Deep Learning-Based Plant Disease Image Recognition for Cyber-Physical Systems

Feiyang Bai
Department of Mechanical Engineering
University of the District of Columbia
Washington, D.C., USA
feiyang.bai@udc.edu

Wagdy H. Mahmoud
Department of Electrical and Computer Engineering
University of the District of Columbia
Washington, D.C., USA
wmahmoud@udc.edu

Nian Zhang
Department of Electrical and Computer Engineering
University of the District of Columbia
Washington, D.C., USA
nzhang@udc.edu

2023 13th International Conference on Information Science and Technology (ICIST) | 979-8-3503-1392-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICIST59754.2023.10367187

***Abstract*— Corn, wheat, and rice are vital as staple foods, affecting our economy, culture, health, and environment. Our research focuses on using a convolutional neural network (CNN) to detect food plant diseases. Early and accurate detection is essential to control the spread of infection and maintain the health of the food plant industry, making sustainable food systems a pressing matter. Traditional neural networks can't cope with the weight increase caused by large image sizes and numerous hidden layer neurons. To solve this, we suggest a new method for identifying plant diseases early using CNN. By adjusting the CNN's hyperparameters, we can optimize it for the given dataset. To train the proposed deep CNN model, we use real plant disease datasets such as the PlantVillage dataset** [19], **Wheat Leaf Dataset** [20], **Rice Leaf Disease Data Set** [21], **and Rice Leaf Disease Image Samples** [22]. **We were able to achieve impressive results for wheat (99.34%), corn (95.15%), and rice (92.47%) plant disease detection, with a promising level of accuracy demonstrated in our experimental findings. The results achieved in this paper exceed the accuracy of other related research works (details in literature review).**

Keywords*— *Food plan disease, Machine Learning (ML), Deep Learning (DL), Transfer Learning (TL), Convolution Neural Network (CNN).

I. Introduction

Agriculture is a significant global industry, employing over 1.3 billion people and contributing to 12.5% of the GDP. Yuan Longping's hybrid rice and Norman Ernest Borlaug's high-yield wheat have been pivotal in increasing food production, particularly in areas prone to famine. However, plants are still at risk of disease, mainly from fungi spores, causing substantial economic losses. Wheat stem rust alone results in around $1 billion in losses per year, according to the Food and Agriculture Organization (FAO).

The objective of this project is to detect plant leaf diseases crossing multiple data sets, specifically those that affect food plants. Such a tool would prove invaluable to farmers, allowing them to promptly identify and treat diseases, and achieve successful and healthy plantations.

Traditionally, detecting crop diseases heavily relied on agricultural experts physically visiting fields. This approach, however, was limited by the number of experts available, weather conditions, and farm locations. Fortunately, the advancement of machine learning (ML) has provided a more efficient means of disease detection. With its robust ability to identify mapping relationships among vast input (staple food plant disease images) and output (diseases), ML offers a faster, less costly, and more effective alternative solution to the traditional approach, thereby reducing the burden on agricultural experts.

One of the most effective ML approaches for image recognition is the CNN [1]-[4]. This subtype of Neural Networks contains a convolutional layer that reduces the high dimensionality of images while preserving their important information. By leveraging CNNs, we can effectively detect plant diseases with a high level of accuracy, providing farmers with a reliable and efficient tool for identifying and treating diseases at an early stage, and ultimately ensuring the health and productivity of their crops. That is why CNNs are especially suited for image processing for plant leaf disease identification in agriculture [5].

After the introduction of the PlantVillage dataset in 2015 [6], deep learning using CNN has emerged as a promising area of research for plant disease identification [7]. The effectiveness of this approach has been demonstrated by experimental results, which indicate that the proposed disease identification method based on CNN achieves an impressive overall accuracy of 99.34%, 95.15%, and 92.47% for wheat, corn, and rice plant diseases, respectively. In recent times, several other plant disease datasets, such as the Northern Leaf Blight (NLB) dataset [8], the RoCoLe coffee disease dataset [9], the rice disease dataset [10], and the cassava disease dataset [11], have been made publicly available for training deep learning models.

While there have been several research efforts on ML-based plant disease detection, few have been able to create models that can consistently achieve high testing accuracies across datasets and various image acquisition conditions. Consequently, there remain several unanswered questions, such as whether it is feasible to develop algorithms that are robust enough to effectively learn from any given data.

In light of this, we propose a novel approach to plant leaf disease diagnosis by segmenting leaf images and detecting any spots or other distortions using MATLAB. Next, we will train a CNN to detect various plant species, including those that are diseased or healthy. With this approach, we achieved greater accuracy and robustness in staple plant disease detection, contributing to more efficient and effective agricultural practices.

The major contributions are given as follows:

- Images obtained from various sources were preprocessed to ensure uniformity in size and format, suitable for the input layer of the ML model;
- The proposed ML model's adaptability and robustness were evaluated by utilizing data from diverse sources;
- The training process was fine-tuned with parameters optimized to handle various situations such as overfitting and underfitting;
- To demonstrate the proposed ML model's robustness and universality, it was applied to various datasets with diverse classification classes, rather than being exclusively suited for specific datasets;
- For the first time, the proposed algorithm demonstrated robustness to all three staple food plant datasets (corn, wheat, and rice) with impressive results across different crops like wheat (99.34%), corn (95.15%), and rice (92.47%), unlike most other researchers who only tested one of the staple food datasets [12]– [17] or compared it with other plant data such as tomato, pepper, potato [18].

II. LITERATURE REVIEW

Applying ML to plant leaf disease detection has been studied by many researchers since this application doesn't heavily depend on traditional field experts, which saves time; it could detect plant disease in the early stage to avoid food reduction and increase productivity to stabilize the basic economy. However, the majority of the researchers only studied 1 plant, like tomato, potato, corn, wheat, or rice. The proposed algorithm is suitable for all 3 data sets.

In their study, Yakkundimath R et al. evaluated pre-trained VGG-16 and GoogleNet convolutional neural network (CNN) models on a held-out dataset using a threefold cross-validation method. The results showed an accuracy of 92.24% and 91.28% for rice plant dis-ease, respectively [12]; Panigrahi K et al. investigated several classification techniques, including Naive Bayes (NB), Decision Tree (DT), K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and Random Forest (RF), for detecting maize plant diseases using plant images. The results showed that the RF algorithm achieved the highest accuracy of 79.23% compared to the other classification techniques [13]; Agarwall M, etc. investigated Convolution Neural Network, and achieved an accuracy of 94% for corn crop [14]; Deng R et al. conducted a study on an Ensemble Model for diagnosing six types of rice diseases and achieved an overall accuracy of 91% [15]; Jahan N, etc. did several experiments on wheat datasets. In terms of model accuracies, VGG16 (98%) outperformed the other models and thus it is suggested to utilize VGG16 for the detection of wheat diseases [16]; Jiang Z and colleagues explored the use of a pre-trained ImageNet model for transfer learning, which resulted in an accuracy of 98.75% for detecting wheat diseases [17].

III. DATA PREPARATION

The data in this study is from PlantVillage dataset [19], Wheat Leaf Dataset [20], Rice Leaf Disease Data Set [21], and Rice Leaf Disease Image Samples [22]. Wheat, corn, and rice data for each category are shown below.

One big problem in image detection of the public data set, including plant disease detection is that the images from different data set sources come with different size, which make it impossible to feed the DL model since each model requires identical image sizes to work properly. Without pre-processing the data from different sets, it is impossible to utilize DL to help disease detection.

In this study, we pre-process the dataset using the MATLAB learning APP: Image Batch Processor. After loading the dataset to the app, we use imresize() command to resize all the images in the data set once and for all.

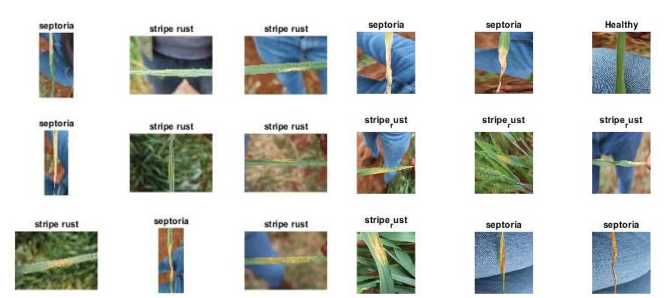
In the pre-trained AlexNet, we use size [227 227 3]. After the image is unified, we can download the image into a unified format, like PNG JPG, or other formats for the learning model. Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit the use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper. Do not number text heads-the template will do that for you.

A. Wheat Data Set

The wheat data set [20] contains 3 classes of data including Healthy, Septoria, and Stripe-rust. The original image size varies (Fig 1. a), and after the preprocessing, we unified all the images into [227 227 3] (Fig 1. b). The table below shows the image numbers of each class, and size of image before and after preprocessing.

TABLE I. WHEAT DATASET

Wheat Dataset from Kaggle [20]	No. of images	The original size of the images	After the pre-processed size of the images
Healthy	102	different size	227*227*3
Septoria	97	different size	227*227*3
Stripe-rust	208	different size	227*227*3
	407 (total)		



a b

Fig. 1. Wheat data: a. original image; b. after pre-processed image.

B. Corn Data Set

The corn data set [19] contains 4 classes of data including Healthy, Common-rust, Gray-leaf-spot and Blight. The original image size varies (Fig 2. a), and after the preprocessing, we unified all the images into [227 227 3] (Fig 2. b). The table below shows the image numbers of each class, and the total number of the data set.

TABLE II. CORN DATASET

Corn Dataset from Kaggle [19]	No. of images	The original size of the images	After the pre-processed size of the images
Healthy	1162	different size	227*227*3
Common-rust	1306	256*256*3	227*227*3
Gray-leaf-spot	574	256*256*3	227*227*3
Blight	1146	256*256*3	227*227*3
	4188 (total)		

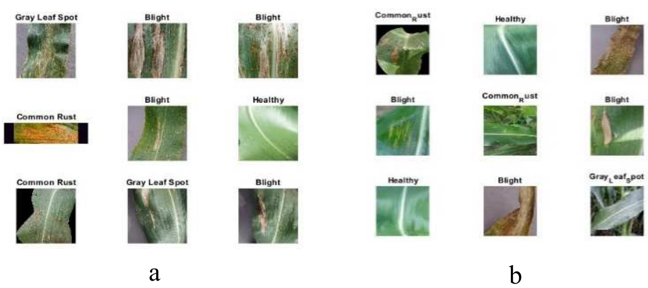


Fig. 2. Corn data: a. original image; b. after pre-processed image.

C. Rice Data Set

The rice data set [21] [22] contains 6 classes of data including Bacterial-blight, Blast, Brown spot, Hisba, Turgro, and Healthy. The original image size varies (Fig 3. a), and after the preprocessing, we unified all the images into [227 227 3] (Fig 3. b). The table below shows the image numbers of each class, and the total number of the data set.

TABLE III. RICE DATASET

Rice Dataset from Mendeley, Kaggle, and UCI [21] [22]	No. of images	The original size of the images	After the pre-processed size of the images
Bacterial-blight	1584	different size	227*227*3
Blast	1440	300*300*3	227*227*3
Brown-spot	1600	300*300*3	227*227*3
Hispa	565	different size	227*227*3
Turgro	1308	different size	227*227*3
Healthy	1488	different size	227*227*3
	7992 (total)		

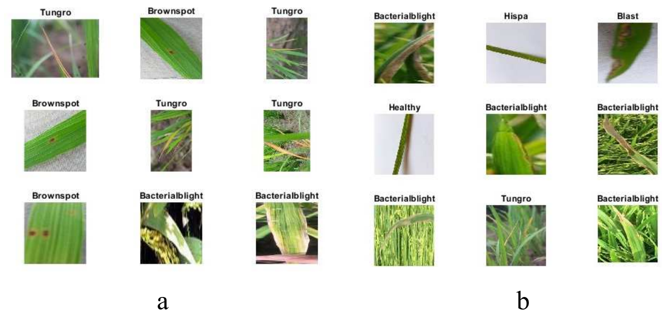


Fig. 3. Rice data: a. original image; b. after pre-processed image.

IV. PROPOSED METHODOLOGY

In this experiment, we used pre-trained AlexNet, which is the first major CNN. AlexNet used GPU for the training process, which will make the training faster than other models. AlexNet, a pre-trained neural network, is capable of classifying images into 1000 object categories, including animals, flowers, cats, hats, and more. Due to this, the network has acquired a robust set of feature representations for a diverse range of images.

By adding a deconvolution layer to the traditional AlexNet and classifying images through the full connection layer, the accuracy of classification has been improved compared to the traditional 8-layer AlexNet. The proposed network is a 30-layer network with a convolution layer, batch normalization layer, polling layer, LeakyRelu layer, drop-off layer, SoftMax layer, and fully connected layer. The hyperparameters are tuned to optimize the optimal performance. (Parameter tuning details are in the experiment section)

A. *Algorithm Architecture*

The new convolutional neural network architecture is shown in Fig. 4.

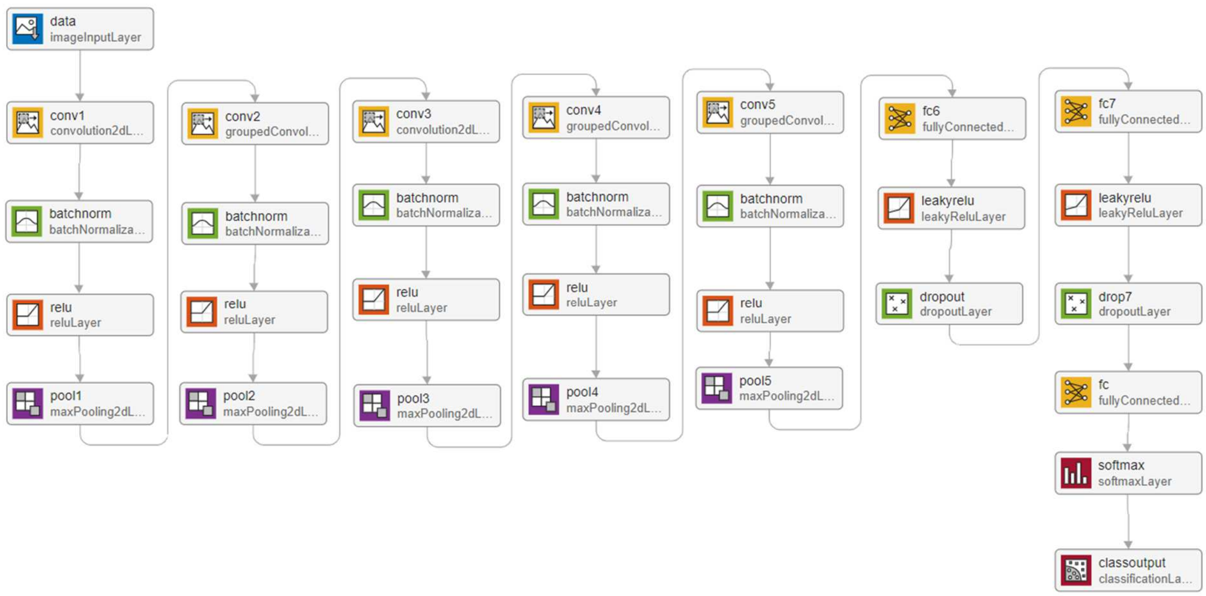


Fig. 4. Proposed Convolutional Neural Network Architecture.

B. *Proposed Algorithm*

The CNN is an algorithm with layer by layer. Images or data were fed into the input layer, and it will pass layer by layer. Features will be extracted throughout layers to train the model to learn the uniqueness of each class, so it could be used to classify new images to the correct class.

In the proposed architecture, the convolution layer is utilized to extract image features like edges, corners, and lines using a filter that moves over the input image and computes the dot product with the corresponding intensity values. The size of the output volume is calculated based on the input volume (V), filter size (F), stride (S), and zero padding amount (P), using the formula (V-F+2P)/S+1. To illustrate, consider the first convolution layer of the architecture, which takes an input volume of 227 x 227 x 3, where 3 represents the number of color channels (R, G, B) in the image. This layer comprises 96 filters of size 11 x 11 x 3, with a stride of 4 x 4, and the same padding (P=1) is applied. By applying the formula, the resulting output dimension is calculated as follows: (227-11+2(1))/4+1 = 56. Therefore, the dimension of the output of the first convolution layer is 56 x 56 x 96.

When training a convolutional neural network, it's recommended to incorporate batch normalization layers between convolutional layers and nonlinearities (like ReLU layers) to speed up the process and minimize the network's sensitivity to initialization. This is achieved by normalizing a mini batch of data across all observations independently for each channel using the batch normalization layer. The network with batch-normalized layer improves the published result, and even exceeds the accuracy of human raters [23].

The ReLU layer applies the ReLU activation function to the input data, and this operation sets any input values that are negative to zero while keeping the positive values unchanged., it helps to improve the network's ability to learn and predict the given data. However, the Leaky ReLU function is a modified version of the ReLU activation function, designed to tackle the issue of the "dying ReLU" problem because the ReLU activation function deactivates neurons in the region where input values are less than zero, due to a gradient of 0, which can lead to the dying ReLU problem. To solve this problem, the Leaky ReLU function is introduced. It differs from the ReLU function in that instead of setting negative input values to 0, it assigns a small linear component of the input value to negative input values. This can be expressed through the following formula. In this experiment, we changed the last 2 ReLU layers to Leaky-ReLU layers, which improved the performance of the algorithm.

ReLU activation function: $f(x)=\begin{cases} x, x > 0 \\ 0, x < 0 \end{cases}$

Leaky-ReLU activation function: $f(x)= \begin{cases} x, & x > 0 \\ scale * x, & x < 0 \end{cases}$

(while default scale is 0.01)

The rectified feature map from the ReLu layer will then go through a pooling layer. Pooling is a down-sampling operation that reduces the dimensionality of the feature map. Max pooling will be used which chooses the maximum value of the filter and then places it in the new pooling feature map. The size of the output of the pooling will be determined as follows. If we assume that the size of the output volume is given by (V-F)/S+1, where V is the input volume, F is the filter size, and S is the stride, then consider the first pooling layer. If the input volume is of size 19 x 19 x 64, and a 3 x 3 max pooling operation with a stride of 2 is applied to it, then the resulting output can be calculated as follows: (19-3)/2+1 = 9. Thus, the output size is 9 x 9 x 64.

After applying convolutional and down-sampling operations one or more fully connected layers are typically added. These layers are called "fully connected" because each neuron in the layer is connected to all the neurons in the preceding layer. The role of the fully connected layer is to combine the features learned by the previous layers across the image, enabling the neural network to identify larger patterns. The final fully connected layer takes the learned features and combines them to classify the images.

During the training of a neural network, overfitting can be a problem that leads to poor performance on new data. To combat this, a technique called "dropout" can be used. The Dropout layer is a type of layer that randomly sets some of the input units to 0 with a certain frequency (called the "rate") at each training step. This helps to prevent the neural network from relying too heavily on any particular input unit or feature, which can lead to overfitting [24].

The activation function used in the output layer is SSoftMax plays an important role for the plant disease classification problem. Each value in the output of the SSoftMax function is interpreted as the probability of membership for each class.

Usually, the final layer in a neural network is the classification layer. The purpose of this layer is to obtain the output probabilities from the SSoftMax activation function, which assigns a probability value to every input for all the mutually exclusive classes. Using these probabilities, the classification layer determines the ultimate class assignment for each input and calculates the corresponding loss.

V. EXPERIMENTAL RESULTS

A. *Experiment with Wheat Data Set*

The wheat data set contains 3 classes of images of corn plant leaves, including Healthy, Septoria, and Stripe-rust. The training and validation data number is shown below.

TABLE IV. WHEAT PLANT DISEASE DATA SET DETAILS TO APPLY PROPOSED ALGORITHM (IMAGE DATA SET AFTER PRE-PROCESSING).

Wheat Dataset from Kaggle	No. of images	Training data (80%)	Validation data (20%)
Healthy	102	82	20
Septoria	97	78	19
Stripe-rust	208	166	42
	407 (total)		

After getting the image processed with the same size and format, we apply the data set in the pre-trained AlexNet. The original AlexNet has 2 layers, including 1 input layer, 5 convolution layers, 6 Relu layers, 2 norm Channel Normalization layers, 3 max-pooling layers, 3 fully connected layers, 2 drop-out layers, 1 SoftMax layer, and 1 output layer. The default set learning rate is 0.01, and L2 regulation is 0.0001, shuffle every epoch, and the drop-off probability rate is 0.5.

During the experiment, we fine-tuned a few parameters, for example changing the layers, adjusting the default learning rate and normalization rate… We finalized the best result by adding a norm Channel Normalization layer behind each convolution layer, updating the last 2 Relu layers to the leaky-Relu layer, setting the learning rate at 0.00015, raising drop off rate to 0.7, dropping the L2 regulation to 0.00025, and increasing max epochs to 60. The fine-tuned network achieved really good results. The best result is that the validation accuracy is 100%. Since every time, the data was shuffled and training data and validation data were randomly chosen for the algorithm, we run 5 times of the algorithm, and got the average result of 99.34%. Multiple time test results and best result plot are shown below.

TABLE V. PROPOSED ALGORITHM ON WHEAT PLANT DISEASE

Test 1	Test 2	Test 3	Test 4	Test 5	Average
99.18%	98.77%	98.77%	100%	100%	99.34%

B. *Experiment with Corn Data Set*

The corn data set contains 4 classes of images of corn plant leaves including Healthy, Common-rust, Gray-leaf-spot, and Blight. The training and validation data number is shown below.

TABLE VI. CORN PLANT DISEASE DATA SET DETAILS TO APPLY PROPOSED ALGORITHM (IMAGE DATA SET AFTER PRE-PROCESSING)

Corn Dataset from Kaggle	No. of images	Training data (80%)	Validation data (20%)
Healthy	1162	930	232
Common-rust	1306	1045	261
Gray-leaf-spot	574	459	115
Blight	1146	917	229
	4188 (total)		

After dividing the dataset, we applied the data to the proposed CNN model. We kept the fine-tuning parameters for the wheat data set, and we achieved good results. The best accuracy result is 95.58%. Same as the wheat data set, since the data was chosen randomly to train the algorithm, so result varies each time you run the algorithm. So, we did 5 times of the experiment and got an average accuracy result of 99.34%. The test run results each time and the best result plot are shown below.

TABLE VII. PROPOSED ALGORITHM FOR CORN PLANT DISEASE

Test 1	Test 2	Test 3	Test 4	Test 5	Average
94.86%	94.50%	95.10%	95.58%	95.70%	95.15%

C. Experiment with Rice Data Set

The corn data set contains 6 classes of images of corn plant leaves including Bacterial-blight, Blast, Gray-leaf-spot, Brown-spot, Hispa, Turgro, and Healthy. The training and validation data number is shown below.

TABLE VIII. RICE PLANT DISEASE DATA SET DETAILS TO APPLY PROPOSED ALGORITHM (IMAGE DATA SET AFTER PRE-PROCESSING)

Rice Dataset from Mendeley, Kaggle and UCI	No. of images	Training data (80%)	Validation data (20%)
Bacterial-blight	1584	1267	317
Blast	1440	1152	288
Brown-spot	1600	1280	320
Hispa	565	452	113
Turgro	1308	1046	262
Healthy	1488	1190	298
	7992 (total)		

After dividing the rice dataset, we applied the data to the proposed CNN model. We kept the fine-tuning parameters for the wheat and corn data set, and we encountered the overfitting problem, which means that the algorithm did great with training data but poorly with the validation data. So, we have to fine-tune some of the training parameters, like raising the L2 regulation parameter to 0.00045, cutting max epochs to stop the training early, updating the drop-off rate to 0.65, etc. The result is not as good as the corn and what data set, the best accuracy result is 92.74%. But the result is still higher than [15] and [12]. Same as the wheat data set, since the data was chosen randomly to train the algorithm, so result varies each time you run the algorithm. So, we did 5 times of the experiment and got an average accuracy result of 92.47%. The test run results each time and the best result plot are shown below.

TABLE IX. PROPOSED ALGORITHM ON RICE PLANT DISEASE

Test 1	Test 2	Test 3	Test 4	Test 5	Average
92.18%	92.30%	92.49%	92.62%	92.74%	92.47%

After all the experiments, the result for each data set and relevant work during the literature review is shown below. Other researchers applied CNN, SVM, random forest tree, VGG16, etc., but the graph shows clearly that the proposed algorithm shows better performance.

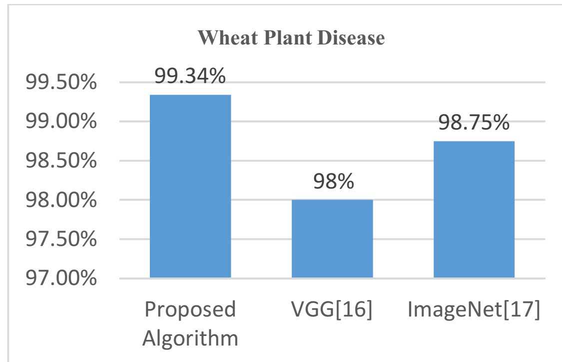


Fig. 5. Results compared with other State-of-Art research on wheat plant disease.

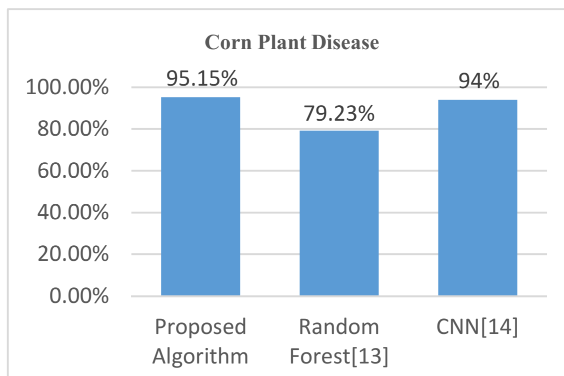


Fig. 6. Results compared with other State-of-Art research on corn plant disease.

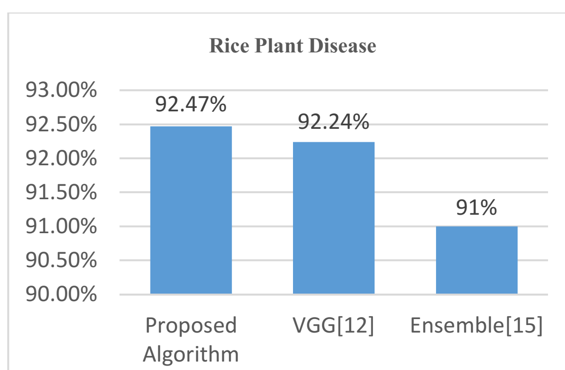


Fig. 7. Results compared with other State-of-Art research on rice plant disease.

VI. CONCLUSIONS

In this paper, the pre-trained AlexNet works with different layers and fine-tuning techniques as the new proposed algorithm is proposed for plant disease detection. The authors achieved an accuracy of 95.15%, 99.34%, and 92.47% for corn disease detection, wheat disease detection, and rice disease detection respectively. In the future, we will try to combine more datasets to evaluate the inclusiveness of the algorithm that was proposed in this paper; and In Situ Resource Utilization (ISRU) image monitoring, so the farmer will be alarmed when disease on a plant is detected automatically.

In this research work, the proposed innovative method was proved to be successful. The main contributions were listed below:

- Various sources provided the images, which were preprocessed to ensure that they were uniform in size and format, making them suitable for the ML model's input layer.
- Data from diverse sources were used to evaluate the proposed ML model's adaptability and robustness.
- The training process parameters were fine-tuned to handle various situations such as overfitting and underfitting.
- To demonstrate the proposed ML model's robustness and universality, it was tested on various datasets with different classification classes instead of being limited to specific datasets.
- Unlike other researchers who only tested one of the staple food plant datasets or compared it to non-staple plant data such as tomato, pepper, or potato [18], the proposed algorithm demonstrated robustness for the first time across all three staple food plant datasets (corn, wheat, and rice) [12]–[17].

ACKNOWLEDGMENT

The authors wish to thank the support from the National Science Foundation (NSF) grant #2011927, CAM-STAR at UDC via NASA MUREP grant under cooperative agreement #80NSSC19M0196 and via Department of Defense (DoD) under award #W911NF2010274.

REFERENCES

[1] J.F. Ramirez Rochac, N. Zhang, T. Deksissa, J. Xu, and L. Thompson, "A Hybrid ConvLSTM Deep Neural Network for Noise Reduction and Data Augmentation for Prediction of Non-linear Dynamics of Streamflow," *The 22nd IEEE International Conference on Data Mining (ICDM)*, Orlando, FL, November 28 - December 1, 2022.

[2] J. F. Ramirez Rochac, N. Zhang, T. Deksissa, and W. H. Mahmoud, "Streamflow Prediction Using a Hybrid Methodology Based on Convolutional Neural Network and Long Short-Term Memory," *2022 IEEE Eighth International Conference on Big Data Computing Service and Machine Learning Applications (BigDataService)*, San Francisco Bay Area, CA, August 15-18, 2022.

[3] J. F. Ramirez Rochac, N. Zhang, and J. Xiong, "A Spectral Feature Based CNN Long Short-Term Memory Approach for Classification", *The Tenth International Conference on Intelligent Control and Information Processing (ICICIP 2019)*, Marrakesh, Morocco, December 14-19, 2019.

[4] J. F. Ramirez Rochac, N. Zhang, J. Xiong, J. Zhong, and T. Oladunni, "Data Augmentation for Mixed Spectral Signatures Coupled with Convolutional Neural Networks", *The 9th International Conference on Information Science and Technology (ICIST 2019)*, Hulunbuir, Inner Mongolia, China, August 2-5, 2019.

[5] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Comput Electron Agric*, vol. 147, pp. 70–90, Apr. 2018, doi: 10.1016/J.COMPAG.2018.02.016.

[6] D. P. Hughes and M. Salathe, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," *Cornell University*, Apr. 2016.

[7] B. Liu, Y. Zhang, D. He, and Y. Li, "Identification of Apple Leaf Diseases Based on Deep Convolutional Neural Networks," *Symmetry (Basel)*, vol. 10, no. 1, 2018, doi: 10.3390/sym10010011.

[8] T. Wiesner-Hanks *et al.*, "Image set for deep learning: field images of maize annotated with disease symptoms," *BMC Res Notes*, vol. 11, no. 1, p. 440, 2018, doi: 10.1186/s13104-018-3548-6.

[9] J. Parraga-Alava, K. Cusme, A. Loor, and E. Santander, "RoCoLe: A robusta coffee leaf images dataset for evaluation of machine learning based methods in plant diseases recognition," *Data Brief*, vol. 25, p. 104414, 2019, doi: https://doi.org/10.1016/j.dib.2019.104414.

[10] Prajapati H.B., Shah J.P., and Dabhi V.K., "Detection and Classification of Rice Plant Diseases," *Intelligent Decision Technologies*, pp. 357–373, 2017.

[11] Oyewola DO, Dada EG, and Damaševičius R, "Detecting cassava mosaic disease using a deep residual convolutional neural network with distinct block processing," *PeerJ Comput Sci*, 2021.

[12] R. Yakkundimath, G. Saunshi, B. Anami, and S. Palaiah, "Classification of Rice Diseases using Convolutional Neural Network Models," *Journal of The Institution of Engineers (India): Series B*, vol. 103, no. 4, pp. 1047–1059, 2022, doi: 10.1007/s40031-021-00704-4.

[13] H. and S. A. K. and M. S. C. Panigrahi Kshyanaprava Panda and Das, "Maize Leaf Disease Detection and Classification Using Machine Learning Algorithms," in *Progress in Computing, Analytics and Networking*, P. K. and R. S. S. and L. K.-C. Das Himansu and Pattnaik, Ed., Singapore: Springer Singapore, 2020, pp. 659–669.

[14] M. Agarwal, V. K. Bohat, Mohd. D. Ansari, A. Sinha, S. Kr. Gupta, and D. Garg, "A Convolution Neural Network based approach to detect the disease in Corn Crop," in *2019 IEEE 9th International Conference on Advanced Computing (IACC)*, 2019, pp. 176–181. doi: 10.1109/IACC48062.2019.8971602.

[15] R. Deng *et al.*, “Automatic Diagnosis of Rice Diseases Using Deep Learning,” *Front Plant Sci*, vol. 12, Aug. 2021, doi: 10.3389/fpls.2021.701038.
[16] N. Jahan, P. Flores, Liu Zhaohui, Friskop Andrew, Mathew Jithin Jose, and Zhang Zhao, “Detecting and Distinguishing Wheat Diseases using Image Processing and Machine Learning Algorithms,” in *2020 ASABE Annual International Virtual Meeting*, 2020. Accessed: Apr. 04, 2023. [Online]. Available: doi:10.13031/aim.202000372
[17] Z. Jiang, Z. Dong, W. Jiang, and Y. Yang, “Recognition of rice leaf diseases and wheat leaf diseases based on multi-task deep transfer learning,” *Comput Electron Agric*, vol. 186, p. 106184, 2021, doi: https://doi.org/10.1016/j.compag.2021.106184.
[18] A. Lakshmanarao, M. R. Babu, and T. S. R. Kiran, “Plant Disease Prediction and classification using Deep Learning ConvNets,” in *2021 International Conference on Artificial Intelligence and Machine Vision (AIMV)*, 2021, pp. 1–6. doi: 10.1109/AIMV53313.2021.9670918.
[19] TAIRU OLUWAFEMI EMMANUEL, “PlantVillage Dataset.” https://www.kaggle.com/datasets/emmarex/plantdisease (accessed Feb. 27, 2023).
[20] OLYAD GETCH, “Wheat Leaf dataset.” https://www.kaggle.com/datasets/olyadgetch/wheat-leaf-dataset (accessed Feb. 27, 2023).
[21] J. S. V. D. HB Prajapati, “Rice Leaf Diseases Data Set,” *Detection and classification of rice plant diseases*, Jan. 01, 2017.
[22] Prabira Kumar Sethy, “Rice Leaf Disease Image Samples.” Mendeley, 2020. doi: 10.17632/FWCJ7STB8R.1.
[23] Sergey Ioffe and Christian Szegedy, “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift,” Cornell University, 2015. Accessed: Apr. 03, 2023. [Online]. Available: https://arxiv.org/abs/1502.03167v3#
[24] Jason Brownlee, “A Gentle Introduction to Dropout for Regularizing Deep Neural Networks,” *Machine Learning Mastery*, Aug. 06, 2019.