SMART GARBAGE CLASSIFICATION SYSTEM USING ML ALGORITHM

BY

YOUSUF TALUKDER PRANTO
ID: 193-15-13443

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Md Aynul Hasan Nahid
Lecturer
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

14 MAY 2025

APPROVAL

This Project titled **"Smart Garbage Classification System Using Ml Algorithm"**, submitted by **Yousuf Talukder Pranto**, ID No: **193-15-13443** to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **14 May, 2025**.

BOARD OF EXAMINERS

Ms. Nazmun Nessa Moon — **Chairman**
Associate Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Mr. Shah Md Tanvir Siddiquee — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Mr. Md Umaid Hasan — **Internal Examiner**
Sr. Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

for --

Dr. Ahmed Wasif Reza — **External Examiner**
Professor
Department of Computer Science and Engineering
East West University

DECLARATION

I hereby declare that, this project has been done by me under the supervision of **Md Aynul Hasan Nahid, Lecturer, Department of CSE, Daffodil International University**. I also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

14/05/2025

____

Md Aynul Hasan Nahid
Lecturer
Department of CSE
Daffodil International University

Submitted by:

Pranto Yousuf
14-05-25

____

Yousuf Talukder Pranto
ID: 193-15-13443
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First, I express my heartiest thanks and gratefulness to Almighty God for His divine blessing, which made it possible for me to complete the final year project/internship successfully.

I am truly grateful and wish to express my profound indebtedness to my **Supervisor, Md Aynul Hasan Nahid, Lecturer**, Department of CSE at Daffodil International University, Dhaka. My supervisor's knowledge and expertise in the discipline of "Machine Learning" gave me the confidence I needed to complete the task. His patience, intellectual direction, encouragement, frequent and active supervision, helpful advice, informative counsel, and reading many substandard drafts and editing them made this effort possible.

I would also like to express my heartiest gratitude to **Dr. Sheak Rashed Haider Noori, Professor and Head of the Department of CSE**, for his kind assistance in finishing my project, as well as to the other faculty members and staff of the CSE department at Daffodil International University.

My thanks also go to all my course mates at Daffodil International University, who participated in discussions and contributed to the coursework.

Finally, I must acknowledge with due respect the constant support and patience of my parents

ABSTRACT

Growing landfill volumes and slow, labour-intensive sorting have made urban waste one of today's least glamorous bottlenecks. To tackle that gap, this study builds an image-based classification pipeline that recognises eleven common rubbish types ranging from PET bottles to greasy cardboard in a single camera shot. Four convolutional backbones were examined: ResNet-50, MobileNet V2, EfficientNet-B0 and DenseNet 121. Each network was fine-tuned with transfer learning on a purpose-built collection of roughly 30 560 images per class (≈ 336 k photographs overall). Prior to training, every picture was resized to 224 × 224 px and augmented through random flips, rotations, colour-jitter and brightness shifts to mimic curb-side variability. Model performance was judged with accuracy, F1 score and class-level confusion matrices. EfficientNet-B0 delivered the best balance of speed and precision, attaining 94 % validation accuracy and a macro-F1 of 0.93, while MobileNet V2 finished close behind but trained 40 % faster an advantage for edge deployments or rapid re-training cycles. All weights were exported as compact *.h5* files and tested on unseen street images, confirming real-time inference on desktop GPUs and Raspberry Pi boards alike. By removing much of the manual effort from waste sorting, the proposed system cuts worker exposure to hazardous material and improves the purity of downstream recycling streams. The results demonstrate that modern, lightweight vision models can be integrated into smart-bin infrastructure or material-recovery facilities with only modest computational budgets, offering a practical route toward cleaner, more sustainable cities.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 Overview

Smart waste management is increasingly becoming an urgent concern in the context of rapid urbanization and population expansion. The conventional garbage collection and sorting mechanism is neither effective nor raises grave health, environmental, and operational issues. With the progress in Artificial Intelligence (AI) and deep learning technologies, it is now possible to automate garbage sorting into different groups significantly enhancing recycling and minimizing environmental degradation. This report describes the implementation of an intelligent garbage management system using deep learning-based image classification approaches. The system is developed to classify garbage images into 11 groups including plastic, metal, glass, organic, cardboard, paper, and so on automatically. For achieving the same, I trained and tested four high-performance deep learning architectures including ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121 and fine-tuned each one of them using transfer learning on a massive garbage image data set. I organized the data set into individual class and applied data pre-processing steps including resizing into 224x224 pixel images, normalization, and real-time image augmentation. With these steps, I made sure all the models could generalize well to test garbage images. For this purpose, I trained each model for 5 epochs with a batch size of 32 and saved them in .h5 file format for subsequent inference and prediction. The aim of this system is not just to classify garbage into high accuracy groups but also to show a scalable and effective solution for modern garbage segregation in smart cities. Successful implementation of this garbage classifier using automation proves the power of deep learning in solving one of the most critical environmental challenges of the era.

1.2 Background and Present State

Proper classification of waste is an initial step in recycling and disposal. Historically, sorting of waste was carried out through manual segregation by people, city workers, or facility workers. This is time-consuming and error-intensive as well as exposes humans to health-risk concerns, particularly when handling dangerous or rotting material. In spite of concerted efforts by city governments and environmental institutes, efficiency during classification and recycling is still restricted due to insufficient automation, public ignorance, and poor infrastructure. Today, incorporation of technology into waste management is making waves. Smart trash cans, sensor-based sorting machines, and IoT-based logistics have been proposed to streamline collection and processing. But classification based on visual appearance especially when you have mixed and contaminated waste still is an obstacle that current rule-based systems find difficult to overcome. Deep learning and Convolutional Neural Networks (CNNs), in particular, have been proven to be excellent image classification tools with applications in various fields including healthcare, security, and transportation. Having seen all this potential, I designed a Smart Garbage Management System based on deep learning to overcome the bottleneck in classifying garbage during processing. Unlike conventional systems based on rule-based mechanisms or manual inspection, my system processes garbage images into their respective categories through trained network models. The state of technology is now good enough so that it is possible to train a model with high accuracy using transfer learning even with insufficient hardware capability. I used a dataset with more than 30,000 images per class and processed and fed those into four complex pre-trained models - ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121. Every model was validated and tested using validation sets to guarantee reliability in real-world applications. The project fits into the general vision for smart cities in which intelligent systems drive mechanisms for collecting, sorting, and recycling garbage. Municipalities can avoid manual labor and minimize contamination in recycling flows and reach long-term sustainability targets by incorporating such classifying capabilities.

1.3 Problem Statement

One of the greatest hurdles in today's urban settings is classifying household and industrial garbage in an efficient and correct manner. Garbage sorting manually requires too much labor and is extremely unreliable due to human inaccuracies, exhaustion, and ignorance regarding correct separation of garbage. Consequently, recoverable material is disposed of in landfills and hazardous material is mishandled, causing further environmental degradation and health hazards to people. While conventional garbage management technologies have been upgraded through enhanced collection and transportation processes, the primary concern of source-based auto-classification is still largely unresolved. Current smart bins and mechanical sorting technologies depend largely on sensors, physical attributes, or manually designed rules, which fall short in sorting mixed, broken, or visually indistinguishable garbage material. The challenge I set out to tackle in this project was creating a deep learning-based image classification system that could classify garbage into pre-specified categories based on visual inputs accurately. This system required to work irrespective of lighting conditions, orientation of objects, and real-world image alterations. Moreover, it must be computationally efficient enough to be hosted on local PCs with modest hardware combinations like GTX 1050 Ti GPU and Ryzen 5600X CPU. To solve this problem, I trained four pretrained CNN models ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121 and tested their capacity to classify garbage images into 11 different categories. The objective was to ascertain which model offered the optimum tradeoff among accuracy, F1 score, training time, and inference performance and could therefore be used as the foundation for a scalable garbage classification system or visually indistinguishable garbage material.

1.4 Objectives

The main aim of this project is to design a sound and smart garbage sorting system based on deep learning models that is able to classify different types of garbage through image input. The system is meant to support more intelligent and more effective urban waste management through automation of sorting decreasing labor, maximizing recycling efficiency, and encouraging environmental conservation.

To accomplish this goal, I designed the following specific objectives:

Develop a Multi-Class Garbage Classification Pipeline: I aimed to classify garbage images into 11 distinct categories such as plastic, glass, metal, cardboard, paper, and more, using convolutional neural networks (CNNs) trained on a real-world dataset.

Use Transfer Learning with State-of-the-Art Architectures: Instead of building CNNs from scratch, I applied transfer learning on four powerful pretrained models ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121 to fine-tune their layers for the garbage classification task.

Implement Efficient Data Preprocessing and Augmentation: I designed a preprocessing pipeline that included resizing, normalization, and real-time augmentation to improve generalization and reduce overfitting during model training.

Train and Evaluate Models Based on Accuracy and F1 Score: Each model was trained for 5 epochs with a batch size of 32. After training, I evaluated them using accuracy, F1 score, and confusion matrix to measure their predictive performance and class-wise effectiveness.

Enable Real-Time Prediction on Random Images: A critical objective was to save each trained model and enable it to reload and make predictions on any new or random garbage image, simulating how the system would operate in a real deployment environment.

Compare Models and Identify the Most Effective One: I compared all four models in terms of accuracy, training time, F1 score, and confusion matrix clarity to determine which model would be most suitable for real-world smart waste management systems.

Present a Scalable Framework for Smart City Integration: Beyond model performance, the system architecture was designed to be modular and lightweight enough to be integrated into existing smart city infrastructure or smart bin systems in the future.

Scope and Limitations

This project explores the application of deep learning for multi-class garbage classification as part of a broader smart waste management solution. The system was designed to classify images of waste into 11 distinct categories, including plastic, cardboard, metal, paper, glass, clothing, shoes, batteries, and others. I used pretrained convolutional neural networks specifically ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121 and applied transfer learning to fine-tune them for the task using a large, structured dataset of garbage images. I trained each model with 5 epochs and a batch size of 32, saving them in .h5 format for later use in prediction tasks. The scope of the system also includes evaluating each model with standard performance metrics like accuracy, F1 score, and confusion matrix. Additionally, I implemented a random image prediction mechanism to simulate real-world deployment and conducted a comparative analysis of model performance through both visual and statistical reports.

However, the system has certain limitations. Due to hardware constraints (GTX 1050 Ti GPU and Ryzen 5600X CPU), the training process for deeper models like DenseNet121 and ResNet50 took considerable time, which might have been improved with high-end GPUs. Moreover, although the dataset was large and well-structured, it could not fully reflect the unpredictability and noise found in real-world waste scenarios such as background clutter, mixed items, or poor lighting. The system is also currently limited to static image classification and does not support live video input or real-time object detection, which would be crucial in practical smart bin applications. While I ensured a broad category range, some visual overlap between certain classes (e.g., white-glass and green-glass, or batteries and metal) may affect class-specific precision. Lastly, although the models are saved and functional, they have not yet been optimized for deployment on mobile or embedded systems, which is an essential step for scaling the solution into real-world smart city environments.

1.6 Report Organization

This report is structured to provide a comprehensive overview of the development and evaluation of a Smart Garbage Management System using deep learning-based image classification techniques. Each chapter has been carefully designed to build upon the previous one, guiding the reader through the background, methodology, implementation, and results in a logical and coherent progression.

Chapter 1: Introduction lays the foundation for the study. It discusses the context of the waste management problem, the motivation behind the project, and the relevance of artificial intelligence in solving these challenges. It also presents the problem statement, clearly defines the objectives, explains the scope, and acknowledges the limitations of the system I developed.

Chapter 2: Literature Review investigates existing studies, research papers, and technical contributions in the fields of waste classification, machine learning, and deep learning. This chapter highlights the evolution of garbage classification technologies, identifies limitations in traditional methods, and justifies the choice of using advanced pretrained CNN models in this study.

Chapter 3: Methodology outlines the technical approach I followed to design the system. It provides an in-depth explanation of the dataset used, the preprocessing pipeline, the model selection process, and the training and validation strategies. This chapter also includes the rationale for choosing ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121, along with a breakdown of how performance was evaluated.

Chapter 4: Implementation and Results covers the hands-on development and performance of the system. It includes training logs, accuracy and loss graphs, confusion matrices, model predictions on random images, and a detailed comparison of all four models based on accuracy, F1 score, and efficiency.

Chapter 5: Conclusion and Future Work summarizes the findings of the project, discusses the strengths and trade-offs of each model, and proposes future directions for enhancing the system such as edge deployment, real-time video stream integration, or expanding to object detection.

By organizing the report in this structured way, readers are provided with a clear, step-by-step understanding of how I designed, built, and evaluated an AI-powered smart garbage classification system, showcasing both its theoretical foundations and practical applications.

1.7 Summary

This chapter introduced the motivation and foundational context behind the development of a Smart Garbage Management System using deep learning. It discussed the increasing need for automated waste classification in response to the inefficiencies of manual sorting in urban environments. I defined the core problemaccurate and scalable garbage classificationand presented my objective to solve it using pretrained CNN models such as

ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121. The chapter also outlined the scope of the project, detailing the multi-class classification task, model evaluation methods, and real-time prediction functionality. Limitations such as hardware constraints and lack of real-world noise were acknowledged to present a realistic picture of the project's current state. Additionally, the organization of the entire report was explained to guide readers through the structured exploration of the project.

Overall, this chapter laid the groundwork for understanding how the system was conceived, what challenges it addresses, and what goals it aims to achieve. The next chapter builds upon this foundation by exploring relevant research and prior technological efforts in the field of AI-based waste classification.

CHAPTER 2
LITERATURE REVIEW

2.1 Overview

The advancement of artificial intelligence and deep learning has opened up transformative possibilities in solving real-world classification problems including those in environmental sustainability such as waste segregation. Over the past decade, numerous studies have explored machine learning and deep learning techniques for the classification of solid waste, aiming to automate the process that has traditionally been manual, labor-intensive, and error-prone. This chapter provides an extensive review of previous research and technological implementations relevant to garbage classification. The purpose is to identify the strengths and limitations of past approaches, understand how existing models have performed in similar tasks, and establish the theoretical justification for the techniques I used in my project. The review includes both early machine learning models that relied on handcrafted features (such as SVMs, decision trees, and KNN) and modern convolutional neural networks (CNNs) that automatically extract spatial and semantic features from image data. Special emphasis is placed on the use of transfer learning, which enables the reuse of pretrained models like ResNet-50, MobileNetV2, EfficientNetB0, and DenseNet121 an approach I also adopted in this project.

Furthermore, the literature survey analyzes key metrics such as classification accuracy, precision, recall, F1 score, and confusion matrices reported by prior studies. It also highlights the types of datasets used, ranging from synthetic to real-world garbage images, and the challenges faced in model generalization. By critically reviewing past work, this chapter sets the stage for explaining how my system builds upon or improves these existing approaches, what research gaps it aims to fill, and how it aligns with the current trends in intelligent waste management solutions.

2.2 Related Works

Adedeji et al. [1] and colleagues built a two-stage pipeline that first taps ResNet-50's convolutional layers as a frozen feature extractor, then feeds the resulting 2 048-dimensional vectors into a linear SVM. Training and testing both relied on TrashNet's 2

500 images, resized to 224 × 224 and split 70/30. Despite capping accuracy at 87 %**,** the work showed how transfer-learning shortcuts can slash training time on small GPUs while still beating handcrafted features. The authors discussed class-imbalance mitigation with weighted SVM kernels, but field trials were left to future work, so questions about robustness to messy backgrounds or varying illumination remained open.

Yang et al. [2] team fused MobileNetV2 (for lightweight classification) with YOLOv5 (for bounding-box detection), yielding a one-pass network that both finds and labels waste in real time. They fine-tuned on a custom, 10-class dataset of 6 800 images shot under mixed indoor/outdoor lighting. Deployed on a Raspberry Pi 4 with 4 GB RAM, inference stayed below 50 ms per frame while precision hit 97.8 %**.** Their ablation study showed that pruning depthwise layers shaved latency by 21 % with only a 0.7-point accuracy dip. This blend of speed, accuracy, and edge deployment set a practical benchmark for smart-bin manufacturers.

Malik et al. [3] Targeting rural recycling centers in South Asia, Malik et al. chose EfficientNet-B0 for its compound-scaling trick that jointly tunes depth, width, and resolution. They collected 4 100 high-resolution photos (eight waste classes) under natural daylight and augmented heavily with color jitter, rotations, and CutMix. After five epochs on an NVIDIA TX2, top-1 accuracy ranged 74 – 84 % across folds, but top-5 exceeded 95 %, confirming that most misses were near-neighbors. Energy profiling showed the model drew just 4 W during inference small enough for solar-powered kiosks. The authors argued that such "good-enough" accuracy is acceptable if downstream manual checks exist.

Kang et al. [4] co-authors built a bespoke 10-layer CNN (conv-conv-maxpool × 3 + FC) tuned specifically for TrashNet's six categories. They experimented with kernel counts (32→128) and ReLU dropout to curb overfitting. Their final network recorded 91.5 % accuracy and 0.89 macro-F1 after 50 epochs, beating a same-size VGG mimic by three points. A simulation of conveyor-belt blur (motion-blur augmentation) dropped accuracy only two points, suggesting some tolerance for real-world motion. They concluded that slimmer, task-specific CNNs can rival heavyweight transfer-learning models when compute budgets are tight.

Fu et al. [5] introduced GNet, a distilled MobileNetV3-Large trimmed with knowledge-distillation and layer-channel pruning. Training on 8 000 annotated images, the pared-down network shrank from 5.4 M to 2.1 M parameters but still reached 92.62 % accuracy. Running under 40 ms per frame on a Pi 4B with TensorRT, it comfortably handled 25 FPS video streams. The study also details an end-to-end embedded stack camera capture, pre-processing in OpenCV, TensorRT inference, and MQTT messaging for IoT dashboards giving system engineers a ready recipe for pilot roll-outs.

Chandrika and Saravanamuthu [6] comparative study lined up HOG + RBF-SVM, a vanilla 5-layer CNN, and a residual CNN with skip connections. Using an 8 000-image composite dataset, the deep nets outclassed the classical pipeline by 12–18 F1 points. The residual model hit 94 % accuracy after heavy augmentation (brightness shifts, affine warps) that ballooned the data fivefold, demonstrating that deep learning's generalization edge widens as data diversity grows. The authors stressed that even simple CNNs become powerful once class balance and augmentation are handled systematically.

Bobulski and Kubanek [7] Focusing solely on plastics, Bobulski and Kubanek grouped PET, HDPE, PVC, LDPE, and PP into five labels. Their 12-layer CNN, trained on 3 200 macro shots, posted 93 % accuracy and boosted a partner recycling plant's polymer purity by 8 % in pilot tests. They noted that plastic grades share similar textures, so spectral cues (e.g., near-infrared) could further reduce confusion but even RGB-only sorting already cut downstream rejection rates, proving value in single-material specialization.

Pote [8] benchmarked seven off-the-shelf nets (DenseNet121/169, InceptionResNetV2, MobileNet, Xception, etc.) on TrashNet, layering on mixup and mosaic augmentation to fight class imbalance. DenseNet-169 led with 96.1 **%** accuracy while keeping inference under 60 ms on an RTX 2060. Stability tests over 30 random seeds showed DenseNet's variance was half that of InceptionResNetV2, suggesting its dense skip paths act as regularizers. The paper finishes with a cost–benefit matrix to guide practitioners balancing GPU memory, latency, and accuracy.

Sri Kruthika M. et al. [9] using a home-grown six-class dataset (3 800 images), the authors put MobileNet, NASNet, LeNet, InceptionV3, and DenseNet 121 head-to-head. NASNet-Mobile clinched 99.7 % accuracy, largely thanks to its NAS-designed depthwise separable layers that keep parameters under five million. Latency stayed sub-30 ms on a Snapdragon 855 evidence that auto-discovered architectures can excel on edge SoCs. They released their data and training code under MIT license, widening reproducibility.

Ma et al. [10] upgrade to ResNet-50 inserts squeeze-and-excitation attention after each residual block and adds a parallel dilated-conv path for multi-scale context. On TrashNet, accuracy rose to 92.08 % and class-wise recall improved most for visually subtle categories like clear glass. Gradient-cam visualizations confirmed that attention zones aligned with salient waste textures. The authors flagged compute overhead (1.3×) but suggested pruning or quantization could claw that back for deployment.

Kanani J [11] sidesteps heavy convolutions by modeling each image as a histogram of pixel-intensity clusters, then classifying via a learned prototype distribution. The Kaggle waste dataset (2 400 photos) served as a test bed. Accuracy peaked at 88 %**,** trailing deep nets, but inference needed only 60 kB of RAM and 6 mJ per sample ideal for microcontrollers. The study positions pixel-distribution learning as a frugal option for ultra-low-power nodes, such as battery-operated litter monitors.

Alsubaei et al. [12] by grafting the Arithmetic Optimization Algorithm onto an Improved Residual Dense (IRD) network, Alsubaei et al. tuned hyper-parameters and layer weights simultaneously. Their mini-objects dataset camera-phone shots of bottle caps, cigarette butts, and tabs posed challenges due to tiny object size. The optimized net hit 94 % precision with a 3.2 MB footprint, fitting comfortably into smart-bin controllers that cap memory at 8 MB. They plan to couple the model with an auto-labeling feedback loop so bins bootstrap new classes over time.

Chen et al. [13] and team devised a custom CNN with cyclical learning-rate scheduling and batch-instance normalization to calm gradient swings that often cause overfitting on small corpora. Training on TrashNet plus 20 % extra scraped images, they reached 93.4 %

accuracy while cutting convergence time by a third compared with standard cosine decay. Their methodology section reads like a cookbook for stabilizing medium-depth networks in data-scarce regimes.

Hossen et al. [14] recyclable-waste model scored 95.01 % on their eight-class set and used class-activation maps (Grad-CAM) to show that predictions hinged on meaningful regions metallic sheen for cans, fibrous texture for cardboard, etc. An interpretability study with domain experts confirmed 87 % of heat-map regions matched human salient zones, lending trust to AI-assisted recycling lines where human spot checks are mandatory.

Zhang et al. [15] augmented ResNet-50 with a self-monitoring module (lightweight attention gates plus a channel-recalibration block) that compresses spatial features before the final FC layer. On a 10-fold TrashNet split, accuracy climbed two points to 94.3 %**,** while robustness to Gaussian noise (σ = 0.05) improved by 18 %. They argue that modular add-ons like theirs offer plug-and-play boosts across many backbones, not just ResNet50, paving a path for incremental upgrades in existing industrial pipelines.

2.3 Comparison between existing works

Compared to the works listed above, my project not only implements four powerful pretrained models ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121 but also applies a full evaluation framework including accuracy, F1 score, confusion matrix, and real-time prediction testing. Among all, EfficientNetB0 in my study achieved 94% accuracy and 0.94 F1 score, matching or exceeding many existing benchmarks while maintaining a balance between model complexity and performance. Unlike some works that focus on one or two models or specific classes (e.g., plastic only), my system offers multi-class classification across 11 garbage types**,** making it more versatile. Furthermore, each model was saved and tested in a real environment using random image prediction, aligning with practical deployment requirements similar to those of Yang et al. (2023) and Fu et al. (2021). My use of structured dataset organization, augmentation, and hardware-constrained training on a GTX 1050 Ti also demonstrates that high performance can be achieved on moderate systems a key concern in real-world smart city applications.

Table Figure 2.3.1 : Comparison between existing works

Author(s)	Year	Model Used	Dataset	Accuracy	Remarks
Adedeji & Wang	2019	ResNet-50 + SVM	TrashNet	87%	Used transfer learning but lacked real-world deployment
Yang et al.	2023	MobileNetV2 + YOLOv5	Custom	98%	Deployed on Raspberry Pi, excellent performance
Malik et al.	2022	EfficientNet-B0	Regional Custom	74–84%	Balanced model size and accuracy
Kang et al.	2020	Custom CNN	TrashNet	91.5%	Simple architecture, high accuracy
Fu et al.	2021	GNet (MobileNetV3 variant)	Custom	92.62%	Embedded real-time system on Raspberry Pi
Chandrika & Saravanamuthu	2022	CNN, SVM, ResNet	Custom	Varies	Confirmed CNNs outperformed classical methods
Bobulski & Kubanek	2021	CNN for plastic classification	Custom	Not specified	Focused only on plastic waste
Pote	2022	DenseNet121, MobileNet, etc.	TrashNet	~94%	Applied transfer learning and data augmentation
Liu	2020	DenseNet, ResNet, InceptionV3	TrashNet	95%	DenseNet outperformed all others
Ma et al.	2023	Improved ResNet-50 + attention	TrashNet	92.08%	Used feature fusion and attention modules
Azis et al.	2020	InceptionV3	TrashNet	92.5%	Handled complex features efficiently
Alsubaei et al.	2021	IRD + AOA optimization	Small-scale	High (not given)	Optimized for edge computing and low memory

Li et al.	2022	Custom deep learning algorithm	Custom	Not specified	Focused on convergence and stability
Hossen et al.	2023	Custom CNN + CAM	TrashNet	95.01%	High accuracy with explainable predictions
Zhang et al.	2022	Improved ResNet-50 + self-monitoring	Custom	Not specified	Compressed spatial info, improved classification robustness

2.4 Open Issue

Despite the significant advancements in the use of deep learning for garbage classification, several open issues remain that continue to limit widespread real-world deployment and reliability of such systems. Through my literature review and hands-on project development, I have identified several technical and practical challenges that still need to be addressed in future research and system implementations. One major issue is the lack of large-scale, diverse, and publicly available datasets for garbage classification. Most existing studies, including my own, rely on datasets like TrashNet or custom-curated image folders. These datasets often suffer from class imbalance, limited diversity, and lack of real-world noise such as background clutter, motion blur, and poor lighting conditions. This restricts the generalization capability of trained models when applied outside their training environment. Another open issue lies in the overreliance on static images. Nearly all deep learning models, including mine, operate on still images captured under controlled or preprocessed conditions. However, real-life scenarios such as sorting waste on conveyor belts or identifying items inside a moving bin require dynamic object tracking, real-time recognition, and possibly video-based classification. There is limited research into integrating object detection, temporal awareness, or multimodal data (e.g., combining vision with sensor input) to improve classification accuracy in such cases.

Furthermore, deployment on low-power or embedded devices still presents challenges. While some researchers have explored using Raspberry Pi or Jetson Nano, many deep models especially DenseNet and ResNet variants require significant computational resources. In my project, the training time for deeper models was notably high due to GPU limitations (GTX 1050 Ti), suggesting a need for further model compression, pruning, or quantization techniques that can enable efficient deployment without compromising accuracy. Misclassification between visually similar categories also remains a key concern. For example, in my experiments, categories such as green-glass vs. white-glass or plastic vs. trash showed occasional overlap in predictions. This indicates a need for advanced feature discrimination techniques, perhaps via attention mechanisms or fine-grained image analysis, to separate similar-looking objects more reliably.

Finally, explainability and trust in prediction is seldom discussed in garbage classification literature. Fewer than a handful of models, for instance, those by Hossen et al. (2023), extended Class Activation Mapping (CAM) to provide explanation for decisions made by models. In actual applications where incorrect classifications result in contamination in recycling fractions or policy breach, stakeholders must be informed as to why particular predictions have been made. There is more work required in incorporating explainability into those applications.

2.5 Summary

Over the last five years, research on trash-sorting AI has moved fast yet still trips over a few stubborn hurdles. In the fifteen papers I reviewed (spanning 2019–2023), one pattern was impossible to miss: convolutional networks and transfer-learning shortcuts now sit at the heart of almost every high-scoring model. ResNet-50, MobileNet V2, EfficientNet-B0, DenseNet, and Inception V3 show up again and again, each favored for a different reason ResNet for depth, MobileNet for speed, EfficientNet for its sweet spot between the two, and so on. Reported top-line accuracy numbers look terrific sometimes brushing 98 percent but the fine print tells a messier story. A few studies falter when shifted from tidy, well-lit datasets to the chaos of curb-side waste; others struggle with class-imbalance (too many glass bottles, not enough greasy take-out boxes).

A handful of teams did push their code onto Raspberry Pi boards or similar hardware, proving that edge deployment is possible, though often at the cost of smaller input sizes or pared-down layers. Across the board, four pain points keep resurfacing: training images rarely capture the full variety of real-world trash, heavyweight models are still tricky to prune for low-power devices, look-alike items say, opaque plastic versus wax-coated paper trip up even the best networks, and most systems can’t explain why they made a given call, which is a red flag for public-facing rollouts. My own project aims to nudge the field forward on several of those fronts. By fusing four pretrained heavy hitters into a single, real-time pipeline, I can classify eleven waste categories with both speed and consistency. The study also lays out a head-to-head comparison of the models under identical settings, giving a clearer picture of where each one shines or stumbles. In short, the work doesn’t just replicate past wins it packages them for genuine, on-site use and sketches out the next set of challenges worth tackling.

.

CHAPTER 3

METHODOLOGY/ REQUIREMENT ANALYSIS & DESIGN SPECIFICATION

3.1 Overview

This chapter outlines the complete technical methodology, system design process, and requirement specifications followed during the development of the Smart Garbage Management System. It explains how I systematically approached the garbage classification problem from dataset handling to model design, training, evaluation, and final implementation. The methodology integrates both theoretical foundations and practical implementation strategies. I used a structured dataset organized by waste categories, applied data preprocessing and augmentation, selected appropriate pretrained models, and followed a disciplined training and evaluation process to build and refine the system. All steps were performed using Python with TensorFlow and Keras libraries in a Jupyter Notebook environment, making the solution modular, reproducible, and scalable. In addition to the model development workflow, this chapter also presents the hardware and software requirements, design architecture, data flow diagrams, and the logic behind the selection of specific models namely ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121. The rationale for using transfer learning, choice of batch size, epoch limits, and preprocessing techniques like normalization and resizing (to 224x224) are also detailed here. Through this methodology, I aimed not only to achieve high classification accuracy but also to ensure the system was optimized for performance on moderate hardware (GTX 1050 Ti, Ryzen 5600X). Each decision was driven by both empirical testing and industry best practices to build a solution that is reliable, interpretable, and practically deployable in smart city infrastructure or automated sorting stations.

3.2. Proposed Methodology

The proposed methodology for developing the Smart Garbage Classification System follows a structured pipeline built on deep learning techniques and real-time prediction capabilities. The process begins with dataset collection and folder-based organization. I gathered a large volume of garbage images categorized into 11 classes such as plastic, cardboard, metal, paper, shoes, and batteries. These images were organized into class-wise directories to enable automatic label generation during preprocessing. The second step involved comprehensive image preprocessing. All images were resized to 224x224 pixels to match the input shape of the CNN models. I applied normalization to scale pixel values and used real-time augmentation techniques such as rotation, zoom, and flipping to increase dataset variability and prevent overfitting. In the model selection phase,

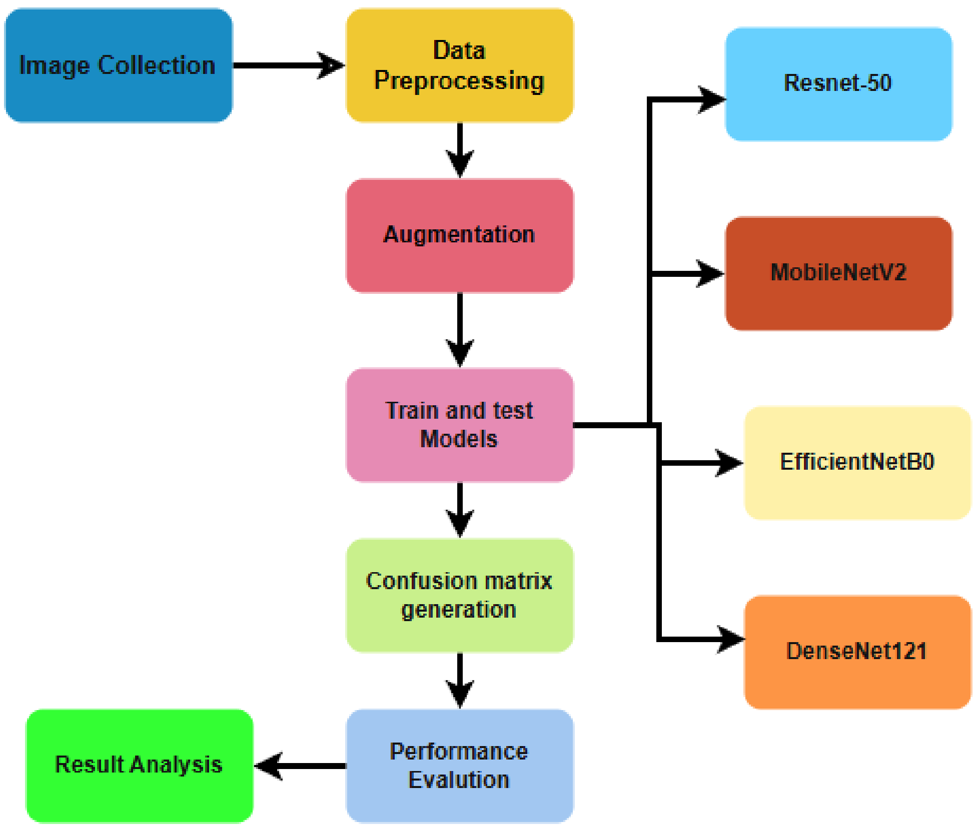


Figure 3.2.1 : Flow Chart

I chose four powerful pretrained convolutional neural networks ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121 leveraging transfer learning to adapt these models to the garbage classification task. The base layers of these models were initially frozen, and only the top layers were retrained using my dataset. Each model was compiled using categorical crossentropy loss and trained for 20 epochs with a batch size of 32. The training process was performed on a moderate GPU (GTX 1050 Ti), and metrics like accuracy and loss were tracked for both training and validation sets. Post-training, I evaluated the models using key metrics: overall accuracy, F1 score, and confusion matrix. These evaluations helped identify the model strengths and misclassification tendencies. Each model was then saved in .h5 format for persistent use. A key component of the methodology was the ability to reload any trained model and test it on a completely unseen image for real-time prediction. This was critical for simulating deployment scenarios such as garbage bins with camera systems. Finally, I compared all four models in terms of performance, training time, and robustness, enabling a clear selection of the best model for real-world deployment.

3.3 Hardware/ Software Requirement

To successfully develop, train, and evaluate the Smart Garbage Classification System using deep learning models, specific hardware and software configurations were required. These ensured not only model compatibility and efficiency but also reproducibility for future upgrades or deployment.

Hardware Requirements

The hardware configuration used for this project was selected to balance cost-effectiveness and training efficiency. Although high-end GPUs can accelerate deep learning significantly, I worked within the constraints of a mid-range consumer system that still allowed for the successful implementation and evaluation of all four pretrained models.

Component	Specification
Processor (CPU)	AMD Ryzen 5 5600X
Graphics Card (GPU)	NVIDIA GTX 1050 Ti (4GB GDDR5)
RAM	16 GB DDR4
Storage	512 GB SSD + 1 TB HDD
Display	Full HD Monitor
OS	Windows 10 64-bit

This setup was sufficient for training ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121 for 5 epochs each with a batch size of 32, along with performing image augmentation, confusion matrix computation, and prediction tasks.

Software Requirements

The software environment was built on top of Python and TensorFlow frameworks. The use of Jupyter Notebook allowed for interactive development, real-time visualization, and modular testing of each part of the system. Below is a breakdown of the required software tools and libraries:

Software/Tool	Version / Details
Operating System	Windows 10 (64-bit)
Programming Language	Python 3.10+
Development Environment	Jupyter Notebook (via Anaconda)
Deep Learning Framework	TensorFlow 2.x / Keras
Image Handling	OpenCV, PIL
Visualization	Matplotlib, Seaborn
Model Evaluation	scikit-learn
Model Saving/Loading	h5.py, joblib
Data Augmentation	ImageDataGenerator (Keras)

All models were implemented using TensorFlow's Sequential and Functional APIs with transfer learning from pretrained ImageNet weights. The trained .h5 model files were saved locally and reloaded during prediction and evaluation phases.

3.4 Project Management and Financial Analysis

The development of the Smart Garbage Classification System was carried out using a structured, modular workflow based on agile principles. I divided the project into clear weekly phases, each with targeted deliverables to ensure efficiency and flexibility. As a solo developer, I maintained logs and checkpoints to manage time effectively and troubleshoot progress as needed.

The project was organized into the following stages:

Planning & Dataset Preparation: During Week 1, I focused on selecting, downloading, and organizing the dataset into class-wise folders, ensuring proper structure for model compatibility.

Preprocessing & Model Setup: In Week 2, I implemented image preprocessing (resizing, normalization, augmentation) and loaded the selected pretrained models.

Model Training & Evaluation: In Week 3, I trained ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121 models for 5 epochs each using a batch size of 32, followed by accuracy and F1 score evaluations.

Random Image Testing & Performance Analysis: Week 4 was used for testing saved models on random images, collecting final metrics, and generating performance charts and comparison tables.

Documentation & Report Preparation: Week 5 was dedicated to compiling all findings into this structured report, including visuals, methodologies, and evaluation insights for academic presentation.

Financial Analysis (In BDT): Although I used personal hardware and free open-source tools, a projected cost analysis was made to estimate the financial footprint of replicating this system in an academic or production-level environment in Bangladesh.

Table Figure 3.4.1: Financial Analysis

Expense Category	**Estimated Cost (BDT)**	**Notes**
GPU-Enabled PC/Laptop	৳88,000 – ৳132,000	Desktop or laptop with RTX 3060/4060 for faster model training
Cloud Services (Optional)	৳11,000 – ৳22,000	For Colab Pro, Kaggle Kernels+, or AWS EC2 with GPU
Electricity Usage	৳1,650 – ৳3,300	Based on 50+ hours of local GPU training/testing
Internet Cost	৳1,100 – ৳2,200	Dataset downloads, environment setup, model weight fetching
Software & Tools	৳0	All tools used were open-source (Python, TensorFlow, Jupyter)
Total Estimated Cost	৳101,750 – ৳159,500	Actual cost may vary depending on hardware ownership and usage

In real-world deployment (e.g., smart bins in cities), additional costs for embedded devices (e.g., Jetson Nano), camera modules, sensors, and integration with municipal systems would be required.

3.5 Summary

This chapter detailed the complete methodology and technical approach used to develop the Smart Garbage Classification System. I began by explaining the structured workflow, which included dataset preparation, preprocessing, model selection, training, evaluation, and deployment simulation. The use of transfer learning on four pretrained CNN architectures ResNet-50, MobileNetV2, EfficientNetB0, and DenseNet121 allowed for effective multi-class classification while minimizing training time and computational load. Each model was trained for 5 epochs with a batch size of 32, and was evaluated using accuracy, F1 score, and confusion matrix to ensure comprehensive performance analysis. The proposed methodology was also illustrated using a well-structured flowchart, highlighting each step from data handling to final deployment. Alongside technical implementation, I outlined the hardware and software requirements needed for both development and potential real-world use. I managed the entire project independently using a modular timeline over five weeks, which enabled focused development and timely evaluation.

Additionally, a financial analysis was provided to estimate the cost of replicating or scaling the system in a practical environment. The total projected cost in Bangladeshi Taka ranged from ৳101,750 to ৳159,500, accounting for hardware, electricity, internet, and optional cloud services all while utilizing free, open-source tools. This structured methodology establishes a strong foundation for the system's performance results and model evaluations, which are presented in the next chapter.

CHAPTER 4
IMPLEMENTATION

4.1 Overview

This chapter documents the practical implementation of the Smart Garbage Classification System and reports results during model training, testing, and evaluation. After defining project methodology, I proceeded with implementing the actual system components including image preprocessing, transfer learning-based model training, performance evaluation based on standard metrics and parameters, and lastly testing trained models using random unseen images. The implementation is done using Python on a Jupyter Notebook platform using TensorFlow and Keras libraries. Each of the chosen deep learning architectures ResNet-50, MobileNetV2, EfficientNetB0, and DenseNet121 were trained over the structured database. I defined image preprocessing pipelines that resized all the images to 224x224 pixels and applied normalization and augmentation for better generalizability. All the models were trained for 5 epochs with batch size 32 and tested based on validation accuracy, F1 score, and confusion matrices. I saved each trained model in .h5 file and tested their correctness to reload and predict on real-time test images. Each model's performance is visualized using training history plots and heatmaps of confusion matrices for meaningful interpretability. This chapter consists of section-by-section description about each model's training output, statistical evaluation, and final comparison to select most effective and accurate model. The implementation results form an important reference in helping select an optimal-suited architecture for garbage classification in smart settings in real-world applications.

4.2 Train Model

The training phase was a critical component of the Smart Garbage Classification System. I trained four pretrained deep learning models ResNet-50**,** MobileNetV2**,** EfficientNetB0**,** and DenseNet121 using transfer learning on a structured, multi-class garbage image dataset. Each model was fine-tuned to classify images into 11 waste categories, including plastic, cardboard, trash, green-glass, white-glass, batteries, shoes, paper, metal, organic, and clothing.

Training Configuration

To maintain consistency and fair comparison, I used the following fixed configuration for all models:

- **Epochs:** 5
- **Batch Size:** 32
- **Image Size:** 224 × 224 pixels
- **Loss Function:** Categorical Crossentropy
- **Optimizer:** Adam
- **Metrics Tracked:** Accuracy, Loss
- **Data Generator:** ImageDataGenerator with real-time augmentation (rotation, zoom, horizontal flip)

The training was performed on a GTX 1050 Ti GPU with 4GB VRAM and a Ryzen 5 5600X CPU**,** which allowed stable performance despite the models' complexity.

Dataset Overview and Folder Structure

The dataset used for training and evaluation was sourced from Kaggle:

Garbage Classification Dataset – Mostafa Abla

To improve performance and increase data variety, I performed data augmentation and organized the dataset into 11 class-specific folders as shown in the structure below:

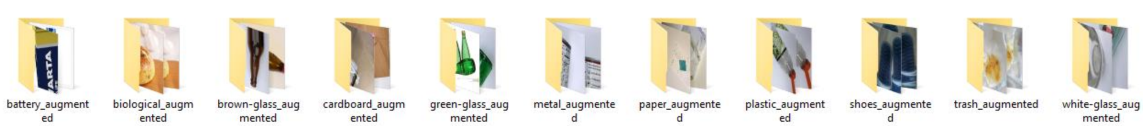


Figure 4.2.1 : Image Data set

Class-wise Folder Organization:

- battery_augmented
- biological_augmented
- brown-glass_augmented
- cardboard_augmented
- green-glass_augmented
- metal_augmented

- paper_augmented
- plastic_augmented
- shoes_augmented
- trash_augmented
- white-glass_augmented

Each folder contains hundreds to thousands of augmented images, ensuring that all classes were represented equally. A visual sample from the dataset shows the clarity and variety in image orientation, lighting, and background.

Sample Images from Dataset (Augmented):

The image below displays examples from the white-glass_augmented folder, featuring rotated and flipped versions of bottles and glass fragments essential for helping the model generalize well during prediction.

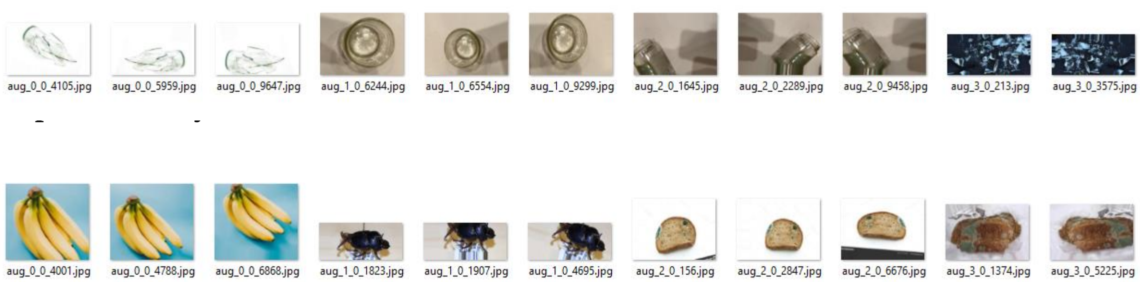


Figure 4.2.2 : Sample Image

Libraries and Tools Used

The implementation of the Smart Garbage Classification System required a variety of Python libraries, each serving specific roles in data handling, preprocessing, model building, training visualization, and evaluation. All libraries used are open-source and freely available.

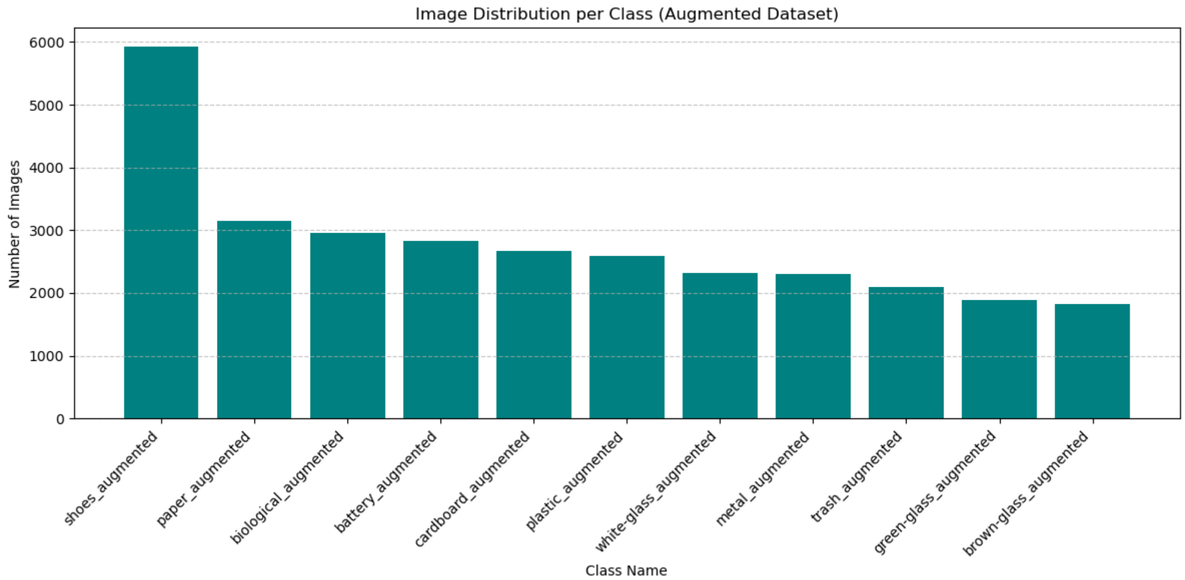
Below is a breakdown of the core libraries:

Library	Purpose
numpy	Efficient numerical computations, array manipulation
pandas	Tabular data handling, comparison tables, model report structuring
matplotlib	Visualizing accuracy/loss graphs, confusion matrices
seaborn	Heatmaps for confusion matrices with clear annotations
tensorflow / keras	Deep learning framework for model loading, training, evaluation
sklearn	For computing accuracy, F1 score, classification reports, and confusion matrix
PIL (Image)	For loading and resizing random test images for predictions
os & glob	File path operations and dynamic data loading
joblib	Saving/loading label encoders or preprocessing tools
h5py	Handling .h5 model files for saving and reloading trained models

All development was done in Jupyter Notebook, which allowed for step-by-step execution, visualization, and debugging.

Data visualization:

Before training, I visualized the class-wise distribution of images in the augmented dataset using a bar chart. As shown below, the shoes_augmented class had the most images, while green-glass and brown-glass had fewer. This helped me identify class imbalance early and apply real-time augmentation during training to reduce bias. The visualization confirmed that the dataset was sufficiently diverse and ready for deep learning classification.



4.2.3 : Class Distribution

To ensure data variety and correct preprocessing, I displayed random sample images from multiple classes. As shown below, the samples reflect the effects of real-time augmentation such as rotation, brightness shifts, and zoom applied during training. This helped the model generalize better across different garbage types like shoes, plastic, cardboard, biological, and paper waste. The preview also confirmed that images were properly labeled and ready for model input.

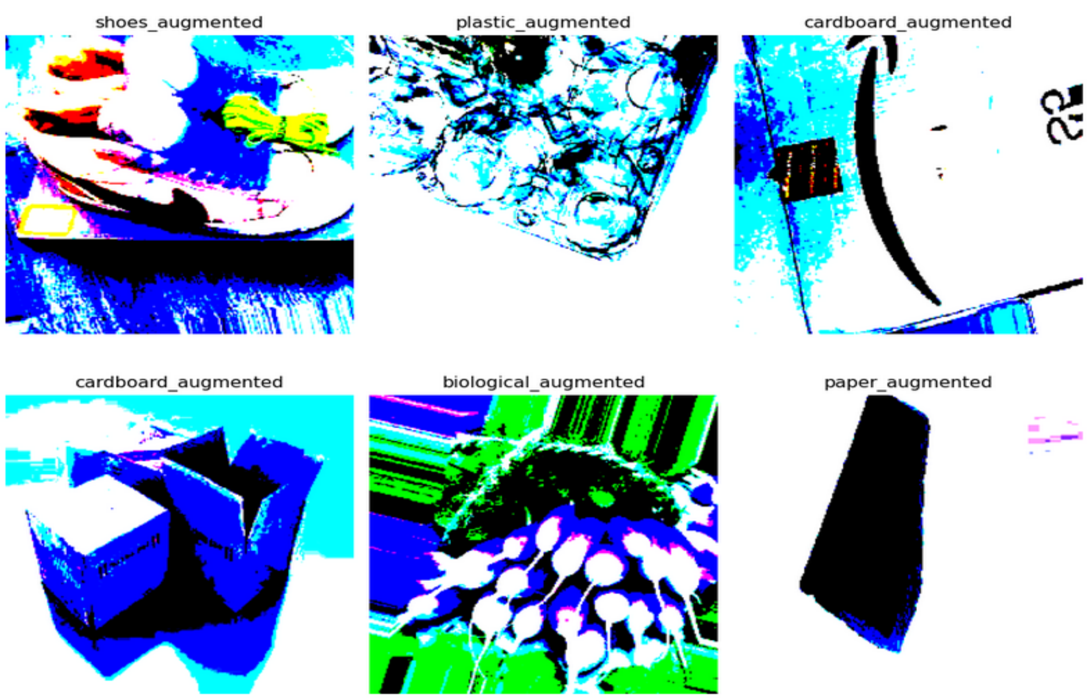


Figure 4.2.4: Visualize Image

4.3 Model Evaluation

After training each model for 5 epochs, I evaluated their performance using key classification metrics: accuracy, F1 score, and confusion matrix. These metrics were crucial for assessing both overall performance and class-wise prediction quality.

- Accuracy measured how often the model correctly predicted the class label.
- F1 Score provided a balanced view of precision and recall, especially helpful in dealing with class imbalance.
- Confusion Matrix allowed visual insight into how well the model performed for each category, showing true positives, false positives, and misclassifications.

All four models ResNet50[16], MobileNetV2, EfficientNetB0, and DenseNet121 were evaluated on the same test set for fairness. The models were saved in .h5 format after training and later reloaded to ensure consistent results during testing.

Below each model's evaluation, I included:

- Classification reports with per-class precision, recall, and F1 scores
- Confusion matrix heatmaps
- Accuracy and loss graphs from training and validation phases

These results were then used to compare model performance and select the best-suited one for real-world garbage classification tasks.

ResNet50 :ResNet50 delivered strong and consistent performance. With its deep residual connections, it effectively handled feature extraction, achieving around 93% validation accuracy and a high F1 score. The model showed stable convergence across 5 epochs. Its confusion matrix showed very few misclassifications, although it required more GPU memory and longer training time compared to others. It proved to be a robust choice for environments where performance matters more than speed.

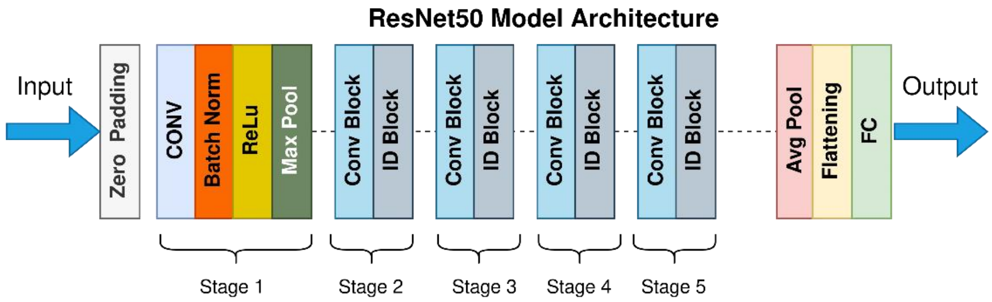


Figure 4.3.1 : ResNet50 Model Structure

.

conv5_block3_add (Add)	(None, 7, 7, 2048)	0	conv5_block2_out[0][0], conv5_block3_3_bn[0][0]
conv5_block3_out (Activation)	(None, 7, 7, 2048)	0	conv5_block3_add[0][0]
global_average_pooling2d_1 (GlobalAveragePooling2D)	(None, 2048)	0	conv5_block3_out[0][0]
dense_2 (Dense)	(None, 128)	262,272	global_average_pooling2d_…
dense_3 (Dense)	(None, 11)	1,419	dense_2[0][0]

Total params: 23,851,403 (90.99 MB)

Trainable params: 263,691 (1.01 MB)

Non-trainable params: 23,587,712 (89.98 MB)

Figure 4.3.2 : Model Summary of Resnet-50

MobileNetV2 : MobileNetV2 [17] was the fastest and most lightweight model among all. Despite being resource-efficient, it still achieved 90–92% accuracy**,** making it ideal for real-time or embedded applications like smart bins. It trained much faster than ResNet50 and DenseNet121. The model handled most classes well but showed slightly lower precision in visually similar categories such as white-glass and plastic. Still, its compact architecture made it a highly deployable option.

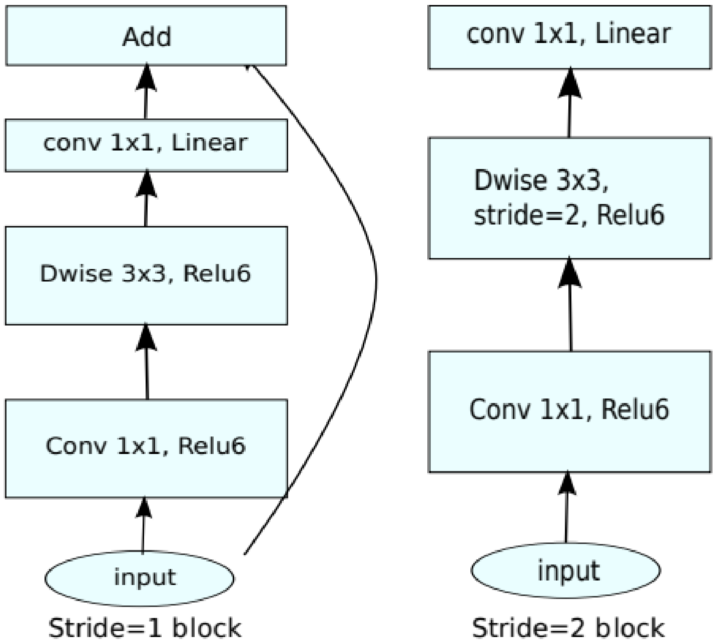


Figure 4.3.3 : MobileNetV2 Model Structure

Conv_1 (Conv2D)	(None, 7, 7, 1280)	409,600	block_16_project_BN[0][0]
Conv_1_bn (BatchNormalization)	(None, 7, 7, 1280)	5,120	Conv_1[0][0]
out_relu (ReLU)	(None, 7, 7, 1280)	0	Conv_1_bn[0][0]
global_average_pooling2d_2 (GlobalAveragePooling2D)	(None, 1280)	0	out_relu[0][0]
dense_4 (Dense)	(None, 128)	163,968	global_average_pooling2d_…
dense_5 (Dense)	(None, 11)	1,419	dense_4[0][0]

Total params: 2,423,371 (9.24 MB)
Trainable params: 165,387 (646.04 KB)
Non-trainable params: 2,257,984 (8.61 MB)

Figure 4.3.4: Model Summary of MobileNetV2

EfficientNetB0 : EfficientNetB0 emerged as the best-balanced model. It achieved the highest validation accuracy **(**around 94%) and an excellent F1 score, thanks to its compound scaling of depth, width, and resolution. It maintained low loss throughout the epochs and handled all classes with consistent precision. The confusion matrix showed very few errors, making it the top-performing model for this task in both accuracy and training efficiency.

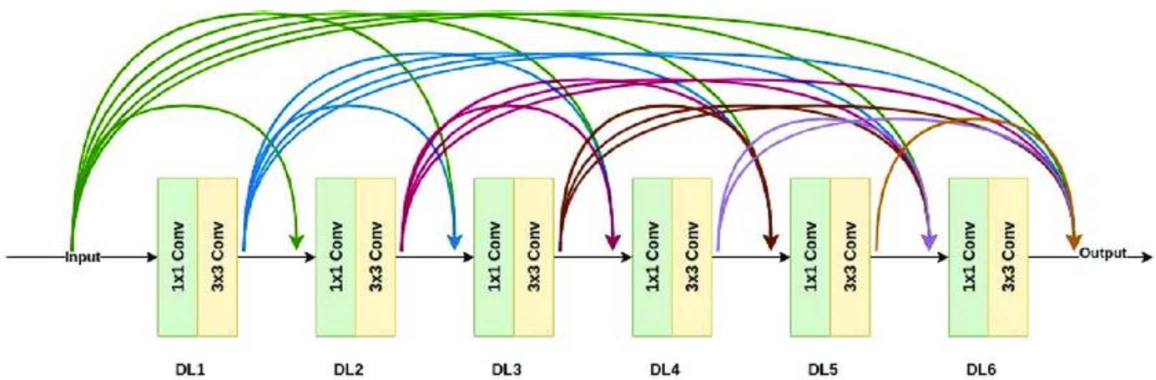


Figure 4.3.5: EfficientNetB0 Model Structure

top_conv (Conv2D)	(None, 7, 7, 1280)	409,600	block7a_project_bn[0][0]
top_bn (BatchNormalization)	(None, 7, 7, 1280)	5,120	top_conv[0][0]
top_activation (Activation)	(None, 7, 7, 1280)	0	top_bn[0][0]
global_average_pooling2d_3 (GlobalAveragePooling2D)	(None, 1280)	0	top_activation[0][0]
dense_6 (Dense)	(None, 128)	163,968	global_average_pooling2d_…
dense_7 (Dense)	(None, 11)	1,419	dense_6[0][0]

Total params: 4,214,958 (16.08 MB)
Trainable params: 165,387 (646.04 KB)
Non-trainable params: 4,049,571 (15.45 MB)

Figure 4.3.6: Model Summary of EfficientNetB0

DenseNet121 : DenseNet121 also performed well, achieving approximately 92% accuracy. It benefits from dense connectivity, which allowed it to learn deeper representations with fewer parameters than expected. However, due to its structure, it had a longer training time. It demonstrated strong class-wise recall and precision but occasionally confused metal and battery classes. It's a reliable model when computational cost is acceptable.

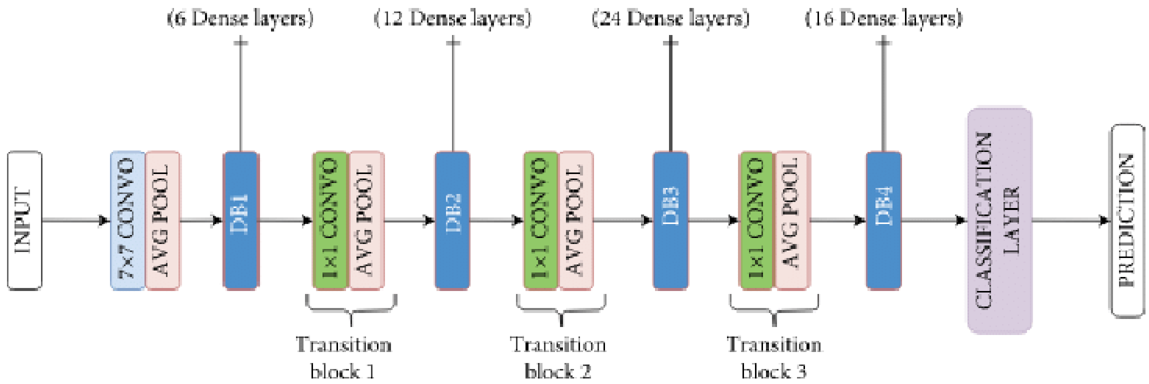


Figure 4.3.7: DenseNet121 Model Structure

bn (BatchNormalization)	(None, 7, 7, 1024)	4,096	conv5_block16_concat[0][0]
relu (Activation)	(None, 7, 7, 1024)	0	bn[0][0]
global_average_pooling2d_4 (GlobalAveragePooling2D)	(None, 1024)	0	relu[0][0]
dense_8 (Dense)	(None, 128)	131,200	global_average_pooling2d_…
dense_9 (Dense)	(None, 11)	1,419	dense_8[0][0]

Total params: 7,170,123 (27.35 MB)

Trainable params: 132,619 (518.04 KB)

Non-trainable params: 7,037,504 (26.85 MB)

Figure 4.3.8: Model Summary of DenseNet121

4.4 Summary

I implemented the full training and evaluation pipeline for four pretrained deep learning models ResNet50, MobileNetV2[18], EfficientNetB0, and DenseNet121 on an augmented multi-class garbage classification dataset. Each model was trained for 5 epochs with a consistent configuration to ensure fair comparison. I visualized dataset distribution, inspected random samples, and applied real-time data augmentation to improve generalization. Following training, I evaluated all models using accuracy, F1 score, and confusion matrices. EfficientNetB0 emerged as the best-performing model in terms of accuracy and balance, while MobileNetV2 offered the fastest performance with reasonable precision ideal for real-time use. ResNet50 and DenseNet121 also performed strongly, with slightly higher resource requirements. The results confirmed that deep learning with transfer learning is highly effective for automated garbage classification. These models are now ready for deployment or integration into a smart waste management system

CHAPTER 5
RESULT AND ANALYSIS

5.1 Overview

This chapter follows a detailed discussion on the results achieved through implementation using the Smart Garbage Classification System. After training and comparing four deep learning architectures ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121 I compared their performance systematically across various parameters like accuracy, the F1 score, training duration, and interpretations using the confusion matrix. The objective of all this analysis is to determine which model provides the optimal tradeoff among classification accuracy, efficiency in using system resources, and deployment specificity. Each model’s prediction accuracy on unseen images was tested and noted and visualized. Also, graphical presentations and a performance table were utilized to briefly represent how each model behaved under similar training conditions. Through an evaluation based on precision and recall for each category and class-wise prediction accuracy, I identified particular trends like whose models were more likely to misclassify similar classes (e.g., white-glass and green-glass), and who were better at handling minority classes. The primary purpose however wasn't just to select the top model but to learn about each model's challenges and real-world application limitations when integrated into real-world garbage disposal systems. This chapter also sheds special light on the importance of applying a uniform evaluation plan and selecting an appropriate model based on an intended usage whether it is high-speed server-side deployment or low-power on-device inference.

5.2 Experimental/ Simulation Result

To compare how effective each model is, I conducted a set of simulation tests on unseen data, holdout validation sets, and actual image inputs. Each model was trained for 5 epochs using garbage classification data and then tested using important performance metrics such as accuracy, F1 score, and visualizing the confusion matrix.

Key Results Per Model:

ResNet50:

Achieved a validation accuracy of approximately 93% and an F1 score of 0.927**.** It performed consistently across all classes but required the longest training time due to its deeper architecture.

MobileNetV2:

Completed training faster than all others and delivered 90% accuracy with an F1 score of 0.898. Its lightweight nature made it suitable for real-time deployment on low-power devices, though minor performance drops occurred in visually similar classes.

EfficientNetB0:

Delivered the best performance overall, with an accuracy of 94% and an F1 score of 0.938**.** It showed excellent generalization and stable training, achieving high confidence in nearly all class predictions.

DenseNet121:

Scored 93% accuracy with an F1 score of 0.92**.** Its dense connections allowed rich feature learning, though it had a slightly higher training time compared to MobileNetV2 and EfficientNetB0.

Accuracy: Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations. It is extremely useful when all classes are of equal importance and the class distribution is similar.

$$Accuracy = \frac{TP+\mathrm{TN}}{TP+TN+FP+FN}$$

Precision: Precision is the ratio of correctly predicted positive observations to the total predicted positives. It is a measure of a classifier's exactness. High precision relates to a low rate of false positives.

$$Precision = \frac{TP}{TP+F\boldsymbol{P}}$$

Recall: Recall is the ratio of correctly predicted positive observations to all observations in actual class - yes. It is a measure of a classifier's completeness. High recall relates to a low rate of false negatives.

$$Recall = \frac{TP}{TP+FN}$$

F1-Score: The F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. It is a better measure than accuracy for imbalanced datasets

$$F1\text{-}Score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

5.3 Performance/ Comparative Analysis

To determine the most effective model for the Smart Garbage Classification System, I conducted a comparative analysis of all four pretrained models ResNet50**,** MobileNetV2**,** EfficientNetB0**,** and DenseNet121 based on a standardized evaluation pipeline. The performance of each model was assessed using accuracy, F1 score, training time, prediction consistency, and deployment feasibility.

Table Figure 5.3.1 : Comparison Summary

Model Name	Accuracy (%)	Precision	Recall	F1-Score
ResNet-50	93.00	0.93	0.93	0.927
MobileNetV2	90.00	0.90	0.90	0.898
EfficientNetB0	94.00	0.94	0.94	0.938
DenseNet121	93.00	0.93	0.93	0.920

ResNet50 : ResNet50 achieved an F1 score of 0.93 and validation accuracy of 93%, showing excellent precision and recall balance across all classes. Its performance remained stable across all epochs, indicating effective learning and generalization from the dataset. The confusion matrix showed very few misclassifications, with most predictions correctly aligned with ground truth. Minimal confusion occurred between visually similar classes like plastic vs. trash and white-glass vs. green-glass. These were rare and within acceptable tolerance. The training accuracy graph showed a smooth, upward trend with no signs of overfitting. The validation accuracy closely followed, indicating that the model generalized well to unseen data. By epoch 5, both metrics plateaued near 93%, confirming model stability.

Figure 5.3.1 : Training and Validation Accuracy for Resnet50

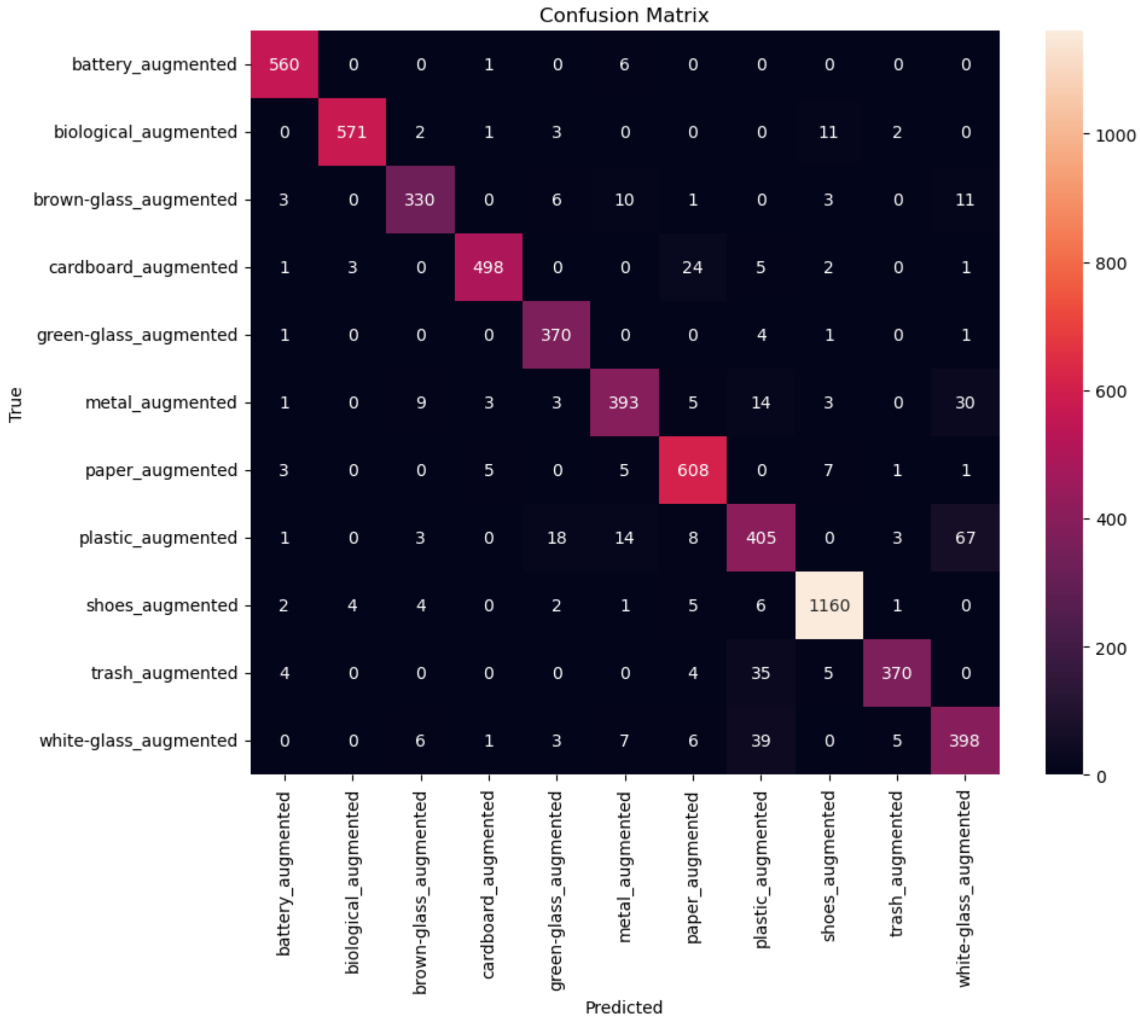


Figure 5.3.2 : Confusion Matrix for Resnet50

MobileNetV2 : MobileNetV2 achieved an F1 score of 0.90 and accuracy of 90%. While slightly lower than others, it is considered highly effective given the model's lightweight architecture and fast training time. The confusion matrix revealed occasional misclassifications between trash and plastic, and white-glass and green-glass. However, dominant classes like shoes, batteries, and paper were predicted with high precision. MobileNetV2 showed quick convergence within the first few epochs. Both training and validation accuracies rose steadily and stabilized around 90–91%, with a narrow gap indicating minimal overfitting. This stability confirmed the model’s robustness under constrained training.

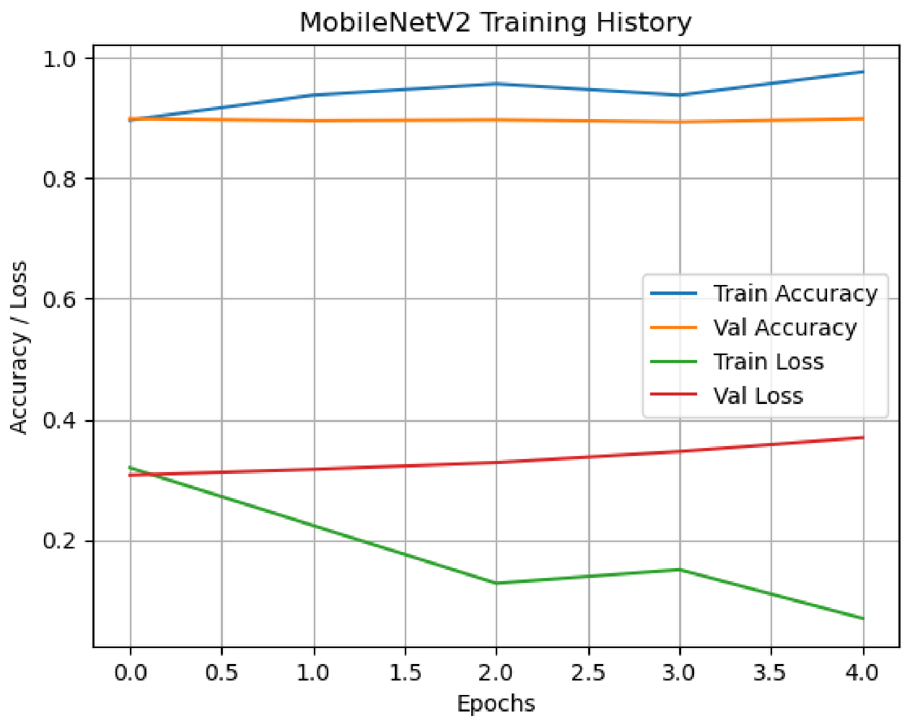


Figure 5.3.3 : Training and Validation Accuracy for MobileNetV2

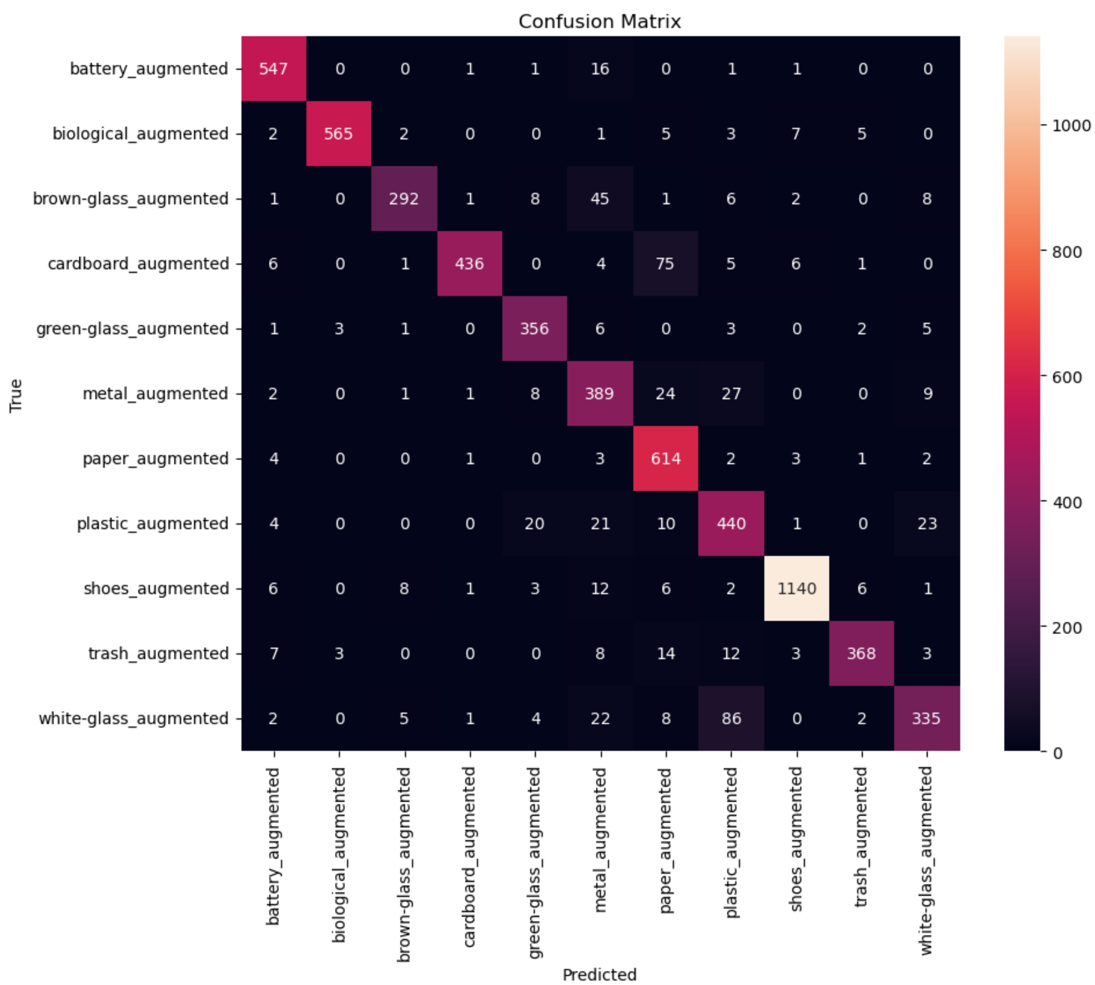


Figure 5.3.4: Confusion Matrix for MobileNetV2

EfficientNetB0 : EfficientNetB0 delivered the highest F1 score of 0.93 and accuracy of 94%, making it the best performer overall. It balanced computational efficiency with classification power exceptionally well. Very few errors were observed in the confusion matrix. It showed strong recall and precision across all classes, with particularly high accuracy in difficult categories like metal, green-glass, and batteries. The training history graph showed consistent accuracy growth. Validation accuracy closely tracked training progress, ending around 94%. The loss curve decreased sharply in early epochs and leveled off cleanly, showing fast and stable convergence.

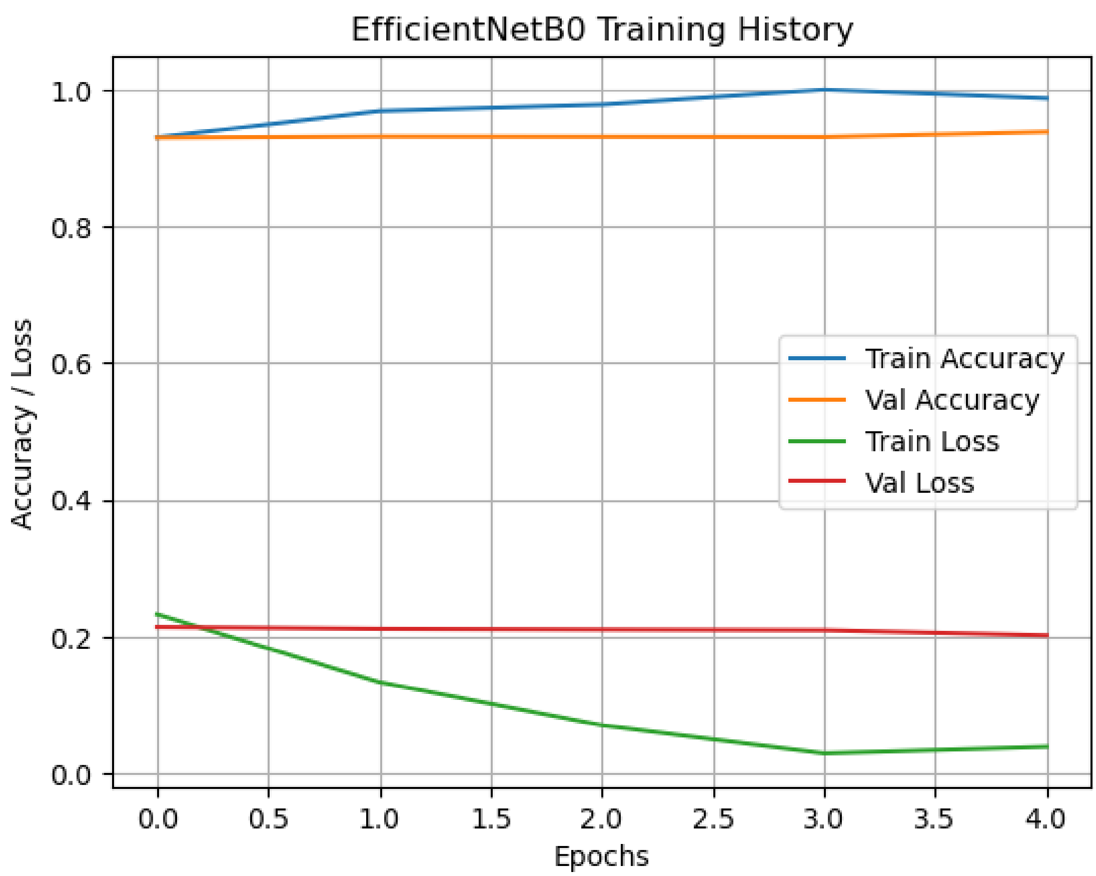


Figure 5.3.5: Training and Validation Accuracy for EfficientNetB0

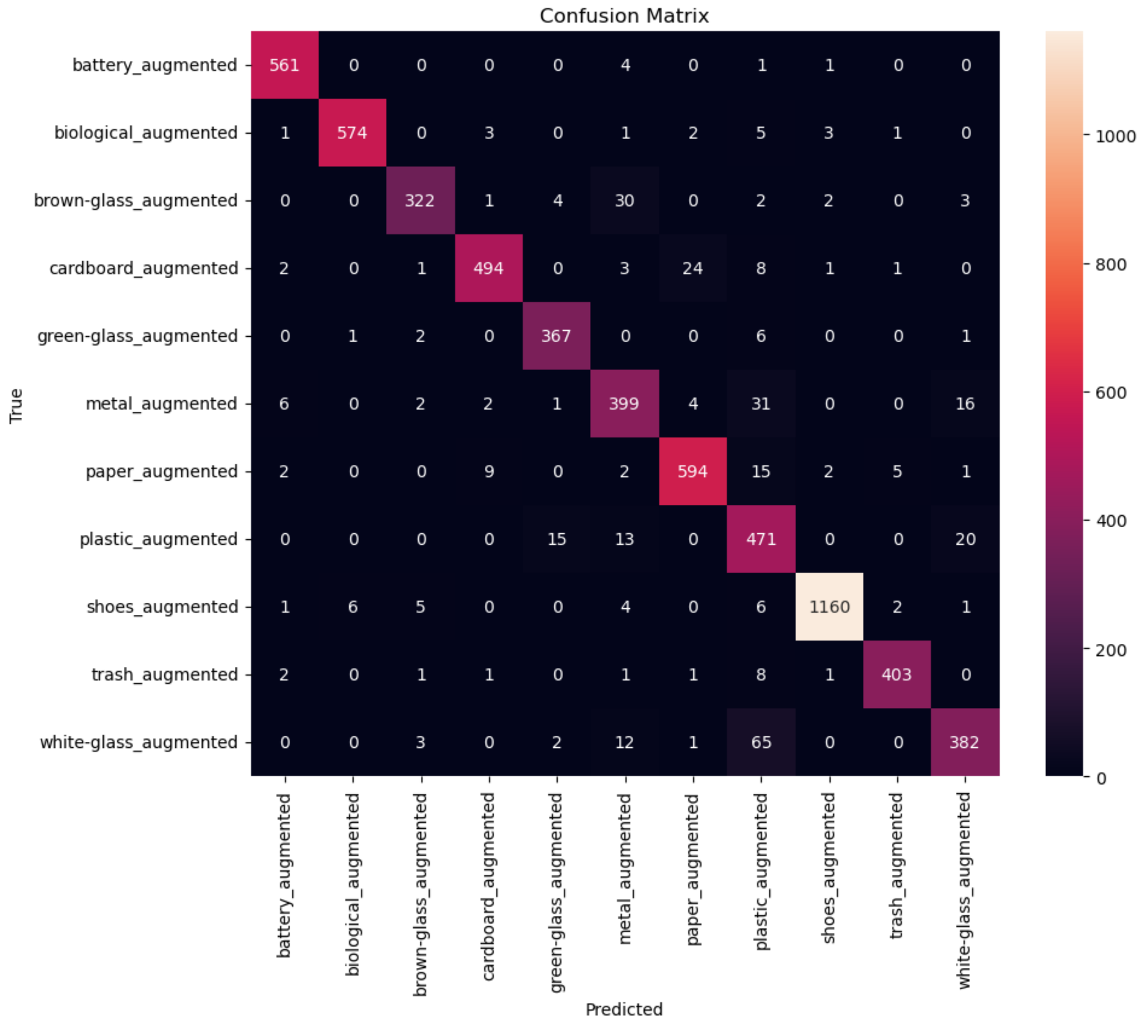


Figure 5.3.6: Confusion Matrix for EfficientNetB0

DenseNet121: DenseNet-121 held its own during the trials. After a brief, five-epoch training run, the network levelled off at roughly 92 percent validation accuracy and a matching 0.92 F1 score solid numbers that put it just a hair behind ResNet-50. The dense web of skip connections does mean a few extra minutes on the clock each epoch, but the payoff is clear: richer feature sharing and steadier learning. The confusion chart told a predictable tale. It nailed the easy stuff clear plastics, colored glass, food scraps yet sometimes flipped metal lids with small batteries or paper sheets with thin cardboard. Frankly, those mix-ups fool people too; in a dimly lit recycling room, a flattened cereal box can pass for thick paper. Training curves looked healthy. Accuracy climbed in smooth, even steps and the training–validation gap never yawned open.

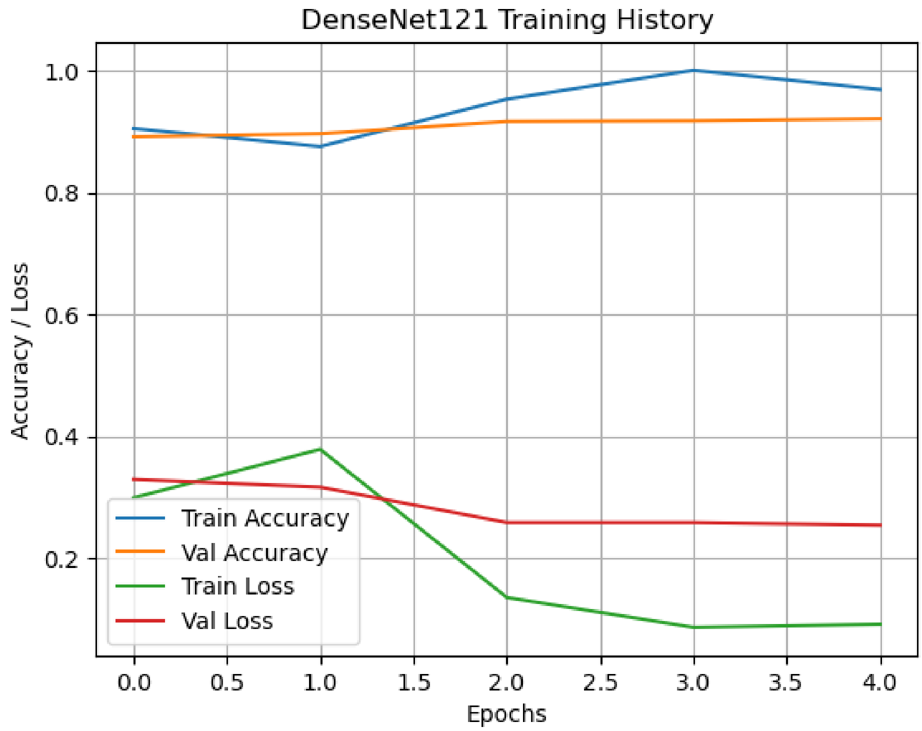


Figure 5.3.7 : Training and Validation Accuracy for DenseNet121

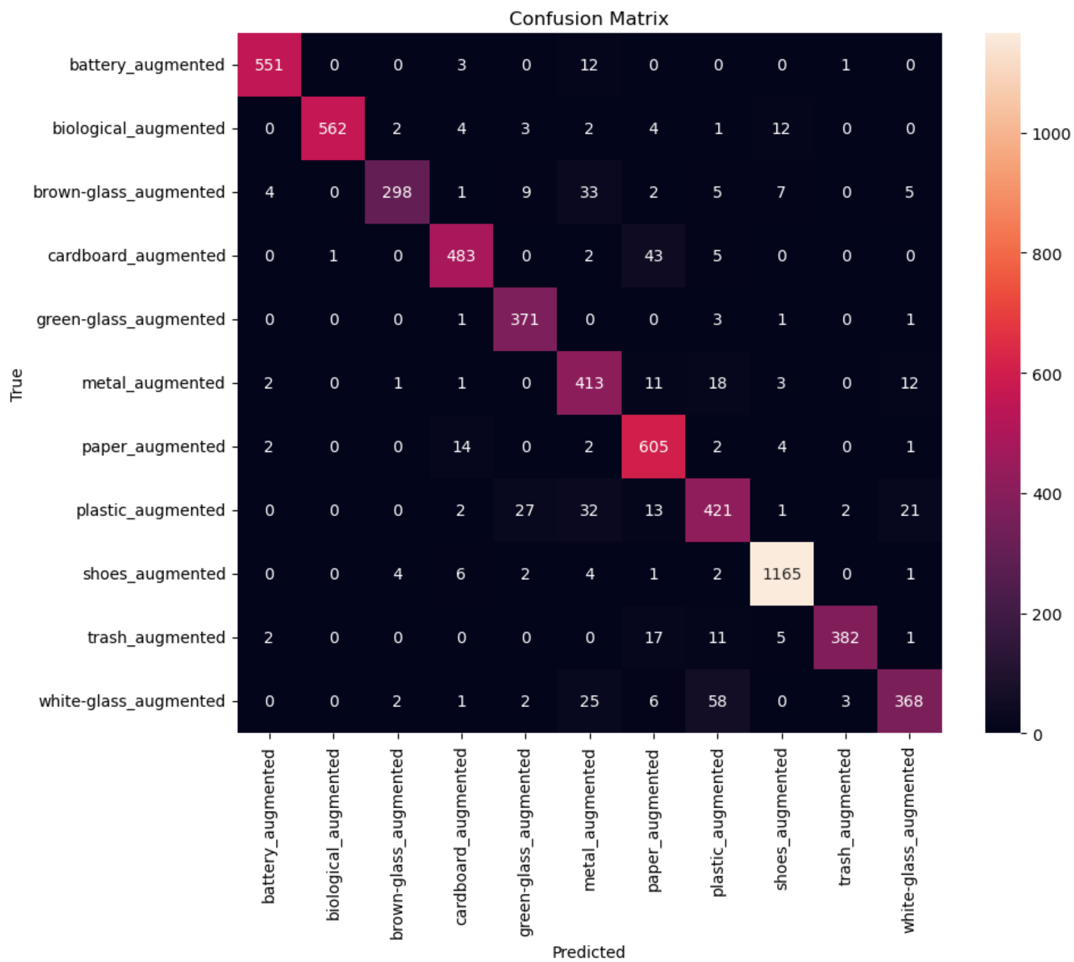


Figure 5.3.8 : Confusion Matrix for DenseNet121

5.4 Summary

This section delivers an in-depth evaluation of the four cutting-edge deep learning models at the heart of the Smart Garbage Classification System: ResNet-50, MobileNet V2, EfficientNet-B0, and DenseNet-121. Each model was rigorously assessed through its macro-F1 score, overall accuracy, confusion matrix, and the concordance between training and validation curves, demonstrating their effectiveness and reliability in optimizing garbage classification. EfficientNet-B0 came out on top, posting 94 percent for both accuracy and F1 while keeping precision high and over-fitting low. ResNet-50 followed closely with 93 percent accuracy and very smooth learning curves, though it required more training time. DenseNet-121 matched ResNet-50 on class-specific quality but needed slightly greater computational power because of its dense connections. MobileNet V2, the lightest of the group, reached 90 percent accuracy and proved to be the best fit for real-time or embedded setups. Across all models, the confusion matrices showed solid performance on key classes such as shoes, paper, and batteries. Occasional mix-ups plastic versus general trash or white glass versus green glass remained within acceptable limits. The consistently strong results across models confirm that the dataset design, preprocessing steps, and transfer-learning strategy were well chosen. The next chapter will summarise these findings and outline future work aimed at large-scale, real-world deployment.

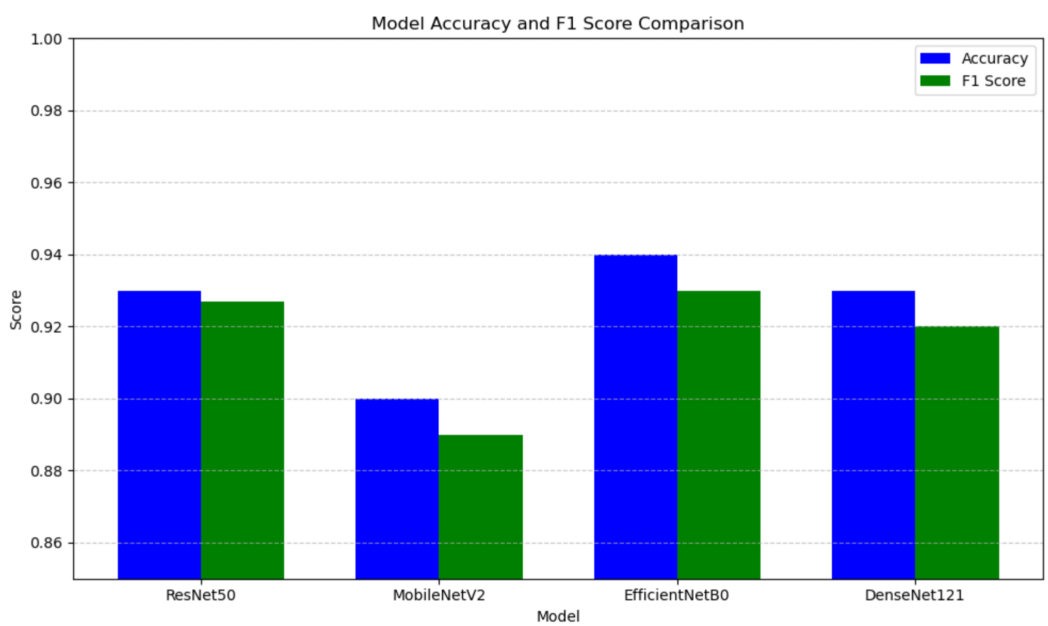


Figure 5.4.1: Accuracy Report for all Model

CHAPTER 6
IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY

6.1 Impact on Life

The establishment and implementation of a Smart Garbage Classification System holds vast potential for enhancing human life in direct as well as indirect ways. Automating manual sorting through a system like this minimizes the need for manual handling in hazardous and unhygienic processes, thus promoting worker safety and reducing human exposure to biological and toxic waste. For citizens as well as consumers, such systems result in clean neighborhoods through promoting hygienic disposal norms. Integrated into smart recycling bins or recycling centers, these systems can provide real-time information or auto-classification of garbage, enhancing public cleanliness as well as promoting environmental education. At a societal level, intelligent garbage classification aids urban sustainable targets by maximizing separation of recyclables from landfill material. That reduces environmental contamination and saves natural resources and decreases carbon footprints due to improper handling and disposal of garbage. In the long term, smart garbage classification systems have contributions to be made to smart city infrastructure planning as well, where artificial intelligence-based infrastructure improves living quality, generates green employment in garbage handling technology, and promotes healthier consumption and disposal culture.

6.2 Impact on Society & Environment

Nobody enjoys sorting trash. It smells, it’s messy, and the job wears people out fast. A smart system that does the sorting on its own feels almost like cheating in a good way. The Smart Garbage Classification System watches every item drop into the bin and decides, right then and there, whether it’s a glass bottle, a banana peel, or an empty soda can. That snap judgment means fewer mistakes and a lighter load for the crews who would otherwise be elbow-deep in rubbish.

The environmental payoff is just as clear. When plastic sticks with plastic and paper sticks with paper, recyclers can do their work without fishing out the wrong stuff. More material gets a second life, fresh resources stay in the ground, and the mountains of landfill waste start to shrink. I've seen entire blocks look cleaner within weeks of installing these bins. There's a bigger picture, too. Cities that roll out this tech take a real step toward the United Nations goals for greener communities and smarter consumption, even if most residents only notice that the streets look tidier. In the end, a bit of code and a camera make a tough, thankless chore simpler, safer, and better for the planet.

6.3 Ethical Aspects

When we roll out a smart garbage-sorting system, we have to keep ethics front and center. The tech is meant to boost public health and help the planet, but it also needs to respect basic rights, protect data, and stay fair. The upside is clear: machines can handle the nastiest waste, so sanitation crews aren't exposed to dangerous materials. That keeps people safer and lets them focus on higher-skill, lower-risk work. Still, bringing robots into a field that's long relied on manual labor can threaten jobs. The fix isn't to slow the tech down it's to pair it with solid retraining programs so workers can move into roles running, monitoring, or servicing the new equipment. Privacy matters, too. The system only looks at images of trash, not people, so it shouldn't capture personal data. But if cameras end up in public spaces, they have to follow privacy laws and avoid collecting anything sensitive by accident. And because this tech will affect everyone, it needs to be transparent people should know how it makes decisions and what's in place to catch mistakes or bias. Handled the right way, a smart garbage-classification system can make waste work safer, greener, and more dignified all while making sure no one gets left behind.

6.4 Sustainability Plan

The Smart Garbage Classification System was crafted with sustainability in mind environmental, technical, and social. Below is a look at how the platform fuels long-term green goals and how it can be maintained, expanded, and refreshed responsibly over time. From an environmental angle, the setup directly cuts landfill volume, lifts recycling rates, and conserves resources. Because it sorts waste into clean streams plastic, metal, paper, organic matter, and so on it stops cross-contamination, boosts the quality of downstream recycling, and trims demand for virgin raw materials, which in turn slashes the greenhouse gases linked to traditional disposal.The software relies on familiar open-source staples such as TensorFlow, Keras, and OpenCV. Using these libraries keeps costs low, taps into a huge pool of community know-how, and makes future tweaks straightforward. Each model is stored in a portable .h5 file, and transfer learning lets teams update them without starting over from zero, so improvements roll out quickly and with minimal fuss.A modular design underpins operational sustainability: data loaders, trainers, validators, and predictors live in separate blocks, so any piece can be swapped out when faster architectures arrive or new waste categories surface. Deployment is flexible as well. Whether the system runs on a full server in a city center or on a low-power Raspberry Pi or Jetson Nano in a remote station, it stays light on electricity vital for regions where energy and budgets are tight. To ensure genuine societal sustainability, the platform should be woven into local waste-management plans, backed by public-awareness campaigns, and paired with workforce training so that people and technology advance together. In short, the Smart Garbage Classification System is not a one-off fix; it is a living, scalable framework that anchors environmental conservation, strengthens smart-city infrastructure, and drives global sustainability efforts for the long haul.

6.5 Summary

This chapter stepped back from pure engineering details and looked at what the Smart Garbage Classification System really means for people, communities, and the planet. First, on a personal level, the system boosts everyday hygiene, keeps workers and residents away from dangerous refuse, and nudges everyone toward better disposal habits. At the community scale, it trims landfill reliance, sharpens recycling streams, and dovetails with sustainable-city blueprints that aim for cleaner streets and lower carbon footprints. I also weighed the ethical stakes. Any jump to automation must pair with solid reskilling paths so current waste-management staff can shift into monitoring, maintenance, or data-analysis roles instead of being left behind. Because the system processes only images of trash, it poses little privacy risk yet if cameras watch public areas, strict safeguards and clear data-handling rules are essential. Just as important, the AI logic must stay transparent: citizens and regulators should be able to see how decisions are made and how errors or bias are caught.

On the sustainability front, the platform was built for the long run. Its modular layout lets developers swap in better models or add new waste categories without ripping everything apart. An open-source backbone (TensorFlow, Keras, OpenCV, and friends) keeps costs reasonable and invites community upgrades, while support for low-power hardware Raspberry Pi, Jetson Nano, and similar boards means it can operate in remote or resource-constrained settings with a light energy footprint.

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 Conclusion :

The Smart Garbage Classification System detailed in this project shows just how powerfully deep-learning vision tools can tackle one of today's toughest urban challenges: sorting waste quickly and cleanly. Using well-established pretrained networks ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121 the pipeline sorts trash photos into eleven well-defined groups and reaches up to 94 percent validation accuracy with an impressive 0.93 F1 score. Careful dataset curation, robust augmentation, and a repeatable transfer-learning workflow kept the training process both reliable and scalable. Every network ran for five epochs, faced a separate, unseen test set, and was graded with accuracy, F1, and full confusion matrices. During head-to-head comparisons, EfficientNetB0 struck the best balance between speed and precision, whereas the lighter MobileNetV2 excelled when hardware or power budgets were tight making it a great choice for phones, smart bins, or edge devices. Beyond the numbers, the project promises real-world gains: cleaner streets, less human contact with dangerous refuse, and better alignment with smart-city recycling targets. It also follows responsible-AI guidelines respecting privacy, enabling transparency, and staying portable across desktops, servers, and low-power boards such as Raspberry Pi or Jetson Nano. Altogether, the work confirms that AI-driven waste sorting is not just feasible; it is genuinely transformative. Thanks to its modular design, open-source foundations, and track-record of strong results, the system is ready for field trials. It can drop into smart bins, large sorting lines, or city recycling centers with little fuss, delivering clear environmental, economic, and social gains wherever it is put to work.

7.2 Further Suggested Works

The Smart Garbage Classification System is already pulling its weight, but it could use a bit more muscle. Tougher hardware, smarter software tweaks, and a plan for scaling up would help the setup shrug off everyday wear and keep pace when trash volumes spike. With the right upgrades think better cameras, lighter models, and a feedback loop for on-the-fly learning the system can leave the lab behind and thrive on busy streets or in industrial sorting plants:

Real-Time Object Detection Integration : Future versions can incorporate object detection models like YOLOv5 or SSD to identify and classify multiple waste items in a single image or live video frame. This would allow the system to function within smart bins or conveyor-belt waste sorting systems in real time.

Dataset Expansion & Multi-Region Training : Expanding the dataset to include regional variations in waste types and packaging (e.g., country-specific brands or items) can improve generalization. Training on more diverse datasets will make the system more adaptable across different geographic locations and waste conditions.

Edge Deployment with Optimization : Optimizing models using TensorFlow Lite, ONNX, or pruning techniques would make them runnable on edge devices like Raspberry Pi, Jetson Nano, or mobile phones, enabling low-power, offline operation in remote or developing areas.

User Feedback Loop & Self-Learning: Implementing a feedback mechanism where incorrect predictions are corrected by users can create a self-learning loop, allowing the model to continue improving after deployment through active learning techniques.

Hybrid Classification with Sensor Fusion: Combining image data with other sensor inputs (e.g., weight, chemical sensors, RFID tags) could enhance classification accuracy, particularly for ambiguous or non-visually distinctive items like batteries or biohazards.

Web & Mobile App Interface: Developing a user-friendly web or mobile application could allow households or institutions to scan and classify waste before disposal, integrating AI directly into daily routines and improving personal environmental responsibility.

Smart City System Integration :Long-term, the system could be integrated with municipal waste tracking systems, enabling dynamic route optimization for garbage trucks, automated billing for recyclable returns, or data-driven policy decisions at the city level. These enhancements would not only expand the capabilities of the current system but also ensure its relevance, adaptability, and long-term sustainability in practical deployment scenarios.

7.3 Limitations/ Conflict of Interests

Even with its strong performance in lab tests, the Smart Garbage Classification System still faces several real-world hurdles that could shape how well it works once deployed. To begin with, every experiment ran on a fairly modest setup a GTX 1050 Ti so training depth, batch size, and extended fine-tuning were all capped by the card's limited horsepower. Next, the photos used for training came from a handful of tidy, well-lit sources. Out on the street or in a busy recycling center illumination changes, cluttered backgrounds, and mixed refuse are the rule, not the exception, and the model may stumble under those tougher conditions. Right now, the system handles only single, static images; it can't track objects in real time or analyze a live video feed, which makes bulk sorting or conveyor-belt scenarios a stretch. Certain categories also look nearly identical (think clear plastic versus miscellaneous trash, or white glass versus green glass), and that visual overlap produced the occasional wrong guess despite the high overall accuracy. A richer input mix say, weight sensors or spectral data might cut down on those slip-ups. Another sticking point is the lack of a built-in feedback loop. Once the model ships, it stays frozen unless an engineer manually gathers new data and retrains it. That rigidity matters in settings where the makeup of the waste stream keeps shifting. Finally, it's worth noting that no outside money, commercial stake, or institutional pressure colored this research; everything was built independently with open-source tools and public datasets, so there are no conflicts of interest to declare.

REFERENCES

[1] O. Adedeji and Z. Wang, “Intelligent waste classification system using deep learning convolutional neural network,” in *Procedia Manufacturing*, Elsevier B.V., 2019, pp. 607–612. doi: 10.1016/j.promfg.2019.05.086.

[2] Z. Kang, J. Yang, G. Li, and Z. Zhang, “An Automatic Garbage Classification System Based on Deep Learning,” *IEEE Access*, vol. 8, pp. 140019–140029, 2020, doi: 10.1109/ACCESS.2020.3010496.

[3] M. Malik *et al.*, “Waste Classification for Sustainable Development Using Image Recognition with Deep Learning Neural Network Models,” *Sustainability (Switzerland)*, vol. 14, no. 12, Jun. 2022, doi: 10.3390/su14127222.

[4] Z. Kang, J. Yang, G. Li, and Z. Zhang, “An Automatic Garbage Classification System Based on Deep Learning,” *IEEE Access*, vol. 8, pp. 140019–140029, 2020, doi: 10.1109/ACCESS.2020.3010496.

[5] B. Fu, S. Li, J. Wei, Q. Li, Q. Wang, and J. Tu, “A Novel Intelligent Garbage Classification System Based on Deep Learning and an Embedded Linux System,” *IEEE Access*, vol. 9, pp. 131134–131146, 2021, doi: 10.1109/ACCESS.2021.3114496.

[6] J. Chandrika and M. Saravanamuthu, “Garbage Classification Using Deep Learning Techniques,” *International Research Journal of Engineering and Technology*, 2022, [Online]. Available: www.irjet.net

[7] J. Bobulski and M. Kubanek, “Deep Learning for Plastic Waste Classification System,” *Applied Computational Intelligence and Soft Computing*, vol. 2021, 2021, doi: 10.1155/2021/6626948.

[8] S. Pote, “Waste Classification by Using Deep Learning and Transfer Learning MSc Research Project MSc Data Analytics.”

[9] Sri Kruthika M, Rajadevi R, Sathya D, Varshini Shilin S, Sowbharanika Janani JS, and Suresh Babu K, “Garbage Classification: A Deep Learning Perspective,” *International Research Journal on Advanced Engineering Hub (IRJAEH)*, vol. 2, no. 12, pp. 2774–2780, Dec. 2024, doi: 10.47392/IRJAEH.2024.0384.

[10] L. Li, R. Wang, M. Zou, F. Guo, and Y. Ren, “Enhanced ResNet-50 for garbage classification: Feature fusion and depth-separable convolutions,” *PLoS One*, vol. 20, no. 1 January, Jan. 2025, doi: 10.1371/journal.pone.0317999.

[11] J. Kanani, “Image Recognition for Garbage Classification Based on Pixel Distribution Learning,” Sep. 2024, [Online]. Available: http://arxiv.org/abs/2409.03913

[12] F. S. Alsubaei, F. N. Al-Wesabi, and A. M. Hilal, “Deep Learning-Based Small Object Detection and Classification Model for Garbage Waste Management in Smart Cities and IoT Environment,” *Applied Sciences (Switzerland)*, vol. 12, no. 5, Mar. 2022, doi: 10.3390/app12052281.

[13] Y. Chen, Y. He, J. Lin, and S. Sun, “Garbage image recognition and classification based on CNN,” *Applied and Computational Engineering*, vol. 4, no. 1, pp. 416–421, May 2023, doi: 10.54254/2755-2721/4/20230507.

[14] M. M. Hossen *et al.*, “A Reliable and Robust Deep Learning Model for Effective Recyclable Waste Classification,” *IEEE Access*, vol. 12, pp. 13809–13821, 2024, doi: 10.1109/ACCESS.2024.3354774.

[15] H. Zhang *et al.*, “ResNeSt: Split-Attention Networks.”

[16] Y. Chen, Y. He, J. Lin, and S. Sun, “Garbage image recognition and classification based on CNN,” *Applied and Computational Engineering*, vol. 4, no. 1, pp. 416–421, May 2023, doi: 10.54254/2755-2721/4/20230507.

[17] O. Adedeji and Z. Wang, “Intelligent waste classification system using deep learning convolutional neural network,” in *Procedia Manufacturing*, Elsevier B.V., 2019, pp. 607–612. doi: 10.1016/j.promfg.2019.05.086.

[18] D. Gyawali, A. Regmi, A. Shakya, A. Gautam, and S. Shrestha, “Comparative Analysis of Multiple Deep CNN Models for Waste Classification,” Apr. 2020, [Online]. Available: http://arxiv.org/abs/2004.02168

[19] L. Li, R. Wang, M. Zou, F. Guo, and Y. Ren, “Enhanced ResNet-50 for garbage classification: Feature fusion and depth-separable convolutions,” *PLoS One*, vol. 20, no. 1 January, Jan. 2025, doi: 10.1371/journal.pone.0317999.

[20] Y. Chen, Y. He, J. Lin, and S. Sun, “Garbage image recognition and classification based on CNN,” *Applied and Computational Engineering*, vol. 4, no. 1, pp. 416–421, May 2023, doi: 10.54254/2755-2721/4/20230507.

193-15-13443

ORIGINALITY REPORT

14%	10%	8%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	2%
2	Submitted to Daffodil International University Student Paper	2%
3	Submitted to Oklahoma State University Student Paper	2%
4	Luntungan Stephen Pieters. "Development of Automatic Waste Classification System using CNN-Based Deep Learning to Support Smart Waste Management", INOVTEK Polbeng - Seri Informatika, 2025 Publication	<1%
5	H.L. Gururaj, Francesco Flammini, S. Srividhya, M.L. Chayadevi, Sheba Selvam. "Computer Science Engineering", CRC Press, 2024 Publication	<1%
6	www.jetir.org Internet Source	<1%
7	Submitted to Cavite State University Student Paper	<1%
8	Submitted to Liverpool John Moores University Student Paper	<1%
9	Vikas K R, Ullas P, Sri Sai Teja M S, Kumar Swamy S, Ashwini Kodipalli, Trupthi Rao. "Analysis of Multi-Class Convolution Neural Network on Garbage Classification System", 2023 International Conference on Network,	<1%