J-COSINE (Journal of Computer Science and Informatics Engineering)
Vol. 9, No. 1, June 2025 E-ISSN:2541-0806
Accredited Sinta-4 by RISTEKDIKTI Decree No. 79/E/KPT/2023 P-ISSN:2540-8895

Comparative Analysis of ResNet-50 and VGG16 Architecture Accuracy in Garbage Classification System

Putu Yudhis Andrean*, Fitri Bimantoro, Regania Pasca Rassy
Dept Informatics Engineering, Mataram University
Jl. Majapahit 62, Mataram, Lombok NTB, INDONESIA
Email: ptydhis@gmail.com, *:* [bimo, ganiarachsy]@unram.ac.id

Corresponding Author

Population growth and urbanization have led to an exponential rise in waste generation, posing significant environmental and health risks. Efficient garbage classification is crucial for optimizing recycling and reducing landfill waste. This study compares the performance of two Convolutional Neural Network (CNN) architectures, VGG16 and ResNet-50, for classifying six garbage categories: cardboard, glass, metal, paper, plastic, and trash. Using a dataset of 2,467 images, the models were trained and evaluated with enhanced preprocessing and data augmentation techniques. The results indicate that VGG16 achieved slightly higher accuracy, precision, recall, and F1-score (97%) compared to ResNet-50 (96%). However, ResNet-50 demonstrated better computational efficiency with a faster average training time (1 second per epoch versus 3 seconds for VGG16). Despite these promising results, the limited dataset size may affect the models' generalization ability, a challenge addressed through data augmentation. This study helps further development of automated waste sorting systems for recycling management, paving the way for more sustainable waste solutions. Recommendations for future research include expanding the dataset, exploring other architectures to improve model accuracy, and developing a system that can assist the community in processing garbage according to its type.

***Key words*: Garbage Classification, CNN, Deep Learning, VGG16, ResNet-50**

I. Introduction

Population growth and rapid urbanization have increased the amount of waste generated. Indonesia, for example, is estimated to produce more than 187.2 million tons of waste per year, ranking second only to China as the largest waste-producing country in the world [1]. The increasing volume of waste has a negative influence on the environment, causing soil, water, and air pollution. Public health can be threatened if waste accumulates and is not managed properly. Waste piles can become breeding grounds for diseases, which can lead to epidemics. In addition, air pollution from burning waste can cause long-term health problems for residents [2]. To solve these problems, effective and sustainable waste management innovations are needed. Recycling and reutilization of waste resources should be increased.

Garbage classification helps reduce harmful environmental impacts. By segregating organic, inorganic, and B3 (hazardous and toxic materials) waste, we can prevent recyclable waste from ending up in landfills, where it can damage soil and water. Sorting organic waste, for example, can be used to make compost, reducing waste volume while producing natural fertilizer [3]. By classifying garbage, the recycling process becomes more efficient. Well-separated waste makes it easier to process and reuse materials that are still useful.

Deep learning has emerged as one of the most effective methods for image classification, utilizing the capacity of artificial neural networks to identify patterns and characteristics in visual data. CNN are specifically built to interpret grid data, such as photographs, and can recognize complex visual patterns. CNN can extract important elements from images using convolution layers, such as edges, textures, and shapes, which are essential for image classification. Research shows that CNN can achieve excellent accuracy in a variety of image classification systems, such as the use of ResNet-50 and VGG16 architectures [4]. With the ability to automatically extract important information from images, CNNs have also proven effective in garbage classification [5].

The challenges in deep learning implementation requires a lot of high-quality training data. CNN models are routinely trained with thousands to millions of images to obtain effective results. Data limitations can lead to weak models and poor generalization to new data [6]. Although CNN can achieve high accuracy, the results are not always stable, and their performance varies during training. This suggests that, although the model may perform well, issues such as overfitting and poor parameter selection may impact accuracy stability [7].

The VGG16 architecture is a 16-layer artificial neural network that recognizes complex visual patterns in images. The network consists of 3 fully linked layers and 13 convolutional layers. VGG16 is popular for its ability to detect local features using convolution and pooling filters, making it useful in image classification systems. However, the main disadvantage of VGG16 is the large model size and high computing resource usage due to the number of parameters reaching hundreds of millions [8]. Meanwhile, ResNet-50 is a residual network design with 50 layers that uses residual blocks to ease the learning process by maintaining the relationship between input and output and is more efficient and has fewer parameters [9]. Fundamental differences in their architectural design, which has a direct impact on accuracy, training time, and computational efficiency.

Previous research in garbage classification has generally focused on only one neural network architecture, such as VGG16, without directly comparing the performance between several different architectures [10]. For example, while some studies have applied VGG16 to classify garbage types and provided good results, few studies have specifically compared the performance of VGG16 with ResNet-50 in the same context. This suggests a gap that needs to be filled to better understand the advantages and disadvantages of each architecture in garbage management.

This study aims to compare the accuracy between ResNet-50 and VGG16 architectures in garbage classification, focusing on determining which architecture is more effective and efficient. Although many previous studies have used VGG16 for image classification, there are still few that compare its performance directly with ResNet-50. By conducting this comparison, the research is expected to provide a deeper insight into the advantages of each architecture, as well as assist in selecting the most suitable architecture for a better garbage classification system.

This study has a possibility to enhance the accuracy of garbage classification by determining a more effective architecture, which is crucial for improving garbage management. A higher increase in accuracy will aid effective garbage sorting, encouraging recycling and waste reduction. By choosing a more efficient architecture, the garbage classification process can be performed faster and with lower resource usage.

II. Literature Review and Basic Theory

A. Related Research

This study purpose to develop a model that can classify garbage types and then compare the performance of the model using two different architectures. It is expected that this model can be the basis of an automated waste sorting system for recycling management so that each type of waste is directed to the right recycling process. In previous research, VGG16, ResNet-50, MobileNet, and Inception-V3 have successfully classified the types of garbage using a dataset of 4 types of garbage and produced relatively high accuracy [11]. In this study, garbage is classified using a dataset of 6 types of garbage including: cardboard, glass, metal, paper, plastic, and trash. The deep learning architectures used are VGG16 and ResNet-50.

In previous studies, the trained VGG16 was widely used to detect and classify garbage types using garbage datasets, with relatively high accuracy [4, 5, 6]. An example of previous research used 8134 images of garbage types. The accuracy obtained from the study was 82.89% and the validation accuracy was 84.62% [12]. Another study used organic and inorganic garbage datasets to identify the type of garbage, the accuracy obtained was 97.99% [13]. In addition, research using a balanced dataset to detect plastic and non-plastic bottle waste, with 10 epochs, the resulting model obtained an accuracy of 96.39% [14].

Another architecture ResNet-50 is known as a model that has good performance on large datasets [5]. In previous research [15], classification of garbage types was carried out using ResNet-50. The results obtained show that ResNet-50 produces good performance for a dataset of 2751 images. Previous research shows that the use of the ResNet model produces better accuracy than the AlexNet architecture model [16]. From the above studies, it is found that the VGG16 and ResNet-50 architectures are effective for image classification. Therefore, this research will use these two methods to classify the type of garbage.

B. Supporting Theory

The following are general theories that are used as support in this research:

B.1. Convolutional Neural Network

Convolutional Neural Network (CNN) is an artificial neural network design frequently employed to evaluate visual images. CNN are very useful in image processing and computerization because they can automatically extract important elements from input images [17]. The CNN design uses specialized layers for hierarchical feature extraction from image data. CNN works similar to Multilayer Perceptron (MLP), but each neuron is represented in two dimensions [18].

There are several previous journals that discuss garbage classification using the CNN method. Previous research [19] successfully classified types of garbage into several different categories using the CNN model with an accuracy rate of 98.45%. In addition, previous research classified recyclable garbage, using the CNN model, the results showed performance with an accuracy rate of 92% in the training process and 79% in the testing process [20]. The latest research also classifies recyclable garbage using the CNN method with accuracy results reaching 95.35% [5]. From some of the above studies, it can be concluded that the use of CNN models to classify the type of garbage has an average accuracy above 90%. There are several architectures of CNN, namely: LeNet, Alexnet, VGG, Inception, ResNet, DenseNet, MobileNet, EfficientNet, Xception, and NASNetMobile. In this research, VGG16 and ResNet-50 are two CNN architectures that will be compared in accuracy to the model built.

B.2. VGG16

VGG16 is a Convolutional Neural Network (CNN) architecture developed by Oxford University's Visual Geometry Group (VGG) that has become one of the most widely used image classification models. The architecture contains 16 layers of small 3x3 convolutional structures stacked sequentially, allowing for additional feature extraction and increased depth of detail [21].

Previous research, trained VGG16 was used to detect and classify tomato leaf diseases with relatively high accuracy reaching 97.78% [10]. In addition, the latest research classifies spice images in Indonesia, using the VGG16 architecture getting an accuracy of 85% [22]. From some of the above studies, it can be concluded that the use of VGG16 architecture provides good accuracy and performance levels.

B.3. ResNet-50

ResNet-50 is a Convolutional Neural Network (CNN) architecture that uses residual connections to overcome the problem of information decay as the depth of the model increases. With 50 layers, this architecture uses shortcut connections that allow inputs to skip one or more layers, facilitating deep model training and resulting in excellent image recognition performance [23].

In previous research, a trained ResNet-50 was used to detect and classify robusta coffee leaf disease with a relatively high accuracy of 92.68% [24]. In addition, the latest research classifies vehicle tire cracks, using the ResNet-50 architecture getting an accuracy of 94% [25]. From some of the above studies, it can be concluded that the use of the ResNet-50 architecture provides good accuracy and performance levels, just like the VGG16 architecture.

This research uses two CNN architectures, namely VGG16 and ResNet-50. By using these two architectures, researchers evaluated each model's performance. The performance comparison is seen in terms of accuracy, computational efficiency, and training time to identify whether they are able to provide comparable or better results than architectures such as MobileNet and Inception-V3. This research explores the advantages and disadvantages of VGG16 and ResNet-50 in image classification, especially on limited datasets. The goal is to produce a model that is small, performs well and can run quickly for best accuracy.

III. Research Methodology

A. Research Flow

The following is a flowchart that explains the flow of this research.

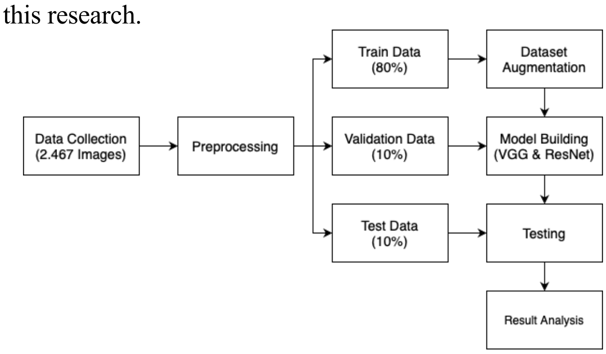


Fig. 1. Research Flow

B. Dataset

This research uses an image dataset of 2,467 types of garbage obtained from Kaggle. The types of garbage contained in this research dataset and their details are listed in TABLE I.

TABLE I. Dataset Distribution

Class	Total
Cardboard	393
Glass	491
Metal	400
Paper	584
Plastic	472
Trash	127

As shown in Table I, the dataset was divided into 3 subsets with a division ratio of 80% training data, 10% validation data and 10% test data. Model training uses 80% training data with a total of 2259 images by applying data augmentation to increase the variety of data available, allowing the model to see a variety of new images during training without increasing the number of physical images in the dataset, as well as increasing the model's ability to generalize on data that has not been seen before. Then initial testing is done using 10% validation data with a total of 898 images. Testing using validation data is necessary to ensure that the model not only learns to memorize training data but also generalizes well to new data that has never been trained before. Finally, data testing uses 10% test data to evaluate the model's performance after training.

C. Preprocessing

Preprocessing aims to improve image quality for optimal training results and ensure all images are the same size. This step includes adjusting the size of the debris images to 180x180 pixels, to ensure uniformity in quality and size, reduce computational load, and maintain consistency during training. Choosing this resolution is the best choice as it balances image detail and computational efficiency, supporting the model's performance in image classification.

D. Data Augmentation

Data augmentation is applied to expand the quantity and variety of training data. In this research, it is exclusively employed to train data because the goal is to expand the variety and diversity of examples seen by the model during the training process, so that the model becomes more robust and able to recognize patterns better under various conditions.

Validation data and test data, on the other hand, must remain representative of real data without modification. This is necessary to verify that the test and validation data are correct, reflect the model's performance under its original conditions and ensure a fair and accurate evaluation of the model, without the influence of unnecessary modifications. In addition, maintaining the authenticity of the validation and test data helps avoid overfitting, when the model looks great on the training set of data but fails to provide good generalization on the real data.

The applied augmentation includes rescale, rotation, width_shift_range, height_shift_range, shear_range, zoom_range, horizontal_flip, and fill_mode techniques. By using these augmentation parameters, the image classification model is kept from overfitting and data variation is increased. So that the model can be trained better and has a higher generalization ability.

E. Classification

This research utilizes two Deep Learning architectures, namely ResNet-50 and VGG16, for garbage type classification. The VGG16 architecture is designed with a focus on simplicity and network depth through an iterative array of convolution layers with small kernels (3x3), then a pooling layer to minimize the spatial dimension (max pooling) without losing important features, along with a fully connected layer for the final prediction. Despite having many parameters, VGG16 excels in identifying image features with high resolution. Meanwhile, ResNet-50 addresses the vanishing gradient issue in deep networks by utilizing the idea of residual learning with shortcut connections. A convolution layer with batch normalization and ReLU activation makes up the residual block of ResNet-50. Following this is a global average pooling (GAP) layer, which comes before a fully connected layer that generates predictions. ResNet-50 efficiently handles complex image datasets with high accuracy without significantly increasing the number of parameters.

IV. Results and Discussion

Some important parameters were used in training this model, such as batch_size = 64, epoch = 60, and image size of 180 x 180 pixels. In addition, a split of 80%, 10%, and 10% of the training, validation, and test data was used. These parameters are used to ensure optimal pattern recognition on the dataset used.

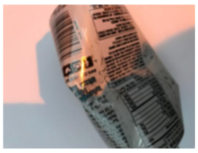
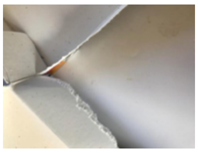
A. Dataset Augmentation

TABLE II. Dataset Augmentation

Parameter	Value
Rescale	1.0/255
Rotation range	30
Width shift range	0.2
Height shift range	0.2
Shear range	0.2
Zoom range	0.2

As shown in Table II, about dataset augmentation, several parameters are used to increase the variety of data. Rescale= 1.0/255 is done to normalize the pixels in the image from values of 0-255 to between 0 and 1, which helps in model convergence. Rotation range of 30 was chosen as a moderate change in angle is enough to represent the natural rotational variation of the object without changing the basic structure of the image too much. Width shift range and height shift range of 0.2 provide flexibility towards shifting the position of the object, while shear range and zoom range allow changing the geometry and size of the object, respectively. These values were chosen based on the dataset's nature, which includes typical garbage items, and experimental results showing that such adjustments enhance model generalization without introducing excessive distortion.

TABLE III. Dataset After and Before Augmentation

Class	Before Augmented	After Augmented
Cardboard		
Glass		
Metal		
Paper		
Plastic		
Trash		

As shown in Table III, the original image was transformed, the enhanced dataset has augmented. Before augmentation, the dataset contains only the original images with no variations. After augmentation, the dataset changes as each original image becomes transformed through various techniques such as rescaling, rotation, translation, shear, and zoom. These transformations create new images with different shapes, positions, sizes, and orientations. As a result, the visual variety of the resulting dataset is greater than the original dataset, which helps the model learn from a more diverse dataset.

B. Dataset Preprocessing

The dataset that has been collected will be processed to make it easier to manage and recognize by the model. To make sure the data structure is appropriate and prepared for model training. This process includes adjusting and enhancing the data to increase the model's precision and effectiveness in identifying patterns. The results show that consistent use of an image resolution of 180 x 180 pixels enables optimal classification, with a good balance between accuracy and computational efficiency, and the ability to retain important details in the image that support effective classification.

C. Model Building

Model building in this study uses the Convolutional Neural Network (CNN) method to perform image classification and will use the TensorFlow library for model initialization and configuration. The following is the configuration of the hyperparameters used in model building.

TABLE IV. HYPERPARAMETER CONFIGURATION

Hyperparameter	Value
Optimizer	Adam (0.0001)
Batch Size	64
Epochs	60
Activation Function	SoftMax

Table IV, this hyperparameter configuration was chosen to balance the efficiency, stability, and generalization ability of the model. The Adam optimizer with learning rate (0.0001) provides fast and stable convergence, batch size 64 maximizes computational efficiency and generalization ability, 60 epochs provides sufficient training time without overdoing it, and the SoftMax activation function in the output layer ensures interpretable probabilities for multiclass classification.

C.1. VGG16 Architecture

TABLE V. VGG16 ARCHITECTURE

Layer Type	Description	Output Shape
Input	Input shape	(180, 180, 3)
Base Model	VGG16 (pretrained on ImageNet, no top)	(5, 5, 512)
GlobalMaxPooling2D	Maximum pooling operation applied across inputs	(512)
Dense	256 units, activation: ReLU	(256)
Dropout	Dropout rate: 0.5	(256)
Dense	6 units, activation: SoftMax	(6)

Table V, displays the architecture of the VGG16 model used in this research and has been adapted to the designed model. In the classification part, there is a GlobalMaxPooling2D layer, two fully connected layers, and a dropout technique. The GlobalMaxPooling2D layer creates a 1-dimensional vector that is utilized as input for the fully connected layer by taking the maximum value of each feature map dimension produced by the final convolution layer. The first completely connected layer has 256 neurons that are activated by ReLU. The second completely connected layer, that is the output layer, has 6 neurons which employs a sigmoid activation function for classification. Dropout is applied to balance accuracy, improve generalization, and mitigate overfitting.

C.2. ResNet-50 Architecture

TABLE VI. RESNET-50 ARCHITECTURE

Layer Type	Description	Output Shape
Input	Input shape	(180, 180, 3)
Base Model	ResNet-50 (pretrained on ImageNet, no top)	(6, 6, 2048)
GlobalAveragePooling2D	Average pooling operation applied across inputs	(2048)
Dense	1024 units, activation: ReLU	(1024)
Dropout	Dropout rate: 0.5	(1024)
Dense	512 units, activation: ReLU	(512)
Dropout	Dropout rate: 0.3	(512)
Dense	6 units, activation: SoftMax	(6)

Table VI, displays the architecture of the ResNet-50 model used in this research and has been adapted to the designed model. In the classification part, there is a GlobalAveragePooling2D layer, three completely connected layers, and a dropout technique. The GlobalAveragePooling2D layer creates a 1-dimensional vector that is used as input for the fully connected layer by taking the average value of each feature map dimension produced by the final convolution layer. The first completely connected layer has 1024 neurons that are activated by ReLU, following that the second completely connected layer which has 512 neurons that are activated by ReLU. The third completely connected layer, that is the output layer, has 6 neurons which employs a sigmoid activation function for classification. Dropout rate adjustment is used to achieve the best balance between accuracy, generalization ability, and avoid overfitting.

D. Discussion and Analysis Results

Before displaying the Accuracy and loss graphs, the models were trained using data split between 80% training, 10% validation, and 10% test. The ResNet-50 and VGG16 models are trained to identify patterns in the training data during the training phase. The model's generalization ability is then assessed using validation data. The ResNet-50 and VGG16 models were trained for 60 epochs and evaluated using Accuracy and loss metrics to measure model performance and error.

The evaluation results were visualized in graphs to show performance stability. Test data of 10% was utilized to assess the performance of the model is based on Accuracy, loss, and effectiveness against data that has never been encountered before to then be used in assessing the ability of the model to generalize and evaluate the success of the training process. The accuracy graph shows how well the model generalizes to unseen data, while the loss curve reveals the optimization process, highlighting potential issues like overfitting or underfitting. The effectiveness graph assesses the model's ability to make accurate predictions in real-world conditions, emphasizing its practical application.

D.1. Classification using VGG16

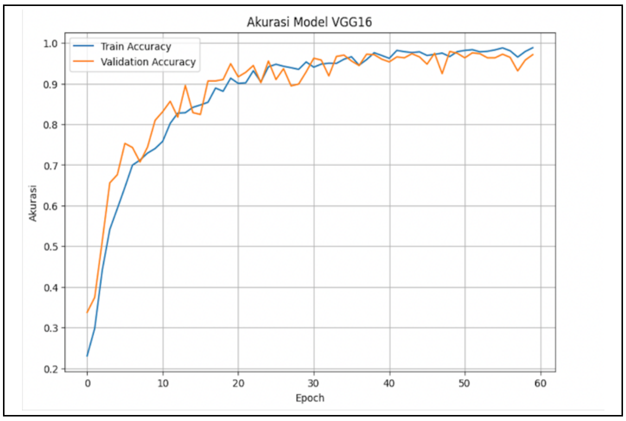


Fig. 2. VGG16 Accuracy Result

In Figure 2, the VGG16 test results illustrate good performance. The training accuracy and validation accuracy graphs show an upward trend as the epochs increase, with both stabilizing after a few epochs. The training accuracy reached values above 0.95, while the validation accuracy stayed above 0.93. To provide better clarity, the x-axis of the graph represents the number of epochs, showing the number of iterations through the dataset, while the y-axis represents the accuracy percentage, indicating the model's performance on both training and validation datasets. The training accuracy graph is shown in blue, while the validation accuracy graph is represented in orange. These visualizations highlight the model's ability to perform well on the classification task. However, the slight discrepancy, with training accuracy being marginally higher than validation accuracy, may suggest potential overfitting to the training data.

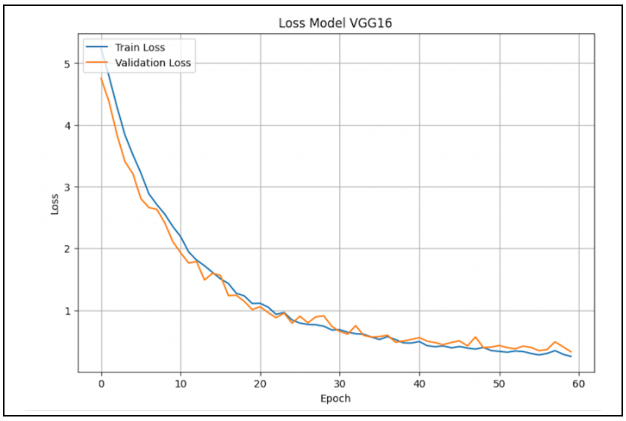


Fig. 3. VGG16 Loss Result

In Figure 3, the graph displays the VGG16 model's performance throughout training in terms of loss. The x-axis represents the number of epochs, while the y-axis shows the loss value, which measures the model's error or discrepancy between the predicted and actual labels. In general, both lines show a downward trend in loss, indicating that the model is learning and improving with each epoch. The blue line represents the training loss, and the orange line represents the validation loss. However, it can be seen that the loss on the training data is typically lower than that on the validation data, which could indicate overfitting. This suggests that while the model performs well on the training data, it may not generalize as effectively to new, unseen data.

To assess the model's performance in predicting the kind of garbage, a confusion matrix was analyzed, the results of which are displayed in Figure 4 for the VGG16 model.

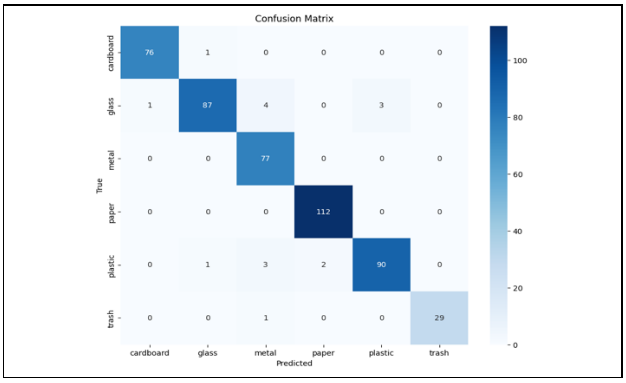


Fig. 4. VGG16 Confusion Matrix Result

Then, a table is displayed that includes columns for accuracy, precision, recall, and F1-score to help visualize the model's capacity to correctly and accurately classify images of garbage types. This is the final result after the model has been completely through the training and testing stages.

TABLE VII. RESULT OF VGG16 ARCHITECTURE

Method	**Accuracy**	**Precision**	**Recall**	**F1-Score**
VGG16	97%	97%	97%	97%

The test results in Table VII, VGG16 Architecture results from performance metrics where VGG16 achieves high Accuracy, Precision, Recall, and F1-score with a value of 97%. Training time is 3 seconds per epoch. The advantages of VGG16 include its simpler architecture, and its ability to adapt to limited datasets, making it a more efficient and optimized choice especially in situations with limited computing resources and limited datasets, without sacrificing the Accuracy and performance of the model.

D.2. Classification using ResNet-50

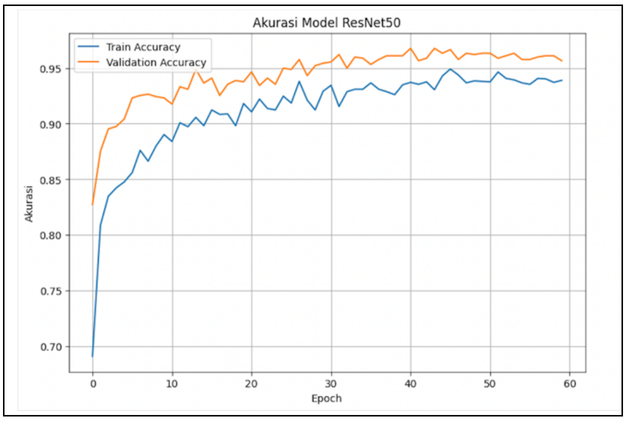


Fig. 5. ResNet-50 Accuracy Result

In Figure 5, illustrates the training accuracy of the ResNet-50 model, with the x-axis representing epochs and the y-axis showing accuracy percentages. The blue and orange lines depict training and validation accuracy, respectively. Accuracy increases significantly early in training, reaching 0.95 for both training and validation accuracy, before training accuracy rises slightly while validation accuracy stabilizes around 0.96. The consistent gap, with training accuracy higher than validation, suggests potential overfitting, indicating the model performs well on training data but struggles to generalize effectively on the validation set.

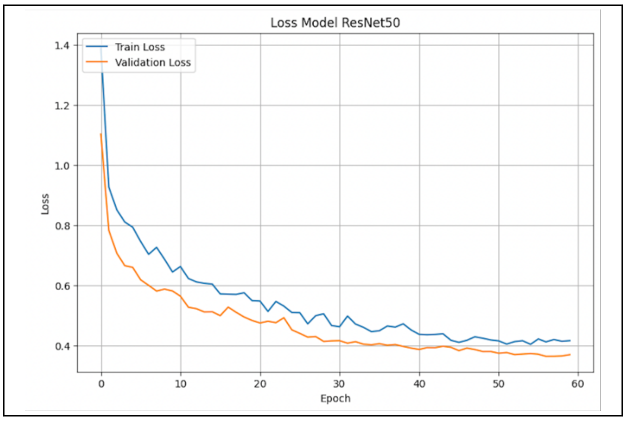


Fig. 6. ResNet-50 Loss Result

In Figure 6, the graph displays the ResNet-50 model's performance throughout training in terms of loss. The x-axis represents the number of epochs, while the y-axis shows the loss value, which measures the model's error. The blue line represents the training loss, and the orange line represents the validation loss. In general, both lines show a decreasing trend in loss, indicating that the model is learning and improving over time. However, it can be seen that the loss on the training data is typically lower than that on the validation data. This discrepancy could indicate overfitting, where the model performs well on the training data but fails to generalize effectively to new, unseen data.

To assess the model's performance in predicting the kind of garbage, a confusion matrix was analyzed, the results of which are displayed in Figure 7 for the ResNet-50 model.

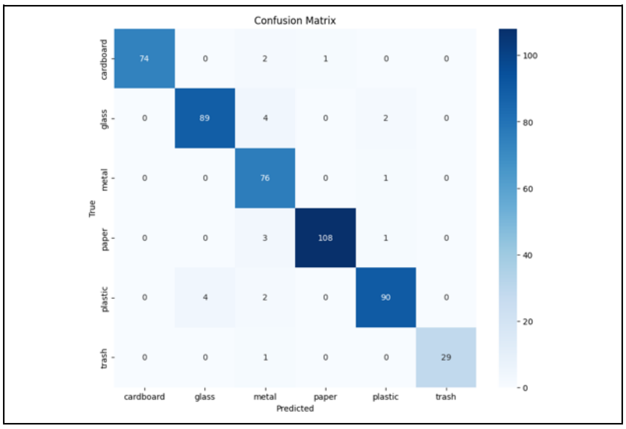


Fig. 7. ResNet-50 Confusion Matrix Result

Then, a table is displayed that includes columns for accuracy, precision, recall, and F1-score to help visualize the model's capacity to correctly and accurately classify images of garbage types. This is the final result after the model has been completely through the training and testing stages.

TABLE VIII. RESULT OF RESNET-50 ARCHITECTURE

Method	Accuracy	Precision	Recall	F1-Score
ResNet-50	96%	96%	96%	96%

The test results in Table VIII, ResNet-50 Architecture results from performance metrics where ResNet-50 achieves high Accuracy, Precision, Recall, and F1-score with a value of 96% although it is still below the VGG16 architecture. But its training time is faster than VGG16 which is only 1 second per epoch. In the case of this dataset, ResNet-50 may struggle to find relevant and effective features making it more prone to overfitting despite having residual connections.

Overall, after training the model by applying data augmentation, the results show that the training does not show a significant difference in results as shown in the accuracy, precision, recall, and F1-score values of 97% and 96%. The training time also differs by 3 seconds per epoch for VGG16 and 1 second per epoch for ResNet-50. One of the reasons behind VGG16 better performance in this context may be related to the size of the dataset used, where VGG16, with its simpler architecture, is better able to utilize features from smaller datasets.

The analysis results show that certain categories, such as plastic and paper, are often misclassified with each other, which may be due to the visual similarity between the two categories, thus confusing the model which can be seen in the confusion matrix.

From the training results in Figure 6, overfitting can be identified through the difference between the training data loss value and the validation data val_loss. Although the accuracy in all scenarios achieved high results, the significant difference between loss and val_loss indicates that the model performed very well on the training data, but could not good generalization on the validation data.

In the ResNet-50 scenario, there is a difference between the accuracy of training and validation, suggesting that the model may be overly complex for the dataset size, leading to overfitting. The residual connections in ResNet-50, while beneficial for deeper networks, may not provide the expected advantage in this case due to the limited amount of training data available. This complexity can hinder the model's ability to learn generalizable features, resulting in a performance gap between training and validation.

In contrast, the VGG16 scenario demonstrates a more consistent performance between training and validation accuracies, indicating that its simpler architecture is better suited for the dataset size. The architecture's ability to effectively extract relevant features without becoming overly complex allows it to maintain high accuracy across both training and validation datasets.

Meanwhile, in the accuracy loss scenario, for both VGG16 and ResNet-50, the difference between loss and val_loss is also visible, especially in the ResNet-50 scenario where loss is higher, but val_loss is lower. This difference indicates that the model is highly focused on the training data and is unable to sustain comparable results on the validation data, which is an indication of overfitting.

To further investigate the impact of dataset size on model performance, it would be beneficial to conduct experiments with larger and more diverse datasets. This could help determine whether the observed performance differences are consistent across various dataset sizes and complexities.

V. Conclusion and Suggestion

Results from this research shows that, classification of garbage types using ResNet-50 and VGG16 architectures with a dataset of 2467 images divided into 6 classes, namely cardboard, glass, metal, paper, plastic, and trash, has a higher accuracy on the VGG16 architecture with 97% accuracy compared to the ResNet-50 architecture with 96% accuracy. With this it can be concluded that the initial model of garbage classification that has been designed by researchers can be used in mobile-based applications, websites and even for IoT devices that will be built to sort garbage.

It is recommended for future research to expand the dataset by increasing the number of images in each class and including various other types of garbage to enhance generalization capabilities. Testing alternative architectures, such as EfficientNet, DenseNet, or MobileNet, could provide valuable insights into identifying models that achieve higher accuracy and better computational efficiency for garbage classification. Additionally, exploring datasets that integrate textual information, such as garbage descriptions or labels, with image data could further improve classification performance. Developing a custom architecture or fine-tuning pre-trained models tailored specifically for this task could also optimize performance, ensuring more accurate results in testing scenarios.

References

[1] Y. Yuniza, “Peningkatan Kesehatan Masyarakat dalam Mengurangi Sampah Rumah Tangga di RT 60 dan RT 71 Banten 6 Kota Palembang,” *Khidmah*, vol. 6, no. 1, pp. 50–56, 2024, doi: 10.52523/khidmah.v6i1.476.

[2] A. Apriyani, M. M. Putri, and S. Y. Wibowo, “Pemanfaatan sampah plastik menjadi ecobrick,” *Masy. Berdaya dan Inov.*, vol. 1, no. 1, pp. 48–50, 2020, doi: 10.33292/mayadani.v1i1.11.

[3] M. Hakam, K. N. Wahyusi, E. N. Hidayah, S. Q Z N, and R. Novembrianto, “Edukasi Pemilahan Sampah Bagi Anak Sekolah Dasar di Desa Giripurno Kecamatan Bumiaji Kota Batu,” *Abdi-mesin J. Pengabdi. Masy. Tek. Mesin*, vol. 2, no. 2, pp. 1–6, 2022, doi: 10.33005/abdimesin.v2i2.28.

[4] G. W. Intyanto, “Klasifikasi Citra Bunga dengan Menggunakan Deep Learning: CNN (Convolution Neural Network),” *J. Arus Elektro Indones.*, vol. 7, no. 3, p. 80, 2021, doi: 10.19184/jaei.v7i3.28141.

[5] S. Meng and W. T. Chu, “A Study of Garbage Classification with Convolutional Neural Networks,” *Indo - Taiwan 2nd Int. Conf. Comput. Anal. Networks, Indo-Taiwan ICAN 2020 - Proc.*, pp. 152–157, 2020, doi: 10.1109/Indo-TaiwanICAN48429.2020.9181311.

[6] I. Susilawati, S. Supatman, and A. Witanti, “Klasifikasi Citra Virus SARS-COV Menggunakan Deep Learning,” *J. Inform. J. Pengemb. IT*, vol. 8, no. 2, pp. 65–70, 2023, doi: 10.30591/jpit.v8i2.4587.

[7] B. Agustina and A. R. Isnain, “Penerapan Deep Learning pada Sistem Klasifikasi Tumor Otak Berbasis Citra CT Scan,” *Indones. J. Comput. Sci.*, vol. 13, no. 2, pp. 284–301, 2024, [Online]. Available: http://ijcs.stmikindonesia.ac.id/ijcs/index.php/ijcs/article/view/3135

[8] R. Rikendry and A. Maharil, “Perbandingan Arsitektur Vgg16 Dan Resnet50 Untuk Rekognisi Tulisan Tangan Aksara Lampung,” *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 3, no. 2, pp. 236–243, 2022, doi: 10.33365/jatika.v3i2.2030.

[9] R. A. Jasin and V. L. Santoso, “Pengembangan Arsitektur VGG16 dan DCNN7 pada Convolutional Neural Network dalam Melakukan Klasifikasi Pose Yoga,” *J. Sist. dan Teknol. Inf.*, vol. 11, no. 2, p. 314, 2023, doi: 10.26418/justin.v11i2.55533.

[10] M. I. Fathur Rozi, N. O. Adiwijaya, and D. I. Swasono, “Identifikasi Kinerja Arsitektur Transfer Learning Vgg16, Resnet-50, Dan Inception-V3 Dalam Pengklasifikasian Citra Penyakit Daun Tomat,” *J. Ris. Rekayasa Elektro*, vol. 5, no. 2, p. 145, 2023, doi: 10.30595/jrre.v5i2.18050.

[11] Stephen, Raymond, and H. Santoso, “APLIKASI CONVOLUTION NEURAL NETWORK UNTUK MENDETEKSI JENIS-JENIS SAMPAH,” *J. Sist. Inf. dan Telemat.*, vol. 7, no. 1, pp. 1–17, 2019.

[12] E. Sutanty, Maukar, D. K. Astuti, and Handayani, “PENERAPAN MODEL ARSITEKTUR VGG16 UNTUK KLASIFIKASI JENIS SAMPAH,” *J. Pendidik. Teknol. Inf.*, vol. 3, no. 2, pp. 407–419, 2023.

[13] R. Permana, H. Saldu, and D. I. Maulana, “Optimasi Image Classification Pada Jenis Sampah Dengan Data Augmentation Dan Convolutional Neural Network,” *J. Sist. Inf. dan Inform.*, vol. 5, no. 2, pp. 111–120, 2022, doi: 10.47080/simika.v5i2.1913.

[14] W. Hutamaputra, R. Y. Krisnabayu, M. Mawarni, N. Yudistira, and F. A. Bachtiar, “Perbandingan Convolutional Neural Network VGG16 dan ResNet34 pada Sistem Klasifikasi Sampah Botol,” *J. Teknol. dan Sist. Komput.*, vol. 10, no. 2, pp. 136–142, 2022, doi: 10.14710/jtsiskom.2021.14045.

[15] J. Nuariputri and P. Sukmasetya, “Klasifikasi Jenis Sampah Menggunakan Base ResNet-50,” *J. Ilm. Komputasi*, vol.

22, no. 3, pp. 379–386, 2023, doi: 10.32409/jikstik.22.3.3380.

[16] M. H. Zayd, M. W. Oktavian, D. G. T. Meranggi, J. A. Figo, and N. Yudistira, “Perbaikan klasifikasi sampah menggunakan pretrained Convolutional Neural Network,” *Teknologi*, vol. 12, no. 1, pp. 1–8, 2022, doi: 10.26594/teknologi.v12i1.2403.

[17] E. Miranda and M. Aryuni, “Klasifikasi Tutupan Lahan Menggunakan Convolutional Neural Network pada Citra Satelit Sentinel-2,” *Sistemasi*, vol. 10, no. 2, p. 323, 2021, doi: 10.32520/stmsi.v10i2.1226.

[18] K. R. Wardani and L. Leonardi, “Klasifikasi Penyakit pada Daun Anggur menggunakan Metode Convolutional Neural Network,” *J. Tekno Insentif*, vol. 17, no. 2, pp. 112–126, 2023, doi: 10.36787/jti.v17i2.1130.

[19] A.-M. Al-Mamun, R. Hossain, M. M. M. Sharmin, E. Kabir, and M. A. Iqbal, “Garbage classification using convolutional neural networks (CNNs),” *Mater. Sci. Eng. Int. J.*, vol. 7, no. 3, pp. 140–144, 2023, doi: 10.15406/mseij.2023.07.00217.

[20] K. M. Sandi, A. P. Yudha, N. D. Aryanto, and M. A. Farabi, “Klasifikasi sampah menggunakan Convolutional Neural Network,” *Indones. J. Data Sci.*, vol. 3, no. 1, pp. 72–81, 2022, doi: 10.28932/jutisi.v8i1.4314.

[21] I. Idawati, D. P. Rini, A. Primanita, and T. Saputra, “Klasifikasi Kanker Payudara Menggunakan Metode Convolutional Neural Network (CNN) dengan Arsitektur VGG-16,” *J. Sist. Komput. dan Inform.*, vol. 5, no. 3, p. 529, 2024, doi: 10.30865/json.v5i3.7553.

[22] E. Tanuwijaya and A. Roseanne, “Modifikasi Arsitektur VGG16 untuk Klasifikasi Citra Digital Rempah-Rempah Indonesia,” *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 21, no. 1, pp. 189–196, 2021, doi: 10.30812/matrik.v21i1.1492.

[23] J. Alberto and D. Hermanto, “Klasifikasi Jenis Burung Menggunakan Metode CNN ResNet-50,” *J. Tek. Inform. dan Sist. Inf.*, vol. 10, no. 3, pp. 34–36, 2023.

[24] S. Suprihanto, I. Awaludin, M. Fadhil, and M. A. Z. Zulfikor, “Analisis Kinerja ResNet-50 dalam Klasifikasi Penyakit pada Daun Kopi Robusta,” *J. Inform.*, vol. 9, no. 2, pp. 116–122, 2022, doi: 10.31294/inf.v9i1.13049.

[25] I. E. Hendarawan, Riza Ibnu Adam, and Chaerur Rozikin, “Klasifikasi Retak Ban Kendaraan Menggunakan Arsitektur ResNet50,” *SATIN - Sains dan Teknol. Inf.*, vol. 9, no. 1, pp. 22–32, 2023, doi: 10.33372/stn.v9i1.902.