YOLOv8-Based Waste Detection System for Recycling Plants: A Deep Learning Approach

Meet Shroff
Department of Computer Engineering,
Devang Patel Institute of Advance Technology and Research, Charotar University of Science and Technology,
Changa, Gujarat, India
meetshroff1762@gmail.com

Akshat Desai
Department of Computer Science & Engineering,
Devang Patel Institute of Advance Technology and Research, Charotar University of Science and Technology,
Changa, Gujarat, India
akshat.desai.754@gmail.com

Dr. Dweepna Garg
Department of Computer Engineering,
Devang Patel Institute of Advance Technology and Research, Charotar University of Science and Technology,
Changa, Gujarat, India

Abstract*—Waste management is increasingly attracting attention due to its role in smart and sustainable development, especially in developed and developing nations. This system consists of a series of interconnected processes that perform various complex functions. Deep learning has recently attracted interest as an alternative computational method to solve various waste classification challenges. Many researchers have focused on this area, yielding significant research results in recent years. Although several in-depth investigations have been conducted on waste detection and classification, the WaRP dataset was created specifically to train and evaluate the proposed algorithms using industrial data from conveyor belt of waste recycling plant. Surprisingly, no research has explored the application of the YOLOv8 model to solve the waste management problem using the WaRP dataset. This experiment makes a notable contribution by detecting waste through a pyramid and direct prediction method, which differs from the traditional model based on anchor boxes. Through experimentation with different YOLOv8 model weights, we found that YOLOv8s provides relatively good results with smaller dataset and lower processing time. On the other hand, YOLOv8l achieves a higher mAP50 value of about 59% on the same dataset, but at the cost of high inference time.

Keywords: YOLOv8; Deep Learning; Waste Detection; Neural Networks; WaRP Dataset;

I. INTRODUCTION

Global challenges related to globalization, urbanization and a rapidly growing population have brought environmental pollution into the spotlight. While the world's population is growing at an alarming rate, our environment is suffering unprecedented degradation. In a recent report by the Energy and Resources Institute (TERI), struggling with a growing population and rapid development, India produced a whopping 279.5 million tonnes of waste in 2022, which included both municipal and industrial waste [1]. This massive generation of waste has far-reaching effects affecting the environment, the economy and public health. Inadequate waste management practices have exacerbated environmental degradation, leading to a global shift towards developing smart cities with efficient urban waste management systems. Recycling plays a key role in this change, providing opportunities for innovative research and sustainable business models.

In the midst of these advances, however, there is an urgent concern - the need for accurate waste sorting based on biodegradability. In India, a diverse and complex country, implementation of effective waste classification faces enormous challenges such as low public awareness and regional differences in waste classification standards. Relying solely on manual tree sorting exacerbates these challenges, leading to high labour costs and inefficiencies and risks to human health. To solve these problems, India is adopting advanced technologies like machine vision and machine learning. These innovations promise to revolutionize waste management by improving sorting accuracy, improving cost-effectiveness, and promoting environmental sustainability. Deep learning models such as YOLOv8 are at the forefront of this change, providing real-time object detection capabilities, especially for waste classification. This work presents an innovative approach that uses YOLOv8 for waste sorting, taking advantage of its accuracy and real-time performance. By embracing the effectiveness of deep learning in waste management and creating a valuable body of knowledge, this effort is an important step towards a more sustainable and effective waste management paradigm. Its purpose is to secure our environment, economy and public health for future generations.

The important contributions that are presented in this paper are as follows:

- We have proposed a unique YOLOv8-based deep convolutional neural network model for waste segregation.
- Our model is trained to classify and detect waste in 28 distinct classes.
- Instead of using the usual anchor box approach, we have opted a model which features pyramid and direct prediction method.
- This approach boosts detection speed, making our model suitable for real-time applications.
- Our research is a unique addition to the field of waste segregation using the YOLOv8 model, which has relatively few existing studies.

The paper's structure is as follows: Section 2 offers an introduction to the relevant methods. Section 3 details the methodology. In Section 4, an extensive analysis of experiments and their results are presented. Lastly, Section 5 wraps up the paper with a conclusion and outlines potential future directions.

II. LITERATURE SURVEY

YOLO, or You Only Look Once, is a real-time object detection algorithm that has revolutionized the field of computer vision. It`s fast, efficient and accurate, making it a great choice for many applications. YOLO works by dividing an image into a grid of cells. For each cell, YOLO predicts the probability that each feature class lies within the cell, as well as the coordinates of the object`s bounding box. YOLO does this in just one pass through the neural network, which is why it is so fast. YOLO has been shown to achieve optimal performance in various object detection tests. It is also widely used in real-world applications, such as video surveillance, self-driving cars, and augmented reality. YOLO can process images in real time, ideal for applications that require speed. It achieved high accuracy in various object detection tests. This is a lightweight model that can be deployed on a variety of devices, including mobile phones and embedded systems. It can be used to detect various objects including people, vehicles, animals, and traffic signs.

A. *YOLOv1*

YOLOv1 is the first version of the YOLO object detection algorithm. Released in 2016, this algorithm was the first real-time object detection algorithm to achieve the highest accuracy. YOLOv1 introduced a ground-breaking approach to object detection by simultaneously predicting bounding boxes for multiple object classes within a grid. It divided the input image into an S×S grid and predicted B bounding boxes per grid cell, each with confidence scores and class probabilities. The model architecture featured 24 convolutional layers followed by two fully-connected layers, leveraging 1×1 convolutions to manage parameters effectively. YOLOv1's training involved pre-training initial layers on [2] ImageNet data and fine-tuning with [3] PASCAL VOC datasets at a higher resolution [4]. It employed a unique loss function that weighted localization, confidence, and classification errors, aiming for accurate object detection. YOLOv1 excelled in real-time performance but faced limitations in handling nearby objects, objects with uncommon aspect ratios, and learning fine-grained features due to down-sampling layers.

B. *YOLOv2*

In 2017, YOLOv2, a significant advancement in object detection, expanded its wide-range detection capabilities to 9,000 categories. The basic architecture used by YOLOv2 is called Darknet-19 [5]. Key enhancements include batch normalization on convolutional layers to improve convergence and regularization. It introduces a high-resolution classifier by fine-tuning ImageNet to 448 x 448 resolutions, thereby improving performance on high-resolution inputs [6]. YOLOv2 adopts a fully convolutional architecture, eliminating dense layers for efficiency. Anchor boxes are used to predict bounding boxes, with multiple anchor points per grid cell to predict coordinates and layers. Dimensional clustering optimized the priorities via k-means clustering, selecting five priorities to balance. YOLOv2 directly predicts the coordinates corresponding to the grid cells, generating five bounding boxes for each cell. It has improved features by using multi-scale training and transfer layer to ensure robustness to different input sizes. [6] These improvements pushed YOLOv2 to an impressive average accuracy of 78.6% on [3]PASCAL VOC2007, surpassing YOLOv1's 63.4%. The Darknet-19 architecture has 19 convolutional layers [5], 1x1 convolutional layers for parameter reduction and batch normalization for regularization, delivering both speed and performance.

C. *YOLOv3*

YOLOv3, introduced in 2018, represents a significant evolution in real-time object detection. It introduces important changes compared to YOLOv2, including prediction of four bounding box coordinates (tx, ty, tw, th) as well as objective scores using logistic regression. YOLOv3 assigns a unique anchor box to each underlying fact object, and if left unspecified, only classification will be lost. Class prediction was converted to binary cross-entropy for multi-label classification, allowing multiple labels for a box. [7] YOLOv3 adopted a larger base architecture called Darknet-53 [8], which consists of 53 convolutional layers with residual connections and step convolutions. A modified SPP (Spatial Pyramid Pooling) block has been added to the backbone, improving performance. Multi-scale prediction has been introduced, providing finer object detail and better detection of small objects. The architecture also uses k-means clustering for the sections before the anchor box, using three priors for different scales. [7] YOLOv3 achieved industry-leading results on the COCO dataset, with YOLOv3-spp showing an AP of 36.2% and AP50 of 60.6% at 20 FPS, marking a significant advancement in object detection of real-time images.

D. *YOLOv4*

YOLOv4, which introduced its open-source, real-time, one-shot object detection philosophy in April 2020, was quickly adopted as the official YOLOv4 due to its significant improvements. He sought a balance between innovations classified as “bags of freebies” (BoF) and “bags of specialties” (BoS). BoF changed the training strategy, increasing costs without affecting inference time through data augmentation. BoS methods slightly increase inference cost but significantly improve accuracy, including receptive field expansion, feature fusion, and post-processing techniques. YOLOv4 incorporates a modified Darknet-53 architecture with partial inter-phase connections (CSPNet) and Mish activation functionality as its backbone [8]. The neck has a modified version of the Spatial Pyramid Pool (SPP), Multi-Scale Prediction, Path Aggregation Network (PANet), and Spatial Attention Module (SAM) [9] [10] [11] [12]. Sensor head policy uses anchor. [13] The CSPDarknet53-PANet-SPP model optimizes the calculation while maintaining accuracy. Training improvements include layer augmentation, DropBlock regularization, layer label smoothing, CIoU loss, and small batch normalization (CmBN). Self-adversarial training (SAT) improved strength. Genetic algorithms and optimized hyper parameters of the cosine are annealing planner. YOLOv4 achieved an AP of 43.5% and an AP50 of 65.7% on the MS COCO test-dev 2017 dataset, demonstrating a significant performance increase [14].

E. *YOLOv5*

YOLOv5, introduced in 2020, maintained by Ultralytics [15], represents the latest evolution of the YOLO

(You Only Look Once) family. Notably, although YOLOv5 was launched without scientific articles, it has been widely adopted thanks to its strong performance and user-friendly approach. YOLOv5 is built on the improvements of YOLOv4, with the special feature of being developed on PyTorch, making it more accessible to users. The YOLOv5 framework provides five scalable versions (YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l and YOLOv5x), meeting different needs and resource constraints. Ultralytics actively maintains and supports YOLOv5, providing integrations for labelling, training, and deployment, as well as mobile versions for iOS and Android. In terms of performance, [16] YOLOv5x achieves an outstanding average precision (AP) of 50.7% in the 2017 test split of the MS COCO dataset with an image size of 640 pixels [14]. It excels in real-time applications, delivering up to 200 frames per second from a highly configurable GPU with a batch size of 32. Additionally, by using input sizes larger than 1536 pixels, YOLOv5 pushes Its AP is up 55.8%, highlighting his versatility and ability.

F. *YOLOv6*

YOLOv6 was released by Meituan Vision AI department on ArXiv in September 2022. It provides different models with different sizes for industrial applications. YOLOv6 used an anchorless detector. The main new features of this model are: [17]The new RepVGG-based backbone called EfficiencyRep uses greater parallelism than previous YOLO backbones. For the neck, they use PAN enhanced with RepBlocks or CSPStackRep blocks for larger models [17]. And inspired by YOLOX, they developed an effective decoupling head. Label using task-appropriate learning methods introduced in TOOD. They used VariFocal loss classifier and SIOU/GIoU regression loss of self-distillation strategy for regression and classification tasks. Quantification scheme for detection using RepOptimizer and per-channel distillation for faster detection [18] . YOLOv6 results evaluated on MS COCO 2017 Developer Test Dataset, YOLOv6-L achieved 52.5% AP and 70% AP50 at ~50 FPS on NVIDIA Tesla T4.

G. *YOLOv7*

YOLOv7, introduced in July 2022 by the creators of YOLOv4 and YOLOR, sets new standards in object detection speed and accuracy, from 5 to 160 FPS. Trained exclusively on the MS COCO dataset without a pre-trained framework, YOLOv7 combines architectural enhancements and a freeware suite to improve accuracy without compromising speed deductive. Key architectural changes include the introduction of Extensible Efficient Layer Aggregation Networks (E-ELAN), gradient path optimization for deep models, and model scaling for on concatenation, ensuring balanced adjustment of properties during scaling. [19] Interesting features in YOLOv7 include planned re-parameterized convolution (RepConvN), coarse label assignment to sub-heads, batch normalization in transformation activation, YOLOR-inspired tacit knowledge and an exponential moving average for the final inference model. When tested on the 2017 developer testbed of the MS COCO dataset, [18] YOLOv7-E6 achieved an impressive AP of 55.9% and AP50 of 73.5% with an input size of 1280 pixels, yielding Fast 50 FPS performance on NVIDIA V100.

III. METHODOLOGY

In this section, we describe our methodology. Our main goal is to identify recyclable waste for reuse. To achieve this, we used YOLOv8, a model that does not rely on anchor boxes. We randomly select batches of labelled training images from WaRP Dataset, start the training process, and extract features from these images. These characteristics are then used to detect and classify recyclable materials.

A. *Dataset*

WaRP, short for Waste Recycling Plant Dataset, is a carefully curated collection of labelled images taken at an industrial waste sorting plant. This dataset is a valuable asset adapted for machine learning and computer vision applications, with a special focus on waste classification and recycling. What sets WaRP apart from many other datasets is its unique features and broad coverage of waste categories.

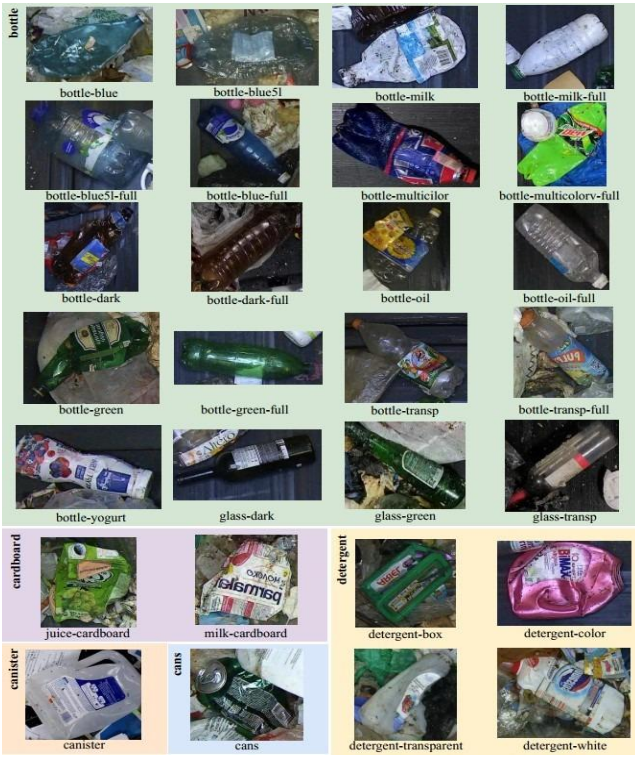

Fig 1. Distinct 28 object classes available in dataset

[20] WaRP is thoughtfully divided into five main sections: Bottles, Carton, Detergent, Canisters and Cans, divided into 28 separate categories. Among them there are 17 categories of plastic bottles marked with the prefix "bottle" and three glass bottle types with the prefix "glass". Cardboard is classified into two categories and four categories include detergents and cans and cans. Some items in the dataset are marked with the suffix "-full" to indicate that these bottles are filled with air, distinguishing them from flat bottles.

Fig 2. Class Object organisation in WaRP-D dataset

A characteristic of the WaRP dataset is its realism and representation of challenging real-world scenarios. The images in this dataset accurately represent conditions where objects often overlap, change significantly, or encounter difficult lighting conditions. This realistic aspect is important for training and rigorous evaluation of machine learning models, especially those designed to classify litter in less-than-ideal environments. [20] Warp's main component, WaRP-D, contains a significant number of images for training and validation. It provides 2452 training images to build robust waste sorting models and 522 additional validation images for performance evaluation. Each Warp image has a high-definition resolution of 1920x1080 pixels, providing a detailed visual representation of waste at recycling sites. This high resolution makes the dataset suitable for various computer vision and deep learning applications, especially those focused on accurate identification and efficient waste sorting.

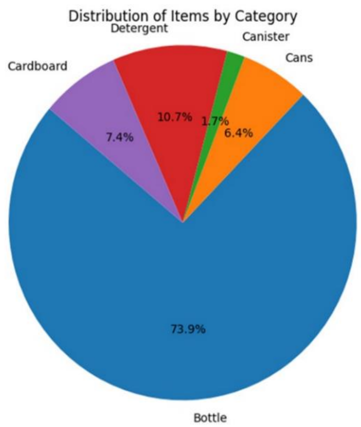

Fig 3. Ratio of different classes in WaRP Dataset

Category	Count
Bottle	1810
Cans	156
Canister	41
Detergent	261
Cardboard	180

Fig 4. Category wise count for different classes in WaRP Dataset

B. *Proposed Model*

We have implemented YOLOv8 model, introduced by Ultralytics on January 10, 2023. Building on the success of the YOLO model series, YOLOv8 stands out as an advanced model, delivering higher accuracy and detection speed than its predecessors, such as YOLOv5 and YOLOv7. Turning to the network architecture, YOLOv8's backbone closely resembles YOLOv5, with CSP replacing C3 modules. [21] [22] The popular SPPF module remains at the end of the backbone, ensuring accuracy across different scales. In the Neck section, PAN-FPN feature fusion effectively utilizes features from various scale layers. The Neck module incorporates multiple C2f modules and up-sampling layers alongside a decoupled head structure, achieving heightened accuracy.

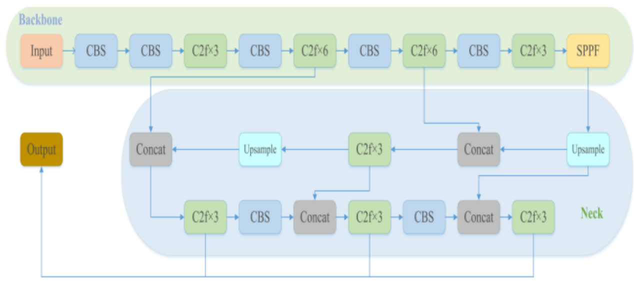

Fig 5. YOLOv8 architecture for object detection

Backbone:

The core of YOLOv8 is anchored in the modified CSPDarknet53 architecture and uses a multi-scale approach by scaling the input features into five distinct scales labelled B1 to B5. The original Cross Stage Partial (CSP) module is replaced by the C2f module, which features pass-through connections to improve information flow while maintaining a lightweight structure. The CBS module performs a convolution operation on the input data, followed by batch normalization and SiLU activation to produce the output. To scale output adaptively, the backbone uses the Spatial Pyramid Pooling Fast (SPPF) module. SPPF reduces computational cost and latency by sequentially connecting three max pooling layers.

Neck:

Inspired by PANet, [11] YOLOv8 integrates the PAN-FPN structure in the neck, optimizing feature fusion and localization. Notably, YOLOv8 streamlines the PAN structure by eliminating post-sampling convolution operations, thereby achieving model efficiency without compromising performance. The PAN-FPN configuration exploits two different feature scales, P4-P5 and N4-N5, in the PAN and FPN structures, respectively. By combining top-down and bottom-up approaches, PAN-FPN ensures feature diversity and completeness by combining deep semantic and shallow location information.

Head:

YOLOv8's detection component uses a split-head structure, with separate branches for object classification and prediction bounding box regression. Different loss functions are applied to these branches; with binary cross-entropy loss (BCE loss) used for classification and distributed focus loss (DFL) as well as CIOU for bounding box regression [18]. This decoupled design improves detection accuracy and speeds up model convergence. The model uses a Task Specifier to dynamically assign samples.

YOLOv8, an evolution of this technology, builds upon the successes of prior real-time object detectors. Drawing inspiration from YOLOv5, YOLOv8 integrates the CSP (Cross Stage Partial) concept, PAN-FPN feature fusion method, and the SPPF (Spatial Pyramid Pooling Fast) module [23] [24]. Its paramount innovation lies in introducing a cutting-edge State-of-the-Art (SOTA) model. This includes the integration of object detection networks operating at resolutions of P5 640 and P6 1280, alongside the YOLACT instance segmentation model. While preserving the foundational idea of YOLOv5, it adopts a C2f module inspired by YOLOv7's ELAN structure. [22] [25] In terms of loss functions, YOLOv8 employs BCE Loss for classification and introduces the CIOU Loss for regression. Additionally, it incorporates the DFL (Distribution Focal Loss) and VFL (Variable Focal Loss) mechanisms, enhancing focus on target locations and optimizing probability density near object positions. Let the network quickly focus on the distribution of the location close to the target location, and make the probability density near the location as large as possible, as shown in formula (1). s_i is the output of sigmoid for the network, y_i and y_i+1 are interval orders, y is label. Compared to the previous YOLO algorithm, YOLOv8 is very extensible. It is a framework that can support previous versions of YOLO, and can switch between different versions, so it is easy to compare the performance of different versions.

$$DFL(s_i, s_i + 1) = -((y_i + 1 - y)*\log(s_i) + (y - y_i)*\log(s_i + 1)) \quad \textbf{(1)}$$

Notably, YOLOv8 transitions from Anchor-Based to Anchor-Free detection and adopts a dynamic Task-Aligned Assigner strategy. [22] [25] This strategy calculates alignment degrees for each instance based on a formula involving classification scores, IOU values, and weighted hyper parameters. It calculates the alignment degree of Anchor-level for each instance using Equation (2), s is the classification score, u is the IOU value, α and β are the weight hyper parameters. It selects m anchors with the maximum value (t) in each instance as positive samples, and selects the other anchors as negative samples, and then trains through the loss function. The outcome of these advancements is a YOLO model that outperforms YOLOv5 by approximately 1% in terms of accuracy, solidifying its position as one of the most precise object detectors available.

$$t = s.\alpha * u.\beta \quad (2)$$

Key to YOLOv8's appeal is its adaptability—it works seamlessly with various YOLO versions, making it a valuable tool for performance evaluation in the field of YOLO-based research.

C. Model Implementation

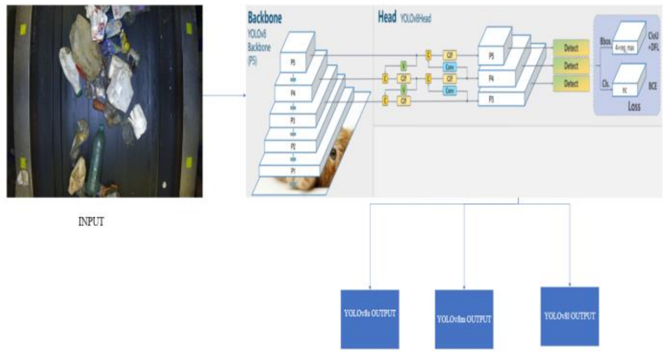

Fig 6. Waste detection process in YOLOv8

Fig 7. YOLOv8s OUTPUT

Fig 8. YOLOv8m OUTPUT

Fig 9. YOLOv8L OUTPUT

IV. Result Discussion

In a comprehensive experiment, we evaluated the performance of [16] YOLOv5 and YOLOv8 models on the same dataset, and interestingly, both models yielded similar results during training. However, it's noteworthy that YOLOv8 exhibited a more competitive performance overall. For our YOLOv5 experiments, we specifically utilized two variants: YOLOv5s and YOLOv5m. In contrast, for the YOLOv8 experiments, we employed a more extensive approach, encompassing three different model sizes: YOLOv8s, YOLOv8m, and YOLOv8l. This encompassed a spectrum of model weights, ranging from smaller to larger configurations. To rigorously assess the models, we compared their precision values and calculated the mean Average Precision (mAP) across various epochs within the same experiment. This meticulous evaluation allowed us to draw meaningful conclusions about the performance of these models in object detection tasks, taking into consideration both precision at individual time points and the overall quality of object detection across all epochs.

A. *YOLOv8s*

We start the training process using the YOLOv8s model. However, due to the lack of observable improvements over the past 20 epochs, we made the decision to end training early, as shown in Figure 7. For your information, we have compiled a detailed breakdown of the precision, recall, mAP50, and mAP50-95 values in Table 1, focusing on a specific set of 28 feature classes is the focus of this phase of training. These values provide a comprehensive evaluation of the performance of the YOLOv8 model in these classes. In a broader context, the YOLOv8s model demonstrated an impressive overall average accuracy (AP) of 51.20%. This figure summarizes the model's cumulative performance across all object classes in our dataset, highlighting its mastery of object detection tasks.

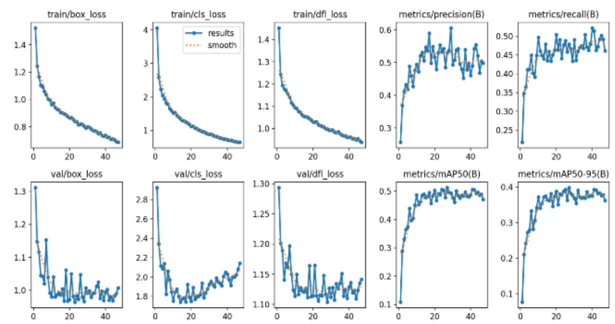

Fig 10. Training results for YOLOv8s model.

Table 1. Modal Summary which describe precision, recall and mean Average Precisions (mAP) during the training phase on YOLOv8s model.

Class	Precision	Recall	mAP50	mAP50-95
bottle-blue	0.406	0.539	0.488	0.375
bottle-green	0.692	0.605	0.694	0.542
bottle-dark	0.793	0.678	0.787	0.608
bottle-milk	0.506	0.597	0.53	0.417
bottle-transp	0.506	0.481	0.493	0.366
bottle-multicolor	0.442	0.25	0.189	0.163
bottle-yogurt	0.595	0.333	0.401	0.315
bottle-oil	0.209	0.137	0.183	0.141
cans	0.581	0.492	0.544	0.385
juice-cardboard	0.383	0.137	0.209	0.15
milk-cardboard	0.31	0.348	0.346	0.273
detergent-color	0.593	0.323	0.398	0.296
detergent-transparent	0.361	0.235	0.25	0.205
detergent-box	0.637	0.643	0.973	0.48
canister	0.438	0.385	0.419	0.369
bottle-blue-full	0.588	0.667	0.723	0.579
bottle-transp-full	0.662	0.683	0.719	0.596
bottle-dark-full	0.616	0.786	0.786	0.675
bottle-green-full	0.609	0.815	0.778	0.648
bottle-multicolorv-full	0.69	0.417	0.592	0.468
bottle-milk-full	0.489	0.727	0.63	0.516
bottle-oil-full	0.343	0.1	0.148	0.129
detergent-white	0.514	0.397	0.449	0.363
bottle-blue5l	0.674	0.63	0.633	0.495
bottle-blue5l-full	0.661	0.651	0.692	0.617
glass-transp	0.518	0.359	0.338	0.223
glass-dark	0.604	0.375	0.564	0.306
glass-green	0.604	0.548	0.669	0.443

B. *YOLOv8m*

We started the training process with the YOLOv8m model, achieving notable mAP50 values between 0.55 and 0.58 over 100 epochs, as shown in Figure 8. In Table 2 you will find full details on the precision, recall, mAP50 and mAP50-95 values, with a specific focus on the 28 feature classes that were the main focus during the training phases. These values provide a detailed evaluation of the performance of the YOLOv8m model in these classes. From a broader perspective, the YOLOv8m model exhibits an impressive overall average accuracy (AP) of 55.47%. This metric covers the model's overall performance across all object classes in our dataset, highlighting the model's excellence in object detection tasks.

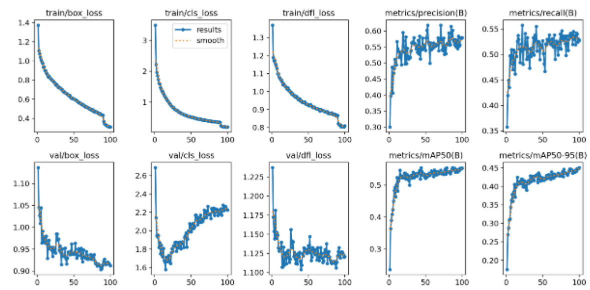

Fig 11. Training results for YOLOv8m model.

Table 2. Modal Summary which describes precision, recall and mean Average Precisions(mAP) during the training phase on YOLOv8m model.

Class	Precision	Recall	mAP50	mAP50-95
bottle-blue	0.49	0.47	0.488	0.395
bottle-green	0.715	0.663	0.725	0.576
bottle-dark	0.794	0.713	0.811	0.655
bottle-milk	0.498	0.4	0.494	0.418
bottle-transp	0.588	0.498	0.531	0.429
bottle-multicolor	0.453	0.292	0.292	0.265
bottle-yogurt	0.536	0.44	0.413	0.345
bottle-oil	0.38	0.275	0.275	0.225
cans	0.697	0.464	0.595	0.43
juice-cardboard	0.495	0.32	0.294	0.234
milk-cardboard	0.433	0.405	0.39	0.325
detergent-color	0.585	0.452	0.464	0.385
detergent-transparent	0.404	0.203	0.235	0.188
detergent-box	0.7174	0.714	0.764	0.644
canister	0.529	0.538	0.576	0.531
bottle-blue-full	0.608	0.667	0.71	0.548
bottle-transp-full	0.637	0.762	0.738	0.622
bottle-dark-full	0.592	0.75	0.793	0.696
bottle-green-full	0.685	0.881	0.817	0.678
bottle-ulticolorv-full	0.644	0.667	0.711	0.541
bottle-milk-full	0.625	0.759	0.629	0.544
bottle-oil-full	0.246	0.2	0.194	0.178
detergent-white	0.538	0.5	0.536	0.442
bottle-blue5l	0.62	0.556	0.606	0.509
bottle-blue5l-full	0.436	0.733	0.708	0.616
glass-transp	0.596	0.313	0.372	0.274
glass-dark	0.842	0.499	0.665	0.477
glass-green	0.794	0.613	0.673	0.455

C. *YOLOv8l*

We started the training process with the YOLOv8l model, achieving an exceptionally high mAP50 value, around 0.6, over 80 epochs, as Figure 9 illustrates. For your information, we have meticulously compiled a detailed breakdown of precision, recall, mAP50 and mAP50-95 values in Table 3. This table focuses specifically on 28 object classes is the main focus of this training phase, providing an in-depth evaluation of the performance of the YOLOv8l model in these classes. By zooming out to capture a broader context, the YOLOv8l model demonstrated an impressive overall average accuracy (AP) of 59.58%. This overall metric encapsulates the model's overall performance across all object classes in our dataset, confirming the model's excellence in the field of object detection.

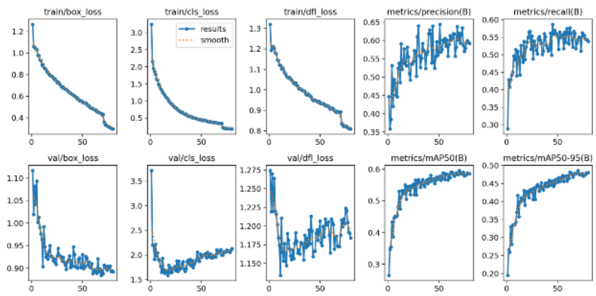

Fig 12. Training results for YOLOv8l model.

Table 3. Modal Summary which describes precision, recall and mean Average Precisions(mAP) during the training phase on YOLOv8l model.

Class	Precision	Recall	mAP50	mAP50-95
bottle-blue	0.52	0.539	0.572	0.464
bottle-green	0.672	0.644	0.749	0.611
bottle-dark	0.841	0.735	0.811	0.655
bottle-milk	0.594	0.433	0.546	0.463
bottle-transp	0.655	0.509	0.568	0.453
bottle-multicolor	0.413	0.208	0.237	0.208
bottle-yogurt	0.758	0.422	0.538	0.437
bottle-oil	0.491	0.353	0.349	0.279
cans	0.743	0.444	0.622	0.455
juice-cardboard	0.512	0.28	0.39	0.292
milk-cardboard	0.444	0.424	0.377	0.302
detergent-color	0.478	0.355	0.451	0.376
detergent-transparent	0.503	0.338	0.326	0.268
detergent-box	0.778	0.725	0.865	0.71
canister	0.467	0.615	0.576	0.501
bottle-blue-full	0.716	0.698	0.743	0.624
bottle-transp-full	0.639	0.762	0.767	0.655
bottle-dark-full	0.661	0.587	0.85	0.735
bottle-green-full	0.714	0.881	0.836	0.724
bottle-multicolorv-full	0.827	0.792	0.852	0.686
bottle-milk-full	0.423	0.653	0.615	0.522
bottle-oil-full	0.382	0.2	0.212	0.194
detergent-white	0.668	0.546	0.548	0.451
bottle-blue5l	0.607	0.631	0.639	0.523
bottle-blue5l-full	0.488	0.8	0.804	0.715
glass-transp	0.644	0.303	0.384	0.276
glass-dark	0.876	0.5	0.73	0.508
glass-green	0.825	0.61	0.732	0.528

D. *Result comparison*

YOLOv8s: This is the smallest variant of YOLOv8, with a moderate mAP0.5 of 0.512 and a relatively fast average inference time of 3.00ms. It strikes a balance between accuracy and speed. YOLOv8m is the medium-sized variant, YOLOv8m, achieves a higher mAP0.5 of 0.5547 but requires a slightly longer average inference time of 6.00ms. YOLOv8l is the largest and most complex variant, providing the highest mAP0.5 of 0.5958. However, this comes at the cost of a longer average inference time of 9.20ms.

Table 4. comparison among YOLOv5 and YOLOv8 models with different weights in terms of best training iterations (epochs) and mean average precision (mAP) with their average inference time.

Model	Weights	epochs	mAP0.5	mAP@0.5:0.95	average inference time(millisecond)
YOLOv8	YOLOv8s	48	0.512	0.3983	3.00
	YOLOv8m	100	0.5547	0.4508	6.00
	YOLOv8l	80	0.5958	0.4860	9.20

The scatter plot aids to visualize and compare the mAP50 (mean average precision at threshold 50 IoU) values for three different variants of the YOLOv8 model: YOLOv8, YOLOv8m and YOLOv8l. This allows a visual assessment of the model's performance with respect to these variations.

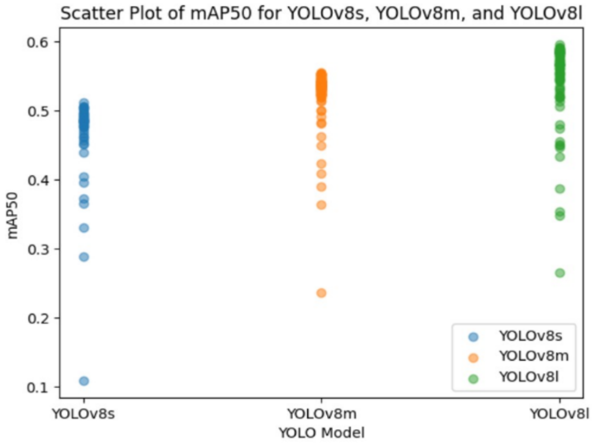

Fig 13. Relationship among 3 distinct YOLOv8 weights in terms of mAP50

V. CONCLUSION

This article presents a new application of the YOLOv8 algorithm, to improve the ability to intelligently classify and manage urban waste. The neural network was trained on a custom dataset of 2,452 images from a waste recycling plant, focusing on detecting 28 different types of waste. The test results illustrate the effectiveness of our method of classifying waste into five distinct groups: Bottles, cans, cans, cardboard, and detergents. In particular, our system allows for almost real-time waste detection. Comparative evaluations between the YOLOv8s, YOLOv8m and YOLOv8l models highlight the effectiveness of YOLOv8 in waste classification. However, due to limitations in the size of our training dataset, the YOLOv8l and YOLOv8m models did not yield significantly improved results compared to YOLOv8s. Therefore, future research will require larger data sets to improve the accuracy and precision of detection. In conclusion, our comparative analysis of the YOLOv8s, YOLOv8m, and YOLOv8l models quantifies the trade-off between precision and speed. Additionally, this research recognizes the complexity of detecting junk images, especially when objects are made from multiple materials or may include components from other layers. Although our approach focuses on parent classes and tangible categories, it paves the way for further research to improve waste classification based on material properties. Additionally, our waste sorting strategy shows promise in improving waste disposal and recycling practices. Future work will focus on optimizing prediction and probabilistic results for other real-world wastes other than former 28 classes.

REFERENCES

[1] S. a. Y. Kumar, "A novel yolov3 algorithm-based deep learning approach for waste segregation: Towards smart waste management," MDPI, p. 14, 2020.

[2] Y. a. Z. Z. a. H. You, Imagenet training in minutes, Proceedings of the 47th International Conference on Parallel Processing, 2018.

[3] S. a. C. J. a. A. L. a. B. J. Vicente, Reconstructing pascal voc, Proceedings of the IEEE conference on computer vision and pattern recognition, 2014.

[4] M. Shenoda, "Real-time Object Detection: YOLOv1 Re-Implementation in PyTorch," arXiv preprint arXiv:2305.17786, p. 4, 2023.

[5] Z. A. a. S. H. a. A. Choudhry, "DarkNet-19 Based Intelligent Diagnostic System for Ocular Diseases," Iranian Journal of Science and Technology, Transactions of Electrical Engineering, pp. 959-970, 2022.

[6] J. a. F. A. Redmon, YOLO9000: better, faster, stronger, 7263-7271: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017.

[7] J. a. F. A. Redmon, "Yolov3: An incremental improvement," arXiv preprint arXiv:1804.02767, p. 6, 2018.

[8] D. a. R. U. Pathak, "Content-based image retrieval using group normalized-inception-darknet-53," International Journal of Multimedia Information Retrieval, pp. 155-170, 2021.

[9] K. a. Z. X. a. R. S. a. S. J. He, "Spatial pyramid pooling in deep convolutional networks for visual recognition," IEEE transactions on pattern analysis and machine intelligence, pp. 1904-1916, 2015.

[10] G. a. Z. Y. a. S. Sun, "Multi-scale prediction of the effective chloride diffusion coefficient of concrete," Construction and Building Materials, pp. 3820-3831, 2011.

[11] S. a. Q. L. a. Q. H. a. S. J. a. J. J. Liu, Path aggregation network for instance segmentation, Proceedings of the IEEE conference on computer vision and pattern recognition, 2018.

[12] Y. a. F. M. a. W. N. Zhang, "Channel-spatial attention network for fewshot classification," Plos one, p. 16, 2019.

[13] A. a. W. C.-Y. a. L. H.-Y. M. Bochkovskiy, "Yolov4: Optimal speed and accuracy of object detection," arXiv preprint arXiv:2004.10934, p. 17, 2020.

[14] T.-Y. a. M. M. a. B. Lin, Microsoft coco: Common objects in context, Computer Vision--ECCV 2014: 13th European Conference, 2014.

[15] G. a. C. A. a. S. Jocher, "ultralytics/yolov5: v6. 1-TensorRT, TensorFlow edge TPU and OpenVINO export and inference," Zenodo, 2022.

[16] S. a. L. G. a. J. Z. a. H. L. Tan, Improved YOLOv5 network model and application in safety helmet detection, 2021 IEEE International Conference on Intelligence and Safety for Robotics (ISR), 2021.

[17] X. a. Z. X. a. M. Ding, Repvgg: Making vgg-style convnets great again, Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021.

[18] M. Hussain, "YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection," MDPI, p. 677, 2023.

[19] G. a. M. G. a. H. M. Yasmine, Overview of single-stage object detection models: from Yolov1 to Yolov7, 2023 International Wireless Communications and Mobile Computing (IWCMC), 2023.

[20] N. Z. A. S. R. F. M. K. V. K. Dmitry Yudin, WaRP - Waste Recycling Plant Dataset, Kaggle, 2023.

[21] Z. a. L. L. a. S. P. Wang, "Smoking behavior detection algorithm based on YOLOv8-MNC," Frontiers in Computational Neuroscience, p. 14, 2023.

[22] B. a. N. M. a. Y. W. Q. Xiao, "Fruit ripeness identification using YOLOv8 model," Multimedia Tools and Applications, p. 18, 2023.

[23] C. a. C. R.-C. Dewi, "Automatic medical face mask detection based on cross-stage partial network to Combat covid-19," Big Data and Cognitive Computing, p. 106, 2022.

[24] K. a. Z. X. a. R. S. a. S. J. He, "Spatial pyramid pooling in deep convolutional networks for visual recognition," IEEE transactions on pattern analysis and machine intelligence, pp. 1904-1916, 2015.

[25] H. a. D. X. a. G. Lou, "DC-YOLOv8: Small-Size Object Detection Algorithm Based on Camera Sensor," MDPI, p. 2323, 2023.