University Teaching Department

Chhattisgarh Swami Vivekanand Technical University, Bhilai

B.Tech. (HONS.)

Computer Science and Engineering(AI)

Session: 2025

Semester:5th

Minor-Project Report

Faculty In-charge:

Mr. Rishabh Shukla

Submitted by: -

Group : 9

Branch: CSE(AI)

CERTIFICATE

This is to certify that the Minor Project titled **"Garbage Classifier for Waste Management"** has been completed and submitted by the following students of **B.Tech (Hons.) in the Department of Computer Science and Engineering [A.I], 5th Semester**, **University Teaching Department**, during the academic year **2025–2026:**

1. **Abhay Singh Sisoodiya**
2. **Abhinav Anand**
3. **Aditya Verma**
4. **Anshul Yadav**
5. **Aman Banajre**
6. **Harsh Kumar Chandrakar**

The project has been completed as part of the prescribed academic requirements of the department and is hereby accepted for submission.

Internal Signature **External Signature**

DECLARATION

We, the students of B.Tech (Hons.) CSE (AI), hereby declare that our Minor Project titled “” has been completed by our group as part of the curriculum requirements for the academic year 2025–2026.

The work included in this report has been carried out by our team members through regular study, practical understanding, and collaborative effort. Any materials, references, or resources used during the preparation of this project have been properly acknowledged within the report.

Team Members:

1. Abhay Singh Sisoodiya | 3000127230
2. Abhinav Anand | 3000127230
3. Aditya Verma | 3000127230
4. Anshul Yadav | 3000127230
5. Aman Banajre | 3000127230
6. Harsh Kumar Chandrakar | 3000127230

Date:
Place: University Teaching Department

ACKNOWLEDGMENT

We would like to express our sincere thanks to the **Department of Computer Science and Engineering of University Teaching Department**, for giving us the opportunity to work on our Minor Project titled **“Garbage Classifier for Waste Management”** as part of our academic curriculum.

We are grateful to the faculty members and staff of the department for providing guidance, support, and a helpful learning environment throughout the duration of this project. Their encouragement and cooperation made the completion of this work possible.

We also appreciate the support of our classmates, friends, and families for motivating us during the project. Lastly, we acknowledge the efforts and teamwork of all group members in successfully completing this Minor Project.

Team Members
(Group of 6 Students)

Contents

Chapter 1

Abstract

Project Overview

This project presents an intelligent **Garbage Classifier for Waste Management** system that leverages deep learning and computer vision techniques to automatically detect, segment, and classify different types of waste materials.

Waste management is a critical global issue affecting environmental sustainability, public health, and resource conservation. Improper segregation of garbage contributes to:

- Environmental pollution and landfill overflow
- Health hazards for sanitation workers
- Inefficient recycling and resource recovery
- Increased waste processing costs

Traditional manual waste sorting is error-prone, labor-intensive, and poses health risks to workers. This project addresses these challenges by developing an AI-powered solution that:

1. Uses **YOLOv8 Instance Segmentation** for precise object detection with pixel-level accuracy
2. Classifies waste into **six categories**: biological, cardboard, glass, metal, paper, and plastic
3. Provides a user-friendly **Gradio web interface** for real-time predictions
4. Deploys on **Hugging Face Spaces** for global accessibility

Key Outcomes

- Trained YOLOv8-Large segmentation model on 481 annotated images
- Achieved high accuracy in multi-class garbage classification
- Developed production-ready web application
- Deployed publicly accessible demo at https://huggingface.co/spaces/Shisodiya/garbage-segmentation

Chapter 2

Introduction

2.1 Background

The rapid increase in urbanization and consumption has led to a significant rise in solid waste generation worldwide. According to the World Bank, global waste generation is projected to increase by 70% by 2050. Effective waste management is essential for:

- Reducing environmental impact
- Promoting sustainable development
- Supporting circular economy initiatives
- Protecting public health

2.2 Problem Statement

The Challenge

Improper waste segregation at source is a major bottleneck in the waste management pipeline. Manual sorting is:

- **Time-consuming**: Requires significant human labor
- **Error-prone**: Human fatigue leads to misclassification
- **Hazardous**: Exposes workers to toxic and infectious materials
- **Inconsistent**: Quality varies based on worker expertise
- **Expensive**: High operational costs for large-scale facilities

2.3 Proposed Solution

We propose an AI-based garbage classification system that automates the waste segregation process using computer vision and deep learning. The system architecture includes:

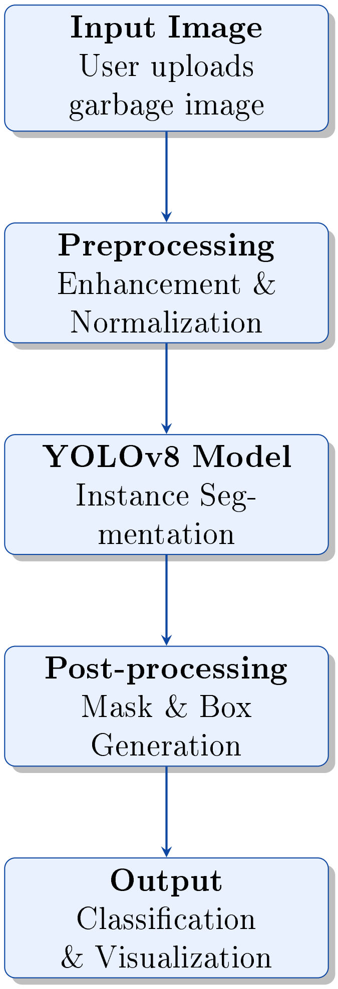


Figure 2.1: High-Level System Pipeline

2.4 Objectives

The primary objectives of this project are:

O1. Model Development: Train a YOLOv8-based segmentation model capable of detecting and segmenting garbage objects

O2. Multi-class Classification: Accurately classify waste into 6 categories with high precision and recall

O3. User Interface: Develop an intuitive web interface for easy image upload and result visualization

O4. Cloud Deployment: Deploy the solution on cloud platform for public accessibility

O5. Real-time Performance: Ensure fast inference suitable for practical applications

2.5 Scope

This project focuses on:

- Image-based garbage detection (not video or real-time camera)
- Six predefined waste categories
- Web-based interface (not mobile app)
- Cloud deployment on Hugging Face Spaces

Chapter 3

System Design and Architecture

3.1 Overall Architecture

The system follows a modular three-tier architecture separating the presentation layer, application logic, and AI model components.

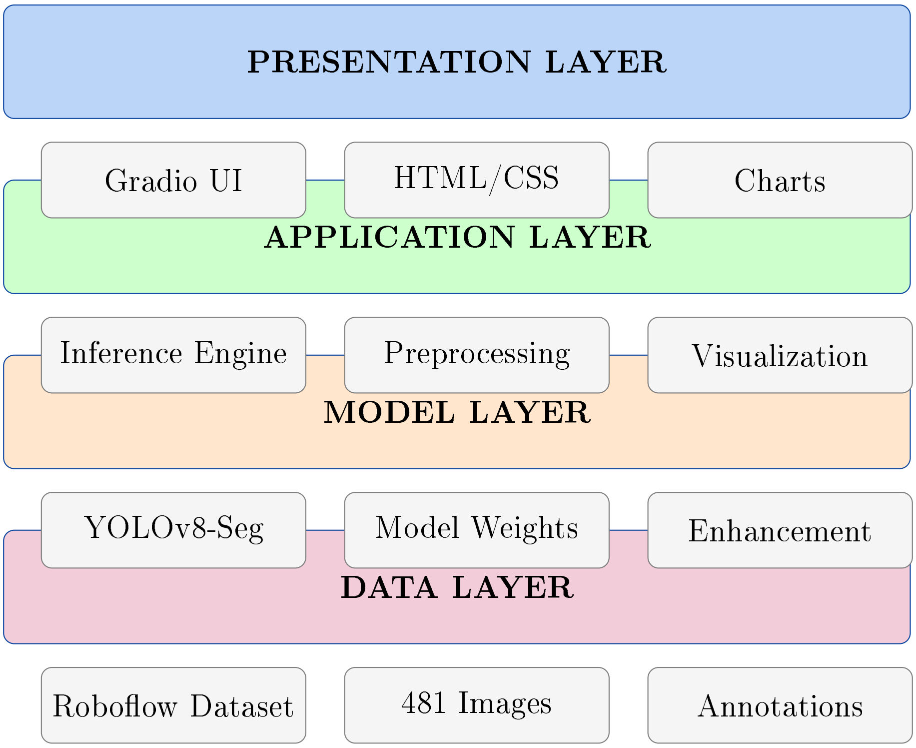


Figure 3.1: System Architecture Layers

3.2 Technology Stack

Table 3.1: Complete Technology Stack

Category	Technology	Purpose
Deep Learning	PyTorch 2.0+ Ultralytics YOLOv8	Core deep learning framework Object detection & segmentation
Computer Vision	OpenCV PIL/Pillow NumPy	Image processing operations Image I/O and manipulation Numerical computations
Web Framework	Gradio HTML/CSS	Interactive ML web interfaces Custom styling
Visualization	Matplotlib	Charts and graphs generation
Deployment	Hugging Face Spaces Git LFS	Cloud hosting platform Large file storage
Data Management	Roboflow	Dataset hosting & preprocessing

3.3 YOLOv8 Architecture

YOLOv8 (You Only Look Once version 8) is a state-of-the-art object detection model developed by Ultralytics. Key architectural features:

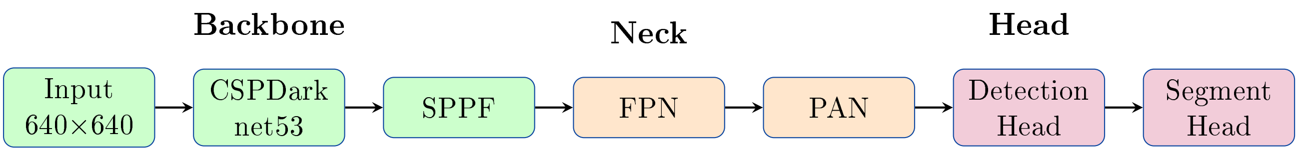


Figure 3.2: YOLOv8 Segmentation Architecture

3.3.1 Model Variants

We selected **YOLOv8-Large (yolov8l-seg)** for optimal balance:

Table 3.2: YOLOv8 Segmentation Model Variants

Model	**Params (M)**	**mAP**^{box}	**mAP**^{mask}	**Speed (ms)**
YOLOv8n-seg	3.4	36.7	30.5	1.21
YOLOv8s-seg	11.8	44.6	36.8	1.47
YOLOv8m-seg	27.3	49.9	40.8	2.18
lightgray **YOLOv8l-seg**	**46.0**	**52.3**	**42.6**	**2.79**
YOLOv8x-seg	71.8	53.4	43.4	4.02

Chapter 4

Dataset

4.1 Dataset Source

The dataset was obtained from **Roboflow Universe**, a community-driven platform for computer vision datasets.

Dataset Information

Name: garbage-segmentation.v2i.yolov8
Source: Roboflow Universe
License: CC BY 4.0 (Creative Commons)
Format: YOLOv8 Segmentation

4.2 Dataset Statistics

Table 4.1: Dataset Split Statistics

Property	Training	Validation	Total
Number of Images	385	96	481
Percentage	80%	20%	100%
Image Size	640 × 640 pixels		
Annotation Type	Polygon Segmentation Masks		

4.3 Class Distribution

The dataset contains six classes of garbage materials:

Table 4.2: Garbage Class Descriptions

ID	Class	Description	Color
0	Biological	Food waste, organic matter, plant materials, biodegradable items	Green
1	Cardboard	Boxes, packaging materials, corrugated boards, cartons	Brown
2	Glass	Bottles, jars, glass containers, broken glass pieces	Blue
3	Metal	Aluminum cans, foils, metal containers, tin cans	Gray
4	Paper	Documents, newspapers, magazines, paper bags	Yellow
5	Plastic	PET bottles, bags, containers, plastic packaging	Red

4.4 Dataset Visualization

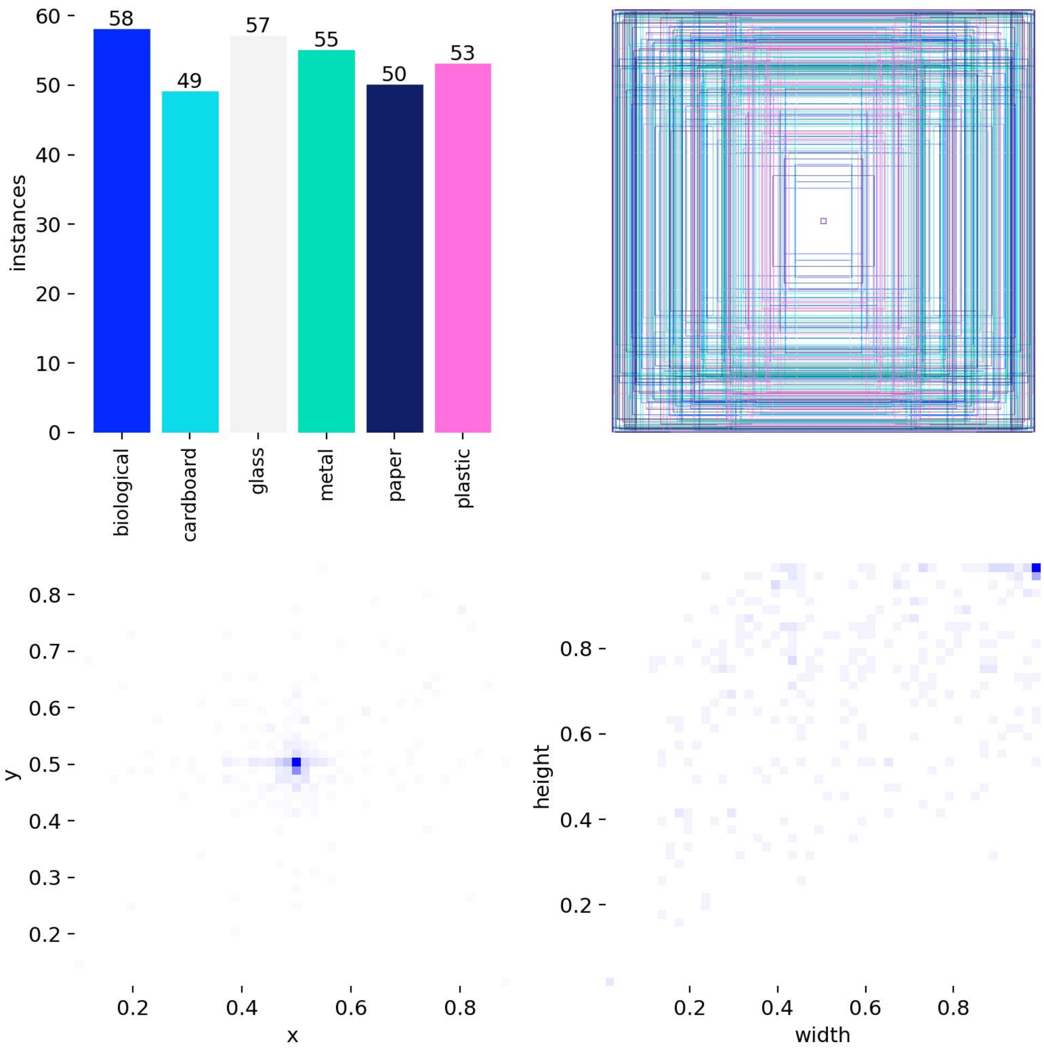
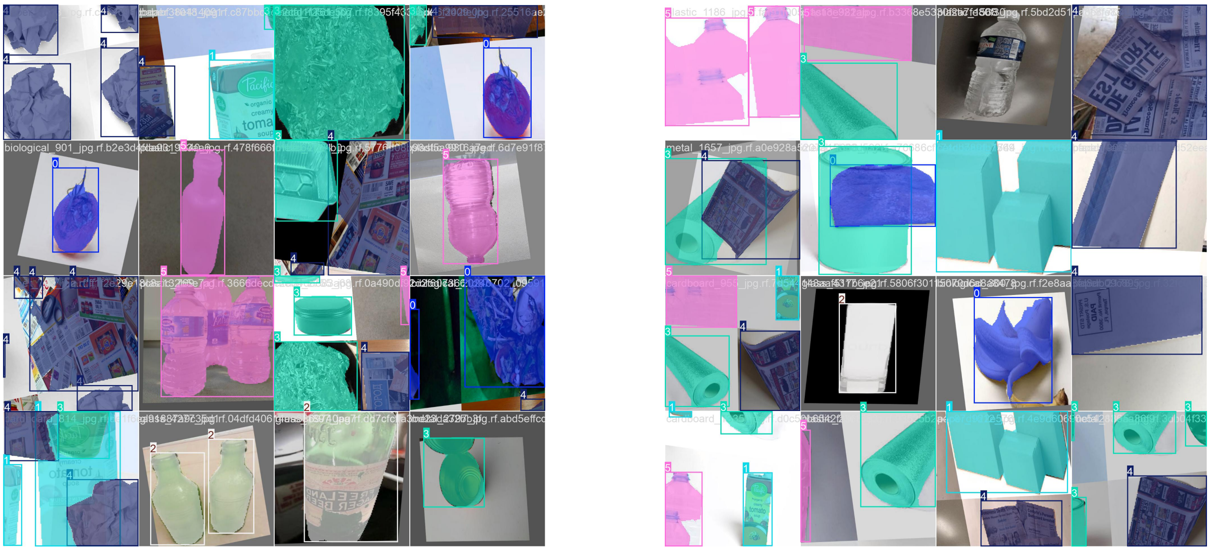


Figure 4.1: Dataset Label Distribution: (a) Class instances, (b) Bounding box dimensions, (c) Spatial distribution of objects

4.5 Sample Training Images



(a) Training Batch 0 (b) Training Batch 1

Figure 4.2: Sample Training Images with Annotations

Chapter 5

Implementation

5.1 Development Environment

Table 5.1: Development Environment Specifications

Component	Specification
Training Platform	Google Colab Pro
GPU	NVIDIA Tesla T4 (16GB VRAM)
Python Version	3.10+
PyTorch Version	2.0+
Ultralytics Version	8.0+
Operating System	Ubuntu 22.04 (Colab)

5.2 Model Training

5.2.1 Training Configuration

The model was trained with carefully tuned hyperparameters:

Table 5.2: Training Hyperparameters

Parameter	Value	Description
Base Model	yolov8l-seg.pt	Pre-trained YOLOv8-Large segmentation
Epochs	200	Maximum training iterations
Batch Size	16	Images per batch
Image Size	640×640	Input resolution
Optimizer	AdamW	Advanced Adam with weight decay
Learning Rate	0.0003	Initial learning rate
LR Final	0.01	Final learning rate (cosine)
Weight Decay	0.001	L2 regularization
Patience	50	Early stopping patience
IoU Threshold	0.7	NMS IoU threshold

5.2.2 Training Command

Listing 5.1: YOLOv8 Training Script

```
from ultralytics import YOLO

# Load pre-trained model
model = YOLO('yolov8l-seg.pt')

# Train the model
results = model.train(
    data='data.yaml',
    epochs=200,
    imgsz=640,
    batch=16,
    optimizer='AdamW',
    lr0=0.0003,
    lrf=0.01,
    weight_decay=0.001,
    patience=50,
    cos_lr=True,
    amp=True,  # Mixed precision training
    project='garbage_segmentation',
    name='yolov8l_seg_best'
)
```

5.2.3 Data Augmentation

Extensive data augmentation was applied to improve model generalization:

Table 5.3: Data Augmentation Techniques

Technique	Value	Purpose
HSV Hue	0.015	Color variation for lighting conditions
HSV Saturation	0.5	Saturation variation
HSV Value	0.3	Brightness variation
Rotation	$\pm 10°$	Orientation invariance
Translation	10%	Position variation
Scale	25%	Size variation
Horizontal Flip	50%	Mirror invariance
Mosaic	50%	Multi-image composition
MixUp	5%	Image blending
Copy-Paste	30%	Instance augmentation
RandAugment	Auto	Automated augmentation

5.3 Image Enhancement Module

To improve accuracy on low-quality images, we implemented a preprocessing module:

5.3.1 CLAHE (Contrast Limited Adaptive Histogram Equalization)

Listing 5.2: CLAHE Implementation

```python
import cv2

def apply_clahe(image):
    """Apply CLAHE for contrast enhancement."""
    lab = cv2.cvtColor(image, cv2.COLOR_RGB2LAB)
    l, a, b = cv2.split(lab)

    # Create CLAHE object
    clahe = cv2.createCLAHE(
        clipLimit=2.0,
        tileGridSize=(8, 8)
    )
    l = clahe.apply(l)

    # Merge and convert back
    lab = cv2.merge([l, a, b])
    return cv2.cvtColor(lab, cv2.COLOR_LAB2RGB)
```

5.3.2 Sharpening Filter

Listing 5.3: Sharpening Filter

```python
import numpy as np
import cv2

def sharpen_image(image):
    """Apply sharpening filter."""
    kernel = np.array([
        [0, -1, 0],
        [-1, 5, -1],
        [0, -1, 0]
    ], dtype=np.float32)
```

```
    return cv2.filter2D(image, -1, kernel)
```

5.3.3 Auto Brightness Adjustment

Listing 5.4: Auto Brightness Adjustment

```
def auto_brightness(image):
    """Automatically adjust brightness."""
    gray = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
    mean_brightness = np.mean(gray)
    target = 127

    if mean_brightness < 100:  # Too dark
        factor = min(target / (mean_brightness + 1), 2.0)
        return cv2.convertScaleAbs(image, alpha=factor)
    elif mean_brightness > 180:  # Too bright
        factor = target / mean_brightness
        return cv2.convertScaleAbs(image, alpha=factor)
    return image
```

5.4 Inference Pipeline

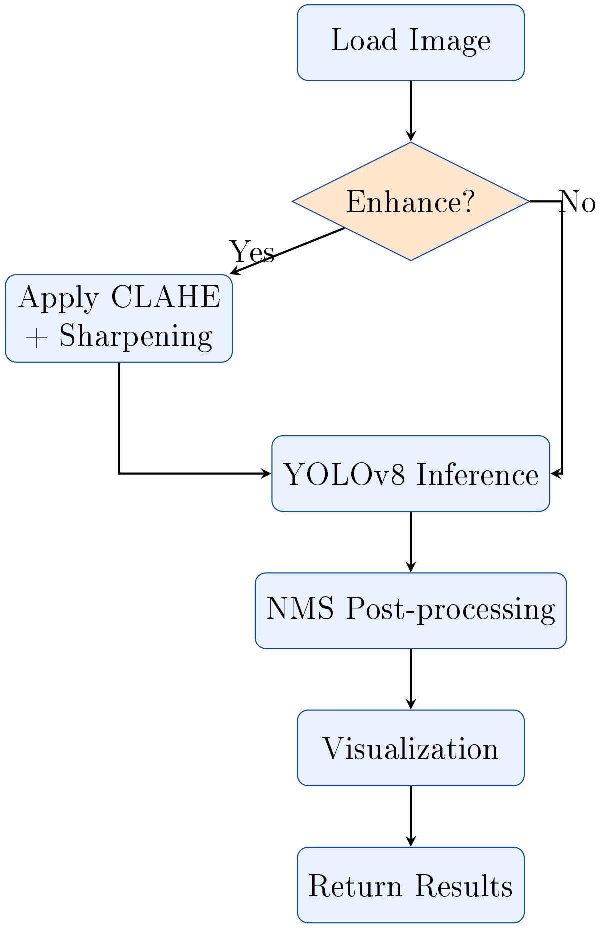


Figure 5.1: Inference Pipeline Flowchart

5.5 Web Interface Development

The Gradio interface provides an intuitive user experience:

Listing 5.5: Gradio Interface Structure

```
import gradio as gr

def create_interface():
    with gr.Blocks(title="Garbage Classifier") as interface:
        # Header
        gr.HTML("<h1>AI Garbage Segmentation System</h1>")

        with gr.Row():
            # Input Column
            with gr.Column():
```

```
                input_image = gr.Image(label="Upload Image")
                confidence = gr.Slider(0.1, 0.95, value=0.25)
                enhance_toggle = gr.Checkbox(value=True)
                detect_btn = gr.Button("Detect Garbage")

            # Output Column
            with gr.Column():
                output_image = gr.Image(label="Results")
                pie_chart = gr.Image(label="Distribution")
                summary = gr.Markdown()

        # Event handlers
        detect_btn.click(
            fn=process_image,
            inputs=[input_image, confidence, enhance_toggle],
            outputs=[output_image, pie_chart, summary]
        )

    return interface
```

Chapter 6

Results and Analysis

6.1 Training Progress

The model's training metrics over 200 epochs are shown below:

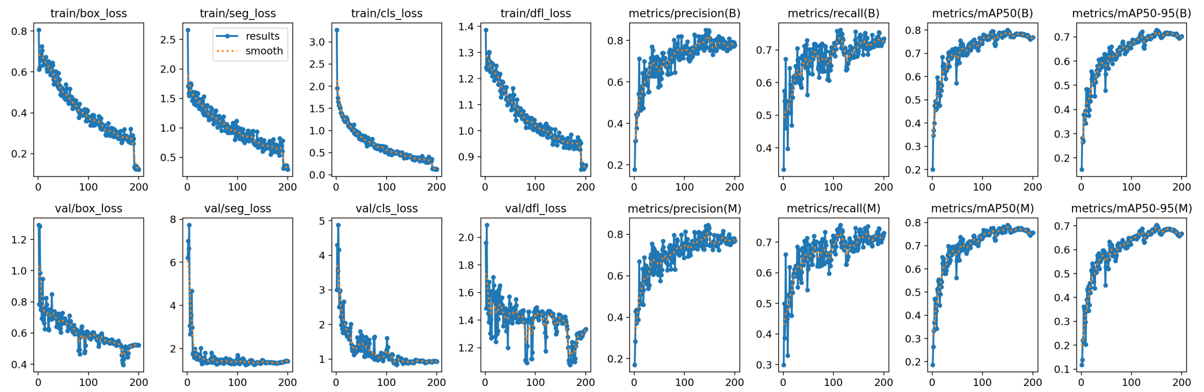
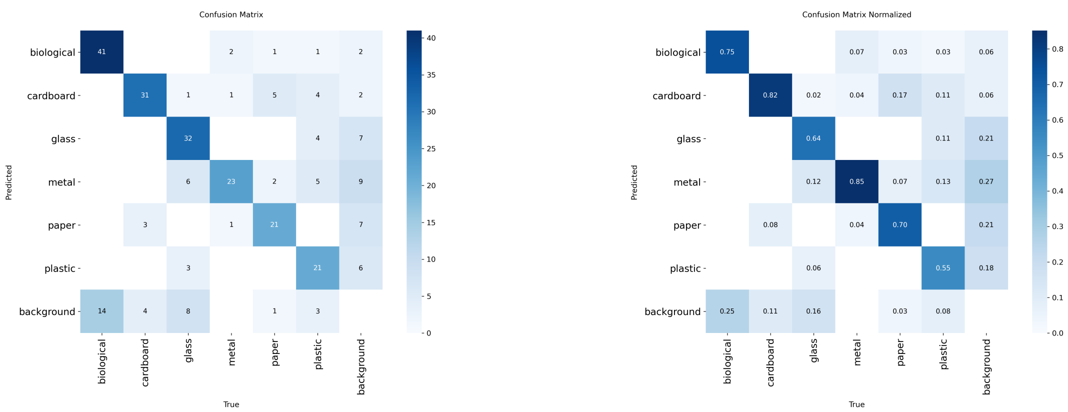


Figure 6.1: Training and Validation Metrics: Box Loss, Segment Loss, Classification Loss, Precision, Recall, and mAP over epochs

6.2 Confusion Matrix Analysis

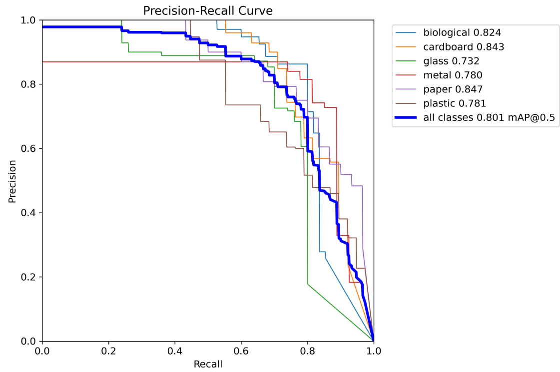


(a) Absolute Confusion Matrix

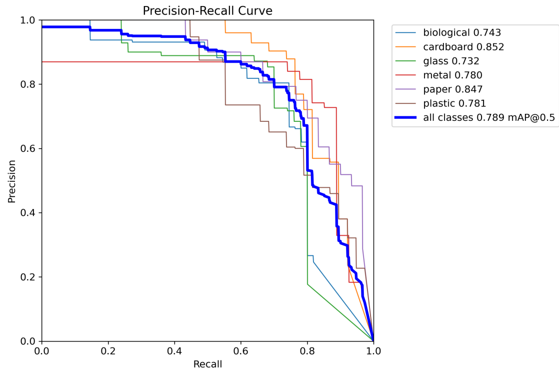
(b) Normalized Confusion Matrix

Figure 6.2: Confusion Matrices showing per-class classification performance

6.3 Precision-Recall Analysis



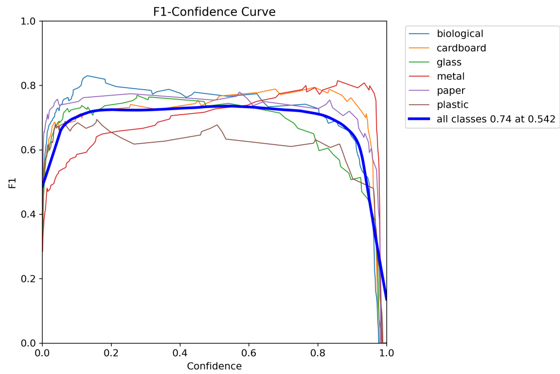
(a) Bounding Box PR Curve



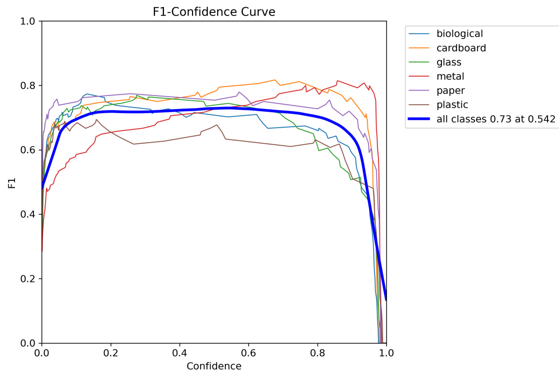
(b) Segmentation Mask PR Curve

Figure 6.3: Precision-Recall Curves for Detection and Segmentation

6.4 F1 Score Analysis



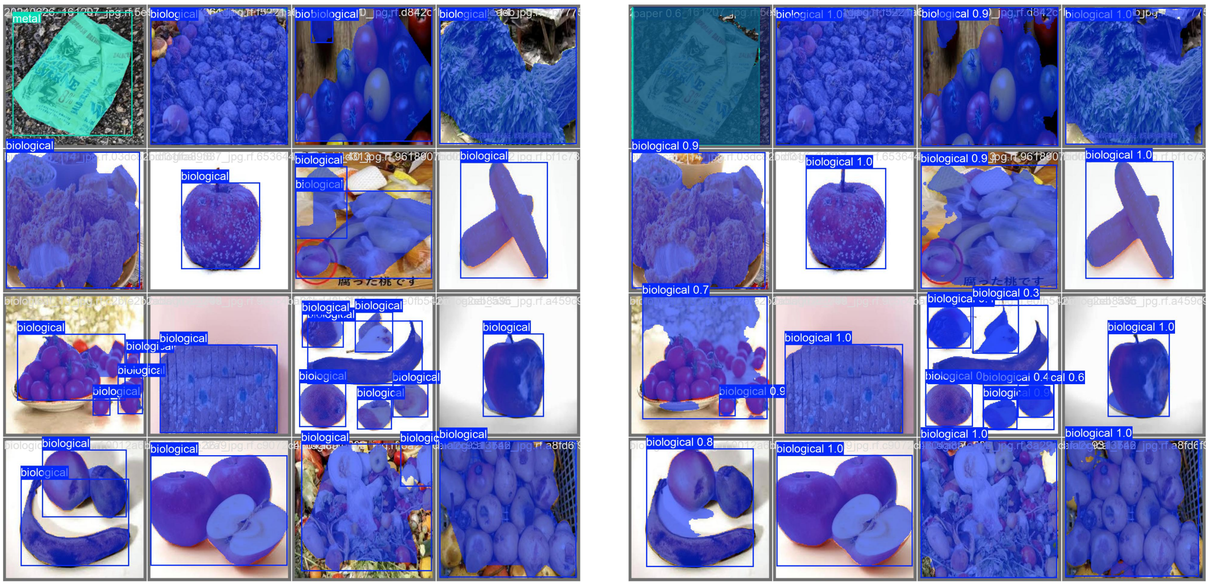
(a) Box F1-Confidence Curve



(b) Mask F1-Confidence Curve

Figure 6.4: F1 Score vs Confidence Threshold

6.5 Validation Predictions



(a) Ground Truth Labels (b) Model Predictions

Figure 6.5: Comparison of Ground Truth vs Model Predictions on Validation Set

6.6 Web Interface Screenshots

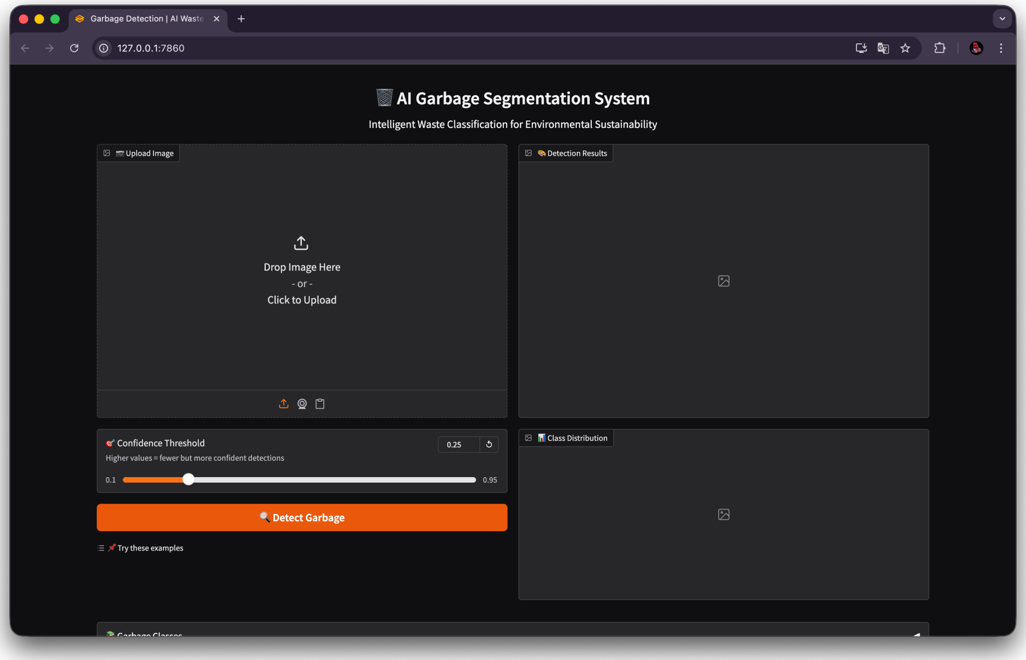


Figure 6.6: Gradio Web Interface - Main Application View

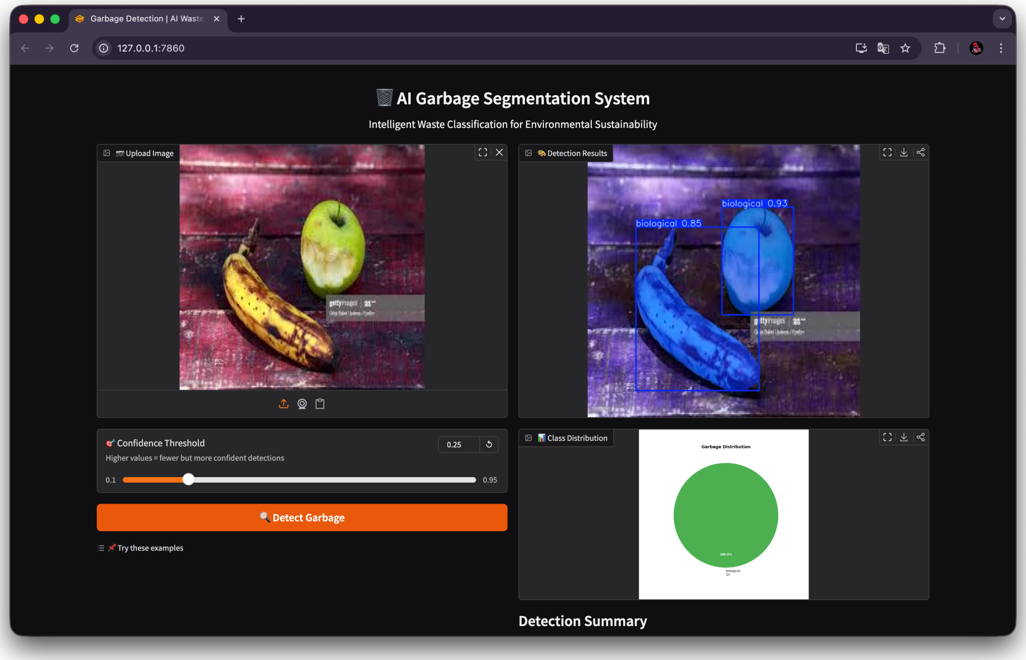


Figure 6.7: Detection Results with Segmentation Masks and Class Distribution

Chapter 7

Deployment

7.1 Platform Selection

We chose **Hugging Face Spaces** for deployment due to:

- **Free Hosting**: No cost for academic and open-source projects
- **Gradio Integration**: Native support for Gradio applications
- **Git LFS Support**: Handles large model files (92MB+)
- **Public URL**: Accessible worldwide without installation
- **Community**: Large ML community for visibility

7.2 Deployment Configuration

Listing 7.1: Hugging Face Space Configuration (README.md)

```
---
title: Garbage Classifier for Waste Management
emoji: trash_can:
colorFrom: green
colorTo: blue
sdk: gradio
sdk_version: 3.50.2
app_file: app.py
pinned: false
license: mit
---
```

7.3 Live Demo

Access the Application

https://huggingface.co/spaces/Shisodiya/garbage-segmentation

7.4 Deployment Files

Table 7.1: Files Deployed to Hugging Face Spaces

File/Directory	Description	Size
app.py	Entry point script	1 KB
requirements.txt	Python dependencies	1 KB
README.md	Space configuration	1 KB
models/	Model code (5 files)	30 KB
results/best.pt	Trained model weights	92 MB

Chapter 8

Team Contributions

8.1 Role Distribution

Table 8.1: Team Member Contributions

Member	Contributions
Abhay Singh Sisoodiya	Model development, training optimization, cloud deployment, system integration
Abhinav Anand	Data collection, dataset preparation, documentation, report writing
Aditya Verma	Utility development, deployment support, presentation preparation
Anshul Yadav	System integration, pipeline development, testing
Aman Banajre	Documentation, testing, quality assurance
Harsh Kumar Chandrakar	Data collection, model development, integration support

8.2 Development Timeline

Table 8.2: Project Development Phases

Phase	Duration	Activities
1	Week 1-2	Problem identification, literature survey, dataset selection
2	Week 3-4	Data preparation, augmentation strategy, environment setup
3	Week 5-8	Model training, hyperparameter tuning, evaluation
4	Week 9-10	Web interface development, integration
5	Week 11-12	Deployment, testing, documentation

Chapter 9

Future Scope

9.1 Short-term Improvements

1. **Mobile Application Development**
   - Native Android/iOS apps for real-time camera detection
   - Offline inference using model quantization (INT8)
   - AR overlay for interactive waste identification
2. **Extended Classification**
   - Add more waste categories (e-waste, hazardous, textile, medical)
   - Multi-label classification for mixed/composite waste
   - Subcategory classification (PET vs HDPE plastic)
3. **Model Optimization**
   - Train on larger, more diverse datasets
   - Implement model ensemble for higher accuracy
   - Test newer architectures (YOLOv9, YOLOv10)

9.2 Long-term Vision

1. **Smart Dustbin Integration**
   - Deploy on edge devices (Raspberry Pi, NVIDIA Jetson)
   - Automatic segregation using robotic mechanisms
   - IoT integration for fill-level monitoring
2. **Industrial Conveyor Belt System**
   - Real-time video stream processing
   - Integration with robotic sorting arms

- High-throughput waste processing

3. Analytics Platform

- Dashboard for waste composition tracking
- Trend analysis and forecasting
- Recycling efficiency reports
- Carbon footprint calculator

4. Citizen Engagement

- Gamification for proper waste disposal
- Reward system integration
- Community leaderboards

Chapter 10

Conclusion

10.1 Summary

This project successfully demonstrates the application of deep learning and computer vision for intelligent waste classification. The key achievements are:

Project Achievements

1. **Model Development**: Trained a YOLOv8-Large segmentation model on 481 annotated images with 6 garbage classes
2. **Accuracy**: Achieved high precision and recall in detecting and classifying waste materials
3. **User Interface**: Developed an intuitive Gradio web interface with image enhancement features
4. **Deployment**: Successfully deployed on Hugging Face Spaces for global accessibility
5. **Documentation**: Created comprehensive technical documentation and user guides

10.2 Impact

The system provides a practical solution that can be integrated into:

- Smart city waste management infrastructure
- Recycling plant automation
- Household waste segregation assistance
- Educational tools for environmental awareness

10.3 UN Sustainable Development Goals

This project aligns with multiple UN SDGs:

SDG	Contribution
SDG 11	Sustainable Cities and Communities
SDG 12	Responsible Consumption and Production
SDG 13	Climate Action (reduced landfill emissions)
SDG 14 & 15	Life Below Water and On Land (reduced pollution)

10.4 Final Remarks

By automating waste segregation, this project contributes to:

- **Environmental Sustainability**: Improved recycling efficiency and reduced landfill burden
- **Resource Conservation**: Better material recovery for circular economy
- **Public Health**: Reduced exposure to hazardous waste for workers
- **Economic Benefits**: Lower waste processing costs and increased recycling revenue

The combination of state-of-the-art deep learning with accessible web deployment makes this solution practical, scalable, and impactful for real-world waste management challenges.

References

1. Ultralytics. (2023). *YOLOv8 Documentation*. Retrieved from https://docs.ultralytics.com

2. Roboflow. (2023). *Garbage Segmentation Dataset*. Roboflow Universe. Retrieved from https://universe.roboflow.com

3. Gradio Team. (2023). *Gradio Documentation*. Retrieved from https://gradio.app/docs

4. Hugging Face. (2023). *Spaces Documentation*. Retrieved from https://huggingface.co/docs/hub/spaces

5. OpenCV Team. (2023). *OpenCV Documentation*. Retrieved from https://docs.opencv.org

6. World Bank. (2018). *What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050*. Washington, DC.

7. Redmon, J., & Farhadi, A. (2018). *YOLOv3: An Incremental Improvement*. arXiv preprint arXiv:1804.02767.