Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 6, No. 4, Agustus 2025, Page. 2367-2379
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4770

Garbage Image Classification Using Deep Learning: A Performance Comparison of InceptionResNetV2 vs ResNet50

Rismiyati*[1], Axelliano Rafael Situmeang[2], Khadijah[3], Sukmawati Nur Endah[4]

[1,2,3,4]Department of Informatics, Faculty of Science and Mathematics, Universitas Diponegoro, Indonesia

Email: [1]rismiyati@live.undip.ac.id

Received : May 23, 2025; Revised : Aug 4, 2025; Accepted : Aug 12, 2025; Published : Aug 19, 2025

Abstract

Garbage problem is a worldwide problem. Efforts to address garbage problem have been performed in several aspect, including automatic garbage classification to support automatic garbage sortation in small scale. In the field of garbage classification, deep learning has been widely used because of its ability to learn feature and also to classify with high accuracy. Several promising architectures in deep learning such as ResNet50 and InceptionNet have been used for this classification task. InceptionResNet is introduced to combine the strength of both architectures. This research aims to classify Garbage Classification data set which consist of 15150 images from 12 classes by using InceptionResNetV2 architecture. In addition, experiment by using ResNet-50 is also performed to provide comparison of its performance. During experiment, Hyperparamater tuning was performed, namely the learning rate, dropout rate, and the number of neuron in the dense layer. The results show that InceptionResNetV2 outperform ResNet50 in all scenarios. This architecture is able to achieve highest accuracy of 97.54%. Even though the classification time is longer for InceptionResNetV2, this finding is able to prove the outstanding performance of InceptionResNetV2 in garbage classification. This study contributes to the field of garbage classification by introducing robust and better model for better classification.

Keywords : *Deep Learning, Garbage Classification, InceptionResNet, ResNet50.*

1. INTRODUCTION

Garbage problem remains being one of concern, as stated by the world bank. Yearly, the world produces more than 2.24 billion tons of solid waste[1]. This problem also happens in Indonesia. Indonesia produces 68.5 million tons of waste in 2022, some of which pollute the river and ocean[2]. In fact Indonesia is the second largest contributor of marine plastic after China. [3]

To respond to this problem, researchers have started to develop automated waste classification systems. The purpose of this is to support more effective waste management system, be it in home usage or industrial usage. The early system used classical machine learning to differentiate different kinds of garbage. The algorithm used in the early system are Support vector machines (SVM), K-Nearest Neigbors (KNN), Random Forest (RF), which used another process to create feature for classification.[4]

Deep learning techniques, such as CNNs, can eliminate the need for separate feature extraction by learning hierarchical representations directly from raw image, hence lead to improved classification accuracy and robustness. Several studies have been performed to classify garbage by using CNN-based models. ResNet50, Xception, and DenseNet121 models were used to classify six classes of garbage, with ResNet50 achieving the highest classification accuracy of 87.50% [5]. VGG architecture has also been used to classify garbage with accuracy of 96.21%[6]. Another research is performed by using InceptionNet which achieve slightly higher accuracy than VGG with 96.23%[7].

Vol. 6, No. 4, Agustus 2025, Page. 2367-2379
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4770

Deep learning techniques has shown promising results in classifying garbage image. Several promising architectures based on previous research are ResNet50 and InceptionNet. ResNet50 relies on residual network to ensure that no vanishing gradient problem occurs, hence optimize learning process. On the other hand, InceptionNet can learn multi feature during training, thus able to achieve good results. To combine the strength of both architecture, researchers create inceptionResNet architecture[8]. InceptionResNet has shown better performance as compared to InceptionNet and ResNet50 in several cases, such as acrosome reaction[9].

The growing usage of deep learning algorithms in garbage classification are mostly performed by previously found architecture, such as ResNet-50. Much recent architecture, such as InceptionResNEtV2, introduced multi-scale features which is able to extract more complex features and differentiate more diverse type of garbage as compared to ResNet50. Based on this finding, this research aims to perform garbage classification by using InceptionResNetV2. The main contribution of this study is to improve the accuracy and robustness of garbage image classification by combining the strengths of residual learning and multi-scale feature extraction, as embodied in the InceptionResNetV2 architecture. In addition, this research also compare the InceptionResNetV2 with its base architecture, ResNet-50.

2. METHOD

The research method in this research is shown in Figure 1. The research is performed by using Garbage Classificarion Dataset. The images in the data will underwent preprocessing steps which consist of resizing of the image, image augmentation, and data splitting. The preprocessed images is trained by using InceptionResNetV2 and also ResNet-50 Architecture. During the training, hyperparameter tuning is performed. The models resulted from the experiment is evaluated to get the best model

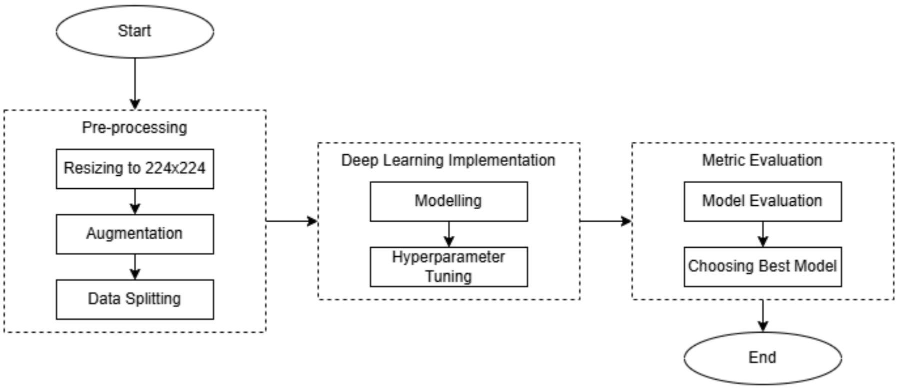


Figure 1. Research Process

2.1. Datasets

The datasets utilized in this study is Garbage Classification dataset and were obtained from Kaggle [10]. This dataset consists of 15150 images from 12 categories of waste types. The example of the image in this data set are shown in Figure 2. The primary challenge presented by these datasets is the severe class imbalance, as illustrated in Figure 3. The number of sample in clothes category is the biggest amongst all classes, with total of 5325 images. The least number of data is owned by class brown glass which only has 602 Images.

In this research, even though there exists the imbalanced class distribution, no modifications such as oversampling or under sampling were applied in the data. Instead, a dropout layer was used at the final stage of the models to ensure no overfitting.

Vol. 6, No. 4, Agustus 2025, Page. 2367-2379
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4770

Figure 2. Examples of images in the dataset for each category

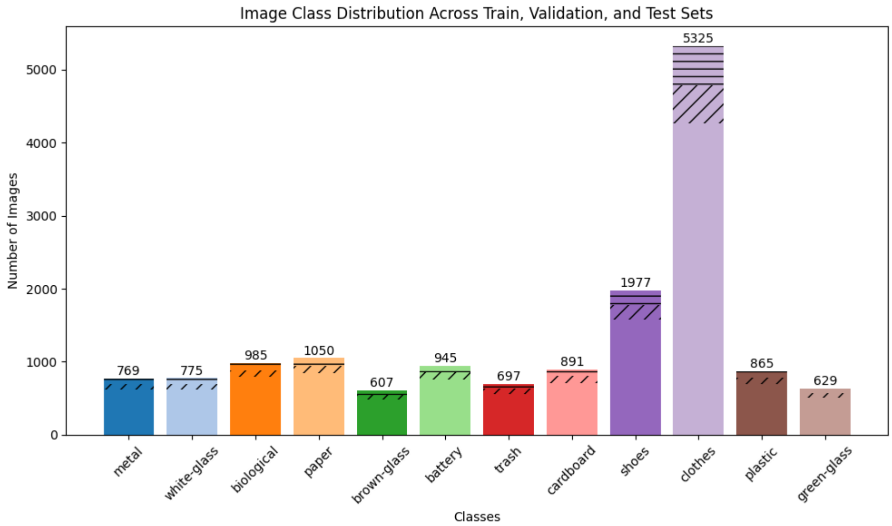


Figure 3. Image Class Distribution

2.2. Preprocessing

The initial data underwent preprocessing step. The preprocessing step involved are resizing all images to a 224×224 pixel format to align with the ImageNet standard [11] for both models. In addition, to ensure that the experiment can be reproduced, the specific parameters used for image augmentation is performed and shown in Table 1. Figure 4 shows the difference between original image and augmented image for some classes are shown in Figure 4. The last step in preprocessing is data split. In this experiment, 80% of the total data is used as training data, 10% for validation and the rest 10% is for testing. The data split is also performed with stratified data split, to put into account the number of sample in train, validation, and testing data.

Table 1. Augmentation Value

Augmentation	Value
rotation	30^{O}
zoom	0.2
width_shift	0.2
height_shift	0.2
horizontal_flip	True
vertical_flip	True
fill_mode	nearest

Vol. 6, No. 4, Agustus 2025, Page. 2367-2379
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4770

Figure 4. Augmentation result for class brown glass, cardboard, clothes and green glass

2.3. Deep Learning Training Implementation with Convolutional Neural Networks

The next step in the experiment is to perform training on the 80% of data. The training is performed by using Convolutional Neural Networks (CNN) architectures, InceptionResNetV2. As comparison, the data are also trained by using ResNet-50.The experimental set up is shown in Figure 5. Figure 5 presents the architecture of the proposed image classification system for waste images, which utilizes two Convolutional Neural Network (CNN) models, namely InceptionResNetV2 and ResNet50. The input image, resized to 224×224 pixels, is simultaneously processed through both CNN architectures to extract high-level visual features. The extracted features from each model are then combined and passed through a dense layer that functions as the final classifier. This approach is designed to take advantage of the strengths of each CNN model, with the expectation of improving overall classification accuracy by leveraging richer feature representations.

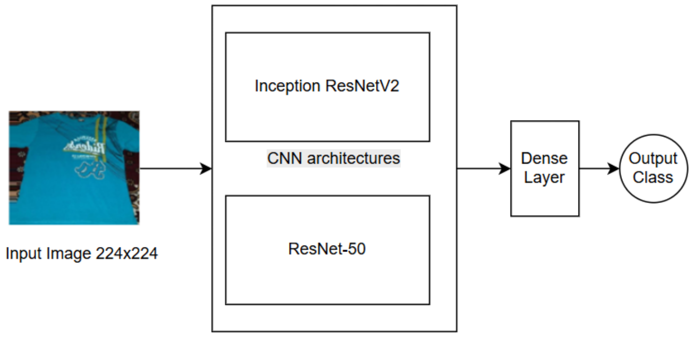


Figure 5. Experimental Set up for training process

Vol. 6, No. 4, Agustus 2025, Page. 2367-2379
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4770

CNN are a specialized architecture, derived from Multi-Layer Perceptrons (MLP), specifically designed to process visual input [12]. Inspired by the hierarchical processing mechanism of the human visual system, CNNs interpret information progressively, from general features to more specific ones [13]. CNN is formed by using multiple types of layers, including convolutional layers, pooling layers, fully connected layers, and dropout layers, each serving distinct functions.

Over time, the original CNN architecture has been extended and enhanced into more sophisticated forms to improve performance and address the limitations of conventional CNN architectures , notably through developments such as ResNet, introduced by He et al. [14], and Inception-ResNet-v2, proposed by Szegedy et al. [15].

One of the main benefits of deep convolutional neural networks (CNN) in image classification is their ability to learn features step by step. The early layers in CNN usually learn simple features like edges, corners, and textures. Meanwhile, the deeper layers enable the network to learn more complex patterns like shapes and object parts. In transfer learning, especially from a big dataset like ImageNet, these early layers already have useful features that can work well for many kinds of data.

In this research, we use this advantage by starting both ResNet-50 and InceptionResNetV2 with weights that have been trained on ImageNet. We do not train the models from the beginning because it needs more data and takes a lot of time. Instead, we use fine-tuning to adjust the model to fit our garbage classification dataset. This method makes the training process faster and helps avoid overfitting, especially because our dataset is not very big and has unbalanced classes.

2.3.1. ResNet-50

Residual Networks (ResNet) were proposed by He et al.[14] to address the vanishing gradient problem in deep neural networks. This problem arises when gradients become exceedingly small during backpropagation, particularly in complex models, there by preventing effective weight updates. As a consequence, deeper networks may fail to properly propagate information across layers, ultimately diminishing the overall performance of the model. To mitigate this issue, He et al. introduced the concept of skip connections, which enable the direct transfer of information across layers, as illustrated in Figure 6 [5]. These skip connections create a network where the output of a layer, denoted as y, can incorporate contributions from several preceding layers, such as F(x), thus facilitating more effective training of very deep networks. By preserving the flow of gradients during training, skip connections allow ResNet to successfully train architectures with hundreds or even thousands of layers without suffering from performance degradation. One of the widely used variants, ResNet-50, consists of 50 convolutional layers and employs bottleneck blocks to improve computational efficiency while maintaining high accuracy, making it a popular backbone for various image recognition and computer vision tasks[16], [17] .

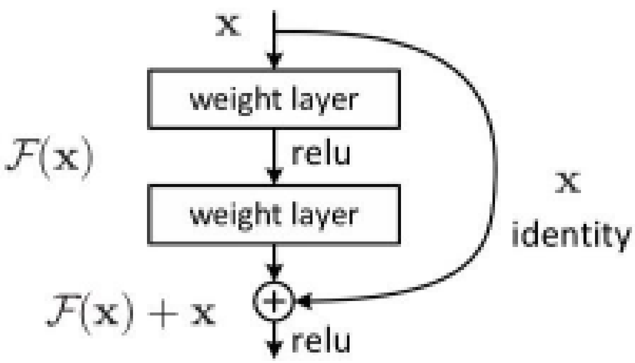


Figure 6. Residual Learning proposed by He et al.[14]

$$y = F(x) + x \tag{1}$$

Vol. 6, No. 4, Agustus 2025, Page. 2367-2379
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4770

2.3.2. InceptionResNetV2

The InceptionResNetV2 architecture [15] is an architecture that fuses the Inception-V2 [15] with the ResNet [14] models. Inception-V2 introduces enhancement improvement by applying convolution integral solution to reduce the number of parameter quantity and to accelerate the calculation process. On the other hand, ResNet proposed the concept of bypassing input data directly to the output by introducing shortcut connections, shifting the learning objective from mapping the target directly to learning the residual between the input and output. The hybrid approach allows the network to take advantage of the strengths of both models, effectively reducing training time and mitigating the vanishing gradient problem. Inception-ResNet-v2 employs residual connections in place of filter concatenation, leading to faster training and enhanced overall performance [15].

The Inception-ResNet model is composed of several blocks that include convolutional layers, filter concatenation, ReLU activation functions, and integrated ResNet and Inception modules, as illustrated in Figure 7. This architecture is specifically designed to address the limitations of its predecessors and achieve high accuracy in image classification tasks [18], [19].

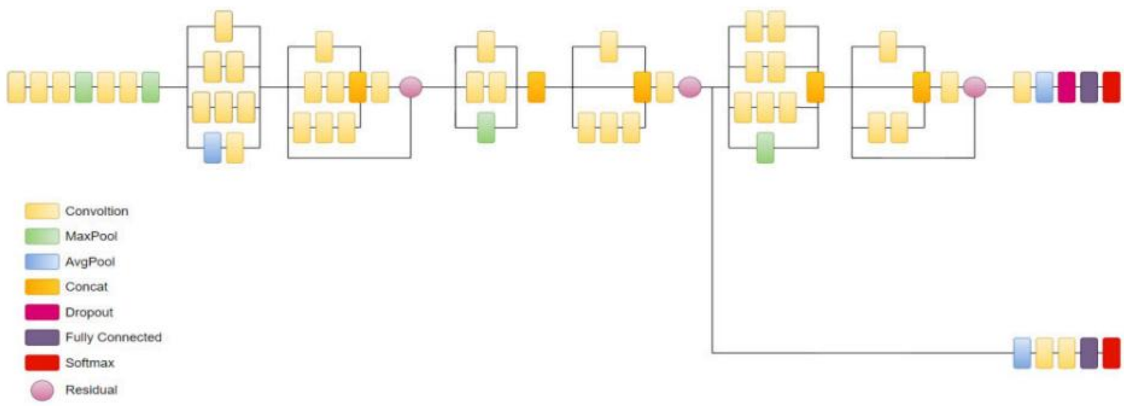


Figure 7. Structure of Inception-ResNet-V2 backbone network [15]

2.4. Fine Tuning

Fine-tuning pre-trained convolutional neural networks (CNN) has become a prevalent strategy in image classification tasks, particularly when dealing with limited labeled data. This approach leverages knowledge from large-scale datasets to enhance performance on specific target tasks [20], [21]. Fine-tuning is a widely used technique in large convolutional deep learning models to customize a pretrained model for a new or unfamiliar dataset [22]. This method involves using a model that was originally trained on a large-scale dataset like ImageNet and refining it by continuing training on a smaller, domain-specific dataset to learn relevant new features. By leveraging the knowledge gained from training on the extensive dataset, fine-tuning facilitates the transfer of learned representations to a new problem, enabling faster and more efficient learning [23].

2.5. Loss Function

Loss Function is a measurement about how close model predictions as compared to the real labels [24]. The loss function used is The Categorical Cross-Entropy (CCE). It quantifies the dissimilarity between the true label distribution and the predicted probability distribution output by the model.

$$L(y,\widehat{y}) = -\sum_{i=1}^{c} y_i \log(\widehat{y_i}) \qquad (2)$$

Vol. 6, No. 4, Agustus 2025, Page. 2367-2379
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4770

Mathematically, for a single sample, the CCE loss is expressed as in Equation 2 where C denotes the number of classes, yi represents the true label, and y^hat is the predicted probability for class i.

2.6. Hyperparameter Tuning

Hyperparameters are configuration variables that govern the training process and model architecture but are not learned from data [25]. Finding optimal hyperparameter values remains challenging and unpredictable, particularly for complex deep learning models. The optimization process involves selecting the best hyperparameter configuration from a multidimensional search space to maximize model performance on validation data. Unlike model parameters learned during training, hyperparameters must be specified before training begins and significantly influence the learning process, model complexity, and ultimate performance metrics [26].

The hyperparameter tuning in this research is performed to determine the value of optimal neuron in the dense layer, the value of dropout and also the learning rate. The search is performed by using grid search by varying the combination of learning rate (0.0001 and 0.00001), dropout (0.1,0.2, and 0.3), and also dense layer (256 and 128). In total there are 12 combinations as shown in Table 2. Both ResNet-50 and InceptionResNetV2 will be trained using these 12 hyperparameter combination to determine the performance of both layer. The minimum learning rate value is chosen to be 0.00001 in this combination because of computational resources limitation. Value smaller than this number results in slower learning, thus longer time to converge, hence additional computing resources.

Table 2. Hyperparameter Tuning

Scenario	Dense Layer	Dropout	Learning Rate
1	128	0.1	0.0001
2	128	0.1	0.00001
3	128	0.2	0.0001
4	128	0.2	0.00001
5	128	0.3	0.0001
6	128	0.3	0.00001
7	256	0.1	0.0001
8	256	0.1	0.00001
9	256	0.2	0.0001
10	256	0.2	0.00001
11	256	0.3	0.0001
12	256	0.3	0.00001

2.7. Evaluation Metrics

Several evaluation metrics were utilized in this experiment such as, Precission, Recall, F1-Score, and Weighted Accuracy [27]. For the best-performing models, we additionally present the Confusion Matrix and class-wise Precision and Recall expressed as percentages. Precision indicates how well the model is able to correctly classify positive instances, and shows the proportion of true positive(TP) predictions among all positive predictions made by the model. A higher precision value means the model produces fewer false positives(FP). Meanwhile, recall measures how effectively the model can identify all actual positive instances, which reflects its ability to minimize false negatives(FN). The F1-Score is the harmonic mean between precision and recall, and is particularly useful when the dataset is imbalanced. Accuracy measures the overall correctness of the model's predictions by comparing the total number of correct predictions with the total number of samples. However, in multiclass classification with imbalanced data, accuracy may not fully represent the model's performance.

P-ISSN: 2723-3863 https://jutif.if.unsoed.ac.id
E-ISSN: 2723-3871 DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4770

Therefore, weighted accuracy is used to give a more balanced evaluation by considering the accuracy of each class along with its proportion in the dataset, so that the performance of the model can be assessed more fairly. These metrics are applied to the testing dataset and are defined in Equations (3) through (7).

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

$$Recall = \frac{TP}{TP+FN} \quad (4)$$

$$F1 = \frac{2\times Precision * Recall}{Precision+Recall} \quad (5)$$

$$Accuracy = \frac{\Sigma TP+\Sigma TN}{N} \quad (6)$$

$$Weighted\ Accuracy = \frac{\sum_{i=1}^{l} \frac{TP_i+TN_i}{TP_i+TN_i+FP_i+FN_i}}{l} \quad (7)$$

3. RESULT

In this study, InceptionResNetV2 and ResNet-50 are used in imbalanced image classification dataset. Both models were implemented using the Keras API with TensorFlow as the backend. To leverage the benefits of transfer learning, initialization of the models is performed with pre-trained ImageNet weights. A full fine-tuning strategy was employed to adapt the networks to the specific characteristics of the target dataset. Training experiments were executed on the Kaggle Notebooks platform, utilizing an NVIDIA Tesla P100 GPU with 16 GB of VRAM and 29 GB of system memory. An EarlyStopping callback was applied to monitor the validation loss and prevent overfitting during the training process.

Table 3. Metric Results for ResNet-50 and InceptionResNetV2

Scenario	ResNet-50		InceptionResNetV2	
	Average Accuracy	F1-Score	Average Accuracy	F1-Score
1	0.940453	0.966193	0.960517	0.960453
2	0.966343	0.966193	0.971521	0.971562
3	0.954045	0.954182	0.963754	0.963716
4	**0.972816**	**0.972734**	**0.975404**	**0.975442**
5	0.941748	0.942342	0.961812	0.961792
6	0.961165	0.961083	0.972815	0.972770
7	0.952751	0.952634	0.965695	0.965691
8	0.962460	0.962620	0.973462	0.973307
9	0.945631	0.945278	0.970226	0.970374
10	0.966990	0.967094	0.972168	0.972105
11	0.952104	0.951990	0.968932	0.968758
12	0.959223	0.959178	0.970226	0.970102

As presented in Table 3, both architectures achieved their highest performance under Scenario 4, which utilized a learning rate of 0.00001, a 128-unit fully connected (Dense) layer, and a dropout rate of 0.2. Under this configuration, ResNet-50 and InceptionResNetV2 achieved competitive results in terms of both accuracy and F1-score. Notably, InceptionResNetV2 outperformed ResNet-50 by a

P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 6, No. 4, Agustus 2025, Page. 2367-2379
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4770

margin of 0.27% in classification accuracy. However, this performance gain comes at a cost. As shown in Table 4, the InceptionResNetV2 model is approximately three times larger in terms of model size and incurs a 162% increase in inference time per image compared to ResNet-50.

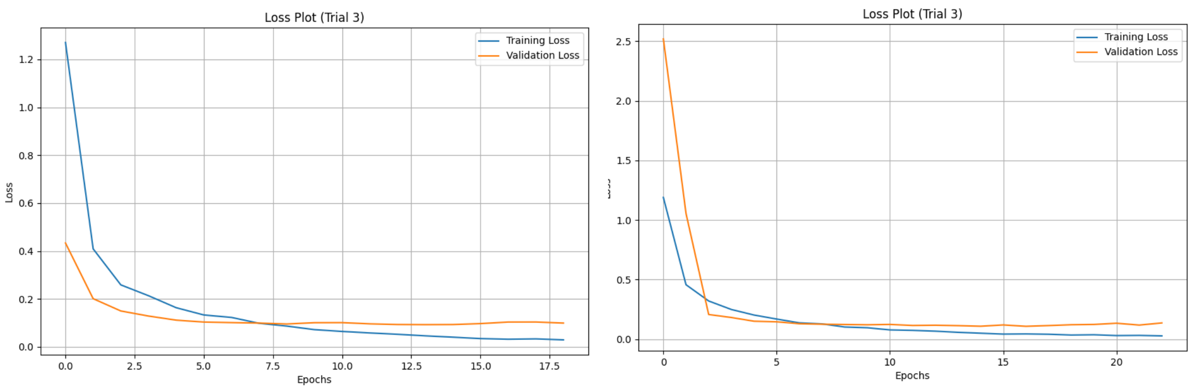
Table 4. Inference Time

Scenario	ResNet-50	InceptionResNetV2
4	140ms/step	367ms/step

These findings highlight a trade-off between classification performance and computational efficiency. While InceptionResNetV2 offers marginally better predictive power, its significantly higher resource consumption may limit its applicability in real-time or resource-constrained environments. Conversely, ResNet-50 provides a more balanced option with respectable accuracy and superior computational performance.

From the best-performing configuration (Scenario 4), both ResNet-50 and InceptionResNetV2 exhibit similar convergence behavior in terms of training and validation loss, as illustrated in Figure 8. However, several noteworthy distinctions are observed during the early training epochs. Initially, the ResNet-50 model experiences a comparatively higher validation loss, which then undergoes a sharp decline within the first three epochs. In contrast, the InceptionResNetV2 model demonstrates a more gradual and steady decrease in both training and validation loss during the same period, indicative of a smoother learning trajectory.

Both models show an intersection point between training and validation loss curves around the sixth epoch. Beyond this point, the validation loss stabilizes for both architectures, while the training loss continues to decrease toward zero. This behavior suggests that the models begin to overfit the training data beyond the sixth epoch. To mitigate this, EarlyStopping was employed, halting training before significant overfitting could occur.The overall trend in the loss curves indicates that while both architectures are capable of generalizing well under the chosen hyperparameter configuration, InceptionResNetV2 may offer a more stable optimization path in the early stages, whereas ResNet-50 demonstrates faster initial convergence despite a higher starting validation loss.



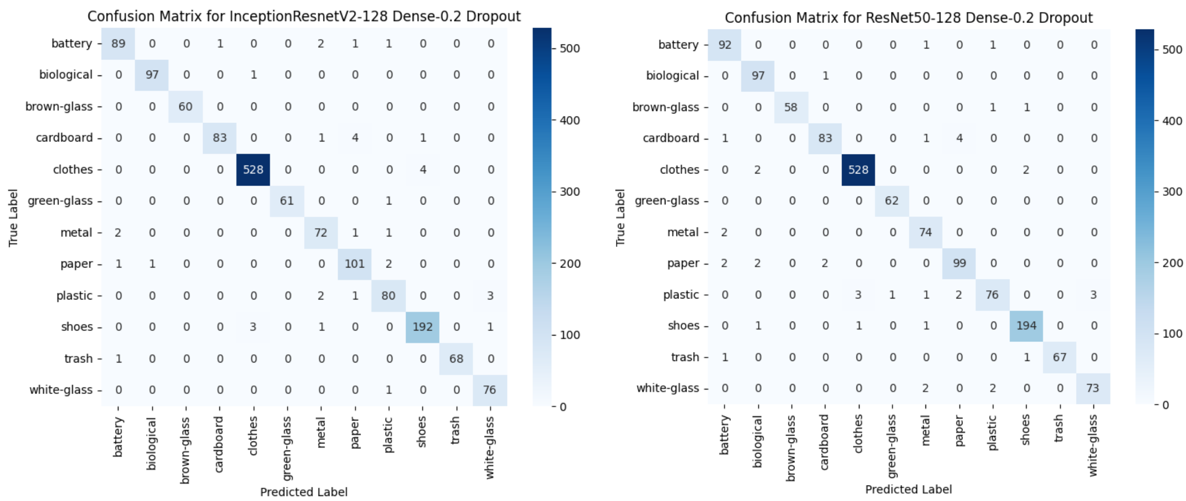
(a) (b)
Figure 8. Loss InceptionResNetV2 (a) versus ResNet50 (b)

In the evaluation of both best-performing models, misclassification rates were found to be minimal across most classes. Although both architectures demonstrate high classification performance, ResNet-50 achieved correct predictions across a greater number of individual classes. This suggests that ResNet-50 possesses stronger generalization capabilities, particularly in handling class imbalance, which is crucial for real-world scenarios involving skewed data distributions. However, when evaluating the total number of correctly classified images, InceptionResNetV2 slightly outperforms ResNet-50,

P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 6, No. 4, Agustus 2025, Page. 2367-2379
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4770

achieving four more correct predictions overall, as depicted in Figure 9. This marginal advantage indicates that while ResNet-50 is more evenly distributed across classes, InceptionResNetV2 may have achieved higher accuracy in undominant classes, leading to a higher total count of correct predictions.

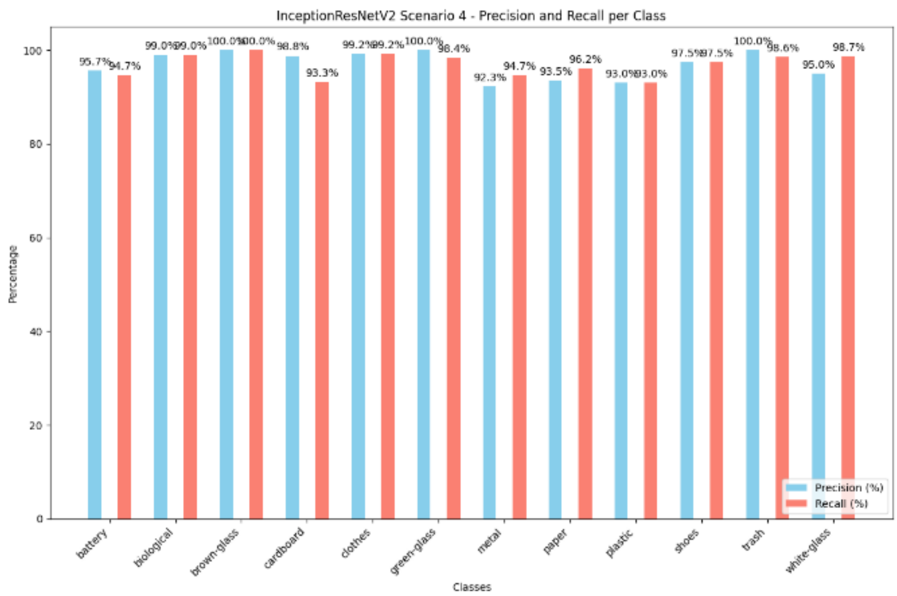
Despite these differences, the Precision-Recall ratio for both models remains consistent, further confirming that both architectures maintain a balanced trade-off between false positives and false negatives under the optimal training configuration. These results reinforce the conclusion that while architectural differences influence class-specific behaviors, overall classification performance remains comparable when properly fine-tuned.



(a) (b)

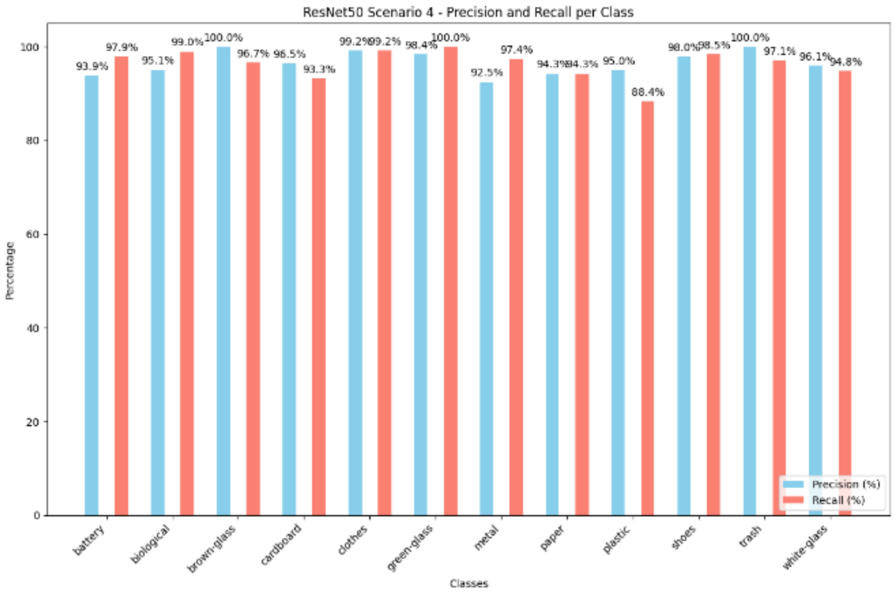
Figure 9. Confusion Matrix InceptionResNetV2 (a) and ResNet50 (b)

The visual comparison at Figure 10 reveals that both models exhibit high accuracy across most categories, with precision and recall values generally exceeding 88%. However, InceptionResNetV2 demonstrates slightly superior overall performance, particularly in maintaining a tighter alignment between precision and recall scores across all classes. Notably, both models achieved perfect recall (100%) on certain classes such as brown-glass and trash, indicating high sensitivity in identifying these categories.



Vol. 6, No. 4, Agustus 2025, Page. 2367-2379
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4770

(a)



(b)

Figure 10. Precission Recall Bar InceptionResNetV2 (a) and ResNet50 (b)

InceptionResNetV2 shows improved recall in classes such as plastic (96.2% compared to ResNet50's 95.0%) and white-glass (98.7% vs. 94.8%), highlighting its enhanced capability to correctly classify underrepresented instances. Meanwhile, ResNet50 marginally outperforms InceptionResNetV2 in shoes recall (100% vs. 97.5%), though it also presents larger gaps between precision and recall in classes like metal and plastic, which may point to an increased rate of false positives or classification ambiguity due to data imbalance. In contrast, InceptionResNetV2 consistently maintains smaller precision-recall gaps, reflecting better handling of imbalanced data distributions and overall classification stability. These findings suggest that while both models are effective, InceptionResNetV2 offers more reliable performance for real-world classification tasks involving imbalanced datasets due to its higher consistency capacity.

4. DISCUSSIONS

This study shows that InceptionResNetV2 consistently perform better than ResNet-50 in terms of overall accuracy and F1-score across all tested scenarios. The best performance was achieved in Scenario 4. While the performance gap may appear small numerically, it reflects the model's ability to handle complex patterns and class imbalance.

In addition to its accuracy, InceptionResNetV2 also demonstrated a smoother learning curve, with more stable training and validation loss. Its precision and recall scores were more balanced across classes, and it showed better recall on several minority classes, such as plastic and white-glass. This indicates not just better raw performance, but also greater consistency and reliability, especially when dealing with imbalanced data. Although InceptionResNetV2 requires more computational resources and longer inference time, the trade-off is justified by its improved classification performance and stability. In scenarios where accuracy, class sensitivity, and generalization are top priorities, InceptionResNetV2 stands out as the more capable and robust architecture. Furthermore, the promising results of this study suggest that InceptionResNetV2 can potentially be applied to other waste-related domains, including recyclable materials, food waste, and medical waste classification.

Vol. 6, No. 4, Agustus 2025, Page. 2367-2379
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.4.4770

5. CONCLUSION

This research performs garbage classification to evaluate the performance of inceptionResNetV2, architecture which theoretically combines the strength of ResNet50 and InceptionV2. The data set used in this research is Garbage classification dataset which consists of 12 classes. The experiment is performed by comparing the performance of InceptionResNetV2 with ResNet-50 for 12 different hyperparameter combinations. These hyperparameter combinations are chosen with model performance and resource limitation in consideration. The experiment results show that InceptionResnetV2 is able to achieve higher accuracy compared to ResNet50 in all hyperparameter combinations. In addition, InceptionResNEt50 is able to achieve the highest accuracy of 97.54%.

Although this system gives good performance, it still has some limitations. First, the InceptionResNetV2 model needs more computing power, so it is not easy to use on devices with low hardware, like edge devices. This problem is important especially in developing countries that don't have strong GPU or computing support. In the future, this model can be improved using methods like model pruning, quantization, or knowledge distillation to make it faster and use less memory.

Second, the dataset in this research is only based on static images. But in the real world, waste classification may use video, different angles, or noisy input. So, this model can be developed further to work with time-based data or combine other information like weight or sound from trash bins.

CONFLICT OF INTEREST

The authors affirm that no conflicts of interest exist among the authors or with the research subject discussed in this paper.

ACKNOWLEDGEMENT

The author would like to thank LPPM UNDIP for funding this research.

REFERENCES

[1] T. W. Bank, "What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050," The World Bank, 2018. [Online]. Available: https://datatopics.worldbank.org/what-a-waste
[2] S. ; L. U. Purnama, " Environment Ministry applies circular economy for waste management."
[3] J. R. Jambeck *et al.*, "Plastic waste inputs from land into the ocean," *Science (1979)*, vol. 347, no. 6223, pp. 768–771, 2015, doi: 10.1126/science.1260352.
[4] J. Prajapati, K. Ribadiya, Y. Shah, K. Patel, B. Patel, and V. Vadhavana, "A Comparative Analysis of Garbage Classification Using ResNet50, Xception and DenseNet121 Architecture: A Review," in *Lecture Notes in Networks and Systems*, 2024, pp. 383–402. doi: 10.1007/978-981-97-2053-8_29.
[5] E. T. Yasin and M. Koklu, "A comparative analysis of machine learning algorithms for waste classification: inceptionv3 and chi-square features," *International Journal of Environmental Science and Technology*, 2024, doi: 10.1007/s13762-024-06233-z.
[6] W. Yanyan, W. Yajie, W. Chenglei, and S. Yinghao, "A Novel Garbage Images Classification Method Based on Improved VGG," in *2022 34th Chinese Control and Decision Conference (CCDC)*, 2022, pp. 1571–1575. doi: 10.1109/CCDC55256.2022.10034310.
[7] T. Gupta *et al.*, "A deep learning approach based hardware solution to categorise garbage in environment," *Complex & Intelligent Systems*, vol. 8, no. 2, pp. 1129–1152, 2022, doi: 10.1007/s40747-021-00529-0.
[8] C. Szegedy, S. Ioffe, and V. Vanhoucke, "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning," *CoRR*, vol. abs/1602.07261, 2016, [Online]. Available: http://arxiv.org/abs/1602.07261
[9] M. Park *et al.*, "Deep Learning-Based Precision Analysis for Acrosome Reaction by Modification of Plasma Membrane in Boar Sperm," *Animals*, vol. 13, p. 2622, Aug. 2023, doi: 10.3390/ani13162622.

[10] M. Mohamed, "Garbage Classification (12 classes)," 2021. Accessed: Apr. 28, 2025. [Online]. Available: https://www.kaggle.com/datasets/mostafaabla/garbage-classification

[11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255. doi: 10.1109/CVPR.2009.5206848.

[12] S. Li, L. Wang, J. Li, and Y. Yao, "Image Classification Algorithm Based on Improved AlexNet," *J Phys Conf Ser*, vol. 1813, p. 12051, Apr. 2021, doi: 10.1088/1742-6596/1813/1/012051.

[13] P. A. Nugroho, I. Fenriana, and R. Arijanto, "IMPLEMENTASI DEEP LEARNING MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK (CNN) PADA EKSPRESI MANUSIA," *ALGOR*, vol. 2, no. 1, pp. 12–20, Nov. 2020, [Online]. Available: https://jurnal.ubd.ac.id/index.php/algor/article/view/441

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," 2015. [Online]. Available: https://arxiv.org/abs/1512.03385

[15] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning," 2016. [Online]. Available: https://arxiv.org/abs/1602.07261

[16] M. Elpeltagy and H. Sallam, "Automatic prediction of COVID- 19 from chest images using modified ResNet50," *Multimed Tools Appl*, vol. 80, no. 17, pp. 26451–26463, 2021.

[17] M. B. Hossain, S. M. H. S. Iqbal, M. M. Islam, M. N. Akhtar, and I. H. Sarker, "Transfer learning with fine-tuned deep CNN ResNet50 model for classifying COVID-19 from chest X-ray images," *Inform Med Unlocked*, vol. 30, p. 100916, 2022.

[18] N. Dominic, T. W. Cenggoro, A. Budiarto, B. Pardamean, and others, "Transfer learning using inception-ResNet-v2 model to the augmented neuroimages data for autism spectrum disorder classification," *Commun. Math. Biol. Neurosci.*, vol. 2021, p. Article–ID, 2021.

[19] J. Wang, X. He, S. Faming, G. Lu, H. Cong, and Q. Jiang, "A real-time bridge crack detection method based on an improved inception-resnet-v2 structure," *IEEE Access*, vol. 9, pp. 93209–93223, 2021.

[20] A. Dash and P. Mohapatra, "A Fine-tuned deep convolutional neural network for chest radiography image classification on COVID-19 cases," *Multimed Tools Appl*, vol. 81, pp. 1–21, Apr. 2022, doi: 10.1007/s11042-021-11388-9.

[21] W. Zhao, W. Jiang, and X. Qiu, "Fine-Tuning Convolutional Neural Networks for COVID-19 Detection from Chest X-ray Images," *Diagnostics*, vol. 11, p. 1887, Apr. 2021, doi: 10.3390/diagnostics11101887.

[22] Y. Wang, Z. Haoxin, and G. Zhang, "cPSO-CNN: An efficient PSO-based algorithm for fine-tuning hyper-parameters of convolutional neural networks," *Swarm Evol Comput*, vol. 49, pp. 114–123, Apr. 2019, doi: 10.1016/j.swevo.2019.06.002.

[23] M. Subramanian, K. Shanmugavadivel, and P. Nandhini, "On fine-tuning deep learning models using transfer learning and hyper-parameters optimization for disease identification in maize leaves," *Neural Comput Appl*, vol. 34, Apr. 2022, doi: 10.1007/s00521-022-07246-w.

[24] L. Ciampiconi, A. Elwood, M. Leonardi, A. Mohamed, and A. Rozza, "A survey and taxonomy of loss functions in machine learning," 2024. [Online]. Available: https://arxiv.org/abs/2301.05579

[25] T. Yu and H. Zhu, "Hyper-Parameter Optimization: A Review of Algorithms and Applications," 2020. [Online]. Available: https://arxiv.org/abs/2003.05689

[26] S. Gupta, "A Comprehensive Study of Hyperparameter Optimization in Deep Learning Models Using Bayesian Optimization," *SSRN Electronic Journal*, Apr. 2024, doi: 10.2139/ssrn.4958505.

[27] P. N. Andono and T. Sutojo, *Pengolahan Citra Digital*. Andi Publisher, 2018. [Online]. Available: https://books.google.co.id/books?id=zUJRDwAAQBAJ