IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 29 September 2022, accepted 10 October 2022, date of publication 19 October 2022, date of current version 27 October 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3215497

RESEARCH ARTICLE

Identification of Foliar Disease Regions on Corn Leaves Using SLIC Segmentation and Deep Learning Under Uniform Background and Field Conditions

HIEU PHAN[1], AANIS AHMAD[2], AND DHARMENDRA SARASWAT[3]
[1]Department of Computer Science and Software Engineering, Miami University, Oxford, OH 45056, USA
[2]Elmore Family School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907, USA
[3]Department of Agricultural and Biological Engineering, Purdue University, West Lafayette, IN 47907, USA

Corresponding author: Dharmendra Saraswat (saraswat@purdue.edu)

This work was supported in part by the Purdue University Summer Undergraduate Research Fellowship (SURF) Program, Wabash Heartland Innovation Network, under Grant 18024589; in part by the Foundation of Food and Agricultural Research under Grant 534662; in part by the United States Department of Agriculture (USDA) National Institute of Food and Agriculture Hatch Project under Grant 1012501; and in part by the Department of Agricultural and Biological Engineering.

ABSTRACT Plant diseases lead to severe losses in crop yield worldwide. The conventional approach for diagnosing diseases relies on manual scouting. In recent years, advances in convolutional neural networks have motivated the use of deep learning-based computer vision for automatically identifying plant diseases. Although image classification techniques are commonly used for analyzing agricultural data, their use for accurately identifying diseased regions corresponding to different disease types on individual plant leaves is limited. In this study, Simple Linear Iterative Clustering (SLIC) segmentation was used on corn leaf images from the PlantVillage and CD&S datasets to create super-pixels, a cluster of pixels representing a region of interest on a corn leaf. The VGG16, ResNet50, DenseNet121, Xception, and InceptionV3, pre-trained deep learning models were utilized to identify diseased regions corresponding to five super-pixel classes (healthy, northern leaf blight (NLB), gray leaf spot (GLS), common rust, and background) for the PlantVillage dataset and four super-pixel classes (NLB, GLS, northern leaf spot, and background) for the CD&S dataset, on corn leaves. After setting the spatial proximity value (sigma) for SLIC segmentation to five, a total of 100 models were trained by using different numbers of segments per image (five and fifteen) in the SLIC algorithm for both datasets and choosing training: testing split ratios of 90:10, 80:20, 70:30, 60:40, and 50:50 for each of the five models. The highest overall testing accuracy of 97.77% was observed after training the DenseNet121 model to identify super-pixels created from the CD&S dataset, belonging to the four classes created using a sigma value of five, five segments per image, and an 80:20 training: testing split ratio. Web and mobile applications were developed to identify diseased regions on corn leaves using the best deep learning model as the classifier. The results suggest that SLIC segmentation on corn leaf images helps accurately identify diseased regions. This research demonstrates the potential of image-based scouting as an efficient alternative to manual scouting for disease monitoring.

INDEX TERMS SLIC segmentation, deep learning, image classification, corn leaf disease identification, field conditions.

I. INTRODUCTION

Corn is an important crop for the USA as its share in total feed grain is over 95% [1]. However, diseases pose a major threat by reducing the corn yield by approximately 10%,

The associate editor coordinating the review of this manuscript and approving it for publication was Prakasam Periasamy.

which resulted in an average economic loss of USD 55.90 per acre in the U.S. and Ontario, Canada, from 2016 to 2019 [2]. Therefore, it is essential to develop a robust system for monitoring the foliar diseases of corn and implementing effective management practices to reduce yield loss.

Traditionally, disease monitoring systems relied on manual scouting and lab analysis, such as immune fluorescence, enzyme-linked immune-sorbent assay, or DNA-based and serological methods for disease identification [3], [4]. Although traditional disease monitoring approaches were effective, they were time-consuming [8] and unreliable at the asymptomatic stage [4]. With the recent development in Artificial Intelligence (AI), the agricultural research community has explored traditional machine learning (ML) approaches, including K-means clustering, Support Vector Machine, and Artificial Neural Networks (ANNs), for plant disease identification [5]. Although such methods had low training times, testing accuracies of under 80% were typically reported, as traditional methods were incapable of extracting and selecting features in complex and irregular diseased plant images [6].

Over the past few years, deep learning models have gained popularity due to their ability to learn robust features directly from images automatically. A simple CNN consisting of only three convolutional layers was used for disease identification from the PlantVillage dataset with accuracies of up to 94% [7]. An optimized dense CNN was proposed, which resulted in higher accuracies of up to 98.56% [8]. Additionally, an end-to-end deep learning framework was developed by combining the EfficientNetB0 and DenseNet121 models, which resulted in disease identification with accuracies of up to 98.56% [9]. Overall, many state-of-the-art models, including ResNet, DenseNet, MobileNet, Inception, and VGG, have achieved accuracies of up to 99.75% for identifying foliar disease images of different plants from the PlantVillage dataset [10], [11]. However, most images within the PlantVillage dataset were acquired under lab conditions with homogeneous backgrounds. Therefore, poor performances were observed when testing was conducted by providing data from complex field conditions to deep learning-based image classification models that were trained using the PlantVillage dataset [12]. Some studies used custom-acquired datasets, such as foliar diseases for coffee or apple, for training and fine-tuning deep learning models capable of identifying diseases with accuracy ranging from 90% to 95% [13], [14].

Most deep learning-based image classification studies focused on identifying diseases using images of entire leaves with various backgrounds rather than identifying diseased regions on individual leaves. Therefore, the idea of segmenting and focusing on diseased regions on plant leaves was recently explored, and it was reported that the performance of deep learning models that were trained on segmented images corresponding to the diseased regions on leaves was better as compared to when the entire leaf images were used [15]. Furthermore, a segmentation approach helped models learn relevant disease features within images, thus making them applicable for testing under field conditions. Recently, OTSU threshold color segmentation was used in fuzzy inference systems for identifying diseased regions with respect to the leaf area to calculate the percentage of infection [16]. However, OTSU thresholding in Fiji software was inefficient when working with large datasets as the hue, saturation, and brightness (HsB) settings in each image needed to be manually modified until the desired area turned black [17].

Unlike OTSU thresholding, Simple Linear Iterative Clustering (SLIC) segmentation is a fast and automatic approach with linear complexity, commonly used to split images into multiple super-pixels with nonuniform boundaries. SLIC segmentation was used for plant disease identification [18]. In addition, SLIC segmentation was recently used to create super-pixels from Unmanned Aerial System (UAS) acquired images for training deep learning models to identify soybean leaf diseases with accuracies of up to 99.04% [19]. The same approach was also used for identifying pests in soybean; however, lower results were reported, partly due to dataset imbalance [12]. SLIC segmentation and deep learning were also used to identify diseased regions in corn from UAS imagery [20]. Although SLIC segmentation was used to create super-pixels for disease and pest identification, its application in agriculture is limited, and its use for identifying multiple foliar diseases in corn was not explored. Furthermore, published studies that utilized SLIC segmentation in agriculture did not report using different segments, sigma values, and training: testing dataset split ratios when training deep learning models, which would be useful for other researchers.

Besides building and training efficient deep learning models, developing and deploying the models on the web and mobile platforms is important for researchers and farmers to identify diseases in real time. A popular disease severity estimation application, Leaf Doctor, was developed to quantify disease severity using color threshold segmentation to calculate the percentage of infection on leaves [21]. A web application was also created for disease identification using deep learning. However, the tool could only identify whether the plant was diseased or not [22]. Although web applications are useful for analyzing diseases after acquiring images, they cannot identify diseases in real time in fields. Recently, a mobile application capable of identifying diseases in various crops was created by training deep learning models using the PlantVillage dataset [23].

This study aims to provide a framework for corn farmers and researchers to identify diseased regions on corn leaves and diagnose multiple diseases that may infect individual leaves. SLIC segmentation and deep learning were used to train multiple deep learning models for identifying diseased regions on leaves. SLIC segmentation was first used to create super-pixels from corn disease images in the PlantVillage and the Corn Disease and Severity (CD&S) datasets. The super-pixels were then used to train deep learning models for each dataset. The performance of five different pre-trained models, namely VGG16, ResNet50, DenseNet121, Xception,

and InceptionV3, was compared using different segments per image, sigma values, and training: testing data split ratios for the segmented datasets. After obtaining the best deep learning configuration, a web and mobile application were developed to identify diseased regions on corn leaf images in real time.

The remaining paper was organized into four main sections. Information regarding the dataset, SLIC segmentation, and deep learning are discussed in the ''Methodology'' section. Key findings were reported and analyzed in the ''Results'' section. More insights about the results and comparison with findings in the literature are shown in the ''Discussion'' section. Finally, the key learning points and future directions are provided in the ''Conclusions'' section.

II. METHODOLOGY

A. DATASETS

Corn disease images from two different datasets were used for the interest of this study: PlantVillage and CD&S. The corn disease images that are present in the PlantVillage dataset consist of uniform backgrounds, whereas the images within the CD&S dataset consist of complex backgrounds with field conditions. Both datasets were recently discussed [24].

The PlantVillage dataset is a free and open dataset consisting of over 54,000 images of 26 diseases from 14 crop species [25]. PlantVillage is the second most used dataset, after custom datasets, in studies that used deep learning applications in agriculture [26]. A subset of the PlantVillage dataset consisting of corn diseases was used for the interest of this study. The corn leaf dataset comprises of four classes, representing three diseases: northern leaf blight (NLB), gray leaf spot (GLS), common rust, and healthy. For each class, 300 images were selected and segmented into super-pixels using SLIC segmentation.

The Corn Disease and Severity (CD&S) dataset is a field-acquired dataset collected by Ahmad et al. at Purdue University's Agronomy Center for Research and Education (ACRE) in West Lafayette, Indiana, in July 2020 [27]. The dataset consists of 4455 images, of which 2112 correspond to three corn diseases: NLB, GLS, and northern leaf spot (NLS). Similar to Plant Village, for each corn disease class in CD&S, 300 images were selected for SLIC segmentation to maintain consistency. The images from the CD&S dataset selected for this study were captured under varying lighting and complex background conditions consisting of grass, soil, weeds, and dead plants.

B. SLIC SEGMENTATION

SLIC segmentation creates super-pixels by grouping different pixels in images based on the similarity in color and their position on an image plane. The approximate size for each super-pixel created is N/K pixels, assuming each image has N pixels and is split into K super-pixels. Hence, supposing each super-pixel is roughly a square, each side of a super-pixel would be calculated using equation 1.

$$S = \sqrt{\frac{N}{K}} \tag{1}$$

Each pixel in the image is represented in the five-dimensional space [labxy], in which l, a, b are values of color vector in CIELAB space, while x and y are coordinates of the pixel on the plane.

After calculating the value of S and converting image pixels to 5D space, centers $C_k = [l_k, a_k, b_k, x_k, y_k]^T$ were initialized by making a grid of points which were spaced by S. Centers were then moved to their lowest gradient position to prevent having noisy pixels as centers. The gradient at a specific position (x, y) was calculated as shown in equation 2.

$$G(x, y) = ||I(x+1, y) - I(x-1, y)||^2 + ||I(x, y+1) - I(x, y-1)||^2 \tag{2}$$

where I(x, y) is the lab color vector of the pixel at position (x, y). The best matching pixel within a 2S × 2S square neighborhood for each modified center was assigned based on a distance formula. The normal Euclidian distance formula cannot be applied because, in the CIELAB space, the difference in lab color value of pixels is constrained below a specific value. In contrast, the difference in the position of pixels could be large if we want to create a big super-pixel. A new way to calculate pixel distance in CIELAB space was proposed that normalizes the position distance by dividing it by the super-pixel size.

$$d_{lab} = \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2} \tag{3}$$

$$d_{xy} = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2} \tag{4}$$

$$D_s = d_{lab} + \frac{m}{S} d_{xy} \tag{5}$$

The distance between two points is the sum of lab and xy distance, where m is the spatial proximity and determines how compact the cluster is. New centers and residual errors will be recalculated until the error is less than a particular threshold [18].

In this study, the SLIC algorithm was imported from the skimage package in Python, consisting of two main parameters: the number of segments (K) and the spatial proximity value (m). By modifying the value of m, the shapes of the super-pixels were modified. Therefore, five different values of m were tested (figure 1), and it was observed that by increasing the value of m, super-pixels could not accurately create boundaries between the leaves and the background. Although setting the value of m to 1 resulted in accurate segments, the boundaries were more irregular. Hence, the value of m was set to 5 to conduct the experiments for disease region identification on corn leaves.

Furthermore, since the PlantVillage dataset consists of images of size 256 × 256 pixels, the number of split segments should not be large. In this study, each image obtained from

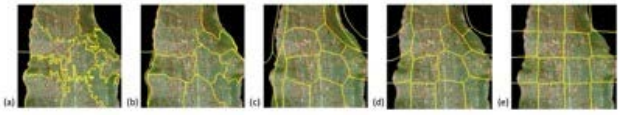

FIGURE 1. SLIC Segmentation with K = 15 and (a) m = 1, (b) m = 5, (c) m = 25, (d) m = 50, and (e) m = 100.

the PlantVillage dataset was split into either five or 15 segments to examine the effect of using a different number of segments in SLIC.

For each corn disease class in the PlantVillage dataset, 300 images were obtained and subjected to SLIC segmentation to create and save super-pixels into five classes: NLB, GLS, common rust, healthy, and background, for training and testing deep learning-based image classification models. For SLIC segmentation using K = 5 and m = 5, 3650 super-pixels were created, with 730 super-pixels per class. When K = 15 and m = 5, a total of 4250 super-pixels were created, with 850 super-pixels per class. The number of super-pixels was kept constant for each class to take advantage of balanced datasets when training deep learning models. A summary of the dataset prepared using the PlantVillage dataset is shown in Table 1.

TABLE 1. PlantVillage dataset after SLIC segmentation was used to create super-pixels.

Class	#Images (K = 5)	#Images (K = 15)
NLB	730	850
GLS	730	850
Common Rust	730	850
Healthy	730	850
Background	730	850

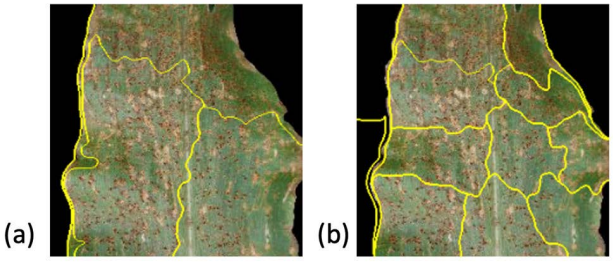

FIGURE 2. Chosen pairs of K and m for SLIC (a) (K = 5, m = 5) and (b) (K = 15, m = 5).

On the other hand, the size of the original images in the CD&S dataset is 3000 × 3000 pixels, which is too large for the SLIC algorithm with a small number of segments (e.g., 5 or 15) to produce accurate and proper segmentation. There are multiple ways to address this issue, including resizing the images before segmenting them, using a larger value of K in SLIC algorithm, or dividing the large image into smaller areas to apply SLIC to those areas. Therefore, to ensure a fair comparison between the deep learning models that were trained using the two datasets, images within the CD&S dataset were resized to 256 × 256 pixels, the same as those in PlantVillage. SLIC segmentation was then applied with K = 5, m = 5, and K = 15, m = 5 (figure 2). For each of the three corn disease classes, 300 images were selected and subjected to SLIC to output super-pixels. An additional background class was created, along with three disease classes in the beginning. For K = 5, m = 5 segmentation, there are 1120 super-pixels in total, with 280 super-pixels per class. For K = 15, m = 5, there are 3200 super-pixels in total, with 800 super-pixels per class. A summary of the dataset that was prepared after using SLIC segmentation to create super-pixels for the CD&S dataset is shown in Table 2.

TABLE 2. CD&S dataset after SLIC segmentation was used to create super-pixels.

Class	#Images (K = 5)	#Images (K = 15)
NLB	280	800
GLS	280	800
NLS	280	800
Background	280	800

In addition to testing different pairs of K and m values for SLIC segmentation, five different training: testing dataset split ratios were also analyzed: 50:50, 60:40, 70:30, 80:20, and 90:10. After splitting the data using the different training: testing ratios, 30% of the training data was used for validation.

C. DEEP LEARNING MODELS

Deep learning approaches, including image classification, object detection, and segmentation, have recently become popular for identifying different objects within images. Deep learning models use CNNs to extract important features from images automatically. CNNs were developed from ANNs using a 3D volume of neurons (convolutional layers) with weights and biases instead of a simple set of neurons, which is more suitable for training models using imagery datasets. Most CNNs also consist of pooling layers and fully connected layers. Pooling layers are used after one or multiple convolutional layers to help reduce the width and height dimensions while maintaining the depth dimension. Fully connected layers are then used to produce the classification output. Activation functions are also an important part of CNNs. Common activation functions include ReLU, Sigmoid, and Tanh.

The first CNN consisted of 5 layers [28]. Recently, from the ImageNet competition, new, giant, and efficient deep learning architectures were introduced and applied in many research areas, including agriculture. In this study, the VGG16 [29], ResNet50 [30], DenseNet121 [31], Xception [32], and InceptionV3 [33], pre-trained deep learning models were imported from the Keras deep learning framework which runs on Tensorflow. Transfer learning was then implemented using the ImageNet pre-trained weights and replacing the final layers in each model with a global pooling layer and a dense layer. For the final dense layer, the softmax activation function was used.

A single NVIDIA Tesla V100 GPU from the Gilbreth community cluster of Purdue's Rosen Center for Advanced Computing (RCAC) was used for deep learning model training. Prior to training the deep learning models, the super-pixel images were resized to 224 × 224 pixels. The training set was augmented by rescaling, rotating, shifting (horizontally and vertically), zooming, and flipping, the images using Keras to avoid saturation and overfitting. Each deep learning model was then trained for 100 epochs using a batch size of 16. In addition, the Adam optimizer was used with a learning rate of 0.0001. For the loss function, Categorical Cross Entropy loss was used. The hyperparameters used for deep learning model training are shown in Table 3. Once training was complete, the training and validation loss and accuracy plots were generated.

TABLE 3. Training hyperparameter details.

HYPERPARAMETERS	Values
Image Size	224 x 224
Batch Size	16
Number of Epochs	100
Loss Function	Categorical Cross Entropy
Optimizer	Adam
Learning Rate	0.0001

A total of 100 deep learning-based image classification models were trained and compared, using two pairs of K and m in SLIC segmentation, five different training: testing split ratios, and five different pre-trained models. The general name would be "values of K and m: training ratio: deep learning model" to represent a configuration. For example, "K5m5:90:VGG16" is the VGG16 model trained on images segmented by SLIC with K = 5, m = 5, and a training: testing split with a ratio of 90:10.

D. EVALUATION METRICS

Accuracy is a widely used evaluation metric that is frequently used for evaluating deep learning-based image classification models when balanced datasets are used. After the 100 deep learning models were trained, two evaluation metrics were used: namely testing accuracies and confusion matrices. After passing the hidden testing images within the dataset into the trained models, testing accuracy was calculated. The true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) were used to calculate the accuracy (equation 6).

In addition, confusion matrices were also created from the hidden testing images. Confusion matrices can provide details regarding each class's correctly and incorrectly classified images. Related metrics from the confusion matrix, including precision, recall, F1-score, and F2-score, could be calculated for each disease class. Hence, a confusion matrix gives a more objective and general evaluation of models.

$$Accuracy = \frac{TP + FN}{TP + FP + TN + FN} \tag{6}$$

The confusion matrices were also created for each model to evaluate the number of correctly and incorrectly classified testing images.

E. CREATION OF WEB APPLICATION

After training and evaluating deep learning models capable of identifying super-pixels corresponding to corn diseases, the best-performing model was deployed onto a web application that was built to provide users with an image-based disease identification tool. The web application was developed using Python/Flask backend and pure JavaScript/HTML/CSS. The development of the application was composed of three primary modules: namely, the creation of the frontend/user interface (UI), the backend, and the SLIC segmentation and deep learning-based corn disease identification. A diagram was created showing how three modules interact with each other (figure 3). The diagram shows that the frontend allows users to upload or directly acquire images that are then sent to the backend through endpoints. The "SLIC and Deep Learning" module is embedded into the backend, where uploaded or acquired images are preprocessed. The deep learning model then classifies the image and passes it to the frontend to display the results. The details for each module are as follows:

Frontend/UI: Initially, the screen displays a "Choose an image" button to prompt users to upload a corn leaf image. After the image is uploaded, it is resized and shown to the users. Then, the "Predict" button appears below the leaf image, which prompts the user to perform the SLIC segmentation and image classification. The frontend sends an HTTP POST request and the uploaded image to the backend. The super-pixels are created and saved in a local project folder for demonstrating the results later. Each generated super-pixel is processed and used as the input for the best deep learning model to perform image classification. A dictionary containing super-pixels and their corresponding classification results is sent from the backend through the "Predict" endpoint as an HTTP response for the frontend. Each pair of "super-pixel: classification results" in the dictionary and how the image is segmented are displayed from top to bottom in the order of segmentation. HTTP request and response through endpoints is how the frontend and backend of a web/mobile application can interact with each other (send or display data).

Backend: The Flask web framework is used to create API endpoints to interact with frontend components such as buttons or text forms. The root "/" endpoint is created to show the main html page. Meanwhile, the "/predict" endpoint obtains the uploaded image via an HTTP POST request, then gets the prediction list from the "SLIC Segmentation and Deep Learning" module to construct the json dictionary object and send it to the frontend.

SLIC Segmentation and Deep Learning: This third module gets the uploaded image and performs SLIC segmentation to create super-pixels. Each super-pixel is masked and saved to show its location on the original image when displayed in the frontend. Data augmentation, such as resizing to 224 × 224 pixels, is used for the super-pixels before

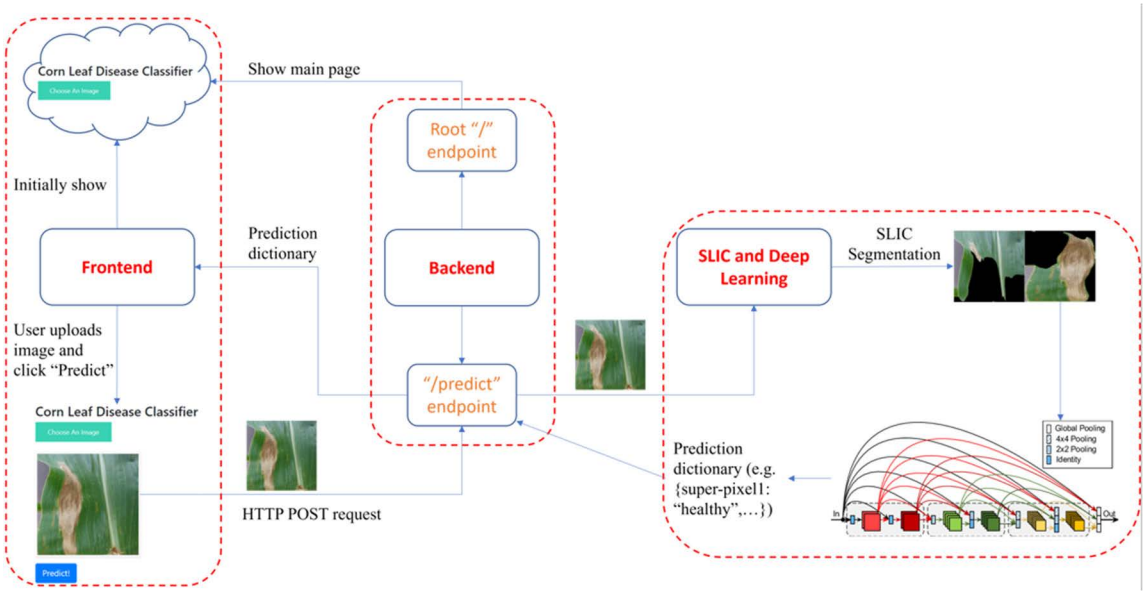

FIGURE 3. Diagram of how modules interact with each other.

providing them as input for the trained model. The best deep learning model, which was previously trained and saved as a .h5 file, is loaded, performs image classification on the super-pixels, and returns a prediction list to the backend.

F. CREATION OF MOBILE APPLICATION

Although a web application provides a tool for users when identifying corn disease regions on leaves, such a tool is difficult to use in real time. Therefore, a mobile application was also created, which allows users to make the same predictions as in the case of the web application, with the added benefit of using the built-in smartphone camera for real time analysis. Once again, the development of the mobile application relied on the same three primary modules: the frontend/user interface (UI), the backend, and the SLIC segmentation and deep learning-based corn disease identification. Hence, the diagram in figure 3 also applies to the Mobile Application with similar interaction between modules.

React Native/Expo was used for frontend development instead of JavaScript/HTML/CSS in the case of the mobile application. React Native was created by Facebook and has become one of the most used mobile application frameworks. The code had to be modified to run the React Native app on different operating systems, such as Android or iOS. Hence, Expo was used in this study to solve the aforementioned difficulty of React Native because it supports building native apps that can run on both Android and iOS with only one codebase. For the mobile application, besides uploading photos, an additional feature of directly taking photos through the smartphone's camera was added. This feature would be helpful for users who want to test the application for identifying corn diseases in real time.

III. RESULTS

A. CLASSIFICATION RESULTS – PLANTVILLAGE DATASET

The overall testing accuracies of all the deep learning models trained on the PlantVillage super-pixels created using $K = 5$, $m = 5$, and $K = 15$, $m = 5$ are shown in figure 4 and figure 5, respectively. It was observed that using $K = 5$ yielded 2% to 5% higher testing accuracies than when using $K = 15$. Because the corn disease images within the PlantVillage dataset were 256×256 pixels, using a larger number of segments ($K = 15$) resulted in the creation of comparatively smaller super-pixels than when using fewer segments ($K = 5$). As the super-pixels were resized to 224×224 pixels for training deep learning models, the image quality for $K = 15$ was reduced.

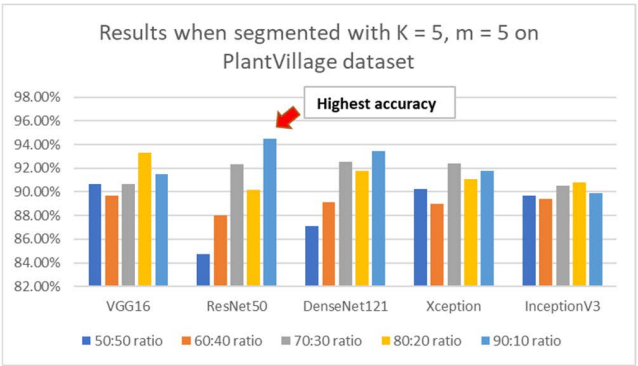

FIGURE 4. Results of all deep learning configurations on PlantVillage dataset when $K = 5$, $m = 5$.

The ResNet50 model achieved the highest overall testing accuracy of 94.52% when it was trained on the super-pixel

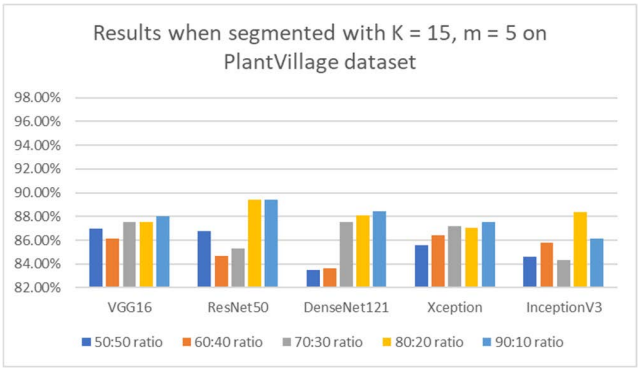

FIGURE 5. Results of all deep learning configurations on PlantVillage dataset when K = 15, m = 5.

classes that were created using K = 5, m = 5 for SLIC segmentation, and a 90:10 training: testing split ratio. In addition, when comparing different training: testing split ratios, the 70:30, 80:20, and 90:10 split ratios generally resulted in higher accuracies compared to 50:50 and 60:40 split ratios. This result was consistent with the findings reported in the literature, where using a higher percentage of training data yielded better testing accuracies [11].

Furthermore, when comparing different split ratios, the differences observed in accuracies of VGG16, Xception, and InceptionV3, were minimal, from which VGG16 performed the best. For example, similar testing accuracies were achieved for VGG16 when the split ratio was 90:10 or 70:30. On the other hand, although the ResNet50 and DenseNet121 models achieved impressive results when the split ratio was 90:10 or 70:30 (92-94%), poor performances were observed with the other split ratios.

A confusion matrix was also generated, as shown in Table 4. From the confusion matrix, it was observed that the model sometimes incorrectly classified the common rust disease as NLB and vice versa. This was due to the presence of both these diseases in a few images, as shown in figure 6. Furthermore, when the value of K was small during SLIC segmentation, some of the created super-pixels consisted of both common rust and NLB diseases. On the other hand, background, healthy, and GLS images were classified almost perfectly.

TABLE 4. Confusion matrix for Diseased Region Location using ResNet50 with 90:10 split ratio.

Class	GLS	NLB	Common Rust	Healthy	Background
GLS	73	0	0	0	0
NLB	0	66	4	3	0
Common Rust	1	8	63	1	0
Healthy	0	1	1	71	0
Background	0	1	0	0	72

B. CLASSIFICATION RESULTS – CD&S DATASET

The overall testing accuracies of all deep learning models trained on super-pixels created using SLIC segmentation with

FIGURE 6. Image of a leaf having both NLB and common rust diseases.

K = 5, m = 5, and K = 15, m = 5 for the CD&S dataset are shown in figure 7 and figure 8, respectively. Overall, the results obtained using the CD&S dataset were observed to be better than when the PlantVillage dataset was used. This can be explained by the fact that the image quality in the CD&S dataset is better than in the PlantVillage dataset. Therefore, when super-pixels were created, it was easier for the models

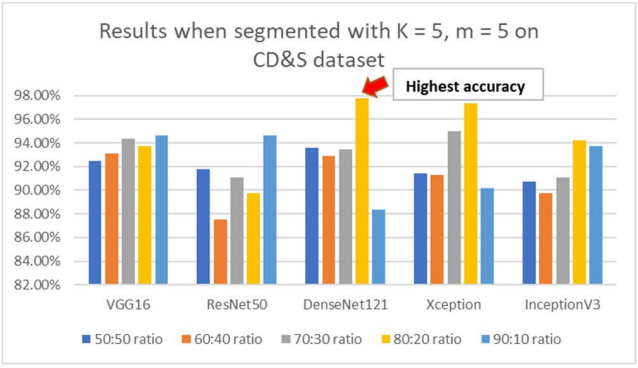

FIGURE 7. Results of all deep learning configurations on CD&S dataset when K = 5, m = 5.

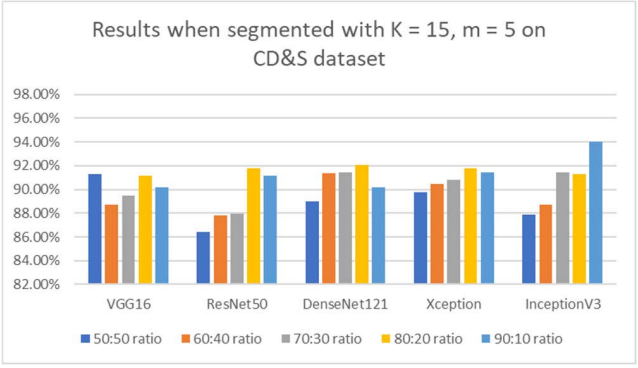

FIGURE 8. Results of all deep learning configurations on CD&S dataset when K = 15, m = 5.

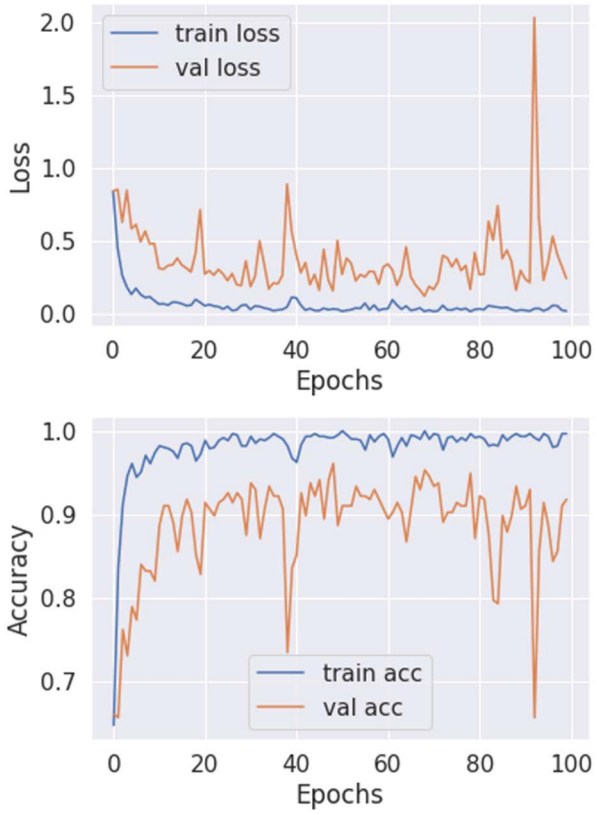

FIGURE 9. K5m5:80:DenseNet121 validation losses and accuracy plots for the CD&S dataset.

to distinguish between different diseases. In addition, most of the images in the CD&S dataset were single-diseased leaves instead of multi-diseased leaves, which reduced the misclassification of the model. As discussed in Section 3.1, using the smaller number of segments (K = 5) in SLIC segmentation also yielded higher accuracies than when larger segments (K = 15) were used.

The highest testing accuracy of 97.77% for the CD&S dataset was obtained by DenseNet121 trained on super-pixels created using K = 5, m = 5 in SLIC segmentation with 80:20 training: testing split ratio. For K = 5, m = 5, while DenseNet121, Xception, and InceptionV3 showed the best results when using the split ratio of 80:20, VGG16 and ResNet50 obtained the highest accuracy when the split ratio was 90:10. Out of the five models, only VGG16 had stable performance across different split ratios without any testing accuracy less than 90%.

For K = 15, m = 5, all models showed a similar trend across different training: testing split ratios. The accuracy increased when the training: testing split ratio for training increased from 50% to 80%. However, the performance with a split ratio of 90:10 was worse. Overall, DenseNet121 and Xception obtained better results compared to other models.

The training and validation accuracy and loss plots are shown in figure 9 for the best model, K5m5:80:DenseNet121. The model, K5m5:80:DenseNet121, performed poorly on the validation set in the initial epochs, but then the training stabilized after the 20th epoch. The confusion matrix for the best model, K5m5:80:DenseNet121, is shown in Table 5. The classification result for each class was almost perfect. Few instances of misclassification were observed, which can be attributed to the presence of additional diseased leaves in the background (e.g., NLS, GLS). Although the background image was blurry, it was independent of the focused parts of the image when segmented into super-pixels. Hence, the model struggled to accurately classify a background image when diseased leaves were present in a blurry context. An example of images with a background consisting of additional diseased leaves is shown in figure 10.

FIGURE 10. Example of NLB diseased corn leaf image having additional diseased leaves in the background.

TABLE 5. Confusion matrix for Diseased Region Location using DenseNet121 with 90:10 split ratio.

Class	NLB	GLS	NLS	Background
NLB	55	1	0	0
GLS	0	54	2	0
NLS	0	0	56	0
Background	1	1	1	53

C. WEB AND MOBILE APPLICATION

After the deep learning models were all trained and compared, the best models for the PlantVillage and CD&S datasets were identified and saved for the web and mobile applications. Once the user interface was prepared, the models were deployed to classify diseased regions on corn leaves. The K5m5:90:ResNet50, ResNet50 model that was trained on the super-pixels created using K = 5, m = 5, and a 90:10 split ratio with the PlantVillage dataset was first deployed. Additionally, K5m5:80:DenseNet121, the DenseNet121 model that was trained on the super-pixels created with K=5, m=5, and 80:20 split ratio with the CD&S dataset, was also deployed.

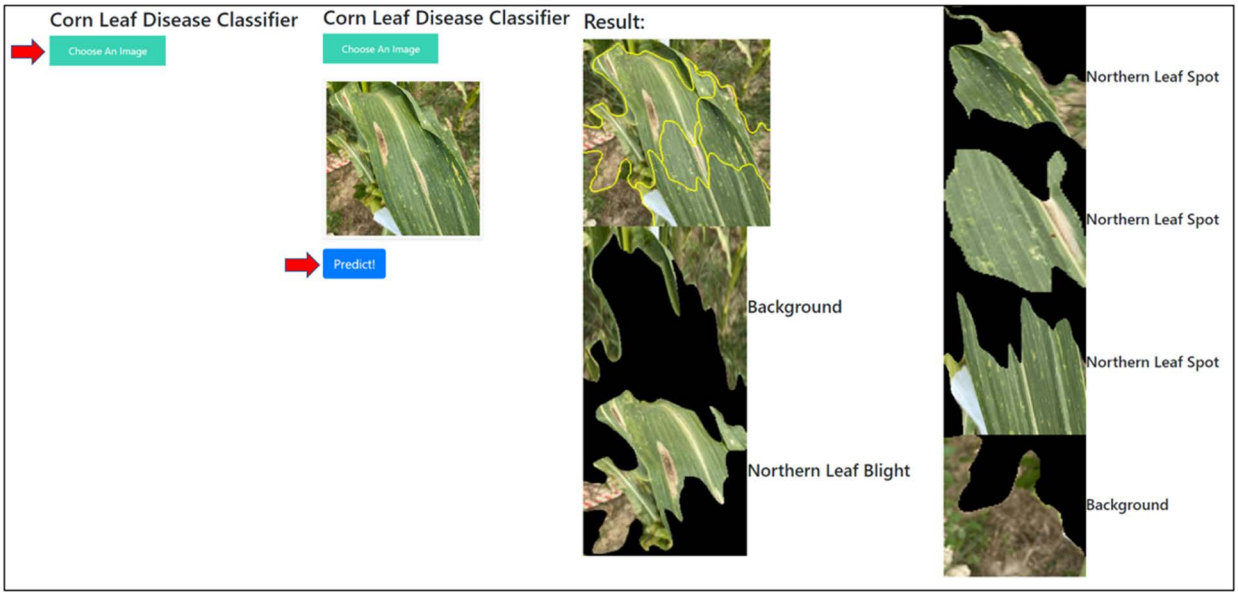

FIGURE 11. Web application plant disease SLIC segmentation and disease region identification.

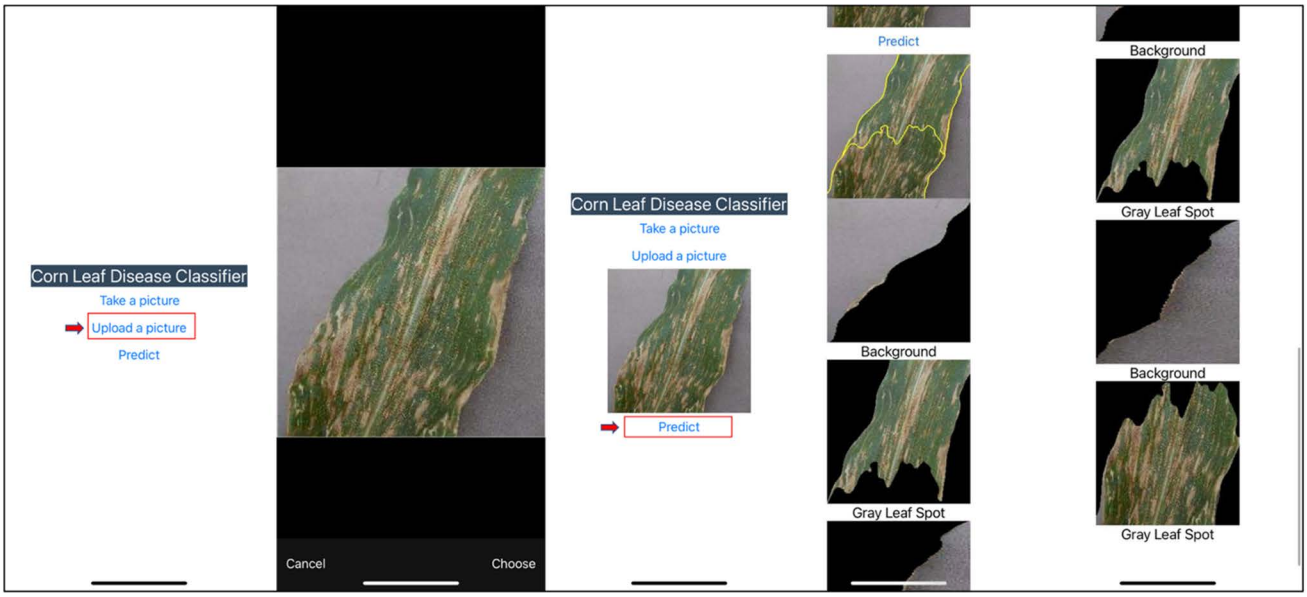

FIGURE 12. Mobile application plant disease SLIC segmentation and disease region identification.

Images were uploaded by clicking the ''Choose an image'' button for the web application, as shown in figure 11. For the mobile application, the users could click either the ''Choose an image'' button or the ''Take a picture'' button, as shown in figure 12. When the user clicks ''Choose an image'', they are allowed to select an image from the smartphone gallery for analysis which provides similar functionality as the web application. However, when the user clicks the ''Take a picture'' button, the camera application on the user's smartphone can acquire an image in real time.

Once the image was uploaded, the ''Predict'' button was clicked, creating and classifying the super-pixels to identify different diseased regions on corn leaves. The result part of the application initially displayed how SLIC segmentation performed on uploaded or acquired images based on the yellow boundaries. Then for each super-pixel, its image and classification results were shown accordingly. As shown in figure 11, it was observed that if the leaf image consisted of multiple diseases (e.g., NLB and Common rust), SLIC segmentation was able to separate them, and the model still classified them correctly.

IV. DISCUSSION

This study was conducted to identify diseased regions on corn leaves using SLIC segmentation and deep learning. Generally, plants' foliar diseases can manifest simultaneously, resulting in lesions corresponding to multiple diseases on individual leaves. Although the manifestation of multiple simultaneous disorders on individual plants was identified as a research gap [34], most current studies within the literature have identified single plant disease per leaf image using deep learning-based disease identification.

This study used the PlantVillage and CD&S datasets to assess the performance of SLIC segmentation and deep learning-based approach under different conditions. The images within the PlantVillage dataset were acquired in lab conditions and consisted of uniform color backgrounds. On the other hand, the images within the CD&S dataset consisted of complex backgrounds as they were acquired in field conditions. SLIC segmentation was used to create super-pixels for images obtained from both datasets using different pairs of parameters. All deep learning models were then trained using the segmented data using various training: testing split ratios.

Studies within the literature have used PlantVillage dataset for training deep learning models to identify diseases with high accuracies of greater than 90%, however, diseased regions on individual leaves were not identified [7], [8], [9]. In addition, SLIC segmentation was used to identify soybean leaf diseases in UAS imagery with accuracies up to 99.04% [19]. However, the UAS images could not acquire details corresponding to disease lesions due to high altitude and low spatial resolution. Our study demonstrates the use of SLIC segmentation and deep learning to identify diseased regions corresponding to multiple diseases on individual corn leaves with accuracies of up 97.77%. In addition, this study also explored the comparison of the combined approach under controlled lab conditions, complex field conditions, and with different training: testing split ratios for developing a framework for disease diagnosis.

Deployment of deep learning models via web and mobile applications is important to provide users with an interactive disease diagnosis tool. Although applications such as Leaf Doctor are capable of disease identification, they require images with uniform backgrounds acquired under controlled lab conditions [21]. Thus, their use under field conditions is limited. The best deep learning models from this study were deployed to develop a web and mobile-based tool to help diagnose multiple simultaneous diseases accurately. Overall, our study shows the benefits of SLIC segmentation and deep learning for identifying multiple simultaneous diseases on individual leaves.

V. CONCLUSION

This study focused on using SLIC segmentation to create super-pixels for training deep learning-based image classification models to identify diseased regions on corn leaf images obtained from the PlantVillage and CD&S datasets. After subjecting the images from both datasets to SLIC segmentation using K = 5, m = 5, and K = 15, m = 5, super-pixels were created to train deep learning models for identifying multiple disease classes within each dataset. The performances of five different pre-trained deep learning models for corn disease region identification were compared using two different SLIC parameters and five different training: testing split ratios. Overall, deep learning models trained on super-pixels created using K = 5, m = 5 yielded higher testing accuracies than when K = 15, m = 5 was used. For the PlantVillage dataset, the ResNet50 model achieved the highest testing accuracy of 94.52% when it was trained on the super-pixels that were created using K = 5, m = 5, and a training: testing split ratio of 90:10. Although this high testing accuracy was achieved with the PlantVillage dataset, deep learning models had difficulty in accurately identifying multiple diseases present on testing images. For the CD&S dataset, the highest testing accuracy of 97.77% was obtained by the DenseNet121 model trained on segments created using K = 5, m = 5, and the training: testing split ratio of 80:20. The results demonstrate that using SLIC segmentation and deep learning helped identify the presence of multiple disease regions on individual leaves under field conditions. After training and comparing the deep learning models, a trained model was deployed onto a web application and a mobile application to help users identify corn disease regions in real time. In the future, images of larger sizes will be used to improve the identification of multiple diseases per image.

CONFLICT OF INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this article.

REFERENCES

[1] D. H. Vo, "Sustainable agriculture & energy in the U.S.: A link between ethanol production and the acreage for corn," *Econ. Sociol.*, vol. 13, no. 3, pp. 259–268, Sep. 2020.
[2] D. S. Mueller, K. A. Wise, A. J. Sisson, T. W. Allen, G. C. Bergstrom, K. M. Bissonnette, C. A. Bradley, E. Byamukama, M. I. Chilvers, A. A. Collins, and P. D. Esker, "Corn yield loss estimates due to diseases in the United States and Ontario, Canada, from 2016 to 2019," *Plant Health Prog.*, vol. 21, pp. 238–247, Jan. 2020, doi: 10.1094/PHP-05-20-0038-RS.
[3] Y. Fang and R. P. Ramasamy, "Current and prospective methods for plant disease detection," *Biosensors*, vol. 5, no. 3, pp. 537–561, 2015, doi: 10.3390/bios5030537.
[4] F. Martinelli, R. Scalenghe, S. Davino, S. Panno, G. Scuderi, P. Ruisi, P. Villa, D. Stroppiana, M. Boschetti, L. R. Goulart, and C. E. Davis, "Advanced methods of plant disease detection. A review," in *Agronomy for Sustainable Development*, vol. 35, no. 1. New York, NY, USA: Springer-Verlag, Jan. 2015, pp. 1–25, doi: 10.1007/s13593-014-0246-1.
[5] T. N. Tete and S. Kamlu, "Detection of plant disease using threshold, K-mean cluster and ANN algorithm," in *Proc. 2nd Int. Conf. Converg. Technol. (I CT)*, Apr. 2017, pp. 523–526.
[6] W. P. Amorim, A. X. Falcão, J. P. Papa, and M. H. Carvalho, "Improving semi-supervised learning through optimum connectivity," *Pattern Recognit.*, vol. 60, pp. 72–85, Dec. 2016, doi: 10.1016/j.patcog.2016.04.020.
[7] M. Agarwal, V. K. Bohat, M. D. Ansari, A. Sinha, S. K. Gupta, and D. Garg, "A convolution neural network based approach to detect the disease in corn crop," in *Proc. IEEE 9th Int. Conf. Adv. Comput. (IACC)*, Dec. 2019, pp. 176–181.
[8] A. Waheed, M. Goyal, D. Gupta, A. Khanna, A. E. Hassanien, and H. M. Pandey, "An optimized dense convolutional neural network model for disease recognition and classification in corn leaf," *Comput. Electron. Agricult.*, vol. 175, Aug. 2020, Art. no. 105456.

[9] H. Amin, A. Darwish, A. E. Hassanien, and M. Soliman, "End-to-end deep learning model for corn leaf disease classification," *IEEE Access*, vol. 10, pp. 31103–31115, 2022.
[10] E. C. Too, L. Yujian, S. Njuki, and L. Yingchun, "A comparative study of fine-tuning deep learning models for plant disease identification," *Comput. Electron. Agric.*, vol. 161, pp. 272–279, Jun. 2019, doi: 10.1016/j.compag.2018.03.032.
[11] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers Plant Sci.*, vol. 7, p. 1419, Sep. 2016, doi: 10.3389/fpls.2016.01419.
[12] E. C. Tetila, B. B. Machado, G. Astolfi, N. A. D. S. Belete, W. P. Amorim, A. R. Roel, and H. Pistori, "Detection and classification of soybean pests using deep learning with UAV images," *Comput. Electron. Agricult.*, vol. 179, Dec. 2020, Art. no. 105836, doi: 10.1016/j.compag.2020.105836.
[13] G. Wang, Y. Sun, and J. Wang, "Automatic image-based plant disease severity estimation using deep learning," *Comput. Intell. Neurosci.*, vol. 2017, pp. 1–8, Sep. 2017, doi: 10.1155/2017/2917536.
[14] J. G. M. Esgario, R. A. Krohling, and J. A. Ventura, "Deep learning for classification and severity estimation of coffee leaf biotic stress," *Comput. Electron. Agricult.*, vol. 169, Feb. 2020, Art. no. 105162, doi: 10.1016/j.compag.2019.105162.
[15] J. G. A. Barbedo, "Plant disease identification from individual lesions and spots using deep learning," *Biosyst. Eng.*, vol. 180, pp. 96–107, Apr. 2019, doi: 10.1016/j.biosystemseng.2019.02.002.
[16] R. Nagi and S. S. Tripathy, "Severity estimation of grapevine diseases from leaf images using fuzzy inference system," *Agricult. Res.*, vol. 11, no. 1, pp. 112–122, Mar. 2022, doi: 10.1007/s40003-021-00540-4.
[17] M. Sibiya and M. Sumbwanyambe, "An algorithm for severity estimation of plant leaf diseases by the use of colour threshold image segmentation and fuzzy logic inference: A proposed algorithm to update a 'Leaf Doctor' application," *AgriEngineering*, vol. 1, no. 2, pp. 205–219, May 2019, doi: 10.3390/agriengineering1020015.
[18] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC Superpixels," EPFL, Lausanne, Switzerland, Tech. Rep. 149300, 2010.
[19] E. C. Tetila, B. B. Machado, G. K. Menezes, A. Da Silva Oliveira, M. Alvarez, W. P. Amorim, N. A. De Souza Belete, G. G. Da Silva, and H. Pistori, "Automatic recognition of soybean leaf diseases using UAV images and deep convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 5, pp. 903–907, May 2020, doi: 10.1109/LGRS.2019.2932385.
[20] A. Ahmad, V. Aggarwal, D. Saraswat, A. El Gamal, and G. S. Johal, "GeoDLS: A deep learning-based corn disease tracking and location system using RTK geolocated UAS imagery," *Remote Sens.*, vol. 14, no. 17, p. 4140, Aug. 2022.
[21] S. J. Pethybridge and S. C. Nelson, "Leaf doctor: A new portable application for quantifying plant disease severity," *Plant Disease*, vol. 99, no. 10, pp. 1310–1316, Oct. 2015, doi: 10.1094/PDIS-03-15-0319-RE.
[22] R. B. S., T. A. Shriram, J. S. Raju, M. Hari, B. Santhi, and G. R. Brindha, "Farmer-friendly mobile application for automated leaf disease detection of real-time augmented data set using convolution neural networks," *J. Comput. Sci.*, vol. 16, no. 2, pp. 158–166, Feb. 2020.
[23] A. A. Ahmed and G. H. Reddy, "A mobile-based system for detecting plant leaf diseases using deep learning," *AgriEngineering*, vol. 3, no. 3, pp. 478–493, Jul. 2021, doi: 10.3390/agriengineering3030032.
[24] A. Ahmad, D. Saraswat, and A. El Gamal, "A survey on using deep learning techniques for plant disease diagnosis and recommendations for development of appropriate tools," *Smart Agricult. Technol.*, vol. 3, Feb. 2023, Art. no. 100083, doi: 10.1016/j.atech.2022.100083.
[25] D. P. Hughes and M. Salathe, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," 2015, *arXiv:1511.08060*.
[26] A. Abade, P. A. Ferreira, and F. de Barros Vidal, "Plant diseases recognition on images using convolutional neural networks: A systematic review," *Comput. Electron. Agricult.*, vol. 185, Jun. 2021, Art. no. 106125, doi: 10.1016/j.compag.2021.106125.
[27] A. Ahmad, D. Saraswat, A. El Gamal, and G. Johal, "CD&S dataset: Handheld imagery dataset acquired under field conditions for corn disease identification and severity estimation," 2021, *arXiv:2110.12084*.
[28] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, "Object recognition with gradient-based learning," in *Shape, Contour and Grouping in Computer Vision*. Cham, Switzerland: Springer, 1999, pp. 319–345.
[29] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.
[31] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, (2017). *Densely Connected Convolutional Networks*. [Online]. Available: https://github.com/liuzhuang13/DenseNet
[32] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1800–1807, doi: 10.1109/CVPR.2017.195.
[33] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826, doi: 10.1109/CVPR.2016.308.
[34] J. G. A. Barbedo, "A review on the main challenges in automatic plant disease identification based on visible range images," *Biosyst. Eng.*, vol. 144, pp. 52–60, Apr. 2016.

HIEU PHAN is currently pursuing the bachelor's degree with the Department of Computer Science and Software Engineering, Miami University, Oxford, OH, USA, and a second major in data science and statistics. He is currently a Software Engineer Intern at Microsoft. He has published conference papers about machine learning applications in network science. His research interests include deep learning, computer vision, and adversarial machine learning.

AANIS AHMAD received the B.Sc. degree in computer engineering from Purdue University, West Lafayette, IN, USA, in December 2019, where he is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering. He received the ASABE ITSC Best Paper Award from the 2020 ASABE AIM Conference, the Best Presentation Award from the 2020 ASABE AIM Conference, and the First Place Award from the 2022 ASABE AETC Poster Competition.

DHARMENDRA SARASWAT received the B.Tech. degree in agricultural engineering from the University of Allahabad, Allahabad, India, in 1988, the M.S. degree in agricultural engineering from the Indian Agricultural Research Institute, New Delhi, India, in 1990, and the Ph.D. degree in food, agricultural, and biological engineering from The Ohio State University, Columbus, OH, USA, in 2007. He has received numerous peer recognitions for his novel contributions in research, teaching, and extension.