

Article

A Multimodal Vision: Language Framework for Intelligent Detection and Semantic Interpretation of Urban Waste

Verda Misimi Jonuzi ^{1,*}  and Igor Mishkovski ²¹ Faculty of Natural Sciences and Mathematics, University of Tetovo, 1200 Tetovo, North Macedonia² Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, 1000 Skopje, North Macedonia; igor.miskovski@gmail.com

* Correspondence: verdamisimi@gmail.com

Abstract

Urban waste management remains a significant challenge for achieving environmental sustainability and advancing smart city infrastructures. This study proposes a multimodal vision–language framework that integrates real-time object detection with automated semantic interpretation and structured semantic analysis for intelligent urban waste monitoring. A custom dataset including 2247 manually annotated images was constructed from publicly available sources (TrashNet and TACO), enabling robust multi-class detection across six waste categories. Two state-of-the-art object detection models, YOLOv8m and YOLOv10m, were trained and evaluated using a fixed 70/15/15 train–validation–test split. Under this configuration, YOLOv8m achieved a mAP@50 of 90.5% and a mAP@50–95 of 87.1%, slightly outperforming YOLOv10m (89.5% and 86.0%, respectively). Moreover, YOLOv8m demonstrated superior inference efficiency, reaching 120 FPS compared to 105 FPS for YOLOv10m. To obtain a more reliable estimate of performance stability across data partitions, stratified 5-Fold Cross-Validation was conducted. YOLOv8m achieved an average Precision of 0.9324 and an average mAP@50–95 of 0.9315 ± 0.0575 across folds, suggesting generally stable performance across data partitions, while also revealing variability associated with dataset heterogeneity. Beyond object detection, the framework integrates MiniGPT-4 to generate context-aware textual descriptions of detected waste items, thereby enhancing semantic interpretability and user engagement. Furthermore, GPT-5 Vision is incorporated as a structured auxiliary semantic classification and category-suggestion module that analyzes object crops and multi-class scenes, producing constrained JSON-formatted outputs that include category labels, concise descriptions, and recyclability indicators. Overall, the proposed YOLOv8–MiniGPT-4–GPT-5 Vision pipeline shows that combining accurate real-time detection with multimodal semantic reasoning can improve interpretability and support interactive, semantically enriched waste analysis in smart-city and environmental monitoring scenarios.



Academic Editors: Antony Bryant and Manuel Pedro Rodríguez Bolívar

Received: 1 January 2026

Revised: 22 March 2026

Accepted: 26 March 2026

Published: 3 April 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.**Keywords:** intelligent waste management; YOLOv8; YOLOv10; Stratified K-Fold; GPT-5 Vision; MiniGPT-4; multimodal learning; computer vision; LLMs; smart cities; sustainability

1. Introduction

Rapid urbanization and population growth have led to a substantial increase in municipal solid waste generation, posing significant challenges for sustainable waste management systems worldwide. Global projections estimate that annual municipal waste production will reach approximately 3.4 billion tons by 2050, placing increasing pressure

on urban infrastructure, natural resources, and environmental sustainability. In addition, inefficient waste handling contributes to pollution and public health risks, particularly in densely populated urban environments.

Traditional waste management systems, which largely rely on manual monitoring, static scheduling, and limited data integration, are increasingly inadequate for addressing the scale and complexity of modern cities. These systems often lack real-time awareness of waste distribution and composition, leading to inefficient collection processes and suboptimal resource utilization [1–3].

To address these limitations, recent research has focused on intelligent waste management systems within smart city frameworks. These approaches leverage Internet of Things (IoT) technologies, sensor networks, and data-driven optimization techniques to improve collection efficiency, reduce operational costs, and enable dynamic decision-making [2,4–7]. However, most existing solutions primarily emphasize logistical optimization and provide limited capabilities for real-time waste identification and semantic understanding, which are essential for automation and informed decision-making.

In parallel, advances in artificial intelligence and computer vision have enabled automated waste detection and classification using deep learning models. Object detection architectures such as YOLO (You Only Look Once) have gained widespread adoption due to their ability to achieve high accuracy while maintaining real-time performance [8–11]. Recent variants, including YOLOv8 and YOLOv10, further improve detection robustness and efficiency in complex and dynamic environments. Despite these advances, most vision-based systems remain limited to predicting object labels and bounding boxes, without providing higher-level semantic interpretation or contextual reasoning.

More recently, the emergence of large language models (LLMs) and vision–language models (VLMs) has introduced new opportunities to bridge visual perception and semantic reasoning. Models such as GPT-4 and MiniGPT-4 enable the generation of natural language descriptions from visual inputs, facilitating richer interaction, interpretability, and knowledge extraction [12–15]. These multimodal systems extend beyond visual recognition by adding contextual understanding and semantic reasoning capabilities. Nevertheless, their integration into real-time waste management systems remains limited, particularly in unified frameworks that combine detection with semantic interpretation.

Despite these advancements, a critical gap persists: existing approaches tend to operate either at the level of logistics optimization or visual classification, without integrating semantic reasoning capabilities that support interpretability, recyclability assessment, and user-oriented explanations [14,16]. This limitation restricts the practical usability of intelligent waste management systems in real-world applications, where both accurate perception and meaningful communication are required.

To address this gap, this study proposes a multimodal vision–language framework for intelligent urban waste analysis. The proposed system integrates three complementary components within a unified pipeline:

- (i) Real-time object detection using YOLOv8m/YOLOv10m;
- (ii) Automated description generation through MiniGPT-4 [12,13];
- (iii) Structured semantic classification and category suggestion using GPT-5 Vision.

Unlike existing approaches, the proposed approach extends beyond detection by incorporating context-aware reasoning, enabling not only accurate localization and classification of waste items but also semantic explanation and recyclability assessment.

Furthermore, this work contributes a manually annotated dataset. It also introduces a comprehensive evaluation protocol based on fixed data splits and Stratified 5-Fold Cross-Validation, ensuring methodological rigor and reproducibility. By bridging computer vision with multimodal semantic reasoning, the proposed framework advances traditional

detection systems toward more interactive, explainable, and scalable solutions for smart city applications.

To the best of our knowledge, only limited prior work has combined real-time waste detection, multimodal description generation, and structured semantic reasoning within a single unified pipeline. The proposed approach highlights the potential of combining perception and reasoning in intelligent systems designed for real-world deployment [14,16].

2. Literature Review

Urban waste management has emerged as one of the most pressing challenges for 21st-century cities. The World Bank report “What a Waste 2.0” emphasizes that urbanization, economic development, and changing consumption patterns have increased global waste generation, placing extraordinary pressure on infrastructure and natural resources [17]. Similar challenges are reported in developing countries, where traditional waste management systems remain inadequate and inefficient [9].

To address these challenges, the adoption of technologies such as the Internet of Things (IoT) and smart sensors has increased significantly. Several studies propose comprehensive IoT-based waste monitoring systems that optimize collection processes [4,5]. Such systems enable fill-level prediction, dynamic route planning, and reduction in operational costs.

The increase in computing power and the advancement of artificial intelligence (AI) have significantly advanced object detection in images and videos. YOLO algorithms have been particularly successful due to their speed and accuracy [10,11].

For automatic waste classification, YOLO has been successfully tested in various studies. For example, one study applied YOLOv5s to the TrashNet dataset, achieving a mean Average Precision (mAP) of 95.2%, while another study [18] employed YOLOv8 for the classification of 17 categories of household waste, achieving an mAP of 89.5%.

More recent research has explored the integration of computer vision with large language models (LLMs), such as GPT-4, to enhance the semantic description and interpretation of detection outputs. Studies such as [7,19,20] introduce MiniGPT-4, which integrates BLIP-2 with Vicuna-7B to generate detailed image descriptions from visual inputs. The use of drones, robotics, and multi-agent networks is also increasing for the monitoring, collection, and automatic classification of waste in large cities [7,19].

Overall, the integration of IoT, AI-driven detection models, and multimodal language systems holds significant potential for improving efficiency, transparency, and sustainability in urban waste management.

In practice, prior work can be broadly categorized into two main directions: (i) IoT- and sensor-driven systems that optimize monitoring and collection logistics (e.g., fill-level prediction and route planning) [4,5,7], and (ii) vision-based approaches that rely on fast one-stage detectors, such as YOLO variants, to produce object labels and bounding boxes for waste detection and classification [8,10,11], including recent domain-specific applications [18]. Although these approaches achieve strong detection accuracy, they typically provide limited user-oriented semantic interpretation beyond simple “label + bounding box” outputs. Recent vision–language models demonstrate the capability to generate richer image descriptions and contextual explanations from visual data [12,13,15]. In contrast, our approach integrates real-time detection (YOLOv8/YOLOv10), a description module (MiniGPT-4), and an auxiliary semantic classification and category-suggestion module (GPT-5 Vision). This unified framework produces interpretable outputs—including contextual descriptions and recyclability cues—and provides additional semantic category suggestions in complex or multiclass scenarios.

3. Materials and Methods

3.1. Dataset Construction and Annotation

For waste detection, the publicly available TrashNet and TACO datasets were examined as primary image sources. TrashNet contains approximately 2527 images categorized into six waste classes: paper, plastic, metal, glass, cardboard, and trash [21]. The TACO (Trash Annotations in Context) dataset includes approximately 1500 real-world images with 4784 annotations designed for waste detection and segmentation tasks [20]. To ensure robustness and generalization, the dataset was constructed by combining images from TrashNet and TACO to balance controlled and in-the-wild scenarios. TrashNet provides clean, well-centered objects under relatively simple backgrounds, while TACO introduces real-world complexity such as clutter, occlusion, and varying object scales. This combination was intentionally selected to improve model robustness and generalization across both laboratory-like and real-world conditions.

The choice of six waste categories (cardboard, glass, metal, paper, plastic, trash) reflects a compromise between practical recyclability relevance and sufficient class representation within the available datasets. A larger taxonomy could introduce class imbalance and reduce detection reliability, given the dataset size.

Based on these sources, we constructed a custom object detection dataset by selecting 2247 images and manually annotating them using LabelImg. Bounding boxes were drawn for each visible waste item, and annotations were exported in YOLO format. The final dataset includes six waste categories—cardboard, glass, metal, paper, plastic, and trash—enabling multi-class detection and classification.

The complete dataset ($N = 2247$ images) was divided into training (70%), validation (15%), and test (15%) subsets using stratified sampling to preserve class distribution across all splits. This ensured balanced representation of each category during model training and evaluation.

3.2. Object Detection Models

Two object detection models were trained and compared: YOLOv8m and YOLOv10m. Both models were trained for 100 epochs using a batch size of 8, an image size of 640 pixels, the Adam optimizer, and COCO pre-trained weights for transfer learning.

All models were trained for the full 100 epochs without early stopping to ensure consistent comparison. The best-performing weights were selected based on validation performance. Training and validation were performed using the Ultralytics YOLO framework on a Tesla T4 GPU. Transfer learning was applied by initializing the models with pre-trained weights and fine-tuning them on the custom waste dataset.

Performance evaluation focused on Mean Average Precision (mAP), Precision, Recall, training convergence behaviour, and inference speed (FPS and latency).

YOLOv8m and YOLOv10m were selected to represent two recent state-of-the-art real-time object detection architectures with different design optimizations. YOLOv8m is known for its strong balance between accuracy and inference speed, making it suitable for real-time applications, while YOLOv10m introduces architectural refinements aimed at improving efficiency and detection performance.

The use of both models enables a controlled comparative analysis under identical training conditions, allowing us to evaluate whether newer architectural improvements translate into practical gains for the waste detection domain. This comparative design strengthens the interpretability of the experimental findings.

3.3. Evaluation Protocol

3.3.1. Object Detection Evaluation

Object detection performance was assessed under two complementary evaluation configurations:

1. Fixed 70/15/15 Split:

Standard train–validation–test evaluation was conducted using mAP@50, mAP@50–95, Precision, and Recall.

2. Stratified 5-Fold Cross-Validation:

A Stratified 5-Fold Cross-Validation procedure was implemented to assess performance stability across alternative data partitions and to quantify fold-wise variability. Each fold preserved class proportions, and the model was trained and evaluated across five distinct splits. Final performance metrics were computed as the average across folds.

Evaluation design rationale. To ensure a reliable and unbiased assessment of model performance, Stratified 5-Fold Cross-Validation preserves class distribution across all folds and mitigates variability associated with a single train–test split. Each fold was used once as a validation set while the remaining folds were used for training.

Stratified 5-Fold Cross-Validation was performed on the merged dataset and serves as an internal robustness and stability analysis procedure rather than an evaluation on an independent external test set. The fixed 70/15/15 split remains the primary benchmark configuration, while cross-validation provides variance-aware performance estimation.

3.3.2. MiniGPT-4 Description Module

In addition to object detection with YOLOv8 and YOLOv10, each detected object in an image was automatically described using the MiniGPT-4 model (Vicuna-7B). This LLM was integrated into the pipeline after the detection phase to generate customized textual descriptions for each object category.

Pipeline description. The proposed system operates as a sequential multimodal pipeline. First, YOLO detects and localizes waste objects within the input image. The corresponding image regions are then cropped and passed to the MiniGPT-4 model, which generates context-aware textual descriptions. For each detected object, the system produces an informative explanation that includes material composition, potential uses, recycling processes, and environmental relevance. A visual representation of the pipeline is shown in Figure 1.

Despite the effectiveness of the proposed pipeline, several limitations exist. First, the accuracy of the system depends directly on the performance of the object detector (YOLOv8m) and the language model (MiniGPT-4). Inaccuracies in object detection (such as missed detections or misclassifications) directly affect the relevance of the generated descriptions. Furthermore, the language model may occasionally produce overly generic or not entirely accurate descriptions, particularly when confronted with unfamiliar objects or ambiguous categories. The performance of the system may also be constrained by hardware resources, especially when deep models are executed on devices with limited computational capacity. Finally, since MiniGPT-4 is not specifically trained for recycling-related descriptions, the quality of its output may vary depending on the prompt and the context.

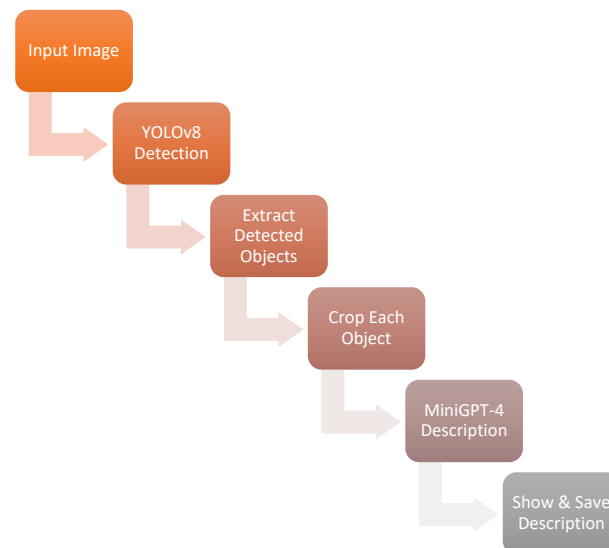


Figure 1. System pipeline for automatic waste object detection and description.

3.3.3. GPT-5 Vision Semantic Module

To further enhance semantic interpretability, GPT-5 Vision was integrated as an auxiliary semantic classification and category-suggestion module. After YOLOv8m or YOLOv10m detected and categorized the waste objects, the cropped regions of each detected object were re-evaluated using GPT-5 Vision. This secondary semantic analysis step served as an auxiliary category-suggestion module, complementing the category predicted by YOLO and providing contextual insights based on the object's visual and semantic features. This hybrid approach improved semantic interpretability and provided an auxiliary category-level check, particularly for visually similar waste categories such as plastic versus cardboard.

Additionally, experiments were conducted using multiclass images containing multiple waste categories in a single scene. These tests aimed to evaluate the system's ability to detect and distinguish overlapping or co-occurring waste types. GPT-5 Vision was employed as a semantic analysis mechanism in this stage, analyzing the full image context to provide scene-level semantic interpretation and category suggestions for multiple objects present in the image, while also allowing comparison with the textual descriptions generated by MiniGPT-4. This multimodal semantic analysis was intended to improve the interpretability of results in complex scenarios and provided auxiliary category suggestions for multiclass scenes.

3.4. Implementation Details of GPT-5 Vision

GPT-5 Vision is incorporated as an auxiliary semantic analysis module rather than a primary detection component. This design choice is motivated by the need to complement the strengths of object detection models, which provide accurate localization and classification, with higher-level semantic interpretation and contextual reasoning.

This separation of roles avoids overloading a single model with multiple tasks and allows each component to operate within its strengths: YOLO for spatial detection and GPT-5 Vision for semantic enrichment and category suggestion. This modular design also improves interpretability and facilitates future extensibility of the system.

GPT-5 Vision was used via API calls (inference-only) as an auxiliary semantic classification and category-suggestion module, and no fine-tuning was performed. In the proposed multimodal pipeline, GPT-5 Vision does not replace the object detector; instead, it

is used to semantically analyze the content of (i) YOLO object crops (one crop per predicted bounding box) and (ii) full-scene images in multiclass scenarios.

The objective is to improve semantic interpretability and reduce errors in visually similar categories by returning structured outputs that include the predicted waste category, a short description, and recyclability information. The experiments were conducted on the validation subset of the dataset using the GPT-5 Vision API under an inference-only setting.

Input preparation. After YOLO detection, each predicted bounding box was used to crop the corresponding region from the original image (RGB). For multiclass experiments, the entire image was provided to GPT-5 Vision to use global scene context. Prior to submission, images (crops or full scenes) were encoded in Base64 and embedded in the API request payload.

API request configuration. To reduce response variability and support reproducibility, we used a fixed inference configuration for all calls. Unless otherwise stated, we used low-temperature decoding (temperature = 0.1) to encourage deterministic, structured outputs and bounded response length (max_tokens = 256) to keep responses concise and consistently formatted.

Structured prompting and output schema. The model was instructed to return JSON-only outputs (no markdown, no extra text) using a constrained set of categories. Enforcing a strict schema and validating outputs programmatically is aligned with recommended practices for reliable use of large language models in structured workflows [13]. In addition to the six dataset classes used for YOLO training (Cardboard, Glass, Metal, Paper, Plastic, Trash), two auxiliary categories (“Organic” and “Other”) were included in the GPT-5 Vision schema to handle out-of-taxonomy objects, background clutter, or ambiguous materials not strictly belonging to the detector’s training set. The prompt template and output schema used in this work are shown in Listing 1 and Listing 2, respectively.

Programmatic validation and error handling. All responses were validated automatically. First, outputs were parsed using a JSON parser. Second, schema checks verified the presence and types of required keys (e.g., category, description, recyclable) and ensured that the category belonged to the allowed set. If parsing or schema validation failed, the request was reissued once with a stricter instruction emphasizing “JSON-only output.” Responses that remained invalid after the retry were logged and excluded from downstream analysis.

Listing 1. GPT-5 Vision prompt template (single object/crop; JSON-only). Note: The GPT-5 schema intentionally excludes ‘Trash’; see evaluation protocol for how ‘Trash’ ground-truth images are handled.

```
Analyze the provided image (object crop) and return ONLY valid JSON (no
markdown, no extra text).
```

```
Choose category from:
```

```
["Plastic", "Paper", "Glass", "Metal", "Cardboard", "Organic", "Other"].
```

```
Return exactly this JSON object:
```

```
{
  "category": "<one of the allowed categories>",
  "description": "<short factual description (max 25 words)>",
  "recyclable": <true/false>
}
```

```
Rules:
```

- Output must be strictly valid JSON.
- Do not add any text outside the JSON.

Listing 2. GPT-5 Vision output schema (multiclass scene; JSON-only). Note: The GPT-5 schema intentionally excludes ‘Trash’; see evaluation protocol for how ‘Trash’ ground-truth images are handled.

Analyze the full image and return ONLY valid JSON:

```
{
  "detections": [
    {
      "category": "<Plastic|Paper|Glass|Metal|Cardboard|Organic|Other>",
      "description": "<short factual description (max 25 words)>",
      "recyclable": <true/false>
    }
  ]
}
```

Rules:

- If no waste items are present, return "detections": []
- Output must be strictly valid JSON.
- Do not include explanations outside the JSON.

Inference procedure. For each input image, a structured inference pipeline was applied using GPT-5 Vision. Images were first encoded and submitted to the model via API calls, together with a predefined prompt that enforced a JSON-formatted response.

The prompt required the model to return three fields: (i) category (Plastic, Paper, Glass, Metal, Cardboard, Organic, Other), (ii) a short textual description of the object, and (iii) a recyclability indicator (true/false). The inclusion of “Organic” and “Other” categories enables the model to represent materials that do not directly map to the six-class detection taxonomy or correspond to organic waste types not explicitly annotated in the dataset.

Model responses were automatically validated through JSON parsing to ensure syntactic correctness. Valid outputs were stored and aggregated for further analysis.

To ensure consistency between detection and semantic evaluation, two related but distinct taxonomies were used in this study. The YOLO-based detection models operate on six categories (Plastic, Paper, Glass, Metal, Cardboard, Trash), while the GPT-5 Vision module uses an extended semantic taxonomy that includes “Organic” and “Other” to improve expressiveness and robustness.

Finally, a category distribution analysis was performed to examine the frequency of material types predicted by GPT-5 Vision.

Ground-truth definition and evaluation protocol. Ground-truth labels used for GPT-5 Vision evaluation were derived from the manually annotated YOLO validation dataset described in Section 3.1. Each validation image has one or more annotated object categories from the six-class waste taxonomy (Cardboard, Glass, Metal, Paper, Plastic, Trash).

GPT-5 Vision was evaluated at the image level. For single-object images, the predicted category returned by GPT-5 Vision was directly compared to the corresponding annotated ground-truth label. For images containing multiple annotated objects, the GPT-5 Vision output was evaluated using a relaxed image-level semantic agreement criterion: a prediction was counted as matched if at least one predicted category corresponded to at least one ground-truth category present in the image. This protocol was adopted because GPT-5 Vision was not used as a bounding-box detector in this study and did not aim to exhaustively recover all annotated instances in the scene. Accordingly, the resulting metrics should be interpreted as indicators of semantic alignment under a constrained image-level protocol, not as strict multi-label completeness or object-level detection accuracy.

True Positives (TP), False Positives (FP), and False Negatives (FN) were computed based on category agreement at the image level. Precision, Recall, and F1-score were then calculated as

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP});$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN});$$

$$\text{F1} = 2\text{PR} / (\text{P} + \text{R}).$$

Because GPT-5 Vision does not produce spatial bounding boxes in this protocol, IoU-based detection metrics such as mAP are not applicable.

It is important to emphasize that this protocol reflects image-level semantic agreement rather than strict multi-label completeness. For multi-object images, a prediction was considered matched if at least one ground-truth category was identified; therefore, this protocol represents a relaxed matching criterion rather than strict multi-label completeness. This relaxed criterion may overestimate performance in complex multi-object scenes because it does not require the model to identify all annotated categories present in the image.

Therefore, the reported GPT-5 Vision metrics should be interpreted as semantic alignment indicators rather than exhaustive object-level verification performance. Within the proposed pipeline, GPT-5 Vision is therefore positioned as an auxiliary semantic interpretation and category-suggestion module, rather than a strict verification mechanism. Future work will adopt stricter evaluation protocols, including multi-label completeness scoring and per-crop object-level evaluation using YOLO bounding boxes.

The proposed methodology follows a modular design that separates detection, description, and semantic reasoning into distinct components. This design improves system interpretability, enables independent evaluation of each module, and facilitates future extensions or component replacement without affecting the overall pipeline.

All implementation details related to model training, multimodal integration, and evaluation protocols are described in this section to ensure clarity and reproducibility. The Results section focuses exclusively on performance analysis and empirical findings.

4. Results

This section presents the experimental results of the proposed multimodal framework, including object detection performance, cross-validation analysis, and multimodal semantic evaluation. The analysis includes a comparative evaluation of the YOLOv8m and YOLOv10m object detection models, performance verification using Stratified K-Fold Cross-Validation, and the integration of Vision-Language Models (MiniGPT-4 and GPT-5 Vision) for semantic classification and descriptive generation. The experiments collectively assess model accuracy, performance stability, and the interpretability of results in both single-class and multi-class waste detection scenarios.

A comparative analysis is conducted on the visual and quantitative results of two models, YOLOv8m and YOLOv10m, trained on a dataset for object detection. The analysis is based on several key aspects of performance: the learning curve during training, the Mean Average Precision (mAP), precision/recall for individual classes, as well as metrics such as the effectiveness of inference and the training time.

Since the first epoch, the YOLOv8m model has shown consistently lower training loss than the YOLOv10m model, indicating that it learns faster. As can be seen in Figure 2, the total loss of YOLOv8m is lower, while YOLOv10m starts with a much higher loss value (close to 20). The loss curve gradually decreases and stabilizes after approximately 40–50 epochs. The gap between training and validation loss remains relatively small for both models, suggesting limited overfitting and reasonable generalization. YOLOv8m converges faster and reaches a lower final loss (≈ 1.8 after 100 epochs) compared to YOLOv10m

(≈ 3.8), indicating more efficient learning under the selected training configuration. Mean Average Precision (mAP) evaluates the overall accuracy of the model in object detection:

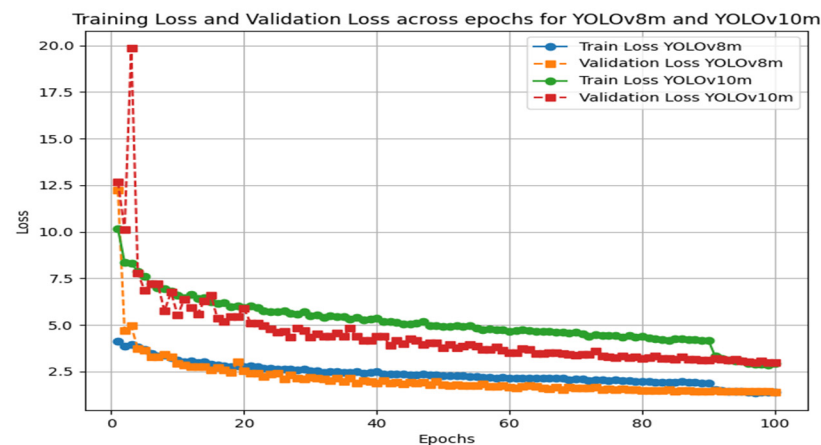


Figure 2. Training Loss and Validation Loss across epochs for YOLOv8m and YOLOv10m.

As shown in Table 1, although the difference is small, YOLOv8m improves mAP@50 by approximately 1.0 percentage point (90.5% vs. 89.5%) and mAP@50–95 by approximately 1.1 percentage points (87.1% vs. 86.0%). Overall, YOLOv8m achieved slightly higher accuracy and efficiency under the evaluated setting.

Table 1. Detection performance (mAP) under the 70/15/15 split for YOLOv8m and YOLOv10m.

Model	mAP@50	mAP@50–95
YOLOv8m	90.5%	87.1%
YOLOv10m	89.5%	86.0%

The training dynamics of YOLOv8m and YOLOv10m, in terms of mAP progression across epochs, are shown in Figure 3.

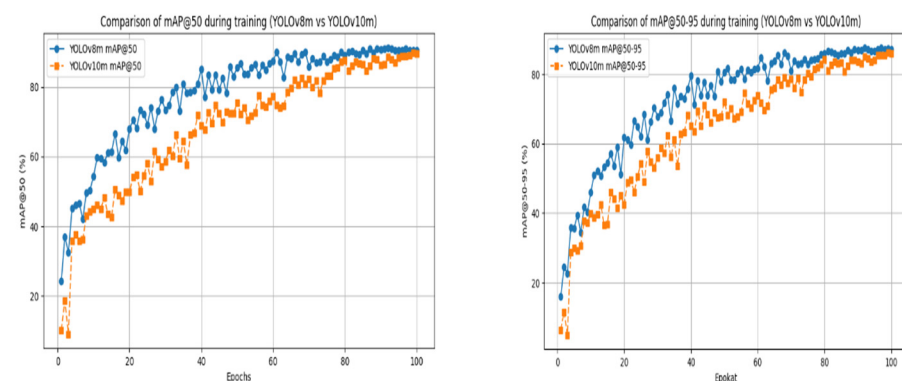


Figure 3. Comparison of mAP during training (YOLOv8m vs. YOLOv10m).

To further analyze model behavior, class-wise precision and recall were examined. Precision measures the correctness of model predictions, whereas recall measures detection completeness.

Figure 4 clearly shows that YOLOv8m achieves higher precision and recall across almost all categories compared to YOLOv10m. In particular:

- Plastic: 88% (YOLOv8m) vs. 85% (YOLOv10m);
- Glass: 89% vs. 87%;
- Trash: 82% vs. 79%.

YOLOv8m also shows consistently higher recall across all categories, indicating fewer missed detections. Although the differences are moderate, they translate into improved reliability under real-world conditions.

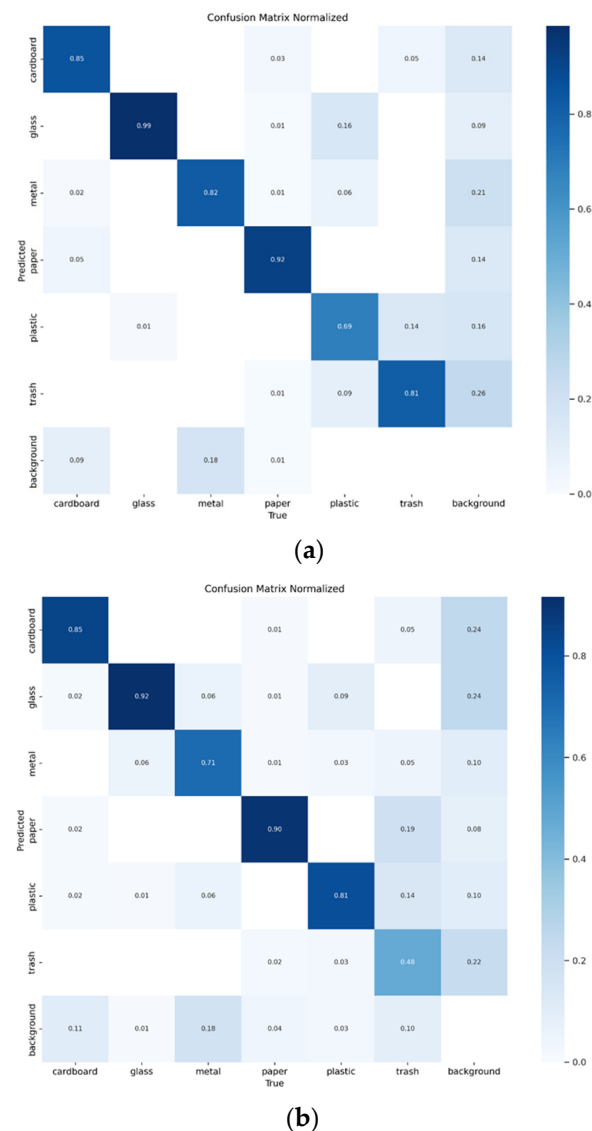


Figure 4. Comparison of YOLOv8m and YOLOv10m via Confusion Matrices (Trash, Plastic, Paper, Metal, Glass, Cardboard). (a) YOLOv8m. (b) YOLOv10m.

The normalized confusion matrices in Figure 4 provide detailed insight into inter-class performance.

For YOLOv8m, the Glass and Paper classes exhibit the highest true positive rates. Plastic and Trash display more frequent confusion with the Background class.

YOLOv10m shows a slight decrease in performance for certain classes (e.g., Metal), but a marginal improvement for Plastic. Overall, both models demonstrate similar inter-class behavior patterns, although YOLOv8m maintains slightly stronger diagonal dominance across most categories.

As shown in Table 2, YOLOv8m achieves approximately 120 FPS with 8.3 ms latency per frame, compared to YOLOv10m's 105 FPS and 9.5 ms latency.

Table 2. Comparison of inference speed between YOLOv8m and YOLOv10m, expressed as FPS (frames per second) and as average latency per frame (in milliseconds).

Model	FPS (Frames per Second)	Latency (ms)
YOLOv8m	120	8.3
YOLOv10m	105	9.5

YOLOv8m achieves slightly higher accuracy and efficiency, making it suitable for real-time applications under the evaluated conditions.

The observed differences should be interpreted as dataset and hyperparameter-dependent rather than universally applicable.

Figures 5 and 6 present the F1–Confidence curves. YOLOv8m achieves its best F1-score (0.87) at a confidence threshold of ~0.51, showing a strong balance between precision and recall.

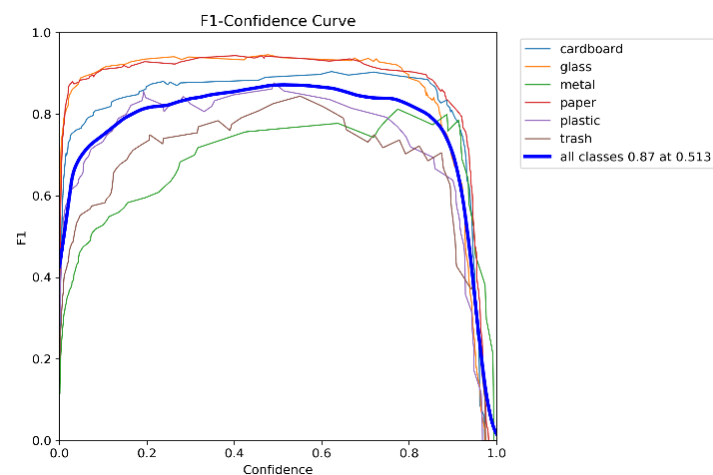


Figure 5. F1-Confidence Curve for YOLOv8m.

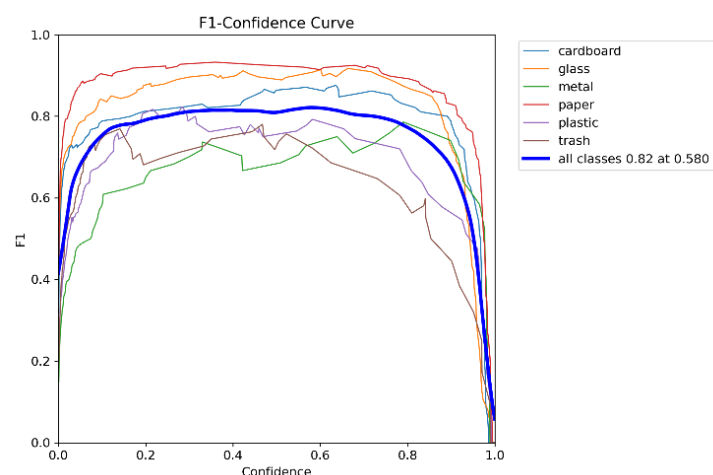


Figure 6. F1-Confidence Curve for YOLOv10m.

YOLOv10m reaches a lower maximum F1-score (0.82) at a higher threshold (~0.58), suggesting greater sensitivity to confidence calibration.

4.1. Quantitative Evaluation of MiniGPT-4 Descriptions

To complement the qualitative analysis, a structured quantitative evaluation of the generated descriptions was conducted. A randomly selected subset of 100 detected object

samples from the validation set was manually assessed according to three predefined objective criteria:

- **Factual Consistency**—The description accurately reflects the visual content of the detected object.
- **Category Alignment**—The description is semantically consistent with the predicted waste category.
- **Recyclability Correctness**—The recycling guidance provided is appropriate for the actual object type shown in the image.

Each criterion was evaluated using a binary scoring scheme (1 = correct, 0 = incorrect). The evaluation was performed manually according to predefined criteria in order to provide a consistent assessment of semantic accuracy and recycling guidance.

The quantitative evaluation of the multimodal description module across factual consistency, category alignment, and recyclability correctness is presented in Table 3.

Table 3. Quantitative Evaluation of Multimodal Description Module ($n = 100$).

Criterion	Accuracy (%)
Factual Consistency	99%
Category Alignment	71%
Recyclability Correctness	81%

The results indicate that the description module demonstrates very strong visual grounding, with 99% factual consistency, confirming that the generated descriptions accurately reflect the visual content of the detected objects.

Category alignment reached 71%, suggesting that while most descriptions remain semantically related to the predicted waste category, certain discrepancies arise in cases of visually similar materials or ambiguous object presentations.

Recyclability correctness achieved 81%, indicating generally reliable recycling guidance. Minor inaccuracies were primarily associated with ambiguous material identification or generalized recycling assumptions.

Overall, the findings suggest that the multimodal description module provides visually grounded descriptions and generally accurate recycling-related guidance, while still leaving room for improvement in category-level semantic alignment.

Following object detection with YOLOv8, the second stage of the pipeline integrates a Large Language Model (LLM) to generate textual descriptions for each detected object.

MiniGPT-4 processes cropped object regions and optionally incorporates predicted class labels to generate context-aware descriptions. The process is fully automated, with each detected object passed to the model using a structured prompt for descriptive output generation.

These outputs support downstream tasks such as visualization, reporting, and user guidance. The evaluation includes both qualitative examples and quantitative analysis.

For illustration, representative outputs are presented for different material categories. For a metal object (e.g., bottle cap), the model generates descriptions that capture material properties and recycling recommendations. Similarly, for paper (envelope) and plastic (water bottle), the generated outputs emphasize material characteristics and provide general recycling guidance. These examples demonstrate the ability of the model to produce context-aware and informative descriptions across diverse waste categories.

Beyond object detection, the integration of YOLOv8m with MiniGPT-4 introduced an additional layer of functionality by enabling the generation of customized textual descriptions for each detected object. This pipeline transcends traditional detection by providing an educational and interactive approach that informs users about the characteristics of

detected waste items while offering practical recycling guidelines. The integration underscores the potential of combining computer vision models with generative language models to enhance not only technical performance but also user experience.

4.2. Cross-Validation Analysis

To further assess the stability of YOLOv8m across alternative data partitions, Stratified 5-Fold Cross-Validation was performed. The quantitative results obtained for each fold are presented in Table 4. The bold formatting is used to distinguish the average and std rows from the individual fold rows and to improve readability.

Table 4. Stratified 5-Fold Cross-Validation results for YOLOv8m.

Fold	Precision	Recall	mAP50	mAP50–95
Fold 0	0.9768	0.9580	0.9835	0.9692
Fold 1	0.9649	0.9384	0.9837	0.9682
Fold 2	0.9585	0.9551	0.9842	0.9713
Fold 3	0.9069	0.8426	0.9275	0.9088
Fold 4	0.8530	0.8132	0.8712	0.8400
Average	0.9324	0.9011	0.9500	0.9315
Std ($\pm$)	0.0516	0.0684	0.0504	0.0575

Despite the overall strong cross-validation performance, variability across folds was observed. While stratified sampling preserves class proportions, it does not account for differences in scene composition, object scale, or visual complexity within each partition. In particular, the lower performance observed in Fold 4 ($\text{mAP@50-95} \approx 0.84$) is plausibly associated with a higher proportion of visually ambiguous scenes, as suggested by the recall–confidence curves and the representative validation examples shown in Figures 7 and 8. These factors increase detection difficulty and highlight dataset heterogeneity beyond simple class balance. The observed variance further justifies the use of cross-validation to obtain a more realistic estimate of performance stability across stratified data partitions.

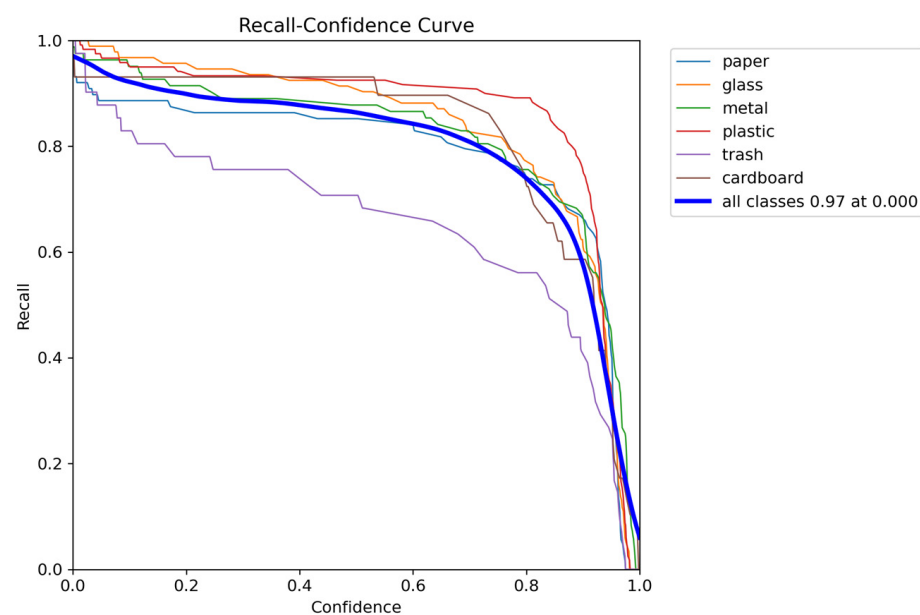


Figure 7. Recall–confidence curves for the evaluated classes in Fold 4.

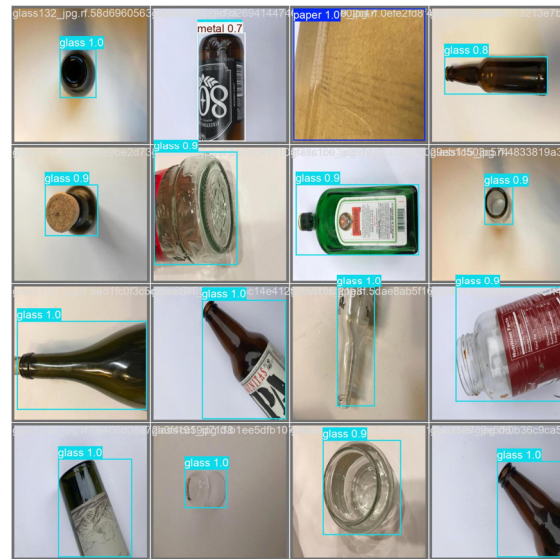


Figure 8. Representative validation samples from Fold 4 illustrating visually challenging cases.

To further analyze the reduced performance observed in Fold 4, both quantitative and qualitative evaluations were conducted.

First, class-level performance metrics show that the degradation is not uniform across categories. Plastic consistently exhibits high stability across confidence thresholds, while trash and cardboard exhibit earlier recall degradation in the Recall–Confidence curves. The per-class AP values show that trash and cardboard are the most affected categories.

The normalized confusion matrix further reveals increased background confusion for trash and cross-class ambiguity between cardboard and visually related categories.

Additionally, qualitative inspection of representative validation samples from Fold 4 shows the presence of partially visible objects, deformed cardboard structures, small-scale metal/glass fragments, and objects positioned against visually uniform backgrounds. These characteristics increase background confusion and inter-class similarity.

Importantly, training and validation loss curves remain stable and converge consistently, suggesting that the observed variability is not driven by optimization instability. Rather, the reduced Fold 4 performance reflects fold-level dataset heterogeneity in object scale and structural appearance, as demonstrated in Figures 7 and 8.

These observations provide empirical support for the fold-specific performance variation observed in Table 4.

The obtained results indicate that YOLOv8 shows strong average performance across folds (Average Precision = 0.932 and mAP@50–95 = 0.9315). However, the fold-wise results also reveal non-negligible variability (Std: Precision $\pm$ 0.0516, Recall $\pm$ 0.0684, mAP@50 $\pm$ 0.0504, mAP@50–95 $\pm$ 0.0575). Folds 3–4 exhibit lower mAP@50–95 compared to Folds 0–2. Although stratified splitting preserves class proportions, it does not fully control for scene difficulty and intra-class variability (e.g., clutter, occlusions, reflections, and transparent materials), which may be unevenly distributed across folds. This observation motivates additional error analysis and the expansion of the dataset with more diverse in-the-wild urban scenes in future work. Overall, the results suggest that YOLOv8m maintains strong average performance across folds, while also exhibiting fold-wise variability associated with dataset heterogeneity. These findings support the usefulness of Stratified K-Fold Cross-Validation as a variance-aware evaluation procedure for the present dataset.

Figure 9 illustrates the evolution of the evaluation metrics during the training process of the YOLOv8 model on one of the folds of the Stratified K-Fold Cross Validation.

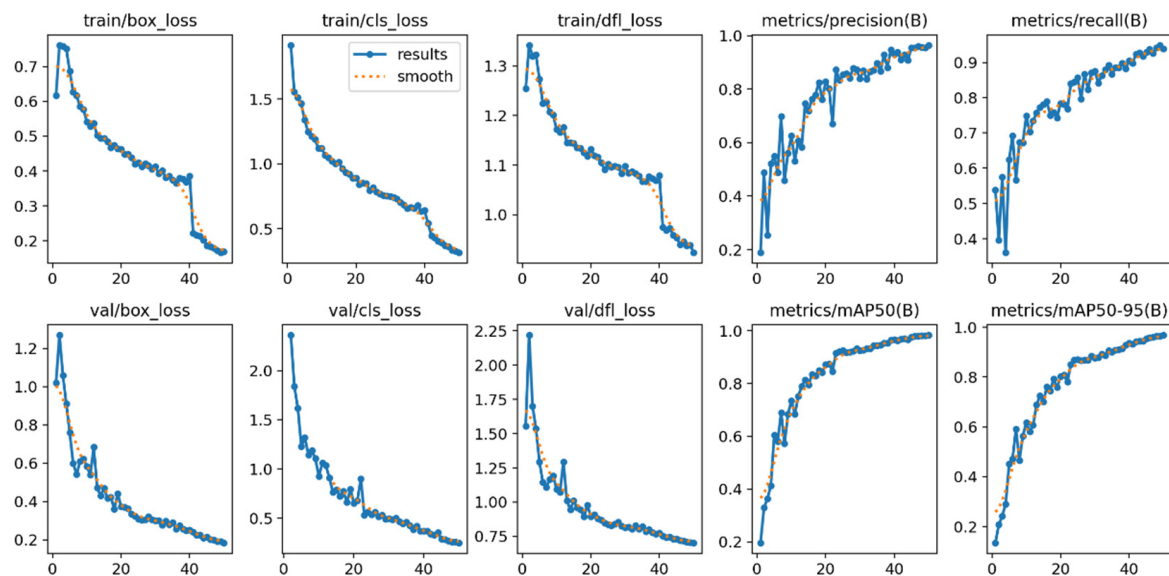


Figure 9. Training loss and validation metric evolution for YOLOv8m in Fold 1.

The upper section of the figure presents the training losses for the components `box_loss`, `cls_loss`, and `dfl_loss`, while the lower section displays the corresponding metrics for the validation data. A gradual and consistent decrease in loss can be observed, indicating good model convergence. Furthermore, the metrics Precision, Recall, mAP@50 and mAP@50–95 reach high values toward the end of training, reflecting stable performance and the absence of overfitting.

Figure 10, illustrates the relationship between Recall and the Confidence Threshold for each class (paper, glass, metal, plastic, trash, and cardboard).

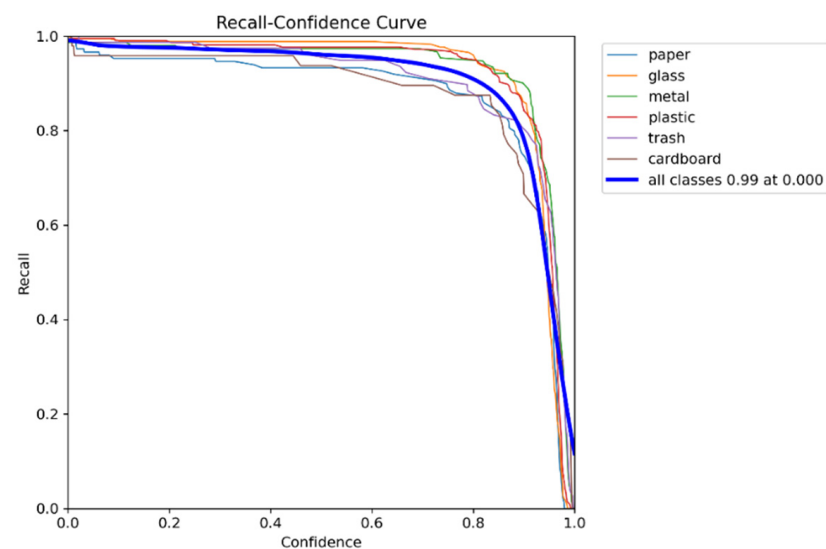


Figure 10. Recall–confidence curves for all dataset classes in Fold 0.

The thick blue curve represents the overall performance across all classes, with a Recall value of approximately 0.99, indicating that the model successfully detects most objects with high reliability across varying confidence levels.

The individual-colored curves corresponding to each class show that the model performs more accurately for materials such as plastic, metal, and glass, while exhibiting slight variations for more challenging categories such as trash and cardboard.

The normalized confusion matrix in Figure 11 illustrates the percentage of correct detections for each class, where darker values indicate higher accuracy.

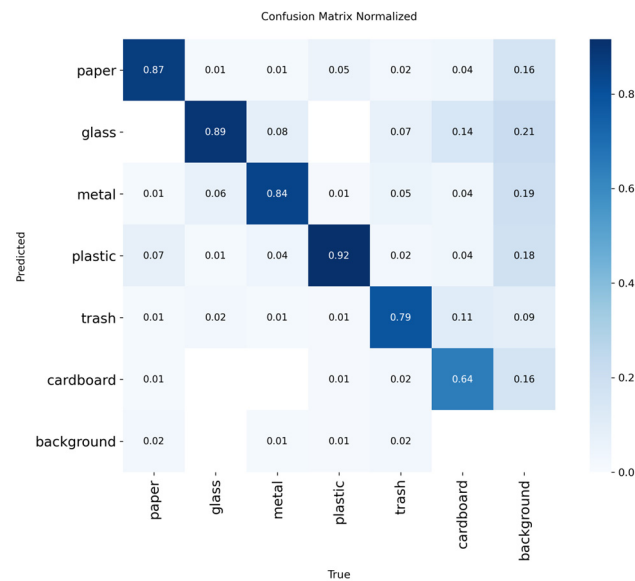


Figure 11. Normalized confusion matrix for YOLOv8m evaluation in Fold 2.

The YOLOv8 model achieved high accuracy for the classes paper (0.87), glass (0.89), and plastic (0.92), demonstrating strong capability in distinguishing recyclable materials. Slight misclassifications are observed between cardboard and trash, likely due to their visual similarity in the images.

Overall, the matrix suggests relatively consistent inter-class performance, with high Recall levels across most categories.

Vision-language models such as GPT-5 Vision provide multimodal processing capabilities that combine image understanding with natural-language generation, enabling richer semantic interpretation of scene content. These models integrate visual representations (derived from images) with semantic understanding (derived from natural language), enabling a deeper and context-aware interpretation of scene content.

In contrast to traditional object detection models such as YOLOv8 and YOLOv10, which focus solely on identifying and localizing objects within an image, GPT-5 possesses the ability to both visually analyze and semantically describe the detected objects. This allows for a richer and more interpretable form of classification that combines perception and reasoning.

To evaluate the potential of multimodal models in the task of visual classification, this study employed GPT-5 Vision as an auxiliary semantic classification and category-suggestion module following the detection phase performed by YOLOv8.

The GPT-5 model was used to directly analyze images and generate structured outputs in JSON format, including the waste category, a textual description, and an indication of whether the object is recyclable.

GPT-5 Vision was applied to the validation dataset to generate structured semantic outputs, including category labels, descriptions, and recyclability indicators. The analysis focused on evaluating the semantic consistency and interpretability of the generated outputs.

In total, 337 validation images were analyzed, all of which were successfully categorized by GPT-5 according to the primary material types. Table 5 presents a selection of the results obtained.

Table 5. Example outputs generated by GPT-5 Vision, including predicted category, description, and recyclability.

Category	Description	Recyclable	Image Name
Cardboard	Flattened cardboard packaging box from a food product.	True	cardboard163.jpg
Plastic	A small piece of clear plastic film or wrapping material.	False	cardboard17.jpg
Glass	A green glass bottle.	True	glass344.jpg
Glass	A clear glass jar of sliced peaches.	True	glass65.jpg
Glass	An empty, clear glass jar.	True	glass396.jpg

GPT-5 Vision successfully analyzed 337 validation images, generating structured semantic outputs with category labels, descriptions, and recyclability indicators. For each detected object, GPT-5 provided additional semantic information, thereby enriching the classification process beyond the basic labeling performed by YOLOv8.

For example:

“A flattened cardboard packaging box”

“A clear glass jar of sliced peaches”

demonstrate the model’s ability to generate context-aware descriptions that can be utilized for automatic interpretation or content reporting.

These results suggest that GPT-5 Vision can serve as a complementary semantic analysis component, providing structured category suggestions alongside visual recognition.

Its integration on top of YOLO’s detection outputs provides additional semantic interpretability and auxiliary category-level support in complex scenes.

4.3. Multiclass Image Analysis Using GPT-5 Vision

The objective of this experiment was to evaluate the ability of the GPT-5 Vision model to analyze images containing multiple objects from different waste categories (multiclass scenario).

In this setting, GPT-5 Vision was employed as a multiclass semantic analyzer and category-suggestion module, tasked with identifying visible objects, assigning categories, generating short descriptions, and determining recyclability. A total of 13 images were selected, each containing several distinct categories (e.g., plastic, metal, cardboard, paper, glass, organic, other). Each image was analyzed using the GPT-5 Vision API with a structured JSON prompt (see Listing 3).

Listing 3. Example GPT-5 Vision multiclass output (JSON format).

```
{
  "detections": [
    {"category": "Plastic", "description": "water bottle",
    "recyclable": true},
    {"category": "Metal", "description": "aluminum can", "recyclable":
    true}
  ]
}
```

The model was guided to

- Identify each distinct object within the image.
- Assign the correct category from the list [Plastic, Paper, Glass, Metal, Cardboard, Organic, Other].
- Provide a short textual description.
- Indicate whether the object is recyclable or not.

The results obtained for each image were stored in a CSV file (multi_detection_results.csv) for subsequent statistical analysis and visualization.

From the 13 tested images, GPT-5 Vision successfully identified a total of 90 individual detections.

Figure 12 illustrates the distribution of categories detected by GPT-5 Vision in multi-class images.

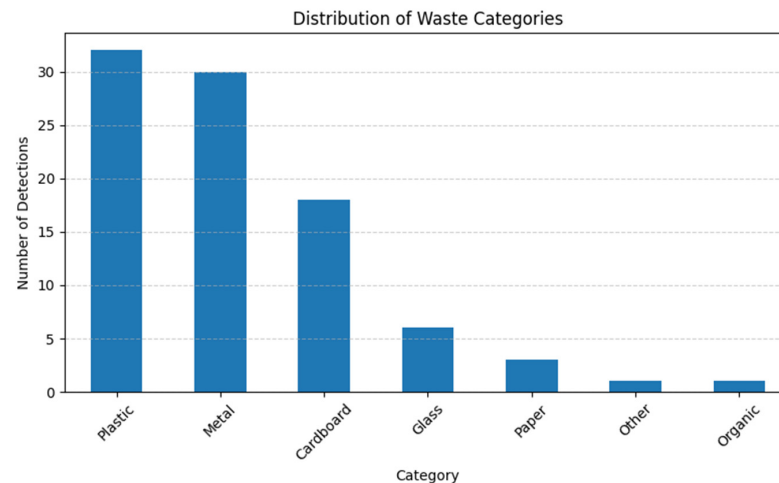


Figure 12. Distribution of categories detected by GPT-5 Vision.

The category distribution graph shows that Plastic (≈ 32 detections) and Metal (≈ 30 detections) are the most frequently identified categories, followed by Cardboard (≈ 18 detections) and Glass (≈ 6 detections), while Paper, Organic, and Other appear less frequently. It should be noted that the auxiliary categories Organic and Other appear only in the GPT-5 prompt schema and therefore represent semantic outputs that fall outside the six-class benchmark taxonomy used for detector training.

This distribution reflects the predominance of certain object types within the evaluated sample and illustrates GPT-5 Vision's ability to generate structured multi-object outputs at the image level. However, these counts should be interpreted as indicators of semantic category presence rather than strict multi-label performance accuracy, since evaluation was conducted using an image-level agreement criterion.

In one representative multiclass example, GPT-5 Vision produced the following structured output (see Listing 4).

Listing 4. Example GPT-5 Vision structured output for a multiclass scene.

```
{
  "detections": [
    {"category": "Plastic", "description": "A transparent plastic cup", "recyclable": true},
    {"category": "Metal", "description": "A crushed aluminum can", "recyclable": true},
    {"category": "Cardboard", "description": "A flattened brown cardboard box", "recyclable": true}
  ]
}
```

GPT-5 Vision was evaluated using an image-level semantic agreement criterion. In multi-object images, predictions were considered matched if at least one ground-truth category was identified. This protocol does not require prediction of all annotated categories and therefore does not represent strict multi-label completeness scoring. Accordingly, the

reported results should be interpreted as image-level semantic agreement rather than exhaustive multi-object recognition accuracy. Within the proposed framework, GPT-5 Vision is positioned as an auxiliary semantic classification and category-suggestion module rather than an object-level verification component. Future work will incorporate stricter object-level and multi-label evaluation protocols.

In this example, GPT-5 successfully

- Distinguished three different categories within the same scene;
- Provided accurate descriptions for each object;
- Correctly assessed the recyclability status of each category.

This example illustrates the multimodal reasoning capability of GPT-5 Vision to interpret complex visual scenes and generate structured, semantically accurate responses.

Table 6 summarizes the category distribution obtained from GPT-5 Vision in multiclass image analysis.

GPT-5 Vision performs well in identifying common categories such as plastic, metal, and cardboard. Minor errors occur when objects overlap or exhibit strong reflections (e.g., transparent plastic near glass). The structured JSON output enables straightforward automated analysis and statistical visualization.

Metrics for GPT-5 Vision were computed against the six-class dataset ground truth (Plastic, Paper, Glass, Metal, Cardboard, Trash). Since ‘Trash’ is not included in the GPT-5 allowed output categories (Listings 1 and 2), GPT-5 predictions were evaluated only on the overlapping classes. Ground-truth images labeled as ‘Trash’ are therefore reported as non-overlapping/out-of-taxonomy cases for GPT-5 in this protocol.

Table 6. Distribution of GPT-5 Vision detections across semantic categories in multiclass images.

Category	Number of Detections	Percentage
Plastic	32	35.2%
Metal	30	33.0%
Cardboard	18	19.8%
Glass	6	6.6%
Paper	3	3.3%
Organic	1	1.1%
Other	1	1.1%
Total	91	100%

Note: The categories “Organic” and “Other” originate from the GPT-5 prompt schema and are not part of the six-class dataset taxonomy used for YOLO training and evaluation.

It is important to note that GPT-5 Vision was evaluated as an image-level semantic classification and category-suggestion module rather than as a bounding-box detector. Ground-truth labels were obtained from the manually annotated validation set used for YOLO training. For single-object images, GPT-5 predictions were directly compared with the annotated label. For multi-object images, predictions were considered correct if they matched at least one annotated category in the image. The reported Precision, Recall, and F1-score reflect category-level agreement at the image level. IoU-based metrics such as mAP are not applicable since GPT-5 Vision does not generate spatial bounding-box outputs.

The comparison presented in Table 7 should be interpreted cautiously, as the two systems operate under different evaluation paradigms. YOLOv8 performs object-level detection with bounding boxes, whereas GPT-5 Vision was evaluated as an image-level semantic classification module. Therefore, the metrics reported for GPT-5 Vision reflect

semantic category agreement rather than spatial detection performance. The purpose of this comparison is to illustrate the complementary semantic capabilities of multi-modal vision–language models rather than to provide a direct benchmark against object detection systems.

Table 7. Image-level semantic agreement metrics for GPT-5 Vision and reference object-level metrics for YOLOv8 under different evaluation protocols.

Model	Precision	Recall	F1-Score
GPT-5	0.916562	0.886986	0.901531
YOLOv8	0.89413	0.85351	0.873348

Note: The values reported for GPT-5 Vision and YOLOv8 reflect different evaluation paradigms and are therefore not directly comparable as object-detection benchmarks. YOLOv8 was evaluated at the object level using spatial annotations, whereas GPT-5 Vision was evaluated at the image level as an auxiliary semantic classification module.

Because the two systems were evaluated under different protocols, the values in Table 7 should be interpreted descriptively rather than as a direct benchmark comparison. These results indicate that GPT-5 Vision can function as a complementary semantic analysis component for multicategory images, providing structured category suggestions alongside traditional object detection approaches.

5. Discussion

This study presents a multimodal “vision–language” pipeline for the detection and semantic interpretation of urban waste, combining object detection models (YOLOv8m/YOLOv10m) with caption generation (MiniGPT-4) and a multimodal auxiliary semantic classification and category-suggestion module (GPT-5 Vision). Given that the growth of urban waste and the increasing pressure on waste management infrastructure are widely documented, the need for intelligent systems with high interpretability is becoming increasingly critical [1,3].

5.1. Relation to Existing Work and Practical Contribution

In the literature, a significant line of research focuses on intelligent waste management through IoT/sensors and collection optimization (e.g., fill-level prediction and route planning), which is considered a core pillar of “smart cities” [4,19]. Additionally, several studies approach waste management as an IoT service and emphasize improvements in efficiency and cost reduction [5,7]. In parallel, a substantial body of computer vision research employs fast one-stage detectors such as YOLO for waste identification and classification [8,10,11].

However, many existing solutions remain at the level of “label + bounding box,” without adding user-understandable semantic interpretation or an educational component. Our approach aims to overcome this limitation by integrating a generative component (MiniGPT-4), proposed as a vision–language model for producing rich and meaningful image descriptions [12], and by adding multimodal semantic analysis and category suggestion with GPT-5 Vision, in line with trends in large multimodal models (e.g., GPT-4 and its technical derivatives) [13]. In this way, the contribution of this work is not merely a comparison of YOLO variants, but a practical, real-time pipeline that combines detection with interpretation and educational feedback.

5.2. Interpretation of Detection Results and YOLOv8m vs. YOLOv10m

Our results showed a consistent advantage of YOLOv8m over YOLOv10m in terms of mAP, training convergence behavior, and inference efficiency. This aligns with the understanding that detector performance depends not only on architecture, but also on domain shift, class distribution, and training configuration. In the detection literature,

YOLO has been introduced as a real-time detection paradigm [8], while modern variants and implementations (e.g., Ultralytics) enable standardized training and comparison workflows [11].

In the waste domain, recent studies report strong performance using YOLO variants (e.g., YOLOv8) for household waste classification; however, results strongly depend on the dataset, number of categories, and image conditions [18]. In our dataset (constructed from well-known public waste sources such as TrashNet and TACO), YOLOv8m appears to provide the best balance between accuracy and speed [20,21]. Therefore, the conclusion that “YOLOv8m is slightly more accurate and slightly faster” should be interpreted as configuration- and data-dependent, rather than a universal judgment for all scenarios.

5.3. Value and Risks of Multimodal Components (MiniGPT-4 and GPT-5 Vision)

The integration of MiniGPT-4 enhances semantic interpretability by generating user-oriented descriptions and recycling guidance. This approach is supported by literature describing MiniGPT-4 as a model that connects visual representations with a large language model (LLM) to generate rich textual output [12], as well as by foundational vision-language bootstrapping work (e.g., BLIP-2) [15].

On the other hand, generative models carry the risk of producing generic or partially incorrect text (“hallucinations”), especially in domains requiring precise terminology or local rules. For this reason, in a practical system, outputs should be accompanied by control mechanisms (e.g., structured prompts, enforced output formats, and output validation), consistent with recommended practices for large models (e.g., technical discussions on GPT-4 capabilities and limitations) [13].

The GPT-5 Vision component is used as an auxiliary semantic classification and category-suggestion step (for visually similar classes and multiclass scenes), adding an additional level of reasoning on top of the detection stage. Although this improves interpretability and may reduce errors, it introduces additional cost and latency if frequently invoked. Therefore, it is reasonable to use it selectively (e.g., only when YOLO confidence is low), while preserving the real-time nature of the system [8,11]. This selective invocation is primarily a deployment recommendation to balance latency and cost; in our experiments, GPT-5 Vision was evaluated as an auxiliary semantic classification module on validation and multiclass samples.

5.4. Dataset Bias, Limitations, and Generalization Across Contexts

Our dataset was built from publicly available sources widely used in the literature (TrashNet and TACO) and manually annotated for detector training [20,21]. While practical, this approach may introduce bias: images may be concentrated in certain viewing angles, controlled lighting conditions, or specific background types, which may not fully represent urban “in-the-wild” conditions. Additionally, the current taxonomy is limited and may under-represent rare or region-specific waste types (e.g., composite packaging, textiles, or e-waste), which can affect scalability to broader real-world waste streams.

Moreover, geographic and cultural differences influence waste appearance (packaging styles, label languages, shapes, and materials), meaning that cross-regional generalization may not hold without additional training or domain adaptation. This is particularly relevant given that the growth and diversification of urban waste is a globally documented phenomenon, with projections extending to 2050 [1].

Consequently, real-world deployment in different cities requires expanding the dataset with local images and additional categories, as well as implementing class-balancing strategies.

A practical way to assess generalization is to report cross-domain performance (e.g., train on one source/city and test on another), which we include as a recommended evaluation protocol for future work.

5.5. Reproducibility and Considerations for Real-World Deployment

For reproducibility, detection literature and modern implementations recommend clear reporting of training configurations, dataset splits, hyperparameters, and evaluation metrics [11]. For multimodal systems, reproducibility is further enhanced when the following are shared:

- (i) The annotated dataset (or at least dataset splits and annotation rules),
- (ii) The pipeline code,
- (iii) Prompts and output formats (e.g., structured JSON outputs).

In this study, prompt templates, structured output schemas, and example outputs are explicitly included to facilitate experimental reconstruction. Additional supporting files, including prompt templates, structured outputs, and dataset reconstruction instructions, are provided in the Supplementary Materials.

This is especially important for vision–language components, where prompt engineering directly affects outcomes [12,13,15].

In real-world deployment, the system must consider hardware limitations (e.g., edge devices), privacy (images captured in public spaces), and the fact that recycling rules are local and vary by municipality. These aspects are also aligned with “smart city” paradigms, where IoT + AI integration must be sustainable and secure [4,5,8].

5.6. Future Directions

Future work should focus on

- (1) Expanding the dataset with real urban scenes under diverse conditions, following “in-context” dataset practices such as TACO [20];
- (2) Increasing the number of classes and developing a richer taxonomy;
- (3) Selectively applying multimodal semantic analysis and category suggestion to balance interpretability with cost/latency;
- (4) Integrating curated recycling knowledge (e.g., local rules) to reduce inaccurate LLM outputs;
- (5) Conducting field pilots within smart city applications, where IoT and AI collaborate for optimization and transparency [4,7,19].

6. Conclusions

This study presented a multimodal vision–language pipeline for urban waste detection and semantic interpretation by combining real-time object detection (YOLOv8m/YOLOv10m) with caption generation (MiniGPT-4) and an auxiliary semantic classification and category-suggestion module (GPT-5 Vision). A custom dataset of 2247 manually annotated images was constructed from publicly available sources and used to train and evaluate both detection models.

The experimental results indicate that YOLOv8m achieved slightly higher detection performance and inference efficiency than YOLOv10m under the evaluated dataset and training configuration. These findings are supported by the reported mAP@50–95 scores, cross-validation results, and inference performance metrics presented in Section 4. Accordingly, YOLOv8m can be considered suitable for real-time applications under the evaluated conditions.

Beyond object detection, the integration of MiniGPT-4 and GPT-5 Vision extends the functionality of the system by enabling the generation of context-aware textual descrip-

tions and structured semantic outputs. These components support interpretability and user-oriented interaction, illustrating the potential of combining perception and semantic reasoning within a unified multimodal framework.

Despite these promising results, several limitations should be acknowledged. The performance of the system is influenced by dataset characteristics, including class distribution, scene complexity, and potential bias introduced by combining multiple data sources. In addition, the quality of generated descriptions and semantic outputs depends on the reliability of generative models, which may occasionally produce generic or partially inaccurate responses.

Future work will focus on expanding the dataset with more diverse real-world urban scenarios, extending the waste taxonomy, and incorporating more rigorous evaluation protocols for multimodal components. Furthermore, selective integration of semantic modules may be explored to balance interpretability with computational efficiency in practical deployments.

Supplementary Materials: The following supporting information can be downloaded at <https://www.mdpi.com/article/10.3390/informatics13040057/s1>: The following materials are provided to support reproducibility: S1_GPT5_single_object_results.csv—Structured GPT-5 outputs for validation samples; S1_GPT5_multiclass_results.csv—Structured outputs for multiclass scene analysis; S1_dataset_reconstruction.txt—Dataset reconstruction instructions; S1_prompts.txt—Prompt templates used for GPT-5 Vision inference; S1_example_outputs.json—Example structured GPT-5 Vision outputs; Waste_Detection_Report.pdf—Example detection reports generated by the pipeline. S1_category_distribution.csv.xlsx—Category-wise distribution of detections used to generate the category distribution summary. Category_distribution.png—Bar chart showing the distribution of detected waste categories.

Author Contributions: Conceptualization, V.M.J.; methodology, V.M.J.; software, V.M.J.; validation, V.M.J.; formal analysis, V.M.J.; investigation, V.M.J.; resources, V.M.J.; data curation, V.M.J.; writing—original draft preparation, V.M.J.; writing—review and editing, I.M.; visualization, V.M.J.; supervision, I.M.; project administration, V.M.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The images used in this study originate from two publicly available datasets: TrashNet and TACO. Due to the licensing conditions of these datasets, the merged dataset and derived annotations created for this study cannot be publicly redistributed. However, the experimental setup can be reconstructed using the same public sources. The dataset composition, class taxonomy, annotation format, and train–validation–test split (70/15/15) are fully described in the manuscript. The original datasets can be accessed at: TrashNet: <https://github.com/garythung/trashnet> (accessed on 21 March 2026); TACO Dataset: <http://tacodataset.org/> (accessed on 21 March 2026). The minimal data required for editorial verification, including dataset reconstruction instructions, prompt templates, example structured outputs, and summary result files, may be made available upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

YOLO	You Only Look Once
LLM	Large Language Model
mAP	Mean Average Precision

References

1. Kaza, S.; Yao, L.C.; Bhada-Tata, P.; Van Woerden, F. *What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050*; World Bank: Washington, DC, USA, 2018. [CrossRef]
2. Fang, B.; Yu, J.; Chen, Z.; Osman, A.I.; Farghali, M.; Ihara, I.; Hamza, E.H.; Rooney, D.W.; Yap, P.-S. Artificial intelligence for waste management in smart cities: A review. *Environ. Chem. Lett.* **2023**, *21*, 1959–1989. [CrossRef] [PubMed]
3. Hannan, M.A.; Al Mamun, M.A.; Hussain, A.; Basri, H.; Begum, R.A. A review on technologies and their usage in solid waste monitoring and management systems: Issues and challenges. *Waste Manag.* **2015**, *43*, 509–523. [CrossRef] [PubMed]
4. Anagnostopoulos, T.; Zaslavsky, A.; Kolomvatsos, K.; Medvedev, A.; Amirian, P.; Morley, J.; Hadjieftymiades, S. Challenges and Opportunities of Waste Management in IoT-Enabled Smart Cities: A Survey. *IEEE Trans. Sustain. Comput.* **2017**, *2*, 275–289. [CrossRef]
5. Medvedev, A.; Fedchenkov, P.; Zaslavsky, A.; Anagnostopoulos, T.; Khoruzhnik, S. Waste Management as an IoT-Enabled Service in Smart Cities. In *Internet of Things, Smart Spaces, and Next Generation Networks and Systems*; Springer International Publishing: Cham, Switzerland, 2015; Volume 9247, pp. 104–115. [CrossRef]
6. Longhi, S.; Marzioni, D.; Alidori, E.; Di Buo, G.; Prist, M.; Grisostomi, M.; Pirro, M. Solid Waste Management Architecture Using Wireless Sensor Network Technology. In *Proceedings of the 2012 5th International Conference on New Technologies, Mobility and Security (NTMS)*; IEEE: Istanbul, Turkey, 2012; pp. 1–5. [CrossRef]
7. Vishnu, S.; Ramson, S.R.J.; Senith, S.; Anagnostopoulos, T.; Abu-Mahfouz, A.M.; Fan, X.; Srinivasan, S.; Kirubaraj, A.A. IoT-Enabled Solid Waste Management in Smart Cities. *Smart Cities* **2021**, *4*, 1004–1017. [CrossRef]
8. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Las Vegas, NV, USA, 2016; pp. 779–788. [CrossRef]
9. Guerrero, L.A.; Maas, G.; Hogland, W. Solid waste management challenges for cities in developing countries. *Waste Manag.* **2013**, *33*, 220–232. [CrossRef] [PubMed]
10. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934. [CrossRef]
11. Jocher, G.; Qiu, J.; Chaurasia, A. Ultralytics YOLO. Python. January 2023. Available online: <https://github.com/ultralytics/ultralytics> (accessed on 11 July 2025).
12. Zhu, D.; Chen, J.; Shen, X.; Li, X.; Elhoseiny, M. MiniGPT-4: Enhancing Vision-Language Understanding with Advanced Large Language Models. *arXiv* **2023**, arXiv:2304.10592. [CrossRef]
13. OpenAI; Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F.L.; Almeida, D.; Altenschmidt, J.; Altman, S.; et al. GPT-4 Technical Report. *arXiv* **2024**, arXiv:2303.08774. [CrossRef]
14. Gan, Z.; Li, L.; Li, C.; Wang, L.; Liu, Z.; Gao, J. Vision-Language Pre-Training: Basics, Recent Advances, and Future Trends. *FNT Comput. Graph. Vis.* **2022**, *14*, 163–352. [CrossRef]
15. Li, J.; Li, D.; Savarese, S.; Hoi, S. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. *arXiv* **2023**, arXiv:2301.12597. [CrossRef]
16. Adadi, A.; Berrada, M. Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI). *IEEE Access* **2018**, *6*, 52138–52160. [CrossRef]
17. Open Knowledge Repository. Available online: <https://openknowledge.worldbank.org/entities/publication/d3f9d45e-115f-559b-b14f-28552410e90a> (accessed on 11 July 2025).
18. Arishi, A. Real-Time Household Waste Detection and Classification for Sustainable Recycling: A Deep Learning Approach. *Sustainability* **2025**, *17*, 1902. [CrossRef]
19. Ferrer, J.; Alba, E. BIN-CT: Urban Waste Collection based in Predicting the Container Fill Level. *Biosystems* **2019**, *186*, 103962. [CrossRef] [PubMed]
20. Tacodataset.org. Available online: <http://tacodataset.org/> (accessed on 11 July 2025).
21. Thung, G. Garythung/Trashnet. Lua. 10 July 2025. Available online: <https://github.com/garythung/trashnet> (accessed on 11 July 2025).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.