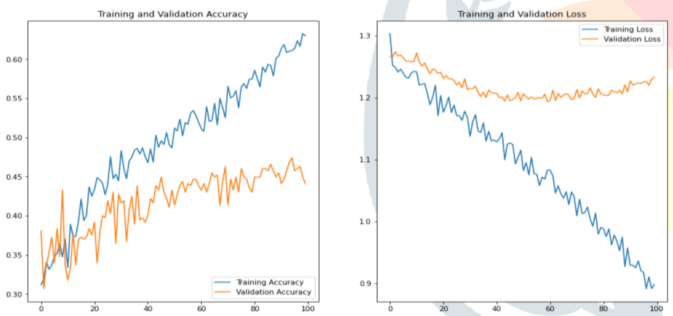

Figure 3: CNN Model's accuracy and loss graph

The present CNN model's accuracy vs. loss curve is depicted in Figure 3. These findings show that the CNN algorithm improves its plant picture outputs after 100 iterations. Next, the dataset is split into 80% for training and 20% for validation. While DA is done to the training dataset, the CNN model is trained on 1821 pictures. The dataset is subject to extensive experimentation throughout training. The method is taught using data from the validation set. Using the validation set, the method explains guidance on how to properly update its weights, increasing the value while minimizing the likelihood of overfitting.

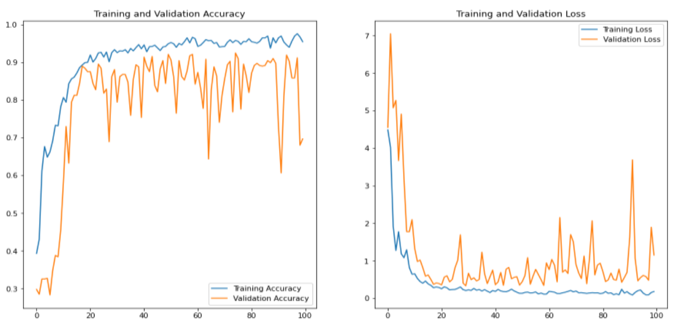

Figure 4: The accuracy and loss graph of the PDConv Model

The outcome, displayed in a graph of the recommended model's training and validation accuracy, demonstrates that DA alone effectively addresses overfitting. Visualizations of the recommended model's training and validation loss are also included. The findings suggest that the PDConv technique improves its plant picture outputs after 100 iterations. It improves the precision of training and reduces the overall loss. The PDConv accuracy and loss graph is depicted in Figure 4. After that, we split the dataset so that 70% is used for training and 30% is used for validation. Figure 4 shows that PDConv outperforms the baseline model in terms of accuracy and detection impact during model training.

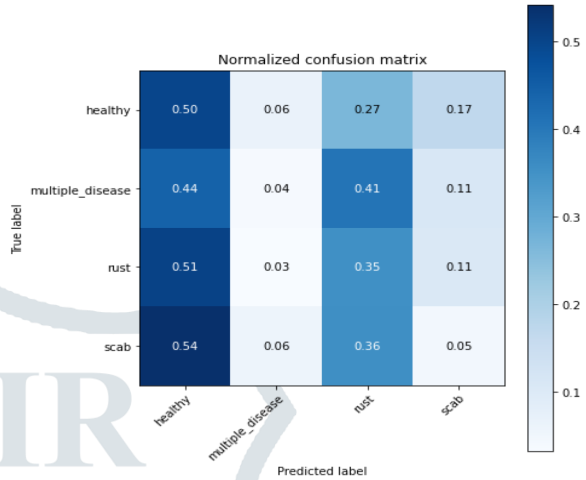

Figure 5: Plant disease confusion matrix with normalization

In figure 5, we see that the model's accuracy in identifying PD is supported by the confusion matrix (CM) shown above, which indicates that the model performs properly when validated on the pictures it already knows or was trained on. Correctly labeled figures are shown along the diagonal of the matrix. The true classes are shown in rows, while the projected label is shown in columns.

Table 1: Table showing accuracy, Val loss, loss, and Val accuracy comparison between baseline CNN and recommended PDConv model

Models	Training Loss	Training Acc	Validation Loss	Validation Acc
Existing CNN	0.8985	0.6299	1.2327	0.4411
Proposed PDConv	0.1788	0.9546	1.1523	0.6967

The suggested PD detection system's performance is backed by accuracy and various essential performance indicators, as shown in Table 1. To demonstrate the performance of the proposed method, conducted by contrasting the CNN with the PDConv. The findings of testing the PDConv model on RGB and grayscale pictures with constant learning parameters are summarized in the table above. As we can see, adjusting any variables influences the model's validation, training loss, and accuracy.

V. CONCLUSION

Economic and agricultural losses can be attributed in part to PD. Accurately diagnosing a medical condition is a challenging and knowledge-intensive endeavor. His illness manifests itself in the form of colored spots or streaks on the leaves of infected plants. Bacteria, Fungi, and viruses are common microbial culprits in PD. This study presents a PDConv model that uses DL to identify PD. Low performance was an issue with the prior model because it was trained without preprocessing. As a result, the difficulties in the previous research employing the CNN model have been resolved. This approach considers a

Deep CNN improvement called the PDConv model to create an autonomous PD classification model that can better identify plant diseases. Plant leaf disease detection using images is addressed by employing the PDConv DL architecture. In this research, we build a CNN for PD detection that can distinguish between a wide range of plants and pathogens. Preprocessing, classification, data enrichment, feature extraction, and the method's four main steps. All of these stages and their respective procedures have been outlined. The FGVC7 dataset is utilized in this study for model development. The FGVC7 dataset is useful for training DL plant disease classification techniques to quickly and accurately identify PD. Results from tests measuring many aspects of performance have been tallied. The PDConv model achieved better results than CNN. An alternative to using whole images for CNN model training is to use segmented and annotated images. Independent data performance went from 61.91% to 95.46% when the PDConv model was trained on the pre-existing CNN model. Further, a statistical analysis of one's trust in their classifications found a notable improvement, with 82% of the test data set showing a confidence boost.

VI. RECOMMENDATION AND FUTURE DIRECTIONS

Extending the suggested work might be the subject of future research, with experiments conducted on data with a higher proportion of damaged leaves and possibly even on data with many illnesses present in a single leaf (diseases with more than one). Diseases affecting plants are the biggest hindrance to global agricultural progress, and they negatively impact smallholder farmers in many developing nations. As a result, both cheap and effective, autonomous PD detection techniques can be invaluable in providing early warnings and predictions that lessen the burden of trying to stem the spread of disease. As an added bonus, we can raise the percentage of accurate predictions by incorporating optimization strategies into CNN structures. It was suggested that a more powerful and hybrid framework be developed in the future for reliable PD identification. We also suggest exploring the application of several ML techniques across many classes utilizing additional datasets of agricultural diseases.

REFERENCES

[1] P. K. Sethy, N. K. Barpanda, A. K. Rath, and S. K. Behera, “Deep feature based rice leaf disease identification using support vector machine,” *Comput. Electron. Agric.*, 2020, doi: 10.1016/j.compag.2020.105527.

[2] J. Chen, J. Chen, D. Zhang, Y. Sun, and Y. A. Nanehkaran, “Using deep transfer learning for image-based plant disease identification,” *Comput. Electron. Agric.*, 2020, doi: 10.1016/j.compag.2020.105393.

[3] X. Bai *et al.*, “Rice heading stage automatic observation by multi-classifier cascade based rice spike detection method,” *Agric. For. Meteorol.*, 2018, doi: 10.1016/j.agrformet.2018.05.001.

[4] A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes, “Deep learning for image-based cassava disease detection,” *Front. Plant Sci.*, 2017, doi: 10.3389/fpls.2017.01852.

[5] T. Vijaykanth Reddy and K. Sashi Rekha, “Deep Leaf Disease Prediction Framework (DLDPF) with Transfer Learning for Automatic Leaf Disease Detection,” 2021, doi: 10.1109/ICCMC51019.2021.9418245.

[6] A. KP and J. Anitha, “Plant disease classification using deep learning,” 2021, doi: 10.1109/icspc51351.2021.9451696.

[7] E. Hirani, V. Magotra, J. Jain, and P. Bide, “Plant Disease Detection Using Deep Learning,” 2021, doi: 10.1109/I2CT51068.2021.9417910.

[8] F. Mohameth, C. Bingcai, and K. A. Sada, “Plant Disease Detection with Deep Learning and Feature Extraction Using Plant Village,” *J. Comput. Commun.*, 2020, doi: 10.4236/jcc.2020.86002.

[9] M. Kumar, P. Gupta, P. Madhav, and Sachin, “Disease detection in coffee plants using convolutional neural network,” 2020, doi: 10.1109/ICCES48766.2020.09138000.

[10] S. V. Militante, B. D. Gerardo, and N. V. DIonisio, “Plant Leaf Detection and Disease Recognition using Deep Learning,” 2019, doi: 10.1109/ECICE47484.2019.8942686.

[11] K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis,” *Comput. Electron. Agric.*, 2018, doi: 10.1016/j.compag.2018.01.009.

[12] G. Kaur, S. Kaur, and A. Kaur, “Plant disease detection: A review of current trends,” *Int. J. Eng. Technol.*, 2018.

[13] V. Singh and A. K. Misra, “Detection of plant leaf diseases using image segmentation and soft computing techniques,” *Inf. Process. Agric.*, 2017, doi: 10.1016/j.inpa.2016.10.005.

[14] K. Bhargavi and S. Jyothi, “A Survey on Threshold Based Segmentation Technique in Image Processing,” *Int. J. Innov. Res. Dev.*, 2014.

[15] Y. Guo *et al.*, “Plant Disease Identification Based on Deep Learning Algorithm in Smart Farming,” *Discret. Dyn. Nat. Soc.*, 2020, doi: 10.1155/2020/2479172.

[16] S. Nigam and R. Jain, “Plant disease identification using Deep Learning: A review,” *Indian Journal of Agricultural Sciences*. 2020.

[17] E. Adama, “Tomato Leaf Diseases Detection and Classification using Convolutional Neural Network (CNN),” 2020. .