Garbage Classification System for Waste Sorting Using Deep Learning

Ahmet Yasin Aytar
DSAI 544 - Computer Vision
Boğaziçi University
2025/2026-1

January 12, 2026

Abstract

This project presents an image classification system for automated waste sorting that categorizes items into six classes: cardboard, glass, metal, paper, plastic, and trash. The project demonstrates three major components as required by the course: **Dataset Creation** through web scraping to augment an existing Kaggle dataset, **Auto-Labeling** using CLIP (Contrastive Language-Image Pre-training) for zero-shot classification of newly collected images, and **Model Fine-Tuning** of a pretrained MobileNetV2 architecture. The final model achieves 90.64% accuracy on the test set, exceeding the target performance of 85%. All experiments were tracked using Weights & Biases for reproducibility and comprehensive analysis.

1 Introduction

Effective waste management is a critical environmental challenge worldwide. Proper sorting of recyclable materials from general waste can significantly reduce landfill usage and promote sustainable resource utilization. However, manual waste sorting is labor-intensive, error-prone, and often impractical at scale. Computer vision offers a promising solution by enabling automated classification of waste items.

This project addresses the waste sorting problem by developing a deep learning-based image classification system. The system classifies waste images into six categories: cardboard, glass, metal, paper, plastic, and trash (non-recyclable). These categories reflect common recycling standards and provide practical utility for real-world waste management applications.

The project encompasses three substantial components aligned with the course requirements:

1. **Dataset Creation**: The original Kaggle dataset contained class imbalance, particularly for the trash category. We augmented the dataset by scraping additional images from the web using the icrawler library, targeting underrepresented classes.

2. **Auto-Labeling**: Rather than manually labeling scraped images, we employed CLIP, a state-of-the-art vision-language model, to automatically classify and label images using zero-shot learning. A confidence threshold was applied to ensure label quality.

3. **Model Fine-Tuning**: We fine-tuned a pretrained MobileNetV2 model on the combined dataset, implementing data augmentation, learning rate scheduling, and early stopping strategies.

2 Methods and Results

2.1 Dataset Preparation

2.1.1 Original Dataset

The base dataset was obtained from Kaggle[1], containing 2,527 images across six classes. Table 1 shows the original class distribution, revealing significant imbalance with the trash class having only 137 images compared to 594 for paper.

Table 1: Original Kaggle Dataset Distribution

Class	Images	Percentage
Cardboard	403	15.9%
Glass	501	19.8%
Metal	410	16.2%
Paper	594	23.5%
Plastic	482	19.1%
Trash	137	5.4%
Total	**2,527**	**100%**

2.1.2 Dataset Creation via Web Scraping

To address the class imbalance and expand the dataset, we implemented a web scraping pipeline using the icrawler library with Bing Image Search. For each class, we designed specific search queries to capture diverse representations of waste items:

- Cardboard: "cardboard waste", "cardboard box garbage", "recyclable cardboard"
- Glass: "glass bottle waste", "broken glass garbage", "glass recycling"
- Metal: "metal can waste", "aluminum can garbage", "metal recycling"
- Paper: "paper waste", "newspaper garbage", "paper recycling"
- Plastic: "plastic bottle waste", "plastic container garbage", "plastic recycling"
- Trash: "general waste", "non-recyclable garbage", "landfill waste", "mixed garbage"

We scraped approximately 250 images in total, with 75 images targeted for the underrepresented trash class and 35 images for each of the other classes.

2.1.3 Auto-Labeling with CLIP

Since web-scraped images may not always match their search query labels, we implemented an auto-labeling pipeline using OpenAI's CLIP model (ViT-B/32). CLIP performs zero-shot classification by computing similarity scores between image embeddings and text embeddings of class descriptions.

For each scraped image, we computed similarity scores against six text prompts (e.g., "a photo of cardboard", "a photo of a glass bottle"). The class with the highest similarity score was assigned as the label, but only if the confidence exceeded a threshold of 0.70. Images below this threshold were rejected to maintain dataset quality.

[1] https://www.kaggle.com/datasets/asdasdasasdas/garbage-classification

Table 2: Auto-Labeling Results with CLIP

Class	Scraped	Accepted	Acceptance Rate
Cardboard	35	20	57.1%
Glass	35	12	34.3%
Metal	35	11	31.4%
Paper	35	13	37.1%
Plastic	35	51	145.7%*
Trash	75	37	49.3%
Total	**250**	**144**	**57.6%**

* Some images originally scraped for other classes were reclassified as plastic by CLIP.

The auto-labeling process revealed interesting patterns. CLIP showed high confidence for clear, single-object images but lower confidence for cluttered scenes. The mean confidence score across all predictions was 0.732, with plastic achieving the highest mean confidence (0.724 for accepted images) while trash showed more variability.

2.1.4 Final Dataset

The final dataset combined the original Kaggle images with CLIP-verified auto-labeled images, totaling 2,671 images. We applied stratified splitting to maintain class proportions across train (70%), validation (15%), and test (15%) sets.

Table 3: Final Dataset Distribution Across Splits

Class	Train	Validation	Test	Total
Cardboard	296	63	64	423
Glass	359	76	78	513
Metal	294	63	64	421
Paper	424	91	92	607
Plastic	373	79	81	533
Trash	121	26	27	174
Total	**1,867**	**398**	**406**	**2,671**

2.2 Model Architecture

We selected MobileNetV2 as the backbone architecture due to its efficiency and strong performance on image classification tasks. MobileNetV2 employs depthwise separable convolutions and inverted residual blocks, making it suitable for deployment scenarios while maintaining competitive accuracy.

The model was initialized with ImageNet pretrained weights to leverage transfer learning. The final classification layer was replaced with a new fully connected layer mapping from 1,280 features to 6 output classes. The total model contains 2,231,558 parameters, all of which were fine-tuned during training.

2.3 Training Configuration

2.3.1 Data Augmentation

To improve model generalization and prevent overfitting, we applied the following augmentations during training:

- Random resized crop to 224×224 pixels (scale: 0.8–1.0)
- Random rotation (±15 degrees)
- Random horizontal flip (probability: 0.5)
- Color jitter (brightness, contrast, saturation: 0.2)
- ImageNet normalization (mean: [0.485, 0.456, 0.406], std: [0.229, 0.224, 0.225])

For validation and testing, images were resized to 256×256, center-cropped to 224×224, and normalized.

2.3.2 Training Hyperparameters

Table 4: Training Hyperparameters

Parameter	**Value**
Optimizer	Adam
Initial Learning Rate	0.001
Weight Decay	0.0001
Batch Size	32
Maximum Epochs	30
LR Scheduler	ReduceLROnPlateau
Scheduler Patience	3 epochs
Scheduler Factor	0.1
Early Stopping Patience	5 epochs
Loss Function	Cross-Entropy
Device	Apple M3 (MPS)

2.4 Training Results

Training was conducted for 24 epochs before early stopping was triggered. All metrics were logged to Weights & Biases for comprehensive tracking and visualization.

2.4.1 Training Curves

Figure 1 shows the training and validation metrics over epochs. The training loss decreased consistently from 0.96 to 0.04, while training accuracy improved from 66% to approximately 98%. The validation accuracy plateaued around 88%, with the best validation accuracy of 88.69% achieved at epoch 19.

The learning rate scheduler reduced the learning rate from 10^{-3} to 10^{-4} around epoch 12 when validation performance plateaued, and further to 10^{-5} around epoch 20. This adaptive learning rate strategy helped the model converge to better optima.

Figure 1: Training curves from Weights & Biases showing (a) validation loss, (b) validation accuracy, (c) training loss, (d) training accuracy, (e) learning rate schedule, and (f) epoch progression. The green line represents the successful training run.

2.5 Evaluation Results

The final model was evaluated on the held-out test set of 406 images, achieving an overall accuracy of **90.64%**, which exceeds the target of 85%. Note that test accuracy exceeding validation accuracy (88.69%) is expected due to variance in stratified splits and the relatively small validation set size.

2.5.1 Per-Class Performance

Table 5 presents the detailed classification metrics for each class.

Table 5: Per-Class Classification Results on Test Set

Class	Precision	Recall	F1-Score	Support
Cardboard	0.910	0.953	0.931	64
Glass	0.944	0.872	0.907	78
Metal	0.863	0.984	0.920	64
Paper	0.899	0.967	0.932	92
Plastic	0.944	0.827	0.882	81
Trash	0.833	0.741	0.784	27
Macro Avg	0.899	0.891	0.893	406
Weighted Avg	0.908	0.906	0.905	406

The model performs well across most classes, with F1-scores above 0.88 for cardboard, glass, metal, paper, and plastic. The trash class shows relatively lower performance (F1: 0.784), which is expected given its smaller representation in the dataset and the inherent ambiguity of non-recyclable waste items.

2.5.2 Confusion Matrix Analysis

Figure 2 shows the confusion matrix for the test set predictions. The diagonal elements indicate correct classifications, while off-diagonal elements represent misclassifications.

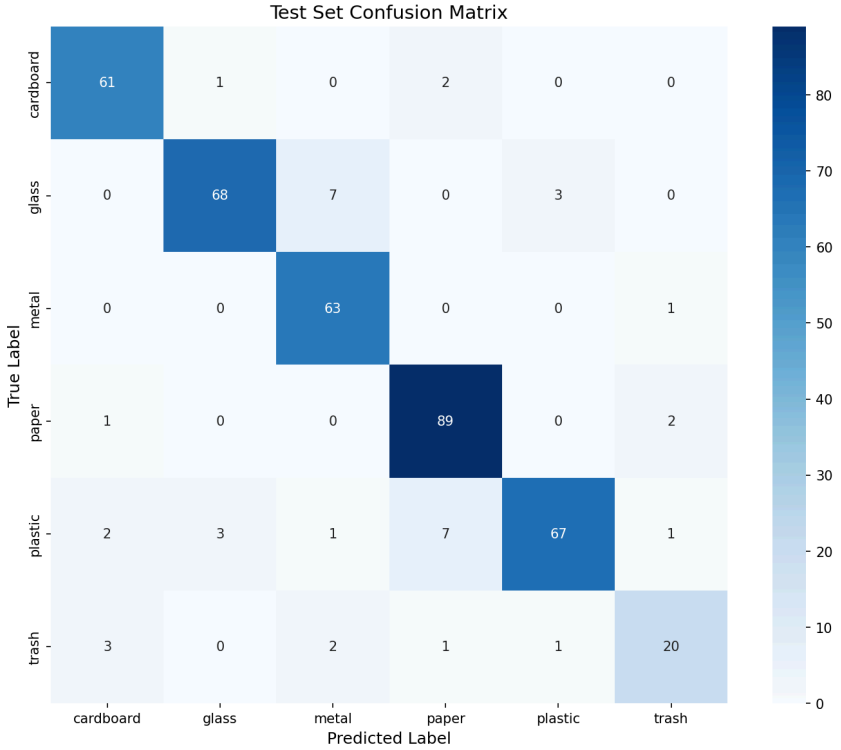


Figure 2: Confusion matrix showing classification results on the test set. The model achieves strong diagonal dominance, indicating accurate predictions across all classes.

Key observations from the confusion matrix:

- Glass is occasionally confused with metal (7 instances), likely due to similar reflective properties.
- Plastic is sometimes misclassified as paper (7 instances), possibly due to similar colors or textures in certain items.
- Trash shows the most confusion, being misclassified as cardboard (3 instances) and metal (2 instances), reflecting the heterogeneous nature of non-recyclable waste.

2.5.3 Per-Class F1-Score Visualization

Figure 3 provides a visual comparison of F1-scores across all classes, highlighting the relative performance differences.

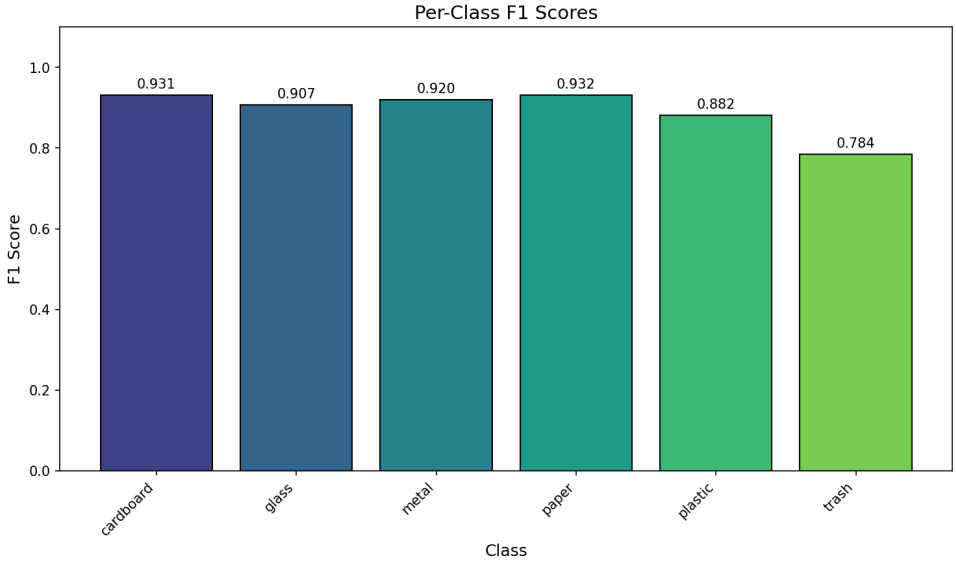
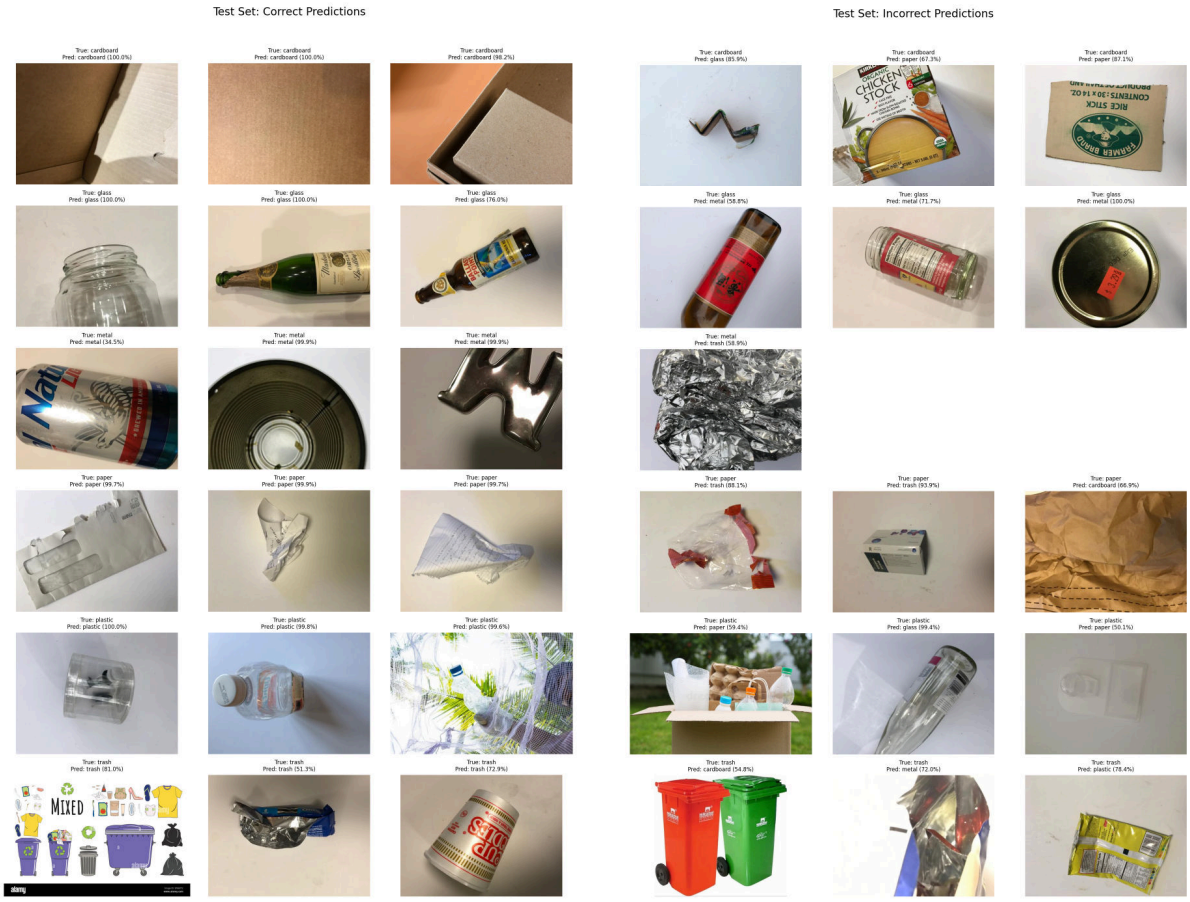


Figure 3: Per-class F1-scores on the test set. Paper and cardboard achieve the highest scores, while trash shows the lowest performance due to class imbalance and category ambiguity.

2.5.4 Sample Predictions

Figure 4 shows example predictions from the test set, including both correct and incorrect classifications. These visualizations help illustrate the model's decision-making patterns and failure modes.



(a) Correct predictions (b) Incorrect predictions

Figure 4: Sample predictions from the test set showing (a) correctly classified images and (b) misclassified images with their true and predicted labels.

3 Conclusion

This project successfully developed a garbage classification system for automated waste sorting, achieving 90.64% accuracy on the test set. The project demonstrated three substantial components: **Dataset Creation** through web scraping, **Auto-Labeling** using CLIP zero-shot classification, and **Model Fine-Tuning** of a pretrained MobileNetV2 architecture.

Key findings from this project include:

1. **Effectiveness of Transfer Learning**: Fine-tuning a pretrained MobileNetV2 model proved highly effective, achieving strong performance with relatively limited training data (1,867 training images).

2. **CLIP for Auto-Labeling**: CLIP demonstrated reasonable capability for automatic labeling of web-scraped images. The 0.70 confidence threshold balanced label quality against dataset expansion, accepting 57.6% of scraped images.

3. **Class Imbalance Challenges**: The trash class remained the most challenging due to its smaller size and inherent variability. Future work could address this through class-weighted loss functions or more targeted data collection.

4. **Training Best Practices**: Learning rate scheduling and early stopping proved essential for achieving good generalization while preventing overfitting.

3.1 Future Work

Several directions could improve upon this work:

- **Class-Weighted Loss**: Implementing weighted cross-entropy to address class imbalance, particularly for the trash category.
- **Advanced Augmentation**: Using techniques like MixUp, CutMix, or AutoAugment to further improve generalization.
- **Ensemble Methods**: Combining multiple models or using test-time augmentation to boost prediction confidence.
- **Larger Backbones**: Experimenting with larger architectures (e.g., EfficientNet, ConvNeXt) for potentially higher accuracy.
- **Human-in-the-Loop Labeling**: Reviewing and correcting auto-labeled images to improve dataset quality.

3.2 Reproducibility

All code and experiments are available in the project repository. Training metrics are logged to Weights & Biases and can be accessed at:

https://wandb.ai/aytarahmetyasin-bo-azi-i-niversitesi/garbage-classification

The random seed (42) was fixed throughout all experiments to ensure reproducibility.

References

[1] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4510–4520).

[2] Radford, A., Kim, J. W., Hallacy, C., et al. (2021). Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning* (pp. 8748–8763).

[3] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770–778).