

WECNN-PDP: Weighted Ensemble Convolutional Neural Networks Models to Improve the Plant Disease Prediction

Deni Sutiaji^{1,2*}, Oktay Yildiz³, Harunur Rosyid¹, and Umi Chotijah¹

¹Department of Informatics Engineering, Faculty of Engineering, Universitas Muhammadiyah Gresik, Indonesia

²Department of Computer Science, Faculty of Informatics, Gazi University, Türkiye

³Department of Computer Engineering, Faculty of Engineering, Gazi University, Türkiye

Abstract. As an agricultural country, Indonesia's agricultural production is essential. However, crop failure will occur if diseases and other factors, such as natural disasters, attack many plant fields. These problems can be minimized by early detection of plant diseases. However, detection will be challenging if done conventionally. Prior research has shown that deep learning algorithms can perform detection with promising results. In this study, we propose a new weighted deep learning ensemble method as a solution for better performance in plant disease detection. We ensemble the model by considering the combination of two and three pre-trained convolutional neural networks (CNNs). Initially, we perform transfer learning on individual CNN models by prioritizing high-dimensional features through weight updates on the last few layers. Finally, we ensemble the models by finding the best weights for each model using grid search. Experimental results on the Plant Village dataset indicate that our model has improved the classification of 38 plant diseases. Based on metrics, the three-model ensemble performed better than the two-model ensemble. The best accuracy results of the ensemble MobileNetV2-DenseNet121 and MobileNetV2-Xception-DenseNet121 models are 99.49% and 99.56%, respectively. In addition, these models are also better than the state-of-the-art models and previous feature fusion techniques we proposed in LEMOXINET. Based on these results, the ensemble technique improved the detection performance, and it is expected to be applied to real-world conditions and can be a reference to be developed further in future research.

1 Introduction

Indonesia is one of the largest agricultural countries in the world. Its agricultural production is highly dependent on tropical weather conditions. Plant disease is one of the determining factors for production success. Apart from weather and natural disasters, these factors are easier to control. Farmers and experts can detect diseases in their plants. They can detect disease by visual observation of the leaves of their plants. However, because of their vast

* Corresponding author: sutaji.deni@umg.ac.id

farming area, this method will be very time-consuming and costly. Therefore, an automatic detection system is needed to help them detect disease. Several studies have been conducted to detect this automatic disease. Deep learning is an artificial intelligence system with a promising ability to do this. Convolutional Neural Networks (CNN) is a widely developed approach for intelligent computing tasks with image objects [1–8]

Recently, [9] introduced a comparison of 10 pre-trained CNN architectures in detecting crop disease. They reported that DenseNet121 with 98.97% of accuracy. Other pre-trained models, such as MobileNetV2, obtained 98.95% accuracy with significantly high speed and lower model size. According to their research, the detection performance can be improved by considering the ensemble strategy. Several ensemble pre-trained CNN architectures have been proposed. Initially, peanut-leaf disease was automatically detected using ensemble ResNet50 and DenseNet121 that proposed by [10] exhibit the accuracy 97.59%. Then, the stacked ensemble was proposed by [10] obtained the best accuracy of 98.36%. Then, Turkoglu et al. in 2021 suggested ensemble AlexNet, ResNet18-101-50, GoogleNet, and DenseNet201 which combined SVM as classifier to detect the plant and pest disease. Despite providing huge parameter from the extreme ensemble, their best results achieve average accuracy of 97.56%. These three results exhibited lower performance than single CNN models that evaluated by [9]. Subsequently, a citrus-pests detection that introduced by [11] considered ensemble of AlexNet, VGG16, ResNet50, and InceptionResNetV2. The experimental results generate accuracy of 99.04%. Eventually, [12] introduced the LEMOXINET. They proposed feature fusion was conducted to predict more than 38 classes of plant diseases from the Plant Village and user-defined datasets. MobileNetV2 and Xception were performed as feature extraction backbones. The performance obtained 99.10% of accuracy in the Plant Village dataset. LEMOXINET improved the performance by about 1.8% from the individual model of MobileNetV2 and Xception. [12]

However, prior studies in automatic plant diseases in an ensemble approach have several limitations. The primary issue is that the models provide performance lower than a single approach, such as the proposed model by [10,13,14]. Remarkably, the LEMOXINET design fuses the extracted features that generated similar features. Moreover, the model size grows linearly based on the feature sizes. Furthermore, this model considered the balance weight of both architectures. Another way of ensemble technique is by fusing the trained model. This strategy can be optimized by considering the weight.

According to the limitations mentioned earlier, we suggested a new ensemble approach by fusing the trained pre-trained considered weight as a selector, which architecture is more prioritized to detect the plant disease with main contributions as follows:

- 1) A weighted ensemble of three CNN models (MobileNetV2-Xception-DenseNet121) improves the performance of plant disease detection. It outperforms the single state-of-the-art, ensemble pre-trained CNN models and our prior LEMOXINET design.
- 2) It provided a more stable model size than the LEMOXINET design with linear increases based on extracted features.
- 3) Found the optimal weight of each trained CNN model.

This study is designed as follows: Section 1 presents prior plant disease detection algorithms, reports the model limitations, and describes the aim and contribution of the suggested approach. Section 2 explains the materials that followed the methods of the offered model. Experimental results, discussions, and comparisons are demonstrated in section 3. Eventually, the conclusion of this research is generated in section 4.

2 Materials and Method

2.1 Dataset

This research used the Plant Village dataset. This dataset consists of 38 different plant disease classes from 54,305 images. The resolution is 256 x 256 pixels, and each image is in RGB color space. The more detailed species of this dataset are apple, blueberry, cherry, corn, grape, orange, pepper, raspberry, potato, pumpkin, peach, soybean, strawberry, and tomato. Then, the diseases are distributed around 14 species as mentioned earlier with details: 17 fungal, 4 bacterial, 2 viral, and the remaining 1 disease from mites. On the other hand, the healthy class originates from 12 plant species [15].

2.2 Data collection and preparation

This dataset was collected from Kaggle, which provides several public datasets [16]. Then, we prepared the images and considered four sequential tasks. Initially, we provided image resizing into 224 x 224 pixels, as suggested by common studies. Then, each class was distributed in the training, validation, and test set of 70:10:20. Subsequently, applied image normalization. This task aims to simplify the mathematical computation of pixels from a range of 0 to 255 to 0 and 1 values [17] Eventually, the image augmentation task was considered for the training set. The benefit is to address the overfitting of the trained model. The images are augmented by the configuration as follows: image rotation is set in 30 degrees, width shift range is set in 0.3 value, height shift range is set in 0.3 value, shear range is set in 0.3 value, zoom range is set in 0.3 value, and the horizontal and vertical flip is set in True.

2.3 Pre-trained CNN models

In recent study, we considered the 10 pre-trained CNN models was used in Sutaji & Harun work, such as AlexNet, DenseNet121, GoogleNet, InceptionV3, InceptionResnetV2, MobileNetV2, ResNet50V2, VGG16, VGG19, and Xception. These models were applied transfer learning to achieve more faster than train these models from the scratch. Due to the weight of parameters from previous training in ImageNet dataset are adopted. At the same time, the last few layers of each model are allowed to learn to update the weights for the feature extraction process. In this way, high-dimensional features can be extracted more significantly. The impact is that it can improves the performance of softmax classifier in performing classification.

2.4 Ensemble CNN models

The ensemble CNN models are divided into two groups: ensemble two CNN models and three CNN models. According to our previous work, MobileNetV2 is the main model combined with other models. The primary ensemble of the two models is MobileNetV2 and Xception. This ensemble had done in feature fusion approach in the LEMOXINET model and provided excellent performance than others. Then, we considered the three-ensemble model from the prior primary ensemble by adding AlexNet, DenseNet121, GoogleNet, ResNetV2, InceptionV3, InceptionResnetV2, VGG16, and VGG19.

2.5 Training ensemble CNN models

During the training, there are several considered hyperparameters to be determined. We set the batch size to 25 and 18 for the training and validation set, respectively. Adam was set as the optimizer with a steady learning rate of $10 \cdot 10^{-4}$. The limited epoch is set to 100 with an early stop rule maximum of 10 times based on stagnant validation set accuracy.

Our experiments were implemented on a Google Colabs Pro with a Pascal P100 16GB GPU, RAM 12.7 GB, and dual virtual CPUs. In addition, for implementing all models, a Python 3.9 version was employed for language programming, along with the Tensorflow and Keras frameworks, Numpy, Pandas, Matplotlib, Seaborn, and Scikit-Learn libraries. The trained single models were saved as files with the *.h5 extension.

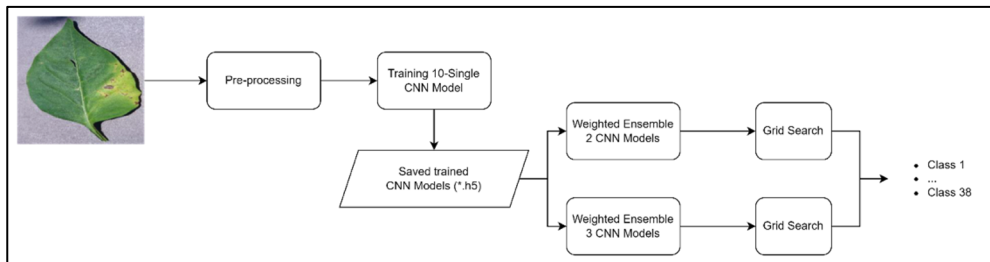


Fig. 1. Proposed weighted ensemble CNN models

The next step is to generate another two-ensemble and three-ensemble models from the saved models. Initially, all trained single models were loaded. Then, models were employed to predict plant disease using the test set. The weight of each model was searched using a grid search between 0.1 to 0.9 with an increasing value of 0.1. The completion of our proposed weighted ensemble model is illustrated in **Figure 1**.

2.6 Evaluation metrics

In order to evaluate the proposed method's performance, we considered accuracy, precision, recall, and f1-score value. There are four fundamental units, such as True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN), that are mathematically calculated as follows:

$$\text{Accuracy} = \frac{TP+TN}{TP + TN + FP + FN} \quad (1)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

$$\text{F1 - score} = 2 \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

3 Result and discussion

3.1 Weighted ensemble two-model CNNs

All models have obtained accuracy performance above 99%. The best model is ensemble MobileNetV2-DenseNet121 which generates accuracy, precision, recall, and f1-score value of 99.49%, 99.39%, 99.12%, and 99.25%, respectively. However, ensemble MobileNetV2 and Xception provide the highest improved accuracy, but the accuracy is under the ensemble

MobileNetV2 and DenseNet121, which reach only 99.32%. On the other hand, when comparing our recent study to the LEMOXINET, the recent study outperformed. It indicates that ensemble model CNN is better than feature concatenation in detecting plant disease.

3.2 Weighted ensemble three-model CNNs

The significantly improved results were reached by performing ensemble three-model CNNs. The proposed model achieved 99.56% from the ensemble of MobileNetV2-Xception-DenseNet121. Absolutely this performance is better than LEMOXINET and the ensemble two-model CNNs. This result can be obtained because each single model has different feature extraction characteristics. Thus, the features obtained from the three complement each other. Therefore, it makes it easier for Softmax to determine the class of plant disease detection. More details about the ensemble three-model CNNs are shown in **Table 2**.

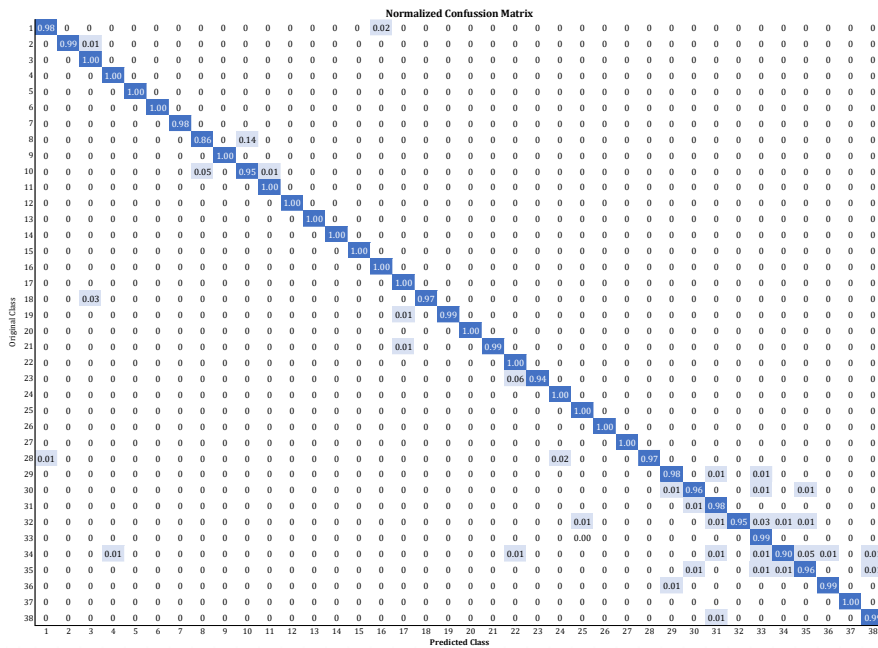


Figure 2. Normalized confusion matrix of proposed model WECNN-PDP

Table 1. Performance results on two-model ensemble CNNs

Models	Original Ensemble (%)				Weighted Ensemble (%)			Acc Improved
	acc	prec	rec	f1-score	w1	w2	acc	
MobileNet-Xception	99.20	98.99	98.68	98.82	0.7	0.6	99.32	0.12
MobileNet-DenseNet121	99.49	99.39	99.12	99.25	0.1	0.1	99.49	0
MobileNet-InceptionV3	99.43	99.30	99.15	99.22	0.1	0.1	99.43	0
MobileNet-ResNet50V2	99.40	99.19	99.00	99.08	0.1	0.1	99.40	0
MobileNet-VGG16	99.25	99.14	98.61	98.86	0.1	0.1	99.25	0
MobileNet-VGG19	99.17	98.90	98.73	98.81	0.8	0.7	99.19	0.02

Table 2. Performance results on three-model ensemble CNNs

Models	Original Ensemble (%)				Weighted Ensemble (%)				Acc Improved
	acc	prec	rec	f1-score	w1	w2	w3	acc	
MobileNet-Xception-DenseNet121	99.50	99.39	99.09	99.23	0.6	0.4	0.7	99.56	0.06
MobileNet-Xception-InceptionV3	99.45	99.26	99.12	99.19	0.6	0.4	0.3	99.54	0.09
MobileNet-Xception-ResNet50V2	99.49	99.35	99.21	99.20	0.7	0.5	0.4	99.52	0.03

Table 3. Comparison results against prior proposed models

Models	accuracy	precision	recall	f1-score
MobileNetV2 (Barman et al., 2020)	95.27	95.33	95.27	95.23
MobileNetV1 (Gandhi et al., 2018)	92.00	-	-	-
AlexNet (Mohanty et al., 2016)	99.35	99.35	99.35	99.34
DenseNets (Too et al., 2019)	99.75	-	-	-
Deep CNN (Geetharamani et al., 2019)	96.46	96.47	99.89	-
EfficientNet (Atila et al., 2021)	99.91	98.42	98.31	-
MobileNetV2 (Chen et al., 2021)	99.71	-	98.56	98.56
LEMOXINET (Sutaji & Yıldız, 2022)	99.10	99.10	99.02	99.03
Ensemble weighted MobileNet, Densenet121, DenseNet201 (Vallabhajosyula et al., 2021)	100	-	-	-
Proposed WECNN-PDP	99.56	99.39	99.09	99.23


3.3 Performance comparison with prior proposed models

To evaluate this recent study's performance, we compare it with previously proposed models that utilized the complete Plant Village dataset both on single and ensemble design—generally, our proposed obtained comparable results against the previous research. **Table 3** represent that the recent study has more suitable performance than other five prior proposed models, such as [12,15,18,19]. However, the model performance is inferior [20–22]. Nevertheless, their proposed models were tested in a poor test set of only 3% of the dataset. In addition, our recent study inferior than model proposed by [23] obtained perfect accuracy of 100%. However, their model generated higher parameters and model size than ours.

3.4 comparison results for some images

This subsection provides a sanity check of the plant disease detection from the proposed model against previously proposed models. We can evaluate the performance based on visual observation described in **Figure 3**. In the recent study, misdetection is fewer than LEMOXINET. It failed to detect the healthy tomato class into tomato spider mites, as illustrated in **Figure 3. (a)**. Based on the visual observation, the two images are similar with contain shadows and spots.

Moreover, according to **Figure. 2**, the recent study experienced confusion distinguishing several images within the same species but with different diseases. In addition, a recent study generates some misdetections of tomato spider mites into tomato target spots, tomato yellow curs virus, and healthy tomato. In addition, the corn cercospora was incorrectly classified as the corn northern blight disease. Fortunately, every disease that is detected as healthy is only 0.01%. It indicates that the disease was 99.99% successfully detected. Another drawback of this model is inaccurate predictions of plant species. For example, the original species is a strawberry but predicted as a raspberry. Furthermore, the prediction result is a peach bacterial spot, but the original species is apple scab. Fortunately, this condition occurs less than equal to 0.02%.



Ground Truth	8	0	1	2	3
Proposed	2	0	1	2	3
LEMOXINET	6	0	5	2	7
	(a)	(b)	(c)	(d)	(e)

Figure. 3. Comparison of prediction results LEMOXINET and WECNN-PDP model for five random images: (0) Strawberry Healthy; (1) Apple Cedar Rust; (2) Tomato Spider Mites; (3) Corn Cercospora; (4) Tomato Target Spot Disease; (5) Apple Black Rot; (6) Tomato Leaf Mold; (7) Corn Blight; (8) Tomato Healthy; (9) Corn Rust.

4. Conclusion

The proposed weighted ensemble of MobileNetV2, Xception, and DenseNet121 improved plant disease detection. The experimental results exhibited the superiority of our proposed model. Compared with the state-of-the-art and prior ensemble approach, particularly our previous study LEMOXINET model, our recent model exhibited better results in all metric performances, such as accuracy, precision, recall, and f1-score of 99.56%, 99.39%, 99.09%, and 99.23%, respectively. As well as this study had improved the accuracy performance from 99.10% (LEMOXINET) to 99.56%. However, considering grid search in determining the optimal weight of the ensemble model is time-consuming when the number of weight ranges is vast. Moreover, the imbalanced dataset is an ongoing issue that needs to be solved to achieve better performance.

There are open issues for future study from the limitation of our proposed model. They are first determining the optimal weight of the ensemble model to provide more resource and time consumption efficiency. Last, how to handle the imbalance dataset issue. Our suggestions are utilizing the meta-heuristic algorithms to address determining the optimal weight issue and up-sample or down-sample dataset to overcome the imbalance dataset issue.

5. Acknowledgment

Thank you to Lembaga Penelitian dan Pengabdian Masyarakat. (LPPM) University of Muhammadiyah Gresik (UMG). This research article supported by LPM UMG Nomor: 019/MoU.In/II.3.UMG/DPPM/F/2022 for Internal Research Funded.

6. 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 paper.

References

1. A. Abade, P. A. Ferreira, and F. de Barros Vidal, *Comput. Electron. Agric.* **185**, (2021)
2. Z. Iqbal, M. A. Khan, M. Sharif, J. H. Shah, M. H. ur Rehman, and K. Javed, *Comput. Electron. Agric.* **153**, 12 (2018)
3. S. Kaur, S. Pandey, and S. Goel, *Arch. Comput. Methods Eng.* **26**, 507 (2019)
4. J. Lu, L. Tan, and H. Jiang, *Agric.* **11**, 1 (2021)
5. R. Manavalan, *Comput. Electron. Agric.* **178**, 105802 (2020)
6. L. C. Ngugi, M. Abelwahab, and M. Abo-Zahhad, *Inf. Process. Agric.* **8**, 27 (2021)
7. S. Sachar and A. Kumar, *Expert Syst. Appl.* **167**, 114181 (2021)
8. V. Singh, N. Sharma, and S. Singh, *Artif. Intell. Agric.* **4**, 229 (2020)
9. D. Sutaji and H. Rosyid, *Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control* **4**, (2022)
10. Y. He, G. Zhang, and Q. Gao, 1 (2024)
11. M. Khanramaki, E. Askari Asli-Ardeh, and E. Kozegar, *Comput. Electron. Agric.* **186**, 106192 (2021)
12. O. Sutaji, D., & Yıldız, *Ecol. Inform.* **70**, (2022)
13. H. Qi, Y. Liang, Q. Ding, and J. Zou, *Appl. Sci.* **11**, 1 (2021)
14. M. Turkoglu, B. Yanikoğlu, and D. Hanbay, *Signal, Image Video Process.* **16**, 301 (2022)
15. S. P. Mohanty, D. P. Hughes, and M. Salathé, *Front. Plant Sci.* **7**, 1 (2016)
16. A. ALI, *Ataset Dis. Plant Leaf Images Corresp. Labels* (n.d.)
17. E. Prasetyo, N. Suciati, and C. Fatichah, *J. King Saud Univ. - Comput. Inf. Sci.* **34**, 5286 (2022)
18. U. Barman, R. D. Choudhury, D. Sahu, and G. G. Barman, *Comput. Electron. Agric.* **177**, 105661 (2020)
19. Institute of Electrical and Electronics Engineers, 2018 IEEE Int. Conf. Innov. Res. Dev. 1 (2018)
20. Ü. Atila, M. Uçar, K. Akyol, and E. Uçar, *Ecol. Inform.* **61**, 101182 (2021)
21. E. C. Too, L. Yujian, S. Njuki, and L. Yingchun, *Comput. Electron. Agric.* **161**, 272 (2019)
22. A. Chen, J., Zhang, D., Suzauddola, M., & Zeb, *Appl. Soft Comput.* 113 (2021)
23. S. Vallabhajosyula, V. Sistla, and V. K. K. Kolli, *J. Plant Dis. Prot.* **129**, 545 (2022)