Journal of Applied Informatics and Computing (JAIC)
Vol.9, No.4, August 2025, pp. 1137~1145
e-ISSN: 2548-6861

Leveraging Convolutional Neural Networks for Multiclass Waste Classification

Apriandy Angdresey [1]*, Indah Yessi Kairupan [2], Andre Gabriel Mongkareng [3]
[1,2,3] Departement of Informatics Engineering, Universitas Katolik De La Salle, Manado
aangdresey@unikadelasalle.ac.id [1*], ikairupan@unikadelasalle.ac.id [2], 20013004@unikadelasalle.ac.id [3]

Article Info

Article history:

Received 2025-04-01
Revised 2025-07-04
Accepted 2025-07-07

Keyword:

Machine Learning,
Convolutional Neural Network,
MobileNetV2,
Multiclass Classification,
Waste.

ABSTRACT

Waste management remains a critical challenge globally, particularly in Indonesia, where low public awareness and suboptimal waste handling practices contribute to serious environmental impacts. To address this issue, we propose a computerized waste classification system using a Convolutional Neural Network (CNN) based on the MobileNetV2 architecture. The model classifies waste into five categories: plastic, cardboard, paper, glass, and organic. Leveraging transfer learning, the model achieved a training accuracy of 95.28% and a validation accuracy of 89.48%, with 5-fold cross-validation confirming its robustness. Evaluation on 1,224 unseen images resulted in an overall accuracy of 86.36%, with precision and recall scores of 87.44% and 85.87%, respectively. While class-wise performance varied, the results highlight the potential of lightweight CNNs to improve waste sorting efficiency in Indonesia. Future work will focus on refining performance for minority classes and exploring integration into real-world applications

I. INTRODUCTION

Indonesia's population development may affect people's consumption habits, which might lead to a rise in the amount of waste created. Indonesia is the world's fifth-largest producer of waste, according to the World Bank publication The Atlas of Sustainable Development Goals 2023[1]. Based on data from SIPSN, the amount of waste in Indonesia has increased significantly over the past five years, with a noticeable rise from 2020 to 2023, followed by a slight decline in 2024. This trend indicates that waste management in Indonesia remains poorly regulated and inconsistent [2]. However, in many areas, there is still little awareness of the significance of waste management. Many people still do not have the habit of sorting and managing waste. This may lead to an uncontrollable amount of garbage, harming the ecosystem, human health, and the planet.

Waste is defined as unused materials or items that have run out of usefulness. Waste is often described as things left over after use, such as food no longer fit for eating, paper, cardboard, glass, plastic, and other waste materials. Waste may be broadly classified into two categories: organic and inorganic. Waste can be classified as either organic or inorganic. Organic waste is derived from the remains of living things, such as food waste or plant remains. Inorganic waste takes longer to decompose naturally and includes plastic, glass, cardboard, and other synthetic materials [3], [4].

Sustainable environmental management initiatives greatly benefit from the distinction between different forms of waste, particularly between organic and inorganic waste. In this instance, separating two waste categories may present chances for improved effectiveness and efficiency in management. With the right processes, organic waste may be converted into compost, fertilizer, and livestock feed. However, inorganic waste may also be recycled to create recyclable materials, which are goods that can be used again. Glass, cardboard, plastic, paper, and other materials are among the recyclable inorganic waste products [5].

The artificial intelligence field of machine learning (ML) aims to create computer systems that can learn and adapt without requiring explicit programming. In machine learning, deep learning is a scientific discipline that uses information from raw data to create complex patterns using algorithms or statistical models. Deep learning (DL) employs stacks of layers to comprehend and extract information from complicated data while creating neural network models. The more intricate the feature representations the model can learn, the deeper the network architecture. Neural Networks possess a flexible structure that indicates a given layer will move to the left of another layer. Neural Networks are based on concepts from the field of neurobiology and are inspired by human beings' ability to understand objects. Research [5] presents a deeper understanding of the significance of ML and DL, as well as related models, in building the structural layers of the evaluated data.

Computer vision is artificial intelligence (AI) that arises from applying many scientific disciplines that explain how computers might get a high-level understanding of digital images or movies. Computer vision has been widely used in various applications, including image detection and categorization. In research [6], the author created a North Sulawesi regional musical instrument program named Kolintang, employing a camera as a PC motion detection sensor. The success of this application introduces a new dimension to playing the Kolintang musical instruments on a portable computer. [7] Developed a Kinect-based training system for slow practice. This technology was designed to replace gym trainers, allowing consumers to train autonomously. The system presents a grading technique for determining the trainer and user's posture and timing similarities. The system can be accurately evaluated by running experiments in real-time. Furthermore, the grading technique can determine how similar the tutor and user's dancing stances and postures are, as well as their dancing times.

An artificial neural network, often known as a neural network, is a concept in ML that functions similarly to the structure and operation of the human brain [8], [9]. Convolutional Neural Networks (CNN), which can recognize and categorize picture objects, are one example of how neural networks are used in this situation [10], [11]. Using ML technology to manage and categorize different kinds of waste is a good first step toward an effective solution [12]. Using deep learning technology, the CNN algorithm is one such approach. CNN can recognize objects or specific areas in an image and classify photos and movies. The CNN approach has been applied in numerous studies [13]. The author analyzes three CNN designs to categorize moving objects in traffic CCTV footage. RetinaNet, EfficientNet, and SSD MobileNet are the designs that are compared. The author attempts to categorize trucks, vehicles, motorcycles, and people. This advancement highlights MobileNet's potential as a lightweight yet effective solution for dermatological diagnostics.

ML enables systems to identify and detect waste and categorize it into organic and inorganic categories. By identifying waste that is recyclable or deserving of recycling within the inorganic waste, this method can assist in optimizing the recycling process. One approach that can address issues with increasing waste management efficiency is the incorporation of ML into the categorization and detection of waste.

In [14], the effectiveness of waste classification. The study compares Res-Net50, achieving 87% accuracy, and Mobile-NetV2, reaching 89.26%, with their own custom 6-layer CNN model that achieves 91% for binary biodegradable/non-biodegradable classification. The previously outlined issues serve as the foundation for developing web-based apps. The developed application can efficiently classify five types of waste: plastic, paper, cardboard, glass, and organic. In addition, it will incorporate a feature that suggests ways to recycle waste to raise environmental awareness, enable waste to be transformed into something more useful, and lower the amount of waste produced in Indonesia.

The rest of this article is arranged as follows: Section 2 focuses on the methodology applied. Section 3 explores the implementation of the system, followed by Section 4, which discusses the testing process and the results achieved. Eventually, Section 5 presents the conclusions and suggestions for future work.

II. METHOD

A. Datasets

The dataset comprises 13,059 images across five waste categories, sourced from Kaggle's "Waste Classification Data" and "Waste Classification". The distribution of classes is as follows:

TABLE 1
DATASET

Class	Number of Images	Percentage
Plastic	2500	19.15%
Cardboard	1800	13.79%
Paper	3000	22.98%
Glass	2759	21.13%
Organic	3000	22.98%

Data augmentation techniques, including rotation, zoom, and flipping, were applied during training to mitigate class imbalance. The dataset was split into 80% training (10,447 images) and 20% validation (2,612 images).

B. Multiclass Classification

The multiclass classification method is one of the ML parts involving classifying more than two classes. The classification method was introduced initially for binary classification with only two classes; however, real-world problems affect more than binary classification. To solve this, multi-class classification is an essential aspect of ML to solve real-world problems with more than two possible outputs or classes[15].

C. TensorFlow

TensorFlow is a powerful library for numerical computation and ML. It allows users to represent complex algorithms and perform tasks based on data stored in specific objects. TensorFlow is commonly used for tasks like building and training neural networks, performing mathematical operations on large datasets, and solving various ML

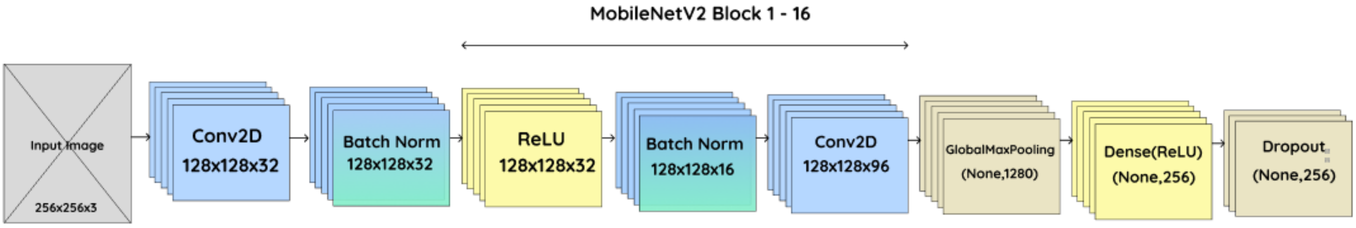

Figure 2. Custom CNN Architecture

problems[16], [17].

D. Proposed Model

The architecture chosen for transfer learning is MobileNetV2. This architecture is renowned for its computational efficiency and fast inference speed, critical for deploying edge devices with limited resources. MobileNetV2 employs depth-wise separable convolutions and inverted residual blocks, significantly reducing parameter count (3.4 million parameters) while maintaining competitive accuracy. Compared to heavier architectures like ResNet-50 (25 million parameters) or VGG16 (138 million parameters), MobileNetV2 balances performance and efficiency, making it ideal for large-scale waste classification tasks.

Figure 1 shows that there will be two processes: preparing the dataset and building the DL model. The program takes an image as input with a size of *256x256* pixels and RGB color depth, using the Python programming language and the TensorFlow library to build this program. In preparing the dataset process, the dataset with five total classes is divided into two directories: training and validation data for the training model process; the dataset is divided into 80% training data and 20% validation data. Using TensorFlow to do this preparation process ensures the data in each directory is balanced, so there will be no bias during the training process.

After the dataset is divided into two directories, the next step is to augment the training data to enrich it with variations of the images within the training data so it can represent the real-world case. Data augmentation involves various transformations, such as rescaling, rotating, cropping, flipping, and zooming the original images to create new, slightly modified versions. Applying this augmentation method to the original images can help expose the model to a broader range of scenarios and conditions, making the model more capable of generalizing well to unseen data.

This program has two main parts in the classification process: building and fitting the model. The model is built by using transfer learning from TensorFlow. The model would be equipped with network parameters such as weight and bias configured according to the dataset earlier through the training process with the optimization method. The model training process will generate a CNN-based model ready for classification.

The program will use MobileNetV2 as the base layer, but with a few modifications, where the top layer of the architecture will not be used, and the weights of the layers are frozen; with this, we can use the architecture as transfer learning and then add a custom layer as shown in Figure 2. Afterward, the custom model based on the MobileNetV2 architecture is built, and the dataset containing images is fed into the model for training. The output of this training process is a model ready for classification.

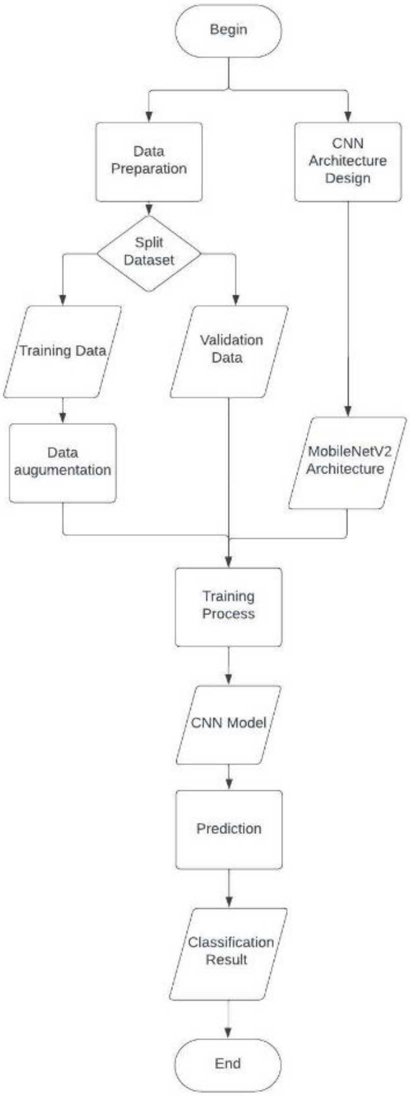

Figure 1. System Design Flowchart

1) Rectified Linear Units (*ReLU*) is an activation function with strong biological and mathematical underpinnings. It works by thresholding the value at *0*, for example, *f(x)=max (0, x)*. simply put, the outputs will be 0 when

x<0, and conversely, it outputs a linear function when x≥0, as present in Figure 3. The *ReLU* activation function will produce zero as an output when *x<0* and produce a linear output with a slope of *1* when *x>0* [18]. The *ReLu* activation function is used in neural networks, facilitates faster training, encourages sparsity, and avoids the vanishing gradient problem.

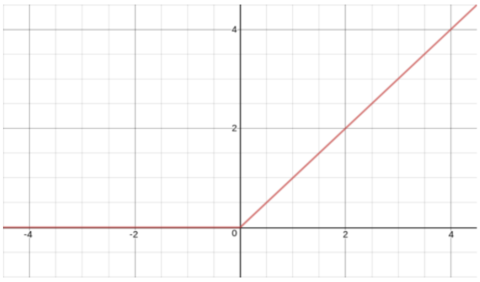

Figure 2 ReLU

2) The *Softmax* function is a DL solution to classification problems usually employing the *softmax* function as a classification function (as the last layer) and is more useful for multi-class classification problems because it provides a clear interpretation of model outputs as probabilities, allowing for straightforward decision-making based on the highest probability class. The *softmax* function specifies a discrete probability distribution for K classes, denoted by $\sum_{k=1}^{K} Pk$ [19], [20].
3) The loss function was introduced as an algorithm to measure how much the error rate is in a model. The output of the neural network model came in the form of a confidence score, which suggested that a bigger correct confidence score proved to be better. It quantifies the error between the predicted outputs of a model and the actual ground truth labels. Measuring the model's performance helps the learning process [21]. Categorical cross-entropy loss measures the dissimilarity between the true distribution (the actual labels) and the predicted distribution (the output probability generated by the *softmax* function). Categorical cross-entropy is also commonly applied in classification cases, where L is the sample loss value, and j is the label. Whereas the target value is denoted by y, and the predicted value is denoted by $\hat{y}$, below is the equation of this loss function [22], [23]:
$$\boldsymbol{L_i = \sum_j y_{i,j} - \log(\widehat{y_{i,j}})} \qquad \ldots 1)$$
4) Optimization is crucial in DL; it finds the best parameters (weight, bias, and learning rate) to minimize or maximize certain objective functions. The optimization method is an algorithm that updates the parameters of an artificial neural network to reduce the loss rate [24]. Adaptive Momentum Estimation (Adam) is an algorithm that can be used as a replacement for the classical SGD optimization algorithm. Adam combines *RMSProp* and Stochastic Gradient Descent (SGD) with momentum. The main component of Adam relying on the adaptive learning rates is that Adam can dynamically adjust the learning rate for each parameter based on the first and second moments of gradients. Adam also incorporates momentum, which helps accelerate optimization by accumulating weighted averages of past gradients. Bias correction is also applied inside Adam; it estimates the first and second moments of gradients, especially during the early iterations when the estimates are biased toward zero. Adam is commonly used because it converges faster than traditional optimizer algorithms such as SGD and requires less parameter tuning[25]. Below is the equation of Adam:
$$\boldsymbol{m_{i,t} = \beta_1 m_{t-1} + (1-\beta_1)\frac{\partial L_t}{\partial w_{i,t}}} \qquad \ldots 2)$$
$$\boldsymbol{v_{i,t} = \beta_2 m_{t-1} + (1-\beta_2)(\frac{\partial L_t}{\partial w_{i,t}})^2} \qquad \ldots 3)$$
$$\boldsymbol{\hat{m}_{i,t} = \frac{m_{i,t}}{1-\beta_1^t}} \qquad \ldots 4)$$
$$\boldsymbol{\hat{v}_{i,t} = \frac{m_{i,t}}{1-\beta_2^t}} \qquad \ldots 5)$$
$$\boldsymbol{W_{i,t} = W_{i,t-1} - \frac{n}{\sqrt{\widehat{v_{i,t}}+\varepsilon}}\widehat{m_{i,t}}} \qquad \ldots 6)$$
The Adam optimization algorithm's weight parameter $\boldsymbol{(W)}$ represents the updated value during training. The learning rate $\boldsymbol{(\eta)}$ controls the step size for these updates, ensuring stable convergence. The term $\boldsymbol{\frac{\partial L_t}{\partial w_{i,t}}}$ denotes the gradient of the loss function with respect to the weight, determining the direction and magnitude of updates. To improve optimization, Adam maintains a first-moment estimate $\boldsymbol{\hat{m}_{i,t}}$, which acts as momentum by averaging past gradients, and a second-moment estimate $\boldsymbol{\hat{v}_{i,t}}$ which tracks the variance of gradients to normalize updates. Since both moments start at zero, bias corrections $\boldsymbol{\hat{m}}$, and $\boldsymbol{\hat{v}}$, is applied to counteract the initial bias in early iterations. A small constant $\boldsymbol{(\varepsilon)}$ is added to prevent division by zero, ensuring numerical stability. The hyperparameters $\boldsymbol{\beta_1}$ *and* $\boldsymbol{\beta_2}$ control the decay rates for the first and second moments, allowing Adam to balance fast convergence with stable weight updates.

E. Validation Method

The next step of this study is Validation. This step should determine whether the proposed method performs effectively based on the previously processed datasets, including the 13.059 images split into five classes: plastic, cardboard, paper, glass, and organic.

1) A confusion matrix is a method in ML and DL that evaluates or measures the performance of the classification model, where the output can be two or more classes. The confusion matrix itself is a table that consists of 4 predicted and actual value combinations, and that is: TP: True Positive, FP: False Positive, FN: False Negative, TN: True Negative [26]. The dimensions of a table depend on the number of model classes it has.
2) The Receiver Operating Characteristic (ROC) Curve is a graphical representation that shows how well a model can distinguish between two classes by varying the decision

threshold. A higher TPR (True Positive rate) and lower FPR (False Positive rate) indicate better performance, as the model effectively identifies true positives while minimizing false positives. In the case of multi-class classification, the ROC curve can still be utilized; it is approachable by computing each class individually using a one-vs-all approach. The ROC curve is an effective method to evaluate diagnostic test quality or performance. The under-the-curve area is a measurement of test accuracy. The value varies between *1* (excellent) and *0.5* (failed), which means it fails to function as it should [26].

3) Area Under the Curve (AUC) is a metric used to evaluate the performance of the ML model; it represents the area under the ROC curve. AUC is usually used for binary classification, but AUC can still be used to compute each class individually using the one-vs-all approach. AUC ranges from *0* to *1*, with higher values indicating better performance. An AUC of *0.5* means the model is random guessing, while an AUC of *1* indicates perfect classification [27].

III. Results and Discussion

A. Optimization Method Testing

Training was conducted using other optimization methods to determine whether the suggested optimization method is suitable. The optimization method tested was an optimization method that has adaptive learning rate mechanisms such as Adam, *RMSProp*, and *AdaGrad*. Moreover, the testing uses a custom model that uses MobileNetV2 as a base layer. The training process for this testing was carried out in as many as 10 epochs with a batch size of *32*. The input size of images is *256x256* with RGB color depth. The best results from testing each optimization method are compared with each other to determine the most suitable one.

Table 2
Optimization Testing

Optimization	Train Accuracy	Train Loss	Validation Accuracy	Validation Loss
RMSProp	*0.8681*	*0.3559*	*0.8771*	*0.3546*
Adam	*0.8636*	*0.3555*	*0.8874*	*0.3047*
AdaGrad	*0.7974*	*0.5300*	*0.8541*	*0.3866*

Based on the provided results, Adam's optimization achieved slightly higher training accuracy (86.36%) compared to RMSprop (86.81%) and *AdaGrad* (79.74%). Additionally, Adam also exhibited lower training loss (0.3555) compared to RMSprop (0.3559) and *AdaGrad* (0.53). However, when evaluating validation performance, Adam outperformed the other optimizers with a validation accuracy of 88.74% and a validation loss of 0.3047. Therefore, based on these observations, Adam optimization may be the most suitable method among the tested optimizers for this model and dataset.

B. Evaluation

The evaluation process was conducted using the Adam optimizer, as it demonstrated better performance than *RMSProp* and *AdaGrad* during the optimization testing phase. The evaluation was performed on a dataset over 30 epochs with a batch size of 32, using a customized MobileNetV2 architecture for a classification task with five labels. The results are presented below, including the model's confusion matrix, model accuracy, loss metrics, precision metrics, recall metrics, ROC curve, and AUC metric.

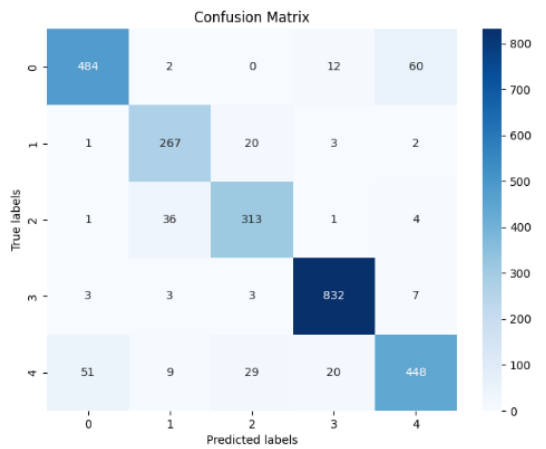

Figure 3 Confusion Matrix

The confusion matrix evaluates the model's performance for waste classification, as shown in Figure 3. The model classifies organic waste (class 3) with high accuracy (832 correct predictions), while misclassifications occur primarily between plastic (class 4) and glass (class 0) (51 instances) and between paper (class 2) and cardboard (class 1) (36 cases). These errors likely result from visual similarities between materials. While overall classification performance is satisfactory, further improvements, such as enhanced feature extraction and data augmentation, could reduce misclassification rates.

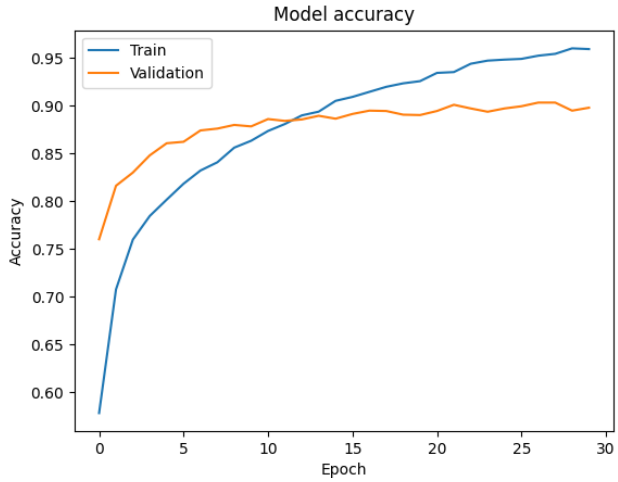

Figure 4 Model Accuracy

Moreover, Figure 4 presents the accuracy curve illustrating the training and validation performance of the model over 30 epochs. The training accuracy steadily increases, reaching approximately 95%, while the validation accuracy improves rapidly in the initial epochs before stabilizing around 90%. The small gap between training and validation accuracy suggests that the model generalizes well, with minimal overfitting. However, slight fluctuations in validation accuracy indicate potential variations in model performance, which could be addressed through regularization techniques or additional data augmentation.

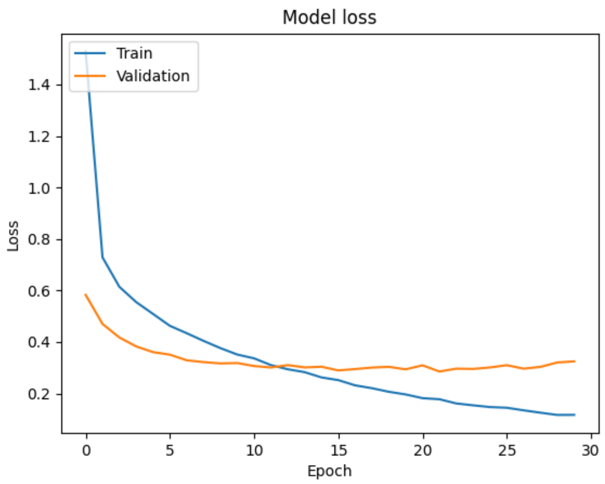

Figure 5 Loss Metrics

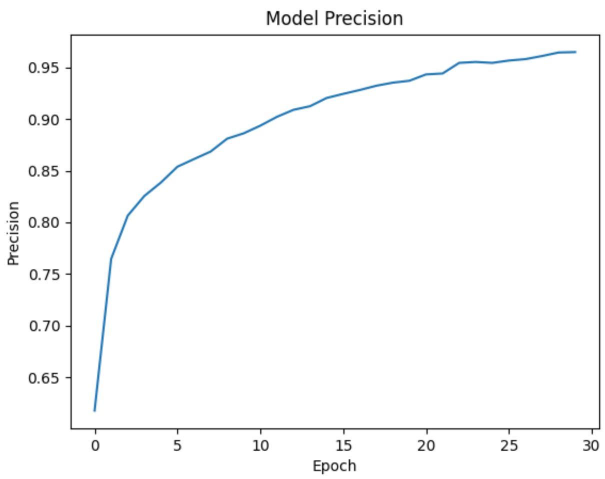

Figure 6 Precision Metrics

The loss metric graph, as demonstrated in Figure 5, illustrates the model's performance during training and validation over 30 epochs. Initially, training and validation losses start high (above 1.0) but decrease rapidly in the first few epochs, indicating effective learning. The training loss continues to decline steadily, reaching a low value (~0.2), while the validation loss stabilizes at approximately 0.4, suggesting the model generalizes well without severe overfitting. The small gap between the two curves implies balanced training. However, the slight increase in validation loss near the end may hint at minor overfitting, which could be addressed with techniques like early stopping or additional regularization. Figure 6 presents the precision metric graph and demonstrates the model's ability to correctly classify waste categories over 30 epochs. Starting at ~0.75, precision increases steadily and stabilizes near 0.95 by the final epochs, indicating significant improvement in the model classification accuracy. The consistent upward trend without significant fluctuations suggests stable learning, with the model becoming increasingly reliable at minimizing false positives. This high precision reflects the model's effectiveness in correctly identifying waste types, particularly for classes like glass and paper. However, lower precision for materials like cardboard reveals room for targeted refinement.

Figure 7 shows the recall graph demonstrating the model improvement in detecting waste classes, rising from 0.6 to 0.9 over 30 epochs. The steady climb indicates fewer false negatives, though early lower values, especially for materials like cardboard, highlight initial challenges. The high final recall confirms strong detection performance.

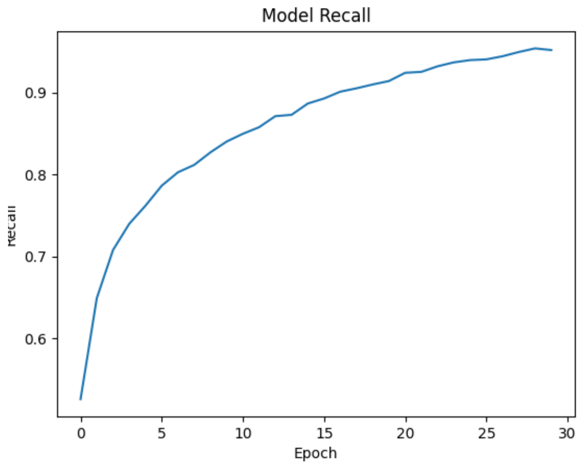

Figure 7 Recall Metrics

The ROC curve in this study, as presented in Figure 8 with AUC scores ranging from 0.88 (cardboard) to 0.98 (plastic), demonstrates intense class discrimination, particularly for glass and paper, which hug the top-left corner, indicating high accuracy with minimal false positives. Although most classes perform exceptionally well ($AUC>0.9$), the flatter curve for cardboard reveals classification challenges, most likely due to visual similarities with paper.

Furthermore, the AUC metrics in this study show that the model achieved strong AUC scores (0.88–0.98), with plastic (0.98), glass (0.97), and paper (0.94) showing near-flawless classification, while cardboard (0.88) had slightly lower performance due to visual similarities with paper, as demonstrated in Figure 9. These results confirm the model's high accuracy for waste sorting but highlight cardboard as needing refinement—likely through enhanced training data or adjusted features.

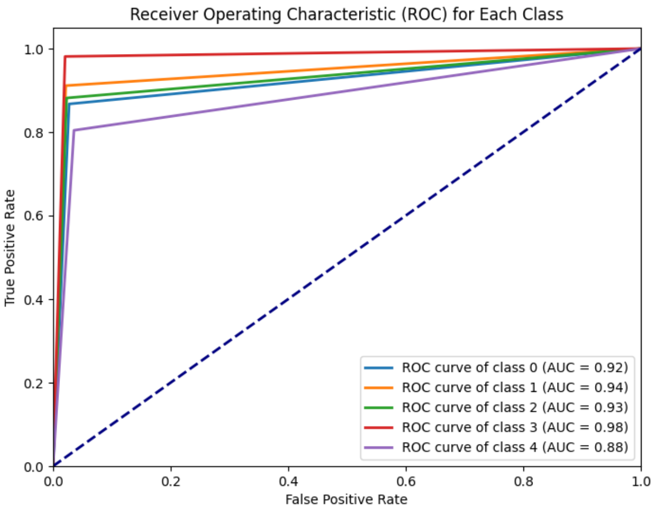

Figure 8 ROC Curve

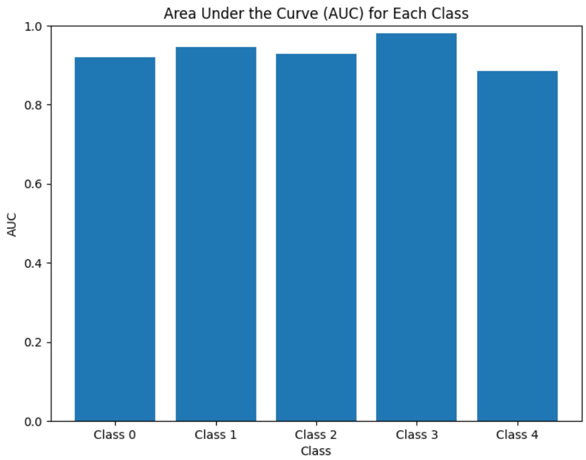

Figure 9 AUC Metrics

The model was evaluated using multiple metrics to ensure robust performance analysis. With an accuracy of 86.36%, precision of 87.44%, and recall of 85.87%, the MobileNetV2-based classifier demonstrates strong capability in handling the five waste categories. The confusion matrix (Figure 3) further confirms consistent performance, with most classes showing highly accurate positive rates, although some confusion exists between visually similar materials, such as plastic and glass. The F1-score, calculated at 86.65%, highlights the model's balanced trade-off between precision and recall. Additionally, the accuracy curve shows convergence and minimal overfitting, while the ROC curve and AUC metric (Figure 8 and Figure 9) validate classification effectiveness across all thresholds.

C. *Validation*

The model was validated on 1,224 unseen test images across five waste categories, achieving strong performance metrics: 86.36% overall accuracy demonstrates reliable classification capability, while the 0.4796 loss value indicates minimal prediction errors. With 87.44% precision and 85.87% recall, the model performs well in correctly identifying waste types (as precision) and capturing the most relevant cases (as the recall).

TABLE 3
VALIDATION TESTING

Total Data	*1224*
Accuracy	*0.8636*
Loss	*0.4796*
Precision	*0.8744*
Recall	*0.8587*
F-1 Score	*0.8685*

The proposed MobileNetV2 model was evaluated against two benchmarks: the VGG-16 architecture [28] and traditional SVM-based methods [29]. MobileNetV2 achieved an 86.36% accuracy, slightly surpassing VGG-16's 85.65% on the Real Waste dataset. While VGG-16 exhibited marginally higher precision (87.74% vs. MobileNetV2's 87.44%), MobileNetV2 demonstrated superior recall (85.87% vs. 84.82%) and a balanced F1-score (86.85% vs. 86.26%). These results highlight MobileNetV2's ability to deliver comparable performance to deeper networks like VGG-16 while maintaining computational efficiency, a critical advantage for real-world deployment.

To contextualize these findings, we compared MobileNetV2 with SVM-based approaches. [29] achieved a maximum accuracy of 76.49% using SVM combined with a pyramid scene parsing network (PSPNet) segmentation and handcrafted features (color, texture, shape). In contrast, MobileNetV2's 86.36% accuracy represents a 24.36% absolute improvement, underscoring the limitations of traditional methods reliant on manual feature engineering. SVM's performance depended heavily on labor-intensive preprocessing (e.g., PSPNet segmentation for object isolation and GLCM texture extraction), whereas MobileNetV2 autonomously learned discriminative features directly from raw, cluttered waste images.

This disparity emphasizes the advantages of modern deep learning frameworks in handling real-world waste classification. MobileNetV2's lightweight design efficiently captures complex patterns in degraded or overlapping materials, eliminating the need for error-prone segmentation steps. The results validate the shift toward end-to-end CNNs for scalable waste management systems, where accuracy, adaptability, and computational practicality are paramount.

D. *Discussion*

The Adam optimizer demonstrated superior performance over *RMSProp* and *AdaGrad* in optimizing the model, achieving higher overall accuracy (86%) and lower loss values, likely due to its adaptive learning rate mechanism, which efficiently navigates sparse gradients and noisy data common in waste classification tasks. Class-specific accuracies, however, revealed significant variability: glass

(97.92%) and paper (96.52%) achieved near-perfect classification, attributed to their distinct visual features (e.g., transparency, gloss, or uniform textures), while organic (87.45%) and plastic (81.92%) showed moderate performance, potentially due to for example deformed plastic bags vs rigid containers. Notably, cardboard (78.63%) lagged, a discrepancy likely rooted in class imbalance (fewer cardboard samples in the dataset) and visual ambiguities, for example, crumpled cardboard resembling wrinkled paper or degraded plastic.

TABLE 5
CLASS ACCURACY

Class Name	Accuracy
Class 0: Glass	97.92%
Class 1: Cardboard	78.63%
Class 2: Paper	96.52%
Class 3: Organic	87.45%
Class 4: Plastic	81.92%

The model's performance varied across classes, with cardboard achieving the lowest accuracy (78.63%), likely due to its underrepresentation in the dataset and visual similarities to paper. In contrast, classes with balanced representation and distinct features, such as glass, benefited from more apparent feature separation. These disparities highlight the impact of class imbalance and intra-class diversity on classification robustness, suggesting targeted augmentation or oversampling for minority classes in future work. Regardless of employing data augmentation to mitigate these issues, the model struggled to disentangle subtle feature overlaps, as evidenced by the ROC curve analysis. While ROC scores for most classes confirmed strong class separation, cardboard's lower performance suggests persistent challenges distinguishing it from visually similar categories.

IV. CONCLUSION

This study utilizes the Convolutional Neural Network (CNN) algorithm for efficient multi-class waste classification. Our TensorFlow-based model employs the MobileNetV2 architecture and Adam optimizer, achieving a training accuracy of 95.28% with a validation accuracy of 89.48%. The model demonstrates high precision (95.18%) and recall (94.00%), validating its ability to distinguish between different waste categories. ROC curve analysis indicates variable performance across waste types, with glass achieving the highest accuracy (97.92%) and cardboard the lowest (78.63%), suggesting areas for further refinement. On unseen data comprising 1,224 images, the model maintains strong generalization with an accuracy of 86.36%, a loss of 0.4796, a precision of 0.8744, and a recall of 0.8587.

Beyond performance metrics, this model is suitable for integration into real-world systems. For instance, it could be deployed on edge computing devices like Raspberry Pi or Jetson Nano for IoT-enabled smart bins, enabling real-time, automated waste sorting in households or public spaces. Alternatively, the model could be embedded within an Android application to assist users in categorizing waste via their smartphone camera. Furthermore, its lightweight architecture makes it compatible with conveyor belt-based automated sorting systems in industrial waste management facilities. Future work will explore object detection techniques for waste localization and evaluate alternative CNN architectures to enhance system robustness and deployment readiness.

REFERENCES

[1] "Sustainable cities & communities | SDG 11: Sustainable cities & communities." Accessed: Apr. 18, 2025. [Online]. Available: https://datatopics.worldbank.org/sdgatlas/goal-11-sustainable-cities-and-communities/?lang=en

[2] "SIPSN - Sistem Informasi Pengelolaan Sampah Nasional." Accessed: Apr. 20, 2025. [Online]. Available: https://sipsn.menlhk.go.id/sipsn/

[3] S. D. Meyrena and R. Amelia, "Analisis Pendayagunaan Limbah Plastik Menjadi Ecopaving Sebagai Upaya Pengurangan Sampah," *Indonesian Journal of Conservation*, vol. 9, no. 2, pp. 96–100, Dec. 2020, doi: 10.15294/ijc.v9i2.27549.

[4] K. K. A. Sholihah, "Kajian Tentang Pengelolaan Sampah di Indonesia," *Swara Bhumi*, vol. 03, no. 03, 2020.

[5] K. S. Nindya Ovitasari, D. Cantrika, Y. A. Murti, E. S. Widana, and I. G. A. Kurniawan, "Edukasi Pengolahan Sampah Organik dan Anorganik di Desa Rejasa Tabanan," *Bubungan Tinggi: Jurnal Pengabdian Masyarakat*, vol. 4, no. 2, 2022, doi: 10.20527/btjpm.v4i2.4986.

[6] A. Angdresey, R. Mandala, and L. Wikarsa, "Application of Kolintang Traditional Music Instrument Using Motion Sensor Detection: Webcam," in *2019 Fourth International Conference on Informatics and Computing (ICIC)*, 2019, pp. 1–5. doi: 10.1109/ICIC47613.2019.8985798.

[7] I. V. Masala and A. Angdresey, "The Real-Time Training System with Kinect: Trainer Approach," in *2017 International Conference on Soft Computing, Intelligent System and Information Technology (ICSIIT)*, 2017, pp. 233–237. doi: 10.1109/ICSIIT.2017.34.

[8] E. Alpaydin, *Introduction to machine learning*. MIT press, 2020.

[9] R. Ahn, S. Supakkul, L. Zhao, K. Kolluri, T. Hill, and L. Chung, "A Goal-Oriented Approach for Preparing a Machine-Learning Dataset to Support Business Problem Validation," *IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech), AB, Canada*, 2021.

[10] K. Hasan Mahmud and S. Al Faraby, "Klasifikasi Citra Multi-Kelas Menggunakan Convolutional Neural Network," Bandung, 2019.

[11] D. Bhatt *et al.*, "Cnn variants for computer vision: History, architecture, application, challenges and future scope," Oct. 01, 2021, *MDPI*. doi: 10.3390/electronics10202470.

[12] C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," *Electronic Markets*, vol. 31, no. 3, pp. 685–695, 2021, doi: 10.1007/s12525-021-00475-2.

[13] A. Angdresey, L. Sitanayah, and E. Pantas, "Comparison of the Convolutional Neural Network Architectures for Traffic Object Classification," in *2023 International Conference on Computer, Control, Informatics and its Applications (IC3INA)*, 2023, pp. 60–65. doi: 10.1109/IC3INA60834.2023.10285791.

[14] A. Pandey, A. Pandey, K. Maharjan, K. Shrestha, and J. Mansur, "Enhancing Waste Management: Automated Classification of Biodegradable and Non-biodegradable Waste using CNN," in *International Conference on Technologies for Computer, Electrical, Electronics & Communication (ICT-CEEL 2023)*, 2023, p. 1. [Online]. Available: https://www.researchgate.net/publication/376028733

[15] M. Grandini, E. Bagli, and G. Visani, "Metrics for Multi-Class Classification: An Overview," *arXiv:2008.05756*, 2020.
[16] B. Pang, E. Nijkamp, and Y. N. Wu, "Deep Learning with TensorFlow: A Review," 2020. doi: 10.3102/1076998619872761.
[17] K. Filus and J. Domańska, "Software vulnerabilities in TensorFlow-based deep learning applications," *Comput Secur*, vol. 124, 2023, doi: 10.1016/j.cose.2022.102948.
[18] L. Yusuf and T. Hidayatulloh, "Implementasi Algoritma Artificial Neural Network dengan Aktivasi ReLU: Klasifikasi Tiroid," *JURNAL SWABUMI*, vol. 12, no. 1, p. 2024, 2024.
[19] A. F. Agarap, "Deep Learning using Rectified Linear Units (ReLU)," Mar. 2018, [Online]. Available: http://arxiv.org/abs/1803.08375
[20] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation Functions in Deep Learning: A Comprehensive Survey and Benchmark," Sep. 2021, [Online]. Available: http://arxiv.org/abs/2109.14545
[21] J. Terven, D. M. Cordova-Esparza, A. Ramirez-Pedraza, and E. A. Chavez-Urbiola, "Loss Functions and Metrics in Deep Learning," Jul. 2023, [Online]. Available: http://arxiv.org/abs/2307.02694
[22] R. R. Akbar, B. Rahmat, and A. Junaidi, "Implementasi Algoritma Convolutional Neural Network Pada Transliterasi Aksara Jawa Ke Aksara Latin Dengan Penerapan Fungsi Hinge Loss," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3S1, Oct. 2024, doi: 10.23960/jitet.v12i3S1.5350.
[23] L. Zhao, "A Robust Loss Function for Multiclass Classification," *Int J Mach Learn Comput*, vol. 3, no. 6, pp. 462–467, Dec. 2013, doi: 10.7763/ijmlc.2013.v3.361.
[24] E. Setia Budi, A. Nofriyaldi Chan, P. Priscillia Alda, and M. Arif Fauzi Idris, "RESOLUSI: Rekayasa Teknik Informatika dan Informasi Optimasi Model Machine Learning untuk Klasifikasi dan Prediksi Citra Menggunakan Algoritma Convolutional Neural Network," *Media Online*, vol. 4, no. 5, pp.509, 2024, [Online]. Available: https://djournals.com/resolusi
[25] M. Irfan Yusuf, A. Wahyu Widodo, V. Wardhani, and B. University, "COVID-19 Detection on Chest X-Ray Images Using Modified VGG-19," 2023. [Online]. Available: www.jitecs.ub.ac.id
[26] A. M. Carrington *et al.*, "Deep ROC Analysis and AUC as Balanced Average Accuracy to Improve Model Selection, Understanding and Interpretation," Mar. 2021, doi: 10.1109/TPAMI.2022.3145392.
[27] B. G. Gaji´c, A. Amato, R. Baldrich, and C. Gatta, "Area Under the ROC Curve Maximization for Metric Learning."
[28] S. Single, S. Iranmanesh, and R. Raad, "RealWaste: A Novel Real-Life Data Set for Landfill Waste Classification Using Deep Learning," *Information (Switzerland)*, vol. 14, no. 12, Dec. 2023, doi: 10.3390/info14120633.
[29] Khadijah, S. N. Endah, R. Kusumaningrum, Rismiyati, P. S. Sasongko, and I. Z. Nisa, "Solid waste classification using pyramid scene parsing network segmentation and combined features," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 19, no. 6, pp. 1902–1912, 2021, doi: 10.12928/TELKOMNIKA.v19i6.18402.