BEST
Journal of Applied Electrical & Science Technology – University of PGRI Adi Buana Surabaya
p-ISSN 2715-2871
e-ISSN 2714-5247

Integration of Concatenated Deep Learning Models with ResNet Backbone for Automated Corn Leaf Disease Identification

Achmad Imam Sudianto
Mechatronics Engineering
University Trunojoyo of Madura
Bangkalan, Indonesia
aimam.sudianto@trunojoyo.ac.id

Sigit Susanto Putro
Informatics Engineering
University Trunojoyo of Madura
Bangkalan, Indonesia
sigit.putro@trunojoyo.ac.id

Eka Mala Sari
Informatics Engineering
University Trunojoyo of Madura
Bangkalan, Indonesia
em_sari@trunojoyo.ac.id

Ika Oktavia Suzanti
Informatics Engineering
University Trunojoyo of Madura
Bangkalan, Indonesia
iosuzanti@trunojoyo.ac.id

Aeri Rachmad
Informatics Engineering
University Trunojoyo of Madura
Bangkalan, Indonesia
aery_r@trunojoyo.ac.id

Wildan Surya Wijaya
Electrical Engineering
University of PGRI Adibuana Surabaya
Surabaya, Indonesia
wildansurya@unipasby.ac.id

Abstract— **Corn is one of Indonesia's food commodities that serves as an alternative food source to support food diversification in Indonesia. However, leaf infections in corn plants often cause significant yield losses and threaten food security. Early detection of this disease is very important, especially for small farmers, because conventional diagnostic methods that rely on agronomists are expensive and time-consuming. Recent advances in Agricultural Artificial Intelligence (AI) and image processing have facilitated automatic plant disease recognition through Convolutional Neural Networks (CNN), with ResNet as the main backbone combined through concatenation with MobileNetV3, DenseNet161, and GoogleNet. The dataset consists of 4,000 images divided into 2,560 training data, 640 validation data, and 800 test data, with image sizes adjusted to 224×224 pixels. The images are divided into four categories: gray leaf spot, common rust, northern leaf blight, and healthy leaves. The testing was conducted using three different optimizers, namely Adam, RMSprop, and SGD, with a learning rate of 0.01. The experimental results show that the SGD optimizer provides the best performance with a loss value of 0.2275, accuracy of 0.9513, precision of 0.9536, recall of 0.9513, and F1 score of 0.9512. These findings prove that the combination of ResNet, MobileNetV3, DenseNet161, and GoogleNet architectures with the SGD optimizer can significantly improve the accuracy of corn leaf disease detection, making it a potential application for automatic detection systems in support of smart farming practices.**

Keywords— Digital Image Processing, Corn Leaf, Machine Learning, CNN

I. INTRODUCTION

Agriculture is a vital economic sector in many countries, including Indonesia, serving as the foundation of rural livelihoods and playing a strategic role in national development [1]. This sector provides employment for the majority of the rural workforce, thereby reducing unemployment and contributing to poverty alleviation. In addition, agriculture ensures food security and supports overall economic resilience. Among Indonesia's major agricultural commodities, corn holds an important position due to its wide use in food, animal feed, and industrial processing [2].

Processed derivatives, such as corn flour and corn oil, are increasing their economic value through downstream industries [3]. However, corn productivity is often limited by various leaf diseases that significantly affect yield and quality. Common infections such as leaf blight, leaf spot, and leaf rust are particularly damaging, as they can spread rapidly within fields and cause widespread damage if not detected and managed in a timely manner [4].

Early detection of leaf diseases in corn is very important to reduce their adverse effects on plant health and productivity [5]. Traditionally, disease identification has relied on agronomists assessing visual symptoms on leaves. While this manual approach is effective, it often requires significant labor, time, and cost, with results highly dependent on human expertise. Therefore, the development of more efficient, accurate, and automated diagnostic methods is needed to enable timely and accurate identification of corn leaf diseases [6].

Recent advances in artificial intelligence (AI) and image processing have significantly improved the development of automated plant disease detection systems [7]. One of the most effective approaches involves the application of convolutional neural networks (CNNs), a class of deep learning models specifically designed for visual pattern recognition and widely recognized for their success in image classification tasks [8]. Among CNN architectures, Residual Networks (ResNet) have gained considerable attention due to their ability to overcome the vanishing gradient problem through residual connections, thereby enabling more effective training of deeper networks. ResNet variants, such as ResNet-50 and ResNet-101, have been widely used in image classification tasks, including plant disease identification. Although ResNet provides strong feature representations, integration with other deep learning models has the potential to improve classification accuracy through the capture of richer spatial and semantic features [9].

However, its effectiveness is not only influenced by the depth of the network but also by the selection and optimization of the appropriate convolutional components. Previous studies [10] have reported significant progress in

p-ISSN 2715-2871
e-ISSN 2714-5247

the use of advanced deep learning architectures, such as CNN and ResNet, for plant disease identification, thereby improving diagnostic accuracy. However, deep neural networks are still vulnerable to challenges such as overfitting, especially when the data set is limited or lacks diversity. To overcome this problem, concatenated models are formed by combining various CNN architectures to improve vulnerability. MobileNetV3, with its lightweight and efficient structure, is suitable for applications that can be installed on mobile handheld devices [11]. Meanwhile, DenseNet161 has advantages in feature reuse and gradient propagation through dense connections [12]. The combination of the GoogleNet (Inception) architecture has multi-scale feature extraction, allowing the network to capture various patterns of disease symptoms on leaves [13]. By integrating these three models into the ResNet backbone, a hybrid framework can be built that overcomes the shortcomings of each. Three types of optimizers, Adam, RMSprop, and Stochastic Gradient Descent (SGD), were used to evaluate the training dynamics and final performance of the model.

This study has important implications for agriculture and artificial intelligence. By examining the influence of various layer configurations in the ResNet architecture on classification accuracy, experiments were conducted by modifying specific layers and evaluating the resulting models. Understanding the contribution of each convolutional component enables the optimization of ResNet for effective corn leaf disease classification. These advances can facilitate faster and more accurate diagnosis and treatment of leaf diseases, thereby increasing crop productivity and supporting the sustainability of long-term agricultural practices. Furthermore, the results of this study provide a basis for the development of more sophisticated plant disease detection systems that can be extended to various plant species.

II. METHODOLOGY AND METHODS

A. Dataset

Open access data sets available online provide high-quality, community-curated resources to accelerate model development and experimentation. Kaggle, in particular, offers a wide variety of data sets contributed by the global community, covering various domains such as images, text, and numerical data. For the purpose of building a deep learning model for corn leaf disease detection, obtaining a dataset from Kaggle is a cost-effective step in training the model to be developed. Specifically, the PlantVillage [14] and PlantDoc [15] repositories, which are accessible through Kaggle, include an extensive collection of corn leaf images. PlantDoc contains 39 different leaf categories. From this open source, this study uses four classes: gray leaf spot, common rust, northern leaf blight, and healthy leaf. Table 1 presents detailed information about these selected categories, which are the main focus of corn leaf classification.

Table 1. Corn dataset

Class	Channel	Format	Total Amount
Gray leaf spot	RGB color image	JPG	1000
Common rust	RGB color image	JPG	1000
Northern leaf blight	RGB color image	JPG and JPEG	1000
Healthy	RGB color image	JPG	1000

The JPG/JPEG format, short for Joint Photographic Experts Group—the organization responsible for setting this standard—represents the same type of image file. The difference between "JPG" and "JPEG" lies in the length of the file extension, a variation that stems from limitations in older operating systems. Table 2 presents the classification of corn leaf images.

Table 2. Image corn leaf

Gray leaf spot	Common rust
	
Northern leaf blight	Healty
	

B. Convolutional Neural Network

Convolutional Neural Networks (CNN) are specifically designed to process structured image data [16]. These networks have become the industry standard in various computer vision applications, including image classification and object detection [17]. CNNs are capable of recognizing visual patterns, such as edges, gradients, shapes, and even complex structures like forms and patterns, making them effective for visual recognition tasks [18]. Generally, CNNs consist of several layers arranged sequentially, with each layer having its own function in feature extraction and classification as shown as Figure 1[19].

- Input layer: Receives raw images and converts them into numerical tensors for further processing.
- Convolutional layer: Detects fundamental features such as edges, textures, and angles by applying filters (kernels) across the image. This produces a feature map that represents the spatial location of these features.

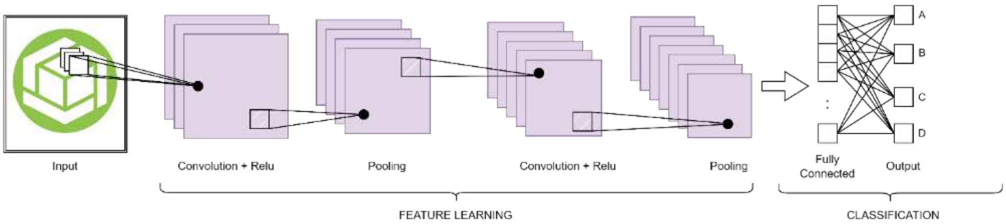

Fig 1. Illustrating the main layers of CNN and their functions

- ReLU (Rectified Linear Unit) activation layer: Introduces nonlinearity into the network, allowing it to learn complex feature representations.
- Pooling layer: Reduces the spatial dimensions of feature maps through down-sampling, thereby minimizing computational costs and reducing overfitting. Common methods include max pooling, which selects the maximum value from a region, and average pooling, which calculates the average value.
- Fully connected layer: Serves as the final classification stage by flattening the feature map into a one-dimensional vector and applying a dense connection to predict class labels.

Additional layers are also often integrated, such as:

- Batch normalization layer: Normalizes the output from the previous layer, improving training efficiency and increasing accuracy.
- Dropout layer: Randomly disables a subset of neurons during training to reduce overfitting and improve generalization.

C. *Residual Networks*

The basic concept of ResNet lies in the use of residual connections, also known as skip connections [20]. These connections allow the network to learn residual mappings with respect to the input, rather than trying to estimate the complete transformation directly, described visually in Figure 2. This is achieved by adding the original input to the output at a certain layer, thereby increasing training stability and accelerating convergence [21]. The typical ResNet architecture is organized into several stages, each consisting of several residual blocks. The initial stage generally includes a convolutional layer followed by a max pooling operation to reduce the spatial dimensions [22]. Various ResNet variants, such as ResNet-18, ResNet-34, ResNet-50, ResNet-101, and ResNet-152, are distinguished based on their architectural depth and the complexity of their residual block structures [23].

III. Result and Discussions

The previous subsection discussed the dataset, which consists of 4,000 images distributed evenly across four classes, with 1,000 samples in each category as shown in Table 1. The data was collected from Kaggle, which consists of images taken using a 48 MP smartphone camera with a native resolution of 4000 × 3000 pixels. For model training, the dataset was resized to 224 × 224 pixels. The data was then partitioned into three subsets: training, validation, and testing. A total of 80% of the data was allocated for training purposes, further divided into training and validation at a ratio of 80:20. This resulted in 2,560 training samples (640 per class: gray leaf spot, common rust, northern leaf blight, healthy leaf) and 640 validation samples (160 per class). The remaining 20% was designated as the test set, consisting of 800 images (200 per class). In this study, experiments were conducted with a focus on layer selection, with the results summarized in Table 2.

Table 2. Scenario of Training and Testing Dataset

Layer selection	Data train	Data validation	Data test
None Activation Batch Normalization Pooling	640 each class	160 each class	200 each class

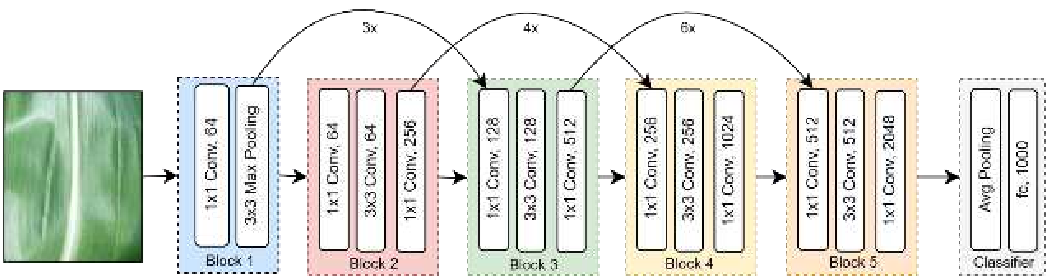

Fig 2. Residual Block

The previous subsection discussed the dataset, which consists of 4,000 images distributed evenly across four classes, with 1,000 samples in each category as shown in Table 1. The data was collected from Kaggle, which consists of images taken using a 48 MP smartphone camera with a native resolution of 4000 × 3000 pixels. For model training, the dataset was resized to 224 × 224 pixels. The data was then partitioned into three subsets: training, validation, and testing. A total of 80% of the data was allocated for training purposes, further divided into training and validation at a ratio of 80:20. This resulted in 2,560 training samples (640 per class: gray leaf spot, common rust, northern leaf blight, healthy leaf) and 640 validation samples (160 per class). The remaining 20% was designated as the test set, consisting of 800 images (200 per class). In this study, experiments were conducted with a focus on layer selection, with the results summarized in Table 2.

The performance of the concatenated model in classifying corn leaf diseases was evaluated using four standard metrics: accuracy, precision, recall, and F1 score. Accuracy indicates the proportion of correctly predicted events relative to the total number of samples, reflecting the closeness of the model's predictions to the actual labels, as illustrated in Figure 3 and defined in equation (1) [24]. Precision measures the ratio of true positives to total positives predicted (TP and FP), thus indicating the reliability of the model in predicting positive cases (pers. (2)) [25]. Recall assesses the model's ability to correctly identify all actual positive cases, expressed as the proportion of true positives among all actual positives (pers. (3)) [26]. The F1 score, defined as the harmonic mean of precision and recall, provides a balanced measure of classification performance. A higher F1 score indicates that the model exhibits strong precision and effective recall, making it a robust indicator of classification quality (equation (4)) [27].

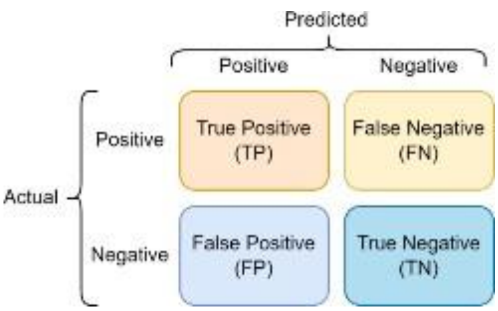

Fig. 3. Confusion matrix

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

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

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

$$F1_{Score} = \frac{2TP}{2TP+FP+FN} \quad (4)$$

Based on the results of testing the validation probability model of a concatenated CNN with a ResNet backbone that combines MobileNetV3, DenseNet161, and GoogleNet using three types of optimizers namely Adam, RMSProp, and SGD, it can be seen that the choice of optimizer has a significant effect on the performance of corn leaf disease classification. The model with the Adam optimizer showed low accuracy (0.5391) with relatively high loss (1.7302), making it less than optimal for detecting disease classes. Meanwhile, the RMSProp optimizer provided better performance with an accuracy of 0.7094 and precision of 0.7221, although the resulting loss value was still quite high (3.4758).

Conversely, the use of the SGD optimizer provided the best results with a low loss value 0.1877, high accuracy score 0.9531, and consistent other evaluation metrics, namely precision 0.9549, recall 0.9531, and F1-score 0.9536. This indicates that SGD is more stable in the training process and capable of producing better generalization compared to Adam and RMSProp in the hybrid CNN architecture used. Thus, it can be concluded that for corn leaf disease classification based on images, the SGD optimizer is the best result, as shown in Table 3.

Table 3. Performance Metrics of CNN Concatenation (Validation Probability)

Model Optimizer	Loss	Acc	Precision	Recall	F1
Concatenate Adam	1,7302	0,5391	0,6955	0,5391	0,4992
Concatenate RMSProp	3,4758	0,7094	0,7221	0,7094	0,6802
Concatenate SGD	0,1877	0,9531	0,9549	0,9531	0,9536

Based on the probability test results on the hybrid CNN model with ResNet backbone combined with MobileNetV3, DenseNet161, and GoogleNet, it appears that model performance is greatly influenced by the choice of optimizer as described in Table 4 . The Adam optimizer provides fairly good results with an accuracy of 0.9088 and precision of 0.9248, accompanied by a relatively low loss value (0.3841). This shows that Adam is capable of achieving a high level of generalization, although there are still weaknesses in the stability of the results when compared to other optimizers. Conversely, RMSProp produces less than optimal performance, with an accuracy of only 0.7113 and a high loss value (3.7360), indicating that this optimizer is less suitable for the architecture and dataset used in this study.

The SGD optimizer once again demonstrated superior performance with consistent results across all evaluation metrics. The loss value obtained was very low (0.2275), accompanied by an accuracy of 0.9513, precision of 0.9536, recall of 0.9513, and an F1-score of 0.9512. These results confirm that the use of SGD provides stability in the training process while improving the model's generalization ability to test data. Thus, it can be concluded that the SGD optimizer is the best choice in implementing a hybrid CNN model for corn leaf disease classification, as it significantly outperforms Adam and RMSProp in terms of accuracy, precision, sensitivity, and balance between evaluation metrics. The data can be validated through the results Table number 4.

Table 4. Performance Metrics of CNN Concatenation (Test Probability)

Model Optimizer	Loss	Acc	Precision	recall	F1
Concatenate Adam	0,3841	0,9088	0,9248	0,9087	0,9100
Concatenate RMSProp	3,7360	0,7113	0,7182	0,7113	0,6764
Concatenate SGD	0,2275	0,9513	0,9536	0,9513	0,9512

To clarify the differences in model performance at the validation probability and test probability stages, the comparison results are displayed visually in Figure 4 and Figure 5. This visualization facilitates the analysis of performance differences between optimizers, making it clear which model provides the most optimal results based on the evaluation metrics used.

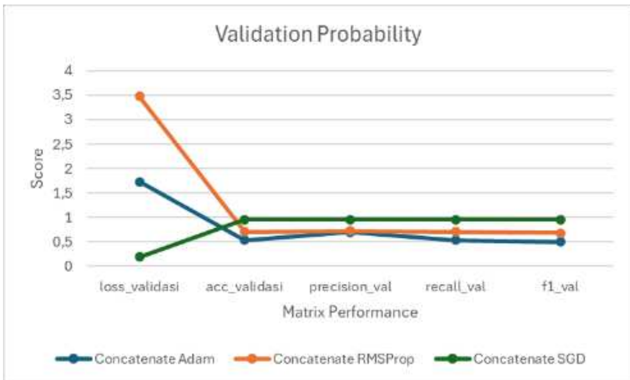

Fig 4. Validation Probability in Graph

The graph in the figure above shows a comparison of the performance of three optimizers, Adam, RMSProp, and Stochastic Gradient Descent (SGD), based on validation evaluation metrics including loss, accuracy, precision, recall, and f1-score. The test results show that RMSProp produces the highest validation loss value (3.478), indicating suboptimal model convergence with this optimizer. In contrast, the SGD optimizer shows the lowest validation loss value, indicating better model generalization on the validation data. Meanwhile, Adam was in the middle with a relatively stable loss value. In terms of accuracy, precision, recall, and f1-score, the three optimizers showed almost similar trends, although Adam and SGD were more consistent than RMSProp. This confirms that the choice of optimizer has a significant effect on model performance, especially in reducing the validation loss value. The same trend is observed in Figure 5, which shows that the Adam optimizer has the lowest loss value among the other optimizers. The trends in precision, recall, and F1 scores are not significantly different between validation probability and test probability for all optimizers used.

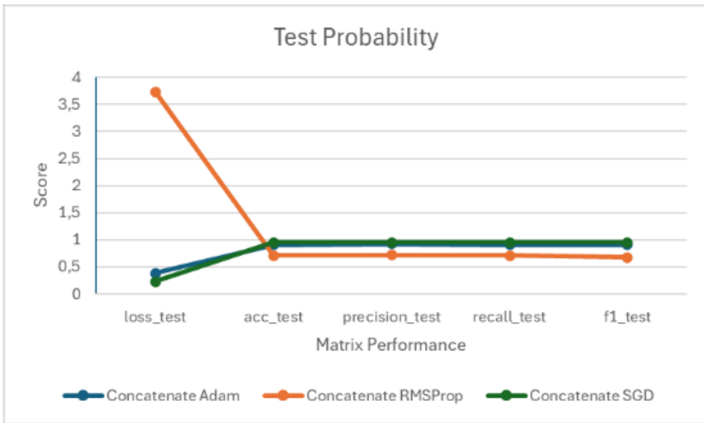

Fig 5. Test Probability in Graph

In the end of this experiment, the performance of SGD outperform over Adam and RMSProp can be explained from both theoretical and empirical perspectives. SGD with properly tuned hyperparameters often provides better generalization performance compared to adaptive optimizers. While Adam and RMSProp adapt the learning rate for each parameter, they sometimes converge too quickly to sharp minima in the loss landscape. These sharp minima can lead to lower validation performance and higher overfitting, as observed in the high validation loss of RMSProp in the results. In contrast, SGD explores the loss surface more conservatively, enabling the model to converge toward flatter minima, which are generally associated with better generalization ability on unseen data.

IV. CONCLUSIONS

This study demonstrated the effectiveness of a hybrid deep learning approach that integrates ResNet as the main backbone with MobileNetV3, DenseNet161, and GoogleNet for automated corn leaf disease identification. Among the tested optimizers, Stochastic Gradient Descent (SGD) consistently provided the most reliable performance, achieving an accuracy of 95.13% with balanced precision, recall, and F1-score. These results highlight that, beyond architectural complexity, the choice of optimizer plays a decisive role in ensuring model generalization and stability.

From a practical perspective, the proposed hybrid framework has the potential to be deployed as part of smart farming solutions, enabling early and accurate disease detection to support smallholder farmers in minimizing crop losses and improving food security. Future research should focus on extending this framework to handle multi-crop disease identification, incorporating real-time deployment on mobile or edge devices, and enhancing robustness through larger, more diverse datasets. Furthermore, investigating advanced optimization strategies, such as adaptive learning rate schedules or meta-learning approaches, could further improve training

p-ISSN 2715-2871
e-ISSN 2714-5247

efficiency and model scalability for broader agricultural applications.

Acknowledgment

This paper was facilitated by LPPM funding for the 2025 Beginner Research initiative, contract number 201/UN46.4.1/PT.01.03/RISMAN/2025, with gratitude to LPPM UTM.

References

[1] O. Erenstein, M. Jaleta, K. Sonder, K. Mottaleb, and B. M. Prasanna, "Global maize production, consumption and trade: trends and R&D implications," Food Security, vol. 14, no. 14, May 2022, doi: https://doi.org/10.1007/s12571-022-01288-7.

[2] S. Amaruzaman, D. K. Bardsley, and R. Stringer, "Analysing agricultural policy outcomes in the uplands of Indonesia: A multi-dimensional sustainability assessment," Sustainable Development, vol. 31, no. 1, pp. 1937–1950, Jan. 2023, doi: https://doi.org/10.1002/sd.2494.

[3] Y. Jiao, H.-D. Chen, H. Han, and Y. Chang, "Development and Utilization of Corn Processing by-Products: A Review," Foods, vol. 11, no. 22, p. 3709, Nov. 2022, doi: https://doi.org/10.3390/foods11223709.

[4] D. L. Nsibo, I. Barnes, and D. K. Berger, "Recent advances in the population biology and management of maize foliar fungal pathogens Exserohilum turcicum, Cercospora zeina and Bipolaris maydis in Africa," *Frontiers in Plant Science*, vol. 15, Aug. 2024, doi: https://doi.org/10.3389/fpls.2024.1404483.

[5] F. Mayo, C. Maina, M. Mgala, and N. Mduma, "Deep learning models for the early detection of maize streak virus and maize lethal necrosis diseases in Tanzania," Frontiers in Artificial Intelligence, vol. 7, Aug. 2024, doi: https://doi.org/10.3389/frai.2024.1384709.

[6] T. Zhu et al., "A Deep Learning Model for Accurate Maize Disease Detection Based on State-Space Attention and Feature Fusion," Plants, vol. 13, no. 22, p. 3151, Nov. 2024, doi: https://doi.org/10.3390/plants13223151.

[7] Saleem, Potgieter, and Mahmood Arif, "Plant Disease Detection and Classification by Deep Learning," Plants, vol. 8, no. 11, p. 468, Oct. 2019, doi: https://doi.org/10.3390/plants8110468.

[8] Y. Lu, S. Yi, N. Zeng, Y. Liu, and Y. Zhang, "Identification of rice diseases using deep convolutional neural networks," Neurocomputing, vol. 267, pp. 378–384, Dec. 2017, doi: https://doi.org/10.1016/j.neucom.2017.06.023.

[9] K. Shaheed *et al.*, "EfficientRMT-Net—An Efficient ResNet-50 and Vision Transformers Approach for Classifying Potato Plant Leaf Diseases," *Sensors*, vol. 23, no. 23, pp. 9516–9516, Nov. 2023, doi: https://doi.org/10.3390/s23239516.

[10] S. PAN et al., "Intelligent diagnosis of northern corn leaf blight with deep learning model," Journal of Integrative Agriculture, vol. 21, no. 4, pp. 1094–1105, Apr. 2022, doi: https://doi.org/10.1016/s2095-3119(21)63707-3.

[11] A. Howard *et al*., "Searching for MobileNetV3," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South), 2019, pp. 1314-1324, doi: https://doi.org/10.1109/ICCV.2019.00140. [12]

[12] G. Huang, Z. Liu, G. Pleiss, L. Van Der Maaten, and K. Weinberger, "Convolutional Networks with Dense Connectivity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2019, doi: https://doi.org/10.1109/tpami.2019.2918284.

[13] H. B. G and V. N, "An efficient image dahazing using Googlenet based convolution neural networks," *Multimedia Tools and Applications*, May 2022, doi: https://doi.org/10.1007/s11042-022-13222-2.

[14] D. Singh, N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, "PlantDoc: A Dataset for Visual Plant Disease Detection," Proceedings of the 7th ACM IKDD CoDS and 25th COMAD, pp. 249–253, Jan. 2020, doi: https://doi.org/10.1145/3371158.3371196.

[15] J. Yao, S. N. Tran, S. Garg, and S. Sawyer, "Deep Learning for Plant Identification and Disease Classification from Leaf Images: Multi-prediction Approaches," arXiv (Cornell University), Jan. 2023, doi: https://doi.org/10.48550/arxiv.2310.16273.

[16] M. Shafiq and Z. Gu, "Deep Residual Learning for Image Recognition: A Survey," Applied Sciences, vol. 12, no. 18, p. 8972, Sep. 2022, doi: https://doi.org/10.3390/app12188972.

[17] M. Krichen, "Convolutional Neural Networks: A Survey," Computers, vol. 12, no. 8, pp. 151–151, Jul. 2023, doi: https://doi.org/10.3390/computers12080151.

[18] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, "A review of convolutional neural networks in computer vision," Artificial Intelligence Review, vol. 57, no. 4, Mar. 2024, doi: https://doi.org/10.1007/s10462-024-10721-6.

[19] S. Cong and Y. Zhou, "A review of convolutional neural network architectures and their optimizations," Artificial Intelligence Review, vol. 56, no. 3, Jun. 2022, doi: https://doi.org/10.1007/s10462-022-10213-5.

[20] F. He, T. Liu, and D. Tao, "Why ResNet Works? Residuals Generalize," IEEE Transactions on Neural Networks and Learning Systems, pp. 1–14, 2020, doi: https://doi.org/10.1109/tnnls.2020.2966319.

[21] A. Kopaničáková and R. Krause, "Globally Convergent Multilevel Training of Deep Residual Networks," *SIAM Journal on Scientific Computing*, vol. 45, no. 3, pp. S254–S280, Aug. 2022, doi: https://doi.org/10.1137/21m1434076.

[22] W. Huang and H. Zhang, "Convergence analysis of deep residual networks," Analysis and Applications, vol. 22, no. 02, pp. 351–382, Sep. 2023, doi: https://doi.org/10.1142/s021953052350029x.

[23] W. Xu, Y.-L. Fu, and D. Zhu, "ResNet and its application to medical image processing: Research progress and challenges," *Computer Methods and Programs in Biomedicine*, vol. 240, p. 107660, Oct. 2023, doi: https://doi.org/10.1016/j.cmpb.2023.107660.

[24] Aeri Rachmad, Muhammad Fuad, and S. Rochman, "Convolutional Neural Network-Based Classification Model of Corn Leaf Disease," Mathematical modelling of engineering problems, vol. 10, no. 2, pp. 530–536, Apr. 2023, doi: https://doi.org/10.18280/mmep.100220.

[25] S. Rochman et al., "Classification of Salt Quality Based on the Content of Several Elements in the Salt Using Machine Learning," Mathematical Modelling and Engineering Problems, vol. 11, no. 4, pp. 1005–1012, Apr. 2024, doi: https://doi.org/10.18280/mmep.110417.

[26] S. Susanto Putro, F. Adiputra, E. Mala Sari Rochman, A. Rachmad, M. A. Syakur, and S. Bayu Seta, "Comparison of SAW and WP methods to determine the best agricultural land," Communications in Mathematical Biology and Neuroscience, Jan. 2021, doi: https://doi.org/10.28919/cmbn/5820.

[27] Aeri Rachmad, Fifin Ayu Mufarroha, S. Rochman, Muhammad Ali Syakur, Firdaus Solihin, and Yuli Panca Asmara, "The Impact of Convolutional Layer Selection in ResNet-50v2 Architecture on Corn Leaf Disease Classification," Ingénierie des systèmes d information, vol. 30, no. 6, Jun. 2025, doi: https://doi.org/10.18280/isi.300609.