International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 30, No. 6, June, 2025, pp. 1509-1515
Journal homepage: http://iieta.org/journals/isi

The Impact of Convolutional Layer Selection in ResNet-50v2 Architecture on Corn Leaf Disease Classification

Aeri Rachmad[1*], Fifin Ayu Mufarroha[1], Eka Mala Sari Rochman[1], Muhammad Ali Syakur[1], Firdaus Solihin[1], Yuli Panca Asmara[2]

[1] Department of Informatics, Faculty of Engineering, University of Trunojoyo Madura, Bangkalan 69162, Indonesia
[2] Faculty of Engineering and Quantity Surveying, INTI International University, Nilai 71800, Malaysia

Corresponding Author Email: aery_r@trunojoyo.ac.id

https://doi.org/10.18280/isi.300609

Received: 21 April 2025
Revised: 24 May 2025
Accepted: 18 June 2025
Available online: 30 June 2025

Keywords:
food production, agricultural AI, ResNet-50v2, batch normalization, corn leaf diseases classification, deep learning

ABSTRACT

Agriculture is a significant economic sector in many nations, including Indonesia, with corn being one of the primary food crops. Leaf infections in corn plants can result in substantial losses for farmers and disruptions in food production. This disease diagnosis application can enable early disease detection for smallholder farmers. This is because current procedures, which rely on manual knowledge from agronomists, are time-consuming and costly. Advances in Agricultural AI (Artificial Intelligence) and image processing have enabled autonomous identification of plant diseases using the convolutional neural networks (CNN) technique, with ResNet-50v2 being one of the established architectures. The main purpose of this research is to investigate the selection and design of appropriate convolutional layers in the ResNet-50v2 model. In order to identify corn leaf diseases, the following layers were chosen: activation, batch normalization, and pooling. There are 4,000 entries in the dataset, distributed among four categories: gray leaf spot, common rust, northern leaf blight, and healthy. The data will be separated into three categories: learning, validation, and testing. According to the study's findings, the chosen convolutional layer using ResNet-50v2 obtained a 95.7% accuracy, precision, recall, and F1-score. The activation layer obtained 92.4% precision and 92.1% accuracy, recall, and F1-score. The batch normalization layer obtained 98.1% precision 98% accuracy, recall, and F1-score. The pooling layer obtained 94.7% precision and 94.5% accuracy, recall, and F1-score. The achievement of this study reveals that the performance of the Batch Normalization convolutional layer outperforms other layers. The findings of this study show that batch normalization layers in the ResNet-50v2 model outperform other layers in Corn leaf disease classification. This highlights the effectiveness of batch normalization in mitigating overfitting, a frequent challenge in deep learning systems.

1. INTRODUCTION

Agriculture is an important economic sector in most nations, including Indonesia. This sector not only serves as the rural economy's backbone, but it also plays an important role in government [1]. Agriculture employs a significant portion of the rural population, lowers unemployment rates, and alleviates poverty. Furthermore, this industry contributes significantly to food security and economic stability. Corn is one of Indonesia's most important food crops, contributing significantly to the country's economy [2, 3]. Corn is used for food, animal feed, and industrial raw materials. Processed corn goods, such as corn flour and corn oil, offer substantial added value. However, corn plant productivity is frequently hampered by a variety of leaf diseases, which can drastically impair crop yield and quality. These diseases, including leaf blight, leaf spot, and leaf rust, can spread swiftly and damage adjacent plants if not recognized and treated promptly.

Early diagnosis of leaf diseases in corn plants is critical for minimizing the disease's detrimental impact. Agronomists have traditionally detected diseases manually by monitoring visual signs on plant leaves. Although this strategy can be beneficial, it is time-consuming, difficult, and costly, with a high reliance on human abilities and expertise. Thus, a more efficient and automated technique is required to identify leaf diseases in corn plants precisely and rapidly.

Recent breakthroughs in artificial intelligence (AI) and image processing have expanded the potential for automated plant disease diagnosis [4-7]. A particularly interesting strategy is the use of convolutional neural networks (CNNs), a form of artificial neural network specifically developed for pattern identification in visual data that has proved to be very effective in a range of picture classification tasks [8, 9]. ResNet-50v2 (Residual Network 50 version 2) is one of the most powerful and popular CNN designs. ResNet-50v2, developed by Microsoft Research, employs the residual learning idea, allowing for the training of extremely deep networks without encountering the vanishing gradient problem [10]. This can be achieved by the use of shortcut connections that connect certain levels in the network, resulting in a smoother and more efficient flow of information. ResNet-50v2 can overcome the network performance decrease. This makes ResNet-50v2 ideal for complicated picture classification jobs. However, the performance of ResNet-50v2 is determined not just by the depth of its architecture, but also by the selection and tuning of its numerous convolutional components. The research [5, 7] shows significant advances in the use of sophisticated deep learning architectures, such as CNNs, ResNet, and others, for the identification of plant diseases, which can increase the precision of diagnostics. According to the research, deep neural networks may have problems like overfitting, especially if the dataset is too small or undiversified. To overcome the drawbacks of feature extraction, research might concentrate on applying the design of the ResNet50-v2 to efficiently extract important characteristics from plant images. As shown in earlier research, overfitting may be lessened by using strategies like dropout or data augmentation with ResNet50-v2.

This research has significant ramifications in the realms of agriculture and AI. By assessing how the selection and setup of various layers in the ResNet-50v2 architecture affect classification performance. Experiments were carried out by altering the layers and evaluating the models. Understanding the effect of each convolutional component should allow us to optimize the ResNet-50v2 architecture for the goal of corn leaf disease classification. This will not only help farmers diagnose and treat leaf diseases more rapidly, but it will also boost the overall productivity and sustainability of the agricultural industry. The findings of this study can be used to design a more complex plant disease detection system that can be applied to a wide range of plants.

2. METHODOLOGY AND METHODS

2.1 Dataset

Freely accessible data on the internet delivers high-quality data that has been selected by the community, which speeds up model building and experimentation. Kaggle hosts a variety of datasets provided by the worldwide community, encompassing a wide range of topics including photos, text, numbers, and more. Collecting corn leaf datasets using Kaggle is a critical step toward constructing deep learning models for corn leaf disease detection. The PlantVillage and PlantDoc databases [11-13] include corn leaf data gathered from Kaggle. PlantDoc provides 39 different leaf classes. We take four classes: gray leaf spot, common rust, northern leaf blight, and healthy. Table 1 shows extensive data from the courses, which focus on the corn leaf.

JPG/JPEG format is an abbreviation for Joint Photographic Experts Group, the organization that developed these format standards (JPG/JPEG) refer to the same file format, differing only in the number of letters due to limitations of older operating systems. The biggest difference is the extension length. Figure 1 displays the corn leaf classifications.

Table 1. Corn dataset

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

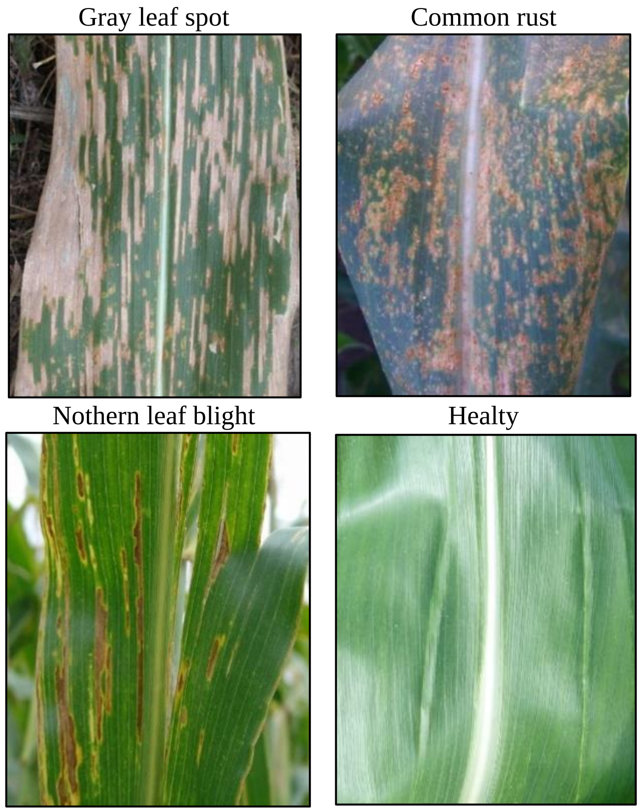

Figure 1. Image corn leaf

2.2 Convolutional neural network

A convolutional neural network (CNN) is an artificial neural network that processes structured data, such as images [14].

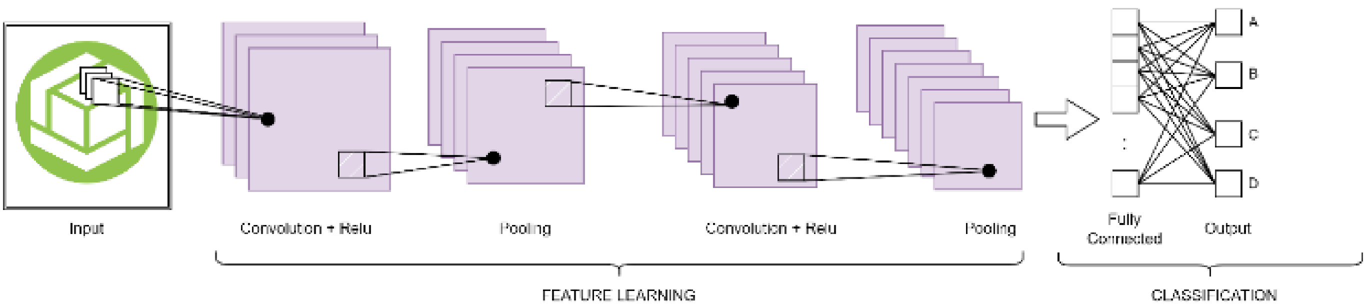

Figure 2. CNN architecture

CNNs are extremely successful in computer vision applications and have become the industry standard for many visual tasks such as picture categorization and object detection [15]. CNNs can detect patterns in pictures, such as lines, gradients, circles, and even eyes and faces, making them extremely useful for computer vision [16]. In general, CNNs are made up of numerous layers that are arranged consecutively. Each layer has a distinct purpose in extracting information from pictures and doing categorization [17]. The layers of a convolutional neural network (CNN) are a crucial component responsible for recognizing characteristics in the picture seen in Figure 2.

Figure 3 shows the layers of the CNN with extensive explanations as follows:

- The input layer receives a raw picture and modifies the numeric tensor before being processed by the following layer.
- The convolution layer detects edges, corners, and textures by sliding a filter or kernel across the picture. This technique calculates the dot product of the image's pixel values and filter weight values. The result is a feature map that depicts the existence of features at various locations in the image.
- The ReLU (Rectified Linear Unit) Activation Layer is used to introduce nonlinearity into the model. This nonlinearity allows the network to learn from increasingly complicated data.
- Pooling layer: CNN employs a pooling layer using a down-sampling approach to minimize the spatial dimensions of the feature map and prevent overfitting. Pooling is classified into two types: max pooling, which takes the largest value from each tiny location on the feature map, and average pooling, which calculates the average value from each small region on the feature map.
- The fully connected layer is the last layer that performs classification using the characteristics retrieved by the preceding layer. The feature map from the pooling results is flattened into a one-dimensional vector and supplied to the fully connected layer.

In addition to the categories listed above, more layers are often used, including:

- Following the convolution or pooling layer comes the normalizing layer, often known as batch normalization. This layer normalizes the preceding layer's output, which speeds up the training process and increases accuracy.

The Dropout layer is added after the fully connected layer to prevent overfitting by randomly deactivating certain neurons during training.

2.3 Residual networks

ResNets' key principle is the utilization of residual connections, also known as skip connections [18]. These connections enable the network to learn residual functions relative to the input rather than the whole function. This is accomplished by adding the input to the output of a layer, which aids in the stabilization of the training process and convergence [19]. ResNets are often composed of numerous stages, each with several residual blocks. To minimize spatial dimensions, the initial step typically contains a convolutional layer, followed by max pooling [20]. There are various varieties of ResNets, including ResNet-18, ResNet-34, ResNet-50, ResNet-101, and ResNet-152, which differ in the number of layers and the complexity of residual blocks.

2.4 ResNet-50v2

ResNet-50v2 is a deep architecture with 50 layers made up of many convolutional layers grouped in a residual block. This residual block enables the model to learn deeper and more complex representations while avoiding the vanishing gradient problem, which is prevalent in very deep CNN models [21]. The residual block in ResNet-50v2 is made up of three convolutional layers, followed by a shortcut connection that connects the block's input and output to address the vanishing gradient problem. The Residual Network design employs residual blocks to aid in the training of extremely deep networks. A residual block is a structure that adds the block's original input to the output of one or more of its layers. ResNet-50v2's residual block differs somewhat from ResNet-50v1. ResNet-50v2 employs pre-activation, which implies that batch normalization and activation functions are done before the convolution process, as seen in Figure 4. ResNet-50v2's design consists of multiple phases. Each stage has a different number of blocks, and increasingly larger filter sizes are represented in Figure 4.

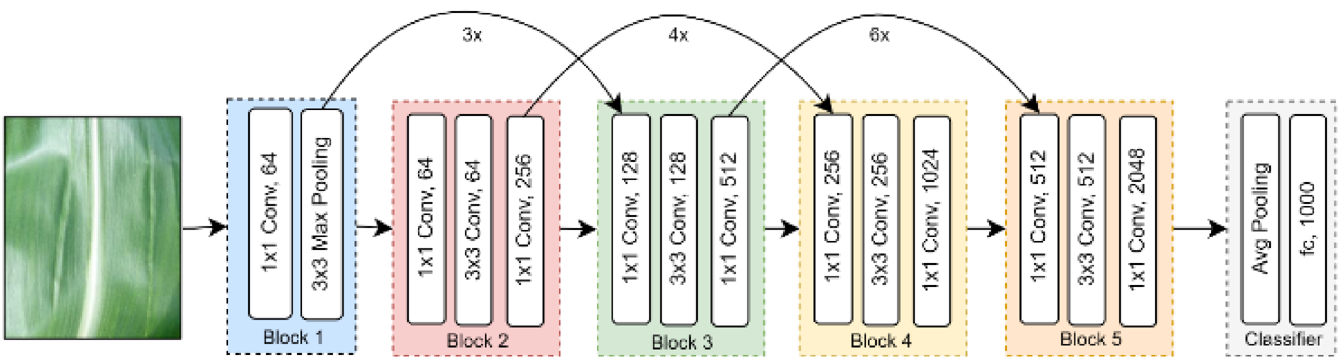

Figure 3. Residual blocks [22, 23]

Each layer of this architectural block has a distinct purpose. The first convolution layer is designed to extract fundamental information from the input. The identity block is a basic block that consists of many convolution layers, Batch Normalization, and ReLU, as well as shortcut connections. The convolution layer here performs the task of combining features that have been extracted by the preceding layer. Shortcut Connection has a direct connection that connects the input to the output of the block, allowing gradients to flow more easily to previous layers. The third layer, more specifically the projection block, adjusts the feature dimensions as needed. Before proceeding to the fully linked layer, the average pooling layer reduces the feature dimensions. The last layer is fully connected/fc, which is responsible for classification based on the extracted features [24].

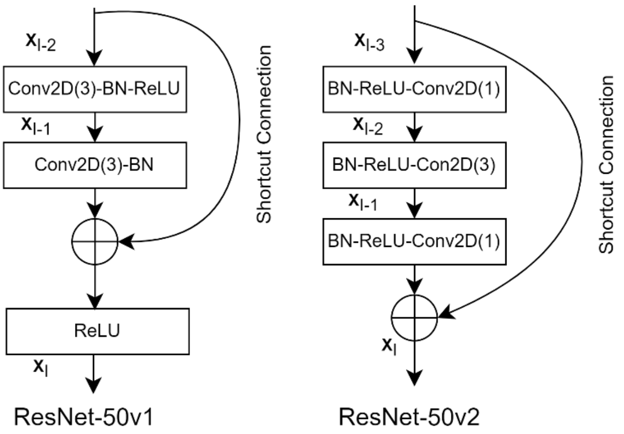

Figure 4. ResNet-50v2 architecture

3. RESULT

The previous sub-chapter addressed the dataset, which had 4000 records divided over four classes. Table 1 shows that there are 1000 data points in each class. The details of the data collection process include 4000 data images from Kaggle sources with an image size of 4000×3000 pixels taken with a 48 MP cellphone camera. The dataset is scaled to 224×224 pixels. The dataset will be separated into three parts: learning, validation, and testing. The percentage of learning data is 80%, but it is divided by the need for validation data at an 80:20 ratio, with a total of 2560 training data (640 gray leaf spot, 640 common rust, 640 northern leaf blight, and 640 healthy) and 640 validation data (160 gray leaf spot, 160 common rust, 160 northern leaf blight, and 160 healthy). The acquisition of test data is 800 data points (200 gray leaf spot, 200 common rust, 200 northern leaf blight, and 200 healthy) at a 20% rate. In this study, an experiment with layer selection was carried out, and the results are displayed in Table 2.

Table 2. Test scenario

Layer Selection	Data Train	Data Validation	Data Test
None Activation Batch Normalization Pooling	640 each class	160 each class	200 each class

We often set the hyperparameters to an Adam optimizer, a learning rate of 0.0001, a batch size of 32, and epochs of 20. To optimize the model's performance, these hyperparameters must be appropriately set because they directly affect the model's stability, generalization ability, and rate of convergence. Every convolutional layer in the ResNet50-v2 architecture is known to play a specific purpose. The model's ability to learn from the data is influenced by the activation function selection. ReLU, for instance, can aid in resolving the vanishing gradient issue, resulting in improved performance and quicker convergence in deeper networks. Therefore, employing an unsuitable activation function may impede training and result in a slower rate of convergence. However, batch normalization enables greater learning rates and can enhance the model's overall performance. It also contributes to the prevention of overfitting. Additionally, the model's capacity to generalize is impacted by the pooling technique. Although max pooling keeps the most important features, it may miss important information that average pooling may have picked up. These specific convolutional layers and their configurations are crucial for ResNet-50v2 to perform effectively in image classification tasks while maintaining efficient training dynamics. These decisions also help ResNet-50v2 tackle difficult challenges.

The performance of the ResNet-50v2 model on corn leaf classification was measured using accuracy, precision, recall, and F1-score calculations. Accuracy refers to how accurately the measurement findings match the actual value [25]. Figure 5 shows that accuracy measurement indicates how near the measurement findings are to the actual value, or in other words, the fraction of right predictions from all data (Eq. (1)) [26]. Precision refers to the accuracy of repeated measurements. Precision is a proportion of correct positive predictions among all positive predictions (TP and FP) (Eq. (2)) [27]. Recall refers to the model's ability to detect all positive cases that exist. Recall lists a number of positive cases identified by the model (Eq. (3)) [28]. The F1-score is a combination of precision and recall used to evaluate the model's performance in classification. A high F1-score value suggests that the model has strong accuracy and recall (see Eq. (4)) [29].

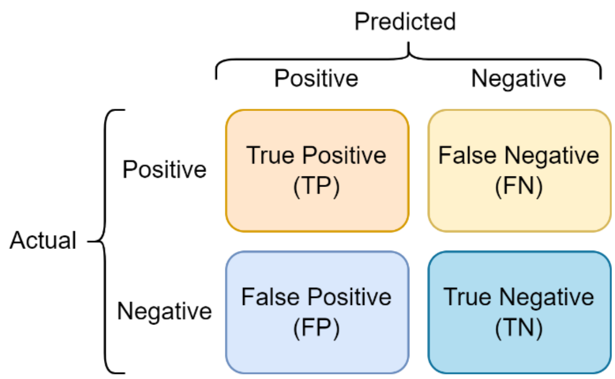

Figure 5. Confusion matrix

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

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

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

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

Table 2 serves as a guideline for evaluating the developed model. Table 3 shows a summary of the outcomes of the model performance metrics, including the accuracy, precision, recall, and F1-score. Table 4 summarizes the findings of the model test performance measures.

The accuracy of validation data is used to track training and compare models. This is done to evaluate the model's performance during learning. When we analyze the model assessment results in Table 3, we see that the batch normalization layer achieves the highest accuracy value of 94.6% compared to the other layers. This is directly related to the outcomes of precision, recall, and F1-score assessments, in which batch normalization outperforms other layers. Although the results from one layer to the next are not very different, this demonstrates that batch normalization dominates overall performance outcomes by predicting the results of the model class with the actual class.

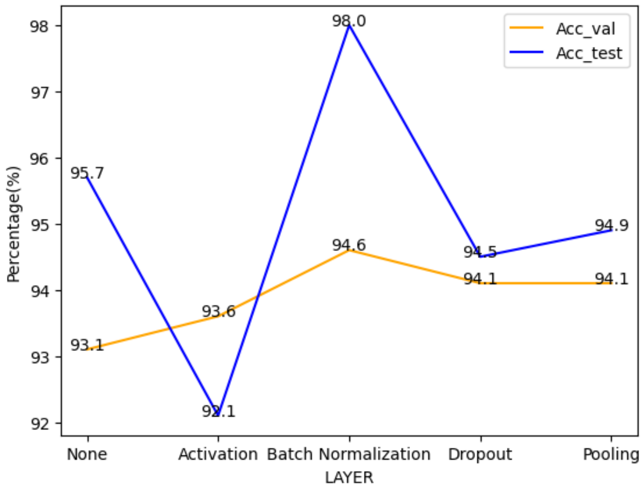

Figure 6. Comparison of accuracy results

On the other side, testing is used to determine real performance once learning is completed. The purpose is to assess the model's ultimate performance. Table 4 demonstrates that the batch normalization layer achieves the highest accuracy, recall, and F1-score values of 98.0% and 98.1%, precision. This number is 2.3% lower than the "None" layer, which has a value of 95.7%, 3.5% in the pooling layer, and 5.9% in the activation layer. Figure 6 shows a comparison of validation and test accuracy results.

In artificial intelligence, loss is a quantifiable statistic that indicates how much the model's prediction deviates from the actual value. The model's ability to predict data improves as the loss value decreases. Figure 7 shows that the batch normalization layer has the minimum loss, which is 16.4%, as found during validation. Similarly, in the test, the least loss is 8.7%, indicating that the model correctly learnt the pattern in the data and generated accurate predictions in that layer. Comparing earlier research using the ResNet-50v2 model demonstrates that this approach can complete its classification goal. The information on several examples of previous studies is detailed in Table 5.

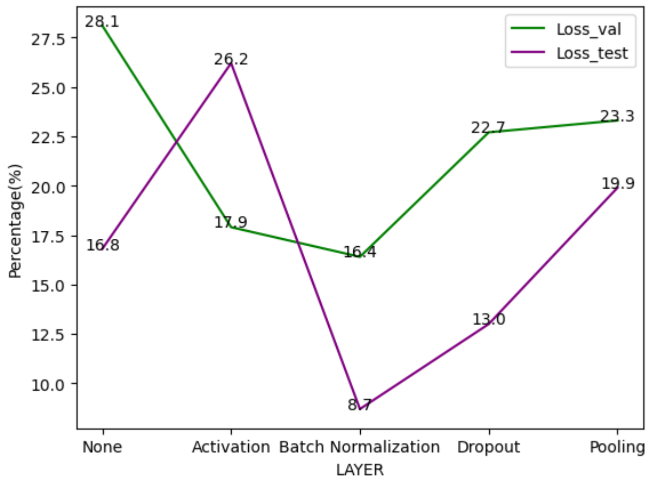

Figure 7. Comparison of loss results

Table 3. Results of model performance

Layer Selection	Acc_val (%)	Precision_val (%)	Recall_val (%)	F1-score_val (%)
None	93.1	93.1	93.1	93.1
Activation	93.6	93.6	93.6	93.6
Batch Normalization	**94.6**	**94.7**	**94.6**	**94.6**
Pooling	94.1	94.1	94.1	94.1

Table 4. Performance results on test data

Layer Selection	Acc_test (%)	Precision_test (%)	Recall_test (%)	F1-score_test (%)
None	95.7	95.7	95.7	95.7
Activation	92.1	92.4	92.1	92.1
Batch Normalization	**98.0**	**98.1**	**98.0**	**98.0**
Pooling	94.5	94.7	94.5	94.5

Table 5. Comparison studies

Year	Circumstances	Term	Accuracy	Reference
2020	ResNet-50V2	Optimizer	89.7%	[30]
2023	ResNet-50V2	Activation	86.32%	[31]
2024	ResNet-50V2	-	93%	[32]
2024	ResNet-50V2	None	95.7%	Proposed
		Activation	92.1%	
		Batch Normalization	98.0%	
		Pooling	94.5%	

4. CONCLUSIONS

The purpose of this study is to investigate the variation of convolutional layers and the optimum settings for CNN models in the ResNet-50v2 architecture to get the greatest performance. This study will look into how differences in the structure of convolutional layers affect model performance in the goal of identifying corn leaf diseases with high accuracy and excellent generalization. Experimentation was used to determine the most efficient convolutional layer in terms of accuracy from corn leaf data with four classes. The experiment's findings show that the batch normalization layer outperforms the other layers in both the validation and testing stages. The batch normalization layer achieved greater accuracy, precision, recall, and F1-score values. Overall, the batch normalization layer is a useful strategy for increasing the performance of neural network models. The batch normalization layer prevents overfitting, enhances model stability, and maximizes model correctness by normalizing the data.

This suggests that using artificial intelligence (AI) to treat corn leaf disease, particularly the CNN approach, has shown significant results in the classification of different kinds. In cases of corn leaf disease, these findings serve as the first stage of any preventive or therapeutic measures. The development of the ResNet-50v2 architecture using various optimizers is another topic of interest. Research on fine-tuning and transfer learning can potentially be conducted. Here, we wish to examine more parameters that could influence model performance to see how the findings are achieved.

ACKNOWLEDGEMENT

This research was funded by a grant from the Ministry of Research and Higher Education through the University of Trunojoyo Madura in 2024.

REFERENCES

[1] Setiartiti, L. (2021). Critical point of view: The challenges of agricultural sector on governance and food security in Indonesia. E3S Web of Conferences. 232: 01034. https://doi.org/10.1051/e3sconf/202123201034

[2] Farid, F., Rehman, A., Bashir, M.T., Khan, M.M.H., Uddin, M.A., Qureshi, H.A., Abbas, M.I. (2024). Synergistic effects of corn stalk ash and fly ash on the properties of lightweight concrete using pumice aggregate. Iranian Journal of Science and Technology, Transactions of Civil Engineering, 49: 1245-1257. https://doi.org/10.1007/s40996-024-01491-w

[3] Putro, S.S., Adiputra, F., Rochman, E.M.S., Rachmad, A., Syakur, M.A., Seta, S.B. (2021). Comparison of SAW and WP methods to determine the best agricultural land. Communications in Mathematical Biology and Neuroscience, 2021: 1-15. https://doi.org/10.28919/cmbn/5820

[4] Jha, K., Aalap, D., Poojan, P., Manan, S. (2019). A comprehensive review on automation in agriculture using artificial intelligence. Artificial Intelligence in Agriculture, 2: 1-12. https://doi.org/10.1016/j.aiia.2019.05.004

[5] Vishnoi, V.K., Krishan, K., Brajesh, K. (2021). Plant disease detection using computational intelligence and image processing. Journal of Plant Diseases and Protection, 128: 19-53. https://doi.org/10.1007/s41348-020-00368-0

[6] Orchi, H., Sadik, M., Khaldoun, K. (2021). On using artificial intelligence and the internet of things for crop disease detection: A contemporary survey. Agriculture, 12(1): 9. https://doi.org/10.3390/agriculture12010009

[7] Yağ, İ., Altan, A. (2022). Artificial intelligence-based robust hybrid algorithm design and implementation for real-time detection of plant diseases in agricultural environments. Biology, 11(12): 1732. https://doi.org/10.3390/biology11121732

[8] Rawat, W., Wang, Z. (2017). Deep convolutional neural networks for image classification: A comprehensive review. Neural Computation, 29(9): 2352-2449. https://doi.org/10.1162/neco_a_00990

[9] Chen, L., Li, S., Bai, Q., Yang, J., Jiang, S., Miao, Y. (2021). Review of image classification algorithms based on convolutional neural networks. Remote Sensing, 13(22): 4712. https://doi.org/10.3390/rs13224712

[10] Landman, T., Nissim, N. (2021). Deep-Hook: A trusted deep learning-based framework for unknown malware detection and classification in Linux cloud environments. Neural Networks, 144: 648-685. https://doi.org/10.1016/j.neunet.2021.09.019

[11] Rachmad, A., Fuad, M., Rochman, E.M.S. (2023). Convolutional neural network-based classification model of corn leaf disease. Mathematical Modelling of Engineering Problems, 10(2): 530-536. https://doi.org/10.18280/mmep.100220

[12] Singh, D., Jain, N., Jain, P., Kayal, P., Kumawat, S., Batra, N. (2020). PlantDoc: A dataset for visual plant disease detection. In Proceedings of the 7th ACM IKDD CoDS and 25th COMAD, pp. 249-253. https://doi.org/10.1145/3371158.3371196

[13] Geetharamani, G., Arun Pandian, J. (2019). Identification of plant leaf diseases using a nine-layer deep convolutional neural network. Computers & Electrical Engineering, 76: 323-338. https://doi.org/10.1016/j.compeleceng.2019.04.011

[14] Tayal, A., Gupta, J., Solanki, A., Bisht, K., Nayyar, A., Masud, M. (2022). DL-CNN-based approach with image processing techniques for diagnosis of retinal diseases. Multimedia Systems, 28(4): 1417-1438. https://doi.org/10.1007/s00530-021-00769-7

[15] Dhillon, A., Verma, G.K. (2020). Convolutional neural network: A review of models, methodologies and applications to object detection. Progress in Artificial Intelligence, 9(2): 85-112. https://doi.org/10.1007/s13748-019-00203-0

[16] Kusumaningrum, R., Arafifin, A.H., Khoerunnisa, S.F., Sasongko, P.S., Wirawan, P.W., Syafrudin, M. (2025). Hyperparameter optimization for convolutional neural network-based sentiment analysis. Journal of Advanced Research in Applied Sciences and Engineering Technology, 53(1): 44-56. https://doi.org/10.37934/araset.53.1.4456

[17] Pralhad Gavali, M.E., Banu, J.S. (2019). Deep convolutional neural network for image classification on CUDA platform. In Deep Learning and Parallel Computing Environment for Bioengineering Systems, pp. 99-122. Academic Press. https://doi.org/10.1016/B978-0-12-816718-2.00013-0

[18] Rachmad, A., Syarief, M., Hutagalung, J., Hernawati, S., Rochman, E.M.S., Asmara, Y.P. (2024). Comparison of CNN architectures for Mycobacterium tuberculosis classification in sputum images. Ingénierie des Systèmes d'Information, 29(1): 49-56. https://doi.org/10.18280/isi.290106
[19] Rachmad, A., Sonata, F., Hutagalung, J., Hapsari, D., Fuad, M., Rochman, E.M.S. (2023). An automated system for osteoarthritis severity scoring using residual neural networks. Mathematical Modelling of Engineering Problems, 10(5): 1849-1856. https://doi.org/10.18280/mmep.100538
[20] Farag, H.H., Said, L.A., Rizk, M.R.M., Ahmed, M.A.E. (2021). Hyperparameters optimization for ResNet and Xception in the purpose of diagnosing COVID-19. Journal of Intelligent & Fuzzy Systems, 41(2): 3555-3571. https://doi.org/10.3233/JIFS-210925
[21] Borawar, L., Kaur, R. (2023). ResNet: Solving Vanishing Gradient in Deep Networks. In: Mahapatra, R.P., Peddoju, S.K., Roy, S., Parwekar, P. (eds) Proceedings of International Conference on Recent Trends in Computing. Lecture Notes in Networks and Systems, vol 600. Springer, Singapore. https://doi.org/10.1007/978-981-19-8825-7_21
[22] Atienza, R. (2018). Advanced Deep Learning with Keras: Apply deep learning techniques, autoencoders, GANs, Variational Autoencoders, Deep Reinforcement Learning, Policy Gradients, and More. Packt Publishing Ltd.
[23] Van Hieu, N., Hien, N.L.H. (2020). Recognition of plant species using deep convolutional feature extraction. International Journal on Emerging Technologies, 11(3): 904-910.
[24] Qazi, E.U.H., Zia, T., Almorjan, A. (2022). Deep learning-based digital image forgery detection system. Applied Sciences, 12(6): 2851. https://doi.org/10.3390/app12062851
[25] Rachmad, A., Chamidah, N., Rulaningtyas, R. (2020). Mycobacterium tuberculosis identification based on colour feature extraction using expert system. Annals of Biology, 36(2): 196-202.
[26] Rochman, E.M.S., Setiawan, W., Hardi, S., Permana, K.E., Husni, Asmara, Y.P., Rachmad, A. (2024). Classification of salt quality based on the content of several elements in the salt using machine learning. Mathematical Modelling of Engineering Problems, 11(4): 1005-1012. https://doi.org/10.18280/mmep.110417
[27] Rachmad, A., Chamidah, N., Rulaningtyas, R. (2020). Mycobacterium tuberculosis images classification based on combining of convolutional neural network and support vector machine. Communications in Mathematical Biology and Neuroscience, 2020: 1-13. https://doi.org/10.28919/cmbn/5035
[28] Rochman, E.M.S., Suprajitno, H., Rachmad, A., Santosa, I. (2023). Utilizing LSTM and K-NN for Anatomical Localization of Tuberculosis: A Solution for Incomplete Data. Mathematical Modelling of Engineering Problems 10(4). https://doi.org/10.18280/mmep.100403
[29] Rochman, E.M.S., Suprajitno, H., Kamilah, I., Rachmad, A., Santosa, I. (2023). Tuberculosis classification using random forest with K-prototype as a method to overcome missing value. Communications in Mathematical Biology and Neuroscience, 2023: 1-26. https://doi.org/10.28919/cmbn/7873
[30] Rahimzadeh, M., Attar, A. (2020). A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet-50V2. Informatics in Medicine Unlocked, 19: 100360. https://doi.org/10.1016/j.imu.2020.100360
[31] Haruna, U., Ali, R., Man, M. (2023). A new modification CNN using VGG19 and ResNet-50V2 for classification of COVID-19 from X-ray radiograph images. Indonesian Journal of Electrical Engineering and Computer Science, 31(1): 369-377. https://doi.org/10.11591/ijeecs.v31.i1.pp369-377
[32] Riyadi, S., Abidin, F.A., Audita, N. (2024). Comparison of ResNet-50V2 and MobileNetV2 models in building architectural style classification. In 2024 International Conference on Intelligent Systems and Computer Vision (ISCV), Fez, Morocco, pp. 1-8. https://doi.org/10.1109/ISCV60512.2024.10620099