*Nigerian Journal of Technology (NIJOTECH)*
*Vol. 44, No. 2, June, 2025, pp.357 - 366*
*www.nijotech.com*

*Print ISSN: 0331-8443*
*Electronic ISSN: 2467-8821*
*https://doi.org/10.4314/njt.v44i2.18*

# EVALUATING DEEP LEARNING MODELS FOR REAL-TIME WASTE CLASSIFICATION IN SMART IOT ENVIRONMENT

**AUTHORS:**
E. Omonayin[1,*], O. N. Akande[2], A. Muhammad[3], and S. Enemuo[4]

**AFFILIATIONS:**
[1,3]Computer Science Department, Baze University, Abuja, Nigeria
[2]Computer Science Department, Nile University of Nigeria, Abuja, Nigeria
[4]Computer Science Department, Landmark University, Kwara State, Nigeria

***CORRESPONDING AUTHOR:**
Email: esther.omonayin@bazeuniversity.edu.ng

**ARTICLE HISTORY:**
**Received:** 02 October, 2024.
**Revised:** 24 March, 2025.
**Accepted:** 25 April, 2025.
**Published:** 07 July, 2025.

**KEYWORDS:**
Waste Classification, IOT, Convolutional Neural Networks (CNNs), Deep Learning Models, MobileNet Architecture, InceptionV3 Architecture, VGG16 Architecture, Transfer Learning, Image Preprocessing.

**ARTICLE INCLUDES:**
Peer review

**DATA AVAILABILITY:**
On request from author(s)

**EDITORS:**
Chidozie Charles Nnaji

**FUNDING:**
None

## Abstract

*Automatic municipal solid waste management systems are integral to every smart city worldwide. They help to separate wastes into different categories for further recycling or effective disposal. This way, waste authorities could mitigate the effect of rapid urbanization, population growth, and the escalating consumption patterns associated with modern living. Deep learning models could play a critical role in the identification and classification of these wastes into their respective categories. Therefore, this study evaluates the performance of three deep learning models: MobileNet, InceptionV3, and VGG16 for waste classification. The evaluation was done under two separate model configurations while their classification accuracy, execution time, precision, recall, and F1-Score were computed across a range of 10 to 100 epochs. MobileNet consistently demonstrated the highest classification accuracy, reaching approximately 90% at 100 epochs, while also maintaining the shortest execution time, starting at 2.13 minutes for 10 epochs and increasing to about 14.34 minutes for 100 epochs. InceptionV3 exhibited a balanced performance, achieving around 83% accuracy at 100 epochs with execution times ranging from 3.57 minutes to 49.42 minutes. VGG16, although started with the lowest accuracy, improved significantly to about 88% at 100 epochs, but at the cost of the longest execution time, starting at 9.45 minutes and rising to 68.72 minutes. The results indicate that MobileNet is the most efficient model for applications requiring both high accuracy and low computational cost, while InceptionV3 and VGG16 are suitable for scenarios where accuracy is prioritized over execution time. This comparative analysis provides valuable insights for selecting appropriate deep-learning models based on specific task requirements and resource constraints.*

***HOW TO CITE:***
Omonayin, E., Akande, O. N., Muhammad, A., and Enemuo, S. "Evaluating Deep Learning Models for Real-Time Waste Classification in Smart IOT Environment", *Nigerian Journal of Technology*, 2025; 44(2), pp. 357 – 366; https://doi.org/10.4314/njt.v44i2.18

## 1.0 INTRODUCTION

An increase in urbanization and population growth has resulted in the massive generation of Municipal Solid Waste (MSW) across developed and developing countries of the world [1, 2]. This has continuously posed a challenge of how these wastes could be properly managed to maintain urban hygiene, improve the quality of life of citizens and protect environmental resources. Municipal solid wastes could be defined as non-biodegradable waste collected from institutions, construction sites, residential and industrial areas. These are majorly household waste (Electronics, textiles, plastics, glass, metals etc), commercial waste (Packaging Materials, Office Supplies, Plastics and Metals), institutional waste (Laboratory Waste, Cafeteria Waste, and

Educational Materials), public space waste (Garden and park waste, litters and debris gathered during street cleaning), construction and demolition waste (Concrete and Masonry, wood, metals and plastics).

The World Bank has estimated that 2.01 billion tonnes of these wastes are generated in cities annually; this is expected to skyrocket to 3.40 billion tonnes by 2050 [3]. While this massive generation of waste could be attributed to urban expansion and lifestyle changes of citizens, inappropriate handling of this waste could result in unembellished environmental and health risks which are not limited to air and water pollution, spread of diseases, and greenhouse gas emissions [4]. Open dumps waste collection approaches which are common practice in some cities could also result in the release of methane to the environment this could also contribute to soil contamination and water pollution when washed by floods during the rainy season. Therefore, robust Municipal Solid Waste Management Systems (MSWMS) that leverages cutting-edge technologies like Artificial Intelligence (AI) and the Internet of Things (IoT) offer promising solutions for sustainable and hygienic urban development [6]. They must incorporate an efficient waste collection approach, a well-organized transportation system, effective processing and disposal methods, productive recycling, and waste-to-energy technologies. Implementing a comprehensive MSWMS will not only alleviate environmental and health hazards but also encourage resource recovery, reduce the burden on landfills, and support economic development through job creation and the generation of renewable energy.

Sorting and classification of wastes into different categories are the first steps towards effective waste management and Artificial Intelligence (AI) techniques can play a transformative role in achieving these tasks. Existing automated MSWMS can be integrated with AI techniques to analyze and classify waste in real-time thereby guaranteeing efficient separation of recyclable waste materials from non-recyclable ones. Most importantly, the integration of AI in waste management is one of the ways to achieve an inclusive and sustainable urbanization which is the third target of SDG 11[7]. When a smart MSWMS is integrated into other smart city technologies, it could create the needed cohesive, data-driven approach to urban sustainability. For instance, a smart bin with AI driven sensors can be used to optimize waste collection routes which could in turn reduce fuel consumption and operational costs incurred during waste management services [8]. Furthermore, a processed waste can be transformed into useful materials such as Cement [9], biopolymer reinforcement [10], building reinforcements [11, 12], and biogas production [13] among others. Several deep learning techniques could be employed for waste classification and analysis.

## 2.0 RELATED WORKS

Research on waste material classification requires the utilization of image processing techniques to extract geometric features like length, perimeter, shape, corner, ridge, and edge. These extracted features are then used to train deep learning algorithms for classification purposes.

Motivated by the need for efficient waste management practices, authors in [15] introduced the DenseNet201 model and compared it with other deep learning architectures like ResNet, MobileNetV2, AlexNet, and GoogleNet. The study leveraged the dense connectivity and feature aggregation capabilities of DenseNet201, allowing the model to aggregate features from multiple network depths and capture intricate patterns and dependencies across different spatial scales. This hierarchical feature learning is useful for accurate waste material classification with diverse datasets, demonstrating that DenseNet201 achieves superior accuracy, average recall, and average precision. In another study, authors in [16] identified major issues in current solid waste management services in India, such as the unavailability of web portals for citizens, lack of real-time monitoring of bins and collection vehicles, and the prevalent problem of illegal dumping in which a Multipath Convolutional Neural Network (MP-CNN) to detect and localize waste dumps on streets and roadsides was proposed.

A dataset was developed upon which weakly supervised learning was applied to train the MP-CNN model, achieving performance evaluation metrics such as 97.82% precision, 98.86% recall, 98.34% F1 score, 98.33% accuracy, and 98.63% AUROC for binary waste classification. Similarly, a study that analyzed various deep learning models for medical waste classification was carried out by authors in [17]. They employed EfficientNet as a classification model for single-target waste recognition. The research involved determining the constants that allocate resources to network depth, width, and resolution through a small grid search. A user-specified coefficient, Ph, controlled the model's increase in line with the increase in resources. This process helped in optimizing the model's performance. In addition, a comparison was made between the image recognition accuracy and a number of parameters of the

https://doi.org/10.4314/njt.v44i2.18

EfficientNet network from B0 to B7 with other models. The results showed that EfficientNet outperformed other models in terms of recognition accuracy and data processing efficiency. This led to a significant increase in recognition speed and accuracy while reducing the amount of data processed. The model also defined a composite coefficient, Ph, to balance the width, depth, and resolution of the network, further enhancing its performance.

Similarly, authors in [18] as a solution, this research develops an image-based litter classification model using Deep Learning DenseNet architecture. The model is designed to address the need for automated waste sorting by classifying waste into ten different categories, using diverse training datasets. The results of this study showed that the model achieved an overall accuracy rate of 93%, with an excellent ability to identify and classify specific materials such as batteries, biological materials, and brown glass. Despite some challenges in metal and plastic classification, these results confirm the great potential of using Deep Learning technology in waste management systems to improve sorting processes and increase recycling efficiency. Authors in [19] proposed an IoT-based waste management system that emphasized on waste classification and optimal waste routing. Conventional deep maxout and Bidirectional-Long Short-Term Memory (Bi-LSTM) networks were employed for waste classification while the Snake Optimization Updated Beluga Whale Optimization algorithm (SOUBWO) was employed for waste routing. The proposed techniques recorded appreciable classification accuracy and minimal energy consumption. Also, authors in [20] proposed a transfer learning-based waste classification technique that employed EfficientNet-CNN architecture for feature extraction and PCA for dimensionality reduction. The proposed technique achieved a remarkable accuracy of 99.07%.

Authors in [21] introduced an automated solid waste sorting machine capable of independently distinguishing between metals and non-metals. The study employed Arduino Mega microcontroller to drive the functionality of the developed system. Additionally, the system used sensors to identify and categorize objects while a smart digital counter was used to monitor metallic items. The primary goal of the study was to enhance the efficiency of waste recycling.

Authors in [22] also proposed a dual ensemble deep learning framework for waste management. The first ensemble layer combines multiple image-segmentation methods, while the second ensemble merges the outputs from the CNN architectures used. The proposed ensemble method outperformed deep learning techniques like VGG19, YoloV5, and InceptionV3.

## 3.0 METHODOLOGY

This section discusses the experimental design behind the study. This includes the architectures of the deep learning models studied, the datasets employed for the evaluation and the performance evaluation metrics used. Figure 1 provides an overview of the waste classification process employed in the study.

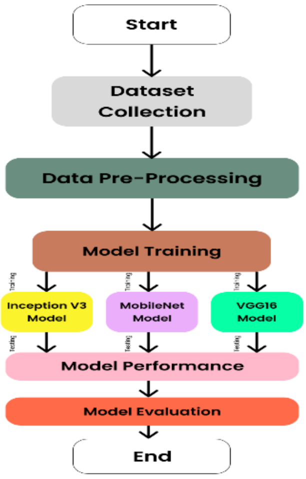


**Figure 1:** Flowchart of the waste classification model analysis

### 3.1 Data Collection

Due to relevance, quality, and credibility, the dataset employed in this study was retrieved from [23]. The dataset contains 2,751 images that are grouped into seven categories: cardboard, compost, glass, metal, paper, plastic, and trash. The dataset was further divided into training and test data. A total number of 2,466 images were used to train the deep learning models while the remaining 285 images were used for testing the models. An overview of the training and test datasets is provided in Tables 1 and 2.

**Table 1:** Categories of the training datasets

| **Classes** | **Quantity** |
|---|---|
| Cardboard | 363 |
| Compost | 139 |
| Glass | 461 |
| Metal | 370 |
| Paper | 554 |
| Plastic | 438 |
| Trash | 141 |
| **Total** | 2466 |

**Table 2:** Categories of the test datasets

| **Classes** | **Quantity** |
|---|---|
| Cardboard | 40 |

| | |
|---|---|
| Compost | 38 |
| Glass | 40 |
| Metal | 40 |
| Paper | 40 |
| Plastic | 44 |
| Trash | 43 |
| **Total** | 285 |

### 3.2 Data Preprocessing and Augmentation

To ensure an accurate, reliable, and efficient training and testing result, the dataset used for the training and testing was pre-processed and augmented. The preprocessing activities were done to improve the quality of the images and make them suitable for training. Pre-processing techniques adopted involve normalization, image cropping, noise reduction and colour space conversion from RGB to grayscale. In addition to data pre-processing, several data augmentation techniques were also carried out to simulate what could happen in real life when the waste images were captured. This in turn increased the diversity and size of the training dataset. Due to the peculiarity of the waste image classification task, the following pre-processing techniques were employed:

a) Image Rotation: since waste images are not expected to be in their original shape, some of the images were randomly rotated up to 20 degrees in either direction. This is expected to help the model to learn how to identify objects regardless of their orientation.
b) Image Shifting: the images were also randomly shifted horizontally and vertically. This was done to accommodate possible variations that could occur when the waste images are being captured.
c) Shear Transformation: A little shear transformation was also used to distort the image along its horizontal or vertical axis. This technique helps to simulate various angles and orientations from which real-life waste images could be captured. This will help the model recognize the images under different distortions and perspectives.
d) Randomized Zooms: the images were also randomly zoomed in and out. This will create different instances of the modified images without creating an entirely new dataset. Image randomization will help the model to recognize different objects at different camera zoom levels, scales, distances and resolutions, thereby, being able to handle instances where the images are closer or farther away.
e) Nearest Neighbour Interpolation: this is done by assigning the value of the nearest pixel to the new pixel location. One advantage of this interpolation is that the pixel swapping does not tamper with the edges and boundaries of the images. The original pixel values are preserved.

### 3.3 Model Training

Three variants of the Convolutional Neural Network deep learning technique were used for solid waste image classification in this study. They are VGG network, MobileNet V1, and Inception V3. Though all three models use Rectified Linear Unit (ReLU) activations and pooling layers and also employ hierarchical approach during feature extraction, they differ in terms of overall architectural designs, computational efficiency, parameter count and model size.

**VGG 16:** This variant of CNN was named after its developer, the Visual Geometry Group at Oxford University [24]. It uses a total of 16 weight layers which includes 13 convolutional layers and 3 fully connected layers. To preserve the spatial resolution of objects, its convolutional layer uses small 3 by 3 filters with a small receptive field. As shown in Figure 2, the small filters are 64, 128, 256, and 512 filters that introduce more depth into the layer and help VGG 16 to reduce the complexity of the model thereby making it easier to train. Its max-pooling layer has a 2 by 2 filter and a stride of 2 that are used to reduce spatial dimensions while retaining important features. Also, its first two fully connected layers have 4096 neurons while the remaining fully connected layer has 1000 neurons which are used for classification.

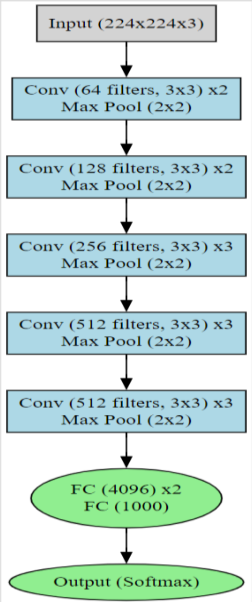


**Figure 2:** Architecture of VGG16 network

**MobileNet V1:** MobileNet is a resource constraint deep learning network introduced by Google [25]. As shown in Figure 3, it uses multiple depthwise separable convolutions for its classification task. This is divided into depthwise convolution and pointwise convolution layers. The depthwise convolution uses a single filter for each of its input channels while pointwise convolution employs a 1 by 1 convolution to merge the output of depthwise convolution. By doing this, the computational complexity of the model is reduced significantly. In addition, it also uses width and resolution multipliers to reduce the model's width and overall computational cost respectively.

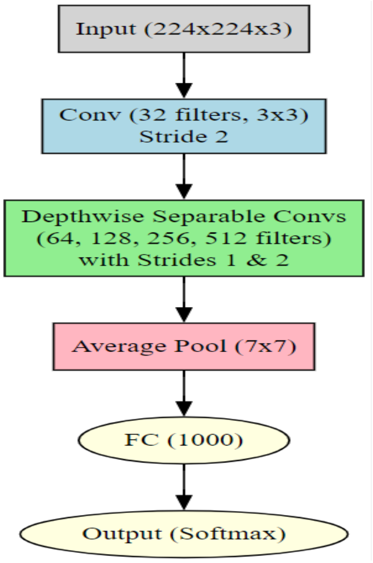


**Figure 3:** Architecture of MobileNet V1 model

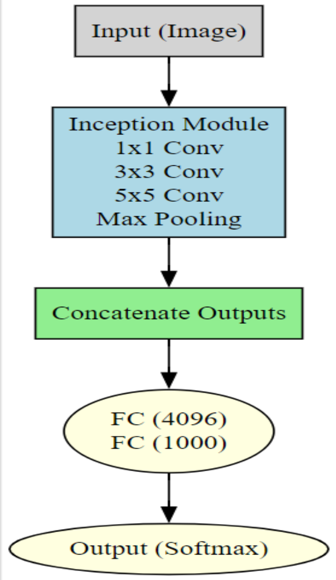


**Figure 4:** Architecture of Inception model

**Inception V3:** InceptionV3 is an improvement over the earlier versions of inception models introduced in [26]. As shown in Figure 4, it uses three convolutions of various sizes (1 by 1, 3 by 3 and 5 by 5) to capture visual information in images at multiple scales. Also, to reduce computational complexity while retaining its receptive field, its 7 by 7 convolutions are factorized into two 3 by 3 convolutions. It also employs auxiliary classifiers to improve its gradient flow and overall performance while strided convolutions and pooling layers are used to reduce the grid size of images thereby reducing excessive computational costs. It also employs batch normalization throughout the network to hasten training and enhance stability.

### 3.4 Experimental Setup

Two different configurations of the models were employed for the detailed analysis of their performance: Simple and Complex configurations. The features of the configurations are provided in Tables 3 and 4.

**Table 3:** Features of the simple model configuration

| | |
|---|---|
| Layer Number | 1 |
| Neurons In Layer | 1024 |
| Layer Activation Function | ReLu |
| Classification Function | Softmax |
| Optimizer | Adam (Learning Rate: 0.0001) |
| Layer Number | 1 |

**Table 4:** Features of the Complex Model Configuration

| | |
|---|---|
| Number of Layers | 2 |
| Neurons In Layer 1/2 | 1024/512 |
| Layer Activation Function | ReLu |
| Layer Dropout value | 0.5 |
| Classification Function | Softmax |
| Optimizer | Adam (Learning Rate: 0.0001) |
| Loss Function | Categorical Cross-Entropy |

Each model was also implemented between 10 and 100 epochs so as to capture the correlation between the different epochs. The following software libraries and frameworks were also used to implement the deep learning models:

(a) Programming Language: Python programming language was used for the model training, testing, and evaluation.
(b) TensorFlow: this is an open-source deep learning framework made available by Google. It has numerous comprehensive sets of tools that could be used to build and train neural networks.
(c) Keras API: this is a high-level neural network Application Programming Interface that could be used to build and train deep learning models. It has a user-friendly interface that is built on top of TensorFlow.

### 3.5 Model Performance Evaluation

The following performance evaluation metrics were used to examine the deep learning models being considered:

(a) Accuracy: This evaluates proportion of correct classifications made by each model. It measures the number of correctly classified waste images against the total number of images. It is computed thus:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (1)$$

(b) Precision: This computes the positive predictive value of the model. This is measured by computing the proportion of waste images predicted as a specific class that actually belongs to that class. A high precision value connotes a low false positive rate. It is computed thus:

$$Precision = \frac{TP}{TP+FP} \qquad (2)$$

(c) Recall: This is called sensitivity. It measures the model's ability to correctly identify the waste images correctly. It's measured as the proportion of actual positive cases that the model correctly predicts. A high recall rate means that a low false positive rate is recorded. It is computed thus:

$$Recall = \frac{TP}{TP+FN} \qquad (3)$$

(d) F1-Score: This is used to evaluate the harmonic mean between precision and recall so as to have a balanced view of the model's performance. It measures the model's ability to correctly identify true positives and avoid false positives. It is computed thus:

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \qquad (4)$$

Where True Positives (TP) are the number of waste images correctly classified as a specific waste category. True Negatives (TN) are the number of images correctly identified as not belonging to any of the waste categories. False Positives (FP) are the number of images incorrectly classified as belonging to a specific waste category when they are actually not a waste image. False Negatives (FN) are the number of images belonging to a specific waste categories that the model incorrectly classified as not belonging to any of the waste category.

## 4.0 RESULTS AND DISCUSSION

A detailed analysis of the performance of InceptionV3, VGG16, and MobileNet models is provided in this section.

### 4.1 Model Comparative Analysis

This section discusses the comparative analysis of the three models under the simple and complex configuration arrangement. Under the simple configuration arrangement illustrated in Figure 5, MobileNet exhibited a significant improvement of 7% when its classification increased from 78% at 10 epochs to 85% at 50 epochs before experiencing a diminishing return in accuracy after 50 epochs. VGG16 model also exhibited a moderate improvement of 9% when its classification increased from 61% at 10 epochs to 70% at 50 epochs before experiencing a diminishing return in accuracy after 50 epochs. However, InceptionV3 model experienced a slight decline in accuracy from 75% to 74% at epochs 10 and 50 respectively. This could be attributed to overfitting the model experienced.

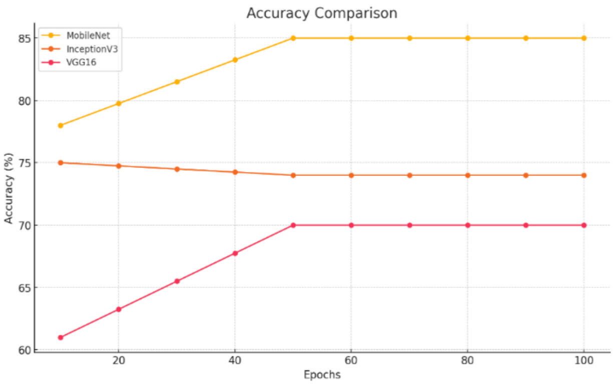


**Figure 5:** Classification accuracies of MobileNet, InceptionV3 and VGG16 Models

In terms of execution time, as displayed in Figure 6, MobileNet displayed the fastest execution time of 2.43 minutes at 10 epochs, this was increased to 10.9 minutes at 50 epochs. InceptionV3 also displayed a moderate execution time of 3.55 minutes at 10 epochs; this was significantly increased to 25.32 minutes at 50 epochs. However, VGG16 exhibited the longest execution time of 3.55 minutes at 10 epochs and 44.03 minutes at 50 epochs. Therefore, MobileNet exhibited the best overall performance in terms of classification accuracy and moderate execution time. In contrast, InceptionV3 and VGG16 are less efficient in terms of classification accuracy and execution time.

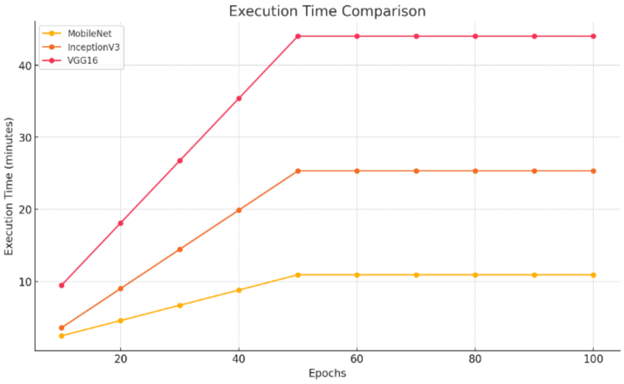


**Figure 6:** Execution time of MobileNet, InceptionV3 and VGG16 Models

Under the complex configuration shown in Figure 7, the MobileNet model still achieved the highest classification accuracy reaching 90% classification accuracy at 100 epochs. VGG16 recorded the lowest classification accuracy of 66% at 10 epochs but this significantly improved to 88% at 100 epochs.

InceptionV3 could only achieve a classification accuracy of 83% at 100 epochs; this was actually an improvement over a classification accuracy of 69% at 10 epochs.

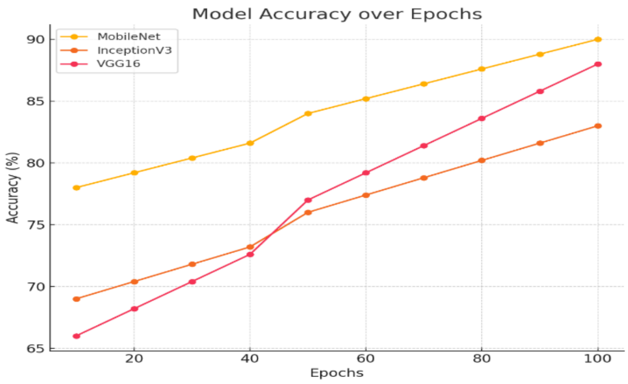


**Figure 7:** Classification accuracies of the models under complex configuration

In terms of execution time under the complex configuration shown in Figure 8, MobileNet achieved the fastest execution time of 2.13 minutes at 10 epochs and 14.34 minutes at 100 epochs. InceptionV3 achieved an execution time of 3.57 minutes at 10 epochs; this increased significantly to 49.42 minutes at 100 epochs. VGG16 exhibited the longest execution time of 9.45 minutes at 10 epochs and 68.72 minutes at 100 epochs.

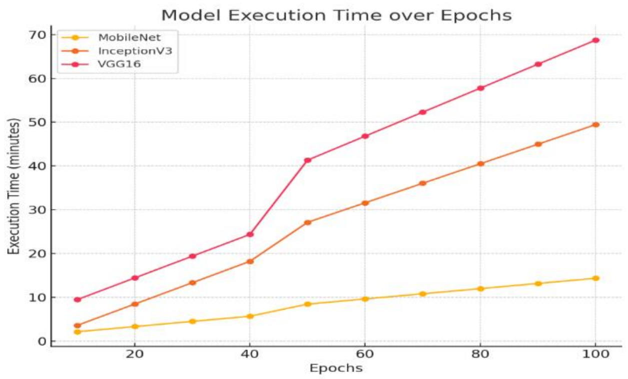
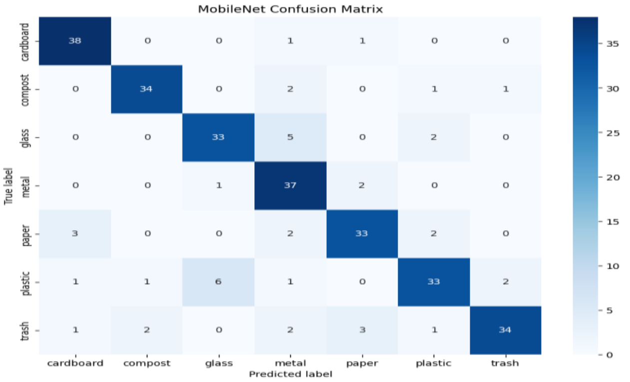


**Figure 8:** Execution time of the models under complex configuration

From the analysis carried out, MobileNet is the most efficient model in terms of classification accuracy, and execution time. Therefore, it is suitable for waste classification especially in instances where speed and performance are critical. A balance was achieved between classification accuracy and execution time when InceptionV3 was used. Therefore, it is preferable where moderate execution time and fair accuracy is expected. VGG16 was also able to achieve a high accuracy, however, it is at the expense of a high execution time. Furthermore, a comparison of the models under simple and complex configuration revealed that the models performed better in terms of classification accuracy under the complex architecture. All the models begin to flatten out at 50 epochs under the simple configuration but no flatten out was experienced under the complex architecture.

## 4.2 Confusion Matrix of the Models under the Complex Configuration

Since all the models performed better under the complex architecture, a comparative analysis of the models was carried out based on their confusion matrix. In addition to the performance of the models in terms of their overall classification accuracy, their confusion matrix provides an in-depth understanding of the category of waste that was rightly or wrongly classified. This was measured in terms of their true positive, true negative, false positive, and false negative rates. As shown in Figure 9, the number of correctly classified instances of waste categories are shown on the diagonals of the confusion matrix while the misclassified instances are shown outside the diagonals. Cardboard and paper wastes have the highest true positive rates with relatively low misclassification rates. Compost, metal only has three and four misclassification instances respectively. The model recorded the highest misclassification instances with glass, metal, and plastic wastes. Glass wastes were incorrectly classified as metal wastes in five instances while plastic wastes were incorrectly classified as glass wastes in six instances. Despite a high precision value, the highest misclassification was recorded with trash wastes as the model misses several trash instances. Therefore, the model needs to be fin-tuned to reduce these misclassification instances.



| | cardboard | compost | glass | metal | paper | plastic | trash |
|---|---|---|---|---|---|---|---|
| cardboard | 38 | 0 | 0 | 1 | 1 | 0 | 0 |
| compost | 0 | 34 | 0 | 2 | 0 | 1 | 1 |
| glass | 0 | 0 | 33 | 5 | 0 | 2 | 0 |
| metal | 0 | 0 | 1 | 37 | 2 | 0 | 0 |
| paper | 3 | 0 | 0 | 2 | 33 | 2 | 0 |
| plastic | 1 | 1 | 6 | 1 | 0 | 33 | 2 |
| trash | 1 | 2 | 0 | 2 | 3 | 1 | 34 |

**Figure 9:** Confusion matrix of MobileNet model

With VGG16 model, compost and paper wastes have the highest true positive values with the least misclassification rates of two and four respectively. A huge number of misclassifications were recorded with the other waste categories as shown in Figure 10.

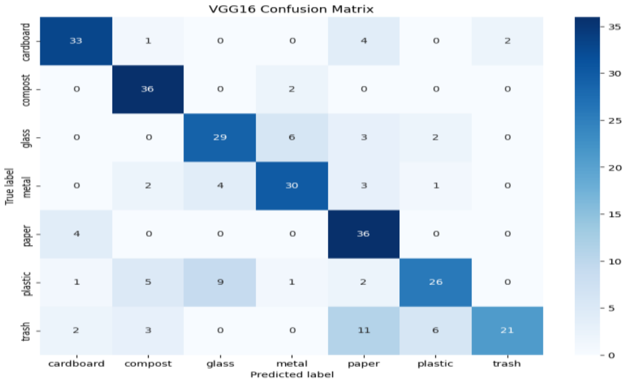


**Figure 10:** Confusion matrix of VGG16 model

As shown in Figure 11, InceptionV3 model recorded the highest true positive rates with cardboard, compost, and metal despite having few misclassification rates. However, the model had many misclassification rates in the remaining waste categories. This further affirms its poor overall performance among other models considered.

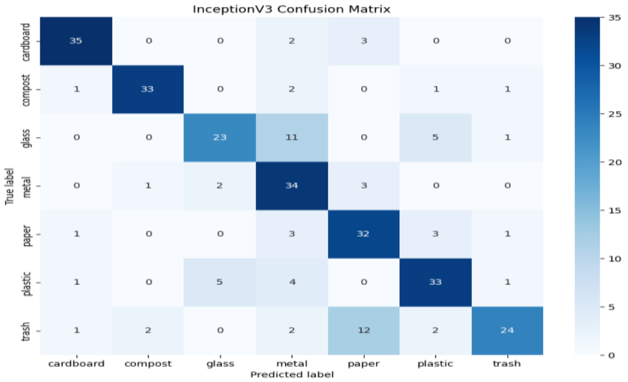


**Figure 11:** Confusion matrix of Inception V3 model

## 5.0 CONCLUSION

The suitability of three deep learning models — MobileNet, InceptionV3, and VGG16 for waste identification and classification have been explored in this study. Their performance in terms of classification accuracy and execution time under simple and complex configurations across epochs ranging from 10 to 100 has also been evaluated. In terms of classification accuracy, MobileNet exhibited the highest classification accuracy by achieving an accuracy of 90% at 100 epochs. InceptionV3 achieved also achieved an appreciable level of accuracy as it was able to attain 83% accuracy at 100 epochs. VGG16 recorded the lowest accuracy at 10 epochs but this significantly improved to 88% at 100 epochs. In terms of execution time, MobileNet recorded the shortest execution time of 2.13 minutes at 10 epochs and 14.34 minutes at 100 epochs. It is therefore the most efficient model in terms of computational resources and time. InceptionV3 consumed more time than MobileNet. It achieved an execution time of 3.57 minutes with 10 epochs; this increased significantly to 49.42 at 100 epochs. However, VGG16 recorded the longest time of 9.45 minutes at 10 epochs and 68.72 minutes at 100 epochs. Though VGG16 eventually achieved high accuracy, it did so at the expense of significantly higher execution time. Therefore, MobileNet is conspicuously the most efficient of all the three models considered as it has the highest classification accuracy with the shortest execution time. Therefore, it is recommended for smart IOT applications where speed and performance are desirable. In instances where moderate execution time is preferable to accuracy, InceptionV3 could be used. In instances where execution time is less critical, VGG16 can be used since it was able to also achieve a certain level of accuracy. However, the specific requirements of the task at hand will eventually determine the model to opt for. Future research can consider how to reduce the misclassifications recorded by all the models across various waste categories. Especially how transfer learning could be used to improve the model's ability to differentiate misclassified instances to achieve an improved overall performance.

## REFERENCES

[1] Ikebude, C. F. “Feasibility Study on Solid Waste Management in Port Harcourt Metropolis: Causes, Effect and Possible Solutions”, *Nigerian Journal of Technology*, 36 (1), 2016, 276-281. https://doi.org/10.4314/njt.v36il.33

[2] David, A. O., Odagbodo, O. O., Opafola, O. T., Amusan, G. M., Badejo, A. A., and Olaniyan, O. S. “Assessment of solid waste management in Ota, Ogun State, Nigeria”, *Nigerian Journal of Technology*, 42(2), 2023, 289-295. https://doi.org/10.4314/ njt.v42i2.18.

[3] Trends in Solid Waste Management available at https://datatopics.worldbank.org/what-a-waste/trends_in_solid_waste_management.html acessed on 24th July, 2024.

[4] Xiaomei, J., Yupeng, L., Zhi-Long, Y., and Weiqiang, C. “Influence of mandatory waste classification on environmental and economic impacts of residual waste treatment in Xiamen, China”, *Waste management and research, International Solid Wastes and Public Cleansing Association, ISWA*. 2014, https://doi.org/10.1177/0734242X241265055

[5] Alshaikh, R., and Abdelfatah, A. “Optimization Techniques in Municipal Solid Waste Management: A Systematic Review”,

*Sustainability* 2024, 16, 6585. https://doi.org/10.3390/su16156585

[6] Ogbonnaya, M., Ojolo, S. J., Oyefule, O., and Abudu, M. "Design and Fabrication of a Waste Plastic Filament Extruder." *Journal of Materials Processing and Manufacturing Science*, 2024, 43(3), 602–609.

[7] Make cities and human settlements inclusive, safe, resilient and sustainable available at: https://sdgs.un.org/goals/goal11#targets_and_indicators accessed on 2nd July, 2024

[8] Israa, N., and Ghaidaa, A. "Waste Classification Using Artificial Intelligence Techniques: Literature Review", *Technium: Romanian Journal of Applied Sciences and Technology*. 2023, 5. 49-59. https://doi.org/10.47577/technium.v5i.8345

[9] Ogork, E. N., and Ibrahim, T. S. "Properties of Cement Paste and Concrete Containing Calcium Carbide Waste as Additive", *Nigerian Journal of Technology*, 2016, 36(1), 26-31, https://doi.org/10.4314/njt.v36i1.4

[10] Onovo, H. O., Agbeleye, A. A., Akano, T. T., Oludele, D. B., Olawoyin, J. O., and Kentosu, I. S. "Comprehensive study of extraction and applicability of nano silicate particles from natural waste for biopolymer reinforcement", *Nigerian Journal of Technology*, 2024, 43(4), 666 – 675. https://doi.org/10.4314/njt.v43i4.7

[11] Adedokun, S. I., Ganiyu, A. A., Adebajo, G. O., and Ogundele, A. S. "Comparative effects of selected wastes on the indices and strength properties of laterite soil", *Nigerian Journal of Technology*, 2024, 43(3), 428 – 435.

[12] Oke A., Ojo O. M., Olabanji T. O., Akinmusere O. K., and Akande S. P. "Mechanical properties of laundry wastewater concrete incorporating polyethylene 2 terephthalate (PET) as partial replacement for sand", *Nigerian Journal of Technology*, 2024, 43(4), 618–627. https://doi.org/10.4314/njt.v43i4.2

[13] Undiandeye, J. A., Kiman, S., Abubakar, M. A., and Mohammed, H. D. "Valorization of food waste for biogas production; effect of co-ensiling with maize straw at different C/N ratios", *Nigerian Journal of Technology*, 2023, 42(2), 282-288. https://doi.org/10.4314/njt.v42i2.17

[14] Laith A., Jinglan Z., Amjad J. H., Ayad A., Ye D., Omran A., J. Santamaría, Mohammed A. F., Muthana A. and Laith F. "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions", *Journal of Big Data*, 8, 53, 2021. https://doi.org/10.1186/s40537-021-00444-8

[15] Tang, Michael Chi Seng. "DenseNet201-Based Waste Material Classification Using Transfer Learning Approach", *Applied Mathematics and Computational Intelligence*, 13(2), 2024, 113-120.

[16] Shahab, S. and Mohd A. "Solid Waste Management Scenario in India and Illegal Dump Detection Using Deep Learning: An AI Approach towards Sustainable Waste Management", *Multidisciplinary Digital Publishing Institute (MDPI): Sustainability*, 2022, 14(23), 58-96. https://doi.org/10.3390/su142315896

[17] Wang, Xiaomo. "Improvement of EfficientNet in medical waste classification", *Science and Technology of Engineering, Chemistry and Environmental Protection*, 2024, 1(3), 1-6. https://doi.org/10.61173/dzxz2j87

[18] Munis, Z., Atika, S., and Eko, R. "Implementation of DenseNet121 Architecture for Waste Type Classification", *Advance Sustainable Science Engineering and Technology*, 2024, 6(3), 1-9. https://doi.org/10.26877/asset.v6i3.673

[19] Sunilkumar, K., Malathi, C., Surendarnath, S., Srilaxmi D., Amesho, K., and Sheela, J. "IoT-based waste management: hybrid optimal routing and waste classification model", *Environmental Science and Pollution Research*, 2024, 1-24. https://doi.org/10.1007/s11356-024-33418-3.

[20] Hadi, S., Ilham, H., Hilyah, M., and Afiyati, A. "A Hybrid Model for Dry Waste Classification using Transfer Learning and Dimensionality Reduction", *International Journal on Informatics Visualization*, 2024, 8(2) 623-634. https://dx.doi.org/10.62527/joiv.8.2.1943

[21] Adebimpe, A. M., Uguru-Okorie, D. C., and Oluwagunwa, E. O. "Design and Production of an Automatic Solid Waste Sorting Machine with Smart Digital Counter," *Nigerian Journal of Technology*, 2022, 41(3), 542-546.

[22] Rapeepan, P., Thanatkij, S., Surajet, K., Paulina, G., Kanchana, S., Natthapong, N., Sarayut, G., Peerawat, L., and Chawis, B. "Optimization-driven artificial intelligence-enhanced municipal waste classification system for disaster waste management", *Engineering Applications of Artificial Intelligence*, 2024, 133(6). https://doi.org/10.1016/j.engappai.2024.108614

[23] Lazrus, Vishal. "Multi class garbage classification Dataset", available at https://www.kaggle.com/datasets/vishallazrus/multi-

class-garbage-classification-dataset accessed on 17th April, 2024

[24] Simonyan, K., and Zisserman, A. “Very Deep Convolutional Networks for Large-Scale Image Recognition”, *arXiv preprint arXiv:1409.1556,* 2014. https://doi.org/10.48550/arXiv.1409.1556

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

[26] Chollet, F. “Xception: Deep Learning with Depthwise Separable Convolutions”, *Proceedings of the IEEE conference on computer vision and pattern recognition,* 2017, 1251-1258. https://doi.org/10.48550/ARXIV.1610.02357.