Image-Based Crop Leaf Disease Automatic Identification Using Convolutional Neural Network (CNN)

Jomarey D. Fernandez and Maria Art Antonette D. Clariño

***Abstract*— Globally, crop leaf diseases have been an enormous burden on farmers and a major threat to food security. The latest breakthrough in machine learning paved the way for automatic identification of crop leaf diseases. Using the PlantVillage dataset consisted of 35, 263 images, a convolutional neural network was trained to identify 19 crop-disease class labels. The trained model yielded an overall accuracy of 98.742%, mean recall of 88.431%, mean precision of 89.073%, and mean F1-Score of 88.408% on 5-fold cross-validation, demonstrating the feasibility of this approach. The proposed convolutional neural network model provides great potential and direction in related crop disease control and machine learning studies.**

***Index Terms*— Agriculture, Convolutional Neural Network (CNN), Crop Disease Detection, K-Fold Cross-Validation, Machine Learning**

I. INTRODUCTION

A. Background of the Study

Several countries' economies are highly dependent on agricultural productivity. According to The World Bank, about 3.432% of all the countries' GDP is shared solely by the agricultural sector [1]. Not only that agriculture provides food and livelihood for millions of people, but an increase in agricultural output and productivity is seen to contribute substantially to the overall economic development of countries. Hence, agriculture is considered as the backbone of most economies.

By 2050, it is projected that the world's population will be at 9.7 billion, 34 percent higher than today [2]. To respond to the rapidly growing food demand, experts presume that there is a need to increase food production by an estimated 70% to feed the world [3]. Many farmers and manufacturers involved in the agriculture sector are already taking advantage of modern technologies– particularly digital tools– to solve the impending food demand. Despite these innovations, food security remains under threat by different factors, including the decline in pollinators, climate change, and plant diseases [4].

Globally, crop diseases have been an enormous burden on farmers. Billions of dollars a year are spent on crop-disease management to cease crop losses [5]. About 40-50% of the total crop yields are lost due to pests, crop diseases, or post-harvest losses in developing countries [6]. And until recently, adequate technical knowledge and support are inaccessible to poor and smallholder farmers, resulting in poor disease control, pollution, and harmful results. These losses on agricultural productivity can be prevented by an accurate diagnosis and early treatment of crop disease.

In this study, the technical feasibility of timely and accurate diagnosis of crop leaf disease through a machine learning approach was demonstrated, utilizing the 35,264 images of 4 crop species with 19 class labels dataset from the project PlantVillage.

B. Statement of the Problem

Despite the vast development and innovations in technology, farmers are still accustomed to using traditional methods in checking crop leaf diseases. Manual checking for crop diseases is a tedious process yet it may lead to inaccurate results. Without the insights of experts, farmers can find themselves withstanding crop disease with one-size-fits-all solutions that not only fail to address their problems but often result in undesired side effects.

With the availability of public datasets of crop leaf diseases and the development of machine learning, farmers may utilize the use of systems to monitor their crops for diseases. Convolutional Neural Network (CNN) is used for creating a model that may classify crop diseases at the early stage of plant growth to reduce crop loss.

C. Objectives of the Study

The general objective of the study is to develop a model that will identify crop leaf diseases using Convolutional Neural Network (CNN). Specifically, the study aims to achieve the following:

1. To train the model using Convolutional Neural Network (CNN) with crop-diseases labeled data from PlantVillage dataset;

2. To validate the model's performance using K-Fold Stratified Cross-Validation;

3. To compute for performance evaluation metrics for the created model and to accurately identify the crop leaf diseases test dataset.

Presented to the Faculty of the Institute of Computer Science, University of the Philippines Los Baños in partial fulfillment of the requirements for the Degree of Bachelor of Science in Computer Science

D. Significance of the Study

With the development of modern technology, it is important for us to incorporate modern technology to the main source of our economic growth, as well as our daily food. The use of machine learning in the identification of crop leaf diseases will benefit the farmers since it will provide an automatic classification of crop leaf diseases for them. Accurate detection and classification of crop leaf diseases will help farmers to maintain and to protect their crops from further infections and deterioration.

E. Scopes and Limitations

The scope of this study is to create a model for classifying crop leaf diseases. This model is limited to detecting the following crop species and diseases: 1) Corn Gray Leaf Spot, 2) Corn Common Rust, 3) Corn healthy, 4) Corn Northern Leaf Blight, 5) Bell Pepper Bacterial Spot, 6) Bell Pepper Healthy, 7) Potato Early Blight, 8) Potato Healthy, 9) Potato Late Blight, 10) Tomato Bacterial Spot, 11) Tomato Early Blight, 12) Tomato Healthy, 13) Tomato Late Blight, 14) Tomato Leaf Mold, 15) Tomato Septoria Leaf Spot, 16) Tomato Two Spotted Spider Mite, 17) Tomato Target Spot, 18) Tomato Mosaic Virus, and 19) Tomato Yellow Leaf Curl Virus.

II. Review of Related Literature

A. Plant Leaf Disease Detection

Song, K et al. (2011), in their study, used image processing and backpropagation neural network to recognize corn leaf diseases. They used the YCbCr color space and YCbCr Spatial color model for image segmentation for segment diseases spot and matrix spatial gray level layer to extract features. The authors have taken a total of 10 pictures of corn leaf blight, sheath blight, southern leaf blight, and applied different degrees of matrix enhancement to each of them: 0 degrees, 45degrees, 90degrees, and 135degrees is applied, respectively. The authors extracted five types of texture features: energy, entropy, a moment of inertia, related and local stationary. Using Backpropagation neural network, the texture features are used as inputs to recognize and classify the maize leaf diseases stated above [7].

Moreover, in 2015, Z Zhiyong and H Xiaoyang et al. developed a system to recognize and classify four types of maize plant diseases that are common in the China farm areas. Multiple images in the JPG format of maize plants with diseases are collected using a digital camera under the sunlight condition. To acquire the useful information, it is then converted into BMP format. After that, the image is segmented using a thresholding value and the image is again transformed from its RGB color model to HIS. The mean and standard deviation of the HIS features are computed afterwards. The researchers extracted a total of 20 features from the dataset of maize plants to recognize and classify the diseases using the GA-SVM algorithm. They used RBF kernel function, SVM, and GA-SVM to detect and classify the diseases. The experimental results achieved precision between 69.63% and 90.09% for SVM and between 88.72% and 92.59% for GA-SVM for each of the maize plant disease [8].

Pujari JD and Yakkundimath R, on the succeeding year (2016), recognize and classified three Cereals plants, which are Wheat, Maize and Jowar leaf diseases using Support Vector Machine(SVM) and Artificial Neural Network(ANN). The said technologies are used for based recognition and classification of visual symptoms affected by the crop diseases. The authors collected four types of maize leaf diseases: leaf spot, leaf blight, smut, powdery mildew, and normal. To recognize and classify fungal disease symptoms, the authors followed the following steps: first, they have collected 750 JPG format images both fungal affected and normal. After that, they have preprocessed the images using color correction, removing unnecessary artifacts, and formatting (or the process of storing the image representation and attributes). Then the image is segmented using the K-means segmentation technique; the color and texture features are extracted from the affected areas. Lastly, the extracted features are used as inputs to the SVM and ANN classifiers. For feature extraction from leaf image, the researchers used Color Co-occurrence Matrix (CCM) algorithm and, the MATLAB tool is used for the program interface. Based on this experimental result SVM and ANN average result, 83.83% and 77.75% classification accuracy was achieved respectively. From this result the authors conclude that the SVM algorithm is more accurate than ANN to recognize and classify cereals fungal diseases [9].

In a paper by Hussain S.A et al. (2018),on the other hand, they have observed the various techniques used in segmenting the infected part of the plant with diseases. The paper also talks about some feature extraction used to get the features of infected leaf and classification techniques to classify plant diseases. They suggested the use of Artificial Neural Network(ANN) methods for classification of disease in plants such as self-organizing feature maps, backpropagation algorithms, SVMs, and others that can be efficiently used. From these methods, you can accurately identify and classify various plant diseases using image processing techniques [10].

B. K-Fold Cross Validation

In machine learning, performances of classifiers are usually measured using prediction error however in a real-world scenario, these errors cannot be accurately calculated but usually estimated. In a study by Rodriguez J., Perez, A. (2020), they implied that there is a need to choose an appropriate estimator for these prediction errors. They analyze the statistical properties, comparing the variance and bias of the k-fold cross-validation classification error estimator (k-cv) for different values of k. Their experimental study has been performed for two classifiers: naive Bayes and nearest neighbor, different number of folds, training sets, and sample sizes. They recommended the practical use of k-fold cross-validation in measuring the performance of classifiers [11].

In today's trend, it is popular to evaluate the performance of classifiers by k-fold cross-validation. In 2019, Wong T., Yeh, P. explored in their study whether k-fold cross-validation should be repeatedly performed for obtaining reliable accuracy estimates. They first analyzed the dependency relationships of the predictions of the same instance in two replications of k-fold cross-validation; using k-nearest neighbors with $k = 1$.

Then after that, statistical methods are proposed to test the correlation of the dependency level between the results above and the accuracy estimates obtained from various replications of k-fold cross-validation. The results show that there is a higher collaboration on the estimates obtained from various replications and those with a higher number of folds. The authors concluded that the k-fold cross-validation with a small number of replications and a large number of folds should be used for performance evaluation of classification algorithms [12].

There are different cross-validation techniques available but the most common are leave-one-out cross-validation and k-fold cross-validation. In the study by Raschka, S. (2020), these cross-validation techniques are reviewed. Several statistical tests for algorithm comparisons, such as omnibus tests and multiple-comparison corrections, are presented and discussed. After using alternative methods for algorithm selection, the authors have recommended the use of combined F-test 5x2 cross-validation and nested cross-validation for comparing machine learning algorithms, mainly when datasets are too small [13].

In the study by Ahmed F., Ali, Y., Shamsuddin S. (2018), they presented a comparison of the performance of the following methods: proposed method, Backpropagation(BP), and Standard SpikeProp. The authors used K-fold cross-validation to investigate these methods and findings revealed that the proposed method has a better performance compared to the other two; K-fold cross-validation is performed for all datasets use [14].

Pal K., Patel, B. (2020) in their research article has collected over 154 Hindi Poetries from the web. It is then processed using NLP techniques and features are extracted from it. They have built a classifier with 5 different machine learning methods namely: Naïve Bayes, Decision Tree Algorithm, K-Nearest Neighbors, Support Vector Machine, and Random Forest Algorithm. To check the reliability of these methods, the results were examined using a holdout accuracy estimator and a k-fold cross-validation accuracy estimator. The authors have found out that the results of NB, SVM, and random forest methods are better machine learning methods compared to K-NN and DTT for all of the datasets [15].

In brief, it is noted that there are various techniques used in identifying crop leaf diseases but Neural Networks are the most prevalent in this kind of studies. It is also considered that the k-fold cross-validation with a small number of replications and a large number of folds should be used for performance evaluation of classification algorithms.

III. Methodology

A. Materials

The following minimum specifications of the software was used to develop the model:

1. Laptop with specifications: Intel Pentium Dual Core 4th Gen Processor @ 2.1 Ghz, 2 GB DDR3 RAM, Linux 16.04 Operating System.

B. Development Tools

The model will be developed under the Linux Operating System. It will use the following tools:

1. Programming Language: Python was used to implement the model.

2. Machine Learning: TensorFlow, an open-source machine learning library for research and production, was used to create the neural network.

C. Dataset Description

Images with crop-disease labels were obtained from Kaggle.com, an online platform where you can get open-sourced datasets. The PlantVillage dataset is composed of 35,263 images of plant leaves that are prelabeled with crop-disease labels, varying from a total of 19 classes. Figure 2 shows each example of leaf images from the PlantVillage dataset [16].

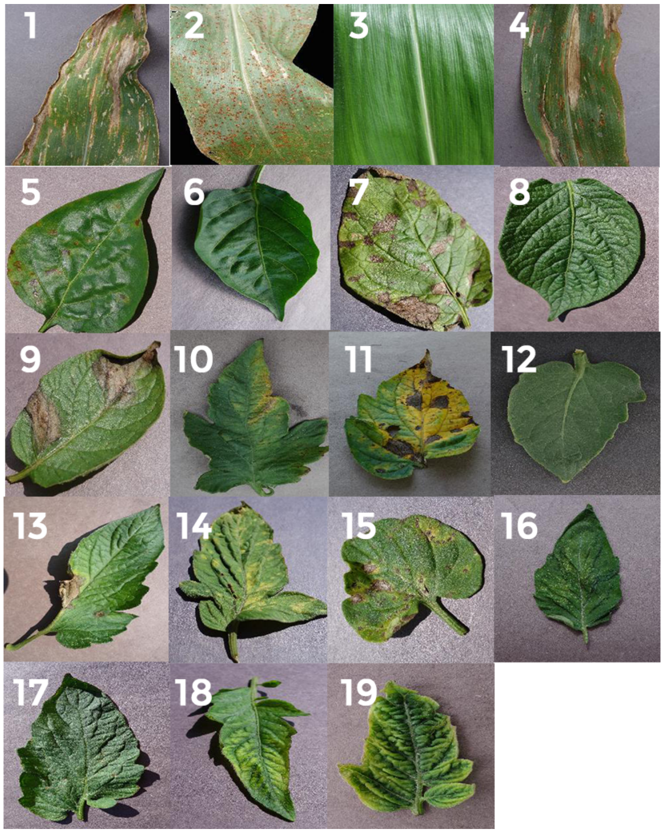


Fig. 1. Each example of crop-disease pair from the PlantVillage dataset. 1)Corn Gray Leaf Spot, 2)Corn Common Rust, 3)Corn healthy, 4)Corn Northern Leaf Blight, 5)Bell Pepper Bacterial Spot, 6)Bell Pepper Healthy, 7)Potato Early Blight, 8)Potato Healthy, 9)Potato Late Blight, 10)Tomato Bacterial Spot, 11)Tomato Early Blight, 12)Tomato Healthy, 13)Tomato Late Blight, 14)Tomato Leaf Mold, 15)Tomato Septoria Leaf Spot, 16)Tomato Two Spotted Spider Mite, 17)Tomato Target Spot, 18)Tomato Mosaic Virus, 19)Tomato Yellow Leaf Curl Virus

D. Dataset Division

The dataset is divided into two subsets: the training dataset and the testing dataset. Following the 80-20 train-test split, 28,209 images are allotted for the training dataset and 7,054 is allocated for testing dataset. As the PlantVillage dataset has multiple images of a leaf in different orientations, it is ensured that all the images of the same leaf are either included in the training dataset or the testing dataset. Overall, there is a total of 35,263 images in our dataset.

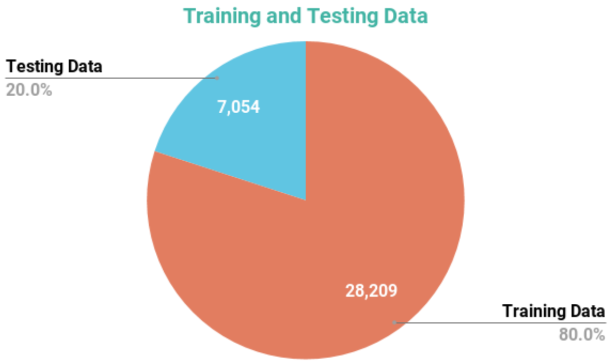


Fig. 2. The number of images for training and testing dataset

Furthermore, the distribution of training and testing images across all 19 class(label) is shown below.

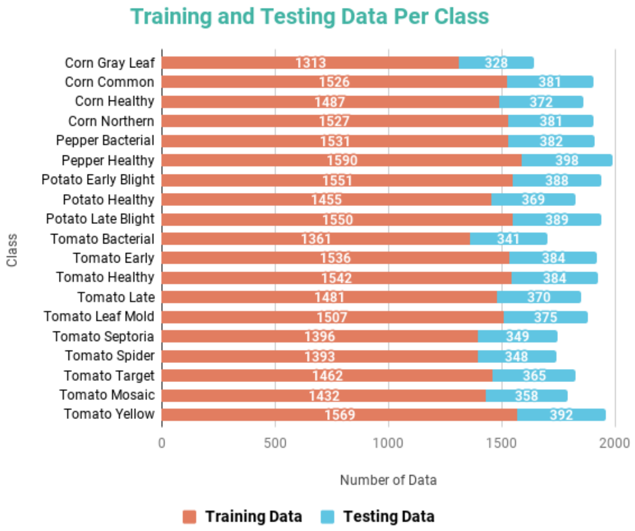


Fig. 3. The number of images per class

E. Image Enhancement

Before feeding the images to the neural network, two necessary pre-processing steps were applied to these images. First, the images were converted to its 3-channel BGR color image. Then after that, these images were resized to match the size of the input layer of the neural network. From their original sizes, each image was resized to 50 x 50 image pixels.

F. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) is a widely used neural network for solving complex deep learning problems. Its classification capability based on contextual information made it useful in image processing and natural language processing. The general model of CCN consists of four components namely (a) convolution layer, (b) pooling layer, (c)activation function, and (d) fully connected layer [18].

The CNN model used is consisted of 5 convolution layers. Convolutional layers are the layers where multiple filters are applied to the input image to extract features [19]. In our model, 32, 64, 128, 32, and 64 filters were utilized on each convolutional layer respectively. Each convolution layer is followed by normalization and max-pooling layer, which takes the maximum value in a certain filter region. To introduce non-linearity in our CNN, all the layers in the network usually have Rectified Linear Unit (ReLu) activation that replaces all negative pixel values in the feature map by zero. After that, it is followed by two fully connected layers, a dropout layer(a technique used to improve over-fit on neural networks) and a softMax layer(used to normalize neural networks output to fit between zero and one). The output from the convolutional and pooling layers represent high-level features of the input image. The purpose of the fully connected layer is to use these features for classifying the input image into various classes based on the training dataset [21].

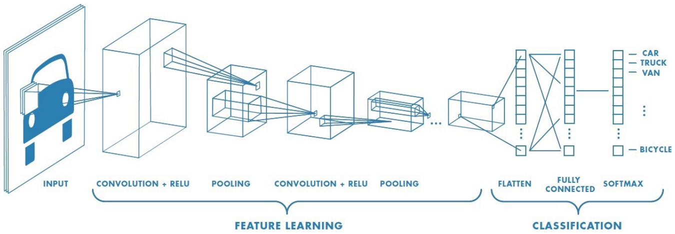


Fig. 4. CNN general steps to classify images as discussed by Saha [20]

The following hyperparameters were used in all model setup:

- Learning Rate: $1^{e^{-3}}$
- Optimizer: Adam
- Loss: Categorical Cross-Entropy
- Epoch: 12
- Solver Type: Batch Gradient Descent

G. Performance Evaluation of the CNN Classifier

Five-fold cross-validation will be used to validate the performance of the CNN Classifier. The training dataset will be divided into 5 equally subsets. Each fold, an assigned subset will be used for the validation and the remaining 4 subsets will be used for training the model. The training subset will be used to fit the model and tune in its hidden parameters: such as weights and biases. While the validation subset, on the other hand, will be used for an unbiased evaluation of a model fit and manually adjusting the hyperparameters. These hyperparameters are the configurations that cannot be automatically learned during training.

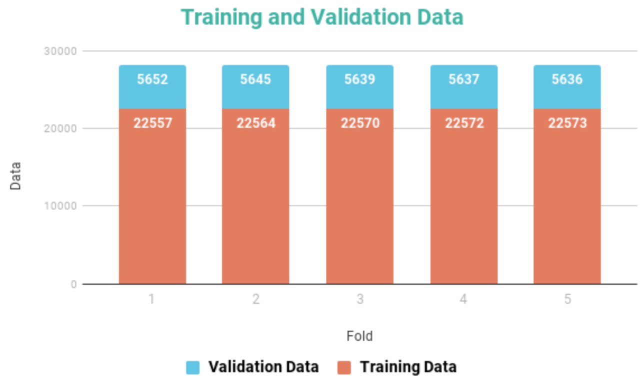


Fig. 5. The distribution of Training-Validation dataset for each fold

The figure above shows the distribution of of Training Dataset vs Validation Dataset for each fold. To ensure that each class is well represented across each fold, the stratified k-fold cross-validation approach was implemented in rearranging the data. Stratification helps with reducing both bias and variance.

To further determine the performance of our CNN classifier, different evaluation metrics for a multiclass classification model were used. All these metrics can be understood in the context of a binary classification model. The following categories will be utilized in deriving our formula for these evaluation metrics.

- True Positives (TP): Items where the true label is positive and whose class is correctly predicted to be positive.
- False Positives (FP): Items where the true label is negative and whose class is incorrectly predicted to be positive.
- True Negatives (TN): Items where the true label is negative and whose class is correctly predicted to be negative.
- False Negatives (FN): Items where the true label is positive and whose class is incorrectly predicted to be negative.

The evaluation metrics for machine-learning classification models that were used in this study are Accuracy, Recall, Precision, and F1-score.

The first metric utilized is accuracy. The accuracy tells us how often the model is making a correct prediction. It shows us the number of items correctly identified as either true positive or true negative out of the total number of items. Secondly, the recall was computed; it is the ratio of correctly predicted positive observations to all observations in the actual class. Moreover, to know the ratio of correctly predicted positive observations to the total predicted positive observations, the precision of the model was calculated. Lastly, the f1-score of the model was determined to measure the model's overall accuracy that combines precision and recall, and to measure the effectiveness of identification. The formula used to calculate these metrics are as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{4}$$

After computing for these metrics, the average accuracy, precision, recall, and f1-score per fold and across all folds were assessed.

IV. Results and Discussion

During the training phase of the developed model in this study, the training and validation accuracies and loss per epoch were recorded. It can be deduced from these metrics if the model is under-fitted, over-fitted, or good-fitted. In the figures below (Fig. 6-7), one observation is that as epoch increases, training accuracy increases. Inversely, as the epoch increases, the loss decreases. On figure 6, it is noticed that there is no divergence and there is a small gap between the validation loss and the training loss, confirming that over-fitting is not a factor to the resulting model obtained across all folds.

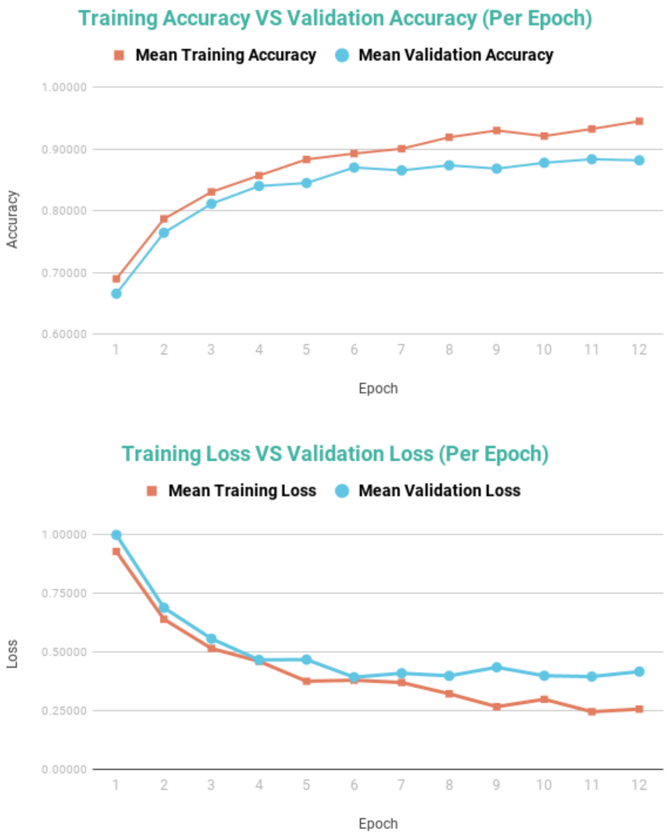


Fig. 6-7. The Mean Training and Validation Accuracy and Loss per Epoch

After the creation of models, the CNN classifier is tested using the 20%(7054) of the original dataset. These data were kept unseen by the model during the training phase.

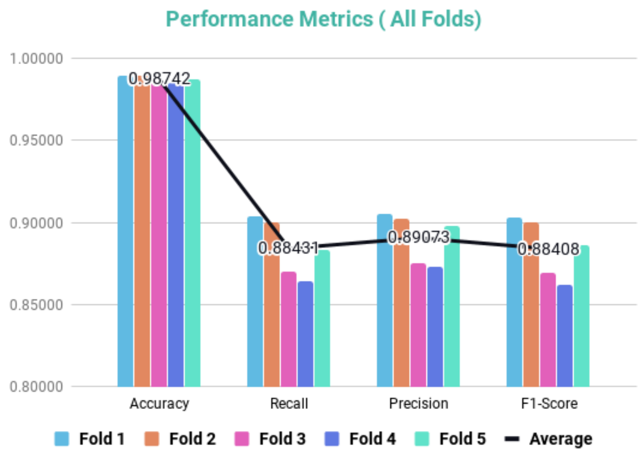
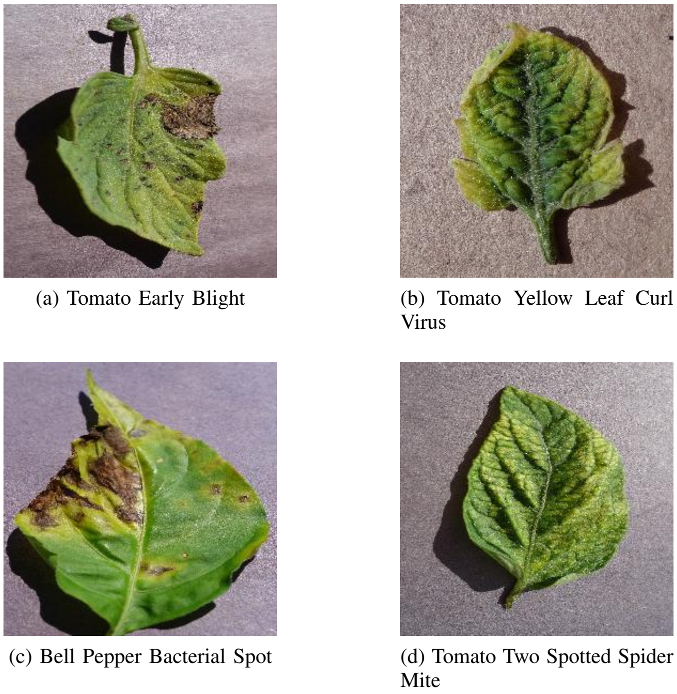


Fig. 8. The computed value of metrics for each fold

Based on the figure above, an overall accuracy of 98.742%, mean recall of 88.431%, mean precision of 89.073%, and mean F1-Score of 88.408% were computed from the performance of the classifier across all folds. Hence, showing the feasibility of using convolutional neural network approach for similar classification problems.

It is perceived that Fold 1 yielded the highest value for the computed metrics; having 98.93% for accuracy, 90.35% for recall, 90.56% for precision, and 90.28% for f1-score. The table 1 below gives information about the confusion matrix, the accuracy and the f1-score per class label of Fold 1.

The confusion matrix allows comparison between the number of predicted label versus the true label. For instance, it is observed that Corn Common Rust performs well in determining its true positives and false positive. Covering all class labels, it only classified 2 false positive; having 99.95% accuracy. At the same time, it is noticed that Tomato Early Bright performed the lowest. With 97.46% accuracy and 77.58% F1-Score, it is noted that Tomato Early Bright is commonly classified as Tomato Yellow Leaf Curl Virus(15), Bell Pepper Bacterial Spot(10), and Tomato Two Spotted Spider Mite(12).



(a) Tomato Early Blight

(b) Tomato Yellow Leaf Curl Virus

(c) Bell Pepper Bacterial Spot

(d) Tomato Two Spotted Spider Mite

Fig. 9. Image comparision of Tomato Early Bright,Tomato Yellow Leaf Curl Virus, Bell Pepper Bacterial Spot, and Tomato Two Spotted Spider Mite

TABLE I
FOLD 1 CONFUSION MATRIX , ACCURACY AND F1-SCORE

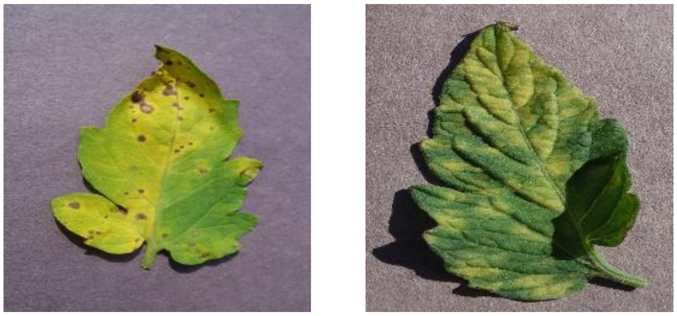
Class Label	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	AC	FS
1	298	0	3	18	0	0	0	0	2	0	4	0	3	0	0	0	0	0	0	98.75%	87.64%
2	1	380	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	99.95%	99.60%
3	0	1	371	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	99.89%	99.06%
4	43	0	3	332	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	98.97%	90.58%
5	0	0	0	0	343	6	10	2	2	1	6	0	1	1	9	0	1	0	0	98.79%	89.43%
6	0	0	0	0	10	359	0	8	0	1	0	0	0	2	18	0	0	0	0	99.17%	92.88%
7	0	0	0	0	1	0	375	0	7	0	0	0	3	0	2	0	0	0	0	99.52%	95.90%
8	0	0	0	0	1	1	0	352	3	0	0	0	0	0	0	4	8	0	0	99.31%	93.86%
9	0	0	0	0	1	0	1	13	343	1	4	0	5	0	6	13	2	0	0	98.89%	90.26%
10	1	0	0	0	0	0	0	1	0	313	6	0	2	0	2	0	4	0	12	99.36%	93.57%
11	3	0	0	1	10	6	0	2	2	7	296	0	9	5	7	12	8	1	15	97.46%	77.58%
12	0	0	0	0	0	0	0	0	0	0	0	379	0	0	0	0	5	0	0	99.70%	97.42%
13	6	0	0	1	4	1	4	2	6	2	46	3	280	1	8	3	1	1	1	98.15%	81.75%
14	0	0	0	0	4	1	1	0	0	0	8	5	2	328	12	11	0	2	1	99.01%	90.85%
15	0	0	0	0	7	0	3	0	5	0	5	3	5	9	290	8	11	2	1	98.09%	81.80%
16	0	0	0	0	0	0	0	0	0	0	1	0	2	0	0	332	7	4	2	97.73%	81.37%
17	0	0	0	0	1	0	0	1	1	0	2	4	0	0	4	81	271	0	0	97.90%	79.23%
18	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	2	1	353	0	99.77%	97.91%
19	0	1	0	0	3	0	0	0	0	3	1	0	0	1	1	2	0	0	380	99.34%	94.52%
																				98.93%	90.28%

Legend: 1-Corn Gray Leaf Spot, 2-Corn Common Rust, 3-Corn healthy, 4-Corn Northern Leaf Blight, 5-Bell Pepper Bacterial Spot, 6-Bell Pepper Healthy, 7-Potato Early Blight, 8-Potato Healthy, 9-Potato Late Blight, 10-Tomato Bacterial Spot, 11-Tomato Early Blight, 12-Tomato Healthy, 13-Tomato Late Blight, 14-Tomato Leaf Mold, 15-Tomato Septoria Leaf Spot, 16-Tomato Two Spotted Spider Mite, 17-Tomato Target Spot, 18-Tomato Mosaic Virus, 19-Tomato Yellow Leaf Curl Virus, AC-Accuracy, and FS-F1-Score.

Figure 9 shows the image comparison of Tomato Early Bright,Tomato Yellow Leaf Curl Virus, Bell Pepper Bacterial Spot, and Tomato Two Spotted Spider Mite class labels. It is observed that the size of infected disease in Tomato Early Bright, and Bell Pepper Bacterial Spot are relatively same based on the image while the Tomato Early Bright, Tomato Yellow Leaf Curl Virus, and Tomato Two Spotted Spider Mite have comparatively the same color on the infected parts. It also noted that Tomato Two Spotted Spider Mite acquired the highest false positive or incorrect classification of class label in this fold: 136 images.

Meanwhile, table 2, on the right, shows the summary of performance metrics per class label. From the results displayed, it is noticeable that Corn Healthy class label yielded the highest overall accuracy (99.922%), mean recall (99.731%), mean precision (98.883%), and mean f1-score (99.304%). On the contrary, Tomato Septoria Leaf Spot produced the lowest computed value for overall accuracy, mean recall, mean precision, and mean f1-score– 97.392%, 70.716%, 78.547%, and 72.884%, respectively.

It is observed that the CNN classifier often distinguish Tomato Septoria Leaf Spot as Tomato Leaf Mold. On average, about 8% of the Tomato Septoria Leaf Spot labeled images were tagged as Tomato Leaf Mold. The images below show the side-by-side image comparison of the two class labels. As observed, both leaf classes shared the same color and textures. The size of infected parts is approximately the same for both classes, leading to the confusion of the CNN classifier.



(a) Tomato Septoria Leaf Spot

(b) Tomato Leaf Mold

Fig. 9. Image comparision of Tomato Septoria Leaf and Tomato Leaf Mold

In summary, we noted that the overall accuracy of the CNN classifier ranges from 97%-99%. Also, we observed that the mean recall, mean precision and mean f1-score span from 72%-99%.

TABLE II

SUMMARY OF OVERALL ACCURACY, MEAN RECALL, MEAN PRECISION, AND MEAN F1-SCORE PER CLASS LABEL

Class Label	Accuracy	Recall	Precision	F1-Score
Corn Gray Leaf Spot (328)	98.824%	87.927%	88.063%	87.858%
Corn Common Rust (381)	99.874%	98.740%	99.063%	98.896%
Corn Healthy (372)	99.922%	99.731%	98.883%	99.304%
Corn Northern Leaf Blight (381)	99.025%	91.076%	91.823%	91.380%
Pepper Bacterial Spot (382)	98.377%	83.717%	87.873%	85.173%
Pepper Healthy (398)	99.043%	91.457%	92.705%	91.940%
Potato Early Blight (388)	99.389%	96.392%	93.361%	94.835%
Potato Healthy (369)	98.905%	94.201%	88.418%	90.719%
Potato Late Blight (389)	98.538%	90.797%	85.438%	87.928%
Tomato Bacterial Spot (341)	98.960%	85.337%	93.655%	89.119%
Tomato Early Blight (384)	97.088%	76.406%	75.165%	74.971%
Tomato Healthy (384)	99.733%	97.708%	97.671%	97.683%
Tomato Late Blight (370)	97.662%	75.081%	81.344%	77.935%
Tomato Leaf Mold (375)	98.459%	88.053%	85.591%	86.579%
Tomato Septoria Leaf Spot (349)	97.392%	70.716%	78.547%	72.884%
Tomato Spider Mites (348)	98.215%	85.172%	82.086%	83.184%
Tomato Target Spot (365)	98.061%	78.904%	84.601%	81.403%
Tomato Mosaic Virus (358)	99.356%	95.196%	93.756%	94.215%
Tomato Yellow Leaf Curl Virus (392)	99.267%	93.571%	94.337%	93.750%
	98.742%	88.431%	89.073%	88.408%

V. CONCLUSION

In this study, it is aimed to demonstrate the technical feasibility of a Crop Leaf Diseases Automatic Identification using Convolutional Neural Network (CNN). With the use of the PlantVillage dataset, the overall accuracy of 98.742%, mean recall of 88.431%, mean precision of 89.073%, and mean F1-Score of 88.408% were computed from the performance of the classifier across all folds. Based on the quantitative performance evaluation metrics used in our model, it can be concluded that the use of Convolutional Neural Network in identifying image-based crop leaf disease is feasible.

VI. FUTURE WORKS AND RECOMMENDATIONS

The images used to train and to test the created models are from the PlantVillage dataset only. Additional datasets from different sources and different environment setups would improve the performance of the model. The model in this study can be used in creating a web-based or a mobile application where users can capture or input a leaf image and know if the crop is healthy or infected by a crop leaf disease.

References

[1] The World Bank. (2018). Agriculture, forestry, and fishing, value added(% of GDP). Retrieved from: https://data.worldbank.org

[2] Preneuf, Flore de. (2018). Understanding poverty: Agriculture and Food. Retrieved from: www.worldbank.org

[3] Fleischmann, M. (2013). World must sustainably produce 70 per cent more food by mid-century – UN report. Retrieved from: https://news.un.org/en/story/2013/12/456912

[4] Strange RN, Scott PR (2005) Plant disease: a threat to global food security. Phytopathology 43.

[5] Environmental Law Institute. (2002). Halting the Invasion: State Tools for Invasive Species Management.

[6] Elliott, Scott. (2015). Global scientists meet for integrated pest management idea sharing. Retrieved from: nifa.usda.gov/blog/

[7] K Song, Z liu, et al. (2011). A Research of maize disease image recognition of Corn Based on BP Networks. Measuring Technology and Mechatronics Automation China.global-scientists-meet-integrated-pest-management-idea-sharing.

[8] Z Zhiyong, H Xiaoyang, et al. (2015). Image recognition of maize leaf disease based on GA-SVM. Chemical Engineering Transactions.

[9] P Jagadeesh, R Yakkundimath, et al.(2016). SVM and ANN Based Classification of Plant Diseases Using Feature Reduction Technique. International Journal of Interactive Multimedia and Artificial Intelligence

[10] Hussain S.A et al. (2018). Classification and Detection of Plant Disease using Feature Extraction Methods.

[11] Rodriguez J., Perez, A. (2020). Sensitivity Analysis of k-Fold Cross Validation in Prediction Error Estimation.

[12] Wong T., Yeh, P. (2019). Reliable Accuracy Estimates from k-fold Cross Validation.

[13] Raschka S. (2018). Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning.

[14] Ahmed F., Ali, Y., Shamsuddin S. (2018). Using K-Fold Cross Validation Proposed Models for Spikeprop Learning Enhancements.

[15] Pal K., Patel, B. (2020). Data Classification with k-fold Cross Validation and Holdout Accuracy Estimation Methods with 5 Different Machine Learning Techniques.

[16] Hughes, D.P et al. (2015).An open access repository of images on plant health to enable the development of mobile disease diagnostics.

[17] Ray, S. (2018). Stratified k-fold cross validation[Online image]. https://www.analyticsvidhya.com/blog/2018/05/improve-model-performance-cross-validation-in-python-r/.

[18] Indolia S., Goswami A., Mishra S., Asopa P. (2018). Conceptual Understanding of Convolutional Neural Network- A Deep Learning Approach.

[19] Stewart, M. (2019). Simple Introduction to Convolutional Neural Networks. https://towardsdatascience.com/simple-introduction-to-convolutional-neural-networks-cdf8d3077bac.

[20] Saha, S. (2018). A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way[Online image]. https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53.

[21] Karn, U. (2016). An Intuitive Explanation of Convolutional Neural Networks. https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/