

Report-Smart Industrial Waste Sorting Systems

by User System

Submission date: 24-Aug-2024 08:34PM (UTC+0800)

Submission ID: 2437210388

File name: Report-Smart_Industrial_Waste_Sorting_Systems.docx (913.33K)

Word count: 8921

Character count: 52924

List of Tables

Chapter 03

Table 1: Hardware components used in the project.
Table 2: Software components used in the project.

Chapter 06

Table 3: Overview of CNN Models and Associated Workflow for Image Classification.
Table 4: Hyperparameters for training.

Chapter 07

Table 5: Testing results.

Chapter 08

Table 6: A result table after model training.
Table 7: A table showing the benchmarking results to compare the simulation results with real-world data.

Chapter 09

Table 8: A table showing the risk assessment of the project.

List of Figures

Chapter 03

Figure 1: Specifications of Software Block Diagram using ML

Chapter 05

Figure 2: Total number count of images per category

Chapter 06

Figure 3: Simulation to replicate the waste sorting process

Figure 4: Simulation of different categories of waste materials

Figure 5: Learning Curve for ResNet50

Figure 6: Learning Curve for EfficientNet B0

Figure 7: Learning Curve for MobileNet V2

Figure 8: Learning Curve for MobileNet V3 Large

Figure 9: Learning Curve for MobileNet V3 Small

Figure 10: Speed and throughput during model training

Chapter 07

Figure 11: A Python simulation of image processing

Figure 12: Testing Results from the training of the models

Chapter 08

Figure 13: A bar chart for result analysis of the trained datasets

Figure 14: Results of Benchmarking Test

Smart Industrial Waste Sorting Systems

Chapter 2

Introduction:

As the global population continues to grow and environmental concerns become increasingly urgent, effective waste management has emerged as a critical challenge. The Smart Waste Sorting System offers an innovative solution designed to address these challenges by leveraging advanced machine learning and computer vision technologies. This project focuses on automating the classification and sorting of waste into three primary categories: recyclable, compostable, and landfill.

The system utilizes a range of sophisticated Convolutional Neural Networks (CNNs) [1] presented in the design and algorithm chapter, including ResNet50, MobileNet V3 (Large & Small), EfficientNet B0, and MobileNet V2, all implemented using the PyTorch deep learning framework. These models are trained on a comprehensive dataset from Kaggle shown in the dataset chapter, which provides a diverse array of images representing the various waste categories. The Adam Optimizer from the optimization techniques chapter is employed to enhance model performance by dynamically adjusting learning rates, thereby improving accuracy and accelerating convergence. The training process aims to minimize cross-entropy loss, ensuring precise classification.

The implementation of this system promises significant improvements in waste management efficiency. By automating the sorting process, the Smart Waste Sorting System is expected to increase resource recovery rates, reduce contamination in recycling streams, and minimize the volume of waste directed to landfills which are mentioned in the analysis chapter. These advancements not only contribute to lowering greenhouse gas emissions associated with improper waste disposal but also support broader environmental sustainability goals.

In this respect, the Smart Waste Sorting System which integrates advanced technologies signifies a significant breakthrough in tackling a major environmentally related problem of our age. Its creative concept presents an applicable measure that supports current endeavors aimed at maintaining eco-friendly waste disposal as well as caring for the environment.

Chapter 03

3.1 Resource List

Hardware Resources:

Table 1: Hardware components used in the project

SL	Product Description	Qty	Unit Price (\$)	Amount (\$)
1.	INTEL 13 TH GEN CORE i5 13500 RAPTOR LAKE PROCESSOR 14 CORE 20 THREAD 2.5GHZ-4.8GHZ 24MB CACHE INTEL UHD 770	1	239.13	239.13
2.	MSI MAG B760M MORTAR DDR5 LGA1700 GAMING MOTHERBOARD	1	183.62	183.62
3.	TEAM T-FORCE DELTA TUF GAMING RGB 16GB 6000MHZ DDR5 GAMING RAM #FF5D516G6000HC38A01	2	67.47	134.94
4.	MAXGREEN 1200VA OFFLINE UPS (PLASTIC BODY)	1	52.10	52.10
5.	GIGABYTE C301 GLASS E-ATX GAMING CASE(BLACK)	1	57.22	57.22
6.	MSI GEFORCE RTX 4060 TI VENTUS 3X 16G OC GDDR6 GRAPHICS CARD	1	555.13	555.13
7.	SAMSUNG 980 PRO 500GB PCIE GEN4 M.2 NVME SSD	1	73.45	73.45
8.	COOLERMASTER MWE 750-WATT GOLD V2 POWER SUPPLY#MPE-7501-AFAAG-IN	1	999.23	999.23
9.	DEEPCOOL LE520 240MM ALL-IN-ONE ARGB LIQUID CPU COOLER	1	56.37	56.37
			Total =	2351.19 \$

This project necessitated the use of a high-performance computing setup to implement a CNN model for computer vision tasks. The key components include a 14-core processor with 20 threads which provides robust processing power for performing complex calculations. It is supported by a gaming motherboard that guarantees high-speed data transfer as well as compatibility with other hardware. A total of 32GB RAM is utilized thus efficiently accommodating large datasets and multitasking demands. The graphics card considerably accelerates deep learning processes, thereby reducing model training periods while data access. Hence, this system maintains its stability and performance owing to a power supply unit rated at 750 watts in addition to a liquid-cooled CPU of size 240 mm. The purpose of this configuration is to efficiently manage the computational requirements needed when training and deploying CNNs.

Software Resources:

Table 2: Software components used in the project

Tool	Functions	Why we selected this tool
Python Language	Programming language for implementing the project.	Widely used in AI/ML projects with extensive libraries and support.
PyTorch Deep Learning Framework	Framework for building and training neural networks.	Flexible and user-friendly, with strong community support for deep learning.
Pandas	Data manipulation and analysis library.	Efficiently handles large datasets and data preprocessing.
Matplotlib/Seaborn	Libraries for data visualization.	Enables clear and informative visualizations of data and model results.
Jupyter Notebook	Interactive coding environment.	Facilitates rapid prototyping, code testing, and easy sharing of results.
Adam Optimizer	Optimization algorithm for training neural networks.	Known for faster convergence and adaptive learning rates, improving model performance.
CrossEntropyLoss	Loss function for classification tasks.	Suitable for multi-class classification problems like waste sorting.

In this project, Python serves as the foundation, integrating essential libraries like PyTorch for building and training the CNN model. Pandas is used for data preprocessing, while Matplotlib/Seaborn visualizes data and training metrics. Jupyter Notebook enables interactive coding and experimentation. The Adam Optimizer updates model weights efficiently, and CrossEntropyLoss measures prediction errors, guiding the CNN model to improve accuracy in classifying waste materials. Together, these tools streamline the development, training, and evaluation of the CNN model in this computer vision task. [2]

3.2 Specifications of Software Block Diagram using ML (Python Programming Language)

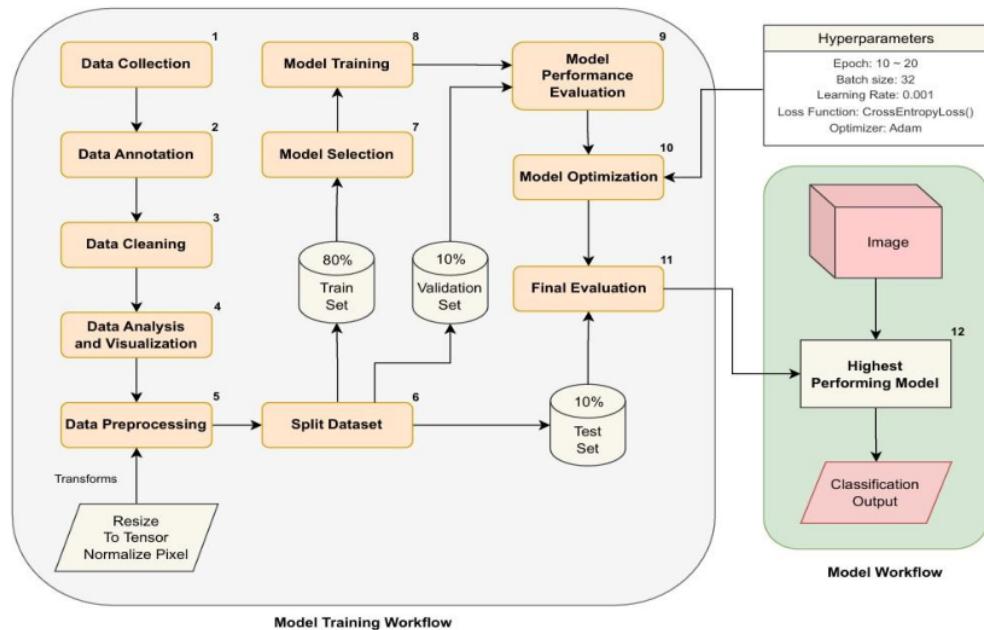


Figure1: Specifications of Software Block Diagram using ML

32

Figure 1 illustrates a comprehensive workflow used to train and assess a convolutional neural network (CNN) in the context of a waste classification project. Firstly, Data Collection and Data Annotation are made where raw data (images of waste) is collected and then labeled into three main categories: recyclable, compostable, and landfill. Thereafter, Data Cleaning is done to eliminate any inconsistencies or irrelevant information followed by Data Analysis and Visualization which establishes the general characteristics and distribution of the dataset. After that, we perform Data Preprocessing which involves resizing images converting them into tensors as well as normalizing pixel values so that they can meet the requirements of our model. Finally, the dataset is divided into training, validation, and test subsets with an allocation of 80% being set aside for training only while portions for validation and testing share 20%.

It's appropriate for us to select a model and then use our model training set in the second phase of the workflow. This training process is guided by specific hyperparameters such as epoch range, batch size, learning rate, loss function (CrossEntropyLoss), and optimizer (Adam). After training, the model undergoes Model Performance Evaluation using the validation set to assess its accuracy and performance. Based on these evaluations, there is an opportunity for improvement if the system is deemed unfit for the prospective outcome. Once optimized, the model must be subjected to Final Evaluation with the test set aimed at its efficiency under real-life conditions. Finally, this will lead to the classification of images in addition to the final classification output depending on the best-performing models generated through this process [3].

Chapter 05

5.1 Data Sources:

The data used in this project is sourced from three different online datasets containing images and information about the three main categories of waste materials: recyclable, compostable, and landfill. Now the three datasets that were used in the project are discussed below:

Dataset 1: Garbage Dataset

The dataset contains images belonging to ten classes, including 994 metal images, 3,039 glass images, 983 biological images, 1,650 paper images, 944 battery images, 772 trash images, 1,810 cardboard images, 1,977 shoe images, 5,323 clothing images, and 1,915 plastic images. The same dataset has also been used in a research paper "Managing Household Waste Through Transfer Learning" which made use of three online datasets for recyclable, compostable, and landfill waste. To balance ~~the~~ class sizes bigger classes have been under sampled while data augmentation was employed. The dataset was separated ~~into training, validation, and testing sets where VGGNet-16 and ResNet-50 models were trained through PyTorch for better performance comparison. It is licensed under the MIT License.~~ [4]

Dataset 2: Garbage Classification

Within the confines of this dataset, there are in total of 15,150 pictures illustrating domestic waste divided into twelve categories: paper, cardboard, biological waste, metals, plastics (in three varieties – green bottles, brown bottles, or white jars.), clothes or dresses; shoes (for both men and women), batteries and ordinary rubbish. For instance, some pieces of clothing and 22% of footwear were taken from a dataset that consists of over 5,000 images belonging to twenty different classes published as CC0: Public Domain on Kaggle – on top of this; it contributed nearly 29% to nine other classes identified with one category each. Furthermore, six main categories are present in the Garbage Classification Dataset: cardboard consists of 393 objects while glass holds 491 scrap pieces related to it; further still metal has 400 items while paper comprises 584 constituents making up some more assemblies like those formed by plastics totaling 472 pieces including trash comprising 127 elements that fall under Data files © Original Authors License. Yet again, this entire dataset has been opened under the Database: Open Database, Contents: © Original Authors License. [5]

Dataset 3: The Recyclable and Household Waste Classification Dataset

This dataset encompasses 30 categories of 15000 images (256 x 256 pixels), with 500 images in each category, to help support waste classification and recycling research. It serves as a basis for developing accurate waste-sorting systems that are used for scientific and educational purposes, as well as non-profit initiatives. The dataset is available under the MIT License. [6]

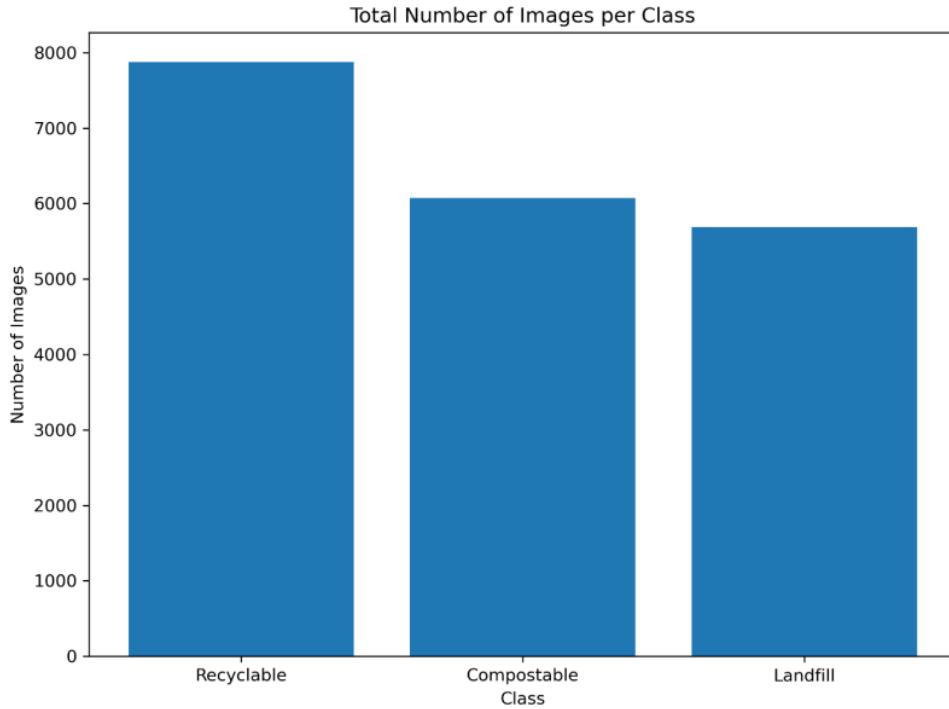


Figure 2: Total number count of images per category

All datasets were obtained from Kaggle for this project. Three types of waste were used—recyclable (7,872 images), compostable (6,071), and landfill (5,689)—for performing sorting ting as shown above in Figure 2. Appropriate waste subtypes for the three categories were locked into one single dataset by a human annotator to maximize data collection and minimize mismatches in the project.

5.2 Data Annotation:

This project makes use of datasets from Kaggle that have been classified into three major categories of waste, namely: recyclable (7,872), compostable (6,071), and landfill (5,689). The data used in the study was gathered from three separate online data repositories where each had a specific categorization of refuse materials. The Garbage Dataset served as Dataset 1 by providing images belonging to ten different classes such as metal, glass, and biological. A total of 15,150 photographic depictions organized into twelve home garbage classes constituted Dataset 2 that is the Garbage Classification Dataset; partly sourced from around Clothing dataset was clothes and shoes among other sections. Recyclable and Household Waste Classification Dataset was Dataset3 which contains 15 thousand pictures grouped under thirty classifications to establish a waste sorting system.

The waste materials were meticulously categorized under the three primary waste categories:

- **Category 1: Recyclable**
 - **Paper and Cardboard:** Newspapers, magazines, cardboard boxes, office paper.
 - **Plastics:** PET bottles, HDPE containers, PVC pipes, LDPE bags, PP containers, PS products.
 - **Glass:** Bottles (clear, green, brown), jars.
 - **Metals:** Aluminum cans, tin cans, foil, scrap metal.
 - **Electronic Waste (e-waste):** Computers, mobile phones, cables, small electronic devices.
- **Category 2: Compostable**
 - **Food Scraps:** Fruit and vegetable peels, leftover food, coffee grounds, tea bags.
 - **Yard Waste:** Leaves, grass clippings, branches, garden trimmings.
 - **Compostable Materials:** Biodegradable packaging, compostable bags, paper towels, and napkins (if not contaminated)
- **Category 3: Landfill**
 - **General Household Trash:** Non-recyclable plastics (e.g., plastic bags, wrappers), contaminated paper products, ceramics, and pottery.
 - **Textiles and Clothing:** Worn-out clothing, shoes, fabric scraps.
 - **Furniture:** Broken furniture, upholstered items.
 - **Hazardous Waste (specific handling required):** Batteries, paints and solvents, chemicals, fluorescent bulbs, medical waste (e.g., syringes, expired medicines).
 - **Construction and Demolition Waste:** Concrete, bricks, insulation materials.

Then, a human annotator combined these classified waste subtypes into a comprehensive dataset to make data collection more efficient and minimize mismatches. Class imbalances were addressed through data augmentation methods, with the dataset divided into training set, validation set, and testing set. This annotated dataset constituted the basis upon which various deep learning models such as MobileNet V3Large and ResNet-50 were trained using the PyTorch framework. [7]

Chapter 06: Design & Development

6.1 Algorithm Development

The CNN model has been employed in the project for interpreting unprocessed image data into substantial categorizations which propel the machine-driven garbage classification operation thereby making waste administration more effective and precise. Besides, Convolutional Neural Networks (CNNs) can be said to achieve an unrivaled performance in computer vision. In such tasks as object recognition, image classification, and segmentation it is able to automatically learn from the photographs' features thereby carrying out these tasks with remarkable precision. Usually, a simple convolutional neural network consists of three types of layers namely convolutional layer(s), pooling layer(s), fully connected layer(s), plus input layer and output layer(s). The functions played by each one of these layers shall now be examined in detail. [8]

- **Input layer:** The CNN takes an input image represented as a grid of pixels. For color images, each pixel has three color channels (red, green, and blue), while for grayscale images, there is only one channel. **Note: ImageNet** is a large-scale dataset that contains millions of labeled images spanning thousands of different categories for data collection in computer vision tasks.
- **Convolutional layer:** The convolutional layer in CNN performs the convolution operation using learnable filters or kernels. These kernels slide over the input image, multiplying their weights with the corresponding pixels in the receptive field and summing the results to produce a feature map. Multiple kernels capture different features, which allows the network to learn a wide range of patterns.
- **Activation function:** After the convolution operation, an activation function (e.g. ReLU) is applied element-wise to introduce nonlinearity into the network. This allows CNNs to learn complex relationships between the extracted features.
- **Pooling layers:** Pooling layers are used to down sample the feature maps generated by the convolutional layers. Pooling reduces the spatial dimensionality of the feature maps while preserving the most important information.
- **Repeat convolution, activation, and pooling:** Each subsequent layer learns more abstract features by building upon the representations learned in the previous layers.
- **Flattening:** After several convolutional and pooling layers, the resulting feature maps are flattened into a one-dimensional vector. This collapses the spatial structure of the features into a linear representation.
- **Fully connected layers:** The flattened vector is fed into fully connected layers, which perform traditional neural network operations. These layers learn to classify the input based on the extracted features. The final fully connected layer produces the output, representing the predicted class probabilities or specific values for the task at hand.
- **Training and optimization:** During the training phase, the CNN's parameters (filter weights, biases, etc.) are optimized to minimize a defined loss function. This is done through backpropagation, where gradients are computed and used to update the parameters via optimization algorithms like gradient descent. The

training process adjusts the network's weights to improve its ability to make accurate predictions. **Note:** ResNet50 is a popular pre-trained model for CV tasks trained on ImageNet.

- **Inference:** Once the CNN is trained, it can be used for inference on new, unseen images. The forward pass through the network generates predictions or class probabilities for the given input image.

There are various types of CNN models for image classifications. For the project, six models of CNN are used to design the algorithm. Here, is a table with the model's name, and their workflow is given below:

Table 3: Overview of CNN Models and Associated Workflow for Image Classification.

Model Name	Model Architecture
ResNet50	~25.6 million parameters
MobileNet V3 Large	~5.4 million parameters
EfficientNet B0	~5.3 million parameters
MobileNet V2	~3.4 million parameters
MobileNet V3 Small	~2.5 million parameters

In the project, various CNN models with PyTorch are used to perform computer vision tasks, particularly image classification. Here's a brief overview of each model and its benefits for image processing:

1. ResNet50

- **Description:** ResNet50 is a deep CNN with 50 layers, leveraging residual learning to prevent vanishing gradients in very deep networks. It includes shortcut connections that skip one or more layers, allowing the model to train effectively even with increased depth.
- **Benefits for Image Processing:** ResNet50 excels at capturing complex image features, making it highly accurate for detailed image classification tasks. [9]

2. MobileNet V3 Large

- **Description:** MobileNet V3 Large is a lightweight model optimized for mobile and embedded devices. It uses depth-wise separable convolutions combined with Squeeze-and-Excite modules to balance accuracy and efficiency.
- **Benefits for Image Processing:** This model is well-suited for real-time image processing in resource-constrained environments due to its low computational cost and high efficiency. [10]

3. MobileNet V3 Small

- **Description:** MobileNet V3 Small is a more compact version of MobileNet V3, designed specifically for applications where even lower computational resources are available. It maintains the core architecture while being optimized for minimal power consumption.
- **Benefits for Image Processing:** Ideal for ultra-efficient applications, MobileNet V3 Small offers fast and lightweight image processing capabilities, making it perfect for small-scale devices.

21 4. EfficientNet B0

- **Description:** EfficientNet B0 is the base model of the EfficientNet family, known for its compound scaling approach, which uniformly scales network depth, width, and resolution. This results in a more efficient model that offers high accuracy with fewer parameters.
- **Benefits for Image Processing:** EfficientNet B0 provides an excellent trade-off between accuracy and computational efficiency, making it ideal for applications where resource constraints exist but high accuracy is still needed. [11]

22

5. MobileNet V2

- **Description:** MobileNet V2 improves upon its predecessor with an inverted residual structure and linear bottlenecks. These features make it more efficient, especially for devices with limited computing power.
- **Benefits for Image Processing:** MobileNet V2 is highly effective in reducing memory usage and computational cost while maintaining accuracy, making it a strong choice for real-time image processing on mobile devices.

PyTorch and Computer Vision

In the project, PyTorch is used as the deep learning framework to implement and train these CNN models. PyTorch provides a flexible and intuitive interface for defining neural networks, making it easier to build and experiment with different architectures. For computer vision tasks, PyTorch offers specialized libraries like torch vision, which includes pre-trained models, image transformations, and datasets tailored for image classification, object detection, and more.

4

The Convolutional Neural Network (CNN) model processes waste images for classification, demonstrating the application of machine learning and computer vision in the Smart Waste Sorting System. It begins with input images of different waste types—such as plastic bottles, snack wrappers, and cardboard—passed through a series of convolutional and max-pooling layers. These convolutional layers extract essential features like edges, textures, and shapes, while max-pooling layers reduce the spatial dimensions, preserving critical features while enhancing computational efficiency. The processed features are then passed through fully connected layers, which serve as the classifier by analyzing and combining these features to determine the type of waste. Finally, the output layer generates the classification result, identifying the waste category, which is then used to automate and optimize the sorting process within the system. This workflow exemplifies how CNNs, integrated with machine learning and computer vision, are effectively applied to enhance the project's waste sorting accuracy and efficiency. [12]

6.2 Simulation Environment Development

6.2 (a). Modeling:

To create a simulation environment to replicate the waste sorting process a diagram is shown below:

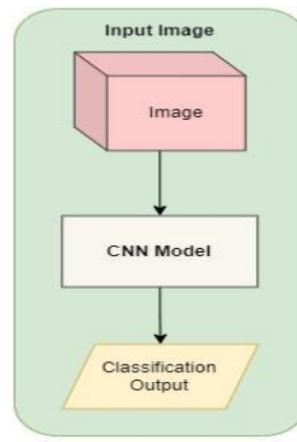


Figure 3: Simulation to replicate the waste sorting process

The diagram presents a simplified outline of the modeling process in the Smart Waste Sorting System project, which uses to simulate waste sorting. The initial stage involves an input image being fed into a Convolutional Neural Network (CNN) model. The CNN model performs relevant feature extraction and classification on the image. The output from this classification process thus determines what kind of waste it is (e.g., recyclable, compostable or landfillable). With such modeling approach, a simulated environment for replicating the waste sorting process is created; henceforth making it possible to test and improve on them before they are deployed into reality

6.2 (b). Parameters:

The hyperparameters table for training is given below:

Table 4: Hyperparameters for training

Hyperparameter	Value
Number of Epochs	20
Batch Size	32
Learning Rate	0.001
Optimizer	Adam
Loss Function	CrossEntropyLoss()
Train Size	15704 images
Test Size	1965 images

The following are the essential hyperparameters and particulars of the dataset employed for conducting the Smart Waste Sorting System. A convolutional Neural Network (CNN) was trained over 20 epochs with batch size 32 so that it could learn effectively from the input data. A learning rate of 0.001 was set to balance speed with stability during training work; Adam Optimizer characterized this configuration through weight-adjusting mechanisms. For multi-class classification problems, loss function choice falls under the CrossEntropyLoss category. There were 15,704 images in the training dataset so that this model could learn more at once finally evaluating generalization capabilities based on its performance using a test set consisting of 1,965 samples taken at random. The use of these hyperparameters together with dataset specifications facilitated the optimization of the model for more precise and efficient waste classification utilizing attaining a perfect balance between speed and stability in the training process while also adjusting weights of the network through Adam optimizer thereby making it possible for CNN to learn efficiently from given input data.

6.2. (c). Scenarios:

To simulate different scenarios a diagram is given below:

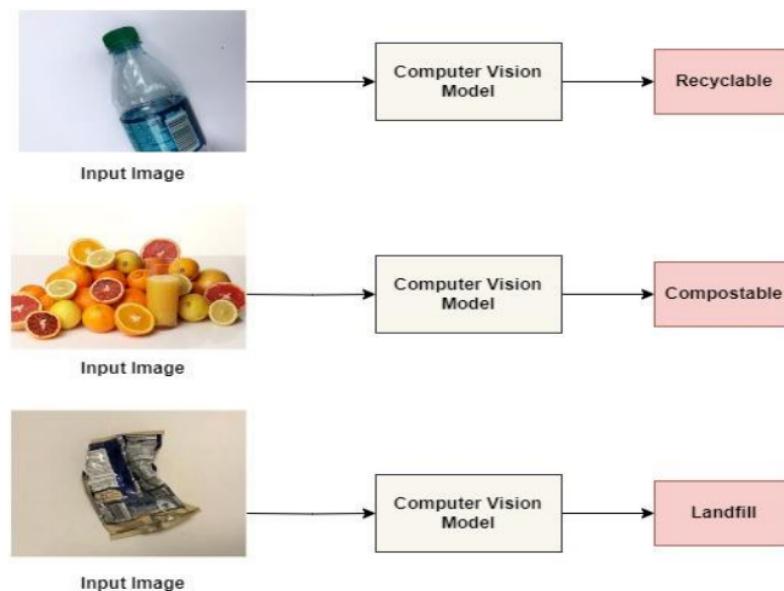


Figure 4: Simulation of different categories of waste materials.

The model takes into account different types of waste and shows how they are sorted and analyzed by an algorithm using computer vision technology. Various scenarios were simulated in a Smart Waste Sorting System, as can be seen from the diagram above. A plastic bottle, grass, and an aluminum can are some examples of the images that were fed into this algorithm to help distinguish one solid waste from another. The bottle would be seen as 'recyclable' by the computer; while grass is 'compostable'. On the other hand, energy drink cans would be regarded as 'landfills' among other wastes. This finding indicates that the model can accurately identify different kinds of waste which could aid in increasing recycling rates and reducing land disposal in real-life cases.

6.3 Optimization Techniques Development

6.3. (a). Optimization Algorithms:

The Adam (Adaptive Moment Estimation) optimizer is a highly effective and widely-used optimization algorithm in deep learning, particularly suitable for training Convolutional Neural Networks (CNNs) like the one in your Smart Waste Sorting System project. By dynamically adjusting the learning rates of the model's parameters during training, Adam facilitates faster convergence and enhances model accuracy, even in the presence of noisy or sparse gradients. This makes it ideal and efficient for your project, where precise classification of waste types is crucial for automated sorting. Adam's ability to fine-tune the CNN model efficiently ensures that the system can accurately distinguish between different waste categories, thereby improving the overall effectiveness of the smart sorting process.

An optimizer is a major determinant of how the learning curve of a model evolves. In the Smart Waste Sorting System project that employs a Convolutional Neural Network (CNN) for waste classification, Adam comes in handy in realizing faster and more stable convergence. Normally, the learning curve which depicts the performance of model (usually loss or accuracy) against training iterations shows a rapid decline in loss or rise in accuracy when using Adam. This is attributed to the fact that it combines the benefits of both momentum (which speeds up convergence) and adaptive learning rates (which fine-tunes the learning process) thus producing smoother and more resourceful learning curves. Thus, during the initial stages of training, there will be quick improvements portrayed by the learning curve before taking an even course towards optimality minimizing chances of overfitting and better generalization.

Now the learning curves for the project are shown below:

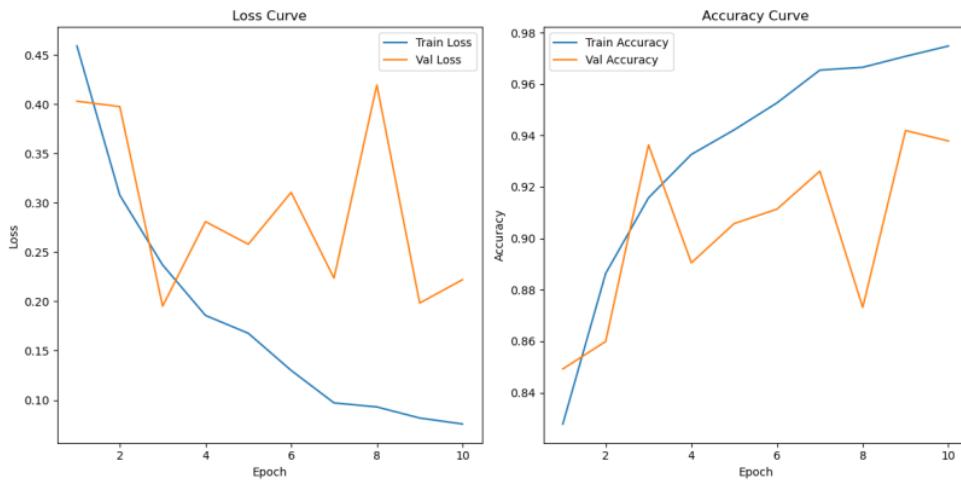


Figure 5: Learning Curve for ResNet50

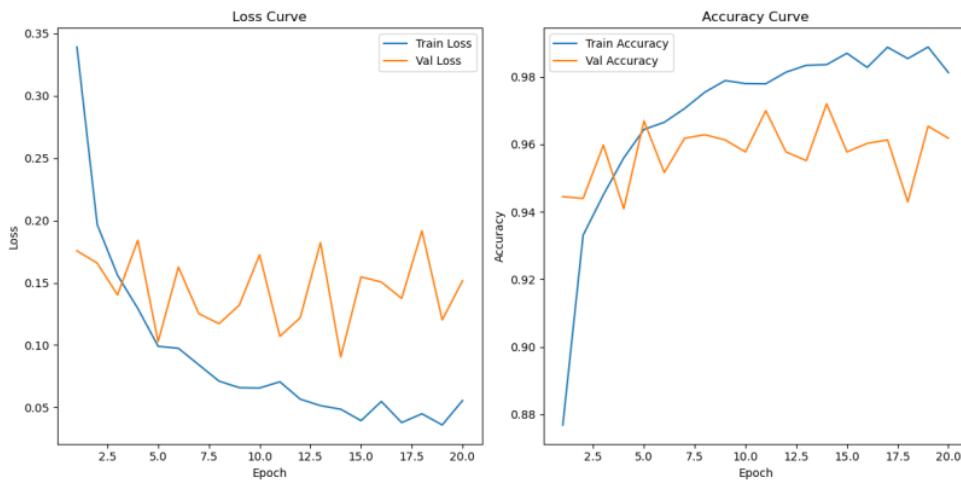


Figure 6: Learning Curve for EfficientNet B0

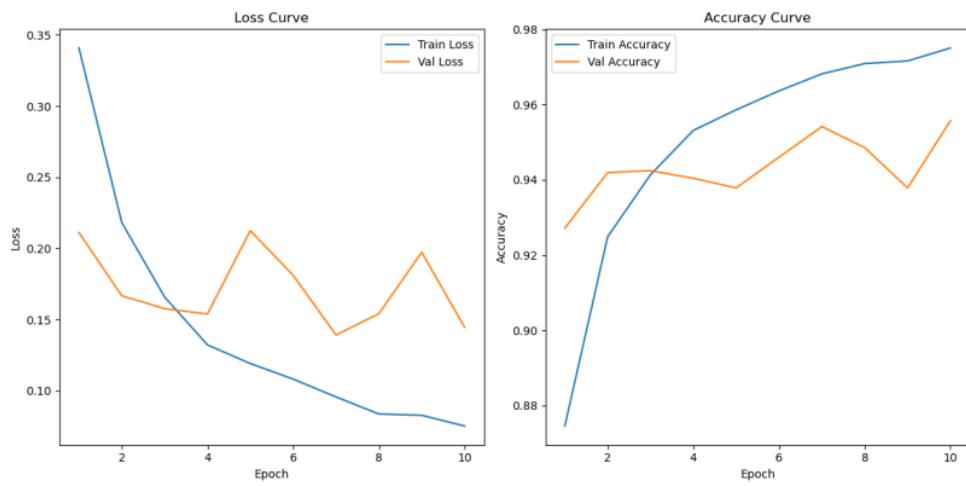


Figure 7: Learning Curve for MobileNet V2

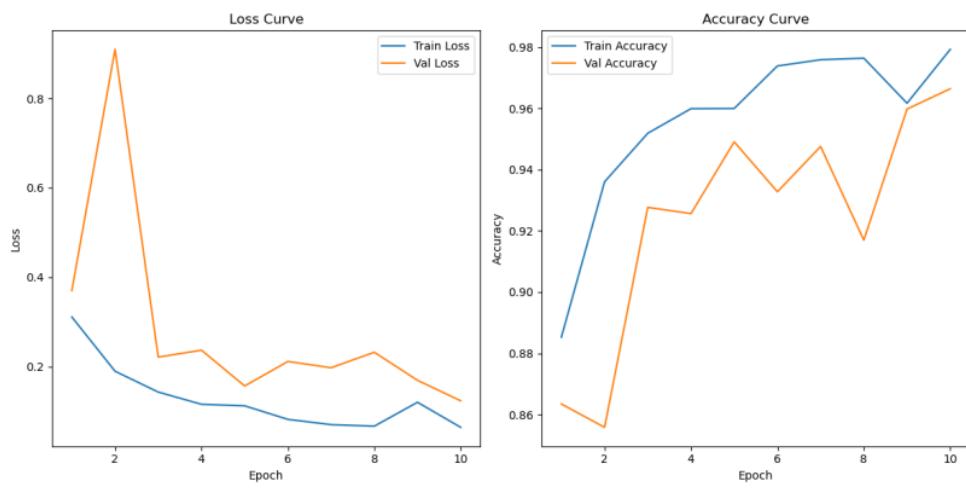


Figure 8: Learning Curve for MobileNet V3 Large

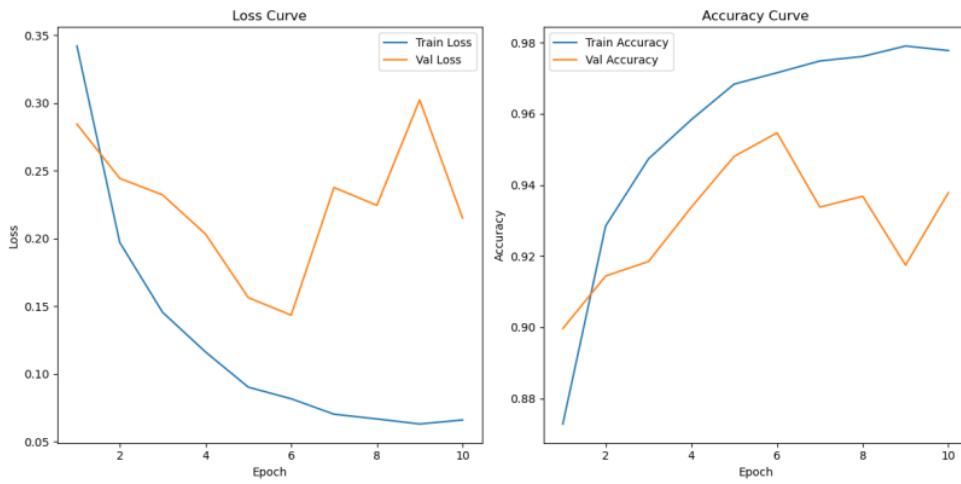


Figure 9: Learning Curve for MobileNet V3 Small

In this case, during model training the goal is to minimize loss, thereby increasing effectiveness. Lower losses generally mean that models can learn better and therefore predict accurately, resulting in increased accuracy. The rise in accuracy accompanying the above reduction of losses confirms that the CNN model effectively distinguishes different types of waste and hence can be trusted for real-life applications.

6.3. (b). Performance Metrics:

In the Smart Waste Sorting System project, accuracy is a critical metric used to evaluate the performance of the CNN models. Accuracy measures the proportion of correctly classified waste items out of the total number of items. It is calculated using the formula:

$$\text{Accuracy} = (\text{Number of Correct Predictions} \times 100) / (\text{Total Number of Predictions})$$

The percentage indicates how well is the model distinguishing between classes of rubbish among, plastic, paper, and metal. However, a higher accuracy value of the model means it learns effectively different shapes and sizes of all waste classes which is very important for automatic separating. When monitoring accuracy over time during training or validation phases, it is possible to evaluate CNN model performance and make necessary changes to enhance it accordingly.

Now to show the speed and throughput during the training of the models a screenshot is given below:

```
Checkpoints/resnet50/model_epoch_9.pth
Testing: 100% | 62/62 [00:15<00:00, 3.97it/s]
Test Loss: 0.2127, Test Accuracy: 94.30%
Checkpoints/mobilenet_v3_large/model_epoch_10.pth
Testing: 100% | 62/62 [00:12<00:00, 4.84it/s]
Test Loss: 0.1742, Test Accuracy: 95.42%
Checkpoints/mobilenet_v3_large_20ep/model_epoch_11.pth
Testing: 100% | 62/62 [00:12<00:00, 4.85it/s]
Test Loss: 0.1659, Test Accuracy: 94.86%
Checkpoints/efficientnet_b0/model_epoch_14.pth
Testing: 100% | 62/62 [00:13<00:00, 4.57it/s]
Test Loss: 0.1549, Test Accuracy: 95.78%
Checkpoints/mobilenet_v2/model_epoch_7.pth
Testing: 100% | 62/62 [00:13<00:00, 4.67it/s]
Test Loss: 0.1662, Test Accuracy: 94.86%
Checkpoints/mobilenet_v3_small/model_epoch_6.pth
Testing: 100% | 62/62 [00:12<00:00, 4.79it/s]
Test Loss: 0.1687, Test Accuracy: 94.55%
```

Figure 10: Speed and throughput during model training

Chapter 07: Testing

7.1 Prototype Development:

To build a prototype system based on the optimized algorithms a simulation diagram is shown below:

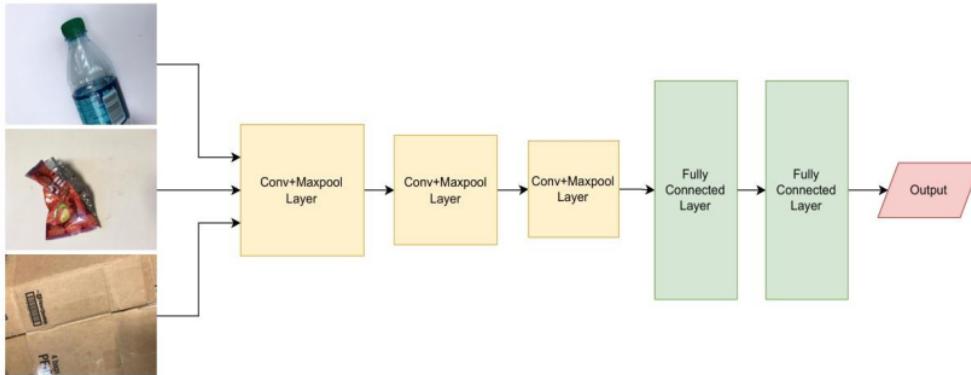


Figure 11: A Python simulation of image processing

Inputting pictures of trash such as a bottle, snack wrappers, and cardboard marks the beginning of the digital (Figure: 10) way followed by CNN models while processing them within a Smart Waste Sorting system project. In this convolutional layer's stage, important features that are distinctive to every type of waste including edges, texture, and shapes are extracted by CNN. Max-pooling layers are then applied to reduce spatial dimensions while preserving the most important features helping in less computing power usage and contained within the layered section before being passed onto fully connected ones that act like classifiers in networks. The extracted features will be combined and analyzed in these layers to classify each waste product as belongs-to what category. Finally, an output layer is generated which divides waste products into corresponding recycling bins leading to an automatic and efficient waste sorting system. This gives an illustration of how machine learning and computer vision could work hand-in-hand when it comes to refining a garbage management approach through precise imaging classification methods.

7.2. Testing:

To test the prototype in a controlled environment with real waste samples a table of testing and a screenshot of model training is given below:

Table 5: Testing results

Model Name	Test Loss	Test Accuracy
ResNet50 [18]	0.2127	94.30%
MobileNet V3 Large	0.1742	95.42%
EfficientNet B0	0.1742	95.78%
MobileNet V2 [18]	0.1662	94.86%
MobileNet V3 Small	0.1687	94.55%
MobileNet V3 Large (Trained for 20 epoch)	0.1659	94.86%

Screen Shot:



Figure 12: Testing Results from the training of the models.

The above diagram and the table both displays the testing results for different Convolutional Neural Network (CNN) models used in the Smart Waste Sorting System project. For each model, the testing process has been completed with 100% coverage, as indicated by the progress bars. The results show the test loss and test accuracy for each model.

- ResNet50 shows a test loss of 0.2127 with a test accuracy of 94.30%.
- MobileNet V3 Large (after 10 epochs) has a test loss of 0.1742 with a test accuracy of 95.42%.
- MobileNet V3 Large (after 20 epochs) displays an improved test loss of 0.1659 with a test accuracy of 94.66%.

- **EfficientNet B0** reports the lowest test loss of 0.1549 and the highest test accuracy of 95.78%.
- **MobileNet V2** shows a test loss of 0.1662 and an accuracy of 94.86%.
- **MobileNet V3 Small** has a test loss of 0.1687 with a test accuracy of 94.55%.

These results reflect the performance of each model in classifying waste items based on images, where lower test loss and higher test accuracy indicate better model performance. EfficientNet B0 stands out with the highest accuracy, suggesting it may be the most effective model for this specific application.

Chapter 08: Analysis

8.1 Validation Result Analysis:

To validate the model's accuracy and reliability, a result table and a diagram of the model's result analysis are given below:

Table 6: A result table after model training

Model Name	Train loss	Train accuracy	Validation loss	Validation accuracy	Test loss	Test accuracy
ResNet50	0.0817	97.08%	0.1983	94.19%	0.2127	94.30%
MobileNet V3 Large	0.0641	97.92%	0.1234	96.64%	0.1742	95.42%
MobileNet V3 Large (Trained for 20 epoch)	0.0666	97.73%	0.1155	96.03%	0.1659	94.86%
EfficientNet B0 (Trained for 20 epoch)	0.0486	98.36%	0.0906	97.20%	0.1742	95.78%
MobileNet V2	0.0957	96.82%	0.1392	95.42%	0.1662	94.86%
MobileNet V3 Small	0.0817	97.15%	0.1435	95.47%	0.1687	94.55%

Here's an explanation of each model's performance based on the table:

1. ResNet50:

- **Training Phase:** ResNet50 shows a train loss of 0.0817 and a high train accuracy of 97.08%, indicating that the model has learned well during training.
- **Validation Phase:** The validation loss increases to 0.1983, with a validation accuracy of 94.19%, showing a slight drop in performance during validation, possibly indicating slight overfitting.
- **Testing Phase:** The model performs similarly in the testing phase with a test loss of 0.2127 and a test accuracy of 94.30%, demonstrating reasonable generalization.

2. MobileNet V3 Large:

- **Training Phase:** Achieves a lower train loss of 0.0641 and a higher train accuracy of 97.92%, indicating efficient learning during the training phase.
- **Validation Phase:** The validation loss is 0.1234, and the validation accuracy is 96.64%, showing good generalization from training to validation.
- **Testing Phase:** MobileNet V3 Large achieves a test loss of 0.1742 with a test accuracy of 95.42%, indicating strong overall performance.

3. MobileNet V3 Large (Trained for 20 Epochs):

- **Training Phase:** Slightly higher train loss of 0.0666 with a train accuracy of 97.73%, showing that additional epochs maintained efficient learning.
- **Validation Phase:** The validation loss drops to 0.1155, with a validation accuracy of 96.03%, suggesting that more training epochs contributed to better performance.
- **Testing Phase:** This version of MobileNet V3 Large achieves a test loss of 0.1659 with a test accuracy of 94.86%, demonstrating consistent performance improvements across phases.

4. EfficientNet B0 (Trained for 20 Epochs):

- **Training Phase:** Displays the lowest train loss of 0.0486 and the highest train accuracy of 98.36%, indicating very efficient learning.
- **Validation Phase:** Validation loss is also low at 0.0906, with a validation accuracy of 97.20%, showing excellent generalization from training to validation.
- **Testing Phase:** Despite the strong validation performance, the test loss increases to 0.1742, but the model still maintains a high-test accuracy of 95.78%, making it one of the best-performing models.

5. MobileNet V2:

- **Training Phase:** The model achieves a train loss of 0.0957 with a train accuracy of 96.82%, slightly lower than the others, indicating good learning but with room for improvement.
- **Validation Phase:** Validation loss is 0.1392, and validation accuracy is 95.42%, showing moderate generalization.

- **Testing Phase:** In the testing phase, MobileNet V2 has a test loss of 0.1662 with a test accuracy of 94.86%, indicating solid but not exceptional performance compared to other models.

6. MobileNet V3 Small:

- **Training Phase:** This model has a train loss of 0.0817 with a train accuracy of 97.15%, indicating efficient learning similar to ResNet50.
- **Validation Phase:** The validation loss is 0.1435, and the validation accuracy is 95.47%, showing good generalization but slightly lower than MobileNet V3 Large.
- **Testing Phase:** With a test loss of 0.1687 and a test accuracy of 94.55%, MobileNet V3 Small performs well but is outperformed by other versions of MobileNet and EfficientNet B0.

Overall, EfficientNet B0 and MobileNet V3 Large emerge as the top performers, while ResNet50, MobileNet V2, and MobileNet V3 Small offer solid, reliable performance.

Now from the table, a graph is plotted below to show the analysis more efficiently:



Figure 13: A bar chart for result analysis of the trained datasets

The bar diagram represents the spread of pictures in three ranges: recyclable, compostable, and landfill. Each of these ranges is present in training with 80%, validation with 10% and testing also with 10%. Among them, the highest number of images can be found in the "Recyclable" class while "Compostable" comes second followed by "Landfill." Although the ratios between classes are somewhat same across all partitions of dataset, there seems to have a decline in total number of pictures given from such sets as training through validation up to testing.

8.2 Benchmarking:

30

To compare the simulation results with real-world data a table is given below:

37

Table 7: A table for showing the benchmarking results to compare the simulation results with real-world data

Benchmarking Dataset	Model Name	Test Loss	Test Accuracy
	ResNet50	0.0786	96.83%
	MobileNet V3 Large	0.0279	98.83%
	MobileNet V3 Large (Trained for 20 epoch)	0.0292	98.67%
	EfficientNet B0 (Trained for 20 epoch)	0.0078	100.00%
	MobileNet V2	0.0506	98.17%
	MobileNet V3 Small	0.0724	97.67%

Screenshot of results from which Table 7 was derived:

Operate Benchmarking Test

```
[25]: num_classes = 3
loss_fn = nn.CrossEntropyLoss()

for get_model, checkpoint_path in zip(model_functions, checkpoint_paths):
    # Load the model using the function
    loaded_model = load_model(get_model, checkpoint_path, num_classes)
    loaded_model = loaded_model.to(device)
    # Evaluate the loaded model on the test set
    print(checkpoint_path)
    evaluate_model(loaded_model, test_loader, loss_fn)

Checkpoints/resnet50/model_epoch_9.pth
Testing: 100%|██████████| 19/19 [00:04<00:00, 4.01it/s]
Loss: 0.0786, Accuracy: 96.83%
Checkpoints/mobilenet_v3_large/model_epoch_10.pth
Testing: 100%|██████████| 19/19 [00:03<00:00, 5.29it/s]
Loss: 0.0279, Accuracy: 98.83%
Checkpoints/mobilenet_v3_large_20ep/model_epoch_11.pth
Testing: 100%|██████████| 19/19 [00:03<00:00, 5.17it/s]
Loss: 0.0292, Accuracy: 98.67%
Checkpoints/efficientnet_b0/model_epoch_14.pth
Testing: 100%|██████████| 19/19 [00:03<00:00, 5.12it/s]
Loss: 0.0078, Accuracy: 100.00%
Checkpoints/mobilenet_v2/model_epoch_7.pth
Testing: 100%|██████████| 19/19 [00:03<00:00, 5.19it/s]
Loss: 0.0506, Accuracy: 98.17%
Checkpoints/mobilenet_v3_small/model_epoch_6.pth
Testing: 100%|██████████| 19/19 [00:03<00:00, 5.46it/s]
Loss: 0.0724, Accuracy: 97.67%
```

Figure 14: Results of Benchmarking Test

The table displays the performance indicators of several CNN architectures tested on a dataset. EfficientNet B0 that had been trained for twenty epochs attained the highest accuracy at 100% with a minimal test loss at 0.0078 which indicates a remarkable performance history. Another notable predictor is MobileNet V3 Large which reached an accuracy of 98.83% against the baseline at test accuracy against the baseline of two with only little loss recorded as 0.0279 while the rest; ResNet50 and MobileNet V3 Small had slightly less but still over 96%.

Comparing Table 5 and Table 7 we get notable advancements in the performance of CNN models following further training. The test accuracy of ResNet50 increased from 94.30% to 96.83%, while the test loss decreased. On the other hand, MobileNet V3 Large had its accuracy improve from 95.42% to 98.83%, accompanied by a lower test loss. Lastly, EfficientNet B0 achieved a perfect accuracy of 100%. This signifies its usefulness. In essence, the improved test accuracies and reduced losses of these models imply better generalizability and efficiency in a smart waste sorting framework. [13]

8.3 Analysis of waste management efficiency:

The importance of optimized sorting in the industry cannot be overemphasized as it plays a key role on enhancing the efficiency of waste management, which comes with numerous advantages including operational, economic, and environmental benefits. Advanced sorting methods like machine learning algorithms, computer vision systems as well as automated machinery enable industries to achieve greater accuracy when it comes to separating various types of waste. Waste materials that can either be recycled or not are hence properly identified and separated from those that are organic or hazardous.

If we consider it operationally, optimized sorting reduces misclassification of wasted products therefore there is no need for secondary sorting processes which take too much time and are expensive. The entire waste management operation becomes streamlined allowing faster processing times as well as better flow rates. Consequently, this leads to lower labor costs since processing materials takes less time and resources thereby reducing electricity consumption at plant locations. The economic implications of an improved sorting process are the increased value of recovered materials in terms of waste recycling by state governments because they will have some amount of resources left that could have been lost through disintegration or incineration-based recycling methods thus during reprocessing everything would become cleaner than before so compared with option number one all inputs remain constant while outputs vary depending only on changes made to those inputs' prices about their scarcity levels on world markets today whereby these per unit values expressed both in real terms (as opposed to nominal).

Therefore, even if more expensive machines might produce smaller percentages, then again cheaper ones would still produce larger ones provided that their operation design allows all possible combinations within this operation cycle space considering other factors such as energy consumption etcetera until finally concluding general observations based upon observational data analyses rather than non-analytical merely mathematical predictions aimed at showing just how well some hypothesis could potentially function

when applied elsewhere beyond this specific domain without specifying distinct roles played by each member involved in teamwork settings under certain conditions taking into account everything else being equal while highlighting major points made earlier concerning this case study including its outcome.

Reduction of overall waste management environmental footprint is therefore largely dependent on properly organized sorting for example in an eco-friendly manner. Less virgin raw materials are therefore used hence lower levels of resource extraction and related environmental degradation as more materials are recycled or reused. In addition, optimized sorting leads to lesser greenhouse gas emissions since more waste is not dumped into landfills where it decomposes releasing methanous gas that is known to have high greenhouse warming potential. Finally, optimized sorting greatly improves the efficiency of the waste management process in the industry by streamlining operational processes, boosting the economic worth of recyclables, and minimizing ecological consequences on. Thus, industries and the environment both benefit from this waste management system that is more eco-friendly and pocket-friendly.

8.4 Reduction in Contamination:

Upgraded sorting considerably diminishes pollution in recycling channels that are vital for the productivity and longevity of the industry. Contamination is when non-recyclable substances or wrongly classified recyclables get into the recycling stream resulting to low quality recycled products. This pollution leads to the rejection of complete consignment hence increasing refuse and operation expenditures. As sorting procedures are improved either by means of technological advancements like machineries using artificial intelligence or enhanced manual sorting protocols, it becomes possible to classify accurately the materials into different groups. Exact sorting curtails the number of foreign substances found in stockpiles hence making sure that each kind of recycled product is clean and ready for re-fabrication.

Industries can enhance the general quality of recycled products by minimizing pollution. Greater high-quality recyclables receive higher market values and can be reused more conveniently in industrial processes to form a complete circle in the circular economy. Also, a low level of contamination means fewer leads in cleaning and process which consumes lot of resources. This not only reduces costs but also lowers the environmental footprint of recycling operations. Improving sorting reduces the chances of equipment breakage that arise from processing non-recycles mistakenly through the machine. The result is shorter periods when machines are down hence cheaper maintenance cost plus steadier recycling operations. Therefore, these efficiencies help to create a sustainable recycling sector where larger amounts will be handled with less waste and lower impact on environment over time.

To put it in simple terms, better sorting leads to lesser contamination in recycling streams which means higher quality of recyclables, lesser operational expenses, more efficient processes, and bigger support for a circular economy. [14]

8.5 Resource Recovery:

As an industry continues to increase resource recovery rates, it demonstrates its effectiveness in reclaiming and reusing materials that would otherwise be wasted. This trend is essential for promoting sustainability and reducing environmental impact. Industries that have increased their resource recovery rates can greatly decrease their reliance on virgin materials, leading to a high efficiency in resource usage. For example, by recovering metals, plastics, and paper as materials, these manufacturers can reintroduce them back into the production cycle which consequently opens up the way for less consumption of new raw materials and, thus conserving natural resources as well as minimizing the degradation of the environment during extraction and processing activities.

Moreover, higher rates of resource recovery assist in lowering waste management expenses. As more and more materials are diversified from landfills as well as are reallocated elsewhere, waste disposal expenditure is reduced while extra income may be generated by selling recovered materials to businesses. This creates positive feedback since it can trigger investment into recovery technologies and systems that raise sustainability.

In addition, resource recovery rates spur innovation among sectors. To augment value derived from their wastes, companies accordingly make or take up various technologies and processes especially those that are efficient in utilizing resources sustainably. This could create new job opportunities or markets; hence contributing an economic benefit to the surroundings. Additionally, resources' retrievals can contribute to novel businesses and job options in the recycling and also resource management sectors, enhancing the financial and ecological advantages of dealing with recoveries.

All in all, the upsurge observed in industry's levels of recovery is an indication that, this stream is now going green. In the long run, it helps in minimizing harm to the environment, saving on costs and opening up avenues for economic development through new inventions and setting up of fresh markets.

8.6 Environmental Benefits:

The ecological benefits that arise from industries embracing sustainable practices, especially those focusing on waste reduction, are wide-ranging and significant. One of the greatest benefits has been reducing landfill usage. By diverting waste from landfills through recycling, composting, or reuse, an industry can bring down its waste levels in these sites considerably. Not only does this reduce the lifespan of existing landfills, but it also reduces the need for new ones that require too much land and often disrupt local ecosystems.

19

The other major ecological benefit is reducing greenhouse gas emissions. Landfills are among the major sources of methane which is a potent greenhouse gas that spells doom for our climate. The industries can help lower methane emissions by reducing organic matter going to dumpsites. Furthermore, sustainable practices such as incorporating

recycled materials into products, optimizing energy consumption as well as embracing cleaner technologies contribute to decreased carbon dioxide emissions thus helping combat global warming.

Thus, industries concentrated on minimizing waste and optimizing use of resources help in conserving nature's resources. This translates into less demand for the extraction of raw materials which are sometimes environmentally unfriendly like cutting down trees, extracting mineral ores and oil drilling activities, thereby saving more energy as well. By and large, this practice is responsible for reduced consumption of energy because it's always cheaper to recycle materials than to manufacture them from scratch.

In retrospect, it is evident that going green in manufacturing has far-reaching environmental implications. They include reduced landfill space usage, decrease in the release of dangerous greenhouse gases into the atmosphere, and reduced use of electric power generated from burning fossil fuels among others. Environmental protection and circular economy are intertwined in a way that these practices promote them both at the same time. [15]

Chapter 09

9.1 Monitoring Procedures for the Model:

20

Monitoring procedures for the models in your Smart Waste Sorting System project are essential to ensure that the system operates efficiently and effectively over time. Here are some key monitoring procedures you can implement:

1. Model Performance Tracking

4

- **Accuracy Monitoring:** Regularly assess the model's accuracy on both training and validation datasets to detect any performance degradation.
- **Loss Function Analysis:** Monitor the loss values during training and testing to ensure the model is converging as expected.
- **Confusion Matrix:** Use confusion matrices to track the model's performance in classifying different categories of waste and identify areas where the model might be struggling.

2. Real-Time Data Monitoring

15

- **Input Data Quality:** Continuously check the quality and consistency of the input data being fed into the model, ensuring that it matches the distribution of the training data.
- **Output Predictions:** Monitor the output predictions in real-time to detect any anomalies or incorrect classifications that may indicate a need for model retraining.

3. Model Drift Detection

- 13
- **Concept Drift:** Implement techniques to detect concept drift, where the statistical properties of the target variable change over time, affecting model accuracy.
 - **Retraining Triggers:** Establish thresholds for performance metrics that, when breached, will trigger model retraining or updates.

4. System Health Monitoring

- **Resource Utilization:** Monitor the computational resources (CPU, GPU, memory) used by the model to ensure that the system is operating within optimal parameters.
- **Latency Tracking:** Keep track of the time taken for each prediction to ensure the system meets real-time processing requirements.

5. Feedback Loop Implementation

- **User Feedback:** Collect feedback from system users to identify any misclassifications or errors in real-world usage, which can be used to refine and improve the model.
- **Continuous Learning:** Implement mechanisms for the model to learn from new data over time, adapting to any changes in the waste sorting environment.

6. Reporting and Alerts

- **Automated Reporting:** Set up automated reports summarizing model performance, including accuracy, loss, and any detected anomalies, to be reviewed periodically.
- **Alert Systems:** Implement an alert system to notify relevant stakeholders if the model's performance drops below a certain threshold or any critical issues are detected.

These procedures will help ensure that your Smart Waste Sorting System remains accurate, efficient, and reliable over time, adapting to changes in the environment and maintaining high-performance levels.

9.2 Risk Assessment of the Model:

Here is a risk assessment table tailored to the Smart Waste Sorting System project, incorporating both Generic Risks (GR) and Specific Risks (SR): [16]

Table 8: A table showing the risk assessment of the project

Risk No.	Description of the Risk	Probability of the Risk	Effect on the Project	Contingencies to Mitigate the Risk
GR1	Technical Risks	MEDIUM TO HIGH	Technical issues could delay the project, impacting model training and deployment, or result in reduced system accuracy.	Conduct regular technical reviews and validations during each phase of the project to identify and address potential issues early.
GR2	Data Quality Risks	MEDIUM	Poor data quality could lead to inaccurate model predictions and ineffective waste sorting.	Implement rigorous data preprocessing and augmentation techniques to enhance the quality and reliability of input data.
GR3	Resource Risks	MEDIUM	Insufficient computational resources or expertise may delay model training and system deployment.	Ensure access to adequate computing resources and allocate sufficient time for model training and validation.
GR4	Schedule Risks	MEDIUM	Delays in project phases could lead to missed deadlines and potential cost overruns.	Utilize project management tools to monitor progress and adjust resource allocation as

				needed to stay on schedule.
SR1	Model Performance Risk	HIGH	Inadequate model performance could result in poor classification accuracy, affecting the overall effectiveness of the waste sorting system.	Continuously monitor model performance using validation data and implement early stopping to prevent overfitting.
SR2	Algorithm Complexity Risk	MEDIUM TO HIGH	High algorithmic complexity might make the model difficult to interpret and maintain, increasing debugging and testing time.	Simplify model architecture where possible and modularize complex algorithms for easier troubleshooting and updates. 23
SR3	Data Drift Risk	MEDIUM	Changes in waste types or distributions over time could lead to a decrease in model accuracy.	Implement monitoring mechanisms to detect data drift and retrain models periodically with updated data.
SR4	Testing and Validation Risk	MEDIUM	Insufficient testing may result in unrecognized errors, leading to system failures during deployment.	Develop and execute a comprehensive testing plan that covers all possible scenarios, including edge cases.

SR5	Integration Risk	MEDIUM	Difficulty integrating the model with existing waste sorting systems may cause delays or reduced effectiveness. 7	Plan for integration early in the project, and conduct incremental testing to ensure compatibility with existing systems.
SR6	User Adoption Risk	LOW TO MEDIUM	Users may resist adopting the new system due to unfamiliarity or perceived complexity.	Provide comprehensive training and support for users, and involve them in the development process to ensure the system meets their needs.

This table 8 outlines potential risks, their probability, effects on the project, and the contingencies in place to mitigate them, ensuring that the Smart Waste Sorting System project is completed successfully and effectively.

Chapter 10

10.1 The work carried out to date:

How We Designed Our Project:

- An intelligent waste sorting system has been developed that relies on advanced Convolutional Neural Networks (CNNs) like ResNet50, MobileNet V3, EfficientNet B0, and MobileNet V2.
- The CNN models were trained on a dataset sourced from Kaggle with various types of recyclable, compostable and landfill waste images.
- We mounted a powerful Intel 13th Gen Core i5 processor together with MSI GeForce RTX 4060Ti Graphics Card for hardware-based computations.
- Adam Optimizer was used to classify images thus enhancing the accuracy as well as performance of the models.
- Lastly, we built a simulation environment which replicates the whole process of waste sorting.

Scenarios: Problems Encountered and How We Solved Them:

Scenario 1: Low Classification Accuracy for Certain Waste Types

Problem: Some waste categories (e.g., compostable items), during initial testing, were misclassified due to overlapping features between their classes.

Solution: Enhanced data augmentation techniques and fine-tuned the CNN models to better distinguish subtle differences between waste types. We also expanded our dataset with diverse examples increasing its generalization capacity.

Scenario 2: Computational Bottlenecks During Real-Time Sorting

Problem: When dealing with huge amounts of data in real-time, it resulted in delays leading to inefficient sorting speed of the system.

Solution: Our optimization of the codebase for parallel processing thus includes fully utilizing hardware capabilities like GPU acceleration which reduces processing time while thereby balancing accuracy and speed through our streamlining approach on the architecture of neural networks.

Scenario 3: Irregular outcomes under diverse lighting conditions:

The performance of the system was inconsistent while classifying waste based on images acquired in various light conditions affecting accuracy.

Measure: Pre-processing steps were taken to ensure that the lighting conditions in the images were uniform across the board before using them as input into CNN models. Moreover, the models were trained using datasets with all sorts of lighting conditions for better robustness.

Suggestions for Future Improvements Integration of Advanced Sensors:

Include multi-spectral imaging or depth sensors for an increased understanding of the waste items, thus improving accuracy during classification.

Example: In future versions, thermal sensors can be included so as to enable distinction between organic and inorganic waste by looking at temperature differences, thereby increasing sorting precision.

Scalability Improvements:

Create a distributed version of the system that allows for larger datasets and more intricate sorting needs to be executed in real-time over multiple nodes. For example, this could mean deploying the system across several processing units within an industrial setup so as to sort on a vast scale like it is done in municipal waste treatment facilities.

AI-Driven Continuous Learning:

Set up a self-learning mechanism where the system can perpetually refashion its models depending on new data, adapting to changing waste typologies and classification standards.

For instance, the model can recalibrate itself automatically by utilizing fresh labeled data from various areas at intervals thereby remaining aligned with local waste handling procedures.

10.2 Conclusion:

In conclusion, sustainable waste management has been made better through the Smart Waste Sorting System in which Machine Learning and Computer Vision principles are used. This project gives an effective way of classifying and sorting wastes by means of advanced CNN structures such as ResNet50, MobileNet V3, EfficientNet B0, and MobileNet V2. The system employs a powerful hardware setup coupled with the Adam Optimizer to guarantee precise classification of waste while it is done in real-time thereby mitigating the environmental effects associated with indiscriminate disposal of refuse. Hence, this project not only tackles current waste management problems, but it also paves way to a greener tomorrow by maximizing resource recovery rates, reducing pollution incidences in recycling streams as well as curbing greenhouse gas emissions. The Smart Waste Sorting System exemplifies how technology can be employed to come up with novel answers to serious environmental challenges as it is a scalable and flexible method for dealing with such issues.

References

- [1] "geeksforgeeks," 21 May 2024. [Online]. Available: <https://www.geeksforgeeks.org/image-classifier-using-cnn/>.
- [2] J. Brownlee, "Gentle Introduction to the Adam Optimization Algorithm for Deep Learning," *machinelearningmastery*, 13 january 2021. [Online].
- [3] A. Al Thunibat and S. A. , "Software Requirements Classification using Machine Learning algorithm's," July 2021. [Online].
- [4] S. Kunwar, "Garbage Dataset," kaggle, [Online]. Available: <https://www.kaggle.com/datasets/sumn2u/garbage-classification-v2/data>.
- [5] M. Mohamed, "Garbage Classification (12 classes)," kaggle, [Online]. Available: <https://www.kaggle.com/datasets/mostafaabla/garbage-classification>.
- [6] A. King, "Recyclable and Household Waste Classification," kaggle, [Online]. Available: <https://www.kaggle.com/datasets/alistairking/recyclable-and-household-waste-classification>.
- [7] "What is Data Annotation? – Definition, Key Benefits and Why You Need It," klippa, 16 May 2024. [Online]. Available: <https://www.klippa.com/en/blog/information/what-is-data-annotation/>.
- [8] E. Soylu, "Creating a CNN Model for Image Classification with TensorFlow," Medium, 21 March 2024. [Online]. Available: <https://medium.com/@esrasoylu/creating-a-cnn-model-for-image-classification-with-tensorflow-49b84be8c12a>.
- [9] T. SHARMA, "Detailed Explanation of Resnet CNN Model..," Medium, 6 March 2023. [Online]. Available: <https://medium.com/@sharma.tanish096/detailed-explanation-of-residual-network-resnet50-cnn-model-106e0ab9fa9e>.
- [10] S.-H. Tsang, "MobileNetV3: Searching for MobileNetV3 (Image Classification)," Medium, 15 November 2020. [Online]. Available: <https://sh-tsang.medium.com/paper-mobilenetv3-searching-for-mobilenetv3-image-classification-5072d4d8703c>.
- [11] "Efficientnet Architecture," geeksforgeeks, 03 June 2024. [Online]. Available: <https://www.geeksforgeeks.org/efficientnet-architecture/>.
- [12] "Computer Vision with PyTorch," geeksforgeeks, 28 March 2024. [Online]. Available: <https://www.geeksforgeeks.org/computer-vision-with-pytorch/>.
- [13] Virgiawan, "12 Classes Garbage Classification (Improvement)," kaggle, [Online]. Available: <https://www.kaggle.com/datasets/virgiawan/12-classes-garbage-classification>.
- [14] I. Zelenika, T. Moreau and J. Zhao, "Toward zero waste events: Reducing contamination in waste streams with volunteer assistance," *sciencedirect*, vol. 76, pp. 39-45, 2018.
- [15] J. B. A.J Morrissey, "Waste management models and their application to sustainable waste management," *sciencedirect*, vol. 24, no. 3, pp. 297-308, 2004.
- [16] H. Kirk, "Prediction Versus Management Models Relevant to Risk Assessment: The Importance of Legal Decision-Making Context," *taylorfrancis*, p. 13, 2007.

Report-Smart Industrial Waste Sorting Systems

ORIGINALITY REPORT



PRIMARY SOURCES

- | Rank | Source | Category | Percentage |
|------|--|-----------------|------------|
| 1 | Syed Nisar Hussain Bukhari. "Data-Driven Farming - Harnessing the Power of AI and Machine Learning in Agriculture", CRC Press, 2024 | Publication | 1% |
| 2 | www.mdpi.com | Internet Source | 1% |
| 3 | Submitted to University of Edinburgh | Student Paper | 1% |
| 4 | Prasenjit Dey, Sudip Kumar Adhikari, Sourav De, Indrajit Kar. "Internet of Things-Based Machine Learning in Healthcare - Technology and Applications", CRC Press, 2024 | Publication | <1% |
| 5 | Mehdi Ghayoumi. "Generative Adversarial Networks in Practice", CRC Press, 2023 | Publication | <1% |
| 6 | Amir Shachar. "Introduction to Algogens", Open Science Framework, 2024 | Publication | <1% |

- 7 Submitted to University of Wales Institute, Cardiff <1 %
Student Paper
-
- 8 Submitted to SASTRA University <1 %
Student Paper
-
- 9 Submitted to University of Glamorgan <1 %
Student Paper
-
- 10 Osamah Abdulrahman Almansari, Nik Nur Wahidah Nik Hashim. "Recognition of Isolated Handwritten Arabic Characters", 2019 7th International Conference on Mechatronics Engineering (ICOM), 2019 <1 %
Publication
-
- 11 Submitted to Vanderbilt University <1 %
Student Paper
-
- 12 assets.researchsquare.com <1 %
Internet Source
-
- 13 Submitted to Liverpool John Moores University <1 %
Student Paper
-
- 14 www.ijert.org <1 %
Internet Source
-
- 15 "Smart Trends in Computing and Communications", Springer Science and Business Media LLC, 2024 <1 %
Publication

16

Submitted to City University of Hong Kong

Student Paper

<1 %

17

Ben Othman Soufiene, Chinmay Chakraborty.
"Machine Learning and Deep Learning
Techniques for Medical Image Recognition",
CRC Press, 2023

Publication

<1 %

18

Hoai Nam Vu, Huong Mai Nguyen, Cuong Duc
Pham, Anh Dat Tran, Khanh Nguyen Trong,
Cuong Pham, Viet Hung Nguyen. "Landslide
Detection with Unmanned Aerial Vehicles",
2021 International Conference on Multimedia
Analysis and Pattern Recognition (MAPR),
2021

Publication

<1 %

19

www.change.org

Internet Source

<1 %

20

Submitted to Midlands State University

Student Paper

<1 %

21

Submitted to National College of Ireland

Student Paper

<1 %

22

www.ncbi.nlm.nih.gov

Internet Source

<1 %

23

Submitted to Instituto de Empress S.L.

Student Paper

<1 %

www.ijraset.com

- Internet Source
- 24 [www.nature.com](#) <1 %
-
- 25 [ijisrt.com](#) Internet Source <1 %
-
- 26 Barbara Kunda, Jackson Phiri. "The Use of Machine Learning in Industry 4.0 as an Educational Tool to Address Plant Diseases for Small-Scale Farmers in Developing Countries: A Review Study", Zambia ICT Journal, 2023 <1 %
- Publication
-
- 27 Mohamed Ali Setitra, Mingyu Fan, Bless Lord Y. Agbley, Zine El Abidine Bensalem. "Optimized MLP-CNN Model to Enhance Detecting DDoS Attacks in SDN Environment", Network, 2023 <1 %
- Publication
-
- 28 [www.amazon.pl](#) Internet Source <1 %
-
- 29 Alex Hultin, Manh-Toan Ho. "Sustaining Interdisciplinary Research: A Multilayer Perspective", Open Science Framework, 2019 <1 %
- Publication

- 31 Rong Liu, Vinay Vakharia. "Optimizing Supply Chain Management Through BO-CNN-LSTM for Demand Forecasting and Inventory Management", Journal of Organizational and End User Computing, 2024
Publication <1 %
- 32 Ton Duc Thang University <1 %
Publication
- 33 dev.to <1 %
Internet Source
- 34 www.amazon.com.au <1 %
Internet Source
- 35 www.irjet.net <1 %
Internet Source
- 36 Ashok Kumar, Geeta Sharma, Anil Sharma, Pooja Chopra, Punam Rattan. "Advances in Networks, Intelligence and Computing - International Conference on Networks, Intelligence and Computing (ICONIC-2023)", CRC Press, 2024 <1 %
Publication
- 37 Yu-Ju Lin, H.A. Latchman, Minkyu Lee, S. Katar. "A power line communication network infrastructure for the smart home", IEEE Wireless Communications, 2002 <1 %
Publication

Exclude quotes On

Exclude bibliography On

Exclude matches Off