



Enhancing Weighted Averaging for CNN Model Ensemble in Plant Diseases Image Classification

Octavian¹, Ahmad Badruzzaman², Muhammand Yusuf Ridho³, Bayu Distiawan Trisedya⁴

^{1,2,3,4}Faculty of Computer Sciences, University of Indonesia, Depok, Indonesia

¹octavian@ui.ac.id, ²ahmad.badruzzaman@ui.ac.id, ³muhammad.yusuf25@ui.ac.id, ⁴b.distawan@cs.ui.ac.id

Abstract

Deep learning, especially convolutional neural networks (CNN), has gained traction in the field of image classification. In the specific case of plant disease classification, improving the accuracy and reliability of image classification is paramount. This paper delves into the ensemble prediction technique using a weighted soft-voting method. Instead of assigning a generalized weight to each CNN model, our approach emphasizes giving weights to each label's prediction within every individual model. We employed three esteemed CNN architectures for our experiments: DenseNet201, InceptionV3, and Xception focusing on classifying various diseases affecting grapes. By harnessing transfer learning coupled with end-to-end fine-tuning, we achieved a streamlined and efficient training process. Notably, the f1-score for each grape disease class was used as a parameter for weight determination and as a metric for the final evaluation. In our study, the newly proposed method was tested across various datasets and ensemble scenarios, demonstrating its effectiveness by not only outperforming the conventional soft-voting and prevalent weighted soft-voting methods, which achieved best scores of 95.68% and 95.81% respectively, but also by achieving a remarkable accuracy of 96.56%. This method's efficacy is heightened when ensemble models exhibit distinct characteristics; the more varied the model characteristics, the more enhanced the ensemble results.

Keywords: ensemble learning; weighted averaging voting; convolutional neural network; image classification; plant diseases

How to Cite: O. Octavian, A. Badruzzaman, Muhammand Yusuf Ridho, and B. D. Trisedya, "Enhancing Weighted Averaging for CNN Model Ensemble in Plant Diseases Image Classification", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 8, no. 2, pp. 272 - 279, Apr. 2024.

DOI: <https://doi.org/10.29207/resti.v8i2.5669>

1. Introduction

Plant diseases have long been a major threat to global agriculture, impacting food security and economic stability. Over time, the methods used for diagnosing these diseases have undergone significant transformations. Initially, basic tools such as magnifying glasses and simple observational techniques were prevalent [1]. However, with the increasing demand for more efficient and scalable diagnostic methods, a shift toward advanced computational techniques has become necessary. This change is further driven by the growing global population and the corresponding rise in food demand, emphasizing the need for rapid and precise diagnosis of plant diseases [2].

Deep learning, an advanced branch of machine learning, is swiftly emerging as a cornerstone in the field of modern plant pathology. This technique, inspired by the human brain's structure and powered by

artificial neural networks, offers extraordinary computational capabilities. Deep learning algorithms are adept at processing extensive numbers of plant images, detecting intricate patterns that are often overlooked by the human eye. They excel not only in identifying plant diseases but also in comprehending the progression and potential impacts of these diseases [3]. With the evolving complexity of plant pathogens, the focus is increasingly shifting from mere disease identification to understanding their entire trajectory and devising effective intervention strategies. Traditional diagnostic methods, largely reliant on human expertise, struggle with the unpredictable nature of disease outbreaks. In stark contrast, deep learning leverages extensive datasets to continuously refine its analysis, emerging as a crucial tool capable of anticipating disease outbreaks. This predictive advantage offers agriculturists and farmers the opportunity for timely interventions, thereby potentially reducing the adverse effects on crop yields. With the growing recognition of its capabilities, deep learning is

steadily being acknowledged by researchers and industry professionals alike as a key to proactive management in the realm of plant disease control [4].

The application of deep learning in plant disease classification is not a nascent concept, and several pioneering works have underscored its efficacy. Zhang et al. [5] embarked on an exploration using convolutional neural networks (CNNs) to classify Maize leaf diseases, and their results indicated a remarkable 98.9% accuracy. Another noteworthy study, by using deep learning methods, Singh et al [6] propose a multilayer convolutional neural network (MCNN) to accurately classify Anthracnose fungal disease in mango leaves, with results outperforming other state-of-the-art approaches based on a dataset of 1070 images. Ghosal S et al [7] researched transfer learning, repurposing pre-trained models for rice plant disease classification, and achieved a reduction in training time without compromising on accuracy. In more recent research, Ho et al [8] and Yuvalatha et al [9] are using ensemble CNN techniques, both studies leveraged various transfer learning models, including ResNet and DenseNet architectures, to achieve high accuracies in identifying plant diseases early, with results showing promise for enhancing agricultural practices and reducing crop losses.

The application of deep learning in plant disease classification is well-established, with several studies demonstrating its efficacy. Pioneering works in this area have included the use of convolutional neural networks (CNNs) for classifying various plant diseases, showcasing remarkable accuracies and outperforming other state-of-the-art approaches and traditional methods. Ensemble learning, particularly methods like soft-voting and its improved variant, weighted soft-voting, has further enhanced model performance [10] – [20].

This research contributes to the field of plant disease classification by addressing a critical gap in the application of ensemble learning methodologies. Existing literature predominantly focuses on the effectiveness of ensemble methods like soft-voting and its more sophisticated variant, weighted soft-voting. However, these methods typically rely on the assumption that a model's performance is consistent across all classes, an assumption that this study challenges. We propose that a model may exhibit suboptimal performance in most classes but excel in specific ones. To capitalize on this insight, our research refines the weighted soft-voting approach by assigning unique weights to each label within a model, rather than applying a uniform weight across the entire model. This strategy aims to significantly improve the precision and accuracy of plant disease classification by harnessing the distinct strengths of individual models in their areas of competence. By adopting this more granular approach to weight distribution, we anticipate not only enhanced predictive performance in complex classification scenarios but also the establishment of a

new benchmark in the utilization of ensemble learning techniques.

In the field of plant disease classification, ensemble learning has become a pivotal approach, harnessing the collective strengths of multiple models to enhance prediction accuracy and robustness [21]. Among these methods, soft-voting and its advanced form, weighted soft-voting, stand out for their effectiveness in complex classification tasks. These techniques integrate diverse model predictions, adjusting the influence of each based on its demonstrated accuracy. However, this approach often overlooks the varying performance of models across different classes. This research aims to refine this aspect by introducing a novel adaptation to weighted soft-voting, assigning individual weights to each label of a model, thereby enhancing precision in plant disease classification.

Ensemble learning is increasingly favoured in machine learning due to its ability to leverage the strengths of multiple models, thereby enhancing overall performance. This approach is particularly effective in complex tasks where a single model might not capture all aspects of the data or may be prone to overfitting. By combining predictions from multiple models, ensemble methods often achieve higher accuracy and robustness compared to individual models. This makes them especially suitable for applications in fields like plant disease classification, where the accuracy and reliability of predictions can have significant practical implications

In the broader scope of ensemble methods, soft voting stands out as a popular technique. Soft-voting is an ensemble method that combines the predictions of multiple models by averaging each prediction score and using the maximum score as the label output [22]. Studies have shown the simplicity and effectiveness of the soft-voting method in improving the performance of machine-learning models [10], [18].

As advancements in ensemble techniques continued, soft-voting underwent refinements to further enhance its classification results. Recognizing that different models exhibit varying levels of performance, some being strong and others weak, it became logical to assign more influence to stronger models in the voting process. This led to the development of weighted soft-voting, where each prediction from a model is given a weight based on its prior prediction accuracy [23]. Research employing this method has demonstrated improved performance across various applications [12], [13], [15], [17], [19], [20].

In prior research, the application of weighted soft-voting for plant disease classification has been investigated, with weights being assigned at the model level [24], [25]. These studies, however, are based on the critical assumption that a model's performance is uniformly effective across all classifications, a presumption that may not always hold. This conventional approach tends to overlook instances

where a model, despite its overall average performance, could exhibit exceptional accuracy in specific classes. Our study addresses this oversight by introducing a refined version of the weighted soft-voting method, wherein weights are allocated to each label within a model independently. This modification aims to enhance the granularity and accuracy of classification outcomes, particularly in complex scenarios where the strengths and weaknesses of models vary markedly across different tasks. By adopting this tailored approach, our research endeavours to bring a more sophisticated and precise dimension to the field of plant disease classification using ensemble learning techniques.

In the initial phase of our research, we embarked on a preliminary experiment to assess the individual performance of various models in classifying specific labels within a grape disease dataset, the detail can be seen in Figure 3, in section 3. This was crucial to understanding the unique strengths and weaknesses of each model. Ensuring data variation and balance is crucial in this phase. Balance is achieved by equalizing data across all labels to prevent f1-score bias while varying the data characteristics enhances the robustness of the f1-score.

Table 1. Accuracy and f1-score of each model for preliminary experiment testing

Label	InceptionV3	DensNet201	Xception
0	0.90307	0.74425	0.61842
1	0.90821	0.81169	0.76364
2	0.82022	0.98246	0.94737
3	0.86242	0.99762	0.98361
Accuracy	0.87500	0.88690	0.83690

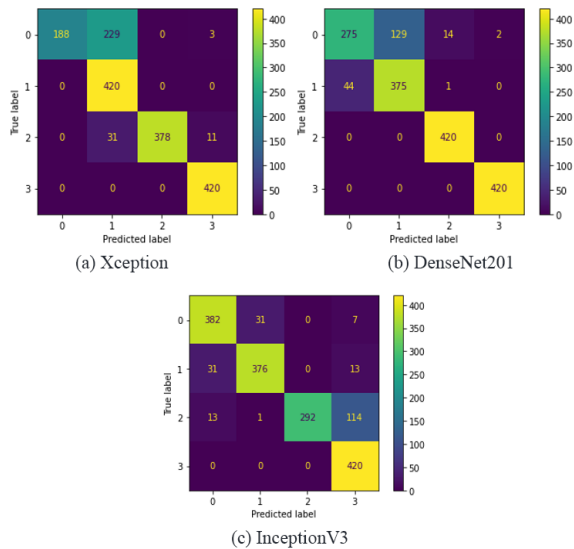


Figure 1. Confusion Matrix of each model for preliminary experiment testing

The results from this Preliminary Experiment can be seen in Table 1 and Figure 1. Models such as DenseNet201 and Xception displayed a propensity for accurately classifying labels 2 and 3, but their performance was less effective for labels 0 and 1. Interestingly, despite their similarities, DenseNet201

generally outperformed Xception. On the other hand, Inception-V3 showed a contrasting strength, being more adept at classifying labels 0 and 1, compared to 3 and 4.

These findings underscored the importance of a nuanced approach in the subsequent phase of the research. By assigning weights based on the demonstrated strengths of each model in classifying specific labels of grape diseases, we anticipate an enhancement in the overall accuracy of the ensemble. The next step involves the application of these insights to compute weight values for each model label, as outlined in Formula 5. This step is a key part of enhancing our ensemble method for the main experiment discussed in section 3.

2. Research Methods

2.1 Soft-voting and Weighted Soft-voting

Soft-voting distinguishes itself as a key technique in ensemble methods. This approach averages the prediction scores from multiple models, using the highest score for the final classification. Further refinement led to the development of weighted soft-voting, where predictions are weighted based on a model's accuracy, thus improving the method's precision in various applications.

Let h_i be the ensemble score result for label- i , T be the count of models, $p_{m,i}$ be the prediction score from model- m for label- i , and w_m be the weight for prediction of model- m then to calculate the ensemble prediction of the CNN model for image classification, the normal *soft-voting* formula can be written as Formula 1, and for the *weighted soft-voting* as Formula 2.

$$h_i = \frac{1}{T} \sum_{m=1}^T P_{m,i} \quad (1)$$

Based on [20], let a_n be the accuracy of model- n and w_m be the weight for model- m ; the weight of the model can be calculated by using Formula 2.

$$w_m = \frac{a_m}{\sum a_i} \quad (2)$$

2.2 Proposed Method

The proposed method in this paper is that, rather than giving weight to the models like Formula 2, giving weight to labels inside the models will give a chance to make the ensemble result more robust and better.

The architecture proposed methods in Figure 2 can explain Formula 3 which is the formula to calculate the ensemble prediction score by our proposed method. Where h_i is the ensemble score result for label- i , T is the count of models, $p_{m,i}$ is the prediction score from model- m for label- i , and $w_{m,i}$ is the weight for prediction of model- m for label- i .

$$h_i = \frac{\sum_{m=1}^T (w_{m,i} p_{m,i})}{\sum_{m=1}^T w_{m,i}} \quad (3)$$

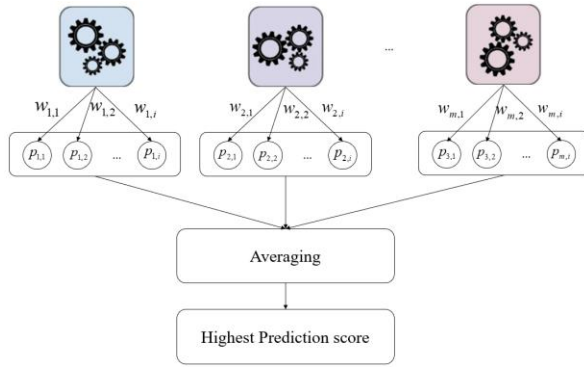


Figure 2. The proposed method of modifying the weighted soft-voting

Additionally, instead of using the accuracy score for calculating the weight of each label, using the f1-score is a better choice, since the f1-score is a metric that can describe the performance of the model in predicting each label or class. The formula can be written as Formula 4 where $w_{m,i}$ be the weight of prediction for model- m for label- i , $f_{m,i}$ be the f1-score of model- m for label- i .

$$w_{m,i} = \frac{f_{m,i}}{\sum f_{n,i}} \quad (4)$$

2.3 CNN Architectures

A convolutional neural network (or CNN) is a special type of multilayer neural network or deep learning architecture inspired by the visual system of living beings [26]. There are so many advanced architectures of CNN that have been discovered in recent years. In this paper, the experiment uses three CNN architectures to be ensemble. Those architectures are DenseNet201, InceptionV3, and Xception.

DenseNet201 is one of the architectures in the DenseNet family, which has 201 layers. The key idea behind DenseNet is to connect all layers in a feed-forward fashion so that each layer receives the feature maps from all preceding layers as input. This creates a dense block, where the input and output of each layer are concatenated together. By doing this, DenseNet can reuse features learned at earlier layers, leading to better feature reuse and higher accuracy while also reducing the chance of gradient vanishing and the number of parameters [26]. DenseNet has been used in varying image classifications, such as in articles [27], [28] and [29].

InceptionV3 is a superior version of the basic model of the Inception family. The InceptionV3 model is made up of 42 layers, which is a bit more than the previous versions (the Inception V1 and V2 models). InceptionV3 uses a combination of 1x1, 3x3, and 5x5 convolutions, as well as max pooling and average pooling, to extract features from images. It also introduces the inception module, which consists of multiple parallel convolutional branches with different filter sizes. By doing this, Inception-V3 can capture both local and global features in an image. In addition,

Inception-v3 factorized convolution to reduce the number of parameters in the network. This involves decomposing a standard convolution into two separate convolutions with smaller filter sizes, which reduces the number of parameters without sacrificing accuracy. Those make the efficiency of the inceptionV3 to be impressive [30]. Some of the research using this model can be read in articles [31] and [32].

The last CNN architecture is Xception. It is based on the Inception architecture, but it replaces the standard Inception modules with depthwise separable convolutions. Depthwise separable convolutions consist of two separate operations: a depthwise convolution and a pointwise convolution. The depthwise convolution applies a single filter to each input channel, while the pointwise convolution applies a 1x1 filter to combine the output of the depthwise convolution across all channels. By doing this, Xception can capture complex patterns in the input data using fewer parameters. In addition to depthwise separable convolutions, Xception also uses a series of residual connections, which help alleviate the problem of vanishing gradients during training. These connections allow gradients to flow more easily through the network, which can lead to faster convergence and better performance. The advantage of Xception is how easy the architecture is to implement in code and reduce the computer cost [33]. Xception has been used for several classification cases, such as image weather classification [34] and plant diseases classification [35].

2.4 Evaluation Models

There are several metrics to measure how well the models or methods perform. In this experiment, the metrics that will be used are *recall*, *precision*, *f1-score*, *accuracy* and the *confusion matrix*. The formula to calculate all of those metrics can be seen in Formulas 5, 6, 7, and 8. where TP is the number of true positives (the model predicts a positive result and the ground truth is also positive), TN is the number of true negatives (the model predicts a negative result and the ground truth is also negative), FP is the number of false positives (the model predicts a positive result, but the ground truth is negative), and FN is the number of false negatives (the model predicts a negative result, but the ground truth is positive).

$$recall = \frac{TP}{TP+FN} \quad (5)$$

$$precision = \frac{TP}{TP+FP} \quad (6)$$

$$f1 - score = \frac{2 \times precision \times recall}{precision + recall} \quad (7)$$

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (8)$$

3. Results and Discussions

3.1 Dataset

In this study, we utilized a dataset comprising classification images of grape leaf diseases, provided by

PlantVillage [36], [37]. The focus is on four primary classes of grape disease: Black Rot (1,180 images), Esca (1,382 images), Leaf Blight (1,076 images), and Healthy (423 images). To create a robust dataset for machine learning analysis, each class has been augmented to a total of 2,000 images. This standardization ensures uniformity across all classes.

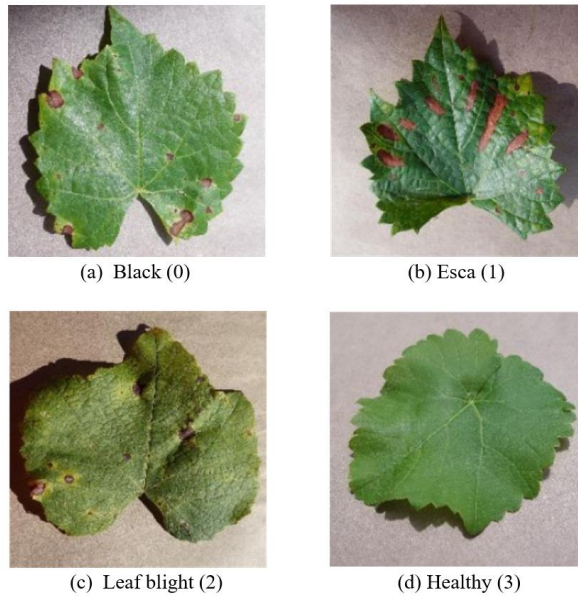


Figure 3. Sample images of grape diseases dataset and their labels

The augmented dataset is systematically divided into three key segments. Firstly, the training phase, which consists of 700 images for model training and 300 images for validation. Secondly, the Preliminary Experiment testing phase utilizes a subset of 500 images. Lastly, the Experiment testing phase also employs 500 images. For ease of analysis and clarity, each class within the dataset is distinctly labelled, as illustrated in Figure 3

3.2 Training Process

Our analysis of the training process, as illustrated in Figure 4 and Figure 5, reveals a stark contrast between models trained with and without pre-trained weights. The use of pre-trained models significantly accelerated the convergence speed, leading to faster stabilization in both accuracy and validation accuracy metrics. This suggests that the pre-trained models were able to leverage previously learned patterns, thereby reducing the time and computational resources needed to reach optimal performance.

Additionally, the figure highlights a more stable and consistent decline in both loss and validation loss for the pre-trained models compared to their counterparts. This stability indicates that pre-trained models are not only faster in reaching convergence but also more reliable in maintaining performance consistency throughout the training process. The combined insights from these figures underscore the efficacy of using pre-

trained models in enhancing the efficiency and robustness of the training phase

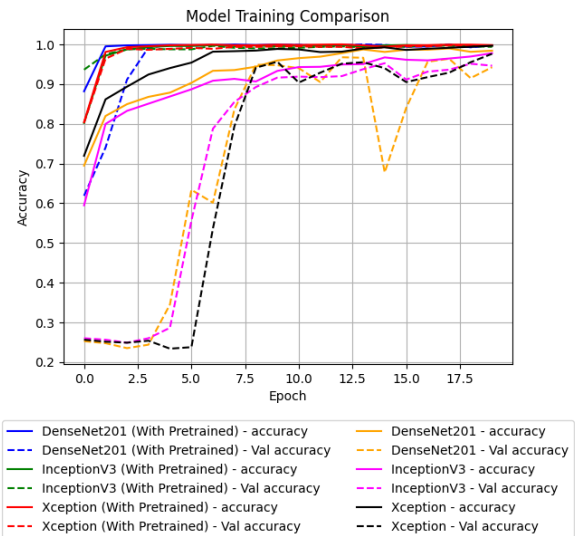


Figure 4. Comparison accuracy and validation accuracy of the training process between using and not using a pre-trained model

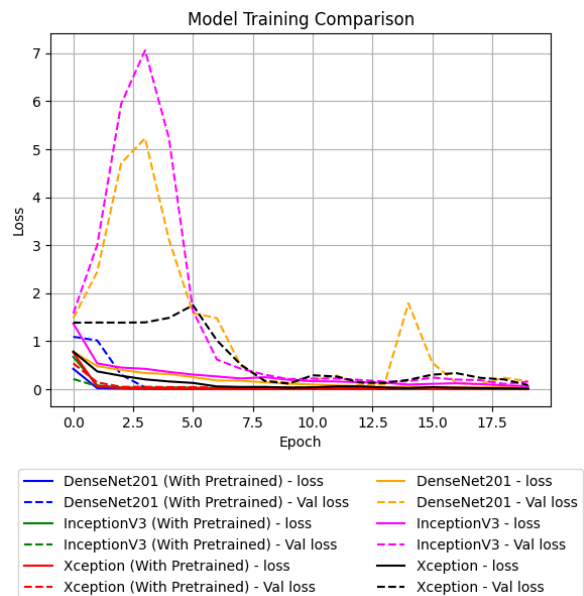


Figure 5. Comparison loss and validation loss of training process between using and not using the pre-trained model

3.3 Result

The outcomes of applying the proposed methods in the experiment testing are detailed in Tables 2-5, which present the classification report, and Figures 6-7, illustrating the confusion matrices. Table 4 and Figure 5 provide an initial assessment of each model's performance in the testing phase, revealing that all models yield results consistent with those observed in the Preliminary Experiment. This consistency validates the reliability of the Preliminary Experiment data for calculating the weight labels in our proposed method. A comprehensive summary of all ensemble testing experiments is documented in Tables 5-8 and Figure 6. These tests unequivocally demonstrate that our

proposed method, which involves assigning weights to each model's label, outperforms traditional soft-voting and weighted soft-voting methods. Notably, the highest accuracy score achieved is 0.96650, facilitated by the combination of Inception-V3 and DenseNet201, while the lowest is 0.90100, resulting from the DenseNet201 and Xception pairing.

One of the most intriguing aspects of our study is the differing outcomes produced by various ensemble combinations. Insights drawn from the Preliminary Experiment and the individual performance metrics of each model, as detailed in Table 1, Table 2, Figure 1, and Figure 6, reveal a notable distinction.

Table 2. Accuracy and f1-score of models in experiment testing

Label	InceptionV3	DensNet201	Xception
0	0.91360	0.77201	0.67550
1	0.90554	0.82803	0.77942
2	0.83759	0.98619	0.94848
3	0.86430	0.99900	0.98912
Accuracy	0.88100	0.89850	0.89850

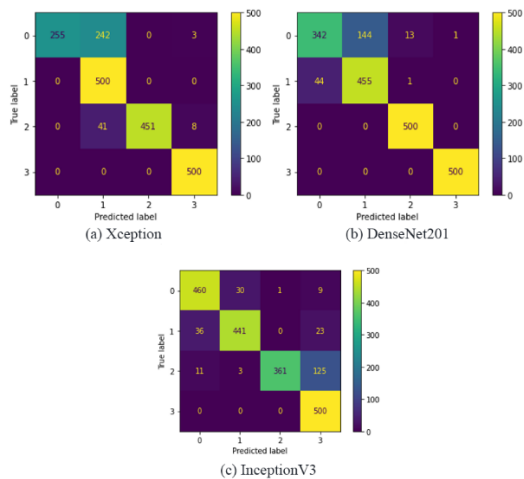


Figure 6. Confusion Matrix of each model for experiment testing

Table 3. Accuracy and f1-score of ensemble models using Xception and InceptionV3

Label	Soft-voting [10], [18]	weighted soft-voting[20]	Proposed Method
0	0.88372	0.90247	0.91536
1	0.89748	0.90909	0.91958
2	0.93276	0.91658	0.94737
3	0.95420	0.93809	0.96525
Accuracy	0.91750	0.91700	0.93700

DenseNet201 and Xception, while sharing similar characteristics, differ fundamentally from Inception-V3. This similarity between DenseNet201 and Xception implies a limitation in their ability to augment each other's performance, as they are prone to similar weaknesses. In contrast, when combined with Inception-V3, which exhibits a different set of characteristics, there is a complementary interaction. This synergy allows for the mutual compensation of weaknesses, where the strengths of one model effectively counterbalance the shortcomings of the other. Such findings highlight the critical importance of

model diversity in constructing effective ensemble approaches, emphasizing how distinct model characteristics can lead to a more robust overall system.

Table 4. Accuracy and f1-score of ensemble models using Xception and DenseNet201

Label	Soft-voting	weighted soft-voting	Proposed Method
0	0.74750	0.74938	0.75921
1	0.83236	0.83221	0.83615
2	0.99800	0.99800	0.99700
3	0.99701	0.99800	0.99701
Accuracy	0.89800	0.89850	0.90100

Table 5. Accuracy and f1-score of ensemble models using InceptionV3 and DenseNet201

Label	Soft-voting	weighted soft-voting	Proposed Method
0	0.92981	0.92762	0.93333
1	0.93478	0.93399	0.93837
2	0.99398	0.99498	0.99700
3	0.99108	0.99305	0.99701
Accuracy	0.96250	0.96250	0.96650

Table 6. Accuracy and f1-score of Xception, InceptionV3 and DenseNet201 ensemble

Label	Soft-voting	weighted soft-voting	Proposed Method
0	0.87486	0.87982	0.89946
1	0.89536	0.89945	0.91179
2	0.99398	0.99498	0.99598
3	0.99800	0.99800	0.99800
Accuracy	0.94100	0.94350	0.95150

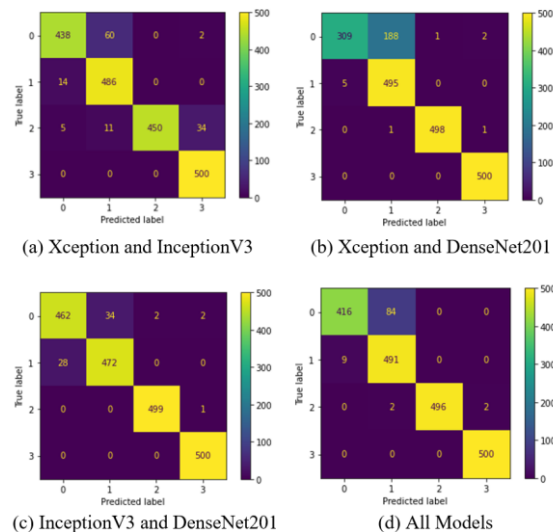


Figure 7. Confusion Matrix of the Proposed method from the Crossing Ensemble model

Closer analysis reveals the reason for the significant outperformance of the DenseNet201 and Inception-V3 ensemble, which achieves an accuracy of 0.96650, compared to the Xception and Inception-V3 combination which attains an accuracy of 0.93700. A detailed analysis of Tables 1 and 4 offers a compelling explanation. It becomes clear that DenseNet201 consistently surpasses Xception across a range of evaluation metrics. This difference in individual model performance becomes even more pronounced when all three models, DenseNet201, Inception-V3, and Xception, are integrated into a single ensemble,

achieving a collective accuracy of 0.95150. From this, we can deduce that Xception, while sharing similarities with DenseNet201, unfortunately, acts as a limiting factor in the ensemble configuration, thereby reducing the overall efficacy of the method. These findings highlight the crucial role of strategic model selection in ensemble techniques, underscoring the need for each component to enhance, rather than impede, the collective performance.

3.4 Additional Testing

The validation of our proposed method is extended by applying it to an alternative dataset [38], thereby examining its generalizability and consistency across different scenarios. This testing is crucial to ascertain the robustness of our ensemble approach, particularly when exposed to diverse data characteristics. The results, presented in Table 7, demonstrate a remarkable consistency in performance, mirroring the high accuracy and precision observed in the primary dataset. Specifically, the model combination of InceptionV3 and DenseNet201, which yielded the highest accuracy in the primary testing, continues to exhibit superior performance with an accuracy score of 0.96562 in the alternate dataset. Similarly, the combination of DenseNet201 and Xception maintains its lower, yet stable, accuracy score of 0.88687.

Table 7. Accuracy score of the proposed method in the alternative dataset

Ensemble Models	Soft-voting	weighted soft-voting	Proposed Method
Xception, Inceptionv3	0.90313	0.90500	0.92812
Xception, DenseNet201	0.88250	0.88687	0.88687
InceptionV3, DenseNet201	0.95688	0.95813	0.96562
Xception, InceptionV3, DenseNet201	0.92500	0.92812	0.93375

These findings suggest that the predictive capabilities of our model are not dataset-specific, but rather indicative of the inherent strength of the proposed ensemble approach. The consistent performance across varied datasets reinforces the validity of our initial hypothesis and underscores the potential of our refined weighted soft-voting method in diverse plant disease classification scenarios.

4. Conclusions

This research marks a significant advancement in the application of convolutional neural networks (CNNs) for the classification of plant disease images, particularly in grape leaves. By innovatively adapting the weighted soft-voting method to assign weights to specific label models, rather than uniformly across all labels, the study introduces a refined approach that considerably enhances classification accuracy. This novel method was rigorously tested on renowned CNN architectures such as Xception, DenseNet201, and Inception-V3. The results consistently demonstrated its

superiority, outperforming traditional ensemble strategies like normal soft-voting and weighted soft-voting. Among the combinations tested, the ensemble of DenseNet201 and Inception-V3 was particularly effective, delivering a commendable accuracy of 96.65% in identifying grape plant diseases.

The research underscores the significant influence of individual model characteristics on the ensemble's overall performance. A critical insight from this study is the synergistic effect observed when models with contrasting strengths are paired. For instance, the pairing of DenseNet201 and Inception-V3, which exhibit complementary characteristics, resulted in a significant boost in accuracy. This contrasts with combinations like DenseNet201 and Xception, where similarity in model characteristics did not produce the same level of efficacy. Moreover, the study also highlights the importance of model selection in ensemble methods. The analysis revealed that certain models, though similar, could act as performance bottlenecks when paired inappropriately. This observation emphasizes the need for careful consideration of individual model performances and their interactions within an ensemble.

Acknowledgements

The author declares no conflict of interest. This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

References

- [1] L. Li, S. Zhang, and B. Wang, "Plant Disease Detection and Classification by Deep Learning—A Review," *IEEE Access*, vol. 9, pp. 56683–56698, 2021, doi: 10.1109/ACCESS.2021.3069646.
- [2] I. Ul Haq and S. Ijaz, "History and Recent Trends in Plant Disease Control: An Overview," 2020, pp. 1–13. doi: 10.1007/978-3-030-35955-3_1.
- [3] Charu C. Aggarwal, *Neural Networks and Deep Learning: A Textbook*, 2nd edition. Cham, Switzerland: Springer International Publishing, 2023.
- [4] M. Albahar, "A Survey on Deep Learning and Its Impact on Agriculture: Challenges and Opportunities," *Agriculture*, vol. 13, no. 3, p. 540, Feb. 2023, doi: 10.3390/agriculture13030540.
- [5] X. Zhang, Y. Qiao, F. Meng, C. Fan, and M. Zhang, "Identification of Maize Leaf Diseases Using Improved Deep Convolutional Neural Networks," *IEEE Access*, vol. 6, pp. 30370–30377, 2018, doi: 10.1109/ACCESS.2018.2844405.
- [6] U. P. Singh, S. S. Chouhan, S. Jain, and S. Jain, "Multilayer Convolution Neural Network for the Classification of Mango Leaves Infected by Anthracnose Disease," *IEEE Access*, vol. 7, pp. 43721–43729, 2019, doi: 10.1109/ACCESS.2019.2907383.
- [7] S. Ghosal and K. Sarker, "Rice Leaf Diseases Classification Using CNN With Transfer Learning," in *2020 IEEE Calcutta Conference (CALCON)*, IEEE, Feb. 2020, pp. 230–236. doi: 10.1109/CALCON49167.2020.9106423.
- [8] S. Van Ho, H. G. Vuong, B. Q. Nguyen, Q.-H. Trinh, and M.-T. Tran, "Ensemble of Deep Neural Networks for Rice Leaf Disease Classification," in *2022 RIVF International Conference on Computing and Communication Technologies (RIVF)*, IEEE, Dec. 2022, pp. 238–243. doi: 10.1109/RIVF55975.2022.10013858.
- [9] S. Yuvalatha, J. Keerthika, S. Prabhavathy, M. Banupriya, and R. Priyadarshini, "Automated Plant Leaf Classification

- using Ensemble Transfer Learning in CNN model,” in *2022 IEEE North Karnataka Subsection Flagship International Conference (NKCon)*, IEEE, Nov. 2022, pp. 1–5. doi: 10.1109/NKCon56289.2022.10126722.
- [10] S. Chaudhary and Y. Qiang, “Ensemble deep learning method for Covid-19 detection via chest X-rays,” in *2021 Ethics and Explainability for Responsible Data Science (EE-RDS)*, IEEE, Oct. 2021, pp. 1–3. doi: 10.1109/EE-RDS53766.2021.9708581.
- [11] M. Patel, A. Das, V. K. Pant, and J. M., “Detection of Tuberculosis in Radiographs using Deep Learning-based Ensemble Methods,” in *2021 Smart Technologies, Communication and Robotics (STCR)*, IEEE, Oct. 2021, pp. 1–7. doi: 10.1109/STCR51658.2021.9588936.
- [12] F. A. Noor, I. Munzerin, A. M. A. Iqbal, T. Islam, and E. Hossain, “An ensemble learning based approach to autonomous COVID19 detection using transfer learning with the help of pre-trained Deep Neural Network models,” in *2021 24th International Conference on Computer and Information Technology (ICCIT)*, IEEE, Dec. 2021, pp. 1–6. doi: 10.1109/ICCIT54785.2021.9689825.
- [13] S. Cyriac, N. Raju, and Y.-W. Kim, “Pneumonia Detection using Ensemble Transfer Learning,” in *2022 13th International Conference on Information and Communication Technology Convergence (ICTC)*, IEEE, Oct. 2022, pp. 479–484. doi: 10.1109/ICTC55196.2022.9952532.
- [14] H. Zhao, Q. Liu, and Y. Yang, “Transfer Learning with Ensemble of Multiple Feature Representations,” in *2018 IEEE 16th International Conference on Software Engineering Research, Management and Applications (SERA)*, IEEE, Jun. 2018, pp. 54–61. doi: 10.1109/SERA.2018.8477189.
- [15] A. Acharya, A. Muvvala, S. Gawali, R. Dhopavkar, R. Kadam, and A. Harsola, “Plant Disease detection for paddy crop using Ensemble of CNNs,” in *2020 IEEE International Conference for Innovation in Technology (INOCON)*, IEEE, Nov. 2020, pp. 1–6. doi: 10.1109/INOCON50539.2020.9298295.
- [16] V. C. Osamor and A. F. Okezie, “Enhancing the weighted voting ensemble algorithm for tuberculosis predictive diagnosis,” *Sci Rep*, vol. 11, no. 1, p. 14806, Jul. 2021, doi: 10.1038/s41598-021-94347-6.
- [17] Md. S. H. Talukder and A. K. Sarkar, “Nutrients deficiency diagnosis of rice crop by weighted average ensemble learning,” *Smart Agricultural Technology*, vol. 4, p. 100155, Aug. 2023, doi: 10.1016/j.atech.2022.100155.
- [18] S. Misra, D. Kim, J. Kim, W. Shin, and C. Kim, “A voting-based ensemble feature network for semiconductor wafer defect classification,” *Sci Rep*, vol. 12, no. 1, p. 16254, Sep. 2022, doi: 10.1038/s41598-022-20630-9.
- [19] K. Chauhan, Kashish, K. Dagar, and R. K. Yadav, “Cataract detection from eye fundus image using an ensemble of transfer learning models,” in *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, IEEE, Apr. 2022, pp. 2194–2198. doi: 10.1109/ICACITE53722.2022.9823638.
- [20] V. C. Osamor and A. F. Okezie, “Enhancing the weighted voting ensemble algorithm for tuberculosis predictive diagnosis,” *Sci Rep*, vol. 11, no. 1, p. 14806, Jul. 2021, doi: 10.1038/s41598-021-94347-6.
- [21] H. K. Kondaveeti, K. G. Ujini, B. V. V. Pavankumar, B. S. Tarun, and S. C. Gopi, “Plant Disease Detection Using Ensemble Learning,” in *2023 2nd International Conference on Computational Systems and Communication (ICCS)*, IEEE, Mar. 2023, pp. 1–6. doi: 10.1109/ICCS56913.2023.10142982.
- [22] H. Wang, Y. Yang, H. Wang, and D. Chen, “Soft-Voting Clustering Ensemble,” 2013, pp. 307–318. doi: 10.1007/978-3-642-38067-9_27.
- [23] Z.-H. Zhou, *Ensemble Methods: Foundations and Algorithms*, 1st ed. Chapman & Hall/CRC, 2012.
- [24] E. Ayan, H. Erbay, and F. Varçin, “Crop pest classification with a genetic algorithm-based weighted ensemble of deep convolutional neural networks,” *Comput Electron Agric*, vol. 179, p. 105809, Dec. 2020, doi: 10.1016/j.compag.2020.105809.
- [25] A. M. P. and P. Reddy, “Ensemble of CNN models for classification of groundnut plant leaf disease detection,” *Smart Agricultural Technology*, vol. 6, p. 100362, Dec. 2023, doi: 10.1016/j.atech.2023.100362.
- [26] A. Ghosh, A. Sufian, F. Sultana, A. Chakrabarti, and D. De, “Fundamental Concepts of Convolutional Neural Network,” 2020, pp. 519–567. doi: 10.1007/978-3-030-32644-9_36.
- [27] S.-H. Wang and Y.-D. Zhang, “DenseNet-201-Based Deep Neural Network with Composite Learning Factor and Precomputation for Multiple Sclerosis Classification,” *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 16, no. 2s, pp. 1–19, Apr. 2020, doi: 10.1145/3341095.
- [28] M. K. Bohmrah and H. Kaur, “Classification of Covid-19 patients using efficient fine-tuned deep learning DenseNet model,” *Global Transitions Proceedings*, vol. 2, no. 2, pp. 476–483, Nov. 2021, doi: 10.1016/j.gltp.2021.08.003.
- [29] A. Shabani, L. Dharmo, and O. Zavalani, “Modelling Building Energy Systems using Electric Circuit Analogy,” *European Journal of Electrical Engineering and Computer Science*, vol. 7, no. 1, pp. 56–61, Jan. 2023, doi: 10.24018/ejece.2023.7.1.491.
- [30] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision,” Dec. 2015.
- [31] C. Wang *et al.*, “Pulmonary Image Classification Based on Inception-v3 Transfer Learning Model,” *IEEE Access*, vol. 7, pp. 146533–146541, 2019, doi: 10.1109/ACCESS.2019.2946000.
- [32] Y. Qian *et al.*, “Fresh Tea Leaves Classification Using Inception-V3,” in *2019 IEEE 2nd International Conference on Information Communication and Signal Processing (ICICSP)*, IEEE, Sep. 2019, pp. 415–419. doi: 10.1109/ICICSP48821.2019.8958529.
- [33] F. Chollet, “Xception: Deep Learning with Depthwise Separable Convolutions,” Oct. 2016.
- [34] N. An, “Xception Network for Weather Image Recognition Based on Transfer Learning,” in *2022 International Conference on Machine Learning and Intelligent Systems Engineering (MLISE)*, IEEE, Aug. 2022, pp. 330–333. doi: 10.1109/MLISE57402.2022.00072.
- [35] V. Menon, V. Ashwin, and R. K. Deepa, “Plant Disease Detection using CNN and Transfer Learning,” in *2021 International Conference on Communication, Control and Information Sciences (ICCIsc)*, IEEE, Jun. 2021, pp. 1–6. doi: 10.1109/ICCIsc52257.2021.9484957.
- [36] D. P. Hughes and M. Salathé, “An open access repository of images on plant health to enable the development of mobile disease diagnostics through machine learning and crowdsourcing,” *CoRR*, vol. abs/1511.08060, 2015, [Online]. Available: <http://arxiv.org/abs/1511.08060>
- [37] G. G. and A. P. J., “Identification of plant leaf diseases using a nine-layer deep convolutional neural network,” *Computers & Electrical Engineering*, vol. 76, pp. 323–338, Jun. 2019, doi: 10.1016/j.compeleceng.2019.04.011.
- [38] M. Khan, “Healthy and Disease affected Leaves of Grape Plant,” Jan. 2020, doi: 10.6084/m9.figshare.13083890.v1.