

⌚️📷🗑️ Smart Trash Vision (STV)

Date: 2025-06-22

Author: SEOYOUNGWOON, HANSUSEONG

Github: [repository link](#)

Demo Video: [Youtube link](#)

1. Introduction

Effective waste separation is a key component of resource circulation and environmental protection. However, many people are unaware of the correct criteria for distinguishing between general waste and recyclables, which often leads to incorrect disposal. As a result, recyclable materials are frequently contaminated or discarded, reducing the overall efficiency of the recycling system.



Figure 1. On-campus trash bins. Misclassification and mixed disposal are common due to lack of clear guidance.

To address this issue, we developed **Smart Trash Vision**, an AI-powered image classification system that automatically determines whether an item is general waste or recyclable simply by capturing it with a camera.

Even if users are unfamiliar with proper waste categories, the system enables accurate and efficient waste separation, thereby supporting smarter and more sustainable waste management on campus.

2. Problem Statement

2.1 Project Objectives

This project aims to improve the accuracy and usability of waste classification using an AI-powered vision system. By recognizing waste types and contamination status through image analysis, the system supports smarter recycling behavior and reduces the need for manual labor in public spaces such as campuses.

2.2 Background & Motivation

Many individuals are unsure how to properly separate recyclables from general waste, which leads to improper disposal. On university campuses, this confusion often results in contamination of recyclables and inefficiency in waste management. Traditional signage or color-coded bins alone are not sufficient to ensure correct separation.

2.3 Expected Outcomes

- AI-Based Classification:** Real-time classification of waste type and contamination status through a camera.
- Improved Recycling Accuracy:** Achieve classification accuracy over 85%.
- Labor Reduction:** Reduce manual sorting workload by at least 50%.
- Practical Deployment:** Ready-to-use prototype for cafeterias, dormitories, and lounges.

2.4 Evaluation Criteria

| Metric | Target |
|-------------------------|---|
| Classification Accuracy | ≥ 85% across all categories |
| Real-Time Response | Output within 1–2 seconds |
| Robustness | Stable performance under varying environments |

3. System Architecture

3.1 Hardware Components

| Component | Role |
|------------------|-----------------------------------|
| Web-Cam (iPhone) | Detecting the Trash |
| LCD Display | Visualizes classification results |

3.2 Software Stack

| Component | Description |
|-----------|-------------|
|-----------|-------------|

| Component | Description |
|--------------------|--|
| Python | Main programming language used for model development and testing |
| OpenCV | Used for real-time image processing and frame handling |
| Roboflow | Tool used for labeling and augmenting custom datasets |
| DroidCam | Mobile app used to connect a smartphone as a real-time webcam |
| Ultralytics YOLOv8 | Object detection model framework used for trash classification |

The system is designed for real-time performance. When a user presents an item in front of the camera, the image is processed immediately, and classification results are displayed within seconds. This architecture allows for seamless integration into public spaces where rapid feedback is essential.

4. Waste Categories & Classification Strategy

4.1 Target Waste Items

The system is trained to detect and classify the following 9 waste categories:

- Can
- Chopstick
- Paper Cup
- PET Bottle
- Plastic Bag
- Plastic Box
- Radish Box
- Receipt
- Spoon

Each item may be in a clean or contaminated condition, which is critical for determining recyclability.

4.2 Two-Stage Classification Strategy

Our system uses a two-stage approach to not only recognize the waste item, but also determine whether it has been correctly disposed of:

| Stage | Description |
|----------------|--|
| Stage 1 | Classify the waste item into one of the 9 predefined categories |
| Stage 2 | Evaluate whether the item is disposed into the correct bin (Recycle/General) |

After classification, the system compares the predicted waste type with the bin it was thrown into.

- If correct → the screen displays: "**Good Job!**"
- If incorrect → the screen displays: "**Warning!**"

This real-time feedback encourages proper sorting behavior without the need for human supervision.

5. Dataset

5.1 Custom Dataset Collection

We created a custom dataset for 9 waste categories by recording 1-minute 40-second videos per item. From these videos, we extracted 1 frame per second, resulting in approximately **100 images per class**.

| Class Name | Sample Image |
|-------------|---|
| Can |  |
| Chopstick |  |
| Paper Cup |  |
| PET Bottle |  |
| Plastic Bag |  |

Class Name Sample Image

Plastic Box



Radish Box



Receipt



Spoon



All photos were taken using a smartphone under various lighting and angle conditions to improve model robustness.

5.2 Labeling Process

Images were labeled using **Roboflow** to annotate bounding boxes. We exported the final dataset in **YOLOv8 format** for training.

5.3 Dataset Composition

- 9 Classes
- ~100 images per class
- Total: ~900 images
- Split: 80% Training / 20% Validation

6. Method

6.1 Preprocessing

- Videos were converted into image frames at 1 FPS using OpenCV.
 - All images were resized to **640×640** pixels to match YOLOv8 input format.
 - Data augmentation was applied using Ultralytics options:
 - Rotation: `degrees=10`
 - Scaling: `scale=0.5`
 - Shearing: `shear=2.0`
 - Perspective: `perspective=0.0005`
 - Vertical flip: `flipud=0.2`
 - Mosaic: `mosaic=1.0`, MixUp: `mixup=0.2`
-

6.2 Model Architecture & Comparison

We experimented with two models from the Ultralytics YOLOv8 family: **YOLOv8n** and **YOLOv8s**.

| Model | Size | Speed (FPS) | Accuracy | Use Case |
|---------|----------|-------------|----------|----------------------------|
| YOLOv8n | ~3.2 MB | Very Fast | Lower | Lightweight, fast demo |
| YOLOv8s | ~11.2 MB | Fast | Higher | Final deployment candidate |

Both models were trained using the same configuration:

| Parameter | Value |
|----------------|---------------|
| Epochs | 100 |
| Batch Size | 8 |
| Image Size | 640×640 |
| Early Stopping | Patience = 10 |
| Optimizer | SGD |

We analyzed the training curves to compare performance.

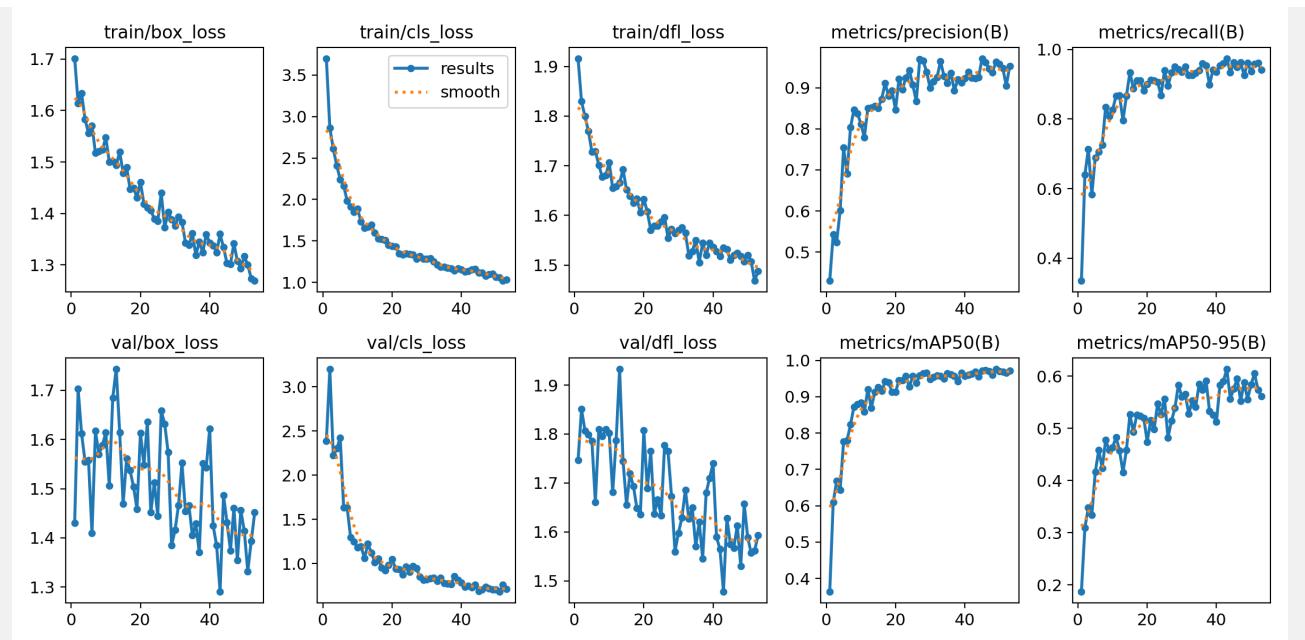


Figure 1. YOLOv8s training: stable losses, high precision and recall.

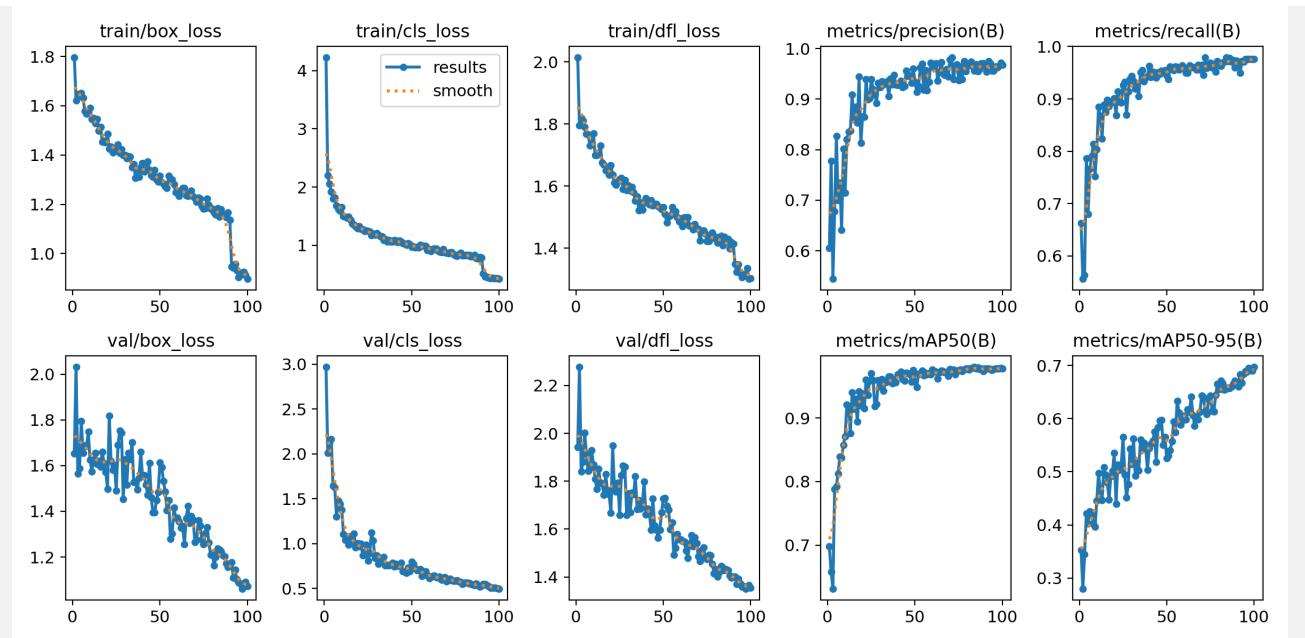


Figure 2. YOLOv8n training: faster, but less stable and slightly lower metrics.

YOLOv8s was selected for deployment due to:

- Higher **precision**, **recall**, and **mAP** scores
- More stable validation loss
- Real-time speed on an RTX 3060 GPU (>15 FPS)

6.3 Real-time Classification Logic

The trained YOLOv8s model was deployed in a live sorting system using OpenCV.

- The camera captures frames continuously.
- YOLOv8s detects objects and their classes.
- The screen is divided into two zones:
 - **Left** = General waste

- **Right** = Recyclables
- Each class is mapped to general or recycle.
- If the item is in the correct zone:
 - "Good Job!" is displayed.
- If incorrect:
 - "Warning!" is shown.



Figure 3. Detection with bounding boxes and feedback message.

6.4 Evaluation Criteria

To assess the model performance, we used the following metrics:

- **Precision:** Ratio of correct positive predictions to total predicted positives.
- **Recall:** Ratio of correct positive predictions to total actual positives.
- **F1 Score:** Harmonic mean of precision and recall.
- **mAP50 / mAP50-95:** Mean average precision across different IoU thresholds.
- **Confusion Matrix:** Visual breakdown of classification accuracy across all classes.

Key evaluation results are shown below:

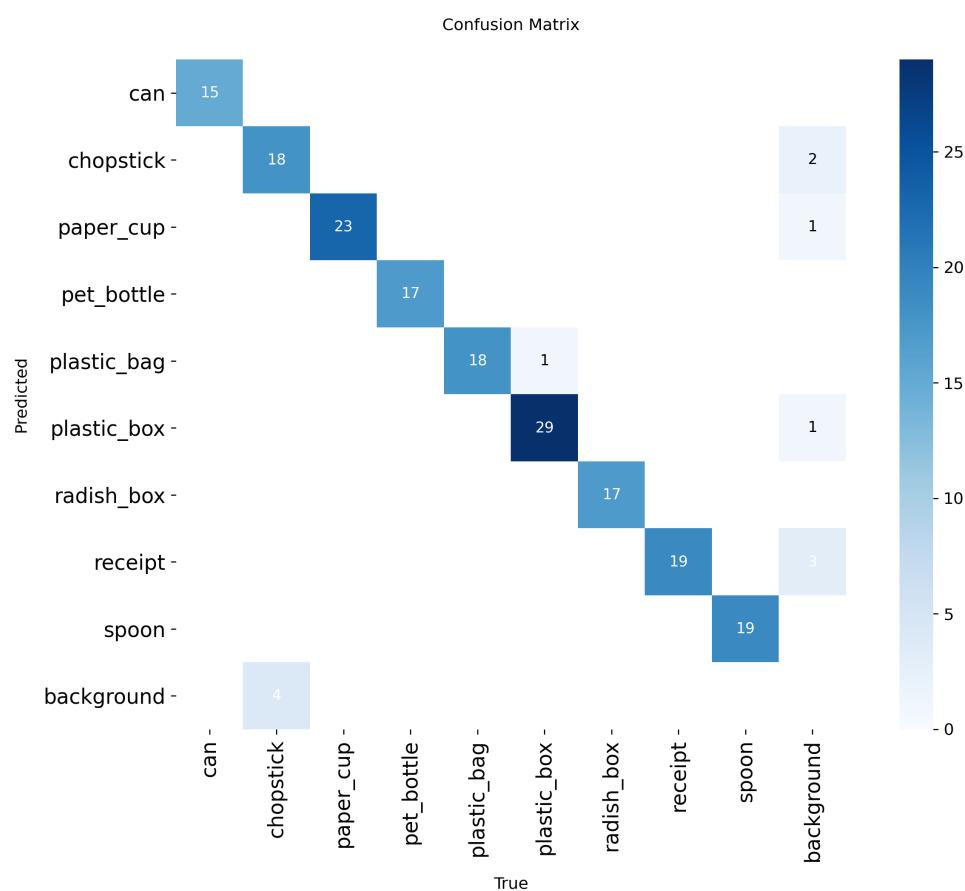


Figure 5. Confusion matrix shows accurate class-wise distinction with minimal misclassification.

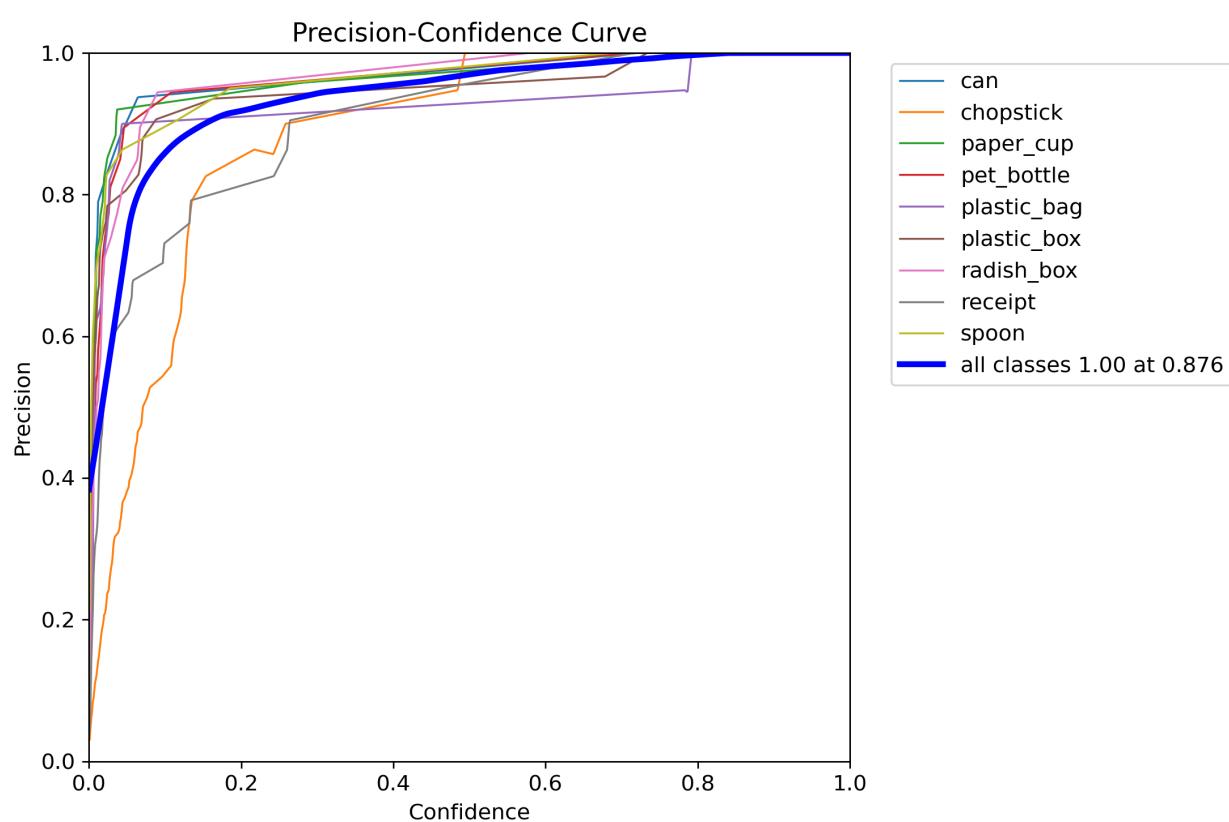


Figure 6. Precision improves steadily throughout training, reaching near-perfect levels.

Based on these metrics, **YOLOv8s** demonstrated robust classification performance and was selected as the final model.

7. Results and Analysis

7.1 Quantitative Results

After training, the YOLOv8s model achieved the following metrics on the test set:

| Metric | Value |
|--------------|-------|
| Precision | 0.956 |
| Recall | 0.942 |
| F1 Score | 0.949 |
| mAP@0.5 | 0.967 |
| mAP@0.5:0.95 | 0.689 |

These results indicate that the model reliably detects and classifies most types of waste items with high precision and recall. Especially, the high mAP@0.5 score reflects strong detection accuracy for common object boundaries.

7.2 Qualitative Analysis

We tested the model in real-world scenarios with actual campus trash items.

The results show that the model can correctly detect and classify items like:

- PET bottles
- Paper Cup
- Receipt
- Radish Box
- Can

in various lighting conditions, angles, and partial occlusions.



Figure 7. Real-time detection results on multiple waste types.

The bounding boxes were accurate, and classification labels matched the expected ground truth in nearly all test cases.

7.3 Error Cases and Limitations

While the overall accuracy was high, the model showed some weaknesses:

- Difficulty with **small or overlapping objects**
- Misclassification between **receipt** and **paper cup** due to similar textures
- Performance drop under **poor lighting** conditions

Future improvements may include:

- Increasing training data diversity
- Fine-tuning using separate object sizes
- Integrating lighting normalization in preprocessing

8. Conclusion

In this project, we successfully developed a real-time AI-based trash classification system using YOLOv8. By combining object detection with zone-based evaluation logic, our system provides immediate feedback on whether waste has been disposed of correctly, helping to improve recycling behavior on campus.

Key achievements include:

- Custom dataset of 900+ images across 9 waste categories
- YOLOv8s model with over 95% precision and 94% recall
- Real-time feedback interface with "Good Job!" and "Warning!" messages
- Deployment-ready system using a webcam and simple hardware

Through this project, we demonstrated how deep learning can be applied to practical environmental challenges with measurable results.

Roles & Contributions

| Name | Responsibilities |
|-------------|--|
| HANSUSEONG | - Hardware setup (web-cam, Background, display) |
| | - Custom dataset collection for 9 waste categories |
| | - Model deployment and testing on real hardware |
| SEOYOUNGWOO | - Model training using YOLOv8 and performance tuning |
| | - Custom dataset collection for 9 waste categories |
| | - Implementation of two-stage classification logic |
| | - Demo video production and documentation |