International Journal on Robotics, Automation and Sciences

A Reproducible Benchmark of AdamW-Augmented Lightweight Models for Trash Classification

Kai Liang Lew, Xin Ming Chee, Chia Shyan Lee and Chean Khim Toa*

***Abstract* – Global waste generation is projected to reach 3.40 billion tonnes by 2050, creating urgent demands for automated waste classification systems that can overcome the limitations of manual sorting methods. Current deep-learning research on waste classification lacks standardised evaluation protocols, preventing meaningful architectural comparisons and hindering the progress of reproducible research. This paper establishes a reproducible benchmark framework for lightweight neural network models designed explicitly for trash classification research applications. Lightweight models are designed for optmised architecture and computation cost while maintain accuracy. Four representative lightweight models, including MobileNet V3 Large, Vision Transformer (ViT) Small, EfficientFormer, and ShuffleNet V2, were systematically evaluated on the TrashNet dataset using identical training protocols. All models employed AdamW optimisation with a learning rate of 1×10^{-4}, weight decay of 1×10^{-4}, and CosineAnnealingLR scheduling through 5-fold stratified cross-validation on RTX 2080 Ti hardware. Experimental results demonstrate that ViT Small achieved the highest classification accuracy at 0.815 but required 21.67M parameters, while MobileNet V3 Large delivered superior computational efficiency with 0.768 accuracy and 0.72ms inference time using only 4.21M parameters. Statistical analysis revealed significant performance differences across models ($p = 0.0002$), with hardware-aware architectural optimisations proving more critical than raw parameter reduction for computational performance on data centre GPU hardware. The standardised evaluation framework and open-source implementation provide rigorous baselines for advancing automated waste classification research.**

***Keywords*—Trash Classification, Lightweight Backbones, AdamW Optimisation, Data Augmentation, Benchmarking Trade-Offs, Deep Learning.**

I. Introduction

Global waste generation is projected to increase from 2.01 billion tonnes in 2016 to 3.40 billion tonnes by 2050 [1]. This increase stems from rapid population growth and changing consumption patterns, exacerbating the environmental waste crisis. In 2016, 1.6 billion tonnes of carbon dioxide, equivalent to a greenhouse gas, were produced during the process of treating and disposing of waste, which accounts for 5 per cent of global emissions [2]. An effective waste management system relies on accurate classification and efficient sorting processes. These two can help maximise recycling efficiency and avoid incorrect waste in the wrong category [3].

Manual sorting of waste remains the standard method in many facilities worldwide, creating challenges that directly impact environmental and economic outcomes. Workers face multiple hazards during sorting due to exposure to toxic materials and hazardous chemicals, which can result in

*Corresponding Author email: cheankhim.toa@xmu.edu.my, ORCID: 0000-0003-0879-4848
Kai Liang Lew is wit Faculty of Engineering and Technology, Multimedia University, Melaka, Malaysia (e-mail: 1132703002@student.mmu.edu.my).
Xin Ming Chee is with Infineon Technologies, Melaka, Malaysia (e-mail: xinming96@hotmail.com).
Chia Shyan Lee is with Curtin University, Perth, Australia (email: cat_lee97@hotmail.com)
Chean Khim Toa is with School of Computing and Data Science, Xiamen University Malaysia, Jalan Sunsuria, Bandar Sunsuria, 43900 Sepang, Selangor. (e-mail: cheankhim.toa@xmu.edu.my)

International Journal on Robotics, Automation and Sciences (2025) 7, 3:58-66
https://doi.org/10.33093/ijoras.2025.7.3.8
Manuscript received: 12 Jun 2025 | Revised: 7 Aug 2025 | Accepted: 11 Aug 2025 | Published: 30 Nov 2025

Published by MMU PRESS. URL: http://journals.mmupress.com/ijoras

musculoskeletal disorders and respiratory problems [4]. Moreover, material recovery facilities reject waste that exceeds the tolerable contamination threshold and must send it to landfills [5]. The processing can cost the material recovery facility an average of $82 per ton, while the value of the recovered materials is around $45 per ton. This shows the limitation of the manual sorting method [6].

With the introduction of deep learning techniques, computer vision tasks have been improved dramatically [7]. This improvement occurred because convolutional neural networks (CNNs) can learn data patterns independently based on the given dataset [8]. CNNs can achieve high accuracy in the classification of waste datasets [9], [10]. There are many approaches to model architecture for converting models into lightweight versions [11]. There is no existing systematic and reproducible evaluation of lightweight models specifically on the TrashNet dataset using standardised training procedures.

The first objective is to establish a standardised benchmarking framework for lightweight neural network models on waste classification tasks. This framework enables researchers to conduct fair comparisons between novel models and training methods and establish baselines using consistent evaluation protocols on TrashNet under controlled GPU computing conditions. RTX 2080 Ti (11GB) is selected as the benchmarking platform ecause it represents a widely-available mid-range data center GPU.

The second objective is to provide a comparative analysis of performance and efficiency trade-offs across different architectural paradigms. This analysis quantifies the relationships between classification accuracy, parameter count, and computational characteristics to inform architectural design decisions for waste classification research rather than practical deployment decisions for automated waste sorting systems.

The main research question can be stated as follows: How do different lightweight backbone models compare in terms of classification accuracy and computational efficiency when evaluated under standardised training conditions on the TrashNet dataset using RTX 2080 Ti hardware?

The first contribution is to provide a fully reproducible benchmark of MobileNet V3 Large, Vision Transformer (ViT) Small, EfficientFormer L1, and ShuffleNet V2 on TrashNet with MLflow logging. The code is available on GitHub.

The second contribution is to provide a systematic analysis of the relationships between performance and efficiency that challenge conventional assumptions about the parameter count versus computational performance. This analysis reveals that architectural optimisation and memory access patterns are more critical than raw parameter reduction for computational efficiency on data centre GPU hardware.

The remainder of this paper is organised as follows. The literature review examines related work on waste classification tasks and lightweight model architectures. The methodology describes the four backbone model architectures and training procedures. The experiment, results, and discussion cover the dataset description, model settings, and evaluation metrics, followed by performance comparisons and a comprehensive analysis. Finally, the conclusion summarises key findings, discusses the effectiveness of different approaches, and suggests future work.

II. Literature Review

A. Deep Learning Approaches for Waste Classification

As discussed earlier, manual sorting faces significant limitations that deep learning techniques can address. Ahmed et al. [11] conducted a comprehensive investigation of waste classification using state-of-the-art (SOTA) models, including CNNs, DenseNet, MobileNet, and Residual Networks (ResNet). The experiments showed that DenseNet169 achieved an accuracy of 94.40%, MobileNetV2 attained an accuracy of 97.60%, and ResNet50V2 achieved an accuracy of 98.95%. These results demonstrate that ResNet50V2 outperforms other SOTA models in waste classification.

Jin et al. [12] proposed an improved version of MobileNetV2, which increased classification accuracy by utilising transfer learning. Principal component analysis (PCA) is used to reduce the dimensionality of the last fully connected layer. This enables real-time operation of the model on an edge device. The experimental results show that their proposed model achieved an accuracy of 90.7% on "Huawei Cloud" datasets. It has an average inference time of 600 ms on the Raspberry Pi 4B microprocessor.

The dataset used to train a model has a significant impact on its performance in waste classification. A particular dataset has become a commonly used benchmark for this task [13]. However, many studies used different datasets to compare the performance of models. Kumsetty et al. [14] introduced the TrashBox dataset, which contains 17,785 images across seven different classes, including medical and e-waste categories. They trained multiple models on the new TrashBox dataset with transfer learning. They achieved an accuracy of 98.47% on ResNet-101.

Transfer learning has been widely used in waste classification tasks to overcome the gap in limited training data. Several studies have demonstrated the effectiveness of fine-tuning pre-trained models rather than training them from scratch [15], [16]. Risfendra et al. [17] trained an EfficientNet-B0 model with transfer learning. They train with a dataset that contains 7014 images with six different classes. The model achieved an accuracy of 91.94%, a precision of 92.10%, a recall of 91.94% and an F1-score of 91.96%.

There are several limitations in the current deep learning techniques for waste classification despite their good performance. The characteristics of the dataset can impact the model's performance in real-world scenarios [11]. Langley et al. [18] identify significant gaps in practical deployment because most existing deep-learning studies for waste sorting are

developed in controlled laboratory environments rather than those involving contaminated material streams.

B. Lightweight Neural Network Architectures

The use of lightweight models is due to the limited resources that can be utilised in an environment. Thus, the lightweight model has been the focus of research aimed at developing even lighter models. These models offer competitive performance while minimising computational resources.

MobileNet is one of the most widely used models in the lightweight family. The MobileNet introduced depthwise separable convolutions to reduce computation costs [19]. MobileNetV2 has further improved its efficiency by utilising inverted residual blocks and linear bottlenecks. It can achieve a competitive accuracy with a large model [20]. Wang et al. [21] proposed a Dense-MobileNet model by implementing DenseNet inside MobileNet. The experiment results showed that Dense-MobileNet can achieve higher accuracy than MobileNet while requiring fewer parameters and computational costs.

EfficientNet has gained attention due to its scaling methods, which optimise the network depth, width, and resolution [22]. This model achieved good accuracy with fewer parameters compared to traditional architecture. The ViT has emerged as an alternative to CNN, but it requires substantial computational resources and a large dataset for training. A lightweight ViT, ViT Small, is developed by reducing the number of attention heads and embedding dimensions [23]. However, the performance on the smaller dataset is poor. ShuffleNet used group convolutions and channel shuffle operations to achieve good efficiency [24].

The depth-wise separable convolutions reduce computational complexity by separating spatial and channel-wise operations [26]. Knowledge distillation allows smaller models to learn from larger teacher networks [27]. Quantisation and pruning techniques can further reduce model size and computational requirements, but they can also degrade model performance [28]. Lightweight neural network architectures become computationally efficient mainly by reducing the number of parameters. This allows them to be deployed on devices with limited resources while maintaining competitive performance [21], [29]. However, the efficiency gained by reducing parameters can worsen the model's performance [30]. The challenge is finding the right balance between computational efficiency and classification accuracy.

Several challenges remain despite the lightweight model demonstrating promising results. The lack of standardised benchmarks and protocols can make direct comparisons with different approaches misleading [31]. The field of lightweight models faces major inconsistencies in its methods. Many studies employ various training procedures, learning rates, and evaluation protocols. This makes results reproduce and lead to misleading comparisons, even with promising model performance [32], [30]. Table 1 shows the summary of waste classification and lightweight model studies from literature review.

TABLE 1. Summary of Waste Classification and Lightweight Model Studies from Literature Review

Study	Model	Dataset	Accuracy	Key Contribution
Ahmed et al. [11]	ResNet50V2	Kaggle "Garbage classification"	98.95 %	Best SOTA performance in comparison study
Jin et al. [12]	Improved MobileNetV2 with PCA	"Huawei Cloud"	90.7%	Real-time edge deployment on Raspberry Pi 4B
Kumsetty et al. [14]	ResNet-101	TrashBox (17,785 images, 7 classes)	98.47 %	Introduced new dataset with medical and e-waste
Risfendra et al. [17]	EfficientNet-B0	Custom (7,014 images, 6 classes)	91.94 %	Transfer learning approach
Wang et al. [21]	Dense2-MobileNet	Caltech datasets	~96%	~50% fewer parameters than MobileNet

III. METHODOLOGY

This section explains the lightweight backbone model and addresses the research question. It also introduces a standardised benchmarking framework with comprehensive reproducibility measures. Figure 1 shows flowchart of the experimental workflow.

A. Backbone Model Architectures

The experimental framework evaluates four different lightweight backbone architectures. Each model has a different architectural design. The selection encompasses a model with depthwise separable convolutions, MobileNet V3, a pure attention mechanisms model, ViT, a hybrid CNN-Transformer approaches model, EfficientFormer, and a channel shuffle operations model, ShuffleNet V2. All models were implemented using the timm library (version $\geqslant$ 0.9) and PyTorch (version $\geqslant$ 2.1), with the classifier head removed and replaced with a task-specific linear layer through a custom GenericClassifier wrapper architecture. This architecture instantiates backbone models via factory functions and adapts them for six-class waste classification.

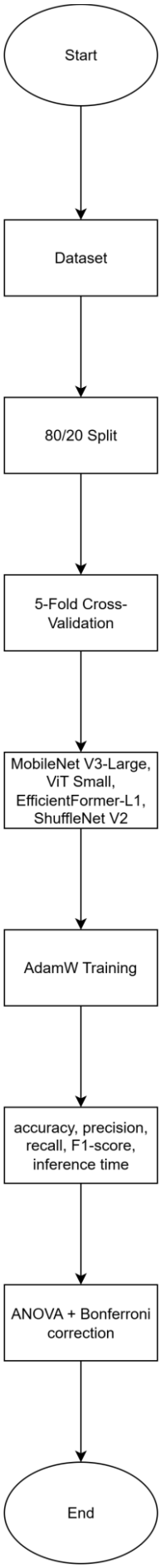

FIGURE 1. Flowchart of the experimental workflow

MobileNet V3-Large employs depthwise separable convolutions with squeeze-and-excitation blocks. The model utilises hard-swish activation functions and implements neural architecture search-derived building blocks. The model was instantiated using mobilenetv3_large_100 configuration with global average pooling, accepting variable input channels while maintaining computational efficiency through inverted residual structures.

ViT Small processes images as sequences of 16×16 pixel patches. This model abandons convolutional operations entirely, employing multi-head self-attention mechanisms across 12 transformer blocks with an embedding dimension of 384. The model uses ViT Small with a patch size of 16 and an input configuration of 224, utilising token-based global pooling, where the classification token aggregates spatial information across all patches.

EfficientFormer represents a hybrid CNN-Transformer model that combines the inductive biases of convolutional networks in early stages with the global modelling capacity of transformers in later stages. The model implements dimension-consistent design with 4D partition operations, enabling efficient mobile deployment while maintaining competitive performance.

SHuffleNetV2 is optimised in deployment due to memory access costs [25]. It utilises channel shuffle operations to enable information to flow across feature channels while maintaining high computational efficiency. For non-RGB inputs, the initial convolutional layer was modified to accept variable input channels while preserving the 24-channel output dimension.

All backbone models were validated for feature extraction by passing dummy inputs (1xCx224x224) through them. This process determined their output feature dimensions, which varied from 448 to 1280 depending on the architecture. The task-specific classifier head was built as a single linear layer that converts the backbone's feature output into the number of target classes.

The experimental framework maintained identical hyperparameters across all backbone models to ensure fair comparison. The selection of AdamW with a learning rate of 1×10^{-4} was based on its proven stability across diverse architectures. AdamW decoupled weight decay helps prevent overfitting on small datasets like TrashNet. The CosineAnnealingLR scheduler was chosen because it provides smooth learning rate decay without requiring architecture-specific tuning. This standardised approach isolates architectural differences as the main cause of performance variations. This directly addresses the research question regarding the optimal selection of a lightweight backbone.

B. Training Procedures

All models were trained using identical training protocols and no hyperparameter tuning. Each model used a batch size of 32, an input resolution of 224×224 pixels, and 100 epochs on an RTX 2080 Ti GPU. Adaptive Moment Estimation with Weight Decay

(AdamW) optimisation uses a learning rate of 1×10^{-4} and weight decay of 1×10^{-4}, with CosineAnnealingLR scheduling configured with a minimum learning rate value of 1×10^{-6}. The scheduler updates the learning rate per batch rather than per epoch, providing fine-grained learning rate decay throughout training. Standard cross-entropy loss was applied throughout all experiments. Model selection was performed based on validation accuracy, with the best-performing checkpoint saved for final evaluation.

Augmentation methods differed for training, validation, and testing to maintain consistent evaluations. Training augmentations consisted of resizing to 224×224 pixels, random horizontal flip with a probability of 0.5, random vertical flip with a probability of 0.5, and random rotation within ±10 degrees. All images underwent tensor conversion and normalisation using ImageNet statistics (mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225]). Validation and test sets received only resize and normalisation transformations to ensure consistent evaluation. Images were automatically converted to RGBA format before being converted to RGB to prevent preprocessing errors.

The paper used 5-fold stratified cross-validation for robust evaluation and to reduce bias. The dataset was initially split 80-20, with a fixed 20% reserved for testing in all final evaluations. Cross-validation was applied only to the remaining 80% of the dataset. This maintained consistent class distributions across all folds through stratified splitting. Performance metrics were then gathered from all five folds, and the mean and standard deviation for each metric were reported.

A random seed was fixed at 42 across all model evaluations using NumPy's random number generator for dataset-splitting operations. The MLflow tracking infrastructure captured comprehensive experimental metadata. Each experiment run was assigned a unique identifier, enabling precise reproduction and comparison. The logging system monitored training dynamics through per-batch gradient norms, learning rate schedules, and computational resource utilisation at each epoch. Performance artefacts were systematically preserved in multiple formats. The confusion matrices are saved in JSON, pickle and PNG format. Inference timing measurements were conducted on the test dataset with CUDA synchronisation to ensure accurate latency assessment.

Inference timing measurements followed a strict protocol, excluding data loading time to focus solely on the model's forward pass execution. Measurements were then aggregated across the entire test dataset to calculate the average inference time per sample. Systematic memory management protocols were implemented after each experimental run to prevent memory overflow during consecutive experiments. This was followed by garbage collection and CUDA cache clearing to ensure clean memory states between all 20 experimental runs.

IV. Experiment, Results and Discussion

A. Dataset

Experiments utilised the TrashNet dataset [33], which contains 2,527 images across six waste categories with significant class imbalance. Class distribution includes 594 papers, 501 glass, 482 plastic, 410 metal, 403 cardboard, and 137 trash images, with the trash class notably underrepresented.

The dataset was partitioned using an 80-20 split for cross-validation and test sets, respectively. To ensure reproducibility, the random seed was fixed at 42 using NumPy's random number generator. The test set contained 505 images, while the remaining 2,022 images were used for 5-fold stratified cross-validation. This process yielded approximately 1,621 training images and 405 validation images per fold, with exact numbers varying slightly due to stratification requirements.

B. Model Setting

The four evaluated backbone models have smaller parameters that directly impact computational efficiency. ShuffleNet V2 has the least parameter models with 1.26M parameters. MobileNet V3-Large contains 4.21M parameters, while EfficientFormer utilises 11.39M parameters. The ViT Small requires 21.67 million parameters, making it the largest model in comparison. All models were adapted for six-class waste classification by replacing the original classifier head with a linear layer mapping from each backbone's feature dimension to the target number of classes.

C. Evaluation Metrics

Model performance was assessed using classification accuracy as the primary metric, supplemented by accuracy, precision, recall, and F1-score to provide a comprehensive evaluation of classification performance. Computational efficiency was measured using the total parameter count and the average inference time per sample. This inference time was calculated on the test dataset with CUDA synchronisation to ensure timing accuracy. All evaluation procedures were logged using MLflow to ensure comprehensive tracking of experimental conditions and results.

D. Results and Discussion

Repeated-measures ANOVA on 5-fold cross-validation results revealed statistically significant differences in mean accuracy scores across the four lightweight backbone models. Table 2 presents the ANOVA results.

TABLE 2. Repeated-measures ANOVA Results

Source	F-Value	Num DF	Den DF	p-value
Model	15.4654	3	12	0.0002

The significant ANOVA result with $p = 0.0002$ (< 0.05) warranted post-hoc pairwise comparisons using the Bonferroni correction to control for multiple testing. Table 3 presents the complete pairwise comparison results.

TABLE 3. Post-Hoc Pairwise T-Test Results with Bonferroni Correction

Model 1	Model 2	T-statistic	p (uncorrected)	p (Bonferroni)
EfficientFormer	MobileNet V3 Large	2.868	0.046	0.273
EfficientFormer	ShuffleNet V2	2.813	0.048	0.289
EfficientFormer	ViT Small	-2.929	0.043	0.257
MobileNet V3 Large	ShuffleNet V2	0	1	1
MobileNet V3 Large	ViT Small	-6.096	0.004	0.022
ShuffleNet V2	ViT Small	-6.437	0.003	0.018

Post-hoc analysis with Bonferroni correction revealed only two statistically significant pairwise differences between MobileNet V3-Large vs ViT-Small ($p = 0.022$) and ShuffleNet V2 vs ViT-Small ($p = 0.018$). While other comparisons showed trends before correction (e.g., EfficientFormer vs ShuffleNet V2, $p = 0.048$), these did not survive multiple testing adjustment.

The experimental evaluation reveals distinct performance characteristics across the four lightweight backbone models. Table 4 presents the comprehensive performance metrics for all models.

TABLE 4. Model Performance Summary

Model	Accuracy	Precision	Recall	F1-Score	Parameters (M)	Inference Time (ms)
ViT Small	0.815 ± 0.0074	0.818 ± 0.007	0.816 ± 0.007	0.816 ± 0.007	21.67	1.15 ± 0.12
EfficientFormer L1	0.799 ± 0.0099	0.804 ± 0.009	0.800 ± 0.010	0.800 ± 0.010	11.39	1.14 ± 0.10
ShuffleNet V2 X1	0.768 ± 0.0161	0.773 ± 0.016	0.768 ± 0.016	0.769 ± 0.016	1.26	1.09 ± 0.03
MobileNet V3 Large	0.768 ± 0.0202	0.774 ± 0.018	0.768 ± 0.020	0.769 ± 0.019	4.21	0.72 ± 0.06

EfficientFormer has the second-highest accuracy at 0.799. The model performed competitively with 11.39M parameters, which is half the number of ViT Small. However, its inference times remained comparable at 1.14 ms per sample on RTX 2080 Ti hardware.

The evaluation shows that ViT Small achieved the highest accuracy at 0.815% but required 21.67 million parameters. In contrast, efficiency-focused models, such as MobileNet V3 Large and ShuffleNet V2, achieved comparable accuracy levels of 0.768 and 0.768, respectively, with significantly fewer parameters.

Computational efficiency analysis reveals that MobileNet V3 Large achieved superior computational efficiency on RTX 2080 Ti hardware, with an inference time of 0.72 ms, while having 4.21 million parameters. It outperforms ShuffleNet V2, which achieves 1.09ms with 1.26M parameters. This 34% speed improvement challenges conventional assumptions about the efficiency of parameter count.

Cross-validation results revealed varying training stability among models. MobileNet V3 Large has the highest variance with $\sigma = 2.02\%$, indicating less robust training dynamics. EfficientFormer had the second highest variance with $\sigma = 0.99\%$. This suggests that EfficientFormer has lower sensitivity to data distribution, making it a good choice for comparative study. Table 5 shows the model performance rankings on each category.

TABLE 5. Model Performance Rankings

Metric	Best Model	Second	Third	Fourth
Accuracy	ViT-Small (0.815)	EfficientFormer-L1 (0.799)	MobileNet V3-Large (0.768)	Shuffle Net V2 (0.768)
Inference Speed	MobileNet V3-Large (0.72ms)	ShuffleNet V2 (1.09ms)	EfficientFormer-L1 (1.14ms)	ViT-Small (1.15ms)
Parameters	Shuffle Net V2 (1.26M)	MobileNet V3-Large (4.21M)	EfficientFormer-L1 (11.39M)	ViT-Small (21.67M)
Overall Trade-off	MobileNet V3-Large	EfficientFormer-L1	ViT-Small	Shuffle Net V2

Confusion Matrix

The confusion matrices show distinct classification patterns among the lightweight models evaluated during 5-fold cross-validation. Figure 2 shows the ViT Small Confusion Matrix for each fold.

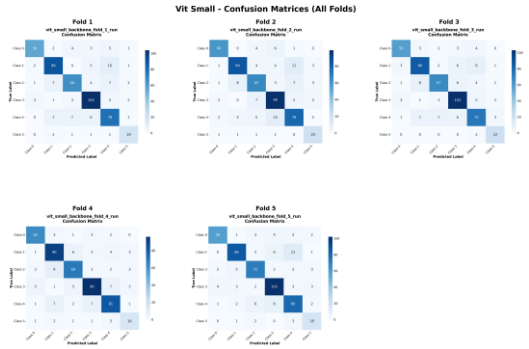

FIGURE 2. ViT Small Confusion Matrix for each fold

The ViT Small confusion matrices show the strongest classification performance across all five folds. The intense and consistent diagonal patterns in these matrices visually confirm ViT Small's reported highest accuracy of 0.815. Figure 3 shows the Efficientformer Confusion Matrix for each fold.

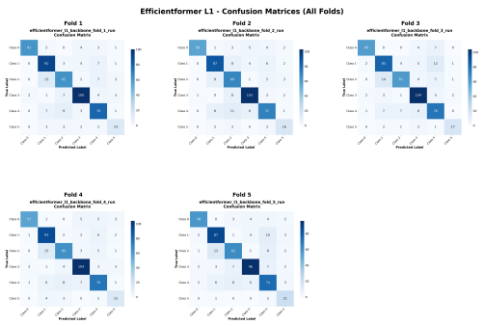

FIGURE 3. Efficientformer Confusion Matrix for each fold.

EfficientFormer exhibits comparable but slightly reduced diagonal intensity compared to ViT Small across all five folds. While the diagonal elements are strong, there is marginally more off-diagonal confusion visible compared to ViT Small, which corresponds to its slightly lower accuracy of 0.799. The matrices demonstrate that EfficientFormer maintains reliable classification boundaries between waste categories. Figure 4 shows the ShufflenetV2 Confusion Matrix for each fold.

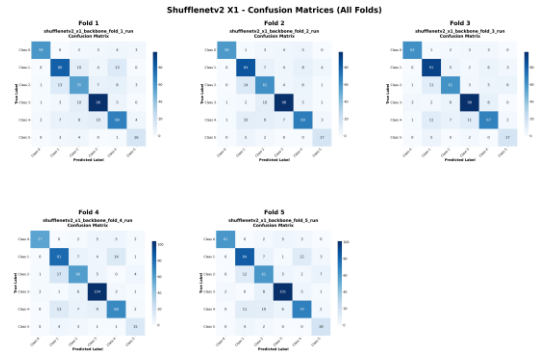

FIGURE 4. ShufflenetV2 Confusion Matrix for each fold.

ShuffleNet V2 displays more variation in classification patterns across folds compared to the transformer-based models. Figure 5 shows the MobileNetV3 Large Confusion Matrix for each fold.

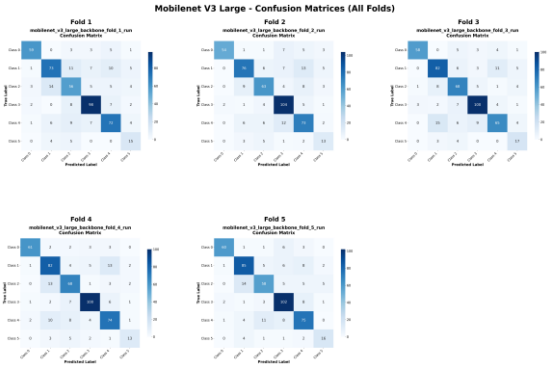

FIGURE 5. MobileNetV3 Large Confusion Matrix for each fold

MobileNet V3 Large demonstrates consistent confusion matrix patterns across all five folds, with a stable diagonal intensity that reflects its reported accuracy of 0.768. The consistency between folds visually confirms that MobileNet V3 Large has a standard deviation of 2.02%.

Discussion

Comparing across all four models, ViT Small demonstrated the strongest diagonal patterns, with the highest classification confidence, corresponding to its superior accuracy of 0.815. EfficientFormer has its competitive edge with an accuracy of 0.799. ShuffleNet V2 exhibits the most scattered confusion patterns, characterised by increased off-diagonal elements, which aligns with its 0.768 accuracy. MobileNet V3 Large maintained in between the model with moderate diagonal strength but variable consistency across folds.

Visual analysis shows that while MobileNet V3 Large and ShuffleNet V2 achieve similar accuracy, their confusion patterns are very different. MobileNet V3 Large maintains strong classification boundaries, but its performance varies. ShuffleNet V2 has more consistent patterns with less scatter in misclassification. This evidence supports the finding that ShuffleNet V2 offers superior training stability with lower computational efficiency compared to MobileNet V3 Large while maintaining comparable overall accuracy.

The precision-recall balance across all models, with precision ranging from 0.773 to 0.818 and recall ranging from 0.768 to 0.816. These values indicate that no model exhibits a bias toward any specific class despite the dataset's class imbalance.

ViT Small provides the highest classification performance for applications where computational resources are less constrained. However, the increased parameter count and moderate training variance require careful consideration for research applications that require consistency.

EfficientFormer provides an effective trade-off between accuracy and computational efficiency. The hybrid model has competitive performance with moderate resource requirements. This performance makes it suitable for controlled experimental scenarios with reasonable computational budgets.

MobileNet V3 Large can be the optimal choice for applications prioritising inference speed and computational efficiency on RTX 2080 Ti hardware. It has superior inference performance, but the training is inconsistent, making it suitable for experimental settings where speed is critical. The combination of competitive accuracy at 0.768 with superior inference speed at 0.72 ms positions this model as the optimal balance point for controlled comparative analysis, where speed is prioritised over training consistency.

The results show that ShuffleNet V2 achieves the lowest parameter count while maintaining moderate efficiency on RTX 2080 Ti hardware. This finding shows the importance of benchmarking over theoretical parameter counting when selecting models for architectural comparison.

A critical insight from computational efficiency results is that MobileNet V3 Large has superior inference performance, even with a moderate number of parameters. This demonstrates that the model is taking full advantage of the RTX 2080 Ti hardware with faster inference performance. This finding suggests that, rather than focusing on the parameter, it is more effective to design the model based on the specific hardware used.

V. CONCLUSION

This paper establishes a comprehensive and reproducible benchmark for lightweight neural network models in waste classification tasks, providing comparative guidance for researchers developing automated waste classification systems [34].

This benchmarking study has limitations that contextualize the findings. TrashNet's limited size and controlled conditions may not represent real-world waste streams with contaminated. The evaluation on

RTX 2080 Ti hardware may not generalize to edge devices or newer architectures with different memory hierarchies. Additionally, the standardized training protocol, while ensuring fair comparison, may not reveal each architecture's optimal performance under architecture-specific tuning. The six-class also simplifies modern recycling requirements. These constraints position the benchmarks as research baselines rather than deployment-ready solutions

Future work should extend this comparative benchmarking framework to evaluate additional architectural and training methodologies under the established standardised protocols. Investigation of knowledge distillation techniques and ensemble methods could enhance accuracy while maintaining computational efficiency. Additionally, extending the evaluation to diverse hardware platforms, including edge computing devices, would provide broader validation of the comparative relationships identified through this controlled experimental analysis on the RTX 2080 TI 11GB rather than the challenging visual conditions encountered in practical waste sorting environments.

VI. Data and Code Availability

The TrashNet dataset used in this paper is publicly available at https://github.com/garythung/trashnet. The complete source code, experimental configurations, and trained model checkpoints are available at https://github.com/lewbei/A-Reproducible-Benchmark-of-AdamW-Augmented-Lightweight-Models-for-Trash-Classification. All experimental results can be reproduced using the provided MLflow configurations and random seed settings.

Acknowledgment

We want to thank peer reviewers for helping to review the paper.

Funding Statement

There is no funding agencies supporting the research work.

Author contributions

Kai Liang Lew: Conceptualization, Data Curation, Methodology, Validation, Writing – Original Draft Preparation;

Xin Ming Chee: Writing – Review & Editing;

Chia Shyan Lee: Writing – Review & Editing;

Chean Khim Toa: Writing – Review & Editing.

Conflict of interests

No conflict of interests were disclosed.

Ethics statements

Ethical approval was not applicable to this research since it did not involve human participants, animals, or sensitive data.

References

[1] S. Kaza, L. Yao, P. Bhada-Tata and F. Van Woerden, “What a waste 2.0: a global snapshot of solid waste management to 2050,” *World Bank Publications*, 2018. URL: https://hdl.handle.net/10986/30317

[2] UNEP ISWA, "Beyond an age of waste: turning rubbish into a resource," *Global waste management outlook* 2024, 2024. DOI: https://doi.org/10.59117/20.500.11822/44939

[3] S.L. Bradshaw, H.A. Aguirre-Villegas, S.E. Boxman and C.H. Benson, "Material Recovery Facilities (MRFs) in the United States: Operations, revenue, and the impact of scale," *Waste Management*, vol. 193, pp. 317-327, 2025. DOI: https://doi.org/10.1016/j.wasman.2024.12.008

[4] S.E. Tshivhase, N.S. Mashau, T. Ngobeni and D.U. Ramathuba, "Occupational health and safety hazards among solid waste handlers at a selected municipality South Africa," *Health SA Gesondheid*, vol. 27, 2022. DOI: https://doi.org/10.4102/hsag.v27i0.1978

[5] K.K. Hansen, P. Rasmussen, V. Schlünssen, K. Broberg, K. Østergaard, E.E. Tranchant, T. Sigsgaard, H.A. Kolstad and A.M. Madsen, "Microbial exposure during recycling of domestic waste: a cross-sectional study of composition and associations with inflammatory markers," *Occupational and Environmental Medicine*, vol. 81, no. 11, pp. 580-587, 2024. DOI: https://doi.org/10.1136/oemed-2024-109628

[6] O. Olafasakin, J. Ma, S.L. Bradshaw, H.A. Aguirre-Villegas, C. Benson, G.W. Huber, V.M. Zavala and M. Mba-Wright, "Techno-Economic and life cycle assessment of standalone Single-Stream material recovery facilities in the United states," *Waste Management*, vol. 166, pp. 368-376, 2023. DOI: https://doi.org/10.1016/j.wasman.2023.05.011

[7] F.R. Sayem, M.S.B. Islam, M. Naznine, M. Nashbat, M. Hasan-Zia, A.K.A. Kunju, A. Khandakar, A. Ashraf, M.E. Majid, S.B.A. Kashem and M.E.H. Chowdhury, "Enhancing waste sorting and recycling efficiency: robust deep learning-based approach for classification and detection," *Neural Computing and Applications*, vol. 37, no. 6, pp. 4567-4583, 2025. DOI: https://doi.org/10.1007/s00521-024-10855-2

[8] M. Chhabra, B. Sharan, M. Elbarachi and M. Kumar, "Intelligent waste classification approach based on improved multi-layered convolutional neural network," *Multimedia Tools and Applications*, vol. 83, no. 36, pp. 84095-84120, 2024. DOI: https://doi.org/10.1007/s11042-024-18939-w

[9] A. Arishi, "Real-Time Household Waste Detection and Classification for Sustainable Recycling: A Deep Learning Approach," *Sustainability*, vol. 17, no. 5, p. 1902, 2025. DOI: https://doi.org/10.3390/su17051902

[10] H.A. Khan, S.S. Naqvi, A.A.K. Alharbi, S. Alotaibi and M. Alkhathami, "Enhancing trash classification in smart cities using federated deep learning," *Scientific Reports*, vol. 14, no. 1, 2024. DOI: https://doi.org/10.1038/s41598-024-62003-4

[11] M. I. B. Ahmed et al., "Deep Learning Approach to Recyclable Products Classification: Towards Sustainable Waste Management," *Sustainability*, vol. 15, p. 11138, 2023. DOI: https://doi.org/10.3390/su151411138

[12] S. Jin, Z. Yang, G. Królczykg, X. Liu, P. Gardoni and Z. Li, "Garbage detection and classification using a new deep learning-based machine vision system as a tool for sustainable waste recycling," *Waste Management*, vol. 162, pp. 123-130, 2023. DOI: https://doi.org/10.1016/j.wasman.2023.02.014

[13] V.V. Santhanalakshmi and H. Nguyen, "TrashNet: An object detection model that classifies images of trash in real-time," *Journal of Student Research*, vol. 13, no. 2, 2024. DOI: https://doi.org/10.47611/jsrhs.v13i2.6533

[14] N. V. Kumsetty, A. Bhat Nekkare, S. K. S., and A. Kumar M., "TrashBox: Trash Detection and Classification using Quantum Transfer Learning," *2022 31st Conference of Open Innovations Association (FRUCT)*, pp. 125–130, 2022. DOI: https://doi.org/10.23919/FRUCT54823.2022.9770922

[15] S. Poudel and P. Poudyal, "Classification of Waste Materials using CNN Based on Transfer Learning," *Proceedings of the 14th Annual Meeting of the Forum for Information Retrieval Evaluation*, pp. 29–33, 2022. DOI: https://doi.org/10.1145/3574318.3574345

[16] Q. Zhang, Q. Yang, X. Zhang, Q. Bao, J. Su and X. Liu, "Waste image classification based on transfer learning and

convolutional neural network," *Waste Management*, vol. 135, pp. 150-157, 2021.
DOI: https://doi.org/10.1016/j.wasman.2021.08.038

[17] R. Risfendra, G.F. Ananda and H. Setyawan, "Deep Learning-Based Waste Classification with Transfer Learning Using EfficientNet-B0 Model," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 8, no. 4, pp. 535-541, 2024. DOI: https://doi.org/10.29207/resti.v8i4.5875

[18] A. Langley, M. Lonergan, T. Huang and M.R. Azghadi, "Analysing mixed construction and demolition waste in material recovery facilities: Evolution, challenges, and applications of computer vision and deep learning," *Resources, Conservation and Recycling*, vol. 217, p. 108218, 2025.
DOI: https://doi.org/10.1016/j.resconrec.2025.108218

[19] A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto and H. Adam, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *arXiv*, 2017.
DOI: https://doi.org/10.48550/arXiv.1704.04861

[20] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. -C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4510–4520, 2018.
DOI: https://doi.org/10.1109/CVPR.2018.00474

[21] W. Wang, Y. Li, T. Zou, X. Wang, J. You and Y. Luo, "A Novel Image Classification Approach via Dense-MobileNet Models," *Mobile Information Systems*, vol. 2020, pp. 1-8, 2020.
DOI: https://doi.org/10.1155/2020/7602384

[22] M. Tan and Q.V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," *arXiv*, 2019.
DOI: https://doi.org/10.48550/arXiv.1905.11946

[23] H. Wu et al., "CvT: Introducing Convolutions to Vision Transformers," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22–31, 2021.
DOI: https://doi.org/10.1109/ICCV48922.2021.00009

[24] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6848–6856, 2018.
DOI: https://doi.org/10.1109/CVPR.2018.00716

[25] N. Ma, X. Zhang, H. -T. Zheng, and J. Sun, "ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design," 2018.
DOI: https://doi.org/10.48550/arXiv.1807.11164

[26] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1800–1807, 2017.
DOI: https://doi.org/10.1109/CVPR.2017.195

[27] G. Hinton, O. Vinyals and J. Dean, "Distilling the Knowledge in a Neural Network," *arXiv*, 2015.
DOI: https://doi.org/10.48550/arXiv.1503.02531

[28] S. Han, H. Mao and W.J. Dally, "Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding," *arXiv*, 2015.
DOI: https://doi.org/10.48550/arXiv.1510.00149

[29] H. Gang, S. Guanglei, W. Xiaofeng and J. Jinlin, "CCNNet: a novel lightweight convolutional neural network and its application in traditional Chinese medicine recognition," *Journal of Big Data*, vol. 10, no. 1, 2023.
DOI: https://doi.org/10.1186/s40537-023-00795-4

[30] Ş.G. Kıvanç, B. Şen, F. Nar and A.Ö. Ok, "Reducing Model Complexity in Neural Networks by Using Pyramid Training Approaches," *Applied Sciences*, vol. 14, no. 13, p. 5898, 2024.
DOI: https://doi.org/10.3390/app14135898

[31] A.C. S., J. Mammoo, A.P. S. and A.S.P. A., "Deep Learning Approaches for Waste Classification," *2024 International Conference on Advancements in Power, Communication and Intelligent Systems (APCI)*, pp. 1-7, 2024.
DOI: https://doi.org/10.1109/APCI61480.2024.10617120

[32] H. Semmelrock, T. Ross-Hellauer, S. Kopeinik, D. Theiler, A. Haberl, S. Thalmann and D. Kowald, "Reproducibility in Machine Learning-based Research: Overview, Barriers and Drivers," *arXiv*, 2024.
DOI: https://doi.org/10.48550/arXiv.2406.14325

[33] M. Yang and G. Thung, "Trashnet", *GitHub repository*, 2016.
URL: https://github.com/garythung/trashnet?tab=readme-ov-file (accessed 10 June 2025)

[34] K.L. Lew, K.S. Sim and Z. Ting, "Deep Learning Approach EEG Signal Classification," *International Journal on Informatics Visualization*, vol. 8, no. 3–2, pp. 1693–1702, 2024.
DOI: https://doi.org/10.62527/joiv.8.3-2.2959