Cureus Journal of Computer Science

Open Access Original Article

AIDoctor-Plant: Lightweight and Bilingual Plant Disease Diagnosis and Remedy Recommender System

Received 04/18/2025
Review began 04/23/2025
Review ended 06/03/2025
Published 06/11/2025

DOI: https://doi.org/10.7759/s44389-025-05171-2

Punam Bedi [1] , Pushkar Gole [1] , Sudeep Marwaha [2]

1. Department of Computer Science, University of Delhi, New Delhi, IND 2. Division of Computer Applications, ICAR-Indian Agricultural Statistical Research Institute, New Delhi, IND

Corresponding author: Pushkar Gole, pgole@cs.du.ac.in

Abstract

Diagnosing plant diseases and providing advisory for disease remediation are crucial for the agricultural sector's growth, as it increases the crop yield and farmer's profit. In literature, numerous works have been done to detect plant diseases and provide advisory for disease remediation. However, all existing works provide advisory without considering the severity of disease, which can significantly affect disease management practices. Moreover, some existing works follow client-server architecture and thus rely on an active Internet connection for plant disease diagnosis, which may not be available in remote locations. Hence, in order to mitigate these research gaps, a lightweight and few-shot framework named "AIDoctor-Plant" is proposed and developed as an Android mobile application for plant disease diagnosis and remedy recommendation in this paper. This application works in both English and Hindi, and it also considers disease severity to generate disease management advisory. The effectiveness of the AIDoctor-Plant application is evaluated with the help of Apple-Tree-Leaf-Disease-Segmentation dataset. Experimental results demonstrate that the developed application can detect diseases with 95.34% accuracy and segment diseased areas from leaf images with a mean intersection over union of 95.43%. Furthermore, Student's t-test has confirmed that the developed application precisely measures the plant disease severity with 99% confidence interval. This makes AIDoctor-Plant application suitable to be deployed in the real world for effectively and efficiently diagnosing plant diseases even in remote locations and for getting remedy measures to manage the identified disease at the estimated severity index.

Categories: AI applications, Computer Vision, Deep Learning
Keywords: plant disease diagnosis, plant disease severity estimation, remedy recommendation, mobile application, bert

Introduction

Agricultural sector provides food to majority of the world's population, which is exponentially increasing day by day. In order to fulfill such huge food demand, the growth of the farming sector is necessary, and it is hampered by various challenges like supply chain management, optimal resource management, pest disease management, etc. [1,2]. Artificial intelligence (AI) is revolutionizing various real-world domains like healthcare, education, social networking, etc. Therefore, various agricultural researchers have utilized different AI-based techniques to mitigate challenges in the growth of the farming sector by solving various agricultural problems like crop-yield estimation, plant stress measurement, plant disease diagnosis, etc. [3].

Plants are susceptible to various pathogens like bacteria, fungi, viruses, etc., which can cause infection at different growing stages. Plant diseases can significantly reduce crop yield and, consequently, farmer's profit. Therefore, agricultural scientists are constantly engaged in developing a cost-effective system that can diagnose diseases in plant at the earliest, as it would be a viable solution to minimize crop yield loss and maximize farmers' income [4]. Initially, diseases in plants were identified by farmers and plant pathologists via manual examination of plants, and then they measure disease severity with their expertise. After disease identification and severity estimation, plant pathologists suggest some remedies to manage the identified disease at the estimated severity level. Subsequently, advancements in the computer vision domain have enabled automatic plant disease diagnosis and remedy recommendations.

In literature, various researchers have designed and developed different frameworks for automatic plant disease diagnosis and remedy recommendations [5-9]. Although most of the existing research works have considered the type of disease to provide the advisory for disease management, the severity of disease can also significantly affect the advisory for managing the plant disease. However, different frameworks are also developed in the literature for diagnosing diseases in plants and recommending remedies for their management.

Moreover, some of the existing frameworks are based on the client-server architecture in which the farmer (client) captures the image of an infected leaf through their smartphone. Then, this image is sent to the centralized server. Afterward, the server identifies the disease using the received leaf image, and the corresponding remedy is recommended to the farmer. This process requires an active Internet connection, which may not be available in remote locations. This limits the usage of existing frameworks in the plant disease diagnosis process and hampers the timely delivery of advisory for managing plant diseases. Hence, in order to address these limitations of existing research works, a lightweight and few-shot recommender system named AIDoctor-Plant is proposed in this paper for detecting diseases in plants and generating advisory in English and Hindi language to manage the identified plant disease by also considering the disease severity value.

For example, Shrimali [10] developed a mobile application named "PlantifyAI" to identify different diseases from plants. The author utilized PlantVillage dataset for training 16 different convolutional neural network (CNN) models to get the best model for plant disease detection. Out of the 16 CNN models, MobileNetV2 with Canny Edge Detection filters outperformed other counterparts by achieving 95.7% accuracy and 96.1% F1-measure. Thereafter, the trained MobileNetV2 model was converted into ".tflite" format for further deploying it into the Android mobile application using the Tensorflow Lite library of Python. Their mobile application also provides advisory for the management of identified diseases. Similar work was also done by Ng et al. [11] for identifying diseases from grape plant leaf images. They have trained the Faster-R-CNN model for localizing infected lesions in grape plant leaf images. They also converted the trained model to the tflite format for using it in the Android mobile application. Their mobile application does not provide advisory for the remediation of diagnosed diseases.

Ahmed and Reddy [12] developed an Android mobile application for identifying plant diseases from digital leaf images. They utilized the PlantVillage dataset to train a novel CNN model with approximately four million trainable weight parameters. As per their research work, the CNN model achieved 94% accuracy in detecting plant diseases. The authors of paper developed an Android mobile application that can identify the disease of plants with the help of trained CNN model deployed on a centralized server. In another work, Kumar et al. [13] developed a cloud-based framework named Deeplens Classification and Detection Model to diagnose diseases from plant leaf images. Their proposed solution was built on SageMaker framework of the Amazon Web Services portal, which utilizes the PlantVillage dataset for training. The major disadvantage of the aforementioned research works [12,13] is that they send the suspected leaf image to the centralized server for disease detection, which cannot work if the mobile device has a slow Internet connection. Moreover, they do not provide remedies in their application for managing the identified disease.

Janarthan et al. [14] developed a novel web-based framework named "Plant Pathology on Palms (P2OP)" for detecting plant diseases from farm fields. They proposed a novel model for plant disease detection based on CNN embedded Siamese Networks and trained it on the combination of three datasets, namely, the Apple dataset, Citrus leaf dataset, and Tomato leaf dataset (extracted from the PlantVillage dataset). Furthermore, their proposed model achieved 97.35%, 97.01%, and 96.36% accuracy in detecting diseases of Apple, Citrus, and Tomato leaves, respectively. Thereafter, they converted their model into tflite format for further deployment on mobile devices. Rimon et al. [15] developed an Android mobile application named "PlantBuddy" for plant disease identification through their digital leaf images. They trained MobileNetV2 and InceptionV3 CNN models on the PlantVillage dataset for identifying plant diseases through smartphones. As per their paper, the MobileNetV2 model outperformed the InceptionV3 model by achieving 95.75% accuracy. The authors also provided the advisory to manage the identified disease along with a facility to contact the nearest agricultural office based on the user's current location.

Yulita et al. [16] designed an Android mobile application for tomato plant disease diagnosis. Their application utilized the DenseNet-201 CNN model for diagnosing diseases in tomato plants, and it obtained an accuracy of 95.4% in detecting nine diseases of tomato plants. In order to facilitate mobile-based disease diagnosis, the authors of paper [16] embedded the trained DenseNet-201 model in the mobile application via Tensorflow Lite library of Python. In another study, Khan et al. [17] utilized the Vision Transformer model for disease detection in tomato plants. In their paper, the authors referred to the transformer model as "TomFormer," and its training was done on three datasets, namely, PlantDoc, PlantVillage, and KuTomaDATA (collected from farm fields of Abu-Dhabi). Furthermore, their proposed TomFormer model achieved 87%, 81%, and 83% accuracy on the above-mentioned datasets. They also deployed the proposed model in a robotic device for easy and accurate disease identification in farm fields.

Pineda Medina et al. [18] explored the potential of five predefined CNN models, namely, VGG-16, VGG-19, InceptionV3, MobileNetV2, and Xception, in diagnosing potato plant diseases. These models were trained on potato plant leaf images extracted from the PlantVillage dataset. The authors claimed in their paper that the MobileNetV2 model outperformed others by attaining 98.7% accuracy. After finding the best model among others, the authors of the paper developed an Android mobile application to facilitate the farmers in disease identification. Elinisa and Mduma [19] evaluated the efficacy of their custom CNN model in banana plant disease detection. As per the paper, their proposed model achieved 80.77% accuracy. This CNN model is converted into tflite format to be further utilized in Android mobile application development. They also provided the advisory for managing the identified disease. However, they have not considered disease severity for advisory generation. The summary of all the above-discussed research works is given in Table 1.

Research Work	Crop/Dataset	Techniques Used	Accuracy	Advisory (Yes/No)	Require Internet Connection (Yes/No)	Considered Severity in Advisory (Yes/No)	Number of Weight Parameters
Shrimali [10]	PlantVillage	MobileNetV2 + Canny Edge Detection filters	95.7%	Yes	No	No	3.5 million
Ng et al. [11]	PlantVillage	Faster-R-CNN	97.9%	No	No	No	-
Ahmed and Reddy [12]	PlantVillage	Custom CNN	94%	No	Yes	No	4 million
Kumar et al. [13]	PlantVillage	DCDM + CNN	98.7%	Yes	Yes	No	-
Janarthan et al. [14]	Apple	CNN embedded Siamese Networks	97.35%	No	No	No	0.26 million
	Citrus		97.01%				
	PlantVillage		96.36%				
Rimon et al. [15]	PlantVillage	MobileNetV2	95.75%	Yes	No	No	3.5 million
Yulita et al. [16]	Tomato leaf images extracted from PlantVillage dataset	DenseNet201	95.4%	No	No	No	20.2 million
Khan et al. [17]	PlantDoc	Vision Transformer	87%	No	No	No	1.3 million
	PlantVillage		81%				
	KuTomaDATA		83%				
Pineda Medina et al. [18]	Potato leaf images extracted from PlantVillage dataset	MobileNetV2	98.7%	No	No	No	3.5 million
Elinisa et al. [19]	Banana	Custom CNN	80.77%	Yes	No	No	6.4 million

TABLE 1: Summary of existing research works based on diagnosing diseases in plants and recommending remedies for their management

CNN, Convolutional Neural Networks; DCDM, Diffusion-Conditioned-Diffusion Model

Most of the above-discussed mobile applications developed in literature can identify plant diseases and some of these applications can also provide advisory for disease remediation. However, the advisory given in these applications does not consider the severity of the identified disease, which can significantly affect disease management practices. Moreover, the mobile applications developed in research work done by Ahmed and Reddy [12] and Kumar et al. [13] follow the client-server architecture in which an active Internet connection is required for diagnosing plant diseases, which limits the usage of their systems in remote locations.

Hence, in order to address these drawbacks of existing works, a novel lightweight and few-shot framework named AIDoctor-Plant has been proposed and implemented as an Android mobile application for automatic plant disease identification and severity estimation. Furthermore, it also provides advisory to the farmers for managing the identified disease at the estimated severity level of the disease in both English and Hindi languages. In this research work, the AIDoctor-Plant can diagnose four types of apple tree diseases (Alternaria Leaf Spot, Brown Spot, Gray Spot, and Rust) and provide remedies for them in the form of advisories. In order to show the feasibility of the proposed system, advisories are generated for Alternaria Leaf Spot and Brown Spot (Marssonina Leaf Blotch) diseases of apple trees.

The healthy and diseased leaf images of apple trees are taken from the ATLDS dataset [20]. However, the system can also be trained to diagnose and provide remedies for other plant diseases. The proposed framework utilizes the PDSE-Lite framework's 2-shot image classification and segmentation models for plant disease identification and severity estimation [21]. However, any deep learning-based image classification and segmentation models can be used for diagnosing plant diseases and estimating their severity. The advisory for managing the diagnosed plant disease at the estimated severity value of the disease has been generated by fine-tuning the bidirectional encoder representations from transformer (BERT) model. The BERT model is fine-tuned on the text of two research papers [22,23], which described different practices required to manage Alternaria Leaf Spot and Brown Spot (Marssonina Leaf Blotch) diseases of apple tree leaves. However, one can also fine-tune the BERT model to generate advisory of other plant diseases in a similar way. Moreover, all deep learning models utilized in AIDoctor-Plant have been directly embedded in the mobile application to work seamlessly in remote locations where Internet connectivity may not be strong.

Materials And Methods

This section is divided into two subsections. The first subsection describes the proposed AIDoctor-Plant framework, and second subsection provides the details of various software and hardware configurations along with the dataset, which have been used in this research work.

Proposed AIDoctor-Plant framework

In this manuscript, a lightweight and few-shot framework named AIDoctor-Plant is proposed as well as implemented to diagnose diseases in plants automatically and recommend different disease management measures. Unlike existing research works, the proposed AIDoctor-Plant generates advisory for managing plant diseases by considering the disease type along with its severity. Moreover, to the best of our knowledge, there is no research work in the literature that has proposed a framework to generate advisory to manage the disease by taking the severity value into consideration. The flow diagram of proposed AIDoctor-Plant framework is depicted in Figure *1*, and its different modules are described below.

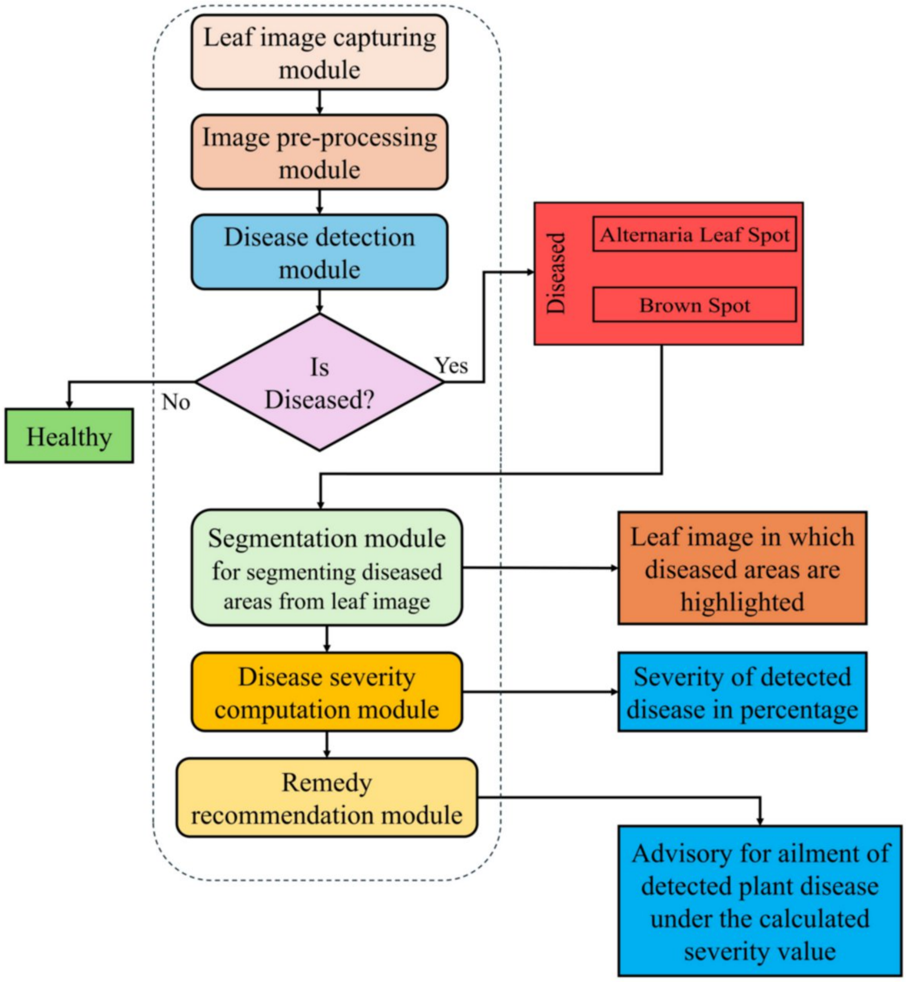

FIGURE 1: Flow diagram of proposed AIDoctor-Plant framework

Leaf Image Capturing Module

This module of the proposed AIDoctor-Plant framework captures the suspected leaf image by utilizing the digital camera of device, i.e., mobile phone, tablet, etc. Alternatively, farmers can also use the leaf image stored on their device's storage for disease diagnosis and to get advisory for disease remediation. In both scenarios, end-users can crop, rotate, and scale the leaf image obtained from the aforementioned methods. The crop, scale, and rotate options help the end-users in selecting the part of the image, which comprises only leaf. This step can significantly improve the performance of disease diagnosis because other objects rather than leaves can distract the models of the AIDoctor-Plant framework in disease diagnosis. After selecting the suspected leaf image either from the device's camera or device's storage, this image is then pre-processed via the Image Pre-Processing Module of AIDoctor-Plant, which is described in the following subsection.

Image Pre-Processing Module

The proposed AIDoctor-Plant framework is implemented as an Android mobile application using Java programming language. In the Android operating system, images are stored as a Bitmap in which each pixel value of image is represented by a 32-bit signed integer. Each 8 bits of 32-bit signed integer from the left denotes the value of Alpha (opacity), Red, Green, and Blue components of image pixels, respectively. This module first resizes the captured leaf image into 256 × 256, as the deep learning models used in the proposed AIDoctor-Plant framework have been trained on the input size 256 × 256. Thereafter, this module extracts the Red, Green, and Blue components of each image pixel in three arrays, namely red-array, green-array, and blue-array, with the help of Equations 1, 2, and 3, respectively. In these equations, $I_{x,y}$ represents the pixel of image located in x^{th} row and y^{th} column, where $x, y \in [0, 255]$. Furthermore, $0xFF$ denotes the hexadecimal representation of the decimal number 255. Additionally, >> and & represent the "binary right shift" and "binary AND" operator.

$$red - array = \frac{((I_{x,y} >> 16)\&(0xFF))}{255.0} \quad (1)$$

$$green - array = \frac{((I_{x,y} >> 8)\&(0xFF))}{255.0} \quad (2)$$

$$blue - array = \frac{(I_{x,y}\&(0xFF))}{255.0} \quad (3)$$

After extracting the Red, Green, and Blue components of image pixels, a three-dimensional integer array of size 256 × 256 × 3 is formed by stacking the red-array, green-array, and blue-array. This three-dimensional array is further passed to the Disease Detection Module of the AIDoctor-Plant framework for detecting the probable disease in the given leaf image. The Disease Detection Module of proposed framework is described in the following subsection.

Disease Detection Module

This module encompasses the 2-shot disease detection model of the PDSE-Lite framework [21]. However, any deep learning or machine learning model that is able to identify diseases in plants with the help of their digital leaf images can be employed in the Disease Detection Module of the AIDoctor-Plant framework. The few-shot disease detection model of the PDSE-Lite framework is a CNN-based model that can identify plant disease very effectively and efficiently despite utilizing only two training instances per class. Therefore, it mitigates the need for a large dataset of annotated leaf images to train the model. Additionally, this model requires only 8,749 trainable weight parameters; therefore, it can be embedded into any lightweight device like mobile phone, Raspberry Pi, etc., for accurate and fast plant disease detection. Although this model can detect plant diseases very effectively and efficiently, it cannot measure the severity of identified disease, which helps the farmers to prioritize the actions required for managing the plant disease. Therefore, in order to measure disease severity, the proposed AIDoctor-Plant system first segments the diseased lesions of leaf image by Segmentation Module, and thereafter, the Disease Severity Computation Module is used to measure the severity of the identified disease. The Segmentation Module has been described in the following subsection.

Segmentation Module

This module aims to segment the infected areas of given leaf image via the image segmentation process. In the proposed AIDoctor-Plant framework, the PDSE-Lite [21] framework's 2-shot image segmentation model is exploited to segment diseased areas of leaf images. This model also utilizes two training instances per class for precisely segmenting diseased lesions from input leaf images, thereby reducing the necessity of a large dataset of annotated leaf images for model training. Moreover, the 2-shot image segmentation model requires only 7,223 weight parameters. Hence, this model can also be used on low computational power devices to segment the infected regions of the input leaf image. After segmenting the disease areas from the leaf image, the proposed AIDoctor-Plant framework utilizes the Disease Severity Computation Module to calculate the severity of the identified disease. The Disease Severity Computation Module of proposed framework is described in the subsequent subsection.

Disease Severity Computation Module

This module computes the disease severity by using the segmented leaf image generated by the Segmentation Module of the proposed framework. In the segmented leaf image, the pixels of diseased areas are highlighted in red color, whereas the healthy pixels and background pixels are highlighted with green and black colors, respectively. In order to calculate the severity of the identified disease, this module first counts the number of diseased pixels (N_D) and healthy pixels (N_H), respectively. Thereafter, the disease severity (represented by s) is computed in percentage with the help of Equation 4.

$$s(in\%) = \frac{N_D}{N_D + N_H} \times 100 \quad (4)$$

After calculating the severity, the advisory for remediation of the identified disease under the computed severity value is generated with the help of the Remedy Recommendation Module of the proposed framework, which is described in the following subsection.

Remedy Recommendation Module

This module generates advisory to help the farmers in natural language for managing the identified disease under the calculated severity value. In order to generate disease managing advisory, a pre-trained BERT model (from the cdQA library) is used in inference mode with the context provided from two research papers [22,23] which have detailed descriptions of Alternaria Leaf Spot and Marssonina Leaf Blotch disease of Apple tree leaves and their management. Remedy Recommendation Module first builds a question, "How to manage disease *d* at severity *s*", where *d* and *s* represent the identified plant disease and severity value of *d* in percentage, respectively. This question is then answered with the help of BERT model's question-answering pipeline by utilizing the text of research papers [22,23]. The question-answering pipeline of BERT model generates answers to the given questions by finding similar text in the given text corpus. In this research work, the answer having maximum matching text is selected and then divided into bullet points for easy understanding of farmers. The quality of these generated advisories was manually validated by agricultural experts for correctness and practical relevance. Furthermore, the advisory generated through this module is also given in the Hindi language to help those farmers who do not know the English language.

All six modules of the proposed AIDoctor-Plant are combined together to build an Android mobile application that can diagnose plant disease, estimate disease severity, and provide advisory in English as well as Hindi language to manage the identified disease. Moreover, the application developed in this research work can even work in remote locations on any device with an Android operating system and camera. The following section describes the details of different experiments performed to evaluate the effectiveness of the AIDoctor-Plant framework in plant disease diagnosis and generating advisory for their remediations.

Experimental study

This section is divided into two subsections. First subsection describes the specifications of different hardware and software that have been utilized in this research work for implementing the proposed AIDoctor-Plant framework. Furthermore, the effectiveness of the proposed AIDoctor-Plant application has been evaluated with the help of ATLDS dataset, which is described in second subsection. Moreover, second subsection also discusses the steps performed to evaluate the performance of the proposed AIDoctor-Plant mobile application in plant disease diagnosis and remedy recommendation.

Hardware and Software Specifications

The implementation of the proposed AIDoctor-Plant framework has been done on a Dell G3 3500 laptop having Intel(R) Core (TM) i7-10750H CPU, 32 GB RAM, and Nvidia GTX1650 GPU with 4GB GPU memory. Moreover, the Samsung A54 5G mobile, which has an Exynos 1380 Octa-core CPU with 8GB RAM and 256 GB internal storage, has been utilized to test and execute the Android application developed in this research work. The Dell G3 3500 laptop is running on Windows 11 Home edition with 23H2 version, and the Samsung A54 5G mobile is running on One UI 6.1 (powered by Samsung) based on the Android 14 operating system.

The PyCharm professional edition version 2023.3.4 and Android Studio Iguana (version 2023.2.1) have been used in the work to write Python scripts and develop the AIDoctor-Plant Android mobile application. The user interface of mobile application is developed with the help of eXtensible Markup Language (XML), and its working is implemented through Java programming language. Additionally, Tensorflow version 2.4.4 library of Python has been used to convert the disease detection and segmentation models into ".tflite" format. These disease detection and segmentation models are further embedded into the mobile application for detecting disease in plant leaves and segmenting diseased areas from infected leaf images. The Tensorflow Lite libraries ("org.tensorflow:tensorflow-lite-support:0.1.0", "org.tensorflow:tensorflow-lite-metadata:0.1.0", "org.tensorflow:tensorflow-lite-gpu:2.3.0") for Android operating system have also been imported from Maven Repository into Android Studio for interacting with the pre-trained disease detection and segmentation models.

Moreover, the "com.github.yalantis:ucrop:2.2.6" library provides the image cropping functionality in the developed mobile application, and "me.biubiubiu.justifytext:library:1.1" aligns the text of advisory provided by the application. The advisory for managing the identified disease in the given leaf image under measured severity value is generated via pre-trained BERT model imported from the cdQA library [24]. Next subsection provides the description of ATLDS dataset, which is used to evaluate the effectiveness of AIDoctor-Plant mobile application. Moreover, the details of different experiments which have been conducted to evaluate the performance of proposed AIDoctor-Plant mobile application are also included in the following subsection.

Experimentation to Evaluate Performance of the Proposed AIDoctor-Plant Application

Apple (*Malus domestica*) is a prominent fruit crop grown in the northwestern region of India, and orchards of this crop suffer from various viral and fungal infections. Some common apple diseases are Alternaria Leaf Spot, Brown Spot (Marssonina Leaf Blotch), Apple Scab, Powdery Mildew, etc. [22]. This research work utilizes the healthy and diseased leaf images of apple trees infected with Alternaria Leaf Spot, Brown Spot

(Marssonina Leaf Blotch), Gray Spot, and Rust diseases. However, one can also train the models of the AIDoctor-Plant application with the leaf images of other apple diseases as well.

Alternaria Leaf Spot disease is caused by *Alternaria alternata* f. sp mali (a type of fungus), and it occurs after May when the temperature is high. Symptoms of this disease appear first in May as a small circle, which is purple or black in color [23]. Brown Spot (Marssonina Leaf Blotch) is caused by a fungus named *Marssonina coronaria*, which occurs in late summer on the mature leaves of apple trees. This disease appears on the upper side of the leaf as a grey-black spot, and later, it is converted into a larger area surrounded by red edges. Multiple brown spots lead to yellow leaves, which ultimately fall prematurely [22]. Apple Rust is caused by the fungus *Gymnosporangium juniperi-virginianae*. It typically appears in early spring when the weather is warm and moist. The disease begins as small yellow or orange spots on the upper surface of the leaves. As the disease progresses, these spots enlarge and develop a distinct red ring. In severe cases, the infection may spread to the fruit, causing blemishes and deformities. Gray Spot disease, caused by a fungus named *Cercospora mali*, appears during mid to late summer, primarily affecting older apple leaves. It begins as small, irregular, grayish-brown spots on the leaf surface, usually surrounded by a dark border. As the spots expand, they merge, forming large necrotic patches. Over time, affected leaves may turn yellow and fall prematurely, which can significantly reduce the tree's photosynthetic capacity and overall health.

Healthy and infected leaf images of apple trees have been extracted from the ATLDS [20] dataset to measure the effectiveness of AIDoctor-Plant mobile application in plant disease detection and severity estimation of identified disease. The leaf images of ATLDS dataset have been taken at different stages of infection and varying weather conditions. Moreover, approximately 52% of leaf images were captured in the controlled environment of laboratory, and 48% of leaf images were taken from real farm fields. This dataset also consists of segmentation masks for each leaf image to highlight the diseased and leaf regions in the image. The ATLDS dataset consists of 278, 215, and 409 leaf images belonging to Alternaria Leaf Spot, Brown Spot, and Healthy class, respectively. Some healthy and diseased leaf images and their segmentation masks obtained from the ATLDS dataset are shown in Figure *2*.

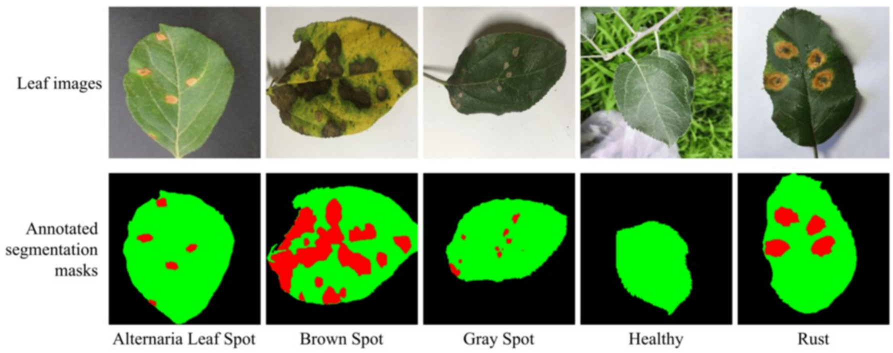

FIGURE 2: Some healthy and diseased leaf images and their segmentation masks obtained from the ATLDS dataset

ATLDS, Apple-Tree-Leaf-Disease-Segmentation

In order to evaluate the effectiveness of AIDoctor-Plant mobile application, this research work randomly selected 30 leaf images from the Alternaria Leaf Spot, Brown Spot, and Healthy class of the ATLDS dataset. These selected leaf images are utilized to measure the performance of application's Disease Detection Module by accuracy, precision, recall, and F1-measure metrics. Furthermore, diseased leaf images of the ATLDS dataset have also been used to measure the performance of AIDoctor-Plant application's Segmentation Module in segmenting diseased areas from given leaf images via MeanIoU and Dice-Score metrics defined in Equations 5 and 6, respectively. In these equations, M denotes the number of leaf images, and $\cup$ and $\cap$ represent the union and intersection operations, respectively. Moreover, Y_k and $Y_k^{'}$ are the k^{th} ground truth (Y) and predicted ($Y^{'}$) segmentation masks.

$$MeanIoU = \frac{1}{M}\sum_{k=1}^{M}\frac{|Y_k \cap Y_k^{'}|}{|Y_k \cup Y_k^{'}|} \quad (5)$$

$$DiceScore = \frac{1}{M}\sum_{k=1}^{M}\frac{2\left|Y_k \cap Y_k'\right|}{\left|Y_k\right| + \left|Y_k'\right|} \tag{6}$$

The effectiveness of the developed mobile application's Disease Severity Computation Module is evaluated by performing statistical hypothesis testing on actual and estimated disease severity values using the Student's t-test. This statistical testing has been done to test the null hypothesis H_0 and the alternative hypothesis H_1, as given in Equations 7 and 8, respectively, where $\bar{s}_g$ and $\bar{s}_e$ represent the means of the actual and estimated disease severity values. Moreover, the probability p is calculated in the Student's t-test at α = 0.01 or 99% confidence interval.

$$H_0 : \bar{s}_g - \bar{s}_e \neq 0 \tag{7}$$

$$(H_1 : \bar{s}_g - \bar{s}_e = 0 \tag{8}$$

Subsequent section discusses the results obtained from different experiments conducted in this research work to validate the AIDoctor-Plant application's effectiveness.

Results

This section comprises of two subsections. The first subsection provides the results obtained while measuring the performance of the proposed AIDoctor-Plant application. The second subsection discusses the features and gives the details of application's user-interfaces.

Experimental results

This section provides and discusses the results obtained from different experiments conducted in this research work to evaluate the performance of Disease Detection, Segmentation, and Disease Severity Computation modules of the AIDoctor-Plant mobile application. In order to evaluate the performance of the developed mobile application, testing has been done on 150 leaf images comprising of randomly selected thirty leaf images from each of the Alternaria Leaf Spot, Brown Spot, Gray Spot, Healthy, and Rust classes of the ATLDS dataset. The testing experiment is repeated ten times, and the results are obtained by taking the average. Performance of the Disease detection module of the mobile application has been measured in terms of accuracy, precision, recall, and F1-measure for each aforementioned class of the ATLDS dataset with the help of the confusion matrix plotted in Figure *3*. The obtained values for accuracy, precision, recall, and F1-measure are 95.34%, 94.52%, 95.56%, and 95.03%, respectively.

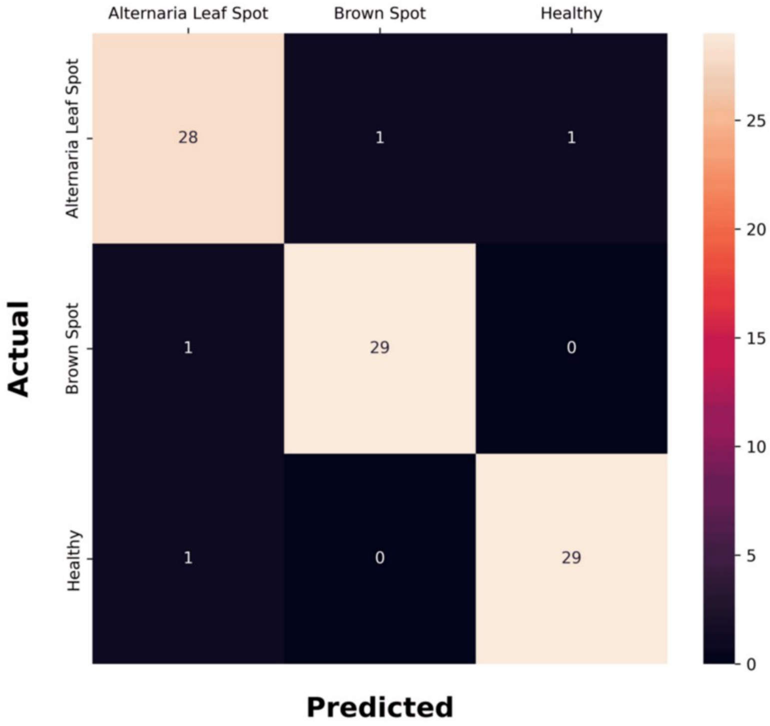

FIGURE 3: Confusion matrix for the predictions of disease detection module of AIDoctor-Plant

Performance of the Segmentation Module of mobile application has been measured only on diseased leaf images in terms of MeanIoU and Dice-Score metrics, as this module is invoked when the leaf images are identified as diseased. The values for aforementioned metrics have been plotted in Figure *4*. Results given in Figure *4* demonstrate that the Segmentation Module of AIDoctor-Plant can segment the diseased lesions of Alternaria Leaf Spot, Brown Spot, Gray Spot, and Rust diseases with 94.92%, 95.04%, 96.16%, and 95.59% MeanIoU, respectively. Moreover, it has achieved 96.57%, 96.49%, 97.51%, and 96.89% Dice-Score in segmenting diseased areas of Alternaria Leaf Spot, Brown Spot, Gray Spot, and Rust diseases, respectively.

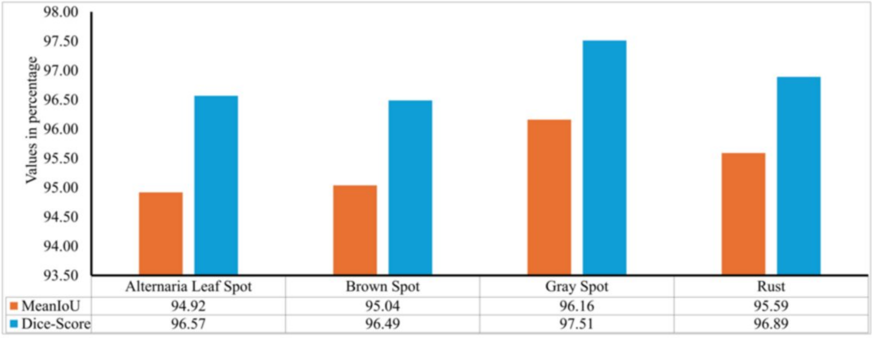

FIGURE 4: Class-wise MeanIoU and Dice-score for AIDoctor-Plant application's segmentation module

In order to evaluate the effectiveness of the Disease Severity Computation Module of the developed mobile application, statistical testing has been performed with the help of the Student's t-test by considering actual and predicted disease severity values. The p value is computed during experimentation at $\alpha = 0.01$, i.e., 99% confidence interval, and the p value obtained from experimentation is $0.008 < \alpha = 0.01$. In the Student's t-test, if the obtained p value is lesser than α value, then the null hypothesis (given in Equation 7) is rejected, and the alternative hypothesis (stated in Equation 8) is accepted with $(1 - \alpha) \times 100$ confidence interval. After analyzing the results of experiments, it is found that p value, i.e., 0.008, is less than α value, i.e., 0.01. Therefore, the alternative hypothesis has been accepted with 99% confidence interval, which states that the difference between means of actual and predicted severity values is zero. Hence, the predicted severity values do not significantly differ from actual severity values. This demonstrates the effectiveness of the Disease Severity Computation Module of the proposed AIDoctor-Plant mobile application.

Furthermore, the output generated by the application's Remedy Recommendation Module has been verified with the help of plant pathologists. After evaluating the effectiveness of each module of the AIDoctor-Plant mobile application, its functionalities are further compared in Table 2 with existing mobile applications available in the literature.

Mobile Application Name	Crop/Dataset	Advisory (Yes/No)	Require Internet Connection (Yes/No)	Considered Severity in the Advisory (Yes/No)	Is bilingual (Yes/No)	Application Size in Megabytes	Accuracy
Potato Crop Diseases (PCD)	Leaf images of potato plants extracted from the PlantVillage dataset	No	No	No	No	77.57	92.86%
Artificial Intelligence-based Disease Identification System for Crops (AI-DISC)	Rice, Wheat, Maize, Tomato, Mustard, Cotton, Brinjal, Soybean, Apple, Peach, Kinnow, Mandarin, Lemon, Chickpea, Green gram, Clusterbean, Mothbean, Cucurbits, Chilli, Coriander	Yes	Yes	No	No	97.34	91.11%*
AIDoctor-Plant	Apple	Yes	No	Yes	Yes	22.78	95.34%

TABLE 2: Comparison of functionalities provided in various mobile applications proposed in the literature for plant disease detection and remedy recommendation

*The developers of AI-DISC application have not measured the accuracy of their application. However, in order to compare its performance, its performance has been measured with the help of same test images of apple tree leaf images, which were used to measure the performance of AIDoctor-Plant application.

Features and user interfaces of the AIDoctor-Plant application

The mobile application developed in this research work is compatible with a smartphone having Android operating system 10 or higher and minimum 8 GB storage along with 2 GB of RAM. Moreover, the camera of device should be working in order to take photographs of suspected leaf images.

The AIDoctor-Plant application would start with a splash screen showing its logo, and the screenshot of welcome screen is shown in Figure *5*(a). Once the application is loaded, it shows the user interface for capturing the leaf image. The leaf image can be captured either from the camera or the gallery of mobile device. In order to start capturing the leaf image from the camera or gallery, the user needs to click icon given on the bottom right corner of the screen shown in Figure *5*(b).

This will provide two choices for the user, as depicted in Figure *5*(c). First choice is to capture the suspected leaf image from the camera, and the second choice is to use an already captured image from the device gallery. If the user selects the second option, then the user's device gallery is opened up (as shown in Figure *5*(d)) for choosing the suspected leaf image. Once the leaf image is selected from the device gallery, the leaf image is displayed on the screen, as shown in Figure *5*(e). Thereafter, in order to identify the probable disease in the given leaf image, the user needs to click the predict button highlighted with red colored rectangle in Figure *5*(e), and the diagnosis result is then prompted to the user, as shown in Figure *5*(f).

It can be seen from Figure *5*(f) that an option to view the infected areas of input leaf image (highlighted with a red colored rectangle) is also provided in the AIDoctor-Plant application. Selecting this option will run the Segmentation Module of the application, which takes the input leaf image and generates its segmentation mask, as shown in Figure *5*(g). In this segmentation mask, the diseased areas are highlighted in red color, and healthy areas are highlighted with green color. After segmenting diseased areas from the given leaf image, the Disease Severity Computation Module is invoked to estimate the severity of identified disease as per Equation 4, and its output is displayed on the user interface of the application, as shown in Figure *5*(g). Once the disease type and its severity value are determined, the user is prompted to view recommendations for managing the identified disease. If the user clicks on the 'View Recommendation' option (highlighted with the red colored rectangle in Figure *5*(g)), then the Remedy Recommendation Module is executed to generate the advisory to manage the identified disease under the calculated severity value. The user interface for providing the advisory to manage the identified plant disease under the estimated severity value is given in Figure *5*(h).

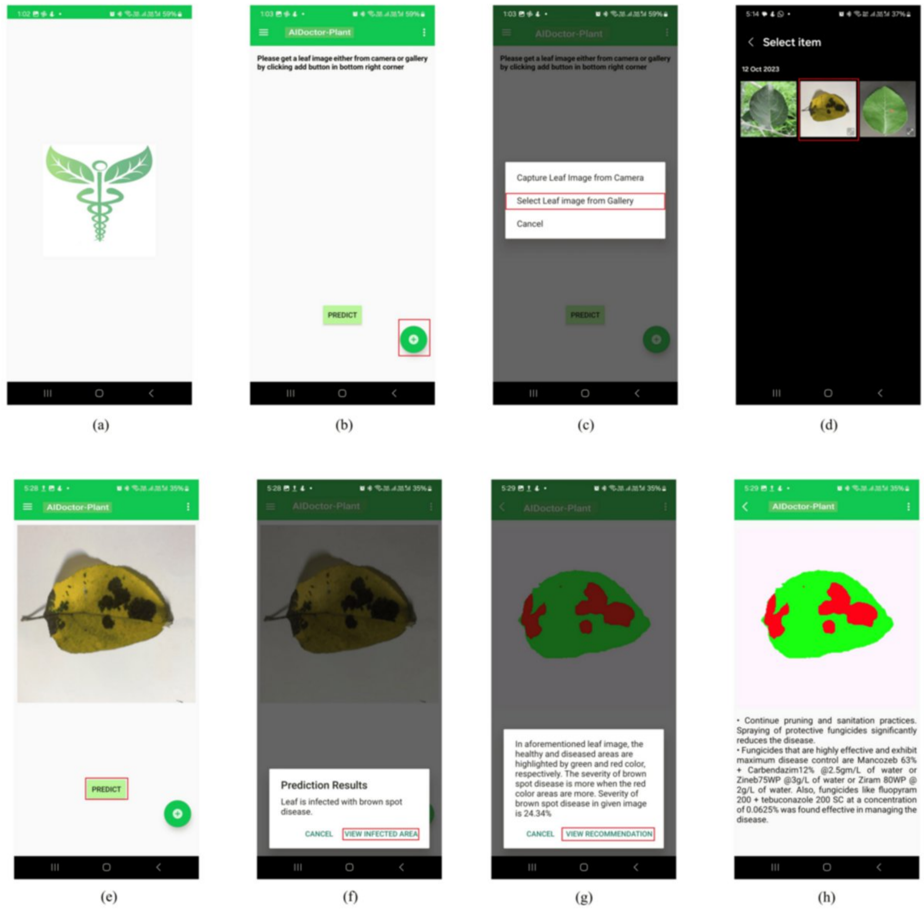

FIGURE 5: User interfaces showing the working of the AIDoctor-Plant application

In order to facilitate non-English-speaking Indian farmers, the AIDoctor-Plant application also provides an option to switch the language of the application (as shown in Figure 6(a)) to Hindi via clicking the three dots icon present at the top right corner of the screen. This will allow the user to choose the default language of the application, as shown in Figure 6(b). If the user has selected Hindi language as the default language, all user interfaces of application are translated into Hindi language. A snapshot of the application's user interface in Hindi language is given in Figure 6(c).

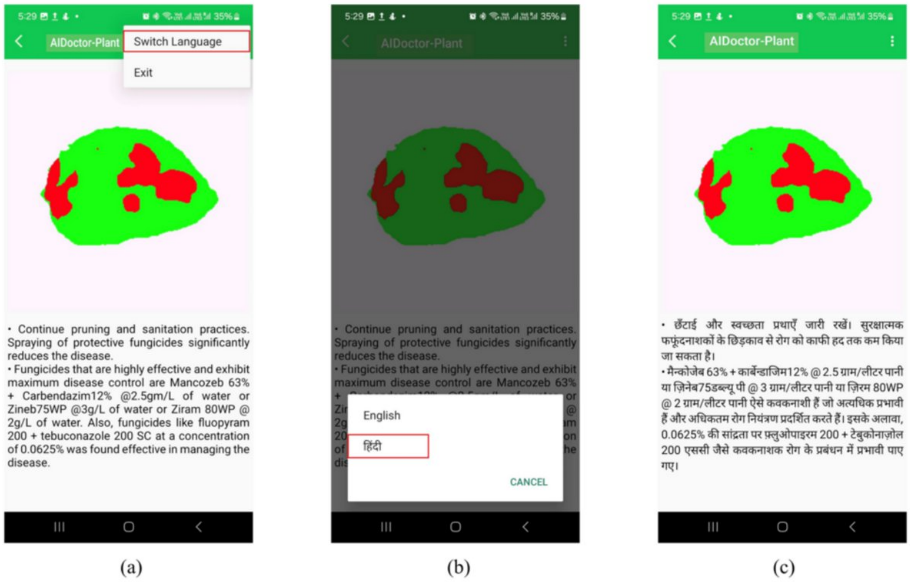

FIGURE 6: User interfaces to change the language of the AIDoctor-Plant application

Discussion

In this research work, an Android application named AIDoctor-Plant has been designed and developed for comprehensive plant disease management, including disease detection, severity estimation, and remedy recommendation. The application utilizes lightweight and few-shot image classification and segmentation models from the PDSE-Lite framework to perform disease diagnosis and severity estimation on devices with limited computational power.

Despite being compact in size (~22.78 MB), the application achieves a high disease detection accuracy of 95.34%, demonstrating the effectiveness of the underlying models in real-world conditions. This balance of performance and efficiency ensures the application's suitability for widespread adoption, particularly in rural areas where low-end smartphones are commonly used. For remedy recommendation, the BERT model has been fine-tuned on two research papers having detailed disease descriptions and their respective management practices. This enables the application to generate context-aware and scientifically grounded remedy suggestions, tailored to the specific disease and its estimated severity level.

To enhance usability and accessibility, the application provides advisory content in both English and Hindi, helping non-English-speaking users benefit equally from the tool. The offline functionality, bilingual support, and lightweight architecture make the application particularly relevant to the Indian agricultural context, where infrastructural limitations such as intermittent internet connectivity often hinder the adoption of digital tools. While the AIDoctor-Plant system demonstrates high accuracy and usability, it currently supports only four diseases of apple trees, i.e., Alternaria Leaf Spot, Brown Spot, Gray Spot, and Rust, with advisory generation available for two diseases. This limitation is primarily due to the availability of literature required for generating precise advisory content. Currently, the advisory language is limited to English and Hindi; thus, it restricts the applicability of the system to other regions of the country.

Conclusions

Diagnosing plant diseases and providing advisory for their remediation are essential aspects of agricultural management practices, as they can significantly reduce crop yield loss and thereby increase the farmer's profit. In literature, various state-of-the-art frameworks have been proposed for identifying diseases in plants and providing recommendations to treat the identified disease. Although many state-of-the-art plant disease detection frameworks are available in the literature, none of the existing research works have proposed a framework that can provide advisory to manage plant diseases by considering both disease type and severity. Additionally, some of the existing frameworks were based on client-server architecture. Therefore, they require an active Internet connection for plant disease diagnosis, which may not be available in remote locations. Hence, in this paper, a lightweight and few-shot framework named AIDoctor-Plant was proposed and developed as an Android mobile application to generate recommendations for managing the identified plant disease under the estimated severity value via a pre-trained BERT model.

Furthermore, the AIDoctor-Plant framework can also facilitate farmers in remote areas, as it does not require a high-speed Internet connection for disease diagnosis and generating recommendations for remediating the identified disease. This mobile application can also be used by non-English-speaking farmers, as it provides the option to change the language of application to Hindi. Effectiveness of the proposed AIDoctor-Plant mobile application was evaluated on 30 randomly selected leaf images from the diseased classes of the ATLDS dataset. Experimental results revealed that the AIDoctor-Plant mobile application can identify apple tree diseases with 95.34% accuracy and 95.03% F1-measure. Additionally, the application can segment diseased lesions of Alternaria Leaf Spot, Brown Spot, Gray Spot, and Rust diseases of apple trees with 94.92%, 95.04%, 96.16%, and 95.59% MeanIoU, respectively. The disease severity estimation ability of AIDoctor-Plant was further verified by performing statistical hypothesis testing on predicted and actual severity values of diseased leaf images via Student's t-test, and it proved that the developed mobile application could measure the identified disease severity with 99% confidence interval. Hence, the AIDoctor-Plant mobile application designed and developed in this research work can help farmers to diagnose plant diseases and take timely actions to manage the identified disease. Moreover, this application also computes the severity of the identified disease and considers it for recommending the necessary steps for managing the disease. Currently, the AIDoctor-Plant mobile application proposed in this paper can diagnose four diseases of apple tree leaves. As the data from experts for disease remedies were available for two diseases the Alternaria Leaf Spot and Brown Spot, therefore, the application includes advisories for these two diseases. However, in future, we plan to extend the AIDoctor-Plant framework to include additional apple diseases such as Apple Scab and Powdery Mildew. Furthermore, we aim to incorporate support for other crops like tomato, potato, and banana, using new datasets and expert-verified advisories. We also plan to integrate more Indian regional languages to improve accessibility and to develop feedback mechanisms for continuous model improvement based on farmer and expert input.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Pushkar Gole, Punam Bedi, Sudeep Marwaha

Acquisition, analysis, or interpretation of data: Pushkar Gole, Punam Bedi, Sudeep Marwaha

Drafting of the manuscript: Pushkar Gole

Critical review of the manuscript for important intellectual content: Punam Bedi, Sudeep Marwaha

Supervision: Punam Bedi

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

References

1. Adve VS, Wedow JM, Ainsworth EA, Chowdhary G, Green Miller A, Tucker C: AIFARMS: Artificial intelligence for future agricultural resilience, management, and sustainability. AI Magazine. 2024, 45:83-88. 10.1002/aaai.12152
2. Priya P, Patil M, Pandey P, Singh A, Babu VS, Senthil-Kumar M: Stress combinations and their interactions in plants database: a one-stop resource on combined stress responses in plants. The Plant Journal. 2023, 116:1097-1117. 10.1111/tpj.16497
3. Kalyanaraman A, Burnett M, Fern A, Khot L, Viers J: Special report: The AgAID AI institute for transforming workforce and decision support in agriculture. Computers and Electronics in Agriculture. 2022, 197:106944. 10.1016/j.compag.2022.106944
4. Gole P, Bedi P, Marwaha S, Haque MA, Deb CK: TrIncNet: a lightweight vision transformer network for identification of plant diseases. Frontiers in Plant Science. 2023, 14:1221557. 10.3389/fpls.2023.1221557
5. Siddiqua A, Kabir MA, Ferdous T, Ali IB, Weston LA: Evaluating plant disease detection mobile applications: Quality and limitations. Agronomy. 2022, 12:1869. 10.3390/agronomy12081869

6. Tembhurne JV, Gajbhiye SM, Gannarpwar VR, Khandait HR, Goydani PR, Diwan T: Plant disease detection using deep learning based mobile application. Multimedia Tools and Applications. 2023, 82:27365-27390. 10.1007/s11042-023-14541-8
7. Balasubramanian S, Rajaram S, Selvabharathy S, Krishnamurthy V, Sakthivel V, Ramprasad A: Mobile application for leaf disease classification. Advances in IoT and Security with Computational Intelligence. Mishra A, Gupta D, Chetty G (ed): Springer, Singapore; 2023. 755:297-307. 10.1007/978-981-99-5085-0_29
8. Ngugi LC, Abdelwahab M, Abo-Zahhad M: Tomato leaf segmentation algorithms for mobile phone applications using deep learning. Computers and Electronics in Agriculture. 2020, 178:105788. 10.1016/j.compag.2020.105788
9. Bedi P, Gole P: Plant disease detection using hybrid model based on convolutional autoencoder and convolutional neural network. Artificial Intelligence in Agriculture. 2021, 5:90-101. 10.1016/j.aiia.2021.05.002
10. Shrimali S: PlantifyAI: A novel convolutional neural network based mobile application for efficient crop disease detection and treatment. Procedia Computer Science. 2021, 191:469-474. 10.1016/j.procs.2021.07.059
11. Ng HF, Lin CY, Chuah JH, Tan HK, Leung KH: Plant disease detection mobile application development using deep learning. 2021 International Conference on Computer & Information Sciences (ICCOINS), Kuching, Malaysia. 2021, 34-38. 10.1109/ICCOINS49721.2021.9497190
12. Ahmed A, Reddy GH: A mobile-based system for detecting plant leaf diseases using deep learning. AgriEngineering. 2021, 3:478-493. 10.3390/agriengineering3030032
13. Kumar P, Raghavendran S, Silambarasan K, Kannan KS, Krishnan N: Mobile application using DCDM and cloud-based automatic plant disease detection. Environmental Monitoring and Assessment. 2023, 195:44. 10.1007/s10661-022-10561-3
14. Janarthan S, Thuseethan S, Rajasegarar S, Yearwood J: P2OP—Plant Pathology on Palms: A deep learning-based mobile solution for in-field plant disease detection. Computers and Electronics in Agriculture. 2022, 202:107371. 10.1016/j.compag.2022.107371
15. Rimon SI, Islam MR, Dey A, Das A: PlantBuddy: An Android-based mobile application for plant disease detection using deep convolutional neural network. Artificial Intelligence and Technologies. Raje RR, Hussain F, Kannan RJ (ed): Springer, Singapore; 2022. 806:275-285. 10.1007/978-981-16-6448-9_28
16. Yulita IN, Amri NA, Hidayat A: Mobile application for tomato plant leaf disease detection using a dense convolutional network architecture. Computation. 2023, 11:20. 10.3390/computation11020020
17. Khan A, Nawaz U, Kshetrimayum L, Seneviratne L, Hussain I: Early and accurate detection of tomato leaf diseases using TomFormer. 2023 21st International Conference on Advanced Robotics (ICAR), Abu Dhabi, United Arab Emirates. 2023, 645-651. 10.1109/ICAR58858.2023.10436499
18. Pineda Medina D, Miranda Cabrera I, de la Cruz RA, Guerra Arzuaga L, Cuello Portal S, Bianchini M: A mobile app for detecting potato crop diseases. Journal of Imaging. 2024, 10:47. 10.3390/jimaging10020047
19. Elinisa CA, Mduma N: Mobile-based convolutional neural network model for the early identification of banana diseases. Smart Agricultural Technology. 2024, 7:100423. 10.1016/j.atech.2024.100423
20. Apple Tree Leaf Disease Segmentation Dataset. (2023). Accessed: June 2, 2023: https://www.scidb.cn/en/detail?dataSetId=0e1f57004db842f99668d82183afd578&dataSetType=personal.
21. Bedi P, Gole P, Marwaha S: PDSE-Lite: lightweight framework for plant disease severity estimation based on Convolutional Autoencoder and Few-Shot Learning. Frontiers in Plant Science. 2024, 14:1319894. 10.3389/fpls.2023.1319894
22. Nabi SU, Parveen S, Raja WH, Javid IM, Yasmin S, Sheikh MA, Sharma OC: Diseases of apple (Malus domestica) and their management. Agrica. 2022, 11:32-40. 10.5958/2394-448x.2022.00003.7
23. Madhu GS, Un Nabi S, Iqbal Mir J, Hassan Raja W, Sheikh MA, Sharma OC, Singh DB: Alternaria leaf and fruit spot in apple: Symptoms, cause and management. European Journal of Biotechnology and Bioscience. 2020, 8:24-26.
24. Haystack: the end-to-end NLP framework for pragmatic builders. (2019). Accessed: November 22, 2024: https://github.com/deepset-ai/haystack.